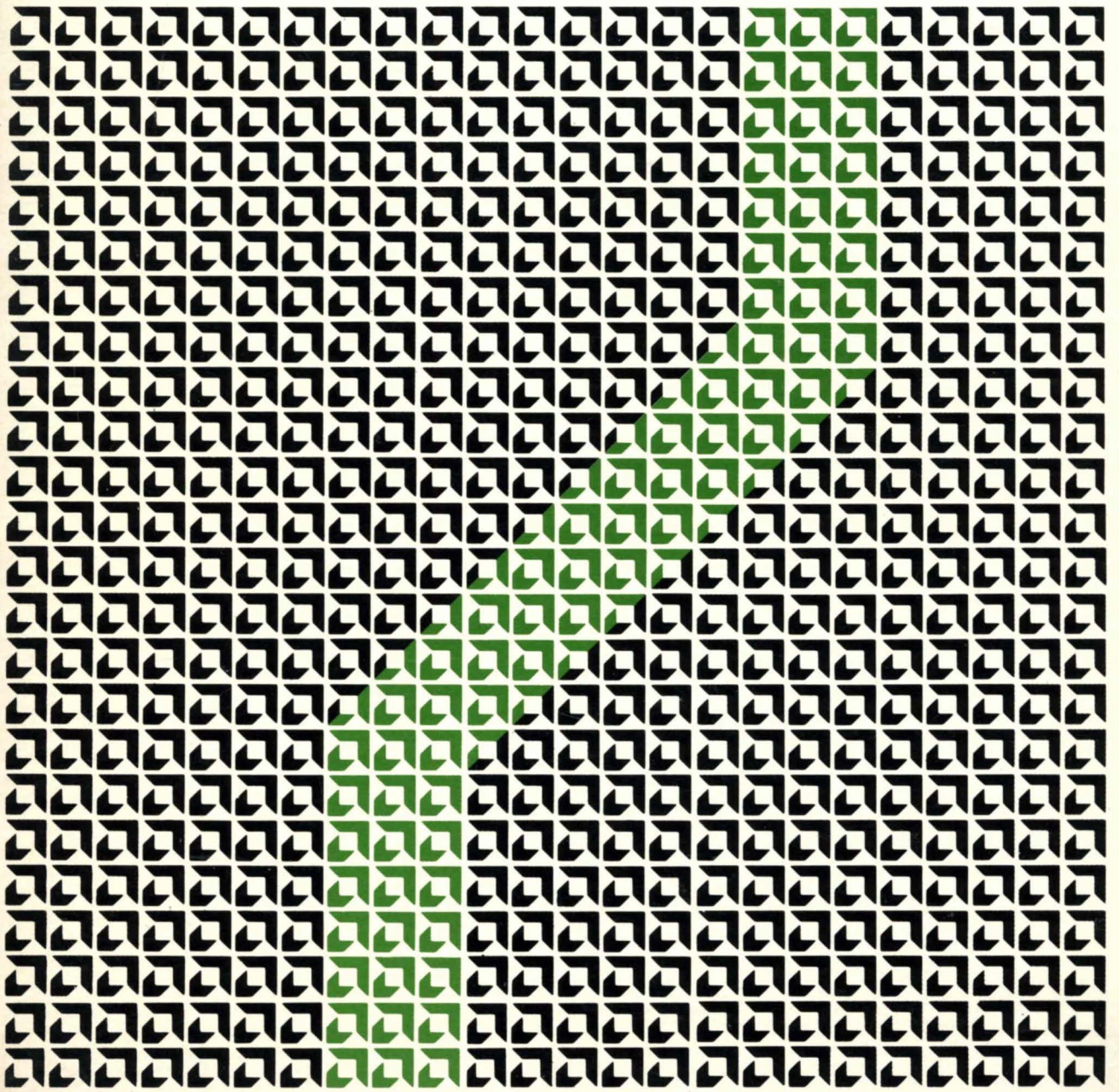
Advanced Micro Devices

The Designers' Guide '80





Advanced Micro Devices

The Designers' Guide '80

Copyright © 1980 by Advanced Micro Devices, Inc.

Advanced Micro Devices reserves the right to make changes in its products without notice in order to improve design or performance characteristics. The company assumes no responsibility for the use of any circuits described herein.

AM-PUB085

Printed in U.S.A.

The Designers' Choice

Advanced Micro Devices, the newest giant among the major semiconductor manufacturers, offers a product portfolio numbering over 800 complex, monolithic, integrated circuits with emphasis on microprocessors, memories and their related peripheral circuits.

Included herein are selected data sheets and application notes on several recently introduced key AMD® circuits. And, a full set of various indexes and cross reference charts.

Advanced Micro Devices, the Designers' Choice, the company with the commitment – A Commitment to Excellence – where all devices are manufactured in compliance with MIL-STD-883, MIL-M-38510 and MIL-Q-9858.

TABLE OF CONTENTS

PRODUCT ASSURANCE 1-1

PRODUCT GUIDES AND CROSS REFERENCE:

- Interface Circuits 2-1
- Alpha-Numeric Cross Reference 2-2
- Selection Guide 2-4
- Linear Circuits 2-7
- Bipolar Logic Circuits 2-8
- Advanced Micro Devices Schottky and Low-Power Schottky MSI Functional Selector Guide 2-9

MICROPROCESSORS:

- Am2901C 3-1
- Am2903 • Am29203 3-3
- Am2903A 3-31
- Am29705A 3-32
- Am8041 3-33
- Am8048/8035 3-34
- Am8085A/Am8085A-2/Am9085ADM 3-35
- AmZ8001 3-46
- AmZ8002 3-68

MEMORIES:

- Am2716 4-1
- Am27S26 • Am27S27 4-7
- Am27S28 • Am27S29 4-15
- Am27S30 • Am27S31 4-19
- Am27S32 • Am27S33 4-23
- Am27S180 • Am27S181 4-27
- Am9016 4-31
- Am9044 • Am9244 4-40
- Am9114 • Am9124 4-44
- Am9218/8316E 4-48
- Am9232 • Am9233 4-50
- Am93412 • Am93422 4-52
- Am93415 • Am93425 4-56

MEMORY MANAGEMENT PRODUCTS:

- Am2960 5-1
- Am2961 • Am2962 5-17
- Am2964 5-18
- Am2965 • Am2966 5-19
- AmZ8163 5-25

TABLE OF CONTENTS (Cont.)

HIGH-PERFORMANCE CONTROLLER PRODUCTS:
- Am29116 6-1
- Am2940 6-2
- Am2942 6-10
- Am2950 • Am2951 6-20

HIGH-PERFORMANCE MULTIPLIERS:
- Am25S557 • Am25S558 7-1

AmZ8000 PERIPHERAL/SUPPORT PRODUCTS:
- AmZ8127 8-3

HIGH-PERFORMANCE MICROPROCESSOR SUPPORT PRODUCTS (Am2900 FAMILY):
- Am2925 9-1
- Am2927 • Am2928 9-2

SUPPORT AND PERIPHERAL PRODUCTS:
- Am6012 10-1
- Am6080 10-9
- Am6081 10-16
- Am9511A 10-27
- Am9512 10-52
- Am9513 10-71
- Am9517A 10-87
- Am9519 10-102

APPLICATIONS:
- The Am9517A Multimode Direct Memory Access Controller 11-1
- Designing Interrupt Systems with the Am9519 Universal Interrupt Controller 11-22

ADVANCED MICRO COMPUTERS BOARDS AND DEVELOPMENT SYSTEMS:
- Am95/4005 12-1
- Am95/6012 12-2
- Am95/5032 12-3
- Am95/6110 12-4
- Am95/6440 and Am95/6448 12-5
- Am96/1000 12-6
- AMC's MACRO8000 12-7
- AMC's PASCAL Compiler 12-8
- AmSYS™8/8 Microcomputer Development System 12-9
- AmZ8000 Evaluation Board 12-20

PRODUCT ASSURANCE
MIL-M-38510 • MIL-STD-883

The product assurance program at Advanced Micro Devices defines manufacturing flow, establishes standards and controls, and confirms the product quality at critical points. Standardization under this program assures that all products meet military and government agency specifications for reliable ground applications. Further screening for users desiring flight hardware and other higher reliability classes is simplified because starting product meets all initial requirements for high-reliability parts.

The quality standards and screening methods of this program are equally valuable for commercial parts where equipment must perform reliably with minimum field service.

Two military documents provide the foundation for this program. They are:

MIL-M-38510 – General Specification for Microcircuits
MIL-STD-883 – Test Methods and Procedures for Microelectronics

MIL-M-38510 describes design, processing and assembly workmanship guidelines for military and space-grade integrated circuits. All circuits manufactured by Advanced Micro Devices for full temperature range (−55°C to +125°C) operation meet these quality requirements of MIL-M-38510.

MIL-STD-883 defines detail testing and inspection methods for integrated circuits. Three of the methods are quality and processing standards directly related to product assurance:

Test Method 2010 defines the visual inspection of integrated circuits before sealing. By confirming fabrication and assembly quality, inspection to this standard assures the user of reliable circuits in long-term field applications. Standard inspection at Advanced Micro Devices includes all the requirements of the latest revision of Method 2010, condition B.

Test Method 5004 defines three reliability classes of parts. All must receive certain basic inspection, preconditioning and screening stresses. The classes are:

Class C – Used where replacement can be readily accomplished. Screening steps are given in the AMD processing flow chart.

Class B – Used where maintenance is difficult or expensive and where reliability is vital. Devices are upgraded from Class C to Class B by 160-hour burn-in at 125°C followed by more extensive electrical measurements. All other screening requirements are the same.

Class S – Used where replacement is extremely difficult and reliability is imperative. Class S screening selects extra reliability parts by expanded visual and X-ray inspection, further burn-in, and tighter sampling inspection.

All hermetically sealed integrated circuits (military and commercial) manufactured by Advanced Micro Devices are screened to MIL-STD-883, Class C. Molded integrated circuits receive Class C screening except that centrifuge and hermeticity steps are omitted.

Optional extended processing to MIL-STD-883, Class B is available for all AMD integrated circuits. Parts procured to this screening are marked with a "−B" following the standard part number, except that linear 100, 200 or 300 series are suffixed "/883B".

Test Method 5005 defines qualification and quality conformance procedures. Subgroups, tests and quality levels are given for Group A (electrical), Group B (mechanical quality related to the user's assembly environment), Group C (die related tests) and Group D (package related tests). Group A tests are always performed; Group B, C and D may be specified by the user.

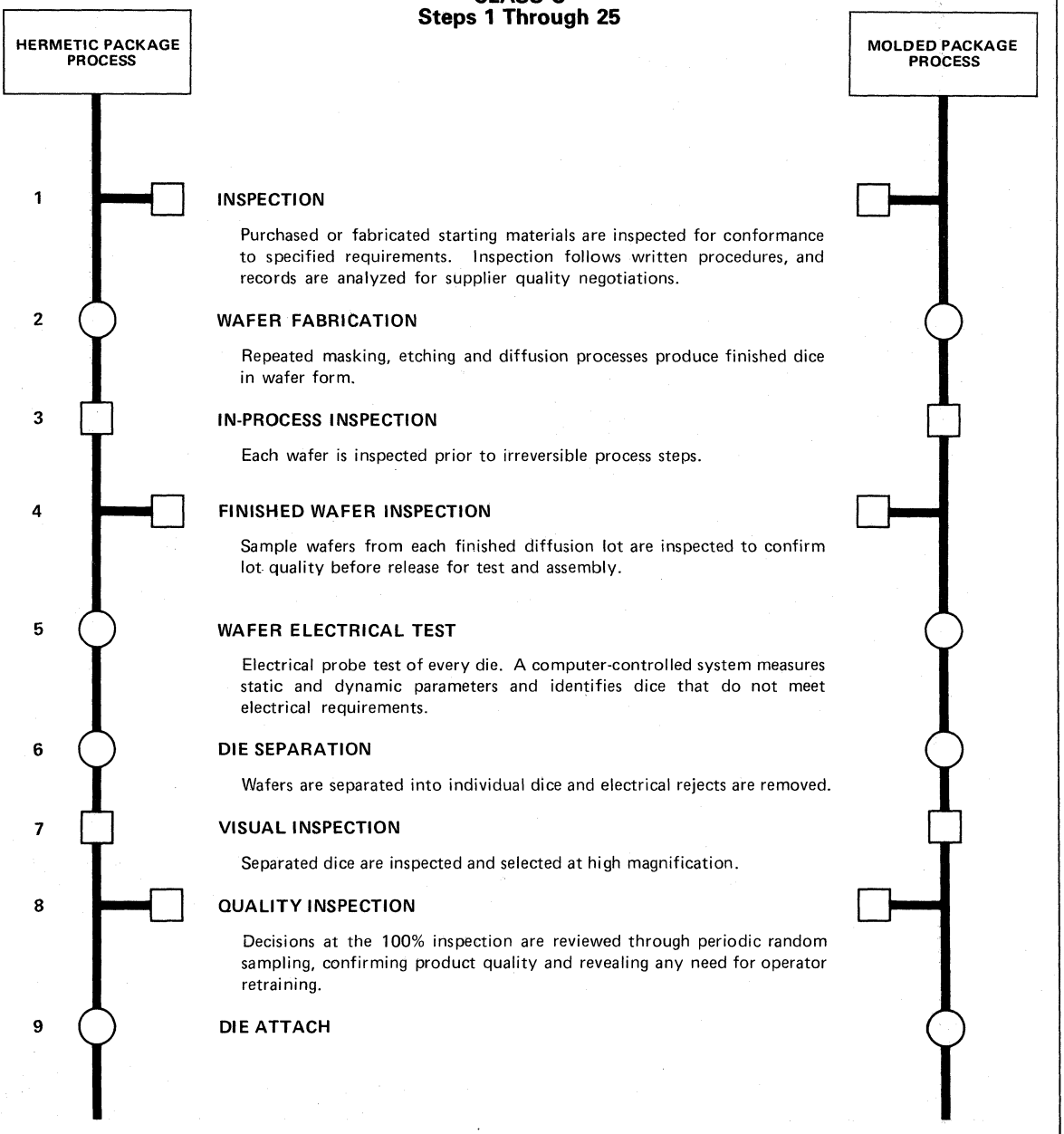
MANUFACTURING, SCREENING AND INSPECTION FOR INTEGRATED CIRCUITS

All integrated circuits are screened to MIL-STD-883, Method 5004, Class C; quality conformance inspection where required is performed to Class B quality levels on either Class B or Class C product.

All full-temperature-range (−55°C to +125°C) circuits are manufactured to the workmanship requirements of MIL-M-38510.

The flow chart identifies processing steps as they relate to MIL-STD-883 and MIL-M-38510.

STANDARD PROCESSING CLASS C Steps 1 Through 25



10 **QUALITY INSPECTION**

Strength of die attachment, position of die and visual quality of eutectic wetting are confirmed periodically by inspecting random samples and push-testing the attached dice.

11 **WIRE BOND**

Hermetic: Aluminum wires, ultrasonic bonding.
Molded: Gold wires, thermocompression bonding.

12 **QUALITY INSPECTION**

Weld strength, bond size and position, wire dress and general workmanship are confirmed periodically by comparing random samples with assembly instructions and quality standards. Bond strength is plotted on statistical control charts, providing early warning of process drifts.

13 **INTERNAL VISUAL INSPECTION**

Assembled but unsealed units are individually inspected at low and high power.

QUALITY STANDARDS:

All devices – MIL-STD-883, Method 2010, Condition B (latest revision). Full temperature devices – MIL-M-38510, Para. 3.7 for workmanship (re-bonding limits).

14 **QUALITY INSPECTION**

Decisions at the 100% inspection are reviewed through periodic random sampling, providing confirmation of product quality and revealing any need for operator retraining.

15a **FINAL SEAL**

(Hermetic devices)

15b **ENCAPSULATE**

(Molded Devices)

16 **HIGH TEMPERATURE STORAGE**

MIL-STD-883, Method 1008, Cond. C: 150°C, 24 hr

17 **TEMPERATURE CYCLE**

MIL-STD-883, Method 1010, Cond. C: −65°C, +150°C, 10 cycles

18 **CENTRIFUGE**

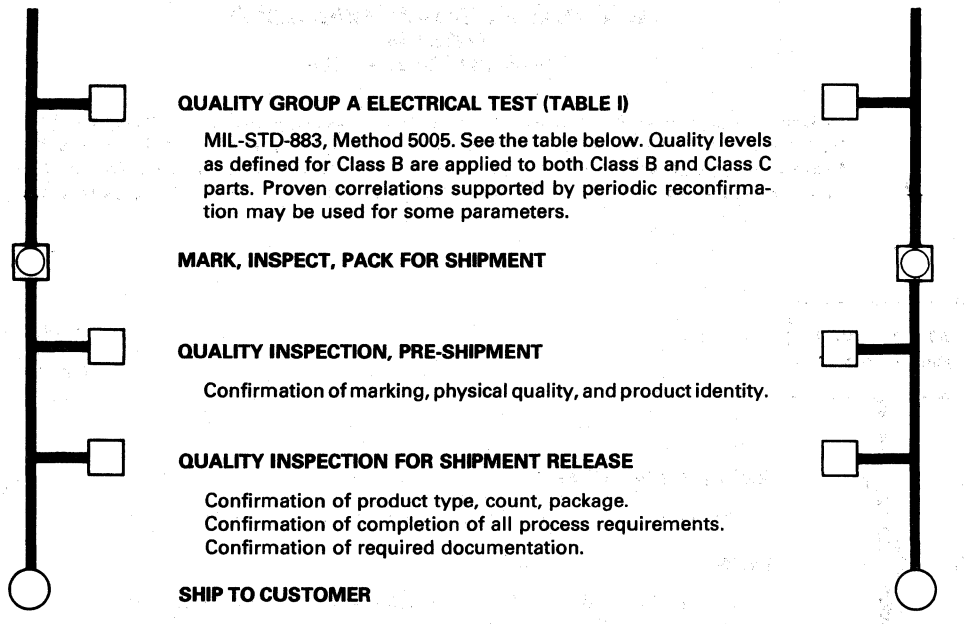
MIL-STD-883, Method 2001, Cond. E: 30,000 G

19 **SEAL (HERMETICITY) TEST**

MIL-STD-883, Method 1014, Cond. A or B: Fine Leak
MIL-STD-883, Method 1014, Cond. C2: Gross Leak

20 **ELECTRICAL TEST**

MIL-STD-883, Method 5004, Para. 3.1.12: Static, dynamic, functional tests at 25°C or in certain products at the most critical extreme temperature to assure accuracy of device selection.



21 **QUALITY GROUP A ELECTRICAL TEST (TABLE I)**

MIL-STD-883, Method 5005. See the table below. Quality levels as defined for Class B are applied to both Class B and Class C parts. Proven correlations supported by periodic reconfirmation may be used for some parameters.

22 **MARK, INSPECT, PACK FOR SHIPMENT**

23 **QUALITY INSPECTION, PRE-SHIPMENT**

Confirmation of marking, physical quality, and product identity.

24 **QUALITY INSPECTION FOR SHIPMENT RELEASE**

Confirmation of product type, count, package.
Confirmation of completion of all process requirements.
Confirmation of required documentation.

25 **SHIP TO CUSTOMER**

This AMD standard product meets screening requirements of MIL-STD-883, Class C.

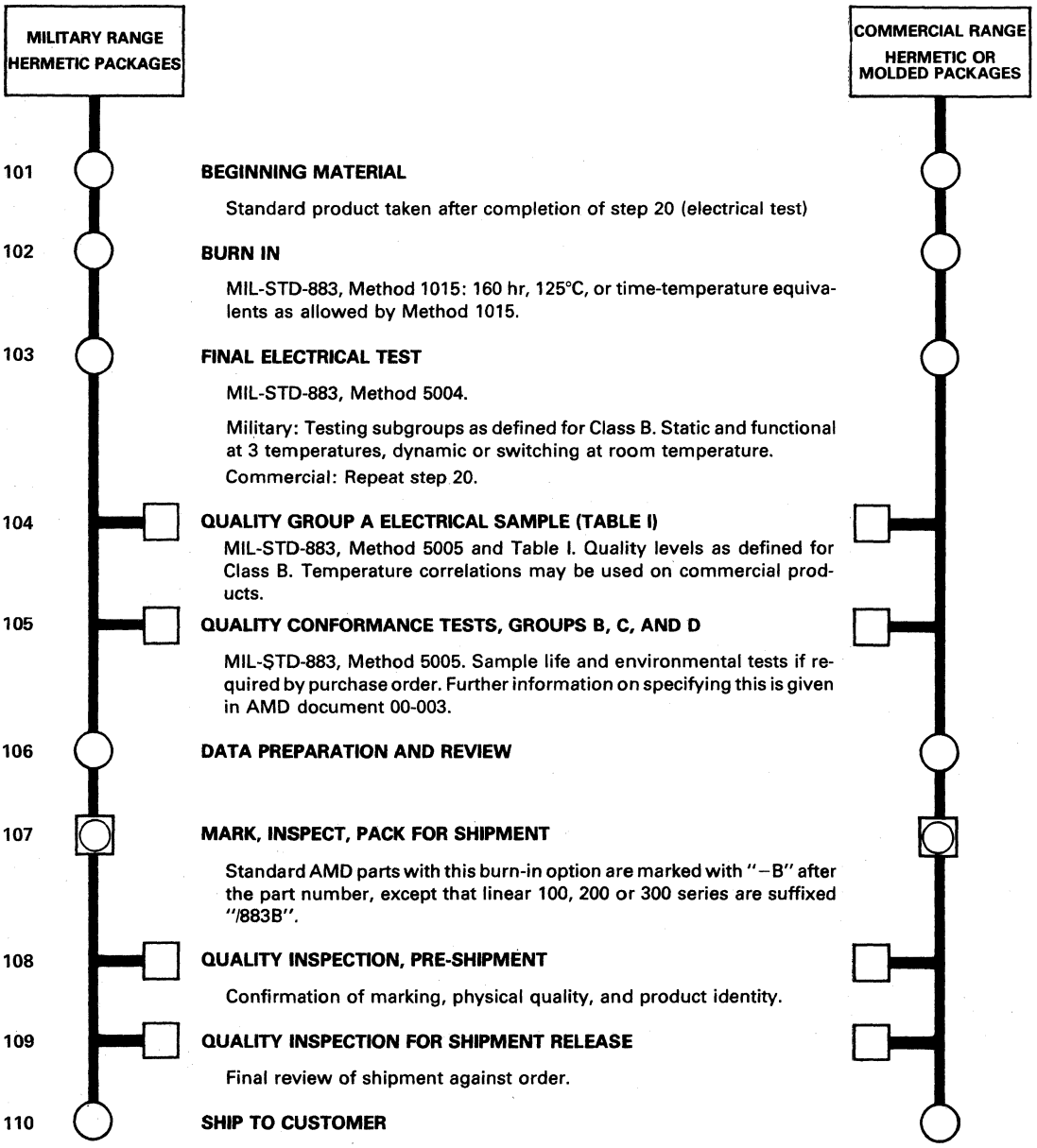
GROUP A ELECTRICAL TESTS
From MIL-STD-883, Method 5005, Table I

Subgroups	LTPD (Note 1)	Initial Sample Size
Subgroup 1 — Static tests at 25°C	5	45
Subgroup 2 — Static tests at maximum rated operating temperature	7	32
Subgroup 3 — Static tests at minimum rated operating temperature	7	32
Subgroup 4 — Dynamic tests at 25°C — Linear devices	5	45
Subgroup 5 — Dynamic tests at maximum rated operating temperature — Linear devices	7	32
Subgroup 6 — Dynamic tests at minimum rated operating temperature — Linear devices	7	32
Subgroup 7 — Functional tests at 25°C	5	45
Subgroup 8 — Functional tests at maximum and minimum rated operating temperatures	10	22
Subgroup 9 — Switching tests at 25°C — Digital devices	7	32
Subgroup 10 — Switching tests at maximum rated operating temperature — Digital devices (Note 2)	10	10
Subgroup 11 — Switching tests at minimum rated operating temperature — Digital devices (Note 2)	10	10

1. Sampling plans are based on LTPD tables of MIL-M-38510. The smaller initial sample size, based on zero rejects allowed, has been chosen unless otherwise indicated. If necessary, the sample size will be increased once to the quantity corresponding to an acceptance number of 2. The minimum reject number in all cases is 3.
2. These subgroups are usually performed during initial device characterization only.

OPTIONAL EXTENDED PROCESSING
CLASS B
Steps 101 Through 110

Advanced Micro Devices offers several extended processing options to meet customer high-reliability requirements. These are defined in AMD document 00-003. The flow chart below outlines Option B, a 160-hr burn in. Military temperature range devices processed to this flow (in the left column) meet the screening requirements of MIL-STD-883, Class B.



OTHER OPTIONS

Document 00-003, "Extended Processing Options", further defines Option B as well as other screening or sampling options available or special order. Available options are listed here for reference.

Option	Description	Effect
A	Modified Class A screen (Similar to Class S screening)	Provides space-grade product, following most Class S requirements of MIL-STD-883, Method 5004.
B	160-hr operating burn in	Upgrades a part from Class C to Class B.
X	Radiographic inspection (X-ray)	Related to Option A. Provides limited internal inspection of sealed parts.
S	Scanning Electron Microscope (SEM) metal inspection	Sample inspection of metal coverage of die.
V	Preseal visual inspection to MIL-STD-883, Method 2010, Cond. A	More stringent visual inspection of assemblies and die surfaces prior to seal.
P	Particle impact noise (PIN) screen with ultrasonic detection.	Detects loose particles of approximately 0.5 mil size or larger, which could affect reliability in zero-G or high vibration applications.
Q	Quality conformance inspection (Group B, C and D life and environmental tests)	Samples from the lot are stressed and tested per Method 5005. The customer's order must state which groups are required. Group B destroys 16 devices; Group C, 92 devices; Group D, 60 devices.

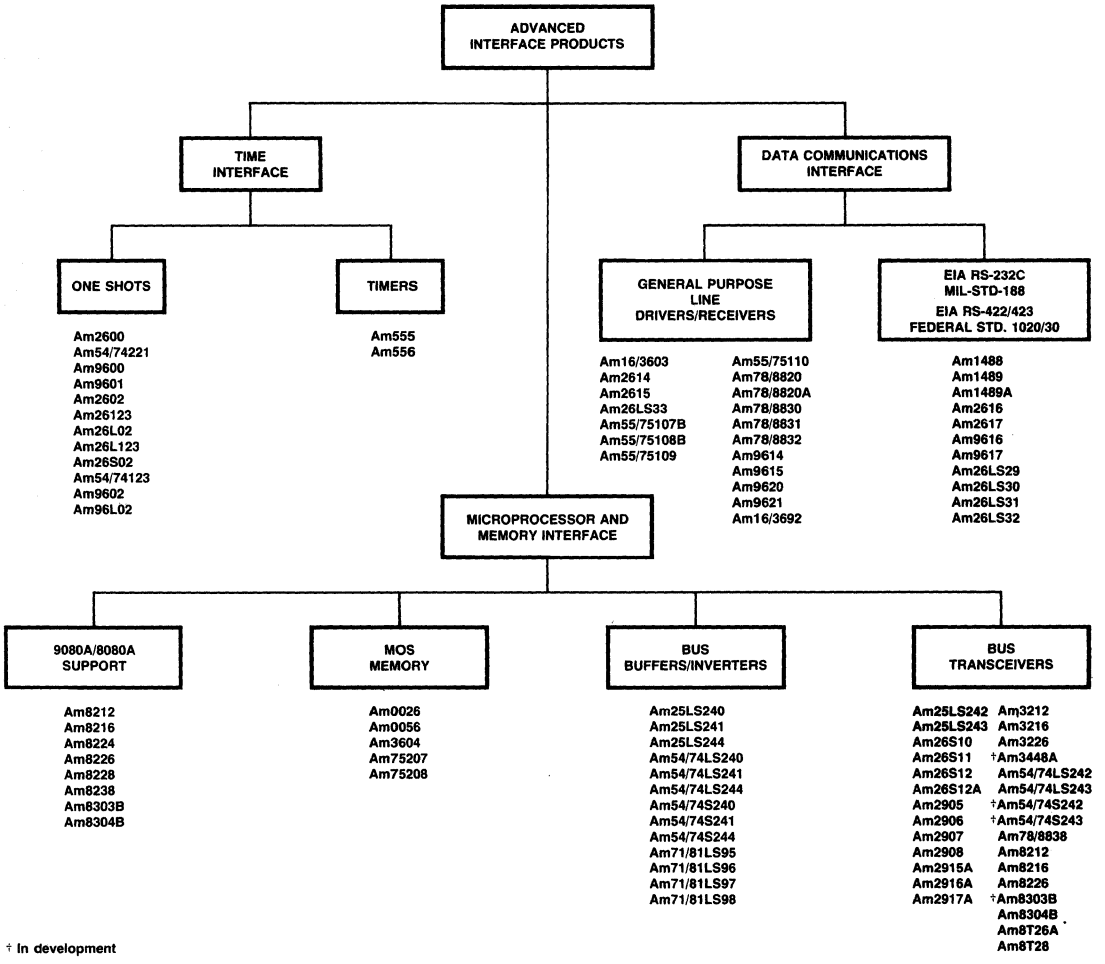
2

Product Guides and Cross References

Interface Circuits

2

- Alpha-Numeric Cross Reference 2-2
- Selection Guide 2-4



† In development

ALPHA NUMERIC CROSS REFERENCE

This list includes devices which can be replaced directly by an Advanced Micro Devices product. In some cases an alternate source vendor may not choose to use the same part number as the original manufacturer. To minimize the number of marking options we recommend ordering the device by the original source designation, as noted in this list.

DEVICE	DESCRIPTION	ORDER #
Am25LS240	Octal Inverting Buffer/Driver	AM25LS240
Am25LS241	Octal Non-Inverting Buffer/Driver	AM25LS241
Am25LS242	Quad Inverting Transceiver	AM25LS242
Am25LS243	Quad Non-Inverting Transceiver	AM25LS243
Am25LS244	Octal Non-Inverting Buffer/Driver	AM25LS244
Am2600	One-Shot	AM2600
Am2602	Dual One-Shot	AM2602
Am2614	Quad Line Driver	AM2614
Am2615	Dual Line Receiver	AM2615
Am2616	EIA/MIL 188C Quad Line Driver	AM2616
Am2617	EIA Quad Line Receiver	AM2617
Am26123	Dual One-Shot	AM26123
Am26L02	Low-Power, Dual One-Shot	AM26L02
Am26L123	Low-Power, Dual One-Shot	AM26L123
Am26LS29	Quad EIA RS-423 Line Driver	AM26LS29
Am26LS30	EIA RS-422/423 Line Driver	AM26LS30
Am26LS31	Quad Differential Line Driver	AM26LS31
Am26LS32	Quad Line Receiver EIA RS-422/423	AM26LS32
Am26LS33	Quad Line Receiver	AM26LS33
Am26S02	Dual One-Shot	AM26S02
Am26S10	Quad Inverting Bus Transceiver	AM26S10
Am26S11	Quad Non-Inverting Bus Transceiver	AM26S11
Am26S12	Quad Bus Transceiver (Hysteresis)	AM26S12
Am26S12A	Quad Bus Transceiver (Hysteresis)	AM26S12A
Am2905	Quad LSI Bus Transceiver – O.C.	AM2905
Am2906	Quad LSI Bus Transceiver – O.C.	AM2906
Am2907	Quad LSI Bus Transceiver – O.C.	AM2907
Am2908	Quad LSI Bus Transceiver – O.C.	AM2908
Am2915A	Quad LSI Bus Transceiver – Three-State	AM2915A
Am2916A	Quad LSI Bus Transceiver – Three-State	AM2916A
Am2917A	Quad LSI Bus Transceiver – Three-State	AM2917A
†DP7303B	Octal Inverting Transceiver	DP7303B
†DP7304B	Octal Bidirectional Transceiver	DP7304B
Am8212	8-Bit I/O Port for 9080A/8080A	AM8212
Am8216	Quad Bus Driver for 9080A/8080A	AM8216
Am8224	Clock Generator for 9080A/8080A	AM8224
Am8226	Quad Bus Inverter for 9080A/8080A	AM8226
Am8228	System Controller for 9080A/8080A	AM8228
Am8238	System Controller for 9080A/8080A	AM8238
DM54123	Dual One-Shot	SN54123
DM71LS95	Octal Non-Inverting Buffer/Driver	DM71LS95
DM71LS96	Octal Inverting Buffer/Driver	DM71LS96
DM71LS97	Octal Non-Inverting Buffer/Driver	DM71LS97
DM71LS98	Octal Inverting Buffer/Driver	DM71LS98
DM74123	Dual One-Shot	SN74123
DM7820	Dual Differential Line Receiver	DM7820
DM7820A	Dual Differential Line Receiver	DM7820A
DM7830	Dual Differential Line Receiver	DM7830
DM7831	Three-State Line Driver	DM7831
DM7832	Three-State Line Driver	DM7832
DM81LS95	Octal Non-Inverting Buffer/Driver	DM81LS95
DM81LS96	Octal Inverting Buffer/Driver	DM81LS96
DM81LS97	Octal Non-Inverting Buffer/Driver	DM81LS97
DM81LS98	Octal Inverting Buffer/Driver	DM81LS98
DM8601	One-Shot	9601*C
DM8602	Dual One-Shot	9602*C
DM8820	Dual Differential Line Receiver	DM8820
DM8820A	Dual Differential Line Receiver	DM8820A
DM8830	Dual Differential Line Driver	DM8830
DM8831	Three-State Line Driver	DM8831
DM8832	Three-State Line Driver	DM8832
DM9601	One-Shot	9601*M
DM9602	Dual One-Shot	9602*M
DP8303B	Octal Inverting Transceiver	DP8303B
DP8304B	Octal Bidirectional Transceiver	DP8304B
DS0026	Two Phase MOS Clock Driver	MH0026
DS0026C	Two Phase MOS Clock Driver	MH0026C
DS0056	Two Phase MOS Clock Driver	DS0056
DS0056C	Two Phase MOS Clock Driver	DS0056C
DS1488	Quad EIA Line Driver	MC1488
DS1489	Quad EIA Line Receiver	MC1489
DS1489A	Quad EIA Line Receiver	MC1489A
DS1603	Dual Differential Line Receiver	DS1603

DEVICE	DESCRIPTION	ORDER #
DS1691	EIA RS-422/423 Line Driver	AM26LS30
DS1692	Dual Differential Driver	DS1692
DS3603	Dual Differential Line Receiver	DS3603
DS3604	Dual MOS Sense Amp	DS3604
DS3691	EIA RS-422/423 Line Driver	AM26LS30
DS3692	Dual Differential Driver	DS3692
DS55107	Dual Differential Line Receiver	SN55107B
DS55108	Dual Differential Line Receiver	SN55108B
DS55109	Dual Differential Line Driver	SN55109
DS55110	Dual Differential Line Driver	SN55110
DS75107	Dual Differential Line Receiver	SN75107B
DS75108	Dual Differential Line Receiver	SN55108B
DS75109	Dual Differential Line Driver	SN75109
DS75110	Dual Differential Line Driver	SN75110
DS7820	Dual Differential Line Receiver	DM7820
DS7820A	Dual Differential Line Receiver	DM7820A
DS7831	Three-State Line Driver	DM7831
DS7832	Three-State Line Driver	DM7832
DS7838	Unified Quad Bus Transceiver	DS7838
DS8820	Dual Differential Line Receiver	DM8820
DS8820A	Dual Differential Line Receiver	DM8820A
DS8830	Dual Differential Line Receiver	DM8830
DS8831	Three-State Line Driver	DM8831
DS8832	Three-State Line Driver	DM8832
DS8838	Unified Quad Bus Transceiver	DS8838
LM163	Dual Differential Line Receiver	DS1603
LM363	Dual Differential Line Receiver	DS3603
LM363A	Dual MOS Sense Amp/Line Receiver	DS3604
LM555	Precision Timer	SE555
LM555C	Precision Timer	NE555
LM556	Dual Precision Timer	SE556
LM556C	Dual Precision Timer	NE556
LM1488	Quad EIA Line Driver	MC1488
LM1489	Quad EIA Line Receiver	MC1489
LM1489A	Quad EIA Line Receiver	MC1489A
LM55107	Dual Differential Line Receiver	SN55107B
LM55108	Dual Differential Line Receiver	SN55108
LM55109	Dual Differential Line Driver	SN55109
LM55110	Dual Differential Line Driver	SN55110
LM75107	Dual Differential Line Receiver	SN75107B
LM75108	Dual Differential Line Receiver	LM75108B
LM75109	Dual Differential Line Driver	SN75109
LM75110	Dual Differential Line Driver	SN75110
LM7520	Dual Sense Amp; ±4mV Threshold	SN7520
LM7521	Dual Sense Amp; ±7mV Threshold	SN7521
LM7524	Dual Sense Amp; ±4mV Threshold	SN7524
LM7525	Dual Sense Amp; ±7mV Threshold	SN7525
LM75325	Core Memory Driver	SN75325
MC1455	Precision Timer	NE555
MC1488	Quad EIA Line Driver	MC1488
MC1489	Quad EIA Line Receiver	MC1489
MC1489A	Quad EIA Line Receiver	MC1489A
MC1555	Precision Timer	SE555
†MC3448A	IEEE-488 Quad Transceiver	MC3448A
MC3456	Dual Precision Timer	NE556
MC3556	Dual Precision Timer	SE556
MC8601	One-Shot	9601*C
MC8602	Dual One-Shot	9602*C
MC9601	One-Shot	9601*M
MC9602	Dual One-Shot	9602*M
MC55107	Dual Differential Line Receiver	SN55107B
MC55108	Dual Differential Line Receiver	SN55108B
MC55109	Dual Differential Line Driver	SN55109
MC55110	Dual Differential Line Driver	SN55110
MC75107	Dual Differential Line Receiver	SN75107B
MC75108	Dual Differential Line Receiver	SN75108B
MC75109	Dual Differential Line Driver	SN75109
MC75110	Dual Differential Line Driver	SN75110
MH0026	Two-Phase MOS Clock Driver	MH0026
MH0026C	Two-Phase MOS Clock Driver	MH0026C
MMH0026	Two-Phase MOS Clock Driver	MMH0026
MMH0026C	Two-Phase MOS Clock Driver	MMH0026C
N8T22	One-Shot	9601*C

ALPHA NUMERIC CROSS REFERENCE (Cont.)

DEVICE	DESCRIPTION	ORDER #
N8T26	Quad Three-State Bus Transceiver	N8T26
N8T26A	Quad Three-State Bus Transceiver	N8T26A
N8T28	Quad Three-State Bus Transceiver	N8T28
NE555	Precision Timer	NE555
NE556	Dual Precision Timer	NE556
S8T22	One-Shot	9601*M
S8T26	Quad Three-State Bus Transceiver	S8T26
†S8T26A	Quad Three-State Bus Transceiver	S8T26A
†S8T28	Quad Three-State Bus Transceiver	S8T28
SE555	Precision Timer	SE555
SE556	Dual Precision Timer	SE556
SN52555	Precision Timer	SE555
SN54123	Dual One-Shot	SN54123
SN54221	Dual One-Shot	SN54221
SN54S240	Octal Inverting Buffer/Driver	†SN54S240
SN54S241	Octal Non-Inverting Buffer/Driver	†SN54S241
†SN54S242	Octal Inverting Buffer/Driver	†SN54S242
†SN54S243	Octal Non-Inverting Buffer/Driver	†SN54S243
SN54S244	Octal Inverting Buffer/Driver	†SN54S244
SN55107A	Dual Differential Line Receiver	SN55107B
SN55107B	Dual Differential Line Receiver	SN55107B
SN55108A	Dual Differential Line Receiver	SN55108B
SN55108B	Dual Differential Line Receiver	SN55108B
SN55109	Dual Differential Line Driver	SN55109
SN55110	Dual Differential Line Driver	SN55110
SN55114	Dual Differential Line Driver	9614*M
SN55115	Dual Differential Line Receiver	9615*M
SN55182	Dual Differential Line Receiver	DM7820A
SN55183	Dual Differential Line Receiver	DM7830
SN55369	Two-Phase MOS Clock Driver	MH0026
SN72555	Precision Timer	SN72555
SN74123	Dual One-Shot	SN74123
SN74221	Dual One-Shot	SN74221
SN74LS424	Clock Generator for 9080A/8080A	Am8224
SN74LS240	Octal Inverting Buffer/Driver	SN74LS240
SN74LS241	Octal Non-Inverting Buffer/Driver	SN74LS241
SN74LS242	Quad Inverting Transceiver	SN74LS242
SN74LS243	Quad Non-Inverting Transceiver	SN74LS243
SN74LS244	Octal Non-Inverting Buffer/Driver	SN74LS244
SN74S240	Octal Inverting Buffer/Driver	SN74S240
SN74S241	Octal Non-Inverting Buffer/Driver	SN74S241
†SN74S242	Quad Inverting Transceiver	†SN74S242
†SN74S243	Quad Non-Inverting Transceiver	†SN74S243
SN74S244	Octal Inverting Buffer/Driver	SN74S244
SN75107A	Dual Differential Line Receiver	SN75107B
SN75107B	Dual Differential Line Receiver	SN75107B
SN75108A	Dual Differential Line Receiver	SN75108B
SN75108B	Dual Differential Line Receiver	SN75108B
SN75109	Dual Differential Line Driver	SN75109
SN75110	Dual Differential Line Driver	SN75110
SN74114	Dual Line Driver	9614*C
SN75115	Dual Differential Line Receiver	9615*C
SN75182	Dual Differential Line Receiver	DM8820A
SN75183	Dual Differential Line Receiver	DM8830
SN75188	Quad EIA Line Driver	MC1488
SN75189	Quad EIA Line Receiver	MC1489
SN75189A	Quad EIA Line Receiver	MC1489A
SN75207	Dual MOS Sense Amp/Line Receiver	SN75207
SN75208	Dual MOS Sense Amp/Line Receiver	SN75208
SN75369	Two-Phase MOS Clock Driver	MH0026
3212	8-Bit I/O Port	3212
3216	Non-Inverting Quad Bus Transceiver	3216
3226	Inverting Quad Bus Transceiver	3226
8212	8-Bit I/O Port for 9080A/8080A	AM8212/8212
8216	Non-Inverting Quad Bus Transceiver	8216
8224	Clock Generator for 9080A/8080A	8224
8226	Inverting Quad Bus Transceiver	8226
8228	System Controller for 9080A/8080A	AM8228/8228
8238	System Controller for 9080A/8080A	AM8238/8238
9600	One-Shot	9600
9601	One-Shot	9601
9602	Dual One-Shot	9602
96L02	Low-Power, Dual One-Shot	96L02
9614	Dual Line Driver	9614
9615	Dual Differential Line Receiver	9615
9616	Triple EIA Line Driver	9616
9617	Triple EIA Line Receiver	9617
9620	Dual Differential Line Receiver	9620
9621	Dual Differential Line Driver	9621

†To be announced. * is the package designator position.

SELECTION GUIDE

LINE DRIVERS

DUAL DIFFERENTIAL		Use With
75109	Open collector differential outputs typical current 6mA, inhibit controls	75107B 75108B
75110	12mA output current version of Am75109	75107B 75108B
8830	Designed for single 5.0V supply operation	7820 or 7820A
8831	Dual differential device which may also be used as a quad single-ended driver. Three-state output.	9615 or 2615
8832	Similar to 8831 but no V_{CC} clamp diodes	9615 or 2615
9614	5 volt supply driver with complementary outputs	9615
9621	200mA transient capability with 130Ω back matching resistor	9620
DIFFERENTIAL EIA RS-422, FEDERAL STD 1020		
26LS31	Quad, high-speed, low output skew	26LS32 or
26LS30	Dual, high output CMR	26LS33
SINGLE ENDED		
2614	High-speed quad driver for multi-channel, common ground operation.	2615
SINGLE ENDED, EIA RS-232-C		
1488	Quad EIA RS-232C driver (14 pins)	1489/ 1489A
2616	Quad 16-pin driver for EIA RS-232C, CCITT V.24 and MIL-188C interface	2617
9616	Triple EIA RS-232C driver (14 pins)	9617
SINGLE ENDED, EIA RS-423, FEDERAL STD 1030		
26LS29	Quad, three-state	26LS32 or
26LS30	Quad, mode control	26LS33

BUS BUFFERS/DRIVERS

		t_{pd} (TYP)	I_{OL} (MAX)
25LS240	Inverting octal buffer/driver with three-state output	10	48
74LS240		10	24
74S240		4.5	68
81LS96		9.0	16
25LS241	Non-inverting octal buffer/driver with three-state output	12	48
74LS241		12	24
74S241		6.0	68
81LS95		12	16
25LS242	Inverting buffer/driver with two quad data paths connected input-to-output	10	48
74LS242		10	24
†74S242		4.5	68
25LS243	Non-inverting buffer/driver with two quad data paths connected input-to-output	12	48
74LS243		12	24
†74S243		6.0	68
25LS244	Non-inverting octal buffer/driver with three-state output and two inverting enables	12	48
74LS244		12	24
74S244		6.0	68
81LS97		12	16
81LS98	Inverting octal buffer/driver with three-state output and two inverting enables	9.0	16

†In development

LINE RECEIVERS

DUAL DIFFERENTIAL		Use With
3603	Receiver with differential input to detect signals > 25mV. Three-state outputs.	75110
75107B	Totem-pole TTL output version of Am363	75109 or 75110
75108B	Open collector TTL output version of Am363	75109 or 75110
8820	Designed for ±15V common mode using 5.0V supply	8830
8820A	Higher speed, tighter spec 8820	8830
9615	±15 volt common mode, 5 volt supply receivers with uncommitted collector and active pull-up controls	9614
9620	±15 volt common mode receiver with direct and attenuated inputs	9621
QUAD DIFFERENTIAL		
26LS33	±15 volt common mode, 5 volt supply, three-state output	26LS31
QUAD DIFFERENTIAL EIA RS-422, FEDERAL STD 1020		
26LS32	±7 volt common mode, 5 volt supply, three-state output	26LS31
SINGLE ENDED		
2615	Receiver for 3 volt single-ended TTL level data	2614
SINGLE ENDED, EIA RS-232-C		
1489	Quad EIA RS-232C receiver with input threshold hysteresis	1488
1489A	Higher threshold version of Am1489	1488
2617	Quad EIA RS-232 receiver specified over military temperature range (same pinout as Am1489A)	2616
9617	Triple EIA RS-232 receiver with adjustable hysteresis	9616
SINGLE ENDED, EIA RS-423, FEDERAL STD 1030		
26LS32	±7 volt common mode, 5 volt supply, three-state output	26LS29 26LS30

SPECIAL FUNCTIONS

TIMERS	
555	Single, Precision oscillator/timer
556	Dual version 555

MOS MEMORY

DRIVERS	
0026	Dual 5MHz Two-Phase MOS clock driver
0056	0026 with added V_{BB} terminal
SENSE AMPLIFIERS	
3604	Differential input for signals > 10mV, Three-state outputs
75207	Totem-pole TTL output 3604
75208	Open-collector 3604

MOS-MICROPROCESSOR INTERFACE CIRCUITS

8080A/9080A	
8212	8-Bit input/output port, with storage
8216	4-Bit parallel bidirectional bus driver
8224	Clock generator and driver
8226	Inverting version 8216
8228	System controller and bus driver
8238	System controller and bus driver with extended IOW/MEMW
8303B	Two 8226's in one 20 pin package
8304B	Two 8216's in one 20 pin package

BUS TRANSCEIVERS

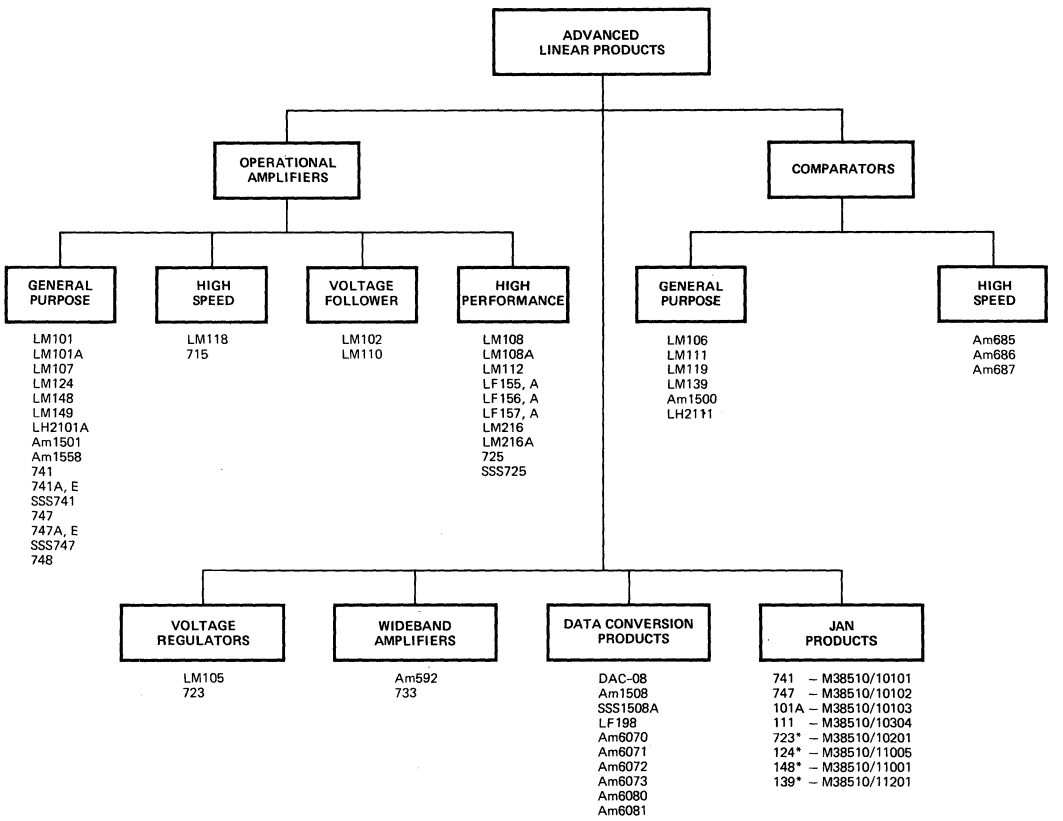
Device	Output	Function	Hysteresis	Speed (Note 1)	Comments
QUAD					
Am26S10	100mA-O.C.	Inverting	No	20ns	SN55/75138 pin out
Am26S11	100mA-O.C.	Non-Inverting to bus; Inverting off bus	No	22ns	Same as Am26S10 except non-inverting to bus
Am26S12	100mA-O.C.	Inverting	Yes-0.6V	32ns	Same pin out as DS78/8838 and 8T38
Am26S12A	100mA-O.C.	Inverting	Yes-1.05V	32ns	Wider threshold Am26S12
Am2905	100mA-O.C.	Inverting	No	31ns (Note 2)	Has 2-input multiplexer
Am2906	100mA-O.C.	Inverting	No	31ns (Note 2)	Has 2-input multiplexer and parity
Am2907	100mA-O.C.	Inverting	No	31ns (Note 2)	Includes parity, 2.0V receiver V_{TH}
Am2908	100mA-O.C.	Inverting	No	31ns (Note 2)	Includes parity, 1.5V receiver V_{TH}
Am2915A	48mA/3-St.	Inverting	No	31ns (Note 2)	Has 2-input multiplexer
Am2916A	48mA/3-St.	Inverting	No	31ns (Note 2)	Has 2-input multiplexer and parity
Am2917A	48mA/3-St.	Inverting	No	31ns (Note 2)	Includes parity
Am3216	50mA/3-St.	Non-Inverting	No	34ns	Same as 8216 except different A.C. loading spec
Am3226	50mA/3-St.	Inverting	No	30ns	Same as 8216 except different A.C. loading spec
Am3448A	48mA/3-St.-O.C.	Non-Inverting	Yes	32ns	IEEE 488 compatible
Am78/8838	50mA-O.C.	Inverting	No	38ns	Same pin out and function as Am26S12A and 8T38
Am8T26A	48mA/3-St.	Inverting	No	19ns	V_{OH} MOS compatible
Am8T28	48mA/3-St.	Non-Inverting	No	25ns	V_{OH} MOS compatible
Am8216	50mA/3-St.	Non-Inverting	No	34ns	Similar to 8T28
Am8226	50mA/3-St.	Non-Inverting	No	30ns	Similar to 8T26A
OCTAL					
Am8303B	48mA/3-St.	Inverting	No	14ns	Same as two 8226's in one 20 pin package
Am8304B	48mA/3-St.	Non-Inverting	No	24ns	Same as two 8216's in one 20 pin package

Notes: 1. Typical delay at 28°C for input to bus plus receiver to output.
2. Bus enable to bus plus bus to receiver output. All parts include register or driver plus receiver with latch.

Linear Circuits

- **Function Selector Guide**
- **Ordering and Package Guide**
- **Jan Cross Reference and Ordering Guide**
- **Competitive Cross Reference**

Advanced Micro Devices offers a broad line of high performance Linear integrated circuits including operational amplifiers, comparators, voltage regulators, wideband amplifiers and D to A conversion circuits. All Advanced Micro Devices Linear circuits are processed to the reliability and quality requirements of MIL-STD-883A and MIL-M-38510.



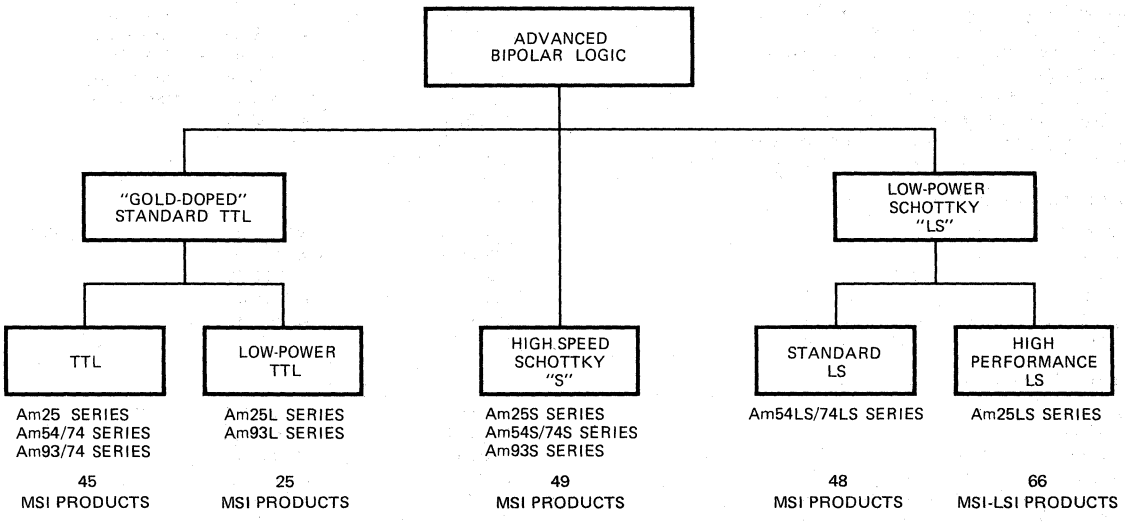
*In Development

Bipolar Logic Circuits

- **Schottky MSI**
- **Low-Power Schottky MSI-LSI**
- **Standard TTL/MSI**
- **Low-Power TTL/MSI**

Advanced Micro Devices offers a broad line of complex TTL/MSI and LSI bipolar logic integrated circuits. Four basic process technologies are employed to meet the various speed, power and cost combinations required by manufacturers of high performance digital systems. The table below compares the relative speed-power performance of each of the families.

The Advanced Bipolar Logic line consists of both proprietary and industry standard devices fabricated with Schottky, Low Power Schottky and conventional "gold-doped" processes. All Advanced Micro Devices' circuits are processed to the reliability and quality requirements of MIL-STD-883.



Family		Ave. Gate Propagation Delay	Ave. Power Per Gate	Typ. Register Clock Freq.
High-Speed Schottky	Am25S Am54S/74S Am93S	3ns	20mW	110MHz
Low-Power Schottky	Am25LS Am54LS/74LS	5ns 10ns	2mW 2mW	65MHz 40MHz
Standard TTL	Am25 Am54/74 Am93	10ns	10mW	40MHz
Low-Power Standard TTL	Am25L Am93L	25ns	2.5mW	15MHz

ADVANCED MICRO DEVICES
SCHOTTKY AND LOW-POWER SCHOTTKY MSI
FUNCTIONAL SELECTOR GUIDE

Advanced Micro Devices offers a complete line of Schottky and Low-Power Schottky MSI products. On the following pages are a selector guide for these products and brief data on several of the most useful parts. For complete data refer to our Schottky and Low-Power Schottky Data Book.

DESCRIPTION	HIGH-PERFORMANCE LOW-POWER SCHOTTKY	STANDARD LOW-POWER SCHOTTKY	HIGH-SPEED SCHOTTKY
REGISTERS			
Four-Bit Register with Common Clock Enable	*25LS08	54/74LS379	*25S08/54/74S379
Four-Bit Register with Two-Input Multiplexers on Inputs	*25LS09	54/74LS399	*25S09/54/74S399
Four-Bit Register with Standard and Three-State Outputs	†25LS2518/29LS18		†25S18/54/74S388/2918
Four-Bit, Two-Output Three-State Register	†25LS2519/2919		
Four-Bit Register with Common Clear	25LS175	54/74LS175	54/74S175
Four-Bit Register; Shift Right, Left or Parallel Load	25LS194A	54/74LS194A	54/74S194
Four-Bit Register; Shift Right or Parallel Load	25LS195A	54/74LS195A	54/74S195
Six-Bit Register with Common Clock Enable	*25LS07	54/74LS378	*25S07/54/74S378
Six-Bit Register with Common Clear	25LS174	54/74LS174	54/74S174
Eight-Bit, Serial-In, Parallel-Out Register	25LS164	54/74LS164	
Eight-Bit Shift/Storage Register; Synchronous Clear	*25LS23		
Eight-Bit Shift/Storage Register; Asynchronous Clear	25LS299	54/74LS299	
Eight-Bit Shift/Storage Register with Sign Extend	25LS22		
Octal D-Type Register, Common Clear	*25LS273	*54/74LS273	
Octal D-Type Register, Common Clear, Buffered Outputs	*25LS273B		
Octal Transparent Latch (Three-State)	*25LS373	*54/74LS373	**54/74S373
Octal Transparent Latch, Inverting (Three-State)	*25LS533	*54/74LS533	**54/74S533
Octal D-Type Register (Three-State)	*25LS374	*54/74LS374	**54/74S374
Octal D-Type Register, Inverting (Three-State)	*25LS534	*54/74LS534	**54/74S534
Octal D-Type Register, Common Enable	*25LS377	*54/74LS377	
Octal D-Type Register, Common Enable, Buffered Outputs	*25LS377B		
Octal D-Type Register, Common Enable and Clear, Three-State	†25LS2520/2920		
DECADE (BCD) COUNTERS			
Asynchronous Clear	25LS160	54/74LS160	54/74S160/93S10
Synchronous Clear	25LS162	54/74LS162	
Up-Down, Synchronous Preset	25LS168	54/74LS168	
Up-Down, Asynchronous Preset, Single Clock	25LS190	54/74LS190	
Up-Down, Asynchronous Preset, Dual Clock	25LS192	54/74LS192	
Up-Down, Synchronous Preset, Three-State	*25LS2568		
BINARY HEXADECIMAL COUNTERS			
Asynchronous Clear	25LS161	54/74LS161	54/74S161/93S16
Synchronous Clear	25LS163	54/74LS163	
Up-Down, Synchronous Preset	25LS169	54/74LS169	
Up-Down, Asynchronous Preset, Single Clock	25LS191	54/74LS191	
Up-Down, Asynchronous Preset, Dual Clock	25LS193	54/74LS193	
Up-Down, Synchronous Preset, Three-State	*25LS2569		
QUAD BUS TRANSCEIVERS/DRIVERS			
Quad Bus Transceiver, Inverting (100mA)			**26S10
Quad Bus Transceiver, Non-Inverting (100mA)			**26S11
Quad Bus Transceiver, Inverting	*25LS242	*54/74LS242	**54/74S242
Quad Bus Transceiver, Non-Inverting	*25LS243	*54/74LS243	**54/74S243
Quad Open-Collector Bus Transceiver			*26S12/12A
Quad Three-State Bus Transceiver (Inverting)			**8T26/8T26A
Quad Three-State Bus Transceiver (Non-Inverting)			*8T28
Quad Three-State Bus Transceiver with Receiver Latch (Inverting)	†2927		
Quad Three-State Bus Transceiver with Receiver Register	†2928		
Quad Two I/P Transceiver with Three-State Receiver (O.C.)	†2905		
Quad Two I/P Transceiver with Parity (O.C.)	†2906		
Quad Two I/P Transceiver with Parity (O.C.)	†2907		
Quad Two I/P Transceiver with Parity (O.C. and DEC Q/LSI-II Bus Compatible)	†2908		
Quad Two I/P Transceiver with Three-State Receiver (Three-State)	†2915A		
Quad Two I/P Transceiver with Parity (Three-State)	†2916A		
Quad Two I/P Transceiver with Parity (Three-State)	†2917A		
Quad IEEE-488 Bidirectional Bus Transceiver	*3448A		

FUNCTIONAL SELECTOR GUIDE (Cont.)

DESCRIPTION	HIGH-PERFORMANCE LOW-POWER SCHOTTKY	STANDARD LOW-POWER SCHOTTKY	HIGH-SPEED SCHOTTKY
OCTAL BUS TRANSCEIVERS/DRIVERS			
Octal Bus Driver, Inverting	*25LS240	*54/74LS240	**54/74S240
Octal Bus Driver, Non-Inverting (Complementary G, $\overline{G}$ Inputs)	*25LS241	*54/74LS241	**54/74S241
Octal Bus Driver, Non-Inverting	*25LS244	*54/74LS244	**54/74S244
Octal Bidirectional Bus Transceiver (Non-Inverting)			**73/8304B
Octal Bidirectional Bus Transceiver (Inverting)			**73/8303
Octal Buffer/Driver (Non-Inverting)		*81LS95	
Octal Buffer/Driver (Inverting)		*81LS96	
Octal Buffer/Driver (Non-Inverting)		*81LS97	
Octal Buffer/Driver (Inverting)		*81LS98	
OPERATORS (ALU, MULTIPLIER, PRIORITY ENCODER, etc.)			
Four by Two Two's Complement Multiplier			25S05
Four-Bit, Four-Way Shifter			**25S10/54/74S350
Four-Bit ALU/Function Generator	25LS181	54/74LS181	54/74S181
Four-Bit ALU/Function Generator	25LS2517		
Four-Bit ALU/Function Generator	25LS381	54/74LS381	
Priority Encoder, Eight Line to Three Line	25LS148	54/74LS148	
Four-Bit Serial Adder/Subtracter	25LS15		
Priority Encoder, Three-State	25LS2513		
Eight by One Serial/Parallel Two's Complement Multiplier	25LS14		
Eight-Bit by Eight-Bit Multiplier/Accumulator	*25LS2516		
Eight-Bit by Eight-Bit Combinatorial Multiplier, Latch Outputs			*25S557
Eight-Bit by Eight-Bit Combinatorial Multiplier			*25S558
Eight-Bit Comparator	25LS2521		
System Clock Generator and Driver	†25LS2525/2925		
MEMORY INTERFACE			
Dynamic Memory Controller	*25LS2564		
DECODER/DEMULTIPLEXERS			
One-of-Ten Decoder/Demultiplexer, Polarity Control	25LS2537		
One-of-Eight Decoder/Demultiplexer	25LS138	54/74LS138	54/74S138
One-of-Eight Decoder/Demultiplexer with Control Storage	*25LS2536		
Dual One-of-Four Decoder/Demultiplexer	25LS139	54/74LS139	54/74S139/93S21
One-of-Eight Decoder/Demultiplexer, Polarity Control	†25LS2538/2921		
Dual One-of-Four Decoder/Demultiplexer, Polarity Control	25LS2539		
MULTIPLEXERS			
Eight-Input Multiplexer	25LS151	54/74LS151	54/74S151
Eight-Input Multiplexer with Control Storage	†25LS2535/2922		
Three-State Eight-Input Multiplexer	25LS251	54/74LS251	54/74S251
Dual Four-Input Multiplexer	25LS153	54/74LS153	54/74S153
Three-State Dual Four-Input Multiplexer	25LS253	54/74LS253	54/74S253
Quad Two-Input Multiplexer; Non-Inverting	25LS157	54/74LS157	54/74S157/93S22
Three-State Quad Two-Input Multiplexer; Non-Inverting	25LS257	54/74LS257	54/74S257
Quad Two-Input Multiplexer; Inverting	25LS158	54/74LS158	54/74S158
Three-State Quad Two-Input Multiplexer; Inverting	25LS258	54/74LS258	54/74S258
MONOSTABLE (ONE-SHOT)			
Dual Retriggerable, Resettable Monostable Multivibrator			26S02
PARITY CHECKER/GENERATORS			
Nine-Input Parity Checker/Generator			82S62
Twelve-Input Parity Checker/Generator			**93S48

*Logic Diagram and Connection Diagram are on following pages. Refer to Schottky and Low-Power Schottky Data Book for complete data sheet information.

**Complete data sheet located on following pages.

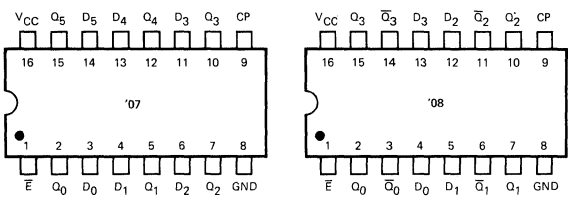
†Refer to previous sections for complete data sheet information (Am29XX).

REGISTERS

Am25LS07 • Am25LS08
Am25S07 • Am25S08

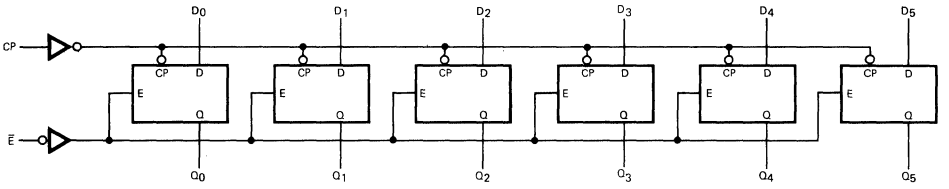
- Positive edge-triggered parallel D registers
- Common clock and common enable

CONNECTION DIAGRAMS

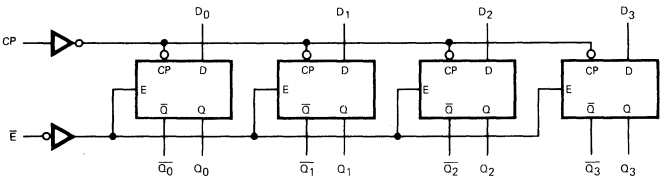


LOGIC DIAGRAMS

'07



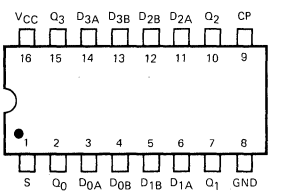
'08



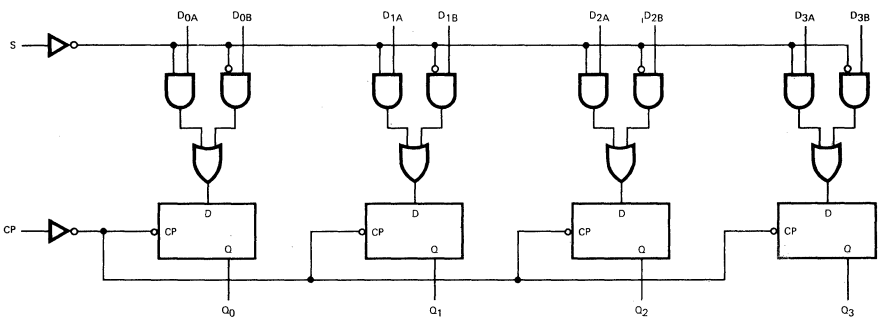
Am25LS09 • Am25S09

- 4-bit register accepts data from one-of-two 4-bit input fields
- Buffered common edge-triggered clock

CONNECTION DIAGRAM



LOGIC DIAGRAM

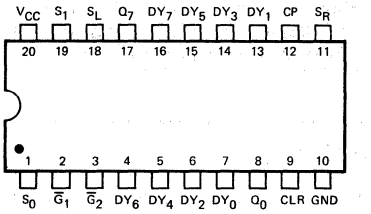


REGISTERS

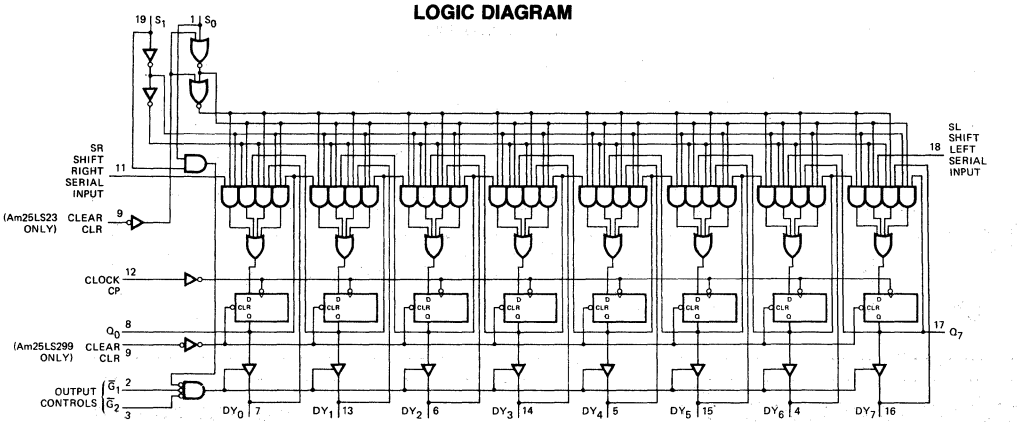
Am25LS23 • Am25LS299

- 8-bit shift/storage registers
- Four modes – Load, Shift Left, Shift Right, Store
- Synchronous clear Am25LS23, Asynchronous clear Am25LS299
- Three-state outputs
- Cascadable left or right
- Common input/output pins

CONNECTION DIAGRAM



LOGIC DIAGRAM

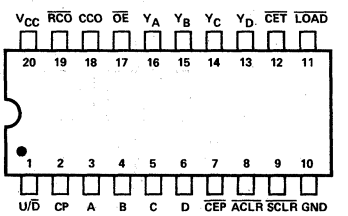


COUNTERS

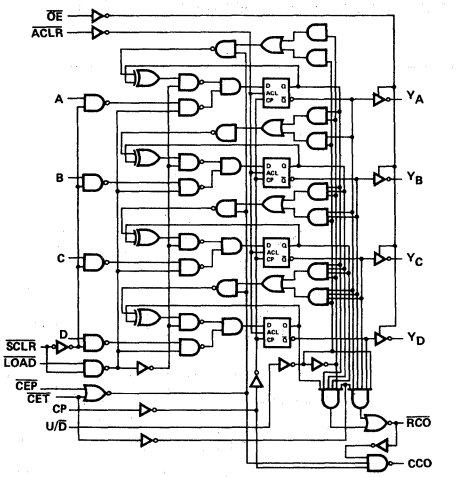
Am25LS2568 • Am25LS2569

- 4-bit synchronous counter, synchronously programmable
- Both synchronous and asynchronous clear inputs
- Three-state counter outputs interface directly with bus organized systems
- Internal look-ahead carry logic and two count enable lines for high-speed cascaded operation
- Ripple carry output for cascading
- Clock carry output for convenient modulo configuration
- Fully buffered outputs

CONNECTION DIAGRAM



LOGIC DIAGRAM Am25LS2569



Am25LS2568 is similar but limits count to binary 9.

REGISTERS

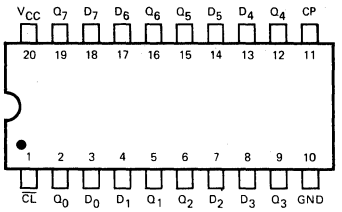
Am25LS273 • Am54LS/74LS273

- 8-bit register with common clock and common clear
- Positive edge-triggered with clock to output delay 15ns typ.
- Buffered common clock and common clear

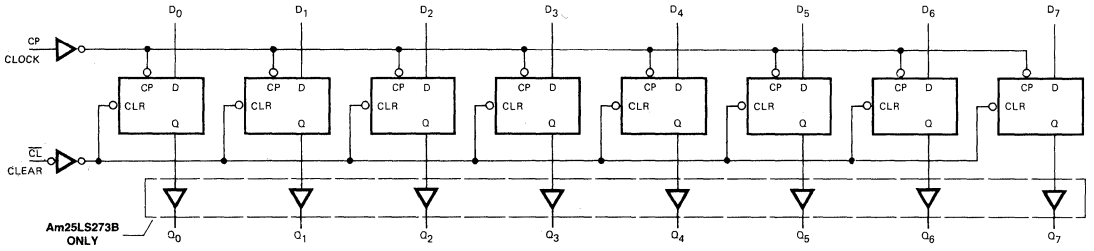
Am25LS273B

- Buffered outputs to eliminate output commutation

CONNECTION DIAGRAM



LOGIC DIAGRAM



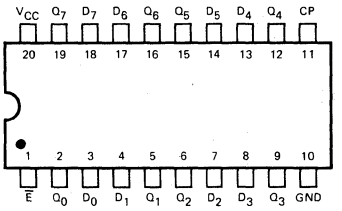
Am25LS377 • Am54LS/74LS377

- 8-bit register with common enable
- Positive edge-triggered with clock to output delay 14ns typ.
- Buffered common clock and common clock enable

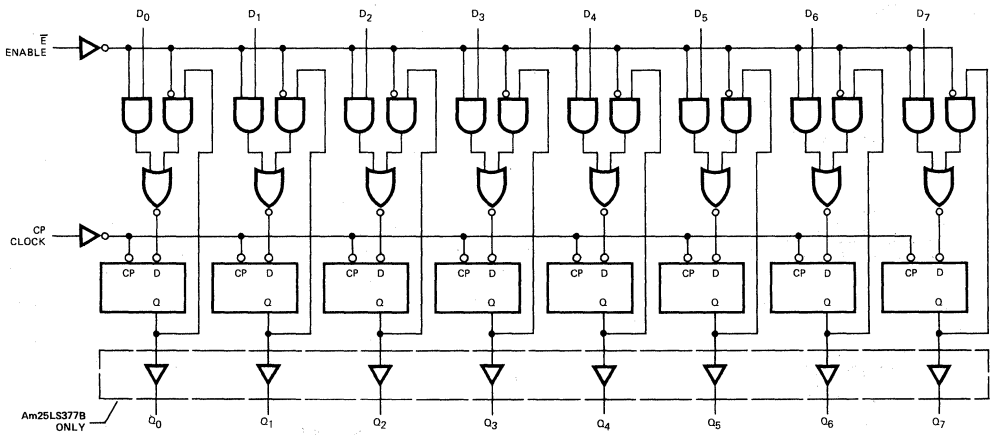
Am25LS377B

- Buffered outputs to eliminate output commutation

CONNECTION DIAGRAM



LOGIC DIAGRAM

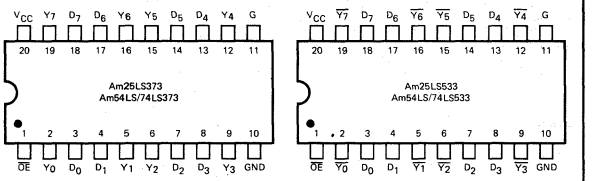


REGISTERS

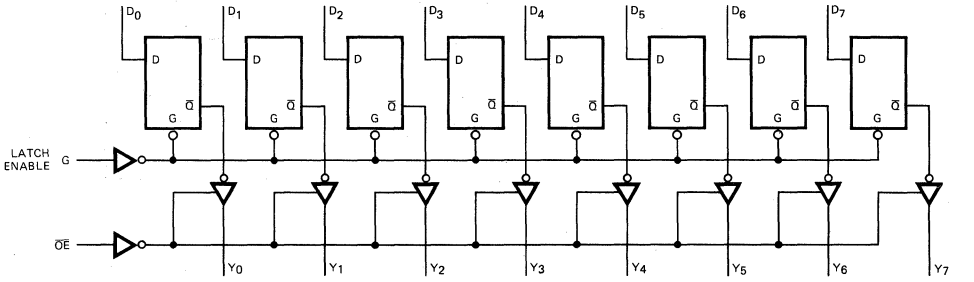
Am25LS/54LS/74LS373 • Am54S/74S373
Am25LS/54LS/74LS533 • Am54S/74S533

- Octal latches with three-state outputs
- Non-inverting outputs on '373
- Inverting outputs on '533
- Hysteresis on latch enable input

CONNECTION DIAGRAMS



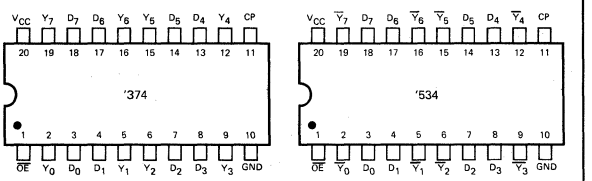
LOGIC DIAGRAM



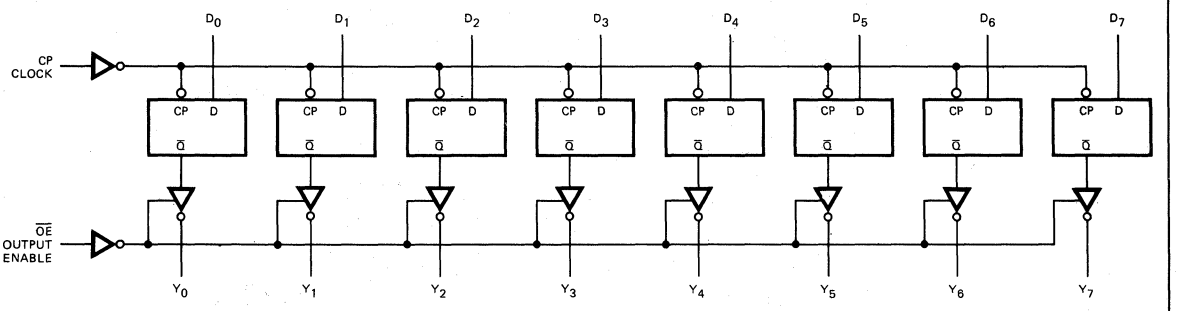
Am25LS/54LS/74LS374 • Am54S/74S374
Am25LS/54LS/74LS534 • Am54S/74S534

- 8-bit registers with three-state outputs
- Buffered common clock and common enable
- Positive edge-triggered, D-type flip-flops
- Non-inverting '374 and inverting '534

CONNECTION DIAGRAMS



LOGIC DIAGRAM



BUS TRANSCEIVERS/DRIVERS

Am25LS/54LS/74LS240/241/242/243/244

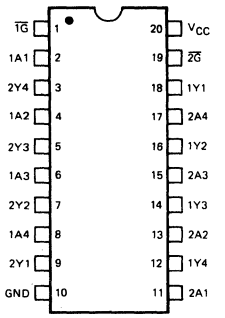
- PNP inputs and three-state outputs
- Am25LS V_{OL} specified at I_{OL} = 48mA
- Data-to-output t_{PD} = 18ns max.
- Input hysteresis 0.4V typ.
- 20-pin package with 0.300" pin row spacing

Am54S/74S240/241/242/243/244

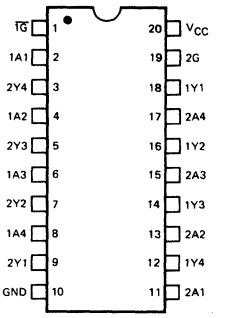
- PNP inputs and three-state outputs
- V_{OL} of 0.55V at 64mA for Am74S; 48mA for Am54S
- Data-to-output t_{PD}: Inverting = 7.0ns max., non-inverting = 9.0ns max.
- Input hysteresis 0.4V typ.
- 20-pin package with 0.300" pin row spacing

CONNECTION DIAGRAMS

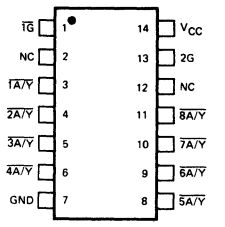
'240



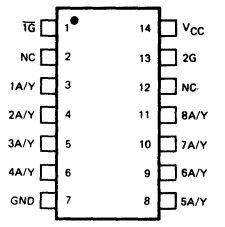
'241



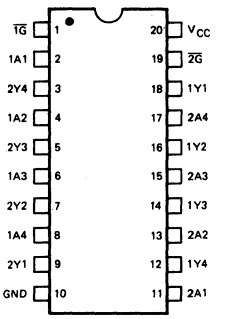
'242



'243

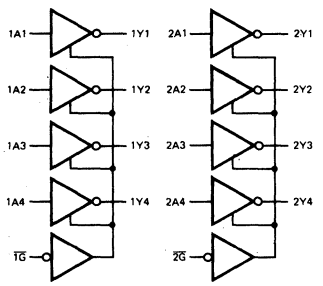


'244

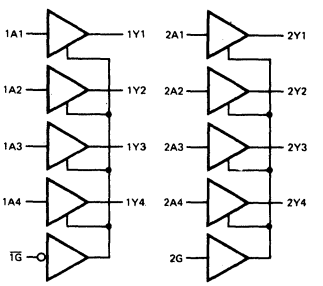


LOGIC DIAGRAMS

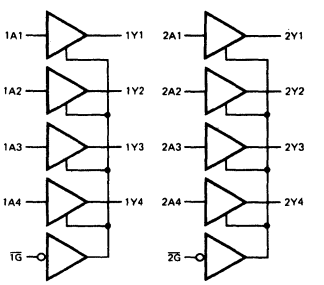
'240



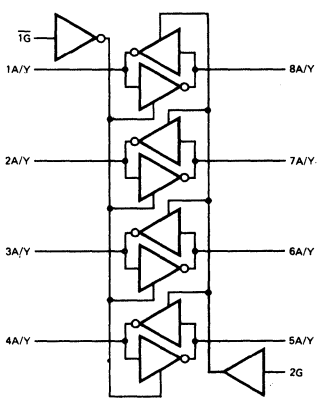
'241



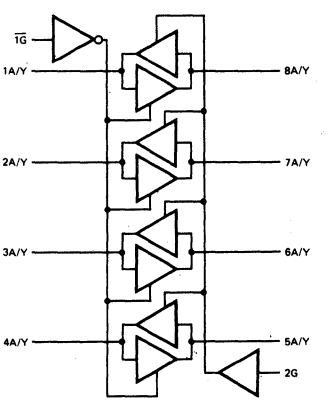
'244



'242



'243



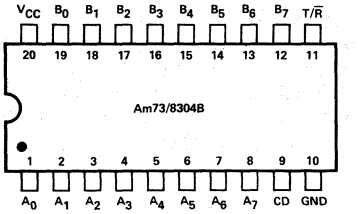
BUS TRANSCEIVERS/DRIVERS

Am73/8303 • Am73/8304B

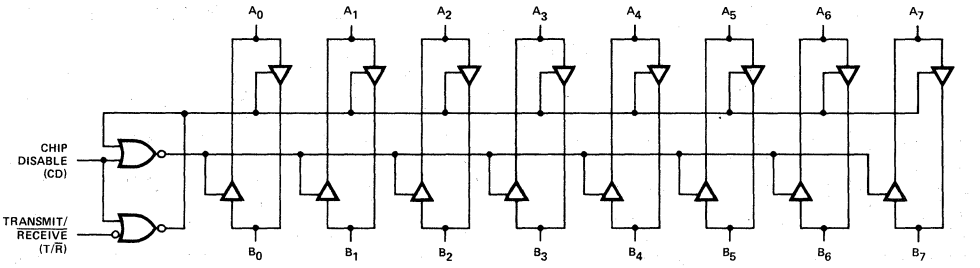
FEATURES

- 8-bit bidirectional data flow reduces system package count
- Three-state inputs/outputs for interfacing with bus-oriented systems
- PNP inputs reduce input loading
- V_{CC} – 1.15V V_{OH} interfaces with TTL, MOS, and CMOS
- 48mA, 300pF bus drive capability
- Transmit/Receive and Chip Disable simplify control logic
- Bus port stays in high-impedance state during power up/down

CONNECTION DIAGRAM



LOGIC DIAGRAM (DM73/8304B)



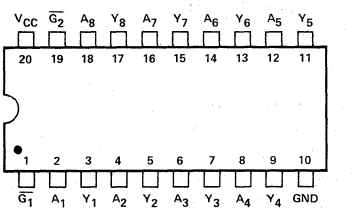
All buffers are inverting on Am73/8303.

Am71/81LS95 • Am71/81LS96
Am71/81LS97 • Am71/81LS98

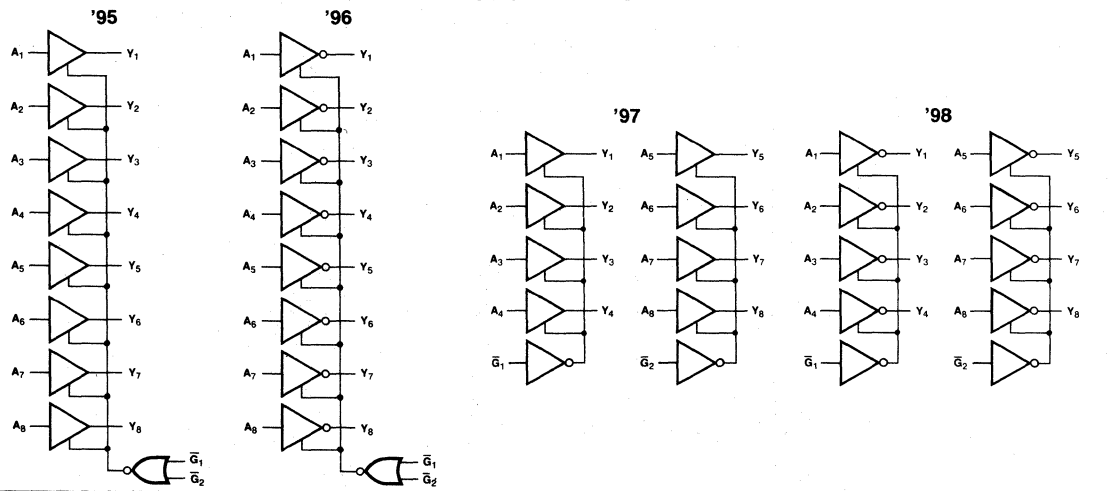
FEATURES

- 8-bit byte/nibble oriented buffers
- Three-state outputs for direct data bus drive
- PNP inputs reduce loading on bus lines
- Inverting and non-inverting versions

CONNECTION DIAGRAM



LOGIC DIAGRAMS



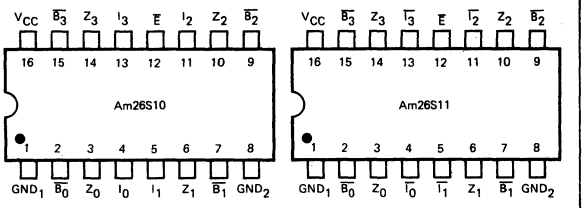
BUS TRANSCEIVERS

Am26S10 • Am26S11

FEATURES

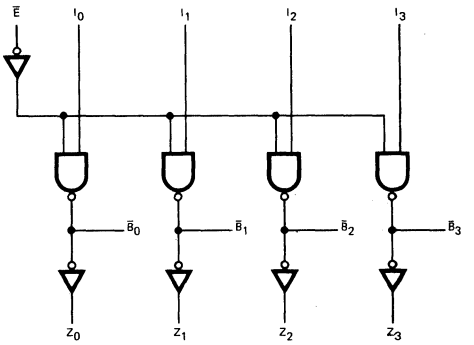
- Input to bus is inverting on Am26S10
- Input to bus is non-inverting on Am26S11
- Quad high-speed open collector bus transceivers
- Driver outputs can sink 100mA at 0.8V max.
- Bus compatible with Am2905, Am2906, and Am2907
- PNP inputs to reduce input loading

CONNECTION DIAGRAMS

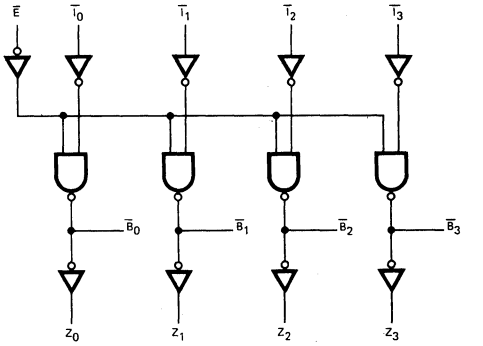


LOGIC DIAGRAMS

Am26S10



Am26S11

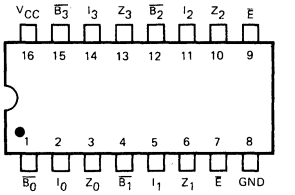


Am26S12 • Am26S12A

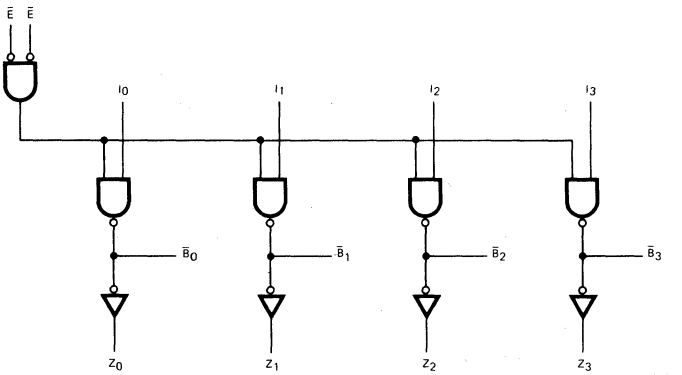
FEATURES

- Quad high-speed bus transceivers
- Driver outputs can sink 100mA at 0.7V typ.
- Open collector outputs
- Am26S12 hysteresis thresholds are 1.4V/2.0V typ.
- Am26S12A hysteresis thresholds are 1.2V/2.25V typ.

CONNECTION DIAGRAM



LOGIC DIAGRAM

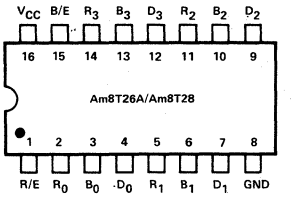


BUS TRANSCEIVERS/DRIVERS

Am8T26A • Am8T28

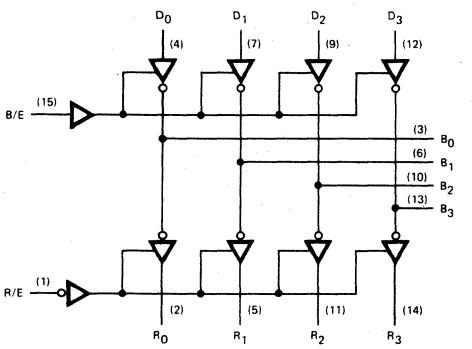
- 48mA driver sink current
- Three-state outputs on driver and receiver
- PNP inputs
- Am8T26A has inverting outputs
- Am8T28 has non-inverting outputs

CONNECTION DIAGRAM

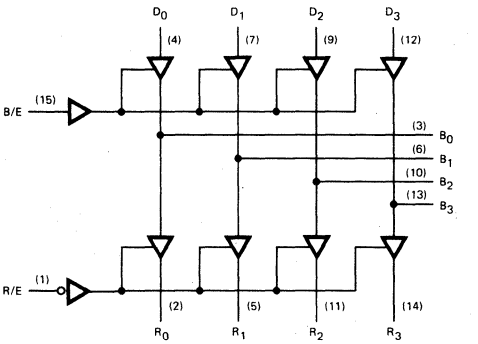


LOGIC DIAGRAMS

Am8T26A



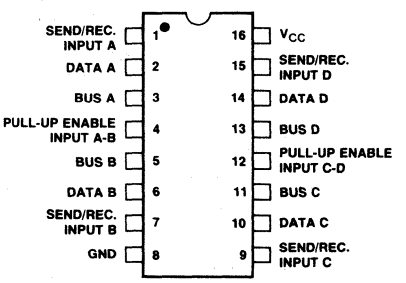
Am8T28



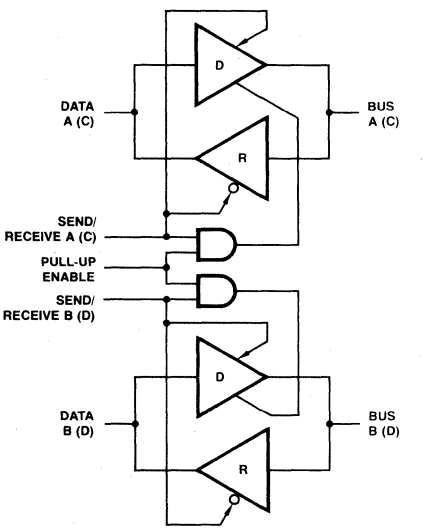
Am3448A

- IEEE-488 quad bidirectional transceiver
- Three-state outputs
- High impedance inputs
- Receiver hysteresis – 600mV typ.
- Fast propagation times – 50-20ns typ.
- TTL compatible receiver outputs
- Single +5 volt supply
- Open collector driver output option
- Power up/power down protection (No invalid information transmitted to bus)
- No bus loading when power is removed from device
- Required termination characteristics provided

CONNECTION DIAGRAM



LOGIC DIAGRAM
1/2 Am3448A



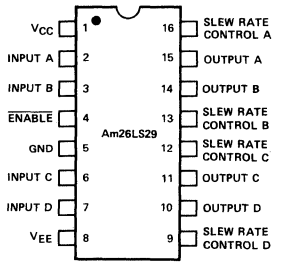
LINE DRIVERS

Am26LS29

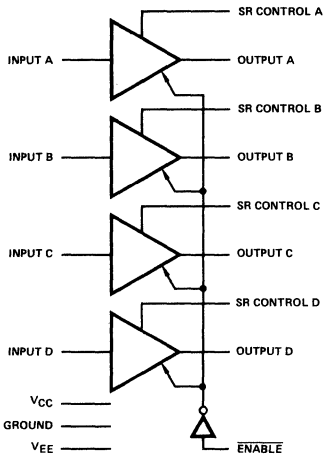
FEATURES

- Four single-ended line drivers in one package
- Meets all requirements of RS-423
- Output short-circuit protection
- Individual rise time control for each output
- 50Ω transmission line drive capability
- High capacitive load drive capability
- Low I_{CC} and I_{EE} power consumption (26mW/driver typ.)
- Three-state outputs for bus oriented systems
- Outputs do not clamp line with power off or in high-impedance state over entire transmission line voltage range of RS-423
- Low current PNP inputs compatible with TTL, MOS and CMOS

CONNECTION DIAGRAM



LOGIC DIAGRAM

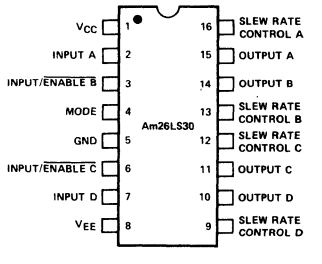


Am26LS30

FEATURES

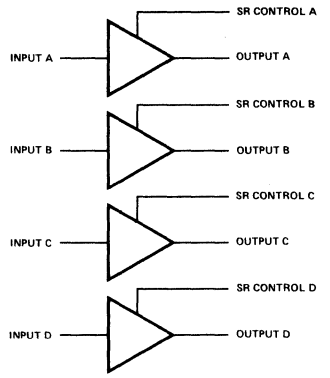
- Dual RS-422 line driver or quad RS-423 line driver
- Driver outputs do not clamp line with power off or in high-impedance state
- Individually three-state drivers when used in differential mode
- Low I_{CC} and I_{EE} power consumption
  - RS-422 differential mode 35mW/driver typ.
  - RS-423 single-ended mode 26mW/driver typ.
- Individual slew rate control for each output
- 50Ω transmission line drive capability (RS-422 into virtual ground)
- Low current PNP inputs compatible with TTL, MOS and CMOS
- High capacitive load drive capability

CONNECTION DIAGRAM

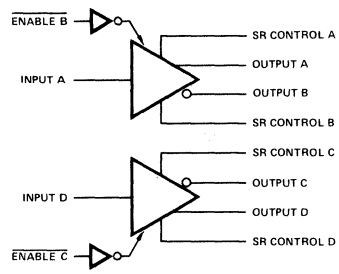


LOGIC DIAGRAMS

Am26LS30 with Mode Control HIGH (RS-423)



Am26LS30 with Mode Control LOW (RS-422)



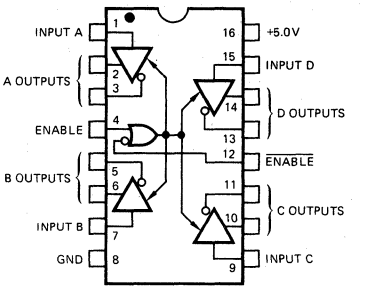
LINE DRIVERS AND RECEIVERS

Am26LS31

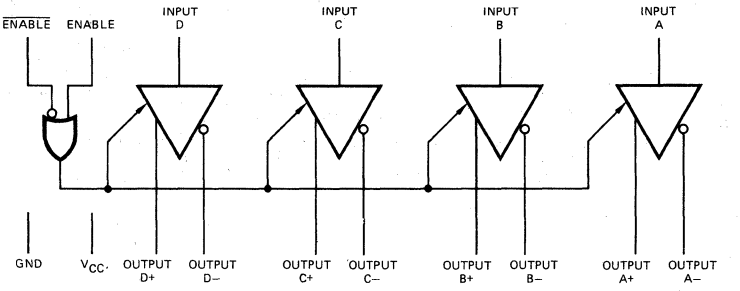
FEATURES

- Four line drivers in one package
- Meets the requirements of EIA standard RS-422
- High output drive capability for 100Ω terminated transmission lines
- Output short-circuit protection
- Complementary outputs
- Outputs won't load line when $V_{CC} = 0$
- Output skew – 2.0ns typ.
- Input to output delay – 12ns
- Operation from single +5V supply

CONNECTION DIAGRAM



LOGIC DIAGRAM

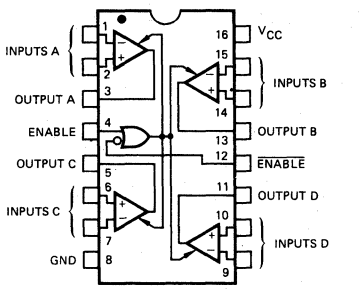


Am26LS32 • Am26LS33

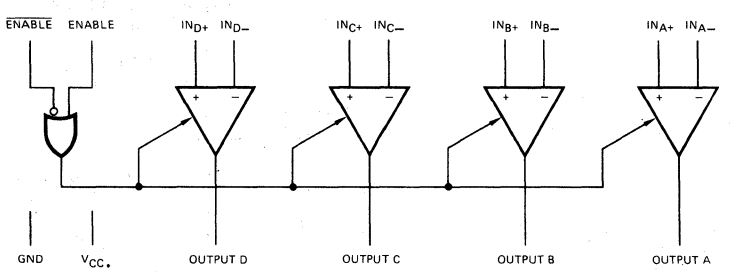
FEATURES

- Quad differential line receivers
- Am26LS32 meets all the requirements of RS-422 and RS-423
- Input voltage range 15V on Am26LS33; 7V on Am26LS32
- ±0.2V sensitivity over the input voltage range on Am26LS32; ±0.5V sensitivity on Am26LS33
- Fail safe input/output relationship. Output always high when inputs are open
- Three-state drive, with choice of complementary output enables, for receiving directly onto a data bus
- 6k minimum input impedance
- 30mV input hysteresis
- Operation from single +5V supply

CONNECTION DIAGRAM



LOGIC DIAGRAM

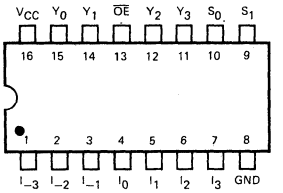


SHIFTERS

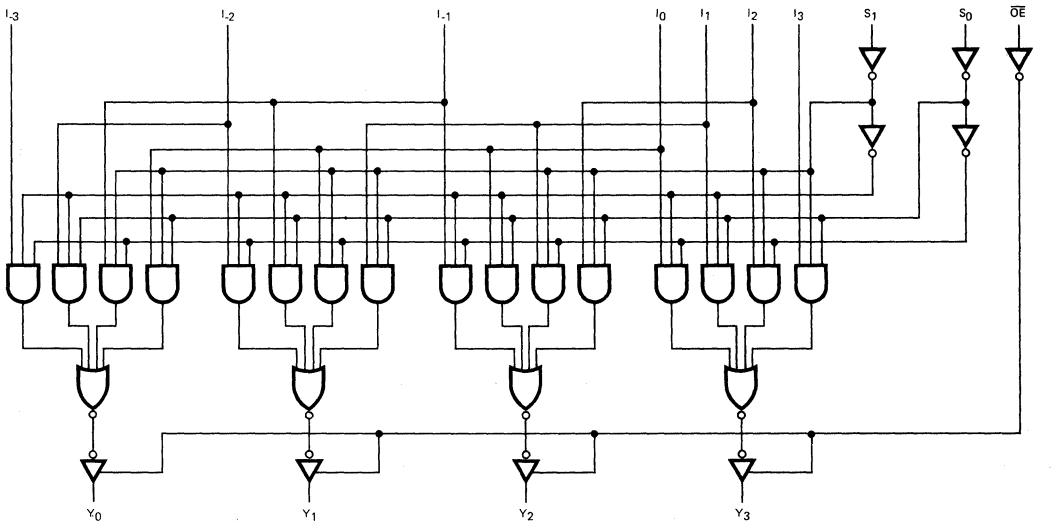
Am25S10

- Shifts 4 bits 0, 1, 2 or 3 places
- Three-state outputs
- t_{PD} 6.5ns typ.
- Easy expansion to any number of bits without propagation delay increase

CONNECTION DIAGRAM



LOGIC DIAGRAM

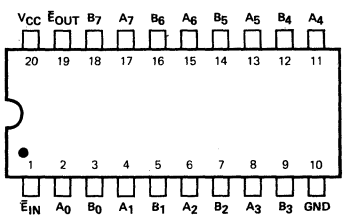


COMPARATORS

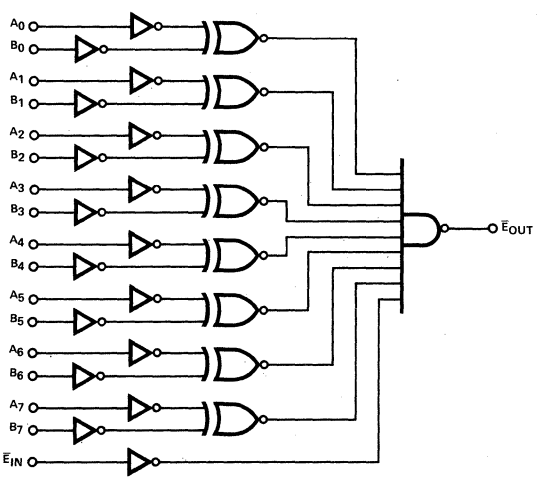
Am25LS2521

- 8-bit byte oriented equal-to comparator
- Easily cascadable using $\overline{E}_{IN}$
- Combinatorial logic with t_{PD} = 9.0ns typ.
- Broad application in conditional gating and microprocessor memory address decoding

CONNECTION DIAGRAM



LOGIC DIAGRAM

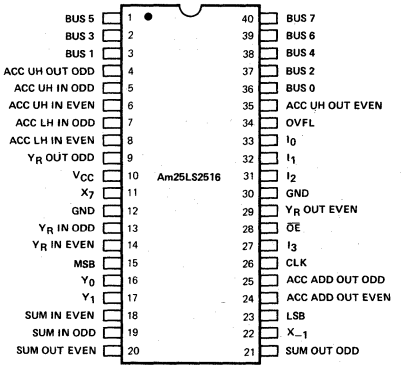


MULTIPLIERS

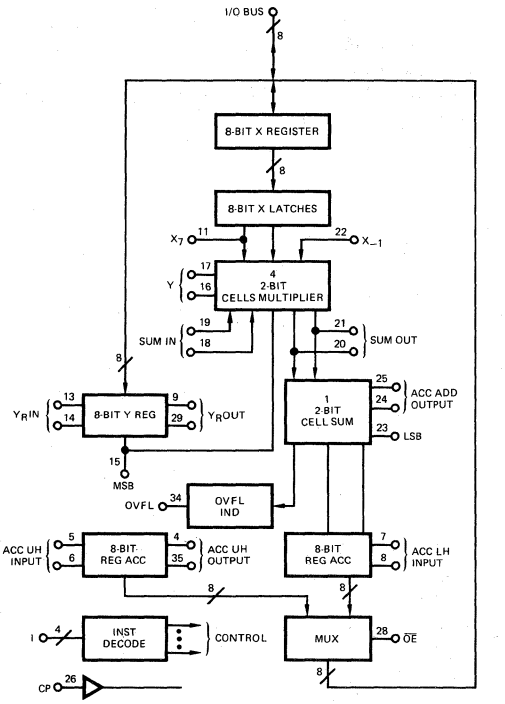
Am25LS2516

- 8-bit by 8-bit serial/parallel multiplier
- Two's complement, two-bit look-ahead carry-save arithmetic
- Microprogrammable – four-bit instruction code for load, multiply, and read operations
- Cascadable, two devices perform full 16-bit multiplication without additional hardware
- 8-bit byte parallel, bidirectional, bussed I/O
- On-chip registers and double length accumulator
- Overflow indicator
- Three-state shared bus input/output lines
- High-speed architecture provides clock rates of 20MHz typ.

CONNECTION DIAGRAM



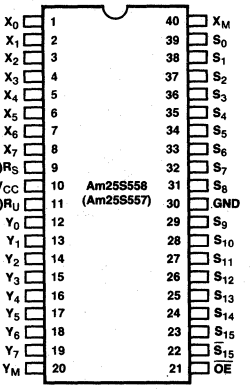
LOGIC DIAGRAM



Am25S557 • Am25S558

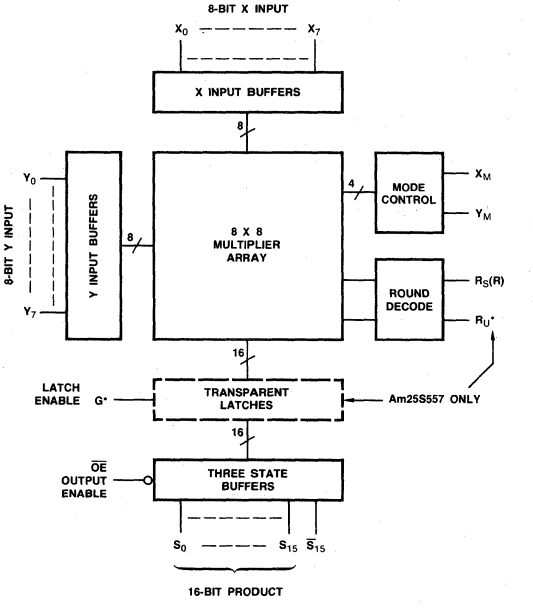
- 8-bit by 8-bit combinatorial multiplier
- Full 8 x 8 multiply in 45ns typ.
- Cascades to 16 x 16 in 110ns typ.
- Unsigned, two's complement or mixed operands
- MSB and MSB outputs for easy expansion
- Implements common rounding algorithms with additional logic
- Three-state outputs
- Transparent 16-bit latch in Am25S557

CONNECTION DIAGRAM



Pin assignments shown are for Am25S558. G and R shown in parentheses are pin assignments for Am25S557.

LOGIC DIAGRAM



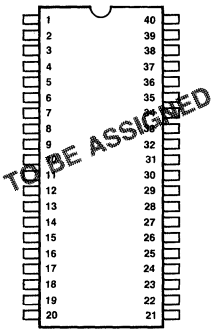
*Pin 11 is G for Am25S557 and R_U for Am25S558.

DYNAMIC MEMORY CONTROLLERS

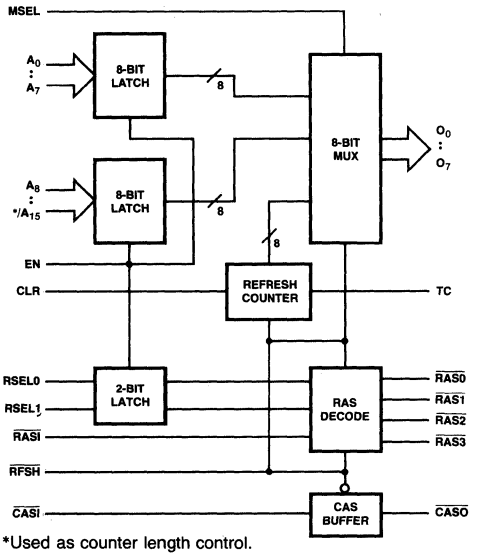
Am25LS2564

- Dynamic memory controller for 16k and 64k MOS RAMs
- 8-bit refresh counter for refresh address generation, has clear input and terminal count output
- Latched input RAS decoder provides four RAS outputs, all active during refresh
- Common chip minimizes speed differential/skew
- 3-port 8-bit address multiplexer with Schottky speed

CONNECTION DIAGRAM



LOGIC DIAGRAM
Dynamic Memory Controller



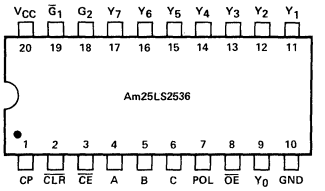
*Used as counter length control.

DECODERS

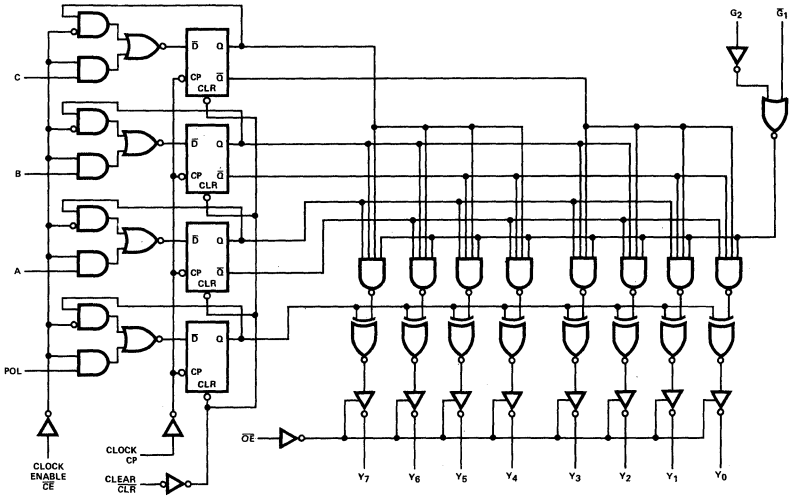
Am25LS2536

- 8-bit decoder/demultiplexer with control storage
- 3-state outputs
- Common clock enable
- Common clear
- Polarity control

CONNECTION DIAGRAM



LOGIC DIAGRAM
8-Bit Decoder/Demultiplexer with Control Storage



3

Microprocessors

Am2901C

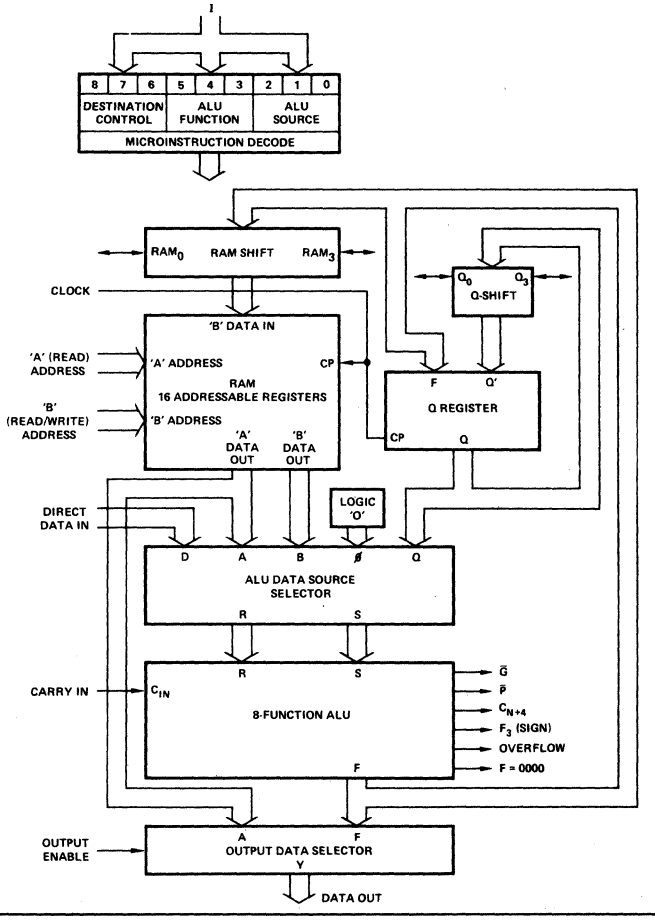
Four-Bit Bipolar Microprocessor Slice

ADVANCED INFORMATION

DISTINCTIVE CHARACTERISTICS

- **Third generation of Am2901 four-bit slice**
  Internal ECL circuitry and state-of-the-art process technology combined to provide fastest version of popular Am2901.
- **Plug-in replacement for Am2901, Am2901A, Am2901B**
  The Am2901C is a pin-for-pin replacement for earlier versions of the device. Only the switching speeds are changed.
- **Improved speed**
  25-30% speed improvement on the critical paths versus the Am2901B

MICROPROCESSOR SLICE BLOCK DIAGRAM



MPR-044

If You Liked Our 2901, You'll Love Our 2903!

Same Powerful Architecture:

- 16 Working Registers in 2-address architecture
- Left-Right shift of Data after ALU
- Auxiliary Register for multiple-length operations
- Expandable to any word length in multiples of four bits
- Carry, Overflow, Zero, and Negative Status Flags

Plus These Added Features:

- Two's Complement and Unsigned Multiply
  The Am2903 performs both signed and unsigned multiplication without requiring any additional hardware. For unsigned n x n-bit multiply, a single microinstruction is repeated n times. For a two's complement multiply, a single microinstruction is executed n-1 times, and a second microinstruction is executed once. Both kinds of multiplies produce 2n bit products.
- Two's Complement Divide
  The Am2903 performs a signed division using a non-restoring algorithm. A 2n bit dividend is divided by an n-bit divisor in n cycles (after justification). An n-bit signed quotient and n-bit signed remainder are produced. Extension to multiple length division is simple. No extra hardware is needed.
- Single and Double Length Normalization
  Both single length words and double length words can be normalized; i.e., shifted up to remove leading zeros or ones. During the normalize instructions, the Am2903 provides special flags signaling the completion of normalization.
- Conversion Between Sign-Magnitude and Two's Complement Notations
  On a single cycle, the Am2903 will switch a word from one notation to the other.
- Increment by One or Two
  On a single cycle, the Am2903 can add either 1 or 2 to a word, depending on carry-in.

If You Didn't Like Our 2901, You'll Love Our 2903!

- Two Data Input Ports
  The Am2903 can operate between any two internal registers, any internal register and an external data bus, or two external data buses.
- Expandable Register File
  The Am29705 hooks directly onto the Am2903 to provide *any number* of working registers, without losing the two-address architecture. You can even go to a three-address system, where on one cycle you operate on two registers and place the result in a third.
- Arithmetic *and* Logical Shifts
  Arithmetic shifts hold the MSB (sign bit) in place and shift the rest of the word *around* the MSB. Logical shifts shift all the bits in the word. The 2903 provides both types of shift.
- Sixteen-Function ALU
  Provides 9 Logic Functions and 7 Arithmetic Functions, twice as many functions as the 2901.
- Parity Generated Internally
  A Parity Generator operates on the ALU output and is cascaded between devices, so that a single pin contains parity across the entire ALU output.

And If You Don't Quite Like Our 2903, You'll Definitely Love Our Am29203!

- BCD Arithmetic
  The Am29203 includes special functions for BCD add and subtract, as well as conversion between binary and BCD notations.
- A Byte Better
  The Am29203 is designed to efficiently handle byte operations with a minimum of external logic.
- Both Data Lines Bidirectional
- Decrement by 1 or 2 Instruction
- RAM is enabled only if instruction execution is enabled.

Am2903 • Am29203

The Superslice®

DISTINCTIVE CHARACTERISTICS

- Expandable Register File –
  Like the Am2901, the Am2903/29203 contains 16 internal working registers arranged in a two-address architecture. But the Am2903/29203 includes the necessary "hooks" to expand the register file externally to any number of registers.
- Built-in Multiplication Logic –
  Performing multiplication with the Am2901A requires a few external gates – these gates are contained on-chip in the Am2903/29203. Three special instructions are used for unsigned multiplication, two's complement multiplication and the last cycle of a two's complement multiplication.
- Built-in Division Logic –
  The Am2903/29203 contains all logic and interconnects for execution of a non-restoring, multiple-length division with correction of the quotient.
- Built-in Normalization Logic –
  The Am2903/29203 can simultaneously shift the Q Register and count in a working register. Thus, the mantissa and exponent of a floating-point number can be developed using a single microcycle per shift. Status flags indicate when the operation is complete.
- Built-in Parity Generation Circuitry –
  The Am2903/29203 can supply parity across the entire ALU output for use in error detection.
- Built-in Sign Extension Circuitry –
  To facilitate operation on different length two's complement numbers, the Am2903/29203 provides the capability to extend the sign at any slice boundary.
- BCD Arithmetic (Am29203 only) –
  Automatic BCD add and subtract and conversion between binary and BCD.
- Improved Byte Handling (Am29203 only) –
  Zero detection and register writing can be performed on a single byte rather than the whole word.
- Two Bidirectional Data Lines (Am29203 only) –

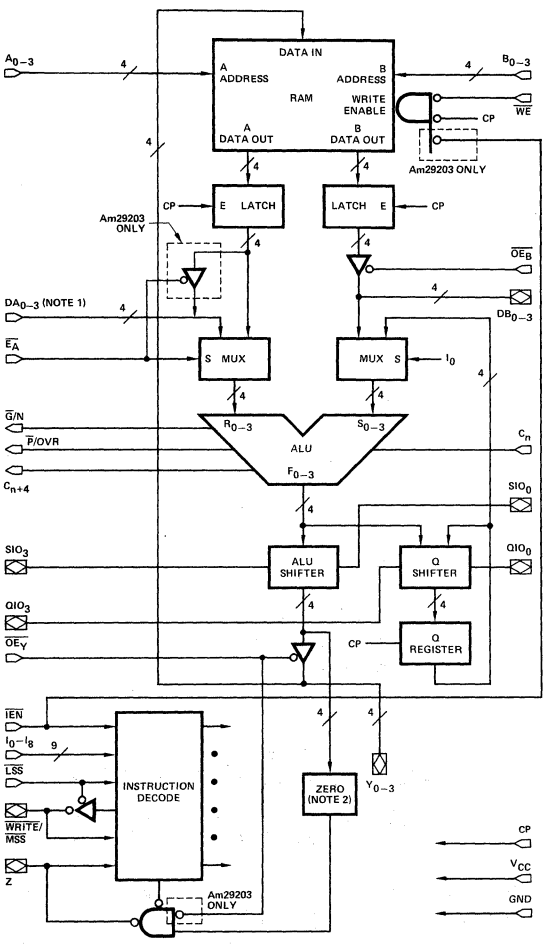
TABLE OF CONTENTS

ALU Functions 3-4
Block Diagram 3-5
Special Functions 3-8
Pin Definitions 3-9
Pin Connections 3-10
Burn-in Circuit 3-10
DC Characteristics 3-12
Switching Characteristics 3-13
Applications
- Expansion 3-16
- Normalization 3-18
- Multiplication 3-21
- Division 3-23
- Byte Swap 3-25
- Memory Expansion 3-27

GENERAL DESCRIPTION

The Am2903 is a four-bit expandable bipolar microprocessor slice. The Am2903 performs all functions performed by the industry standard Am2901 and, in addition, provides a number of significant enhancements that are especially useful in arithmetic-oriented processors. Infinitely expandable memory and three-port, three-address architecture are provided by the Am2903. In addition to its complete arithmetic and logic instruction set, the Am2903 provides a special set of instructions which facilitate the implementation of multiplication, division, normalization, and other previously time-consuming operations. The Am29203 is a similar device, but has additional I/O capability, more special instructions and will be at least 30% faster.

BLOCK DIAGRAM



Notes: 1. DA_{0-3} is input only on Am2903, but is I/O port on Am29203.
2. On Am2903, zero logic is connected to Y, after the $\overline{OE}_Y$ buffer.

MPR-030

ARCHITECTURE OF THE Am2903 AND Am29203

The Am2903/29203 is a high-performance, cascadable, four-bit bipolar microprocessor slice designed for use in CPUs, peripheral controllers, microprogrammable machines, and numerous other applications. The microinstruction flexibility of the Am2903/29203 allows the efficient emulation of almost any digital computing machine. The nine-bit microinstruction selects the ALU sources, function and destination. The Am2903/29203 is cascadable with full lookahead or ripple carry, has 3-state outputs, and provides various ALU status flag outputs. Advanced Low-Power Schottky processing is used to fabricate this 48-pin LSI circuit.

All data paths within the device are four bits wide. As shown in the block diagram, the device consists of a 16-word by 4-bit, two-port RAM with latches on both output ports, a high-performance ALU and shifter, a multi-purpose Q Register with shifter input, and a nine-bit instruction decoder.

Two-Port RAM

Any two RAM words addressed at the A and B address ports can be read simultaneously at the respective RAM A and B output ports. Identical data appear at the two output ports when the same address is applied to both address ports. The latches at the RAM output ports are transparent when the clock input, CP, is HIGH and they hold the RAM output data when CP is LOW. Under control of the $\overline{OE}_B$ three-state output enable, RAM data can be read directly at the Am2903 DB I/O port. On the Am29203, E_A provides the same feature at the DA port.

External data at the Am2903/29203 Y I/O port can be written directly into the RAM, or ALU shifter output data can be enabled onto the Y I/O port and entered into the RAM. Data is written into the RAM at the B address when the write enable input, $\overline{WE}$, is LOW and the clock input, CP, is LOW.

Arithmetic Logic Unit

The Am2903 high-performance ALU can perform seven arithmetic and nine logic operations on two 4-bit operands. Multiplexers at the ALU inputs provide the capability to select various pairs of ALU source operands. The $\overline{E}_A$ input selects either the DA external data input or RAM output port A for use as one ALU operand and the $\overline{OE}_B$ and I_0 inputs select RAM output port B, DB external data input, or the Q Register content for use as the second ALU operand. Also, during some ALU operations, zeroes are forced at the ALU operand inputs. Thus, the Am2903 ALU can operate on data from two external sources, from an internal and external source, or from two internal sources. Table I shows all possible pairs of ALU source operands as a function of the $\overline{E}_A$, $\overline{OE}_B$, and I_0 inputs.

When instruction bits I_4, I_3, I_2, I_1, and I_0 are LOW, the Am2903 executes special functions. Table 4 defines these special functions and the operation which the ALU performs for each. When the Am2903 executes instructions other than the nine special functions, the ALU operation is determined by instruction bits I_4, I_3, I_2, and I_1. Table 2 defines the ALU operation as a function of these four instruction bits. The Am29203 ALU is identical, but executes 16 special instructions.

Am2903/29203s may be cascaded in either a ripple carry or lookahead carry fashion. When a number of Am2903/29203s are cascaded, each slice must be programmed to be a most significant slice (MSS), intermediate slice (IS), or least significant slice (LSS) of the array. The carry generate, $\overline{G}$, and carry propagate, $\overline{P}$, signals required for a lookahead carry scheme are generated by the Am2903/29203 and are available as outputs of the least significant and intermediate slices.

TABLE 1. ALU OPERAND SOURCES

$\overline{E_A}$	I_0	$\overline{OE_B}$	ALU Operand R	ALU Operand S
L	L	L	RAM Output A	RAM Output B
L	L	H	RAM Output A	DB_{0-3}
L	H	X	RAM Output A	Q Register
H	L	L	DA_{0-3}	RAM Output B
H	L	H	DA_{0-3}	DB_{0-3}
H	H	X	DA_{0-3}	Q Register

L = LOW H = HIGH X = Don't Care

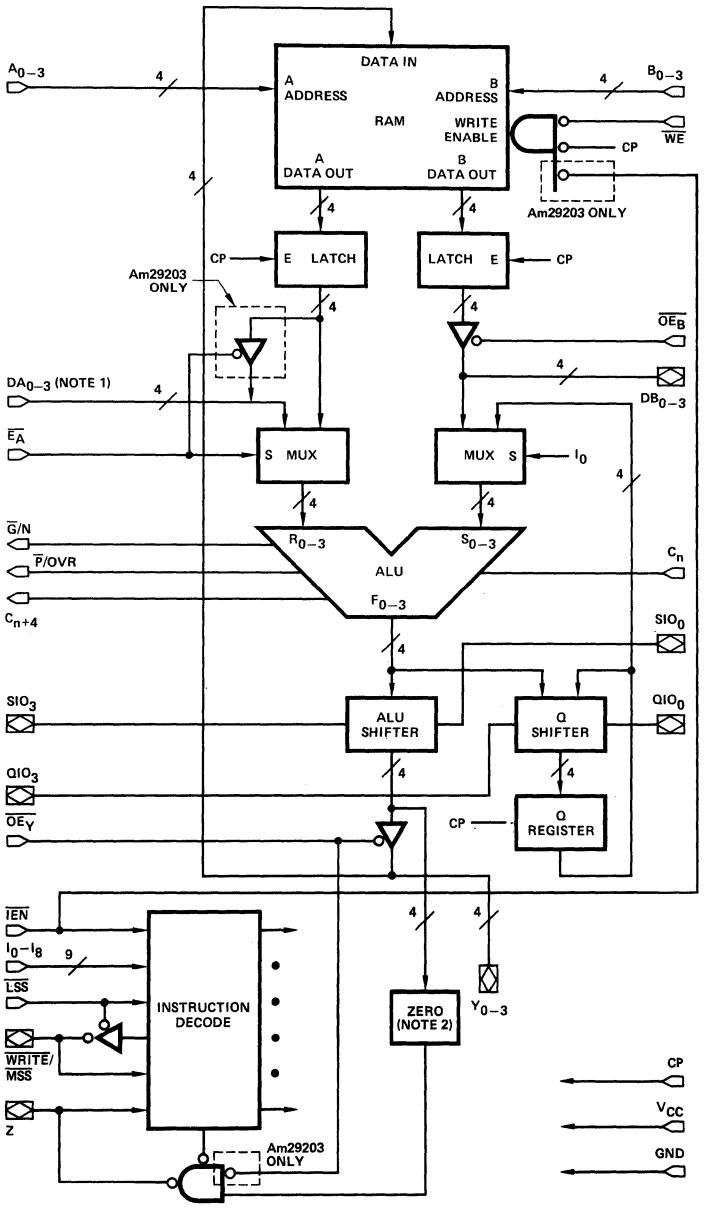
TABLE 2. ALU FUNCTIONS

I_4	I_3	I_2	I_1	I_0	ALU Functions
L	L	L	L	L	Special Functions
L	L	L	L	H	F_i = HIGH
L	L	L	H	X	F = S Minus R Minus 1 Plus C_n
L	L	H	L	X	F = R Minus S Minus 1 Plus C_n
L	L	H	H	X	F = R Plus S Plus C_n
L	H	L	L	X	F = S Plus C_n
L	H	L	H	X	F = S Plus C_n
L	H	H	L	L	Reserved Special Functions
L	H	H	L	H	F = R Plus C_n
L	H	H	H	L	Reserved Special Functions
L	H	H	H	H	F = R Plus C_n
H	L	L	L	L	Reserved Special Functions
H	L	L	L	H	F_i = LOW
H	L	L	H	X	F_i = R_i AND S_i
H	L	H	L	X	F_i = R_i EXCLUSIVE NOR S_i
H	L	H	H	X	F_i = R_i EXCLUSIVE OR S_i
H	H	L	L	X	F_i = R_i AND S_i
H	H	L	H	X	F_i = R_i NOR S_i
H	H	H	L	X	F_i = R_i NAND S_i
H	H	H	H	X	F_i = R_i OR S_i

L = LOW H = HIGH i = 0 to 3
X = LOW or HIGH

The Am2903/29203 also generates a carry-out signal, C_{n+4}, which is generally available as an output of each slice. Both the carry-in, C_n, and carry-out, C_{n+4}, signals are active HIGH. The ALU generates two other status outputs. These are negative, N, and overflow, OVR. The N output is generally the most significant (sign) bit of the ALU output and can be used to determine positive or negative results. The OVR output indicates that the arithmetic operation being performed exceeds the available two's complement number range. The N and OVR signals are available as outputs of the most significant slice. Thus, the multipurpose $\overline{G}$/N and $\overline{P}$/OVR outputs indicate $\overline{G}$ and $\overline{P}$ at the least significant and intermediate slices, and sign and overflow at the most significant slice. To some extent, the meaning of the C_{n+4}, $\overline{P}$/OVR, and $\overline{G}$/N signals vary with the ALU function being performed. Refer to Table 5 for an exact definition of these four signals as a function of the Am2903/29203 instruction.

BLOCK DIAGRAM

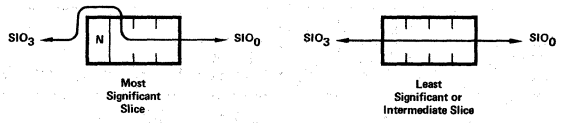


Notes: 1. DA_{0-3} is input only on Am2903, but is I/O port on Am29203.
2. On Am2903, zero logic is connected to Y, after the $\overline{OE}_Y$ buffer.

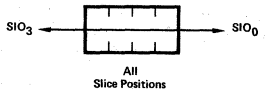
MPR-030

ALU Shifter

Under instruction control, the ALU shifter passes the ALU output (F) non-shifted, shifts it up one bit position (2F), or shifts it down one bit position (F/2). Both arithmetic and logical shift operations are possible. An arithmetic shift operation shifts data around the most significant (sign) bit position of the most significant slice, and a logical shift operation shifts data through this bit position (see Figure A). SIO_0 and SIO_3 are bidirectional serial shift inputs/outputs. During a shift-up operation, SIO_0 is generally a serial shift input and SIO_3 a serial shift output. During a shift-down operation, SIO_3 is generally a serial shift input and SIO_0 a serial shift output.



Am2903/29203 Arithmetic Shift Path



Am2903/29203 Logical Shift Path

Figure A. MPH-031

To some extent, the meaning of the SIO_0 and SIO_3 signals is instruction dependent. Refer to Tables 3 and 4 for an exact definition of these pins.

The ALU shifter also provides the capability to sign extend at slice boundaries. Under instruction control, the SIO_0 (sign) input can be extended through Y_0, Y_1, Y_2, Y_3 and propagated to the SIO_3 output.

A cascadable, five-bit parity generator/checker is designed into the Am2903/29203 ALU shifter and provides ALU error detection capability. Parity for the F_0, F_1, F_2, F_3 ALU outputs and SIO_3 input is generated and, under instruction control, is made available at the SIO_0 output. Refer to the Am2903/29203 applications section for a more detailed description of the Am2903/29203 sign extension and parity generation/checking capability.

The instruction inputs determine the ALU shifter operation. Table 4 defines the special functions and the operation the ALU shifter performs for each. When the Am2903/29203 executes instructions other than the special functions, the ALU shifter operation is determined by instruction bits $I_8I_7I_6I_5$. Table 3 defines the ALU shifter operation as a function of these four bits.

Q Register

The Q Register is an auxiliary four-bit register which is clocked on the LOW-to-HIGH transition of the CP input. It is intended primarily for use in multiplication and division operations; however, it can also be used as an accumulator or holding register for some applications. The ALU output, F, can be loaded into the Q Register, and/or the Q Register can be selected as the source for the ALU S operand. The shifter at the input to the Q Register provides the capability to shift the Q Register contents up one bit position (2Q) or down one bit position (Q/2). Only logical shifts are performed. QIO_0 and QIO_3 are bidirectional shift serial inputs/outputs. During a Q Register shift-up operation, QIO_0 is a serial shift input and QIO_3 is a serial shift output. During a shift-down operation, QIO_3 is a serial shift input and QIO_0 is a serial shift output.

Double-length arithmetic and logical shifting capability is provided by the Am2903/29203. The double-length shift is performed by connecting QIO_3 of the most significant slice to SIO_0 of the least significant slice, and executing an instruction which shifts both the ALU output and the Q Register.

The Q Register and shifter are controlled by the instruction inputs. Table 4 defines the Am2903/29203 special functions and the operations which the Q Register and shifter perform for each. When the Am2903/29203 executes instructions other than the special functions, the Q Register and shifter operation is controlled by instruction bits $I_8I_7I_6I_5$. Table 3 defines the Q Register and shifter operation as a function of these four bits.

Output Buffers

The DB and Y ports are bidirectional I/O ports driven by three-state output buffers with external output enable controls. On the Am29203, the DA port is also bidirectional. The Y output buffers are enabled when the $\overline{OE_Y}$ input is LOW and are in the high impedance state when $\overline{OE_Y}$ is HIGH. The DB output buffers are enabled when the $\overline{OE_B}$ input is LOW and the DA buffers are enabled when $\overline{E_A}$ is LOW. (On the Am2903 DA is input only; the pins are never outputs.)

The zero, Z, pin is an open collector input/output that can be wire-OR'ed between slices. As an output it can be used as a zero detect status flag and generally indicates that the $Y_{0\text{-}3}$ pins are all LOW. To some extent the meaning of this signal varies with the instruction being performed. Refer to Table 5 for an exact definition of this signal as a function of the Am2903 and Am29203 instructions. On the Am29203, the Z pin will be HIGH if $\overline{OE_Y}$ is HIGH, allowing zero detection on less than the full word.

TABLE 3. ALU DESTINATION CONTROL FOR I_0 OR I_1 OR I_2 OR I_3 OR I_4 = HIGH, $\overline{IEN}$ = LOW.

I_8	I_7	I_6	I_5	Hex Code	ALU Shifter Function	SIO_3		Y_3		Y_2		Y_1	Y_0	SIO_0	$\overline{Write}$	Q Reg & Shifter Function	QIO_3	QIO_0
						Most Sig. Slice	Other Slices	Most Sig. Slice	Other Slices	Most Sig. Slice	Other Slices							
L	L	L	L	0	Arith. F/2→Y	Input	Input	F_3	SIO_3	SIO_3	F_3	F_2	F_1	F_0	L	Hold	Hi-Z	Hi-Z
L	L	L	H	1	Log. F/2→Y	Input	Input	SIO_3	SIO_3	F_3	F_3	F_2	F_1	F_0	L	Hold	Hi-Z	Hi-Z
L	L	H	L	2	Arith. F/2→Y	Input	Input	F_3	SIO_3	SIO_3	F_3	F_2	F_1	F_0	L	Log. Q/2→Q	Input	Q_0
L	L	H	H	3	Log. F/2→Y	Input	Input	SIO_3	SIO_3	F_3	F_3	F_2	F_1	F_0	L	Log. Q/2→Q	Input	Q_0
L	H	L	L	4	F→Y	Input	Input	F_3	F_3	F_2	F_2	F_1	F_0	Parity	L	Hold	Hi-Z	Hi-Z
L	H	L	H	5	F→Y	Input	Input	F_3	F_3	F_2	F_2	F_1	F_0	Parity	H	Log. Q/2→Q	Input	Q_0
L	H	H	L	6	F→Y	Input	Input	F_3	F_3	F_2	F_2	F_1	F_0	Parity	H	F→Q	Hi-Z	Hi-Z
L	H	H	H	7	F→Y	Input	Input	F_3	F_3	F_2	F_2	F_1	F_0	Parity	L	F→Q	Hi-Z	Hi-Z
H	L	L	L	8	Arith. 2F→Y	F_2	F_3	F_3	F_2	F_1	F_1	F_0	SIO_0	Input	L	Hold	Hi-Z	Hi-Z
H	L	L	H	9	Log. 2F→Y	F_3	F_3	F_2	F_2	F_1	F_1	F_0	SIO_0	Input	L	Hold	Hi-Z	Hi-Z
H	L	H	L	A	Arith. 2F→Y	F_2	F_3	F_3	F_2	F_1	F_1	F_0	SIO_0	Input	L	Log. 2Q→Q	Q_3	Input
H	L	H	H	B	Log. 2F→Y	F_3	F_3	F_2	F_2	F_1	F_1	F_0	SIO_0	Input	L	Log. 2Q→Q	Q_3	Input
H	H	L	L	C	F→Y	F_3	F_3	F_3	F_3	F_2	F_2	F_1	F_0	Hi-Z	H	Hold	Hi-Z	Hi-Z
H	H	L	H	D	F→Y	F_3	F_3	F_3	F_3	F_2	F_2	F_1	F_0	Hi-Z	H	Log. 2Q→Q	Q_3	Input
H	H	H	L	E	SIO_0→Y_0, Y_1, Y_2, Y_3	SIO_0	SIO_0	SIO_0	SIO_0	SIO_0	SIO_0	SIO_0	SIO_0	Input	L	Hold	Hi-Z	Hi-Z
H	H	H	H	F	F→Y	F_3	F_3	F_3	F_3	F_2	F_2	F_1	F_0	Hi-Z	L	Hold	Hi-Z	Hi-Z

Parity = $F_3 \veebar F_2 \veebar F_1 \veebar F_0 \veebar SIO_3$
$\veebar$ = Exclusive OR

L = LOW
H = HIGH

Hi-Z = High Impedance

TABLE 4. SPECIAL FUNCTIONS: $I_0 = I_1 = I_2 = I_3 = I_4$ = LOW, $\overline{IEN}$ = LOW

I_8	I_7	I_6	I_5	Hex Code	Available On	Special Function	ALU Function	ALU Shifter Function	SIO_3 Most Sig. Slice	SIO_3 Other Slices	SIO_0	Q Reg & Shifter Function	QIO_3	QIO_0	$\overline{WRITE}$
L	L	L	L	0	Am2903 Am29203	Unsigned Multiply	F= S+C_n if Z=L F=R+S+C_n if Z=H	Log. F/2→Y (Note 1)	Hi-Z	Input	F_0	Log. Q/2→Q	Input	Q_0	L
L	L	L	H	1	Am29203										
L	L	H	L	2	Am2903 Am29203	Two's Complement Multiply	F=S+C_n if Z=L F=R+S+C_n if Z=H	Log. F/2→Y (Note 2)	Hi-Z	Input	F_0	Log. Q/2→Q	Input	Q_0	L
L	L	H	H	3	Am29203										
L	H	L	L	4	Am2903 Am29203	Increment by One or Two	F=S+1+C_n	F→Y	Input	Input	Parity	Hold	Hi-Z	Hi-Z	L
L	H	L	H	5	Am2903 Am29203	Sign/Magnitude-Two's Complement	F=S+C_n if Z=L F=$\overline{S}$+C_n if Z=H	F→Y (Note 3)	Input	Input	Parity	Hold	Hi-Z	Hi-Z	L
L	H	H	L	6	Am2903 Am29203	Two's Complement Multiply, Last Cycle	F=S+C_n if Z=L F=S−R−1+C_n if Z=H	Log. F/2→Y (Note 2)	Hi-Z	Input	F_0	Log. Q/2→Q	Input	Q_0	L
L	H	H	H	7	Am29203										
H	L	L	L	8	Am2903 Am29203	Single Length Normalize	F=S+C_n	F→Y	F_3	F_3	Hi-Z	Log. 2Q→Q	Q_3	Input	L
H	L	L	H	9	Am29203										
H	L	H	L	A	Am2903 Am29203	Double Length Normalize and First Divide Op.	F=S+C_n	Log 2F→Y	$R_3 \forall F_3$	F_3	Input	Log. 2Q→Q	Q_3	Input	L
H	L	H	H	B	Am29203										
H	H	L	L	C	Am2903 Am29203	Two's Complement Divide	F=S+R+C_n if Z=L F=S−R−1+C_n if Z=H	Log. 2F→Y	$R_3 \forall F_3$	F_3	Input	Log. 2Q→Q	Q_3	Input	L
H	H	L	H	D	Am29203										
H	H	H	L	E	Am2903 Am29203	Two's Complement Divide, Correction and Remainder	F=S+R+C_n if Z=L F=S−R−1+C_n if Z=H	F→Y	F_3	F_3	Hi-Z	Log. 2Q→Q	Q_3	Input	L
H	H	H	H	F	Am29203										

NOTES: 1. At the most significant slice only, the C_{n+4} signal is internally gated to the Y_3 output.
2. At the most significant slice only, $F_3 \forall$ OVR is internally gated to the Y_3 output.
3. At the most significant slice only, $S_3 \forall F_3$ is generated at the Y_3 output.

L = LOW
H = HIGH
X = Don't Care

Hi-Z = High Impedance
$\forall$ = Exclusive OR
Parity = $SIO_3 \forall F_3 \forall F_2 \forall F_1 \forall F_0$

Instruction Decoder

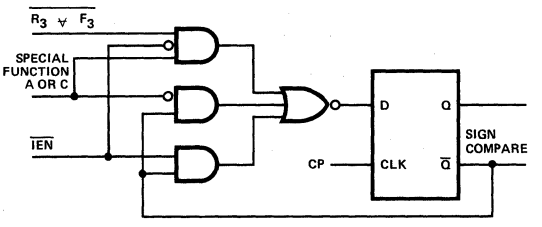
The Instruction Decoder generates required internal control signals as a function of the nine Instruction inputs, I_{0-8}; the Instruction Enable input, $\overline{IEN}$; the $\overline{LSS}$ input; and the $\overline{WRITE}/\overline{MSS}$ input/output.

The $\overline{WRITE}$ output is LOW when an instruction which writes data into the RAM is being executed. Refer to Tables 3 and 4 for a definition of the $\overline{WRITE}$ output as a function of the Am2903 instruction inputs.

On the Am2903, when $\overline{IEN}$ is HIGH, the $\overline{WRITE}$ output is forced HIGH and the Q Register and Sign Compare Flip-Flop contents are preserved. When $\overline{IEN}$ is LOW, the $\overline{WRITE}$ output is enabled and the Q Register and Sign Compare Flip-Flop can be written according to the Am2903 instruction. The Sign Compare Flip-Flop is an on-chip flip-flop which is used during an Am2903 divide operation (see Figure B). On the Am29203, $\overline{IEN}$ controls internal writing, but does not affect $\overline{WRITE}$. The $\overline{IEN}$ signal can then be controlled separately at each chip to facilitate byte operations.

Programming the Am2903/29203 Slice Position

Tying the $\overline{LSS}$ input LOW programs the slice to operate as a least significant slice (LSS) and enables the $\overline{WRITE}$ output signal onto the $\overline{WRITE}/\overline{MSS}$ bidirectional I/O pin. When $\overline{LSS}$ is tied HIGH, the $\overline{WRITE}/\overline{MSS}$ pin becomes an input pin; tying the $\overline{WRITE}/\overline{MSS}$ pin HIGH programs the slice to operate as an intermediate slice (IS) and tying it LOW programs the slice to operate as a most significant slice (MSS). The $\overline{W}/\overline{MSS}$ pin must be tied HIGH through a resistor. $\overline{W}/\overline{MSS}$ and $\overline{LSS}$ should not be connected together. See Figure 2 of applications.



The sign compare signal appears at the Z output of the most significant slice during special functions C, D and E, F. Refer to Table 5.

Figure B. Sign Compare Flip-Flop.

MPR-032

TABLE 5. Am2903/29203 STATUS OUTPUTS

(Hex) $I_8I_7I_6I_5$	(Hex) $I_4I_3I_2I_1$	I_0	GI (I=0 to 3)	PI (I=0 to 3)	C_{n+4}	$\overline{P}$/OVR		$\overline{G}$/N		Z ($\overline{OE}_Y$ = LOW)		
						Most Sig. Slice	Other Slices	Most Sig. Slice	Other Slices	Most Sig. Slice	Intermediate Slice	Least Sig. Slice
X	0	H	0	1	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	1	X	$\overline{R}_i \wedge S_i$	$\overline{R}_i \vee S_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	2	X	$R_i \wedge \overline{S}_i$	$R_i \vee \overline{S}_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	3	X	$R_i \wedge S_i$	$R_i \vee S_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	4	X	0	$\overline{S}_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	5	X	0	$\overline{S}_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	6	X	0	R_i	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	7	X	0	$\overline{R}_i$	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	8	X	0	1	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	9	X	$\overline{R}_i \wedge S_i$	1	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	A	X	$R_i \wedge S_i$	$R_i \vee S_i$	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	B	X	$\overline{R}_i \wedge S_i$	$\overline{R}_i \vee S_i$	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	C	X	$R_i \wedge S_i$	1	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	D	X	$\overline{R}_i \wedge \overline{S}_i$	1	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	E	X	$R_i \wedge S_i$	1	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
X	F	X	$\overline{R}_i \wedge \overline{S}_i$	1	0	0	0	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
0	0	L	0 if Z=L $R_i \wedge S_i$ if Z=H	S_i if Z=L $R_i \vee S_i$ if Z=H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	Input	Input	Q_0
2	0	L	0 if Z=L $R_i \wedge S_i$ if Z=H	S_i if Z=L $R_i \vee S_i$ if Z=H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	Input	Input	Q_0
4	0	L	See Note 1	See Note 2	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$	$\overline{Y}_0\overline{Y}_1\overline{Y}_2\overline{Y}_3$
5	0	L	0	S_i if Z=L $\overline{S}_i$ if Z=H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3 if Z=L $F_3 \forall S_3$ if Z=H	$\overline{G}$	S_3	Input	Input
6	0	L	0 if Z=L $\overline{R}_i \wedge S_i$ if Z=H	S_i if Z=L $\overline{R}_i \vee S_i$ if Z=H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	Input	Input	Q_0
8	0	L	0	S_i	See Note 3	$Q_2 \forall Q_1$	$\overline{P}$	Q_3	$\overline{G}$	$\overline{Q}_0\overline{Q}_1\overline{Q}_2\overline{Q}_3$	$\overline{Q}_0\overline{Q}_1\overline{Q}_2\overline{Q}_3$	$\overline{Q}_0\overline{Q}_1\overline{Q}_2\overline{Q}_3$
A	0	L	0	S_i	See Note 4	$F_2 \forall F_1$	$\overline{P}$	F_3	$\overline{G}$	See Note 5	See Note 5	See Note 5
C	0	L	$R_i \wedge S_i$ if Z=L $\overline{R}_i \wedge S_i$ if Z=H	$R_i \vee S_i$ if Z=L $\overline{R}_i \vee S_i$ if Z=H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	Sign Compare FF Output	Input	Input
E	0	L	$R_i \wedge S_i$ if Z=L $\overline{R}_i \wedge S_i$ if Z=H	$R_i \vee S_i$ if Z=L $\overline{R}_i \vee S_i$ if Z=H	$G \vee PC_n$	$C_{n+3} \forall C_{n+4}$	$\overline{P}$	F_3	$\overline{G}$	Sign Compare FF Output	Input	Input

L = LOW = 0
H = HIGH = 1
V = OR
$\wedge$ = AND
$\forall$ = EXCLUSIVE OR
$P = P_3P_2P_1P_0$
$G = G_3 \vee G_2P_3 \vee G_1P_2P_3 \vee G_0P_1P_2P_3$
$C_{n+3} = G_2 \vee G_1P_2 \vee G_0P_1P_2 \vee C_nP_pP_1P_2$

NOTES:
1. If $\overline{LSS}$ is LOW, $G_0 = S_0$ and $G_{1,2,3} = 0$
   If $\overline{LSS}$ is HIGH, $G_{0,1,2,3} = 0$
2. If $\overline{LSS}$ is LOW, $P_0 = 1$ and $P_{1,2,3} = S_{1,2,3}$
   If $\overline{LSS}$ is HIGH, $P_i = S_i$
3. At the most significant slice, $C_{n+4} = Q_3 \forall Q_2$
   At other slices, $C_{n+4} = G \vee PC_n$
4. At the most significant slice, $C_{n+4} = F_3 \forall F_2$
   At other slices, $C_{n+4} = G \vee PC_n$
5. $Z = \overline{Q}_0\overline{Q}_1\overline{Q}_2\overline{Q}_3\overline{F}_0\overline{F}_1\overline{F}_2\overline{F}_3$

Am2903 SPECIAL FUNCTIONS

The Am2903 provides nine Special Functions which facilitate the implementation of the following operations:

- Single- and Double-Length Normalization
- Two's Complement Division
- Unsigned and Two's Complement Multiplication
- Conversion Between Two's Complement and Sign/Magnitude Representation
- Incrementation by One or Two

Table 4 defines these Special Functions.

The Single-Length and Double-Length Normalization functions can be used to adjust a single-precision or double-precision floating point number in order to bring its mantissa within a specified range.

Three Special Functions which can be used to perform a two's complement, non-restoring divide operation are provided by the Am2903. These functions provide both single- and double-precision divide operations and can be performed in "n" clock cycles, where "n" is the number of bits in the quotient.

The Unsigned Multiply Special Function and the two Two's Complement Multiply Special Functions can be used to multiply two n-bit, unsigned or two's complement numbers, respectively, in n clock cycles. These functions utilize the conditional add and shift algorithm. During the last cycle of the two's complement multiplication, a conditional subtraction, rather than addition, is performed because the sign bit of the multiplier carries negative weight.

The Sign/Magnitude-Two's Complement Special Function can be used to convert number representation systems. A number expressed in Sign/Magnitude representation can be converted to the Two's Complement representation, and vice-versa, in one clock cycle.

The Increment by One or Two Special Function can be used to increment an unsigned or two's complement number by one or two. This is useful in 16-bit word, byte-addressable machines, where the word addresses are multiples of two.

Refer to Am2903 applications section for a more detailed description of these Special Functions.

PIN DEFINITIONS

A_{0-3} Four RAM address inputs which contain the address of the RAM word appearing at the RAM A output port.

B_{0-3} Four RAM address inputs which contain the address of the RAM word appearing at the RAM B output port and into which new data is written when the $\overline{WE}$ input and the CP input are LOW.

$\overline{WE}$ The RAM write enable input. If $\overline{WE}$ is LOW, data at the Y I/O port is written into the RAM when the CP input is LOW. When $\overline{WE}$ is HIGH, writing data into the RAM is inhibited.

DA_{0-3} A four-bit external data input which can be selected as one of the Am2903 ALU operand sources; DA_0 is the least significant bit. On the Am29203, the DA path is bidirectional, operating as either an ALU source operand or as an external output for the RAM A-port.

$\overline{EA}$ A control input which, when HIGH selects DA_{0-3} as the ALU R operand, and, when LOW, selects RAM output A as the ALU R operand and the DA_{0-3} output data.

DB_{0-3} A four-bit external data input/output. Under control of the $\overline{OE_B}$ input, RAM output port B can be directly read on these lines, or input data on these lines can be selected as the ALU S operand.

$\overline{OE_B}$ A control input which, when LOW, enables RAM output B onto the DB_{0-3} lines and, when HIGH, disables the RAM output B tri-state buffers.

C_n The carry-in input to the Am2903/29203 ALU.

I_{0-8} The nine instruction inputs used to select the Am2903/29203 operation to be performed.

$\overline{IEN}$ The instruction enable input which, when LOW, allows the Q Register and the Sign Compare flip-flop to be written. When $\overline{IEN}$ is HIGH, the Q Register and Sign Compare flip-flop are in the hold mode. On the Am2903, $\overline{IEN}$ also controls $\overline{WRITE}$. On the Am29203, $\overline{WRITE}$ is not affected by $\overline{IEN}$, but internally disables the RAM write enable.

C_{n+4} This output generally indicates the carry-out of the Am2903/29203 ALU. Refer to Table 5 for an exact definition of this pin.

$\overline{G}$/N A multi-purpose pin which indicates the carry generate, $\overline{G}$, function at the least significant and intermediate slices, and generally indicates the sign, N, of the ALU result at the most significant slice. Refer to Table 5 for an exact definition of this pin.

$\overline{P}$/OVR A multi-purpose pin which indicates the carry propagate, $\overline{P}$, function at the least significant and intermediate slices, and indicates the conventional two's complement overflow, OVR, signal at the most significant slice. Refer to Table 5 for an exact definition of this pin.

Z An open-collector input/output pin which, when HIGH, generally indicates the outputs are all LOW. For some Special Functions, Z is used as an input pin. Refer to Table 5 for an exact definition of this pin.

SIO_0, SIO_3 Bidirectional serial shift inputs/outputs for the ALU shifter. During a shift-up operation, SIO_0 is an input and SIO_3 an output. During a shift-down operation, SIO_3 is an input and SIO_0 is an output. Refer to Tables 3 and 4 for an exact definition of these pins.

QIO_0, QIO_3 Bidirectional serial shift inputs/outputs for the Q shifter which operate like SIO_0 and SIO_3. Refer to Tables 3 and 4 for an exact definition of these pins.

$\overline{LSS}$ An input pin which, when tied LOW, programs the chip to act as the least significant slice (LSS) of an Am2903/29203 array and enables the $\overline{WRITE}$ output onto the $\overline{WRITE}$/$\overline{MSS}$ pin. When $\overline{LSS}$ is tied HIGH, the chip is programmed to operate as either an intermediate or most significant slice and the $\overline{WRITE}$ output buffer is disabled.

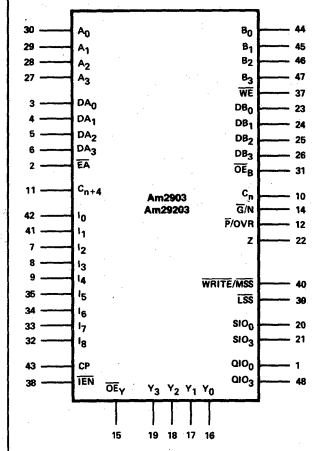
$\overline{WRITE}$/ $\overline{MSS}$ When $\overline{LSS}$ is tied LOW, the $\overline{WRITE}$ output signal appears at this pin; the $\overline{WRITE}$ signal is LOW when an instruction which writes data into the RAM is being executed. When $\overline{LSS}$ is tied HIGH, $\overline{WRITE}$/$\overline{MSS}$ is an input pin; tying it HIGH programs the chip to operate as an intermediate slice (IS) and tying it LOW programs the chip to operate as the most significant slice (MSS).

Y_{0-3} Four data inputs/outputs of the Am2903/29203. Under control of the $\overline{OE_Y}$ input, the ALU shifter output data can be enabled onto these lines, or these lines can be used as data inputs when external data is written directly into the RAM.

$\overline{OE_Y}$ A control input which, when LOW, enables the ALU shifter output data onto the Y_{0-3} lines and, when HIGH, disables the Y_{0-3} three-state output buffers.

CP The clock input to the Am2903/29203. The Q Register and Sign Compare flip-flop are clocked on the LOW-to-HIGH transition of the CP signal. When enabled by $\overline{WE}$, data is written in the RAM when CP is LOW.

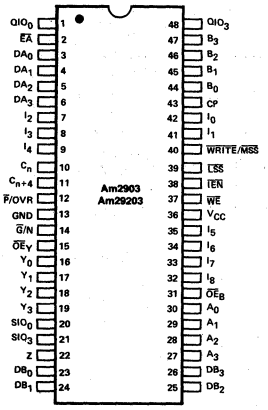
LOGIC SYMBOL



V_{CC} = Pin 36
GND = Pin 13

MPR-033

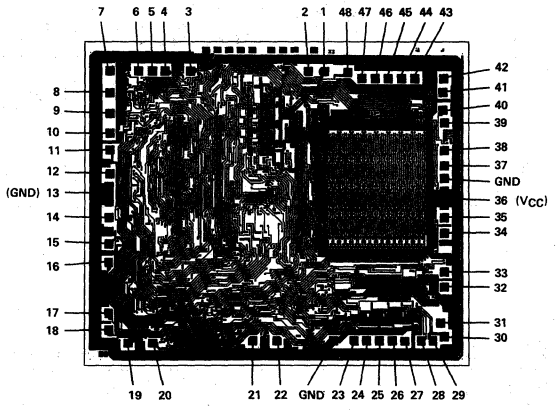
CONNECTION DIAGRAM
Top View



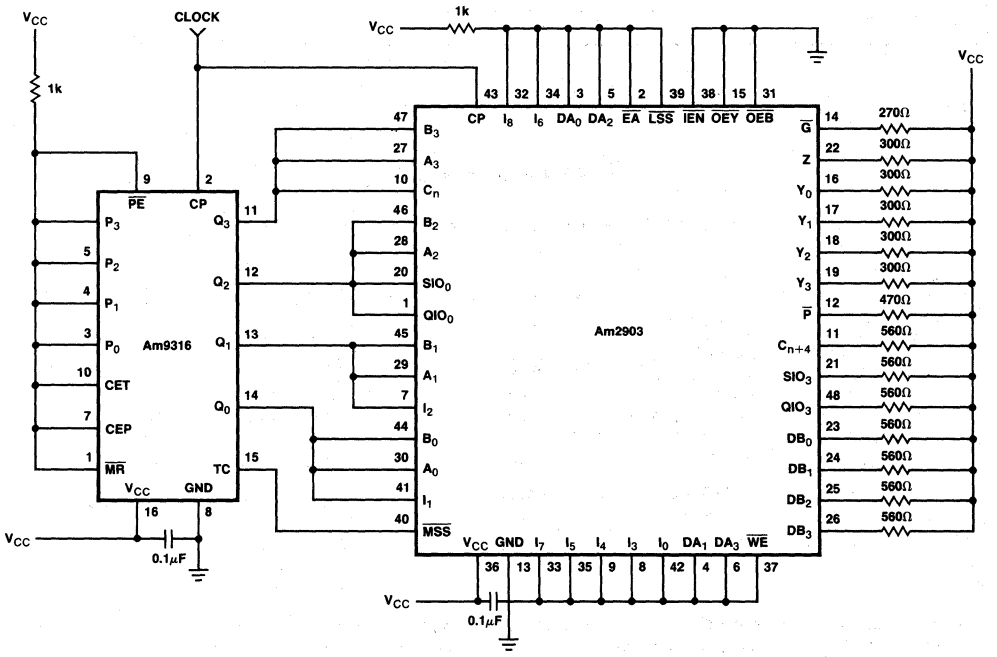
Note: Pin 1 is marked for orientation.

MPR-034

METALLIZATION AND PAD LAYOUT



DIE SIZE 0.163" X 0.197"
Note: Pin numbers correspond to DIP package.



MPR-582

V_{CC} = 5.0V
Frequency = 100kHz
T_A = +125°C
All registers are 1/4 watt ±5%
This circuit conforms to MIL-STD-883, Methods 1005 and 1015, Condition D.
One Am9316 Can Drive Maximum of Five Am2903s.

Am2903 Burn-in and Life Test Circuit

Am2903 • Am29203
OPERATING RANGES (over which DC, switching, and functional specifications apply)

Range	Part Number Suffix	Temperature	V_{CC}	V_{IH}	V_{IL}
COM'L	PC, PCB, DC, DCB, XC	T_A = 0 to 70°C	4.75 to 5.25V	2.0V	0.8V
MIL	DM, DMB, FM, FMB, XM	T_C = −55 to +125°C	4.50 to 5.50V	2.0V	0.3V

3

ABSOLUTE MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Temperature (Ambient) Under Bias	−55 to +125°C
Supply Voltage to Ground Potential Continuous	−0.5 to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5 to +V_{CC} max.
DC Input Voltage	−0.5 to +5.5V
DC Output Current, Into Outputs	30mA
DC Input Current	−30 to +5.0mA

ORDERING INFORMATION

Order the part number according to the table below to obtain the desired package, temperature range, and screening level.

Am29203 Order Number	Am2903 Order Number	Package Type (Note 1)	Operating Range (Note 2)	Screening Level (Note 3)
AM29203DC	AM2903DC	D-48	C	C-1
AM29203DC-B	AM2903DC-B	D-48	C	B-2 (Note 4)
AM29203DM	AM2903DM	D-48	M	C-3
AM29203DMB	AM2903DM-B	D-48	M	B-3
AM29203FM	AM2903FM	F-48	M	C-3
AM29203FMB	AM2903FM-B	F-48	M	B-3
Am29203XC	Am2903XC	Dice	C	Visual inspection to MIL-STD-883 Method 2010B.
Am29203XM	Am2903XM	Dice	M	

Notes: 1. P = Molded DIP, D = Hermetic DIP, F = Flat Pak. Number following letter is number of leads. See Appendix B for detailed outline. Where Appendix B contains several dash numbers, any of the variations of the package may be used unless otherwise specified.
2. C = 0°C to +70°C, V_{CC} = 4.75V to 5.25V, M = − 55°C to +125°C, V_{CC} = 4.50V to 5.50V.
3. See Appendix A for details of screening. Levels C-1 and C-3 conform to MIL-STD-883, Class C. Level B-3 conforms to MIL-STD-883, Class B.
4. 96 hour burn-in.

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)			Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN. V_{IN} = V_{IH} or V_{IL}	I_{OH} = −1.6mA Y_0-Y_3, $\overline{G}$/N		2.4			Volts
			I_{OH} = −800μA DB_{0-3}, $\overline{P}$/OVR SIO_0, SIO_3, QIO_0, QIO_3, $\overline{WRITE}$, C_{n+4}		2.4			
I_{CEX}	Output Leakage Current for Z Output (Note 4)	V_{CC} = MIN., V_{OH} = 5.5V V_{IN} = V_{IH} or V_{IL}					250	μA
V_{OL}	Output LOW Voltage	V_{CC} = MIN. V_{IN} = V_{IH} = or V_{IL}	Y_0, Y_1, Y_2 Y_3, Z	I_{OL} = 20mA (COM'L)			0.5	Volts
				I_{OL} = 16mA (MIL)				
			DB_0, DB_1, DB_2, DB_3	I_{OL} = 12mA (COM'L)			0.5	
				I_{OL} = 8.0mA (MIL)				
			$\overline{G}$/N	I_{OL} = 18mA			0.5	
			$\overline{P}$/OVR	I_{OL} = 10mA			0.5	
			C_{n+4}, SIO_0 SIO_3, QIO_0 QIO_3, $\overline{WRITE}$	I_{OL} = 8.0mA			0.5	
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs (Note 6)			2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs (Note 6)					0.8	Volts
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA					−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.5V (Note 4)	C_n				−3.6	mA
			Y_0, Y_1, Y_2, Y_3				−1.13	
			I_0, I_1, I_2, I_3, I_4 DA_0, DA_1, DA_2, DA_3				−0.72	
			SIO_0, SIO_3, QIO_0, QIO_3, $\overline{MSS}$, DB_0, DB_1, DB_2, DB_3				−0.77	
			All other inputs				−0.36	
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V (Note 4)	C_n				200	μA
			Y_0, Y_1, Y_2, Y_3				110	
			I_0-I_4, DA_0-DA_3				40	
			SIO_0, SIO_3, QIO_0, QIO_3, DB_{0-3}, $\overline{MSS}$				90	
			All other inputs				20	
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V					1.0	mA
I_{OZH} I_{OZL}	Off State (HIGH Impedance) Output Current	V_{CC} = MAX., (Note 4)	Y_0-Y_3	V_O = 2.4V			110	μA
				V_O = 0.5V			−1130	
			DB_{0-3}, QIO_0, QIO_0, SIO_0, SIO_3, $\overline{MSS}$/IS	V_O = 2.4V			90	
				V_O = 0.5V			−770	
I_{OS}	Output Short Circuit Current (Note 3)	V_{CC} = MAX + 0.5V V_O = 0.5V			−30		−85	mA
I_{CC}	Power Supply Current (Note 5)	V_{CC} = MAX.	T_A = 25°C			220	335	mA
			COM'L	T_A = 0 to 70°C			350	
				T_A = 70°C			291	
			MIL	T_C = −55 to 125°C			395	
				T_C = 125°C			258	

Notes:
1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. Y_{0-3}, DB_{0-3}, $SIO_{0,3}$, $QIO_{0,3}$ and WRITE/MSS are three state outputs internally connected to TTL inputs. Z is an open-collector output internally connected to a TTL input. Input characteristics are measured under conditions such that the outputs are in the OFF state.
5. Worst case I_{CC} is at minimum temperature.
6. These input levels provide zero noise immunity and should only be tested in a static, noise-free environment.

Am2903 SWITCHING CHARACTERISTICS

Am2903 switching characteristics are dependent on temperature, voltage, and the operating mode of the device. The detailed data for the part is given in the 24 tables comprising Appendix A of this data sheet. For reference, one set of these tables is reproduced on this page and the next. For switching data for special functions and military devices, refer to Appendix A.

TABLE III A
Guaranteed Combinational Delays
T_A = 0°C to +70°C, V_{CC} = 4.75V to 5.25V
Standard Functions

From Input \ To Output	Y	C_{n+4}	$\overline{G}$, $\overline{P}$	Z (s)	N	OVR	DB	$\overline{WRITE}$	QIO_0 QIO_3	SIO_0	SIO_3	SIO_0 Parity
A Address (Arith. Mode)	86	81	69	110	86	108	–	–	–	84	94	115
B Address	99	88	81	123	99	112	49	–	–	94	104	140
A Address (Logic Mode)	87	–	68	111	89	–	–	–	–	79	94	115
B Address	84	–	73	108	84	–	49	–	–	84	90	120
DA Inputs (Arith. Mode)	63	60	49	87	64	89	–	–	–	60	70	101
DB Inputs	61	59	47	85	62	84	–	–	–	62	68	98
DA Inputs (Logic Mode)	64	–	48	88	66	–	–	–	–	61	72	101
DB Inputs	55	–	32	79	57	–	–	–	–	52	61	93
$\overline{EA}$	59	53	42	83	59	83	–	–	–	57	64	98
C_n	40	30	–	64	40	58	–	–	–	38	46	67
I_0	52	48	36	76	52	63	–	49	*	50*	58*	93*
I_{4321}	71	65	72	95	69	84	–	49	*	66*	73*	105*
I_{8765}	42	–	–	66	–	–	–	50	60*	42*	45*	42*
$\overline{IEN}$	–	–	–	–	–	–	–	22	–	–	–	–
SIO_3, SIO_0	26	–	–	50	–	–	–	–	–	–	29	36
Clock	87	87	71	111	88	108	37	–	40	84	92	105
Y	–	–	–	24	–	–	–	–	–	–	–	–
$\overline{MSS}$	44	–	44	68	44	44	–	–	–	44	46	44

Note: A "*" means the output is enabled or disabled by the input. See Tables C for enable and disable times. A number shown with a * is the delay to correct data on an enabled output. A * shown without a number means the output is disabled by the input or it is enabled but the delay to correct data is determined by something else.

TABLE III B
Guaranteed Set-up and Hold Times
T_A = 0°C to +70°C, V_{CC} = 4.75V to 5.25V
All Functions

CAUTION: READ NOTES TO TABLE B. NA = Not Applicable; no timing constraint.

To Output / From Input	With Respect to this Signal	HIGH-to-LOW Set-up	HIGH-to-LOW Hold	LOW-to-HIGH Set-up	LOW-to-HIGH Hold	Comment
Y	Clock	NA	NA	20	3	To store Y in RAM or Q
$\overline{WE}$ HIGH	Clock	25	Note 2	Note 2	0	To Prevent Writing
$\overline{WE}$ LOW	Clock	NA	NA	30	0	To Write into RAM
A, B as Sources	Clock	27	3	NA	NA	See Note 3
B as a Destination	Clock and WE both LOW	6	Note 4	Note 4	3	To Write Data only into the Correct B Address
QIO_0, QIO_3	Clock	NA	NA	21	3	To Shift Q
I_{8765}	Clock	24	Note 5	Note 5	0	
$\overline{IEN}$ HIGH	Clock	30	Note 2	Note 2	0	To Prevent Writing into Q
$\overline{IEN}$ LOW	Clock	NA	NA	30	0	To Write into Q
I_{43210}	Clock	24	–	68	0	See Note 6

Notes:
1. For set-up times from all inputs not specified in Table B, the set-up time is computed by calculating the delay to stable Y outputs and then allowing the Y set-up time. Even if the RAM is not being loaded, the Y set-up time is necessary to set-up the Q register. All unspecified hold times are less than or equal to zero relative to the clock LOW-to-HIGH edge.
2. $\overline{WE}$ controls writing into the RAM. $\overline{IEN}$ controls writing into Q and, indirectly, controls $\overline{WE}$ through the write output. To prevent writing, $\overline{IEN}$ and $\overline{WE}$ must be HIGH during the entire clock LOW time. They may go LOW after the clock has gone LOW to cause a write provided the $\overline{WE}$ LOW and $\overline{IEN}$ LOW set-up times are met. Having gone LOW, they should not be returned HIGH until after the clock has gone HIGH.
3. A and B addresses must be set-up prior to clock LOW transition to capture data in latches at RAM output.
4. Writing occurs when CP and $\overline{WE}$ are both LOW. The B address should be stable during this entire period.
5. Because I_{8765} control the writing or not writing of data into RAM and Q, they should be stable during the entire clock LOW time unless $\overline{IEN}$ is HIGH, preventing writing.
6. The set-up time prior to the clock LOW-to-HIGH transition occurs in parallel with the set-up time prior to the clock HIGH-to-LOW transition and the clock LOW time. The actual set-up time requirement on I_{43210}, relative to the clock LOW-to-HIGH transition, is the longer of (1) the set-up time prior to clock L → H, and (2) the sum of the set-up time prior to clock H → L and the clock LOW time.

TABLE III C
Guaranteed Enable/Disable Times
T_A = 0°C to +70°C, V_{CC} = 4.75V to 5.25V
All Functions

From	To	Enable	Disable	
OEY	Y_i	27	25	ns
OEB	DB_i	31	25	ns
I_8	SIO_0, SIO_3		25	ns
I_{8765}	QIO_0, QIO_3		60	ns
I_{43210}	QIO_0, QIO_3	65	60	ns
$\overline{LSS}$	$\overline{WRITE}$	31	25	ns

Note:
1. C_L = 5.0pF for output disable tests. Measurement is made to a 0.5V change on the output.

TABLE III D
Guaranteed Clock and Write Pulse Characteristics
T_A = 0°C to +70°C, V_{CC} = 4.75V to 5.25V
All Functions

Minimum Clock LOW Time	30	ns
Minimum Clock HIGH Time	30	ns
Minimum Time CP and WE both LOW to Write	30	ns

CYCLE TIMES FOR 16-BIT SYSTEM FOR COMMON OPERATIONS

The illustration below shows a typical configuration using 4 Am2903 Superslices, an Am2902A carry lookahead chip, and the Am2904 for shift multiplexers, status registers, and carry-in control. For the system enclosed within the dashed lines, there are four major switching paths whose values for various kinds of cycles are summarized below, and shown on the timing waveform.

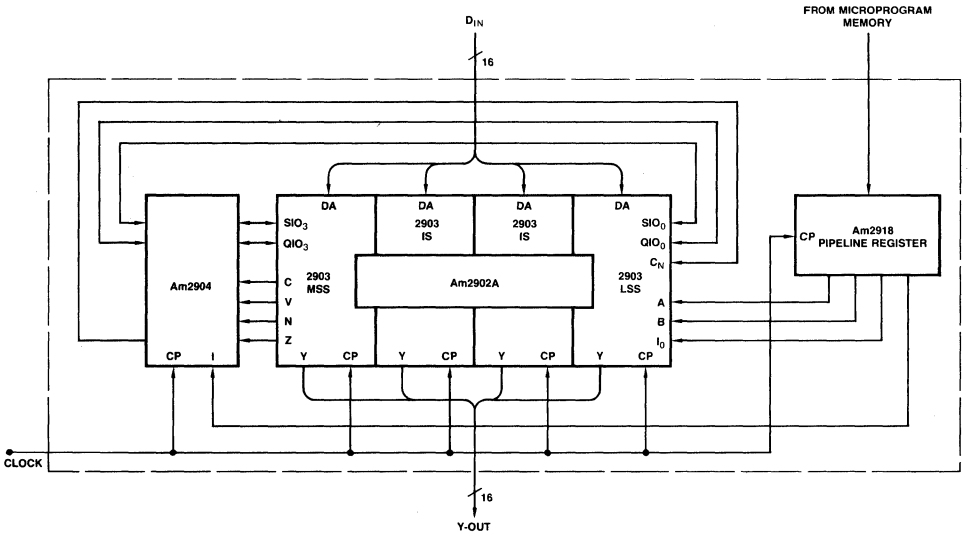
1. MICROCYCLE TIME (TCHCH).
   The minimum time which must elapse between a LOW-TO-HIGH clock transition and the next LOW-TO-HIGH clock transition.
2. DATA SET-UP TIME (TDVCH).
   The minimum time which must be allowed between valid, stable data on the D inputs and the clock LOW-TO-HIGH transition.
3. D TO Y (TDVYV).
   The maximum time required to obtain valid Y output data after the D inputs are valid. This is the combinational delay through the parts from D to Y.
4. CP TO Y (TCHYV).
   The maximum time required to obtain valid Y outputs after a clock LOW-TO-HIGH transition.

The types of cycles for which data is summarized are as follows:

1. Logic – Any logical operation without a shift.
2. Logic Rotate – Any logic operation with a rotate or shift.
3. Arithmetic – An add or subtract with no shift.
4. Multiply – The first cycle of a 2's complement multiply instruction. Subsequent cycles require less time.
5. Divide – The iterative divide cycle. The first divide instruction and the last divide (correction) instruction require less time.

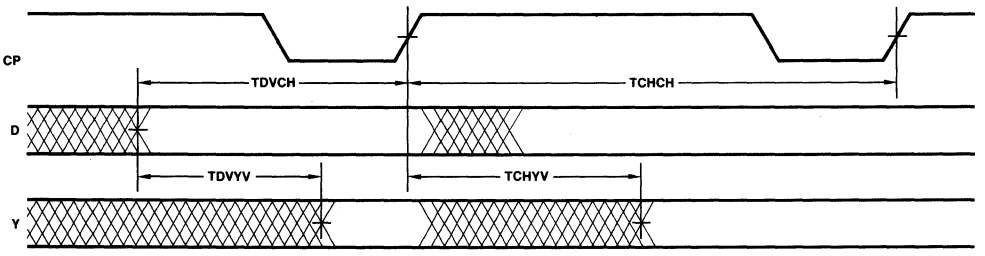
Time in ns Over Commercial Operating Range

CYCLE	TCHCH	TDVCH	TDVYV	TCHYV
LOGIC	143	105	64	102
LOGIC ROTATE	180	143	123	160
ARITHMETIC	184	137	96	143
MULTIPLY	200	140	120	180
DIVIDE	228	167	128	189



16-Bit System with Am2903, Am2902A, Am2904

MPR-583



Timing Waveforms for Data In, Clock, and Y Out

MPR-584

USING THE Am2903 AND Am29203
Except Where Otherwise Noted, All References to the Am2903 Also Apply to the Am29203.

Am2903 APPLICATIONS

The Am2903 is designed to be used in microprogrammed systems. Figure 1 illustrates a recommended architecture. The control and data inputs to the Am2903 normally will all come from registers clocked at the same time as the Am2903. The register inputs come from a ROM or PROM – the "microprogram store". This memory contains sequences of microinstructions which apply the proper control signals to the Am2903's and other circuits to execute the desired operation.

The address lines of the microprogram store are driven from the Am2910 Microprogram Sequencer. This device has facilities for storing an address, incrementing an address, jumping to any address, and linking subroutines. The Am2910 is controlled by some of the bits coming from the microprogram store. Essentially, these bits are the "next instruction" control.

One Level Pipeline Based System

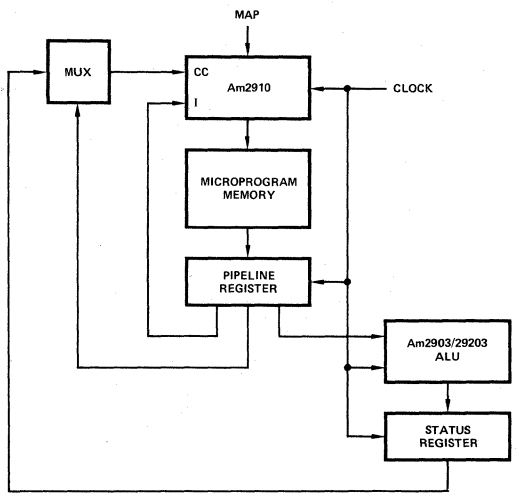


Figure 1. Typical Microprogram Architecture.

MPR-035

Note that with the microprogram register in between the microprogram memory store and the Am2903's, a microinstruction accessed on one cycle is executed on the next cycle. As one microinstruction is executed, the next microinstruction is being read from microprogram memory. In this configuration, system speed is improved because the execution time in the Am2903's occurs in parallel with the access time of the microprogram store. Without the "pipeline register", these two functions must occur serially.

Expansion of the Am2903

The Am2903 is a four-bit CPU slice. Any number of Am2903's can be interconnected to form CPU's of 8, 16, 32, or more bits, in four-bit increments. Figure 2 illustrates the interconnection of four Am2903's to form a 16-bit CPU, using ripple carry.

With the exception of the carry interconnection, all expansion schemes are the same. The QIO_3 and SIO_3 pins are bidirectional left/right shift lines at the MSB of the device. For all devices except the most significant, these lines are connected to the QIO_0 and SIO_0 pins of the adjacent more significant device. These connections allow the Q Registers of all Am2903's to be shifted left or right as a contiguous n-bit register, and also allow the ALU output data to be shifted left or right as a contiguous n-bit word prior to storage in the RAM. At the LSB and MSB of the CPU, the shift pins should be connected to a shift multiplexer which can be controlled by the microcode to select the appropriate input signals to the shift inputs.

Device 1 has been defined as the least significant slice (LSS) and its $\overline{LSS}$ pin has accordingly been grounded. The Write/Most Significant Slice ($\overline{WRITE}/\overline{MSS}$) pin of device 1 is now defined as being the Write output, which may now be used to drive the write enable ($\overline{WE}$) signal common to the four devices. Devices 2 and 3 are designated as intermediate slices and hence the $\overline{LSS}$ and $\overline{WRITE}/\overline{MSS}$ pins are tied HIGH. Caution: $\overline{W}/\overline{MSS}$ must be tied to V_{CC} through a resistor; $\overline{W}/\overline{MSS}$ and $\overline{LSS}$ may not be shorted directly together. Device 4 is designated the most significant slice ($\overline{MSS}$) with the $\overline{LSS}$ pin tied HIGH and the $\overline{WRITE}/\overline{MSS}$ pin held LOW. The open collector, bidirectional Z pins are tied together for detecting zero or for inter-chip communication for some special instruction. The Carry-Out (C_{n+4}) is connected to the Carry-In (C_n) of the next chip in the case of ripple carry. For a faster carry scheme, an Am2902 may be employed (as shown in Figure 3) such that the $\overline{G}$ and $\overline{P}$ outputs of the Am2903 are connected to the appropriate $\overline{G}$ and $\overline{P}$ inputs of the Am2902, while the C_{n+x}, C_{n+y}, and C_{n+z} outputs of the Am2902 are connected to the C_n input of the appropriate Am2903. Note that $\overline{G}$/N and $\overline{P}$/OVR pin functions are device dependent. The most significant slice outputs N and OVR while all other slices output $\overline{G}$ and $\overline{P}$

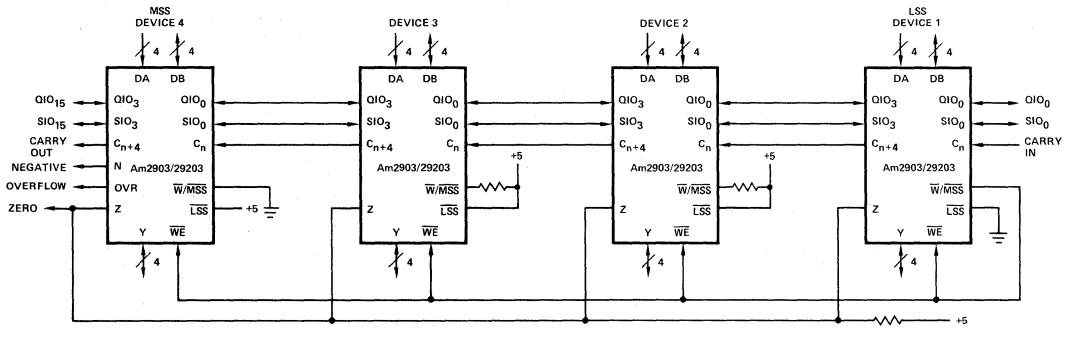


Figure 2. 16-Bit CPU with Ripple Carry.

MPR-036

The $\overline{IEN}$ pin of the Am2903 allows the option of conditional instruction execution. If $\overline{IEN}$ is LOW, all internal clocking is enabled, allowing the latches, RAM, and Q Register to function. If $\overline{IEN}$ is HIGH, the RAM and Q Register are disabled. The RAM is controlled by $\overline{IEN}$ if $\overline{WE}$ is connected to the $\overline{WRITE}$ output.

It would be appropriate at this point to mention that the Am2903 may be microcoded to work in either two-or three-address architecture modes. The two-address modes allow A+B→B while the three-address mode makes possibleA+B→C. Implementation of a three-address architecture is made possible by varying the timing of $\overline{IEN}$ in relationship to the external clock and changing the B address as shown in Figure 4. This technique is discussed in more detail under Memory Expansion.

Parity

The Am2903 computes parity on a chosen word when the instruction bits I_{5-8} have the values of 4_{16} to 7_{16} as shown in Table 3. The computed parity is the result of the exclusive OR of the individual ALU outputs and SIO_3. Parity output is found on SIO_0. Parity between devices may be cascaded by the interconnection of the SIO_0 and SIO_3 ports of the devices as shown in Figure 3. The equation for the parity output at SIO_0 port of device 1 is given by $SIO_0 = F_{15} \forall F_{14} \forall F_{13} \forall \ldots \forall F_1 \forall F_0 \forall SIO_{15}$.

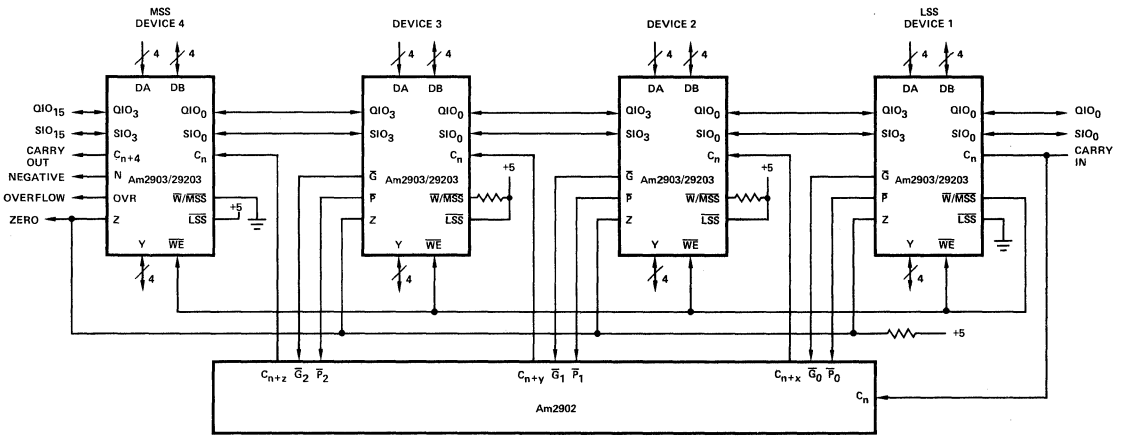


Figure 3. 16-Bit CPU with Carry Look Ahead.

MPR-037

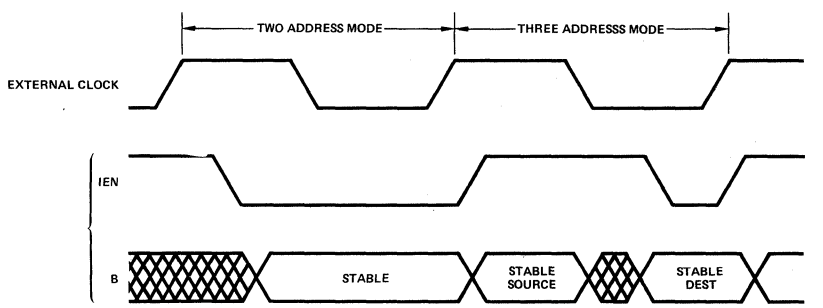


Figure 4. Relationship of $\overline{IEN}$ and Clock During Two Address and Three Address Modes.

MPR-038

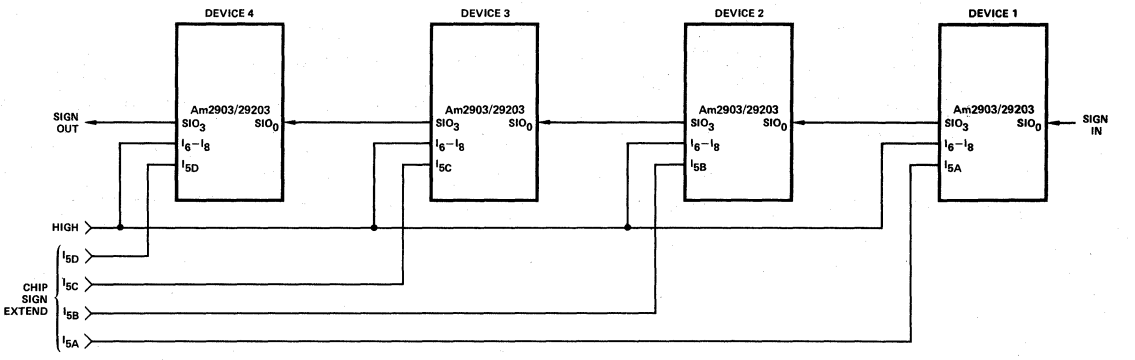


Figure 5. Sign Extend. MPR-039

Sign Extend

Sign extension across any number of Am2903 devices can be done in one microcycle. Referring again to the table of instructions (Table 3), the sign extend instruction (Hex instruction E) on I_{5-8} causes the sign present at the SIO_0 port of a device to be extended across the device and appear at the SIO_3 port and at the Y outputs. If the least significant bit of the instruction (bit I_5) is HIGH, Hex instruction F is present on I_{5-8}, commanding a shifter pass instruction. At this time, F_3 of the ALU is present on the SIO_3 output pin. It is then possible to control the extension of the sign across chip boundaries by controlling the state of I_5 when I_{6-8} are HIGH. Figure 5 outlines the Am2903 in sign extend mode. With I_{6-8} held HIGH, the individual chip sign extend is controlled by I_{5A-D}. If, for example, I_{5A} and I_{5B} are HIGH while I_{5C} and I_{5D} are LOW, the signal present at the boundaries of devices 2 and 3 (F_3 of device 2) will be extended across devices 3 and 4 at the SIO_3 pin of device 4. The output of the four devices will be available at their respective Y data ports. The next positive edge of the clock will load the Y outputs into the address selected by the B port. Hence, the results of the sign extension is stored in the RAM.

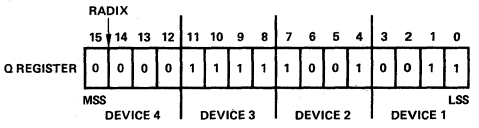
SPECIAL FUNCTIONS

When $I_{0-4} = 0$, the Am2903 is in the Special Function mode. In this mode, both the source and destination are controlled by I_{5-8}. The Special Functions are in essence special microinstructions that are used to reduce the number of microcycles needed to execute certain functions in the Am2903.

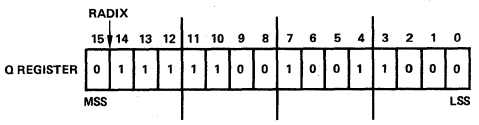
Normalization, Single- and Double-Length

Normalization is used as a means of referencing a number to a fixed radix point. Normalization strips out all leading sign bits such that the two bits immediately adjacent to the radix point are of opposite polarity.

Normalization is commonly used in such operations as fixed-to-floating point conversion and division. The Am2903 provides for normalization by using the Single-Length and Double-Length Normalize commands. Figure 6a represents the Q Register of a 16-bit processor which contains a positive number. When the Single-Length Normalize command is applied, each positive edge of the clock will cause the bits to shift toward the most significant bit (bit 15) of the Q Register. Zeros are shifted in via the QIO_0 port. When the bits on either side of the radix point (bits 14 and 15) are of opposite value, the number is considered to be normalized as shown in Figure 6b. The event of normalization is externally indicated by a HIGH level on the C_{n+4} pin of the most significant slice (C_{n+4} MSS = Q_3 MSS $\forall$ Q_2 MSS).



a) Unnormalized Positive Number.



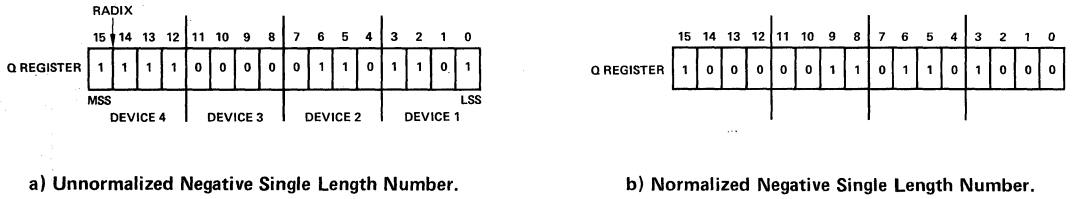
b) Normalized Positive Number.

Figure 6. MPR-040

There are also provisions made for a normalization indication via the OVR pin one microcycle before the same indication is available on the C_{n+4} pin (OVR = Q_2 MSS $\forall$ Q_1 MSS). This is for use in applications that require a stage of register buffering of the normalization indication.

Since a number comprised of all zeros is not considered for normalization, the Am2903 indicates when such a condition arises. If the Q Register is zero and the Single-Length Normalization command is given, a HIGH level will be present on the Z line. The sign output, N, indicates the sign of the number stored in the Q register, Q_3 MSS. An unnormalized negative number (Figure 7a) is normalized in the same manner as a positive number. The results of single-length normalization are shown in Figure 7b. The device interconnection for single-length normalization is outlined in Figure 8. During single length normalization, the number of shifts performed to achieve normalization can be counted and stored in one of the working registers. This can be achieved by forcing a HIGH at the C_n input of the least significant slice, since during this special function the ALU performs the function [B] + C_n and the result is stored in B.

Normalizing a double-length word can be done with the Double-Length Normalize command which assumes that a user-selected RAM Register contains the most significant portion of the word to be normalized while the Q Register holds the least significant half (Figure 9). The device interconnection for double-length normalization is shown in Figure 10. The C_{n+4}, OVR, N, and Z outputs of the most significant slice perform the same functions in double-length normalization as they did in single-length normalization except that C_{n+4}, OVR, and N are derived from the output of the ALU of the most significant slice in the case of double-length normalization, instead of the Q Register of the most significant



a) Unnormalized Negative Single Length Number.

b) Normalized Negative Single Length Number.

Figure 7.

MPR-041

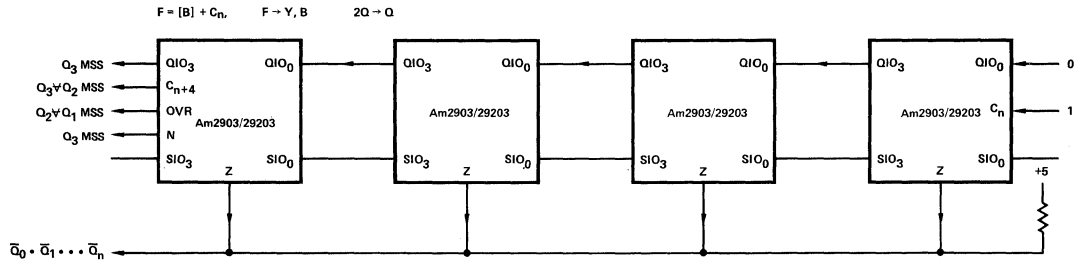


Figure 8. Single Length Normalize.

MPR-042

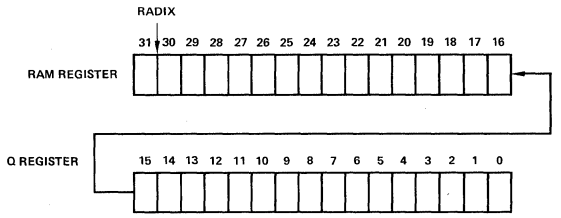


Figure 9. Double Length Word.

MPR-043

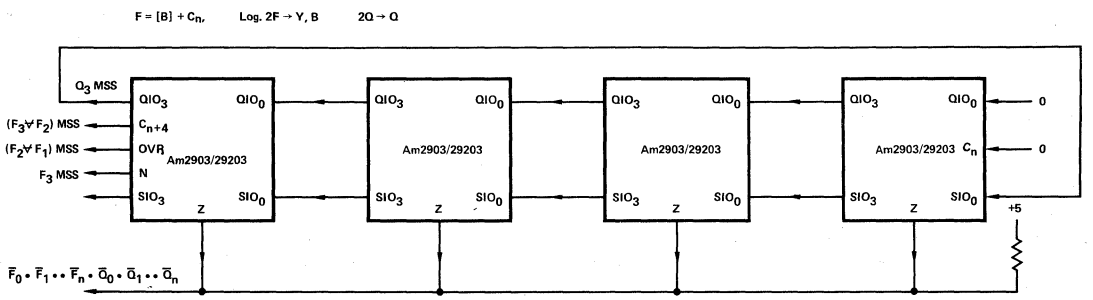


Figure 10. Double Length Normalize.

MPR-044

slice as in single-length normalization. A high-level Z line in double-length normalization reveals that the outputs of the ALU and Q Register are both zero, hence indicating that the double-length word is zero.

When double-length normalization is being performed, shift counting is done either with an extra microcycle or with an external counter.

Sign Magnitude, Two's Complement Conversion

As part of the special instruction set, the Am2903 can convert between two's complement and sign/magnitude representations. Figure 11 illustrates the interconnection needed for sign magnitude/two's complement conversion. The word to be converted is applied to the S input port of the ALU (from the RAM B port or the DB I/O port). The C_n input of device 1 is connected to the Z pin. The sign bit (S_3MSS) is brought out on the Z line and informs the other ALU's if the conversion is being performed on a negative or positive number. If the number to be converted is the most negative number in two's complement [i.e., 100 . . . 00 (-2^n)], an overflow indication will occur. This is because -2^n is one greater than any number that can be represented in sign magnitude notation and hence an attempted conversion to sign magnitude from -2^n will cause an overflow. When minus zero in sign magnitude notation (100 . . . 0) is converted to two's complement notation, the correct result is obtained (0 . . .0).

Increment by One or Two

Incrementation by One or Two is made possible by the Special Function of the same name. This command is quite useful in the case of byte addressable words. Referencing Figure 12, a word may be incremented by one if C_n is LOW or incremented by two if C_n is HIGH.

Unsigned Multiply

This Special Function allows for easy implementation of unsigned multiplication. Figure 13 is the unsigned multiply flow chart. The algorithm requires that initially the RAM word addressed by Address port B be zero, that the multiplier be in the Q Register, and that the multiplicand be in the register addressed by Address port A. The initial conditions for the execution of the algorithm are that: 1) register R_0 be reset to zero; 2) the multiplicand be in R_1; and 3) the multiplier be in R_2. The first operation transfers the multiplier, R_2, to the Q Register. The Unsigned Multiply instruction is then executed 16 times. During the Unsigned Multiply instruction, R_0 is addressed by RAM address port B and the multiplicand is addressed by RAM address port A.

When the unsigned Multiply command is given, the Z pin of device 1 becomes an output while the Z pins of the remaining devices are specified as inputs as shown in Figure 15. The Z output of device 1 is the same state as the least significant bit of the multiplier in the Q Register. The Z output of device 1 informs the ALU's of all the slices, via their Z pins, to add the partial product (referenced by the B address port) to the mul-

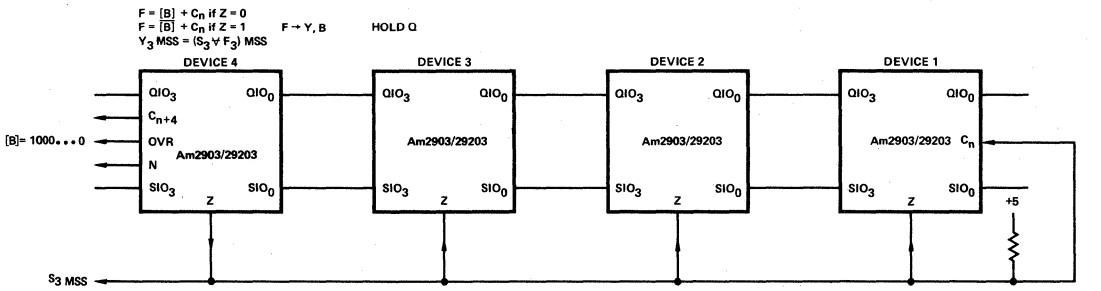


Figure 11. 2's Complement←→Sign/Magnitude.

MPR-045

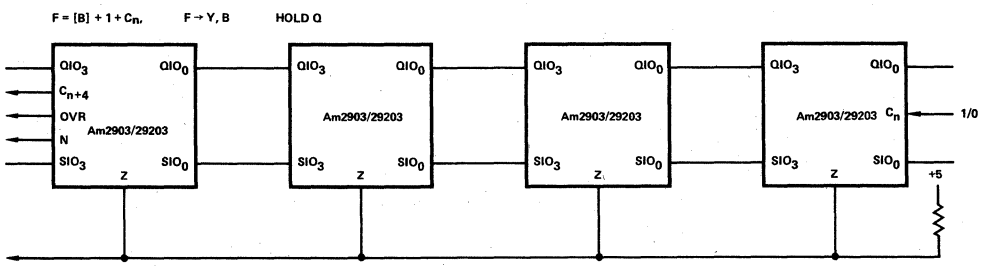


Figure 12. Increment by 2/1.

MPR-046

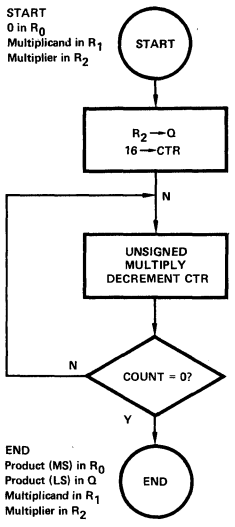


Figure 13. Unsigned 16 X 16 Multiply Flowchart.

tiplicand (referenced by the A address port) if Z = 1. If Z = 0, the output of the ALU is simply the partial product (referenced by the B address port). Since C_n is held LOW, it is not a factor in the computation. Each positive-going edge of the clock will internally shift the ALU outputs toward the least significant bit and simultaneously store the shifted results in the register selected by the B address port, thus becoming the new partial sum. During the down shifting process, the C_{n+4} generated in device 4 is internally shifted into the Y_3 position of device 4. At this time, one bit of the multiplier will down shift out of the QIO_0 ports of each device into the QIO_3 port of the next less significant slice. The partial product is shifted down between chips in a like manner, between the SIO_0 and SIO_3 ports, with SIO_0 of device 1 being connected to QIO_3 of device 4 for purposes of constructing a 32-bit long register to hold the 32-bit product. At the finish of the 16 x 16 multiply, the most significant 16 bits of the product will be found in the register referenced by the B address lines while the least significant 16 bits are stored in the Q Register. Using a typical Computer Control Unit (CCU), as shown in Figure 16, the unsigned multiply operation requires only two lines of microcode, as shown in Figure 17, and is executed in 17 microcycles.

Two's Complement Multiplication

The algorithm for two's complement multiplication is illustrated by Figure 14. The initial conditions for two's complement multiplication are the same as for the unsigned multiply operation. The Two's Complement Multiply Command is applied for 15 clock cycles in the case of a 16 x 16 multiply. During the down shifting process the term N∀OVR generated in device 4 is internally shifted into the Y_3 position of device 4. The data flow shown in Figure 15 is still valid. After 15 cycles, the sign bit of the multiplier is present at the Z output of device 1. At this time, the user must place the Two's Complement Multiply Last cycle command on the instruction lines. The interconnection for this instruction is shown in Figure 18. On the next positive edge of the clock, the Am2903 will adjust the partial product, if the sign of the multiplier is negative, by subtracting out the two's complement representation of the multiplicand. If the sign bit is positive, the partial product is not adjusted. At this point, two's complement multiplication is completed. Using a typical CCU, as shown in Figure 16, the two's complement multiply operation requires only three lines of microcode, as shown in Figure 19, and is executed in 17 microcycles.

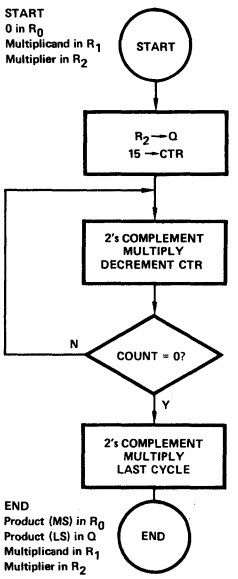
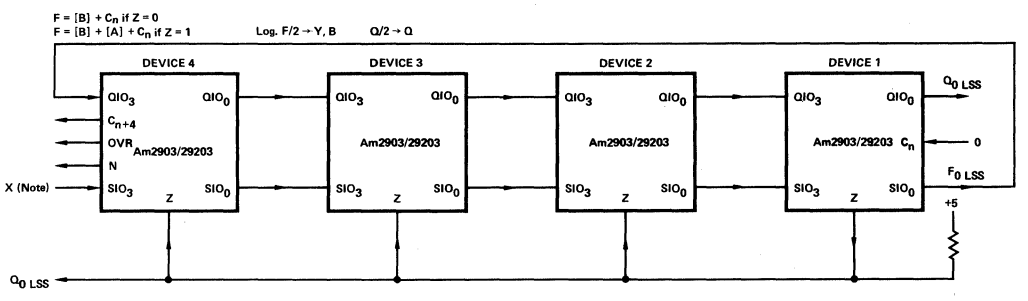


Figure 14. 2's Complement 16 X 16 Multiply.



Note: For unsigned multiply, C_{n+4} MSS is internally shifted into position Y_3 MSS; 2's complement multiply N∀OVR is internally shifted into position Y_3 MSS.

Figure 15. Multiply.

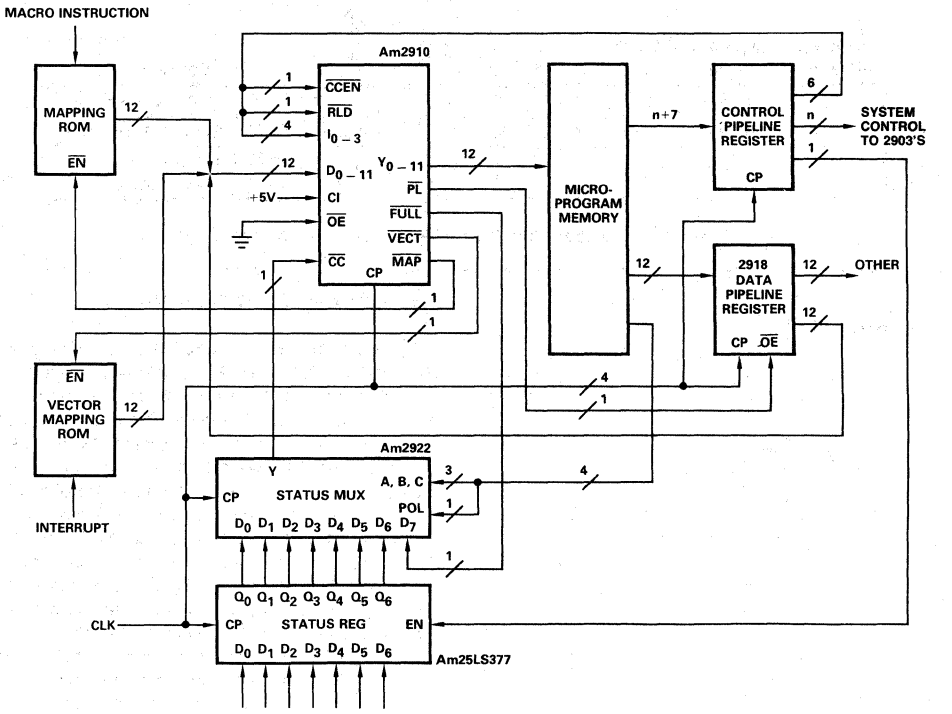
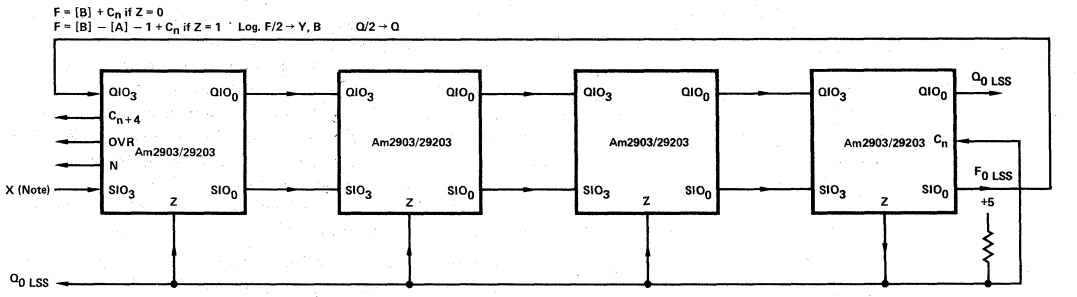


Figure 16. Typical Computer Control Unit (CCU).

MPR-050

Micro Memory Address	Am2910 Inst	Data Pipeline Reg.	I_0	I_4-I_1	I_8-I_5	$\overline{OEB}$	$\overline{OEY}$	A_3-A_0	B_3-B_0	C_n	Comment
n	LDCT	$00F_{16}$	X	6	6	X	X	R_2	X	0	Load Counter & $R_2 \rightarrow Q$
n+1	RPCT	n+1	0	0	0	0	0	R_1	R_0	0	Unsigned Multiply

Figure 17. Micro Code for Unsigned 16 X 16 Multiply.



Note: N ∀ OVR is internally shifted into position Y_3 MSS.

Figure 18. 2's Complement Multiply, Last Cycle.

MPR-051

Memory Address	Am2910 Inst	Data Pipeline Reg.	I_0	$I_4 - I_1$	$I_8 - I_5$	$\overline{OEB}$	$\overline{OEY}$	$A_3 - A_0$	$B_3 - B_0$	C_n	Comment
n	LDCT	$00E_{16}$	X	6	6	X	X	R_2	X	0	Load Counter & $R_2 \rightarrow Q$
n+1	RPCT	n+1	0	0	2	0	0	R_1	R_0	0	2's Complement Multiply
n+2	X	X	0	0	6	0	0	R_1	R_0	Z	2's Complement Multiply (Last Cycle)

Figure 19. Microcode for 2's Complement 16 x 16 Multiply.

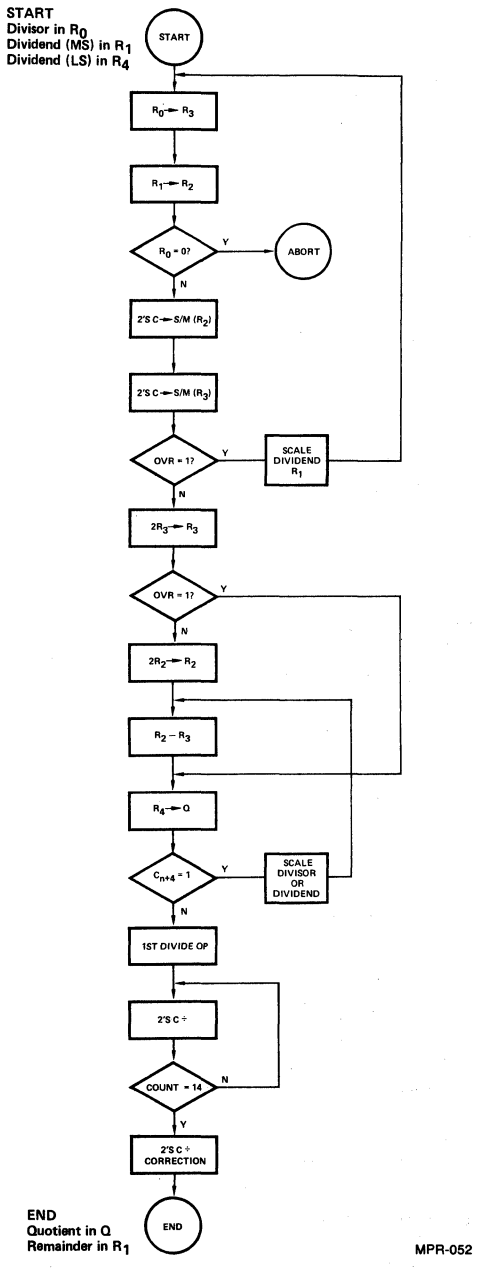


Figure 20. Division Flow Chart – Double Precision Divide.

TWO'S COMPLEMENT DIVISION

The division process is accomplished using a four quadrant non-restoring algorithm which yields an algebraically correct answer such that the divisor times the quotient plus the remainder equals the dividend. The algorithm works for both single precision and multi-precision divide operations. The only condition that needs to be met is that the absolute magnitude of the divisor be greater than the absolute magnitude of the dividend. For multi-precision divide operations the least significant bit of the dividend is truncated. This is necessary if the answer is to be algebraically correct. Bias correction is automatically provided by forcing the least significant bit of the quotient to a one, yet an algebraically correct answer is still maintained. Once the algorithm is completed, the answer may be modified to meet the user's format requirements, such as rounding off or converting the remainder so that its sign is the same as the dividend. These format modifications are accomplished using the standard Am2903 instructions.

The true value of the remainder is equal to the value stored in the working register times 2^{n-1} when n is the number of quotient digits.

The following paragraphs describe a double precision divide operation. The double precision flow chart is based upon the use of the architecture detailed in Figure 16.

Referring to the flow chart outlined in Figure 20, we begin the algorithm with the assumption that the divisor is contained in R_0, while the most significant and least significant halves of the dividend reside in R_1 and R_4 respectively. The first step is to duplicate the divisor by copying the contents of R_0 into R_3. Next the most significant half of the dividend is copied by transferring the contents of R_1 into R_2 while simultaneously checking to ascertain if the divisor (R_0) is zero. If the divisor is zero then division is aborted. If the divisor is not zero, the copy of the most significant half of the dividend in R_2 is converted from its two's complement to its sign magnitude representation. The divisor in R_3 is converted in like manner in the next step, while testing to see if the results of the dividend conversion yielded an indication on the overflow pin of the Am2903. If the output of the overflow pin is a 'one' then the dividend is -2^n and hence is the largest possible number, meaning that it cannot be less than the divisor. What must be done in this case is to scale the dividend by down shifting the upper and lower halves stored in R_1 and R_4 respectively. After scaling, the routine requires that the algorithm be reinitiated at the beginning.

Conversely, if the output of the overflow pin is not a one, the sign magnitude representation of the divisor (R_3) is shifted up in the Am2903, removing the sign while at the same time testing the results of two's complement to sign magnitude conversion of the divisor in the Am2910. If the results of the test indicate that the divisor is -2^n i.e., overflow equals one, then the lower half of the dividend is placed in the Q register

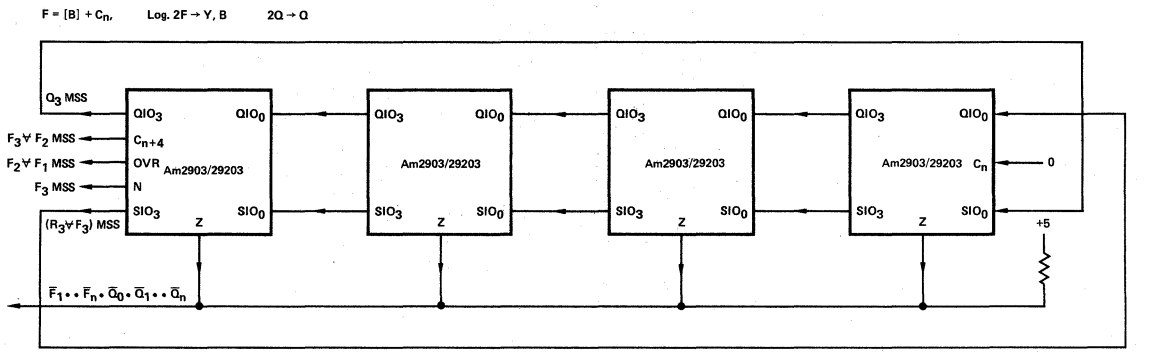


Figure 21. Double Length Normalize/First Divide Operation.

MPR-053

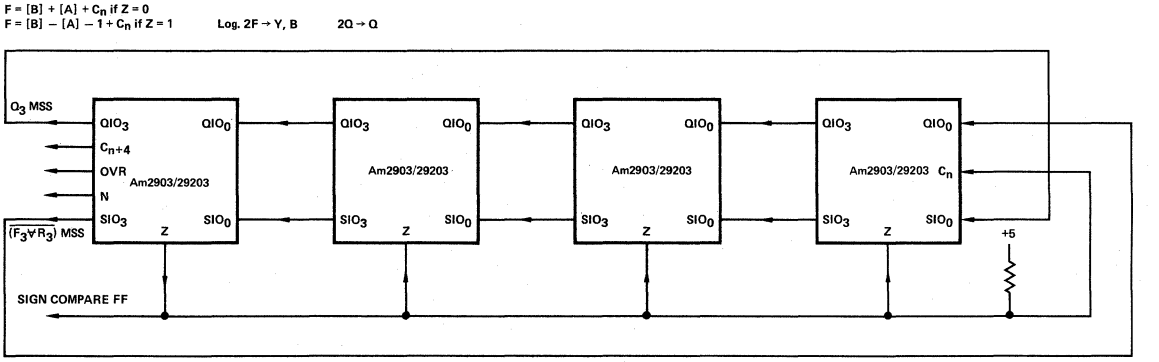


Figure 22. 2's Complement Divide.

MPR-054

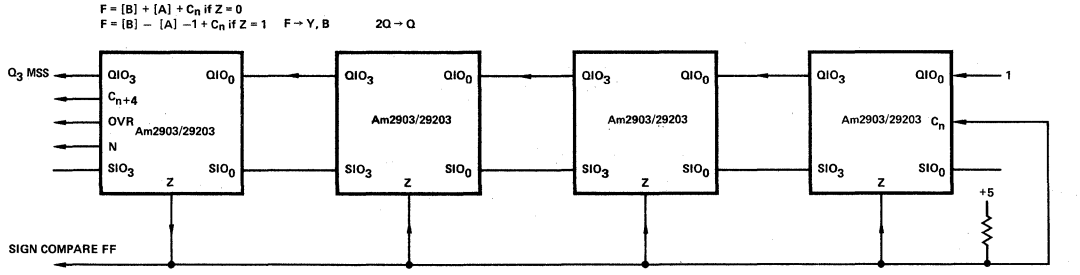


Figure 23. 2's Complement Divide Correction.

MPR-055

and division may proceed. This is possible because the divisor is now guaranteed to be greater than the dividend. If overflow is not a one then we must proceed by shifting out the sign of the sign magnitude representation of the dividend stored in R_2. At this point we are able to check if the divisor is greater than the dividend by subtracting the absolute value of the divisor (R_3) from the absolute value of the upper half of the dividend (R_2) and storing the results in R_3. Next, the least significant half of the dividend is transferred from R_4 to the Q register while simultaneously testing the carry from the result of the divisor/dividend subtraction. If the carry (C_{n+4}) is one, indicating the divisor is not greater than the dividend then a scaling operation must occur. This involves either shifting up the divisor or shifting down the dividend. If the carry is not one then the divisor is greater than the dividend and division may now begin.

Micro Memory Address	Am2910 Inst.	Data Pipeline Reg.	Am2903/29203 I_0	I_4-I_1	I_8-I_5	$\overline{EA}$	A_3-A_0	B_3-B_0	C_n	Am2922 SEL	POL	Comments
n	CONT	X	0	6	4	0	R_0	R_3	0	X	X	$R_0 \rightarrow R_3$
n+1	CJP	Abort	0	6	4	0	R_1	R_2	0	Z	1	$R_1 \rightarrow R_2$, if $R_0 = 0$ Abort
n+2	CONT	X	0	0	5	X	X	R_2	Z	X	X	2's C to S/M (R_2)
n+3	CJP	Scale Dividend	0	0	5	X	X	R_3	Z	OVR	1	2's C to S/M (R_3), if OVR ≥ 1, scale
n+4	CJP	n+7	0	4	9	X	X	R_2	0	OVR	1	Shift out sign of divisor
n+5	CONT	X	0	4	9	X	X	R_3	0	X	X	Shift out sign of divisor
n+6	CONT	X	0	2	F	0	R_2	R_3	1	X	X	Dividend − Divisor
n+7	CJP	Scale Dividend or Divisor	0	6	6	0	R_4	X	0	C_{n+4}	1	$R_4 \rightarrow Q$, if Carry = 1, scale
n+8	PUSH	$00D_{16}$	0	0	A	0	R_0	R_1	0	0	1	Loop set up and First Divide Operation
n+9	RFCT	X	0	0	C	0	R_0	R_1	Z	X	X	Test Loop Count and 2's C Divide
n+A	CONT	X	0	0	E	0	R_0	R_1	Z	X	X	2's C Divide Correction

Figure 24. Microcode for Double Precision Divide.

The first divide operation is used to ascertain the sign bit of the quotient. The two's complement divide instruction is then executed repetitively, fourteen times in the case of a sixteen bit divisor and a thirty-two bit dividend. The final step is the two's complement correction command which adjusts the quotient by allowing the least significant bit of the quotient to be set to one. At the end of the division algorithm the sixteen bit quotient is found in the Q register while the remainder now replaces the most significant half of the dividend in R_1. It should be noted that the remainder must be shifted down fifteen places to represent its true value. The interconnections for these instructions are shown in Figures 21, 22, 23. Using a typical CCU as shown in Figure 16, the double precision divide operation requires only eleven lines of microcode, as shown in Figure 24.

For those applications that require truncation instead of bias correction, the same algorithm as above should be implemented except one additional Two's Complement Divide instruction should be used in lieu of the Two's Complement Divide Correction and Remainder instruction. However, this technique results in an invalid remainder.

It is possible to do multiple-precision divide operations beyond the double precision divide shown above. For example, to do a triple precision divide for a 16-bit CPU, the upper two thirds of the dividend are stored in R_1 and Q as in the case for double precision divide. The lower third of the dividend is stored in a scratch register, R_5. After checking that the magnitude of the divisor is greater than the magnitude of the dividend, using the same tests as defined in Figure 20, the procedure is as follows:

1. Execute a Double Length Normalize/First Divide Operation instruction.
2. Execute the Two's Complement Divide instruction fifteen times.
3. Transfer the contents of Q, the most significant half of the quotient, to R_2.
4. Transfer R_5 to Q.
5. Execute the Two's Complement Divide instruction fifteen times.
6. Execute the Two's Complement Divide Correction and Remainder instruction.

The upper half of the quotient is then in R_2, the lower half of the quotient is in Q and the remainder is in R_1. The flow chart for this is shown in Figure 25. This technique can be expanded for any precision which is required.

BYTE SWAP

The multi-port architecture of the Am2903 allows for easy implementation of high- and low-order byte swapping. Figure 26 outlines a byte swap implementation utilizing two data ports. Initially, the lower order 8-bit byte is stored in devices 1 and 2, while the high-order byte is in devices 3 and 4. When the user wishes to exchange the two bytes, the register location of the desired word is placed on the B address port. When the byte swap line is brought LOW, the bytes to be swapped will be flowing from the DB ports of the Am2903 through the Am2958/2959 Three-state Buffers. The outputs of the three-

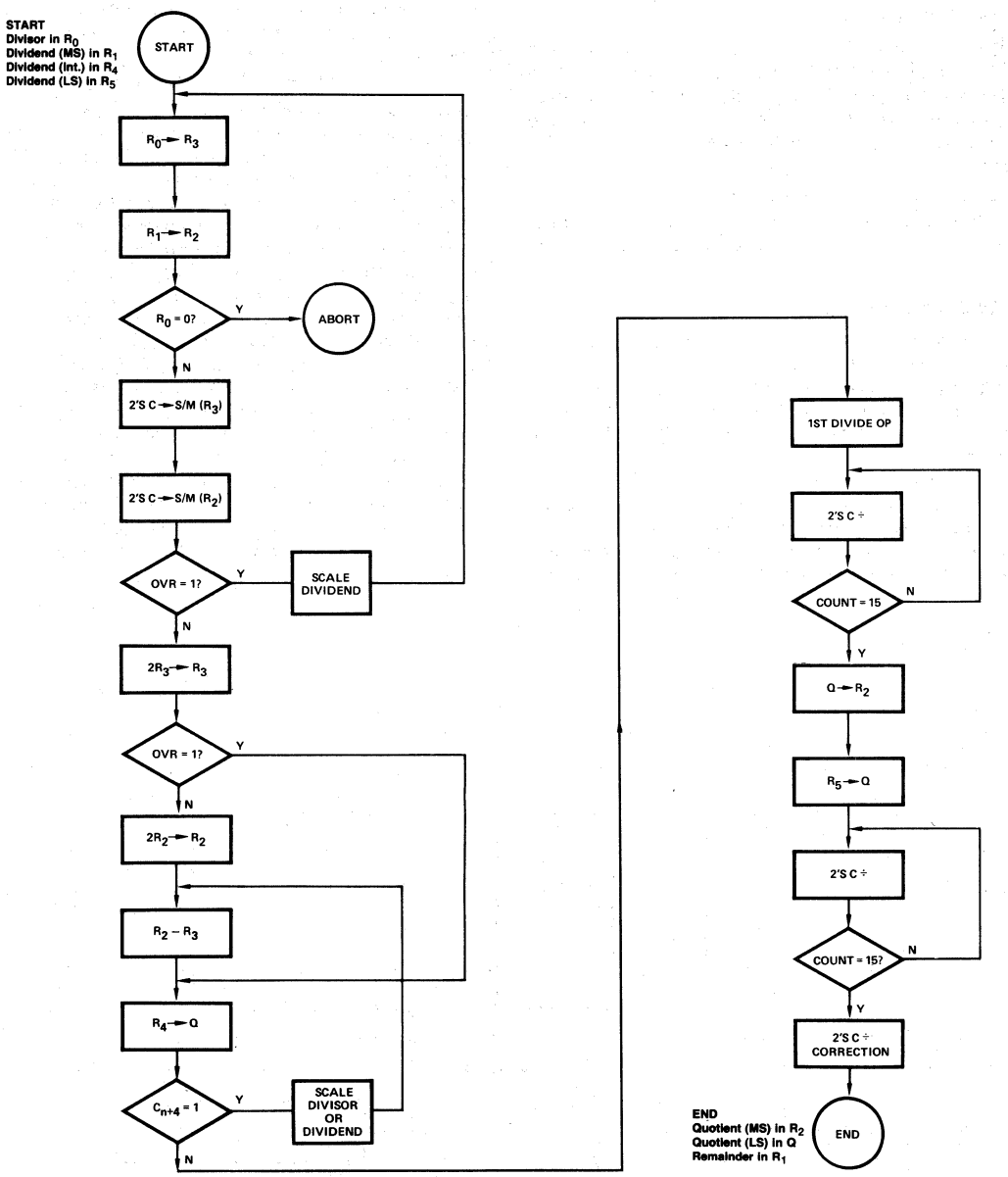


Figure 25. Division Flow Chart – Triple Precision Divide.

MPR-056

state buffers are permuted such that the byte swap is achieved. The resultant permuted data is presented to the DA ports of the Am2903 where it is re-loaded into the memories of the Am2903 on the next positive edge of CP using the source and function commands of F = A plus C_n (C_n = 0) for the Am2958 or F = A plus C_n (C_n = 0) for the Am2959 and the destination command F Y, B.

A higher speed technique for achieving the byte swap operation is illustrated in Figure 27. Instead of inputting the permuted data via the DA ports, the permuted data is entered via the Y input/output ports with $\overline{OE_Y}$ held HIGH. This technique bypasses the ALU, thus allowing faster operation. The Am2903 destination command F → Y, B should be used.

WORD/BYTE OPERATIONS

The Am29203 allows for Word/Byte Operations. Figure 28 pictures a 16-bit system which is capable of doing word or byte (lower half of word) operations.

In the Byte mode the BYTE/$\overline{\text{WORD}}$ line is HIGH which in turn asserts a LOW on the $\overline{\text{W}}$/$\overline{\text{MSS}}$ input of Device 2 making it the MSS device. At the same time the multiplexer selects the status flags of Device 2. The $\overline{\text{IEN}}$ and $\overline{\text{OE}}_Y$ of Devices 4 and 3 are forced HIGH which disables them from writing into RAM or onto the Y bus.

In the word mode Device 4 is the MSS device and the multiplexer selects its status flags. The $\overline{\text{IEN}}$ inputs are brought low which enables writing in to RAM. The $\overline{\text{OE}}_Y$ is also allowed to go low.

MEMORY EXPANSION

Both the Am2903 and Am29203 allow for a theoretically infinite memory expansion, but the technique is slightly different. (The Am29203 allows writing less than a full word, e.g., a byte.) Figure 29, *Am2903 and Am29705*, pictures a 4-bit slice of a system which has 48 words of RAM and 16 words of ROM. RAM storage is provided by the Am2903 and the Am29705s. The Am29705 RAM is functionally identical to the Am2903 RAM. The Am27S19 is used to store constants and masks and is addressable from address port A only. The system is organized around five data buses. Inter-bus communication may be done through the Am29705s or the Am2903. The memory addressing scheme specifies the data source for the R input of the ALU eminating from the register locations specified by address field A. A_{0-3} addresses 16 memory locations in each chip while address bits A_{4-6} are decoded and used for the output enable for the desired chip. The B address field is used to select the S input of the ALU and the C field is used to specify the register location where the result of the ALU operation is to be stored.

Bits B_{0-3} are for source register addressing in each chip. Bits B_4 and B_5 are used for chip output enable selection. C_{0-3} access the 16 destination addresses on each chip while bits C_4 and C_5 control the Write Enable of the desired chip. The source and destination register address are multiplexed such that when the clock is HIGH, the source register address is presented to the B address ports of the RAM's. The Instruction Enable ($\overline{\text{IEN}}$) is HIGH at this time. The data flows from the Y port or the internal B port as selected by the decoder whose inputs are B_4 and B_5. When the clock goes LOW, the data eminating from the selected Y outputs of the Am29705's and the RAM outputs of the Am2903 are latched and the destination address is now selected for use by the RAM address lines. When the destination address stabilizes on the address lines, the $\overline{\text{IEN}}$ pin is brought LOW. The $\overline{\text{WRITE}}$ output of the Am2903 will now go LOW, enabling the decoder sourced by address bits C_4 and C_5. The selected decoder line will go LOW, allowing the desired memory location to be written into. To switch between two- and three-address architecture, the user simply makes the source and destination addresses the same; i.e., $B_{0-3} = C_{0-3}$ and $B_{4-5} = C_{4-5}$. For two-address architecture, the MUX is removed from the circuit.

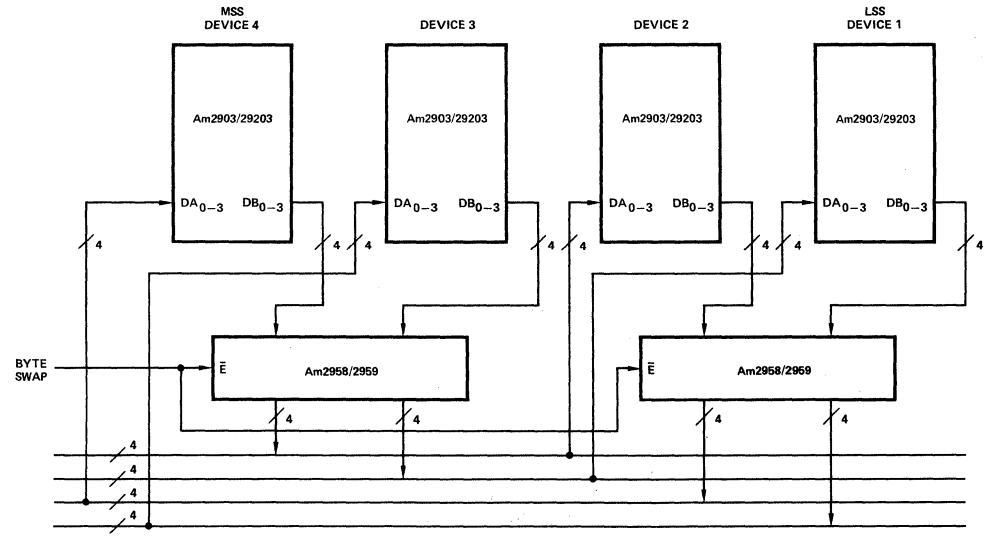


Figure 26. Byte Swap.

MPR-057

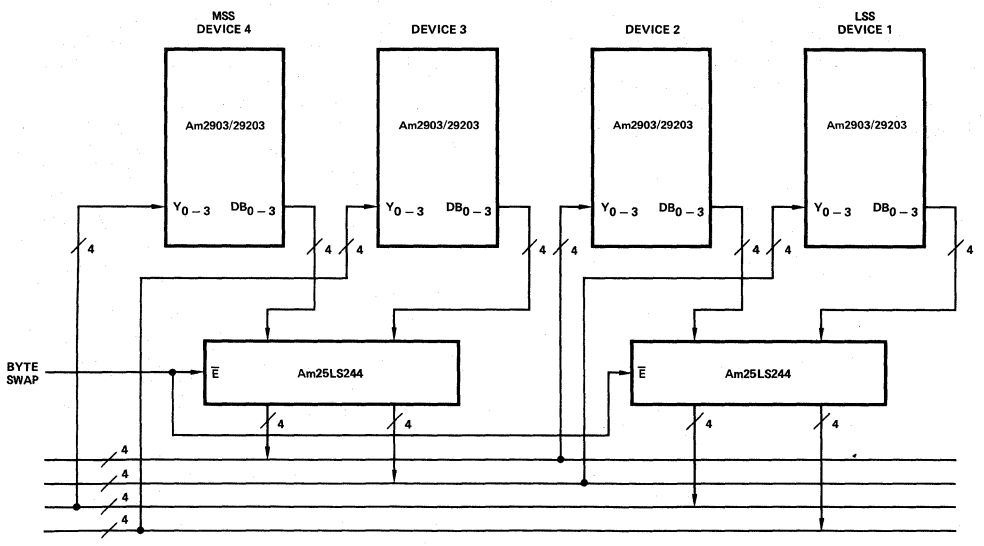
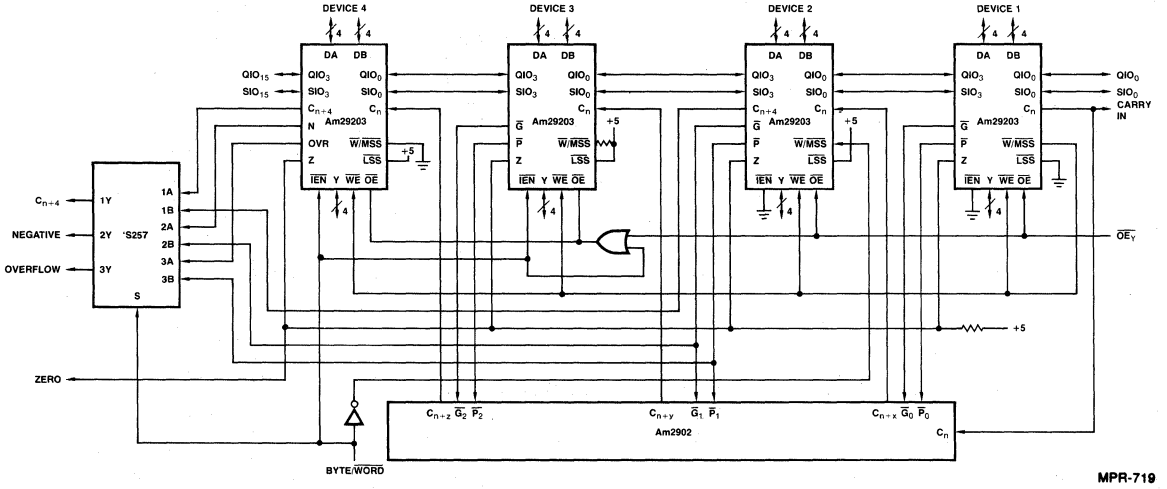


Figure 27. High Speed Byte Swap.

MPR-058



MPR-719

Figure 28. Connections for Word/Byte Operations (Am29203 Only).

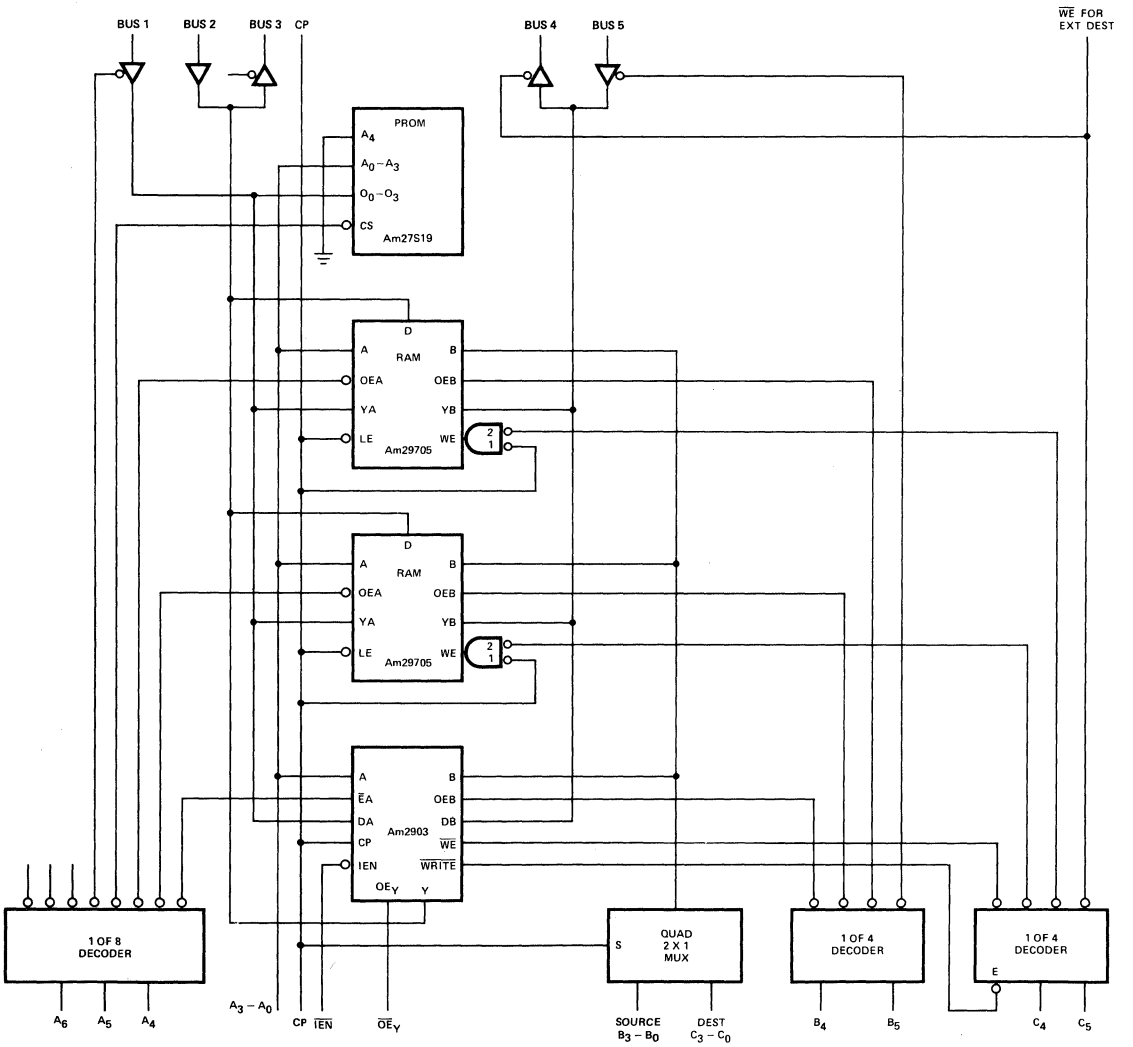


Figure 29. Expanded Memory on Am2903.

MPR-059

Memory Expansion with the Am29203

The expansion scheme using the Am29203 and Am29705 is only slightly different from that for the Am2903, and is illustrated in Figure 30. The difference is due to the fact that the $\overline{\text{WRITE}}$ signal from the Am29203 is not internally gated by $\overline{\text{IEN}}$. This gating is performed external to the Am29203, either in a gate or, as shown in Figure 30, by using the enable on the chip select decoder. The advantage of separating the write signal from the $\overline{\text{IEN}}$ signal is that writing can be controlled over less than the full word length. For example, in a 16-bit system, the lower two devices can have one $\overline{\text{IEN}}$ signal and the upper two devices a second $\overline{\text{IEN}}$ signal. Controlling these two signals separately allows data to be written in either byte without disturbing the other byte. The 2- and 3-address architecture is handled in the same way as with the Am2903.

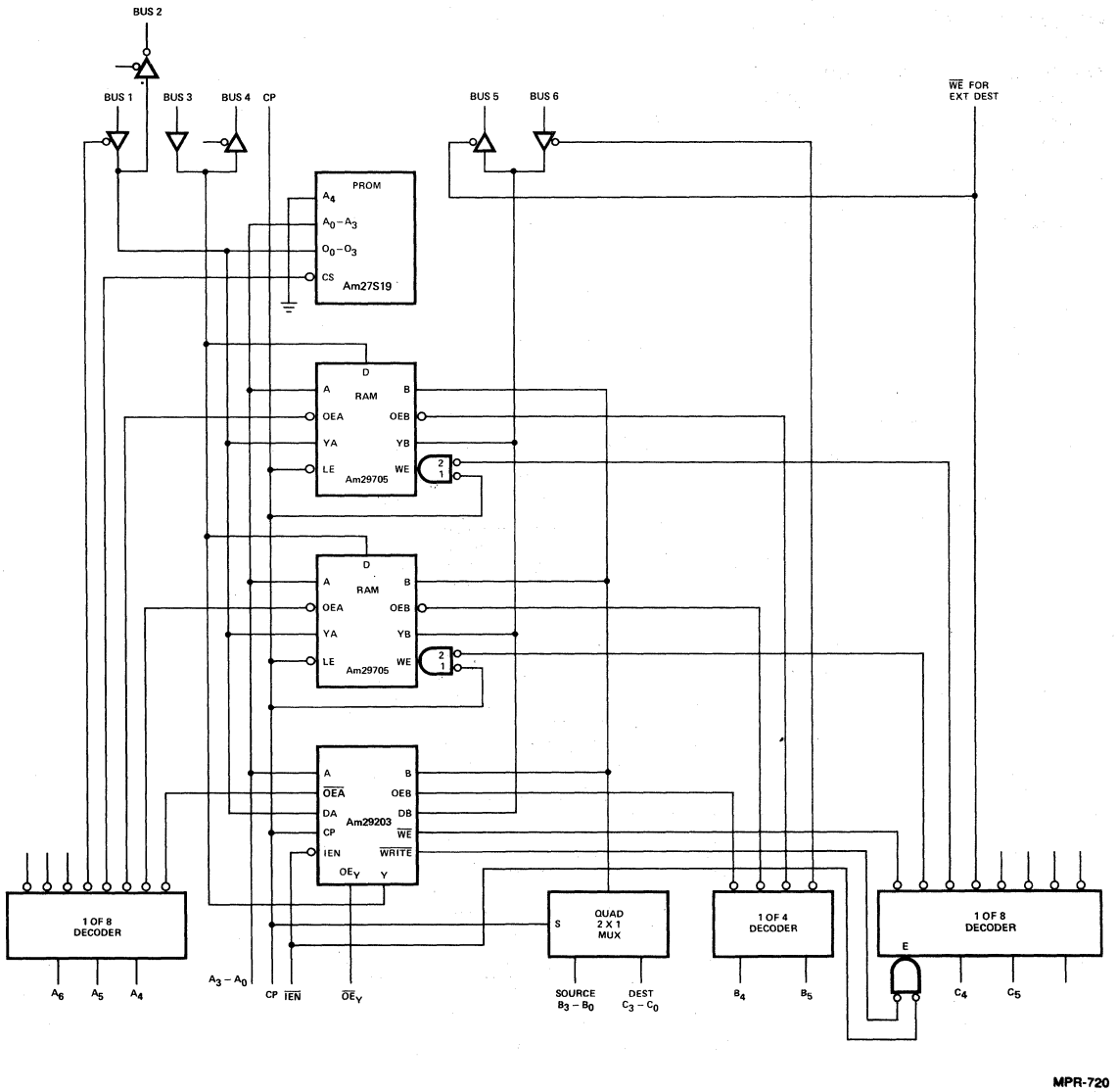


Figure 30. Expanded Memory for Am29203.

Am2903A

The Superslice®

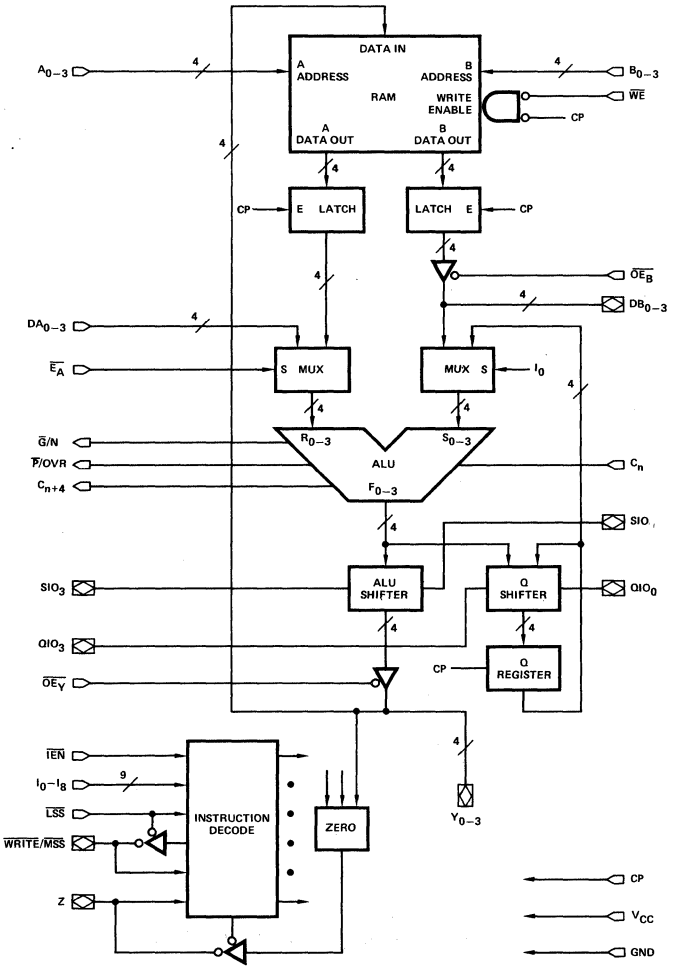
ADVANCED INFORMATION

3

DISTINCTIVE CHARACTERISTICS

- **Second generation of Am2903 Superslice® –**
  Improved design/process results in fastest version of the Am2903.
- **Plug-in replacement for Am2903 –**
  The Am2903A is a pin-for-pin replacement for the original version of the Am2903. Only the switching speeds have changed.
- **At least 30% faster –**
  The design objective is for the Am2903A to be at least 30% faster than the original Am2903 on critical paths.

BLOCK DIAGRAM



MPR-721

Am29705A

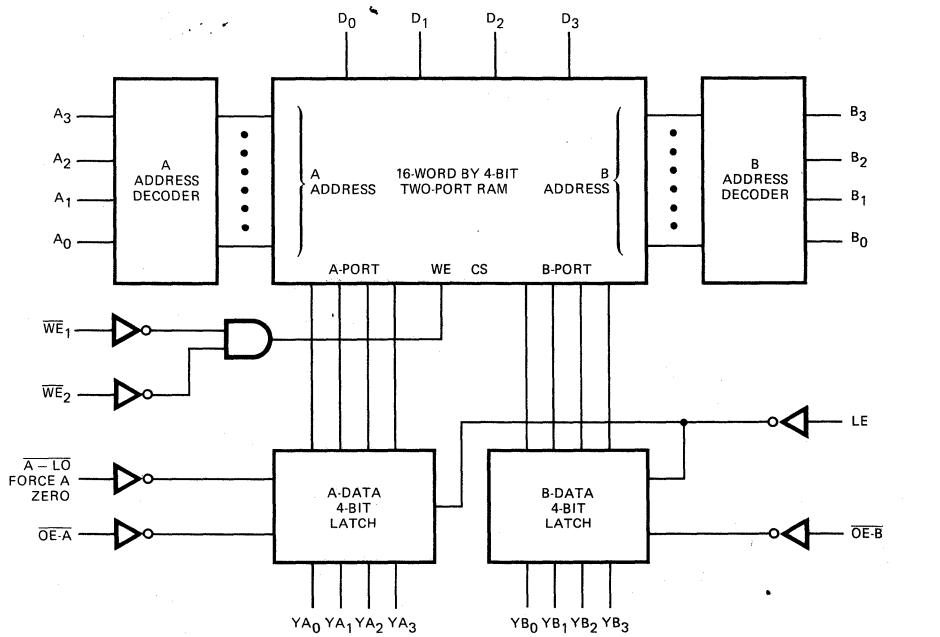
16-Word by 4-Bit 2-Port RAM

ADVANCE INFORMATION

DISTINCTIVE CHARACTERISTICS

- **Faster Version of the Am29705**
  The Am29705A has a design objective of a 30-40% speed improvement on the critical paths versus the Am29705.
- **Plug-in Replacement for the Am29705**
  The Am29705A is a pin-for-pin replacement for the Am29705. Systems using the Am29705 will be able to use the Am29705A instead with no design changes.

LOGIC DIAGRAM



Am8041

Universal Peripheral Interface 8-Bit Microcomputer

DISTINCTIVE CHARACTERISTICS

- Fully compatible with Am8080A, Am8085A, and Am8048 microprocessors
- Single level interrupts
- 8-bit CPU plus ROM, RAM, I/O, timer and clock in a single package
- Single +5V supply
- Alternative to custom LSI
- Pin compatible ROM versions
- 1K x 8 ROM, 64 x 8 RAM, 18 programmable I/O pins
- Asynchronous data register for interface to master processor
- Expandable I/O

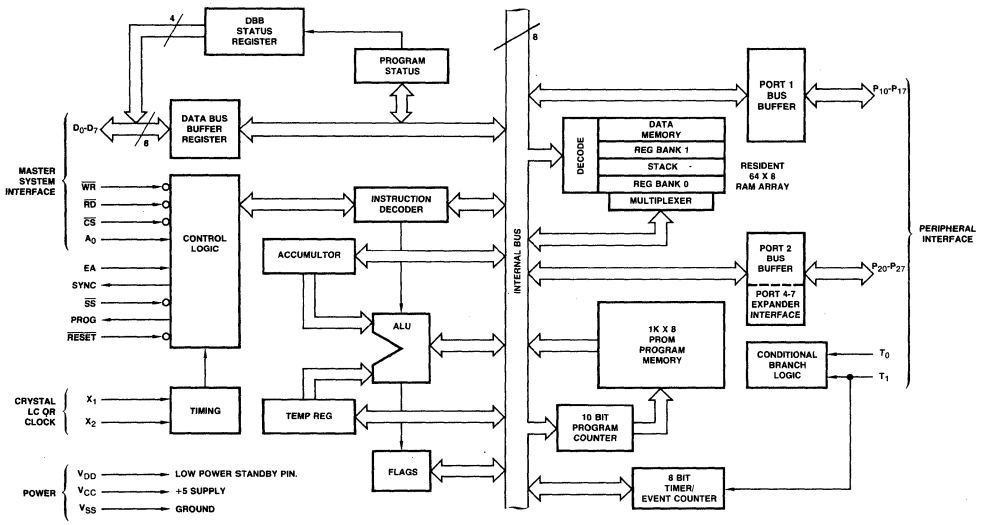
GENERAL DESCRIPTION

The AMD® Am8041 is a general purpose, programmable interface device designed for use with a variety of 8-bit microprocessor systems. It contains a low-cost microcomputer with program memory, data memory, 8-bit CPU, I/O ports, timer/counter, and clock in a simple 40-pin package. Interface registers are included to enable it to function as a peripheral controller in Am8080A, Am8085A, Am8048 and other 8-bit systems.

The Am8041 has 1K words of program memory and 64 words of data memory on-chip. The device has two 8-bit, TTL compatible I/O ports and two test inputs. Individual port lines can function as either inputs or outputs under software control. I/O can be expanded with Am8243 device which is directly compatible and has 16 I/O times. An 8-bit programmable timer/counter is included in the device for generating timing sequences or counting external inputs. Additional features include single 5V supply, low-power standby mode, single-level interrupt, and dual working register banks.

As a complete microcomputer, the Am8041 provides more flexibility for the designer than conventional LSI interface devices. It is designed to be an efficient controller. Applications include keyboard scanning, printer control, display multiplexing and similar functions which involve interfacing peripheral devices to microprocessor systems.

BLOCK DIAGRAM



MOS-159

ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Hermetic DIP	0°C to ≤ T_A ≤ +70°C	AM8041CC
Molded DIP		Am8041PC
Hermetic DIP	−55°C ≤ T_A ≤ +125°C	AM8041DM

Am8048/8035

Single Chip 8-Bit Microcomputers

DISTINCTIVE CHARACTERISTICS

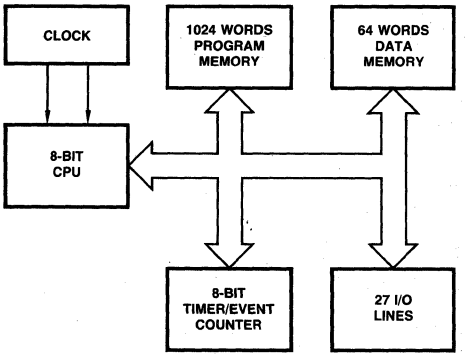
- 8-bit CPU, ROM, RAM, I/O in single package
- Single +5V supply
- All instructions 1 or 2 cycles
- Over 90 instructions: 70% single byte
- 1K x 8 ROM
- 64 x 8 RAM
- 27 I/O lines
- Interval timer/event counter
- Easily expandable memory and I/O
- Single level interrupt
- 100% reliability assurance testing to MIL-STD-883

GENERAL DESCRIPTION

The Am8048 contains a 1k x 8 program memory, a 64 x 8 RAM data memory, 27 I/O lines, and an 8-bit timer/counter in addition to on board oscillator and clock circuits. For systems that require extra capability, the Am8048 can be expanded using standard memories and Am9080A peripherals. The Am8035 is the equivalent of an Am8048 without program memory.

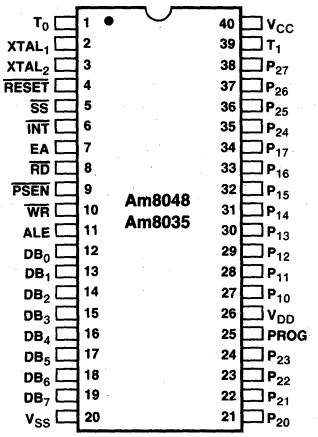
The microprocessor is designed to be an efficient controller. The Am8048 has extensive bit handling capability as well as facilities for both binary and BCD arithmetic. Efficient use of program memory results from an instruction set consisting mostly of single byte instructions and no instructions over two bytes in length.

BLOCK DIAGRAM



MOS-163

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

MOS-164

ORDERING INFORMATION

Package Type	Ambient Temperature Specification	Order Numbers	
Hermetic DIP	0°C ≤ T_A ≤ 70°C	AM8048CC	AM8035CC
Molded DIP		AM8048PC	AM8035PC

Am8085A/Am8085A-2/Am9085ADM

Single Chip 8-Bit N-Channel Microprocessor

DISTINCTIVE CHARACTERISTICS

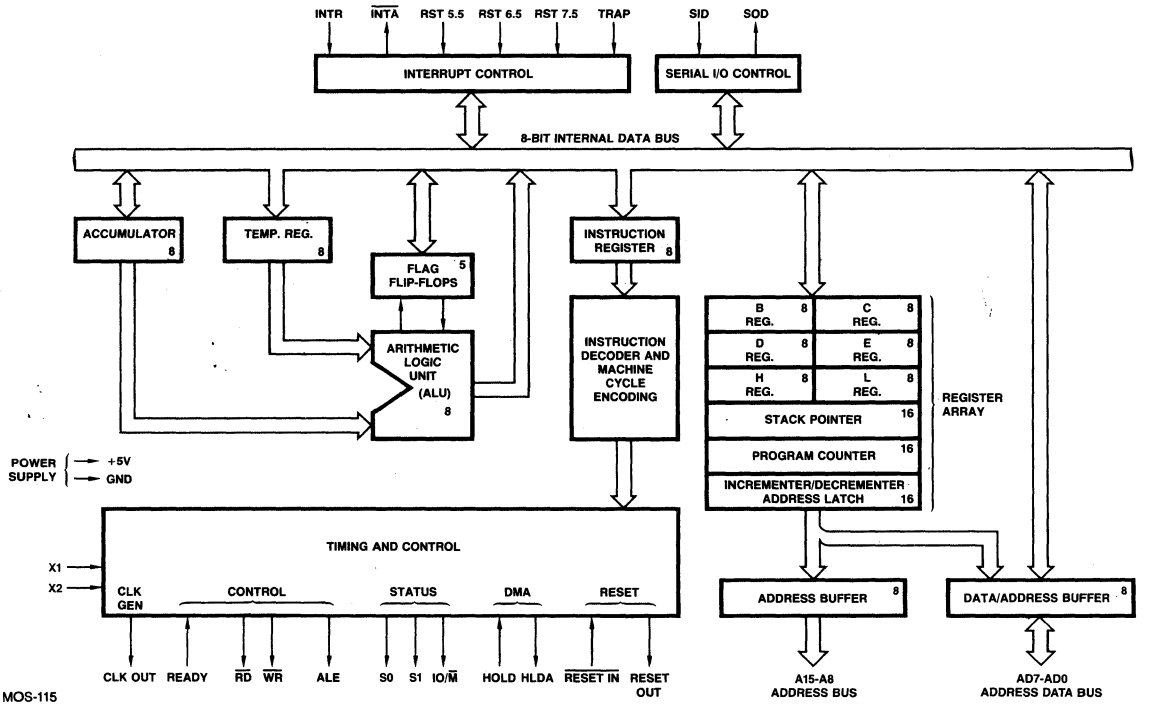
- Complete 8-bit parallel CPU
- On-chip system controller; advanced cycle status information available for large system control
- Four vectored interrupts (one is non-maskable)
- On-chip clock generator (with external crystal, LC or R/C network)
- Serial in/serial out port
- Decimal, binary and double precision arithmetic
- Direct addressing capability to 64K bytes of memory
- 1.3μs instruction cycle (Am8085A)
- 0.8μs instruction cycle (Am8085A-2)
- 100% software compatible with Am9080A
- Single +5V power supply
- 100% MIL-STD-883, Level C processing

GENERAL DESCRIPTION

The Am8085A is a new generation, complete 8-bit parallel central processing unit (CPU). Its instruction set is 100% software compatible with the Am9080A microprocessor. Specifically, the Am8085A incorporates all of the features that the Am8224 (clock generator) and Am8228 (system controller) provided for the Am9080A. The Am8085A-2 is a faster version of the Am8085A.

The Am8085A uses a multiplexed Data Bus. The address is split between the 8-bit address bus and the 8-bit data bus. The on-chip address latches of Am8155/Am8355 memory products allows a direct interface with Am8085A. The Am8085A components, including various timing compatible support chips, allow system speed optimization.

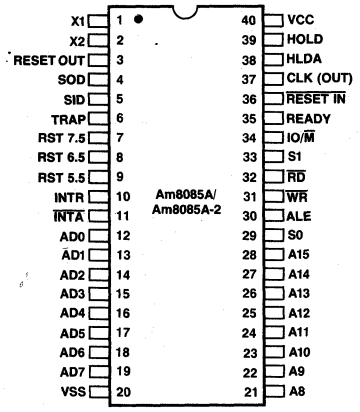
BLOCK DIAGRAM



ORDERING INFORMATION

Package Type	Temperature Range	Maximum Clock Frequency	
		3MHz	5MHz
Molded DIP	0°C ≤ T_A ≤ 70°C	AM8085APC/P8085A AM8085ADC/C8085A AM8085ACC/D8085A	AM8085A-2PC/P8085A-2 AM8085A-2DC/C8085A-2 AM8085A-2CC/D8085A-2
Hermetic DIP			
Hermetic DIP	−55°C ≤ T_A ≤ +125°C	AM9085ADM	

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

Figure 1. MOS-116

Am8085A FUNCTIONAL PIN DEFINITION

The following describes the function of each pin:

A8-A15 (Output 3-State)

Address Bus – the most significant eight bits of the memory address or the eight bits of the I/O address, 3-stated during Hold and Halt modes and during RESET.

AD0-AD7 (Input/Output 3-State)

Multiplexed Address/Data Bus – lower eight bits of the memory address (or I/O address) appear on the bus during the first clock cycle of a machine cycle. It then becomes the data bus during the second and third clock cycles.

Three-stated during Hold and Halt modes.

ALE (Output)

Address Latch Enable – it occurs during the first clock cycle of a machine cycle and enables the address to get latched into the on-chip latch of peripherals. The falling edge of ALE is set to guarantee setup and hold times for the address information. The falling edge ALE can also be used to strobe the status information. ALE in never 3-stated.

S0, S1 (Output)

Data Bus Status. Encoded status of the bus cycle.

S1	S0	
0	0	HALT
0	1	WRITE
1	0	READ
1	1	FETCH

S1 can be used as an advanced R/$\overline{W}$ status.

$\overline{RD}$ (Output 3-State)

READ – A low level on $\overline{RD}$ indicates the selected memory or I/O device is to be read and that the Data Bus is available for the data transfer. Three-stated during Hold and Halt and during RESET.

$\overline{WR}$ (Output 3-State)

WRITE – A low level on $\overline{WR}$ indicates the data on the Data Bus is to be written into the selected memory or I/O location. Data is set up at the trailing edge of $\overline{WR}$. Three-stated during Hold and Halt modes.

READY (Input)

If READY is high during a read or write cycle, it indicates that the memory or peripheral is ready to send or receive data. If READY is low, the CPU will wait an integral number of clock cycles for READY to go high before completing the read or write cycle.

HOLD (Input)

HOLD – indicates that another Master is requesting the use of the Address and Data Buses. The CPU, upon receiving the Hold request, will relinquish the use of buses as soon as the completion of the current machine cycle. Internal processing can continue. The processor can regain the buses only after the Hold is removed. When the Hold is acknowledged, the Address, Data, $\overline{RD}$, $\overline{WR}$ and IO/$\overline{M}$ lines are three-stated.

HLDA (Output)

HOLD ACKNOWLEDGE – indicates that the CPU has received the Hold request and that it will relinquish the buses in the next clock cycle. HLDA goes low after the Hold request is removed. The CPU takes the buses one half clock cycle after HLDA goes LOW.

INTR (Input)

INTERRUPT REQUEST – is used as a general purpose interrupt. It is sampled only during the next to the last clock cycle of the instruction. If it is active, the Program Counter (PC) will be inhibited from incrementing and an $\overline{INTA}$ will be issued. During this cycle a RESTART or CALL instruction can be inserted to jump to the interrupt service routine. The INTR is enabled and disabled by software. It is disabled by Reset and immediately after an interrupt is accepted.

$\overline{INTA}$ (Output)

INTERRUPT ACKNOWLEDGE – is used instead of (and has the same timing as) $\overline{RD}$ during the Instruction cycle after an INTR is accepted. It can be used to activate the Am9519 Interrupt chip or some other interrupt port.

RST 5.5
RST 6.5
RST 7.5
} (Inputs)

RESTART INTERRUPTS – these three inputs have the same timing as INTR except they cause an internal RESTART to be automatically inserted.

RST 7.5 ——► Highest Priority
RST 6.5
RST 5.5 ——► Lowest Priority

The priority of these interrupts is ordered as shown above. These interrupts have a higher priority than the INTR. However they may be individually masked out using the SIM instructions.

TRAP (Input)

Trap interrupt is a non-maskable restart interrupt. It is recognized at the same time as INTR. It is unaffected by any mask or Interrupt Enable. It has the highest priority of any interrupt.

$\overline{\text{RESET IN}}$ (Input)

Reset sets the Program Counter to zero and resets the Interrupt Enable and HLDA flip-flops. None of the other flags or registers (except the instruction register) are affected. The CPU is held in the reset condition as long as RESET is applied.

RESET OUT (Output)

Indicates CPU is being reset. Can be used as a system RESET. The signal is synchronized to the processor clock.

X1, X2 (Input)

Crystal, LC or R/C network connections to set the internal clock generator. X1 can also be an external clock input instead of a crystal. The input frequency is divided by 2 to give the internal operating frequency.

CLK (Output)

Clock Output for use as a system clock when a crystal or R/C network is used as an input to the CPU. The period of CLK is twice the X1, X2 input period.

IO/$\overline{\text{M}}$ (Output)

IO/$\overline{\text{M}}$ indicates whether the Read/Write is to memory or I/O. 3-stated during Hold and Halt modes.

SID (Input)

Serial input data line. The data on this line is loaded into accumulator bit 7 whenever a RIM instruction is executed.

SOD (Output)

Serial output data line. The output SOD is set or reset as specified by the SIM instruction.

VCC

+5 volt supply.

VSS

Ground reference.

FUNCTIONAL DESCRIPTION

The Am8085A is a complete 8-bit parallel central processor. It is designed with N-channel depletion loads and requires a single +5 volt supply. Its basic clock speed is 3MHz (5MHz: Am8085A-2) thus improving on the present Am9080's performance with higher system speed. Also it is designed to fit into a minimum system of three ICs: The CPU, a RAM/IO, and a ROM or PROM/IO chip.

The Am8085A uses a multiplexed Data Bus. The address is split between the higher 8-bit Address Bus and the lower 8-bit Address/Data Bus. During the first cycle the address is sent out. The lower eight bits are latched into the peripherals by the Address Latch Enable (ALE). During the rest of the machine cycle the Data Bus is used for memory or I/O data.

The Am8085A provides $\overline{\text{RD}}$, $\overline{\text{WR}}$ and IO/$\overline{\text{Memory}}$ signals for bus control. An Interrupt Acknowledge signal ($\overline{\text{INTA}}$) is also provided. Hold, Ready and all Interrupts are synchronized. The Am8085A also provides serial input data (SID) and serial output data (SOD) lines for simple serial interface.

In addition to these features, the Am8085A has three maskable, restart interrupts and one non-maskable trap interrupt.

Am8085A vs. Am8080A

The Am8085A includes the following features on-chip in addition to all of the Am9080A functions.

a. Internal clock generator
b. Clock output
c. Fully synchronized Ready
d. Schmitt action on $\overline{\text{RESET IN}}$
e. RESET OUT pin
f. $\overline{\text{RD}}$, $\overline{\text{WR}}$ and IO/$\overline{\text{M}}$ Bus Control Signals
g. Encoded Status information
h. Multiplexed Address and Data
i. Direct Restarts and non-maskable Interrupt
j. Serial Input/Output lines

The internal clock generator requires an external crystal or R/C network. It will oscillate at twice the basic CPU operating frequency. A 50% duty cycle, two phase, non-overlapping clock is generated from this oscillator internally and one phase of the clock (ϕ2) is available as an external clock. The Am8085A directly provides the external RDY synchronization previously provided by the Am8224. The $\overline{\text{RESET IN}}$ input is provided with a Schmitt action input so that power-on reset only requires a resistor and capacitor. RESET OUT is provided for System RESET.

The Am8085A provides $\overline{\text{RD}}$, $\overline{\text{WR}}$ and IO/$\overline{\text{M}}$ signals for Bus control. An $\overline{\text{INTA}}$ which was previously provided by the Am8228 in Am9080A systems is also included in Am8085A.

STATUS INFORMATION

Status information is directly available from the Am8085A. ALE serves as a status strobe. The status is partially encoded and provides the user with advanced timing of the type of bus transfer being done. IO/$\overline{\text{M}}$ cycle status signal is provided directly also. Decoded S0, S1 carries the following status information:

MACHINE CYCLE STATUS

IO/$\overline{\text{M}}$	S1	S0	Status
0	0	1	Memory write
0	1	0	Memory read
1	0	1	I/O write
1	1	0	I/O read
0	1	1	Opcode fetch
1	1	1	Interrupt Acknowledge
•	0	0	Halt
•	X	X	Hold
•	X	X	Reset

• = 3-state (high impedance)
X = unspecified

S1 can be interpreted as R/$\overline{\text{W}}$ in all bus transfers.

In the Am8085A the eight LSB of address are multiplexed with the data instead of status. The ALE line is used as a strobe to enter the lower half of the address into the memory or peripheral address latch. This also frees extra pins for expanded interrupt capability.

INTERRUPT AND SERIAL I/O

The Am8085A/Am8085A-2 has 5 interrupt inputs: INTR, RST 5.5, RST 6.5, RST 7.5 and TRAP. INTR is identical in function to the Am8080A INT. Each of three RESTART inputs, 5.5, 6.5, 7.5, has programmable mask. TRAP is also a RESTART interrupt except it is non-maskable.

The three RESTART interrupts cause the internal execution of RST (saving the program counter in the stack and branching to the RESTART address) if the interrupts are enabled and if the interrupt mask is not set. The nonmaskable TRAP causes the internal execution of a RST independent of the state of the interrupt enable or masks.

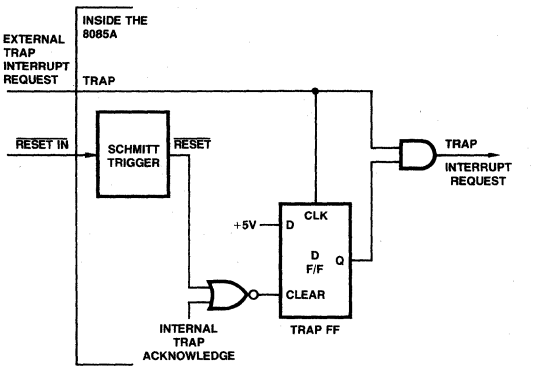
Name	RESTART Address (Hex)
TRAP	24_{16}
RST 5.5	$2C_{16}$
RST 6.5	34_{16}
RST 7.5	$3C_{16}$

There are two different types of inputs in the restart interrupts. RST 5.5 and RST 6.5 are high level-sensitive like INTR (and INT on the Am8080A) and are recognized with the same timing as INTR. RST 7.5 is rising edge-sensitive. For RST 7.5, only a pulse is required to set an internal flip-flop which generates the internal interrupt request. The RST 7.5 request flip-flop remains set until the request is serviced. Then it is reset automatically. This flip-flop may also be reset by using the SIM instruction or by issuing a RESET IN to the Am8085A. The RST 7.5 internal flip-flop will be set by a pulse on the RST 7.5 pin even when the RST 7.5 interrupt is masked out.

The status of the three RST interrupt masks can only be affected by the SIM instruction and RESET IN.

The interrupts are arranged in a fixed priority that determines which interrupt is to be recognized if more than one is pending as follows: TRAP – highest priority, RST 7.5, RST 6.5, RST 5.5, INTR – lowest priority. This priority scheme does not take into account the priority of a routine that was started by a higher priority interrupt. RST 5.5 can interrupt a RST 7.5 routine if the interrupts were re-enabled before the end of the RST 7.5 routine.

The TRAP interrupt is useful for catastrophic errors such as power failure or bus error. The TRAP input is recognized just as any other interrupt but has the highest priority. It is not affected by any flag or mask. The TRAP input is both edge and level sensitive. The TRAP input must go high and remain high to be acknowledged, but will not be recognized again until it goes low, then high again. This avoids any false triggering due to noise or logic glitches. The following diagram illustrates the TRAP interrupt request circuitry within the Am8085A.



MOS-117

Note that the servicing of any interrupt (TRAP, RST 7.5, RST 6.5, RST 5.5, INTR) disables all future interrupts (except TRAPs) until an EI instruction is executed.

The TRAP interrupt is special in that it preserves the previous interrupt enable status. Performing the first RIM instruction following a TRAP interrupt allows you to determine whether interrupts were enabled or disabled prior to the TRAP. All subsequent RIM instructions provide current interrupt enable status.

The serial I/O system is also controlled by the RIM and SIM instructions. SID is read by RIM, and SIM sets the SOD data.

MAXIMUM RATINGS above which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
V_{CC} with Respect to V_{SS}	−0.5V to +7.0V
All Signal Voltages with Respect to V_{SS}	−0.5V to +7.0V
Power Dissipation	1.5W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	T_A	VCC	VSS
Am8085A/Am8085A-2	0 to 70°C	+5.0V ±5%	0V
Am9085ADM	−55 to +125°C	+5.0V ±10%	0V

DC CHARACTERISTICS

			Am8085A/Am8085A-2		Am9085ADM		
Parameter	**Description**	**Test Conditions**	**Min**	**Max**	**Min**	**Max**	**Units**
VIL	Input Low Voltage		−0.5	+0.8	−0.5	+0.8	Volts
VIH	Input High Voltage		2.0	VCC+0.5	2.2	VCC+0.5	Volts
VOL	Output Low Voltage	IOL = 2.0mA		0.45		0.45*	Volts
VOH	Output High Voltage	IOH = −400μA	2.4		2.4		Volts
ICC	Power Supply Current			170		200	mA
IIL†	Input Leakage	VIN = VCC		±10		±10	μA
ILO	Output Leakage	0.45V ≤ VOUT ≤ VCC		±10		±10	μA
VILR	Input Low Level, RESET		−0.5	+0.8	−0.5	+0.8	Volts
VIHR	Input High Level, RESET		2.4	VCC+0.5	2.4	VCC+0.5	Volts
VHY	Hysteresis, RESET		0.25		0.25		Volts

*IOL = 1.6mA

†Except Pin 1 and Pin 2.

AC CHARACTERISTICS

Parameters	Description		Am8085A Min	Am8085A Max	Am8085A-2 Min	Am8085A-2 Max	Am9085ADM Min	Am9085ADM Max	Units
tCYC	CLK Cycle Period		320	2000	200	2000	320	2000	
tr, tf	CLK Rise and Fall Time			30		30		30	
tAL	A8-A15 Valid before Trailing Edge of ALE (Note 1)		115		50		115		
tACL	A0-A7 Valid to Leading Edge of Control		240		115		240		
tXKR	X1 Rising to CLK Rising		30	120	30	100	30	120	
tXKF	X1 Rising to CLK Falling		30	150	30	110	30	150	
t1	CLK Low Time	Standard 150pF Loading	80		40		80		
		Lightly Loaded (Note 8)	100				100		
t2	CLK High Time	Standard 150pF Loading	120		70		120		
		Lightly Loaded (Note 8)	150				150		
tALL	A0-A7 Valid to Leading Edge of Control		90		50		90		
tLRY	ALE to READY Stable			110		30		110	
tLA	Address Hold Time after ALE		100		50		100		
tLL	ALE Width		140		80		140		
tLCK	ALE Low During CLK High		100		50		100		
tLC	Trailing Edge of ALE to Leading Edge of Control		130		60		130		
tAFR	Address Float after Leading Edge of $\overline{READ}$ ($\overline{INTA}$)			0		0		0	
tAD	Valid Address to Valid Data In			575		350		575	
tRD	$\overline{READ}$ (or $\overline{INTA}$) to Valid Data			300		150		300	
tRDH	Data Hold Time after $\overline{READ}$ ($\overline{INTA}$) (Note 7)		0		0		0		
tRAE	Trailing Edge of $\overline{READ}$ to Re-Enabling of Address		150		90		150		
tCA	Address (A8-A15) Valid after Control		120		60		120		
tDW	Data Valid to Trailing Edge of $\overline{WRITE}$		420		230		420		
tWD	Data Valid after Trailing Edge of $\overline{WRITE}$		100		60		100		
tCC	Width of Control Low ($\overline{RD}$, $\overline{WR}$, $\overline{INTA}$)		400		230		400		
tCL	Trailing Edge of Control to Leading Edge of ALE		50		25		50		
tARY	READY Valid from Address Valid			220		100		220	
tRYS	READY Setup Time to Leading Edge of CLK		110		100		110		
tRYH	READY Hold Time		0		0		0		
tHACK	HLDA Valid to Trailing Edge of CLK		110		40		110		
tHABF	Bus Float after HLDA			210		150		210	
tHABE	HLDA to Bus Enable			210		150		210	
tLDR	ALE to Valid Data In			460		270		460	
tRV	Control Trailing Edge to Leading Edge of Next Control		400		220		400		
tAC	A8-A15 Valid to Leading Edge of Control (Note 1)		270		115		270		
tHDS	HOLD Setup Time to Trailing Edge of CLK		170		120		170		
tHDH	HOLD Hold Time		0		0		0		
tINS	INTR Setup Time to Falling Edge of CLK, also RST and TRAP		160		150		160		
tINH	INTR Hold Time		0		0		0		

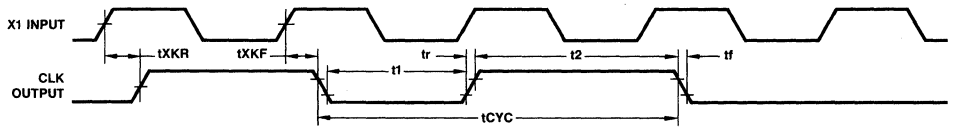
Notes:
1. A8-A15 Address Specs apply to IO/$\overline{M}$, S0 and S1. Except A8-A15 are undefined during T4-T6 of OF cycle whereas IO/$\overline{M}$, S0 and S1 are stable.
2. Test Conditions: tCYC = 320ns (Am8085A)/200ns (Am8085A-2); CL = 150pF.
3. For all output timing where CL = 150pF use the following correction factors.
   25pF $\leq$ CL $<$ 150pF: $-$.10ns/pF
   150pF $<$ CL $\leq$ 300pF: +.30ns/pF
4. Output timings are measured with purely capacitive load.
5. All timings are measured at output voltage VL = 0.8V, VH = 2.0V and 1.5V with 20ns rise and fall time on inputs.
6. To calculate timing specifications at other values of tCYC use the table on Page 9.
7. Data Hold Time is guaranteed under all loading conditions.
8. Loading equivalent to 50pF +1 TTL input.

BUS TIMING SPECIFICATION AS A TCYC DEPENDENT

Parameter	Description	Am8085A/Am8085A-2 Min	Am8085A/Am8085A-2 Max	Am9085ADM Min	Am9085ADM Max	Units
tAL	Address Valid before Trailing Edge of ALE	(1-2)T−45		(1/2)T−50		ns
tLA	Address Hold Time after ALE	(1/2)T−60		(1/2)T−50		ns
tLL	ALE Width	(1/2)T−20		(1/2)T−20		ns
tLCK	ALE Low During CLK High	(1/2)T−60		(1/2)T−50		ns
tLC	Trailing Edge of ALE to Leading Edge of Control	(1/2)T−30		(1/2)T−40		ns
tAD	Valid Address to Valid Data In		(5/2+N)T−225		(5/2+N)T−150	ns
tRD	READ (or INTA) to Valid Data		(3/2+N)T−180		(3/2+N)T−150	ns
tRAE	Trailing Edge of READ to Re-Enabling of Address	(1/2)T−10		(1/2)T−10		ns
tCA	Address (A8-A15) Valid after Control	(1/2)T−40		(1/2)T−40		ns
tDW	Data Valid to Trailing Edge of WRITE	(3/2+N)T−60		(3/2+N)T−70		ns
tWD	Data Valid after Trailing Edge of WRITE	(1/2)T−60		(1/2)T−40		ns
tCC	Width of Control LOW (RD, WR, INTA)	(3/2+N)T−80		(3/2+N)T−70		ns
tCL	Trailing Edge of Control to Leading Edge of ALE	(1/2)T−110		(1/2)T−75		ns
tARY	READY Valid from Address Valid		(3/2)T−260		(3/2)T−200	ns
tHACK	HLDA Valid to Trailing Edge of CLK	(1/2)T−50		(1/2)T−60		ns
tHABF	Bus Float after HLDA		(1/2)T+50		(1/2)T+50	ns
tHABE	HLDA to Bus Enable		(1/2)T+50		(1/2)T+50	ns
tAC	Address Valid to Leading Edge of Control	(2/2)T−50		(2/2)T−85		ns
t1	CLK Low Time	(1/2)T−80		(1/2)T−60		ns
t2	CLK High Time	(1/2)T−40		(1/2)T−30		ns
tRV	Control Trailing Edge to Leading Edge of Next Control	(3/2)T−80		(3/2)T−80		ns
tLDR			(4/2)T−180		(4/2)T−130	ns

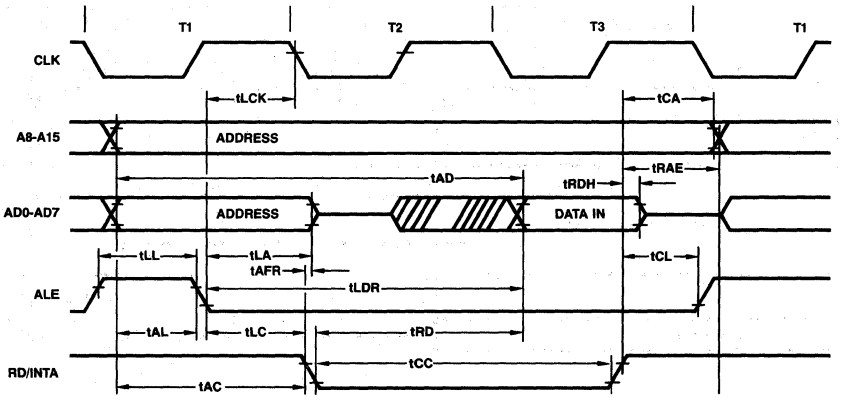
NOTE: N is equal to the total WAIT states.
T = tCYC.

CLOCK TIMING WAVEFORM



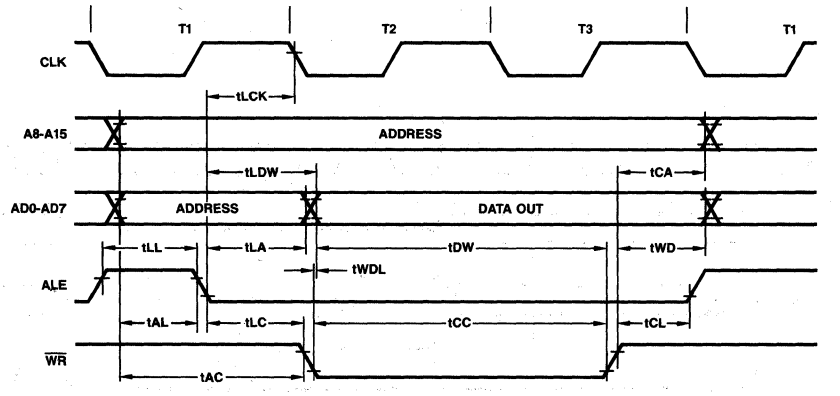
MOS-269

READ OPERATION



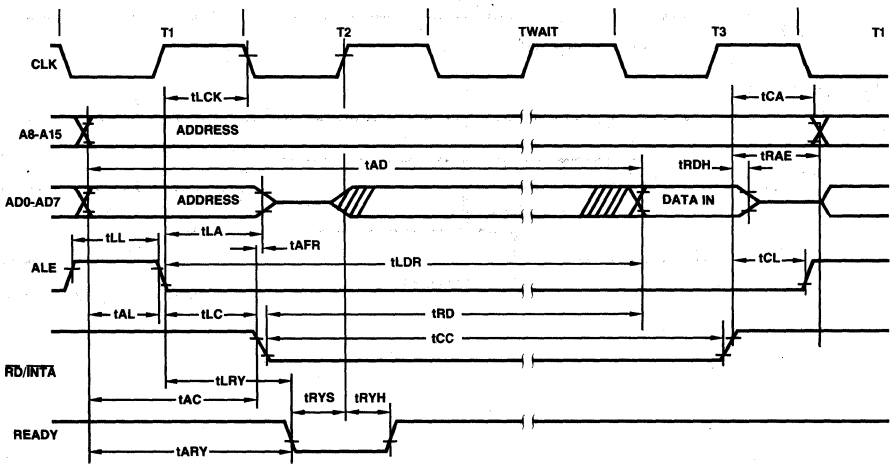
MOS-270

WRITE OPERATION



MOS-271

TYPICAL READ OPERATION WITH WAIT CYCLE



Same READY timing applies to WRITE operation.

MOS-272

Figure 6. Am8085A/Am8085A-2 Bus Timing

HOLD OPERATION

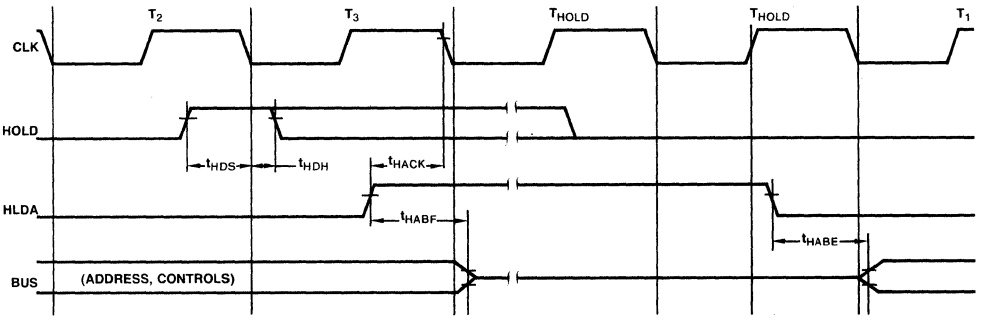
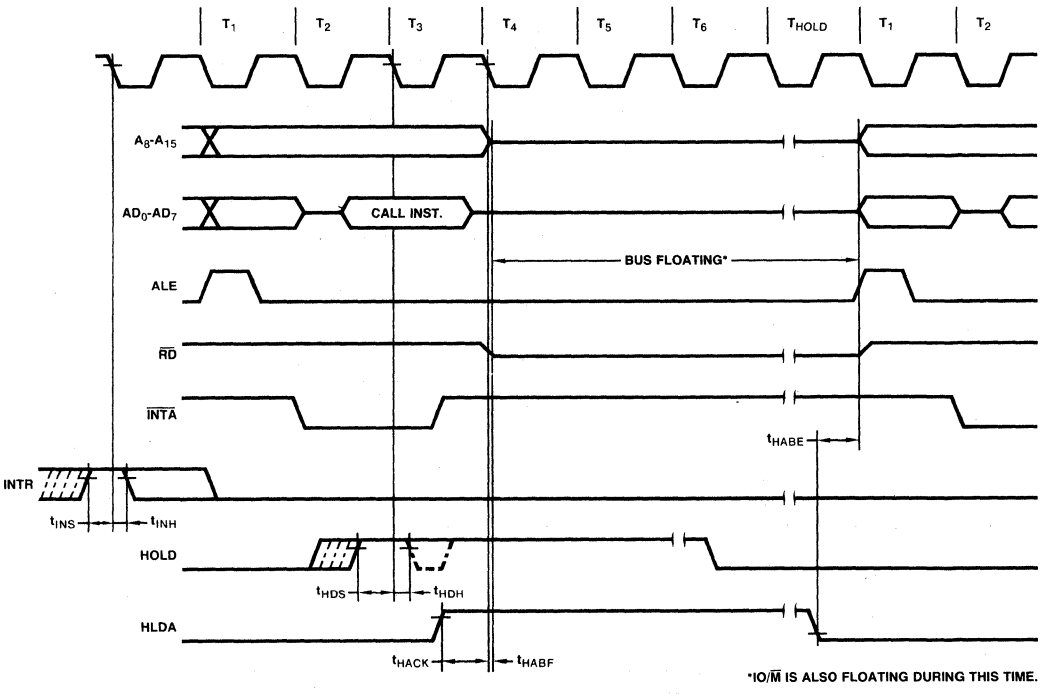


Figure 7. Am8085A Hold Timing.

MOS-130



MOS-131

Figure 8. Am8085A Interrupt and Hold Timing.

INSTRUCTION SET SUMMARY

Mnemonic*	Description	Instruction Code (Note 1) D7	D6	D5	D4	D3	D2	D1	D0	Clock Cycles (Note 2)
MOVE, LOAD AND STORE										
MOVr1r2	Move register to register	0	1	D	D	D	S	S	S	4
MOV Mr	Move register to memory	0	1	1	1	0	S	S	S	7
MOV rM	Move memory to register	0	1	D	D	D	1	1	0	7
MVI r	Move immediate register	0	0	D	D	D	1	1	0	7
MVI M	Move immediate memory	0	0	1	1	0	1	1	0	10
LXI B	Load immediate register Pair B & C	0	0	0	0	0	0	0	1	10
LXI D	Load immediate register Pair D & E	0	0	0	1	0	0	0	1	10
LXI H	Load immediate register Pair H & L	0	0	1	0	0	0	0	1	10
LXI SP	Load immediate stack pointer	0	0	1	1	0	0	0	1	10
STAX B	Store A indirect	0	0	0	0	0	0	1	0	7
STAX D	Store A indirect	0	0	0	1	0	0	1	0	7
LDAX B	Load A indirect	0	0	0	0	1	0	1	0	7
LDAX D	Load A indirect	0	0	0	1	1	0	1	0	7
STA	Store A direct	0	0	1	1	0	0	1	0	13
LDA	Load A direct	0	0	1	1	1	0	1	0	13
SHLD	Store H & L direct	0	0	1	0	0	0	1	0	16
LHLD	Load H & L direct	0	0	1	0	1	0	1	0	16
XCHG	Exchange D & E, H & L Registers	1	1	1	0	1	0	1	1	4
STACK OPS										
PUSH B	Push register Pair B & C on stack	1	1	0	0	0	1	0	1	12
PUSH D	Push register Pair D & E on stack	1	1	0	1	0	1	0	1	12
PUSH H	Push register Pair H & L on stack	1	1	1	0	0	1	0	1	12
PUSH PSW	Push A and Flags on stack	1	1	1	1	0	1	0	1	12
POP B	Pop register Pair B & C off stack	1	1	0	0	0	0	0	1	10
POP D	Pop register Pair D & E off stack	1	1	0	1	0	0	0	1	10
POP H	Pop register Pair H & L off stack	1	1	1	0	0	0	0	1	10
POP PSW	Pop A and Flags off stack	1	1	1	1	0	0	0	1	10
XTHL	Exchange top of stack H & L	1	1	1	0	0	0	1	1	16
SPHL	H & L to stack pointer	1	1	1	1	1	0	0	1	6
JUMP										
JMP	Jump unconditional	1	1	0	0	0	0	1	1	10
JC	Jump on carry	1	1	0	1	1	0	1	0	7/10
JNC	Jump on no carry	1	1	0	1	0	0	1	0	7/10
JZ	Jump on zero	1	1	0	0	1	0	1	0	7/10
JNZ	Jump on no zero	1	1	0	0	0	0	1	0	7/10
JP	Jump on positive	1	1	1	1	0	0	1	0	7/10
JM	Jump on minus	1	1	1	1	1	0	1	0	7/10
JPE	Jump on parity even	1	1	1	0	1	0	1	0	7/10
JPO	Jump on parity odd	1	1	1	0	0	0	1	0	7/10
PCHL	H & L to program counter	1	1	1	0	1	0	0	1	6
CALL										
CALL	Call unconditional	1	1	0	0	1	1	0	1	18
CC	Call on carry	1	1	0	1	1	1	0	0	9/18
CNC	Call on no carry	1	1	0	1	0	1	0	0	9/18
CZ	Call on zero	1	1	0	0	1	1	0	0	9/18
CNZ	Call on no zero	1	1	0	0	0	1	0	0	9/18
CP	Call on positive	1	1	1	1	0	1	0	0	9/18
CM	Call on minus	1	1	1	1	1	1	0	0	9/18
CPE	Call on parity even	1	1	1	0	1	1	0	0	9/18
CPO	Call on parity odd	1	1	1	0	0	1	0	0	9/18
RETURN										
RET	Return	1	1	0	0	1	0	0	1	10
RC	Return on carry	1	1	0	1	1	0	0	0	6/12
RNC	Return on no carry	1	1	0	1	0	0	0	0	6/12
RZ	Return on zero	1	1	0	0	1	0	0	0	6/12
RNZ	Return on no zero	1	1	0	0	0	0	0	0	6/12
RP	Return on positive	1	1	1	1	0	0	0	0	6/12
RM	Return on minus	1	1	1	1	1	0	0	0	6/12
RPE	Return on parity even	1	1	1	0	1	0	0	0	6/12
RPO	Return on parity odd	1	1	1	0	0	0	0	0	6/12
RESTART										
RST	Restart	1	1	A	A	A	1	1	1	12
INPUT/OUTPUT										
IN	Input	1	1	0	1	1	0	1	1	10
OUT	Output	1	1	0	1	0	0	1	1	10

INSTRUCTION SET SUMMARY (Cont.)

Mnemonic*	Description	Instruction Code (Note 1) D7	D6	D5	D4	D3	D2	D1	D0	Clock Cycles (Note 2)
INCREMENT AND DECREMENT										
INR r	Increment register	0	0	D	D	D	1	0	0	4
DCR r	Decrement register	0	0	D	D	D	1	0	1	4
INR M	Increment memory	0	0	1	1	0	1	0	0	10
DCR M	Decrement memory	0	0	1	1	0	1	0	1	10
INX B	Increment B & C registers	0	0	0	0	0	0	1	1	6
INX D	Increment D & E registers	0	0	0	1	0	0	1	1	6
INX H	Increment H & L registers	0	0	1	0	0	0	1	1	6
INX SP	Increment stack pointer	0	0	1	1	0	0	1	1	6
DCX B	Decrement B & C	0	0	0	0	1	0	1	1	6
DCX D	Decrement D & E	0	0	0	1	1	0	1	1	6
DCX H	Decrement H & L	0	0	1	0	1	0	1	1	6
DCX SP	Decrement stack pointer	0	0	1	1	1	0	1	1	6
ADD										
ADD r	Add register to A	1	0	0	0	0	S	S	S	4
ADC r	Add register to A with carry	1	0	0	0	1	S	S	S	4
ADD M	Add memory to A	1	0	0	0	0	1	1	0	7
ADC M	Add memory to A with carry	1	0	0	0	1	1	1	0	7
ADI	Add immediate to A	1	1	0	0	0	1	1	0	7
ACI	Add immediate to A with carry	1	1	0	0	1	1	1	0	7
DAD B	Add B & C to H & L	0	0	0	0	1	0	0	1	10
DAD D	Add D & E to H & L	0	0	0	1	1	0	0	1	10
DAD H	Add H & L to H & L	0	0	1	0	1	0	0	1	10
DAD SP	Add stack pointer to H & L	0	0	1	1	1	0	0	1	10
SUBTRACT										
SUB r	Subtract register from A	1	0	0	1	0	S	S	S	4
SBB r	Subtract register from A with borrow	1	0	0	1	1	S	S	S	4
SUB M	Subtract memory from A	1	0	0	1	0	1	1	0	7
SBB M	Subtract memory from A with borrow	1	0	0	1	1	1	1	0	7
SUI	Subtract immediate from A	1	1	0	1	0	1	1	0	7
SBI	Subtract immediate from A with borrow	1	1	0	1	1	1	1	0	7
LOGICAL										
ANA r	And register with A	1	0	1	0	0	S	S	S	4
XRA r	Exclusive Or register with A	1	0	1	0	1	S	S	S	4
ORA r	Or register with A	1	0	1	1	0	S	S	S	4
CMP r	Compare register with A	1	0	1	1	1	S	S	S	4
ANA M	And memory with A	1	0	1	0	0	1	1	0	7
XRA M	Exclusive Or memory with A	1	0	1	0	1	1	1	0	7
ORA M	Or memory with A	1	0	1	1	0	1	1	0	7
CMP M	Compare memory with A	1	0	1	1	1	1	1	0	7
ANI	And immediate with A	1	1	1	0	0	1	1	0	7
XRI	Exclusive Or immediate with A	1	1	1	0	1	1	1	0	7
ORI	Or immediate with A	1	1	1	1	0	1	1	0	7
CPI	Compare immediate with A	1	1	1	1	1	1	1	0	7
ROTATE										
RLC	Rotate A left	0	0	0	0	0	1	1	1	4
RRC	Rotate A right	0	0	0	0	1	1	1	1	4
RAL	Rotate A left through carry	0	0	0	1	0	1	1	1	4
RAR	Rotate A right through carry	0	0	0	1	1	1	1	1	4
SPECIALS										
CMA	Complement A	0	0	1	0	1	1	1	1	4
STC	Set carry	0	0	1	1	0	1	1	1	4
CMC	Complement carry	0	0	1	1	1	1	1	1	4
DAA	Decimal adjust A	0	0	1	0	0	1	1	1	4
CONTROL										
EI	Enable Interrupts	1	1	1	1	1	0	1	1	4
DI	Disable Interrupts	1	1	1	1	0	0	1	1	4
NOP	No operation	0	0	0	0	0	0	0	0	4
HLT	Halt	0	1	1	1	0	1	1	0	5
NEW Am8085A INSTRUCTIONS										
RIM	Read Interrupt Mask	0	0	1	0	0	0	0	0	4
SIM	Set Interrupt Mask	0	0	1	1	0	0	0	0	4

Notes: 1. DOD or SSS: 8=000, C=001, D=010, E=011, H=100, L=101, Memory=110, A=111.
2. Two possible cycle times (6/12) indicate instruction cycles dependent on condition flags.

*All mnemonics copyright © Intel Corporation 1977

AmZ8001

16-Bit Microprocessor

PRELIMINARY DATA

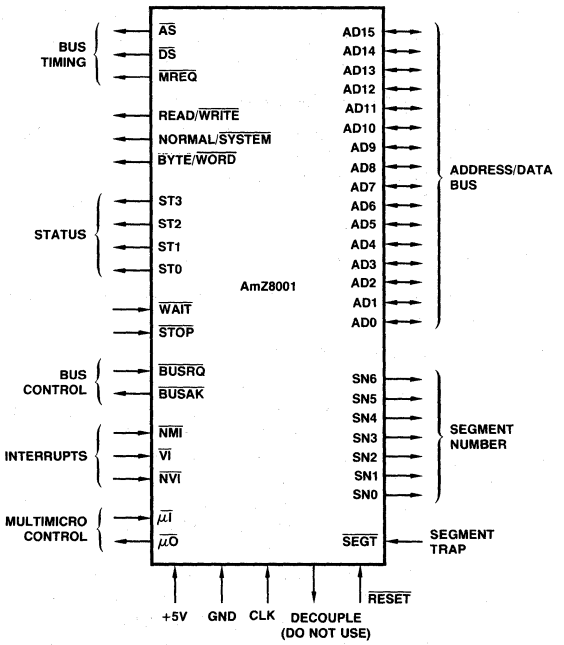
DISTINCTIVE CHARACTERISTICS

- Sixteen general purpose registers
- Direct addressing up to 8MB segmented memory
- Software compatible with AmZ8002 microprocessor
- Powerful instructions with flexible addressing modes
- Privileged/Non-Privileged mode of operation
- Sophisticated interrupt structure
- On-chip memory refresh facility
- TTL compatible inputs and outputs
- Single phase clock
- Single +5V power supply
- 48-pin package

GENERAL DESCRIPTION

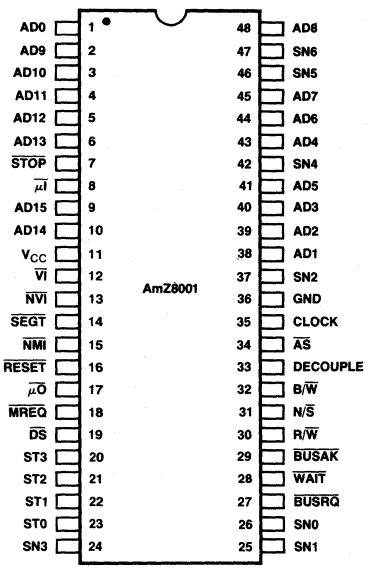
The AmZ8001 is a general-purpose 16-bit CPU belonging to the AmZ8000 family of microprocessors. Its architecture is centered around sixteen 16-bit general registers. The CPU deals with 23-bit address spaces and hence can address directly 8MB of memory. The 23-bit address consists of two components: 7-bit segment number and 16-bit offset. Facilities are provided to maintain three distinct address spaces – code, data and stack. The AmZ8001 implements a powerful instruction set with flexible addressing modes. These instructions operate on several data types – bit, byte, word (16-bit), long word (32-bit), byte string and word string. The CPU can execute instructions in one of two modes – System and Normal. Sometimes these modes are also known as Privileged and Non-Privileged, respectively. The CPU also contains an on-chip memory refresh facility. The AmZ8001 is software compatible with the AmZ8002 microprocessor. The AmZ8001 is fabricated using silicon-gate N-MOS technology and is packaged in a 48-pin DIP. The AmZ8001 requires a single +5 power supply and a single phase clock for its operation.

LOGIC SYMBOL



MOS-240

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

MOS-241

ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency 4MHz
Hermetic DIP	$0°C \leq T_A \leq 70°C$	AmZ8001DC

INTERFACE SIGNAL DESCRIPTION

V_{CC}: +5V Power Supply

V_{SS}: Ground

AD0-AD15: Address/Data Bus (Bidirectional, 3-State)

This 16-bit multiplexed address/data bus is used for all I/O and memory transactions. HIGH on the bus corresponds to 1 and LOW corresponds to 0. AD0 is the least significant bit position with AD15 is most significant. The $\overline{AS}$ output and $\overline{DS}$ output will indicate whether the bus is used for address offset or data. The status output lines ST0-ST3 will indicate the type of transaction; memory or I/O.

$\overline{AS}$: Address Strobe (Output, 3-State)

LOW on this output indicates that the AD0-AD15 bus contains address information. The address information is stable by the time of the LOW-to-HIGH transition of the $\overline{AS}$ output (see timing diagrams). The status outputs ST0-ST3 indicate whether the bus contains a memory address or I/O address.

$\overline{DS}$: Data Strobe (Output, 3-State)

LOW on this output indicates that the AD0-AD15 bus is being used for data transfer. The R/$\overline{W}$ output indicates the direction of data transfer – read (or in) means data into the CPU and write (or out) means data from the CPU. During a read operation, data can be gated on to the bus when $\overline{DS}$ goes LOW. A LOW-to-HIGH transition on the $\overline{DS}$ output indicates that the CPU has accepted the data (see timing diagram). During a write operation, LOW on the $\overline{DS}$ output indicates that data is setup on the bus. Data will be removed sometime after the LOW-to-HIGH transition of the $\overline{DS}$ output (see timing diagram).

R/$\overline{W}$: Read/Write (Output, 3-State)

This output indicates the direction of data flow on the AD0-AD15 bus. HIGH indicates a read operation, i.e., data into the CPU and LOW indicates a write operation, i.e., data from the CPU. This output is activated at the same time as $\overline{AS}$ going LOW and remains stable for the duration of the whole transaction (see timing diagram).

B/$\overline{W}$: Byte/Word (Output, 3-State)

This output indicates the type of data transferred on the AD0-AD15 bus. HIGH indicates byte (8-bit) and LOW indicates word (16-bit) transfer. This output is activated at the same stage as $\overline{AS}$ going LOW and remains valid for the duration of the whole transaction (see timing diagram). The address generated by the CPU is always a byte address. However, the memory is organized as 16-bit words. All instructions and word operands are word aligned and are addressed by even addresses. Thus, for all word transactions with the memory the least significant address bit will be zero. When addressing the memory for byte transactions, the least significant address bit determines which byte of the memory word is needed; even address specifies the most significant byte and odd address specifies the least significant byte. In the case of I/O transactions, the address information on the AD0-AD15 bus refers to an I/O port and B/$\overline{W}$ determines whether a data word or data byte will be transacted. During I/O byte transactions, the least significant address bit A0 determines which half of the AD0-AD15 bus will be used for the I/O transactions. The ST0-ST3 outputs will indicate whether the current transaction is for memory, normal I/O or special I/O.

ST0-ST3: Status (Outputs, 3-State)

These four outputs contain information regarding the current transaction in a coded form. The status line codes are shown in the following table:

ST3	ST2	ST1	ST0	
L	L	L	L	Internal Operation
L	L	L	H	Memory Refresh
L	L	H	L	Normal I/O Transaction
L	L	H	H	Special I/O Transaction
L	H	L	L	Segment Trap Acknowledge
L	H	L	H	Non-Maskable Interrupt Acknowledge
L	H	H	L	Non-Vectored Interrupt Acknowledge
L	H	H	H	Vectored Interrupt Acknowledge
H	L	L	L	Memory Transaction for Operand
H	L	L	H	Memory Transaction for Stack
H	L	H	L	Reserved
H	L	H	H	Reserved
H	H	L	L	Memory Transaction for Instruction Fetch (Subsequent Word)
H	H	L	H	Memory Transaction for Instruction Fetch (First Word)
H	H	H	L	Reserved
H	H	H	H	Reserved

$\overline{WAIT}$: Wait (Input)

LOW on this input indicates to the CPU that memory or I/O is not ready for the data transfer and hence the current transaction should be stretched. The $\overline{WAIT}$ input is sampled by the CPU at certain instances during the transaction (see timing diagram). If $\overline{WAIT}$ input is LOW at these instances, the CPU will go into wait state to prolong the transaction. The wait state will repeat until the $\overline{WAIT}$ input is HIGH at the sampling instant.

N/$\overline{S}$: Normal/System Mode (Output, 3-State)

HIGH on this output indicates that the CPU is operating in Normal Mode and LOW indicates operation in System Mode. This output is derived from the Flag Control Word (FCW) register. The FCW register is described under the program status information section of this document.

$\overline{MREQ}$: Memory Request (Output, 3-State)

LOW on this output indicates that a CPU transaction with memory is taking place.

$\overline{BUSRQ}$: Bus Request (Input)

LOW on this input indicates to the CPU that another device (such as DMA) is requesting to take control of the bus. The $\overline{BUSRQ}$ input can be driven LOW anytime. The CPU synchronizes this input internally. The CPU responds by activating $\overline{BUSAK}$ output LOW to indicate that the bus has been relinquished. Relinquishing the bus means that the AD0-AD15, $\overline{AS}$, $\overline{DS}$, B/$\overline{W}$, R/$\overline{W}$, N/$\overline{S}$, ST0-ST3, SN0-SN6 and $\overline{MREQ}$ outputs will be in the high impedance state. The requesting device should control these lines in an identical fashion to the CPU to accomplish transactions. The $\overline{BUSRQ}$ input must remain LOW as long as needed to perform all the transactions and the CPU will keep the $\overline{BUSAK}$ output LOW. After completing the transactions, the device must disable the AD0-AD15, $\overline{AS}$, $\overline{DS}$, B/$\overline{W}$, R/$\overline{W}$, N/$\overline{S}$, ST0-ST3, SN0-SN6 and $\overline{MREQ}$ into the high impedance state and stop driving the $\overline{BUSRQ}$ input LOW. The CPU will make $\overline{BUSAK}$ output HIGH sometime later and take back the bus control.

$\overline{BUSAK}$: Bus Acknowledge (Output)

LOW on this output indicates that the CPU has relinquished the bus in response to a bus request.

$\overline{NMI}$: Non-Maskable Interrupt (Input)

HIGH to LOW transition on this input constitutes non-maskable interrupt request. The CPU will respond with the Non-maskable Interrupt Acknowledge on the ST0-ST3 outputs and will enter an interrupt sequence. The transition on the $\overline{NMI}$ can occur anytime. Of the three kinds of interrupts available, the non-maskable interrupt has the highest priority.

$\overline{VI}$: Vectored Interrupt (Input)

LOW on this input constitutes vectored interrupt request. Vectored interrupt is next lower to the non-maskable interrupt in priority. The VIE bit in the Flag and Control Word register must be 1 for the vectored interrupt to be honored. The CPU will respond with Vectored Interrupt Acknowledge code on the ST0-ST3 outputs and will begin the interrupt sequence. The $\overline{VI}$ input can be driven LOW anytime and should be held LOW until acknowledged.

$\overline{NVI}$: Non-Vectored Interrupt (Input)

LOW on this input constitutes non-vectored interrupt request. Non-vectored has the lowest priority of the three types of interrupts. The NVIE bit in the Flag and Control Word register must be 1 for this request to be honored. The CPU will respond with Non-Vectored Interrupt Acknowledge code on the ST0-ST3 outputs and will begin the interrupt sequence. The $\overline{NVI}$ input can be driven LOW anytime and should be held LOW until acknowledged.

$\overline{\mu I}$: Micro-In (Input)

This input participates in the resource request daisy chain. See the section on multimicroprocessor support facilities in this document.

$\overline{\mu O}$: Micro-Out (Output)

This output participates in the resource request daisy chain. See the section on multimicroprocessor support facilities in this document.

$\overline{RESET}$: Reset (Input)

LOW on this input initiates a reset sequence in the CPU. See the section on Initialization for details on reset sequence.

CLK: Clock (Input)

All CPU operations are controlled from the signal fed into this input. See DC characteristics for clock voltage level requirements.

DECOUPLE: (Output)

Output from the on-chip substrate bias generator. Do not use.

$\overline{STOP}$: Stop (Input)

This active LOW input facilitates one instruction at a time operation. See the section on single stepping.

SN0-SN6: Segment Number (Outputs, 3-State)

These seven outputs contain the segment number part of a memory address. A HIGH on the output corresponds to 1 and a LOW corresponds to 0. SN0 is the least significant bit position and SN6 is the most significant bit position.

$\overline{SEGT}$: Segment Trap (Input)

LOW on this input constitutes a segment trap request. If the line is driven LOW, the CPU will respond with the Segment Trap Acknowledge code on the Status lines, and commence a trap sequence. The $\overline{SEGT}$ input may be driven LOW at any time and is customarily held LOW until acknowledged. This input has priority over the interrupts.

PROCESSOR ORGANIZATION

The following is a brief discussion of the AmZ8001 CPU. For detailed information, see the AmZ8001/AmZ8002 Processor Instruction Set Manual (Publication No. AM-PUB086).

GENERAL PURPOSE REGISTERS

The CPU is organized around sixteen 16-bit general purpose registers R0 through R15 as shown in Figure 1. For byte operations, the first eight registers (R0 through R7) can also be addressed as sixteen 8-bit registers designated as RL0, RH0 and so on to RL7 and RH7. The sixteen registers can also be grouped in pairs RR0, RR2 and so on to RR14 to form eight long word (32-bit) registers. Similarly, the sixteen registers can be grouped in quadruples RQ0, RQ4, RQ8 and RQ12 to form four 64-bit registers.

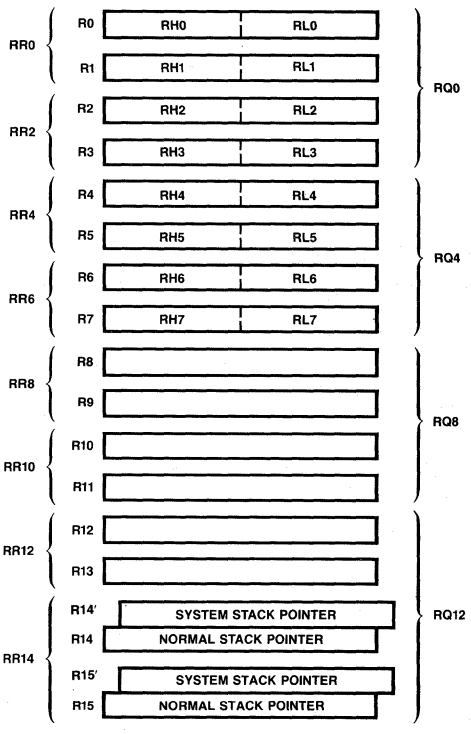


Figure 1. AmZ8001 General Registers.

MOS-242

STACK POINTER

The AmZ8001 architecture allows stacks to be maintained in memory. Any general-purpose register pair except RR0 can be used as a stack pointer in stack manipulating instructions such as PUSH and POP. The designated register pair holds a 23-bit segmented address. Certain instructions (such as subroutine call and return) make implicit use of the register pair RR14 as the stack pointer. Two implicit stacks are allowed – normal stack using RR14 as the stack pointer, and system stack using RR14' as the system stack pointer (see Figure 1). If the CPU is operating in the Normal Mode, RR14 is active, and if the CPU is in System Mode, RR14' will be used instead of RR14. The implied stack pointer is a part of the general registers and hence can be manipulated using the instructions available for register operations.

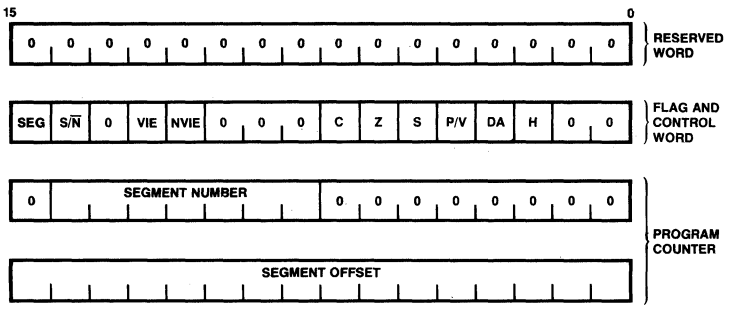


Figure 2. AmZ8001 Processor Status.

MOS-243

PROCESSOR STATUS

The CPU status consists of the 16-bit flag and control word (FCW) register, and the 23-bit program counter (see Figure 2). A reserved word is also included for future expansion. The following is a brief description of the FCW bits.

SEG: Segmented/Non-Segmented Bit. Indicates whether the AmZ8001 is running in segmented or non-segmented mode. 1 indicates segmented, 0 indicates non-segmented. See the section on non-segmented mode, elsewhere in this document.

S/$\overline{N}$: System/Normal – 1 indicates System Mode and 0 indicates Normal Mode.

VIE: Vectored Interrupt Enable – 1 indicates that Vectored Interrupt requests will be honored.

NVIE: Non-Vectored Interrupt Enable – 1 indicates that Non-vectored interrupt requests will be honored.

C: Carry – 1 indicates that a carry has occurred from the most significant bit position when performing arithmetic operations.

Z: Zero – 1 indicates that the result of an operation is zero.

S: Sign – 1 indicates that the result of an operation is negative i.e., most significant bit is one.

P/V: Parity/Overflow – 1 indicates that there was an overflow during arithmetic operations. For byte logical operations this bit indicates parity of the result.

DA: Decimal Adjust – Records byte arithmetic operations.

H: Half Carry – 1 indicates that there was a carry from the most significant bit of the lower digit during byte arithmetic.

DATA TYPES

The AmZ8001 instructions operate on bits, digits (4 bits), bytes (8 bits), words (16 bits), long words (32 bits), byte strings and word strings type operands. Bits can be set, reset or tested. Digits are used to facilitate BCD arithmetic operations. Bytes are used for characters and small integers. Words are used for integer values and addresses while long words are used for large integer values and addresses. All operands except strings can reside either in memory or general registers. Strings can reside in memory only.

INTERRUPT AND TRAP STRUCTURE

Interrupt is defined as an external asynchronous event requiring program interruption. For example, interruption is caused by a peripheral needing service. Traps are synchronous events resulting from execution of certain instructions under some defined circumstances. Both interrupts and traps are handled in a similar manner by the AmZ8001.

The AmZ8001 supports three types of interrupts in order of descending priority – non-maskable, vectored and non-vectored. The vectored and non-vectored interrupts can be disabled by appropriate control bits in the FCW. The AmZ8001 has four traps – system call, segment trap, unimplemented opcode and privileged instruction. The traps have higher priority than interrupts.

When an interrupt or trap occurs, the current program status is automatically pushed on to the system stack. The program status consists of processor status (i.e., PC and FCW) plus a 16-bit identifier. The identifier contains the reason, source and other coded information relating to the interrupt or trap.

After saving the current program status, the new processor status is automatically loaded from the new program status area located in the memory. This area is designated by the New Program Status Area Pointer (NPSAP) register. See AM-PUB086 publication for further details.

SEGMENTED ADDRESSING

The AmZ8001 can directly address up to 8MB of memory space, using a 23-bit segmented address. The memory space is divided up into 128 segments, each up to 64KB in size. The upper seven bits of address designate the segment number, and are available on the SN0-SN6 outputs during a memory transaction. See the section on memory transactions for details.

The lower sixteen bits of address designate an offset within the segment, relative to the start of the segment, and are available on AD0-AD15 during part of the memory transaction. See the section on memory transactions for details.

The segmented address may be stored as a long word in memory, or in a register pair. The segment number and offset can be manipulated separately or together, by suitable use of the instruction set.

When the segmented address is contained in code space, a short offset format may be adopted. The segmented address is stored as one word, seven bits of segment number and eight bits of offset. Figure 3 shows the format for segmented addresses.

ADDRESSING MODES

Information contained in the AmZ8001 instructions consists of the operation to be performed, the operand type and the location of the operands. Operand locations are designated by general register addresses, memory addresses or I/O addresses. The addressing mode of a given instruction defines the address space referenced and the method to compute the operand address. Addressing modes are explicitly specified or implied in an instruction. Figure 4 illustrates the eight explicit addressing modes: Register (R), Immediate (IM), Indirect Register (IR), Direct Address (DA), Indexed (X), Relative Address (RA), Base Address (BA) and Base Indexed (BX).

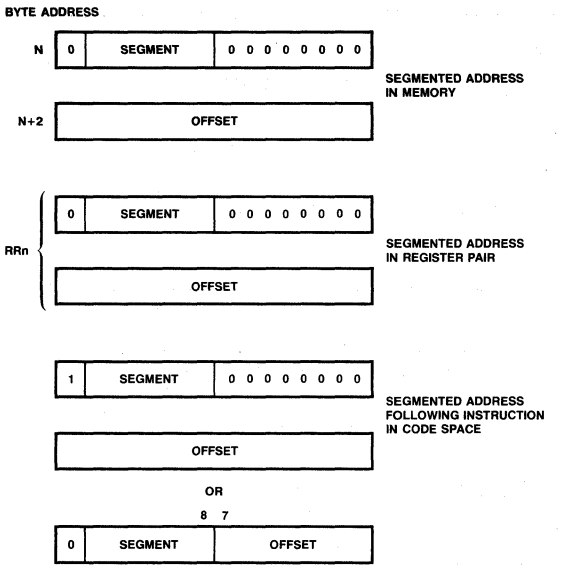


Figure 3. Segmented Address Formats.

MOS-244

When an effective segmented address is being computed according to the designated addressing mode, the segment number is not affected by any carry from the 16-bit offset.

NON-SEGMENTED MODE ON THE AmZ8001

The AmZ8001 can execute code designed to run on the non-segmented AmZ8002. This is achieved by changing the mode of execution of the AmZ8001 from segmented to non-segmented by writing a 0 to the SEG bit in the FCW. (See the section on processor status.) The change to non-segmented mode sets up a suitable environment for running non-segmented code. However, this environment only exists within the code segment that caused the change of mode from segmented to non-segmented.

SN0-SN6 will continue to indicate the code segment until a reset, interruption or return to segmented mode is encountered.

The effects of the non-segmented mode of operation on the AmZ8001 are described below.

a) The AmZ8001 will interpret instruction length as if it was a non-segmented AmZ8002.
b) The AmZ8001 will implement address computation in an identical manner to the AmZ8002.

Other CPU functions, such as interrupt and trap handling, reset and stack pointer manipulation are unaltered. These functions are characterized by the type of CPU, not by the state of the SEG bit in the FCW.

INPUT/OUTPUT

A set of I/O instructions are provided to accomplish byte or word transfers between the AmZ8001 and I/O devices. I/O devices are addressed using 16-bit I/O port addresses and I/O address space is not a part of the memory address space. Two types of I/O instructions are provided; each with its own 16-bit address space. I/O instructions include a comprehensive set of In, Out and Block transfers.

CPU TIMING

The AmZ8001 accomplishes instruction execution by stepping through a pre-determined sequence of machine cycles, such as memory read, memory write, etc. Each machine cycle requires between three and ten clock cycles. Bus Requests by DMA devices are granted at machine cycle boundaries. No machine cycle is longer than ten clock cycles; thus assuring fast response to a Bus Request (assuming no extra wait states). The start of a machine cycle is always marked by a LOW pulse on the $\overline{AS}$ output. The status output lines ST0-ST3 indicate the nature of the current cycle in a coded form.

STATUS LINE CODES

Status line coding was listed in the table shown under ST0-ST3 outputs in the Interface Signal Description. The following is a detailed description of the status codes.

Internal Operation:

This status code indicates that the AmZ8001 is going through a machine cycle for its internal operation. Figure 5 depicts an internal operation cycle. It consists of three clock periods identified as T1, T2 and T3. The $\overline{AS}$ output will be activated with a LOW pulse by the AmZ8001 to mark the start of a machine cycle. The ST0-ST3 will reflect the code for the internal operation. The $\overline{MREQ}$, $\overline{DS}$ and R/$\overline{W}$ outputs will be HIGH. The N/$\overline{S}$ and SN0-SN6 outputs will remain at the same level as in the previous machine cycle. The AmZ8001 will ignore the $\overline{WAIT}$ input during the internal operation cycle. The CPU will drive the AD0-AD15 bus with unspecified information during T1. However, the bus will go into high impedance during T2 and remain in that state for the remainder of the cycle. The B/$\overline{W}$ output is also activated by the CPU with unspecified information.

Memory Refresh:

This status code indicates that AmZ8001 is accessing the memory to refresh. The refresh cycle consists of three clock periods as depicted in Figure 6. The CPU will activate the $\overline{AS}$ output with a LOW pulse to mark the beginning of a machine cycle and ST0-ST3 outputs will reflect the refresh cycle code. The least significant 9 lines of the AD0-AD15 bus contain the refresh address. Because the memory is word organized, the AD0 will always be LOW. The most significant 7 bus lines are not specified. The $\overline{DS}$ output will remain HIGH for the entire cycle while R/$\overline{W}$, B/$\overline{W}$, SN0-SN6 and N/$\overline{S}$ outputs will remain at the same level as in the machine cycle prior to refresh. The AD0-AD15 bus will go into high impedance state during T2 period and remain there for the remainder of the cycle. The AmZ8001 will activate the $\overline{MREQ}$ output LOW during the refresh cycle. It should be noted that $\overline{WAIT}$ input is ignored by the CPU for refresh operations.

I/O Transactions:

There are two status line codes used for I/O transaction cycles. The AmZ8001 provides two separate I/O spaces and two types of instructions called Normal I/O and Special I/O. Each I/O space is addressed by a 16-bit address called port address. The timing for both types of I/O transactions is essentially identical. A typical I/O cycle consists of four clock periods T1, T2, TWA and T3 as shown in Figure 7. The TWA is the wait state; insertion of one wait state for an I/O cycle is always automatic. Additional wait cycles can be inserted by LOW on the $\overline{WAIT}$ input. The $\overline{WAIT}$ input is sampled during every TW state. If this input is LOW, one more wait state will be inserted. Insertion of wait states continues until $\overline{WAIT}$ input is HIGH. T3 state will follow the last wait state to complete the I/O cycle.

Mode	Operand Addressing			Operand Value
	In the Instruction	In a Register	In Memory	
Register		REGISTER ADDRESS → OPERAND		The content of the register.
Immediate	OPERAND			In the instruction
Indirect Register		REGISTER ADDRESS → ADDRESS	→ OPERAND	The content of the location whose address is in the register.
Direct Address			ADDRESS → OPERAND	The content of the location whose address is in the instruction.
Index		REGISTER ADDRESS → DISPLACEMENT BASE ADDRESS → ⊕	→ OPERAND	The content of the location whose address is the address in the instruction, offset by the content of the working register.
Relative Address		PC VALUE DISPLACEMENT → ⊕	→ OPERAND	The content of the location whose address is the content of the program counter, offset by the displacement in the instruction.
Base Address		REGISTER ADDRESS → BASE ADDRESS DISPLACEMENT → ⊕	→ OPERAND	The content of the location whose address is the address in the register, offset by the displacement in the instruction.
Base Index		REGISTER ADDRESS → BASE ADDRESS REGISTER ADDRESS → DISPLACEMENT → ⊕	→ OPERAND	The content of the location whose address is the address in the register, offset by the displacement in the register.

Figure 4. Addressing Modes.

MOS-245

During I/O cycles the ST0-ST3 outputs will reflect appropriate code depending on the type of instruction being executed (Normal I/O or Special I/O). $\overline{AS}$ output will be pulsed LOW to mark the beginning of the cycle. The CPU drives the AD0-AD15 bus with the 16-bit port address specified by the current instruction. The N/$\overline{S}$ output will be LOW indicating that CPU is operating in the system mode. It should be recalled that the N/$\overline{S}$ output is derived from the appropriate bit in the FCW register. All I/O instructions are privileged instructions and will be allowed to execute only if the FCW specifies system mode operation. The $\overline{MREQ}$ output will be HIGH. The AmZ8001 I/O instructions provide both word or byte transactions. The B/$\overline{W}$ output will be HIGH or LOW depending whether the instruction specifies a byte or word transfer. The SN0-SN6 output will remain at the same level as in the machine cycle prior to the I/O cycle.

Two kinds of I/O transfers should be considered: Data In means reading from the device and Data Out means writing into the device. For In operations, the R/$\overline{W}$ output will be HIGH. The AD0-AD15 bus will go into high impedance state during T2. During byte input instructions, the CPU reads either the even or odd half of the Data Bus dependent upon the port address. If the port address is even, the most significant half of the Data Bus is read. If the port address is odd, the least significant half of the Data Bus is read. During word input instructions, the CPU reads all 16 bits of the Data Bus. The AmZ8001 will drive the $\overline{DS}$

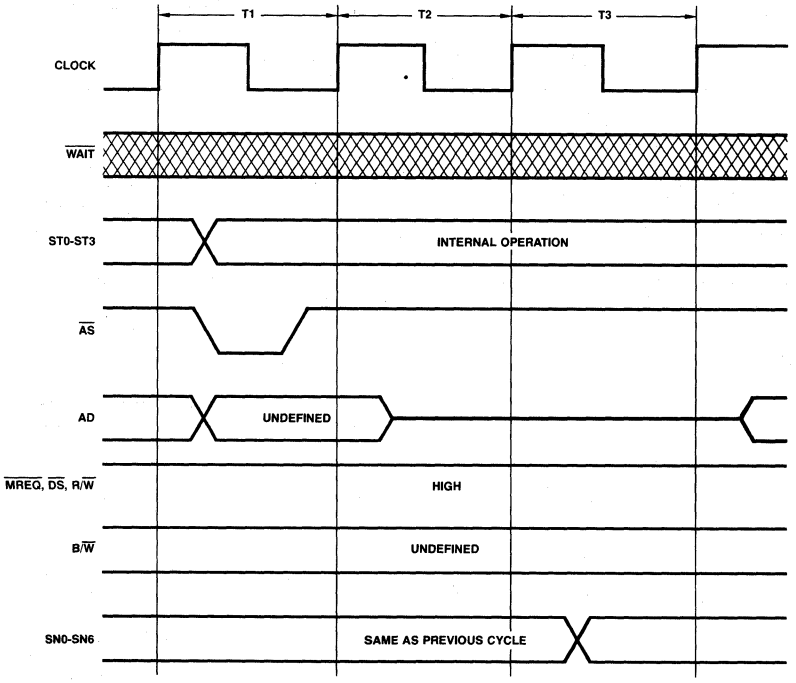


Figure 5. Internal Operation Cycle.

MOS-246

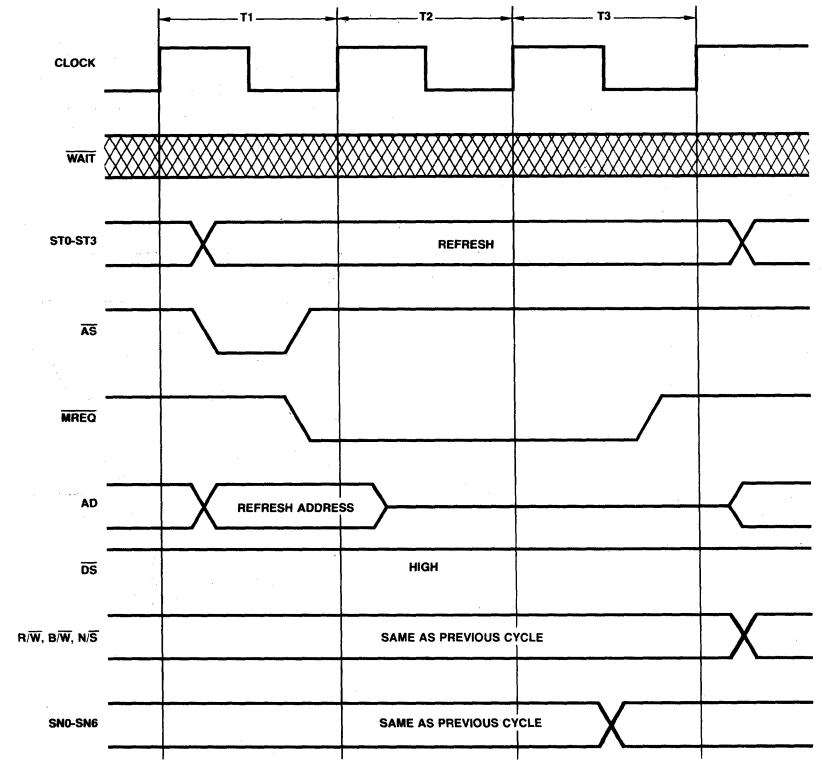


Figure 6. Refresh Cycle.

MOS-247

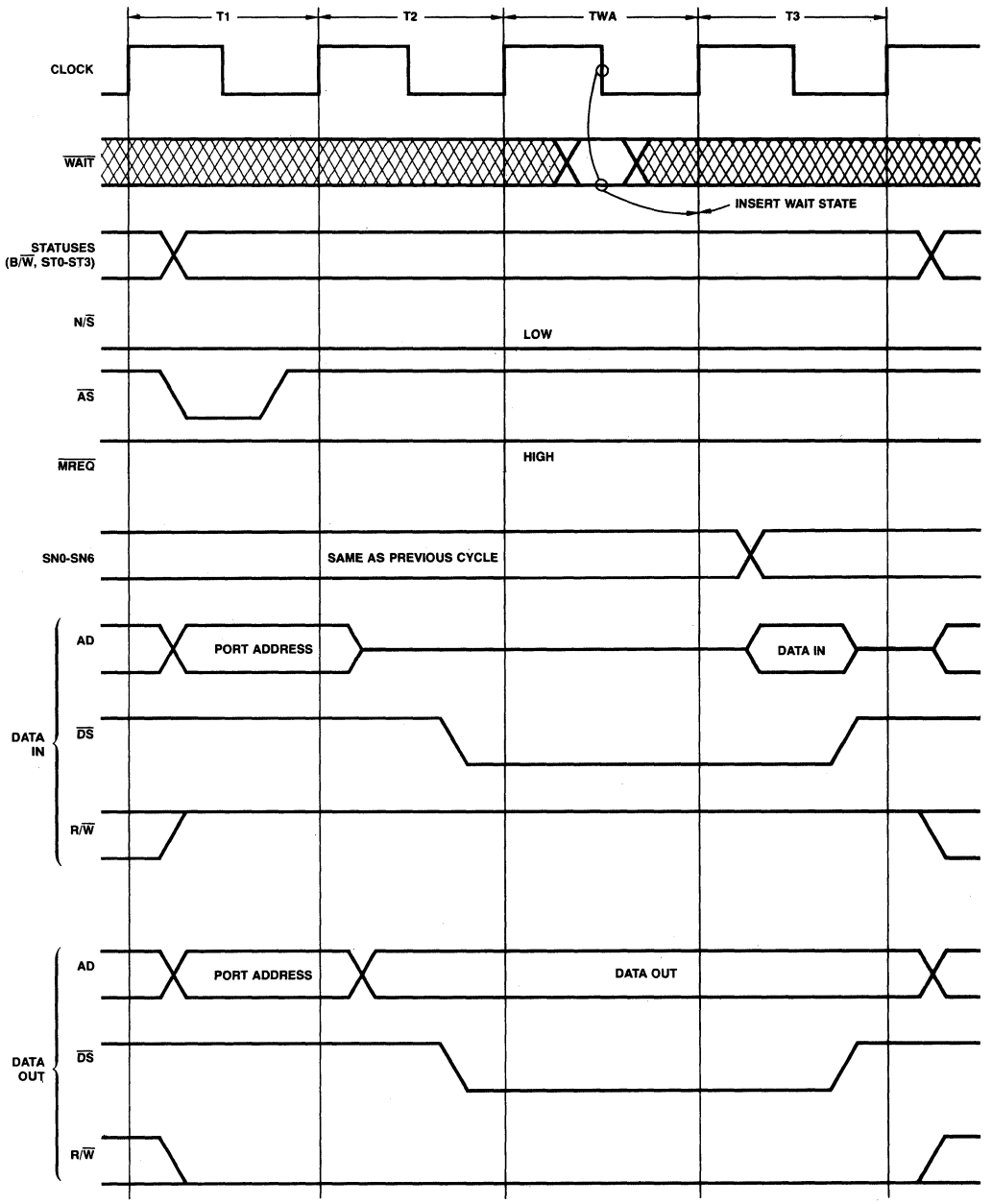


Figure 7. AmZ8001 I/O Cycle.

MOS-248

output LOW to signal to the device that data can be gated on to the bus. The CPU will accept the data during T3 and $\overline{DS}$ output will go HIGH signalling the end of an I/O transaction.

For Data Out, the R/$\overline{W}$ output will be LOW. The AmZ8001 will provide data on the AD0-AD15 bus and activates the $\overline{DS}$ output LOW during T2. During byte output instructions, the CPU duplicates the byte data onto both the high and low halves of the Data Bus and external logic, using A0, enables the appropriate byte port. During word output instructions the CPU outputs data onto all 16 bits of the Data Bus. The $\overline{DS}$ output goes HIGH during T3 and the cycle is complete.

Memory Transactions:

There are four status line codes that indicate a memory transaction:

a) Memory transaction to read or write an operand
b) Memory transaction to read from or write into the stack
c) Memory transaction to fetch the first word of an instruction (sometimes called IF1)
d) Memory transaction to fetch the subsequent word of an instruction (sometimes called IFN).

It can be appreciated that all the above transactions essentially fall into two categories: memory read and memory write. In the case of IF1 and IFN cycles, the memory will be read at the address supplied by the program counter. All AmZ8001 instructions are multiples of 16-bit words. Words are always addressed by an even address. Thus IF1 and IFN cycles involve performing a memory read for words. On the other hand, a memory transaction for operand and stack operation could be a read or write. Moreover, an operand could be a word or a byte. For stack operation involving the implied stack pointer the address will be supplied by the RR14 (or RR14'). For operand transactions, the memory address will come from several sources depending on the instruction and the addressing mode. Memory transaction cycle timing is shown in Figure 8. It typically consists of three clock periods T1, T2 and T3. Wait states (TW) can be inserted

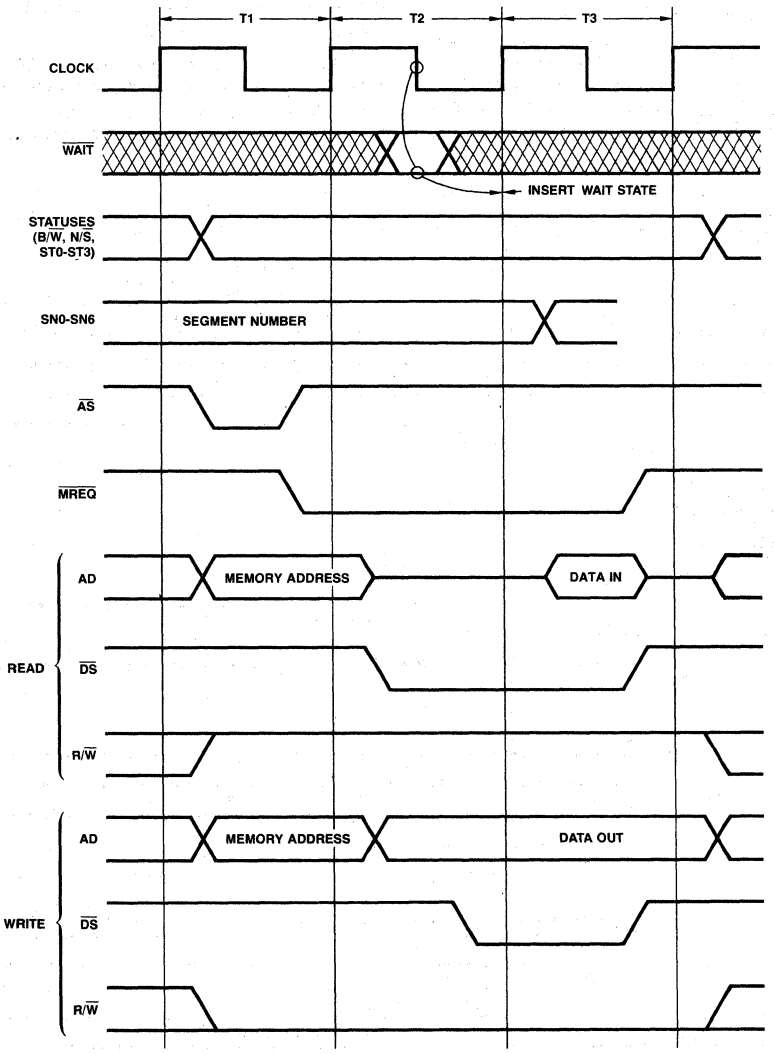


Figure 8. Memory Transactions.

MOS-249

between T2 and T3 by activating the $\overline{WAIT}$ input LOW. The $\overline{WAIT}$ input will be sampled during T2 and during every subsequent TW. The ST0-ST3 outputs will reflect the appropriate code for the current cycle early in T1 and the $\overline{AS}$ output will be pulsed LOW to mark the beginning of the cycle. The N/$\overline{S}$ output will indicate whether the normal or system address space will be used for the current cycle. As shown in the figure the $\overline{MREQ}$ output will go LOW during T1 to indicate a memory operation.

The segment number becomes valid on the segment lines one clock period before the start of the memory operation, and remains valid until the start of T3.

Consider a read operation first. The R/$\overline{W}$ output will be HIGH. The AmZ8001 will drive the AD0-AD15 with the appropriate address early in T1. During T2, the bus will go into high-impedance state and $\overline{DS}$ output will be activated LOW by the CPU. The data can be gated on to the bus when $\overline{DS}$ is LOW. During T1 the B/$\overline{W}$ will also be activated to indicate byte or word will be transacted. The AmZ8001 memory is word organized and words are addressed by even addresses. However, when addressing bytes, the memory address may be odd or even; an even address for most significant byte of a word and the next odd address for the least significant byte of that word. When reading a byte from the memory, the least significant address bit can be ignored and the whole word containing the desired byte is gated on to the bus. The CPU will pick the appropriate byte automatically. The AmZ8001 will drive the $\overline{DS}$ output HIGH indicating data acceptance.

Consider the write operation next. The R/$\overline{W}$ output will be LOW. The AmZ8001 removes the address and gates out the data to be written on the bus and activates the $\overline{DS}$ output LOW during T2. If the data to be written is a byte then the same byte will be on both halves of the bus. The $\overline{DS}$ output will go HIGH during T3 signifying completion of the cycle.

Interrupt and Segment Trap Acknowledge:

There are four status line codes devoted to interrupt and trap acknowledgement. These correspond to non-maskable, vectored and non-vectored interrupts, as well as segment trap. The Interrupt Acknowledge cycle is illustrated in Figure 9. The $\overline{NMI}$ input of the AmZ8001 is edge detected i.e., a HIGH to LOW input level change is stored in an internal latch. Similar internal storage is not provided for the $\overline{VI}$, $\overline{NVI}$, and $\overline{SEGT}$ inputs. For $\overline{VI}$ and $\overline{NVI}$ inputs to cause an interruption, the corresponding interrupt enable bits in the FCW must be 1. For the following discussion, both the VIE and NVIE bits in the FCW are assumed to be 1.

As shown in the figure, the $\overline{VI}$, $\overline{NVI}$ and $\overline{SEGT}$ input and the internal $\overline{NMI}$ latch output are sampled during T3 of the last machine cycle of an instruction.

A LOW on these signals triggers the corresponding interrupt acknowledge sequence described on the following page. The AmZ8001 executes a dummy IF1 cycle prior to entering the actual acknowledge cycle (see memory transactions for IF1 cycle description).

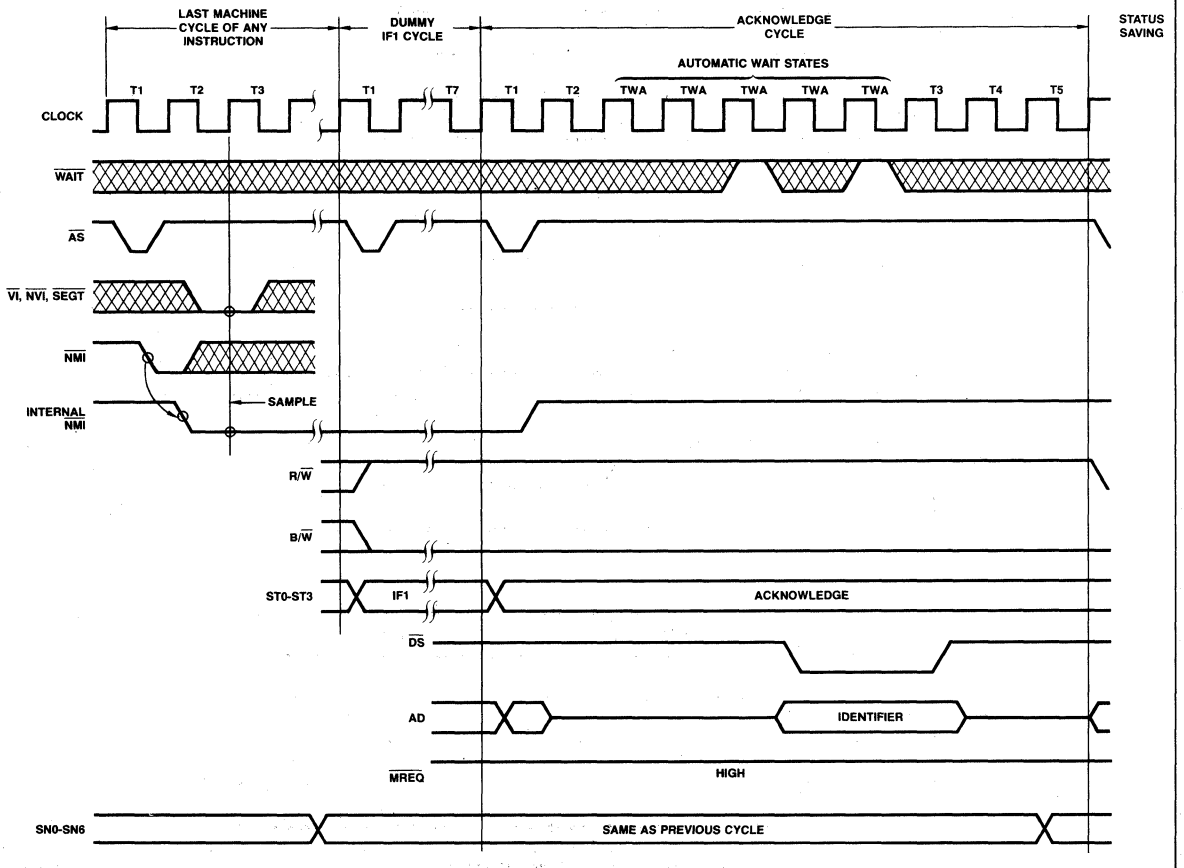


Figure 9. Interrupt Acknowledge Cycle.

MOS-250

During this dummy IF1 cycle, the program counter is not updated; instead the implied.system stack pointer (RR14') will be decremented. Following the dummy IF1 cycle is the actual interrupt/trap acknowledge cycle.

The interrupt acknowledge cycle typically consists of 10 clock periods; T1 through T5 and five automatic TW (wait states). As usual, the $\overline{AS}$ output will be pulsed LOW during T1 to mark the beginning of a cycle. The ST0-ST3 outputs will reflect the appropriate interrupt acknowledge code, the $\overline{MREQ}$ output will be HIGH, the N/$\overline{S}$ output remains the same as in the preceding cycle, the R/$\overline{W}$ output will be HIGH and the B/$\overline{W}$ output will be LOW. The AmZ8001 will drive the AD0-AD15 bus with unspecified information during T1 and the bus will go into the high impedance state during T2. Three TWA states will automatically follow T2. The $\overline{WAIT}$ input will be sampled during the third TWA state.

If LOW, an extra TW state will be inserted and the $\overline{WAIT}$ will be sampled again during TW. Such insertion of TW states continues until the $\overline{WAIT}$ input is HIGH. After the last TW state, the $\overline{DS}$ output will go LOW and two more automatic wait states (TWA) follow. The interrupting device can gate up to a 16-bit identifier on to the bus when the $\overline{DS}$ output is LOW. The $\overline{WAIT}$ input will be sampled again during the last TWA state. If the $\overline{WAIT}$ input is LOW one TW state will be inserted and the $\overline{WAIT}$ will be sampled during TW. Such TW insertion continues until the $\overline{WAIT}$ input is HIGH. After completing the last TW state T3 will be entered and the $\overline{DS}$ output will go HIGH. The interrupting device should remove the identifier and cease driving the bus. T4 and T5 states will follow T3 to complete the cycle. Following the interrupt acknowledge cycle will be memory transaction cycles to save the status on the stack. Note that the N/$\overline{S}$ output will be automatically LOW during status saving. The SN0-SN6 outputs are undefined during the acknowledge cycle.

The internal NMI latch will be reset to the initial state at $\overline{AS}$ going HIGH in the interrupt acknowledge cycle. The $\overline{VI}$, $\overline{NVI}$ and $\overline{SEGT}$ input should be kept LOW until this time also.

STATUS SAVING SEQUENCE:

The machine cycles following the interrupt acknowledge cycle push the old status information on the system stack in the following order: program counter, the flag and control word and the interrupt/trap identifier. Subsequent machine cycles fetch the new program status from the new program status area, and then branch to the interrupt/service routine.

BUS REQUEST/BUS ACKNOWLEDGE TIMING:

A LOW on the $\overline{BUSRQ}$ input is an indication to the AmZ8001 that another device (such as DMA) is requesting control of the bus. The $\overline{BUSRQ}$ input is synchronized internally at T1 of any machine cycle. (See next paragraph for exception.) The $\overline{BUSAK}$ will go LOW after the last clock period of the machine cycle. The LOW on the $\overline{BUSAK}$ output indicates acknowledgement. When $\overline{BUSAK}$ is LOW the following outputs will go into the high impedance state; AD0-AD15, $\overline{AS}$, $\overline{DS}$, $\overline{MREQ}$, ST0-ST3, B/$\overline{W}$, R/$\overline{W}$, SN0-SN6 and N/S. The BUSRQ must be held LOW until all transactions are completed. When $\overline{BUSRQ}$ goes HIGH, it is synchronized internally, the $\overline{BUSAK}$ output will go HIGH and normal CPU operation will resume. Figure 10 illustrates the $\overline{BUSRQ}$/$\overline{BUSAK}$ timing.

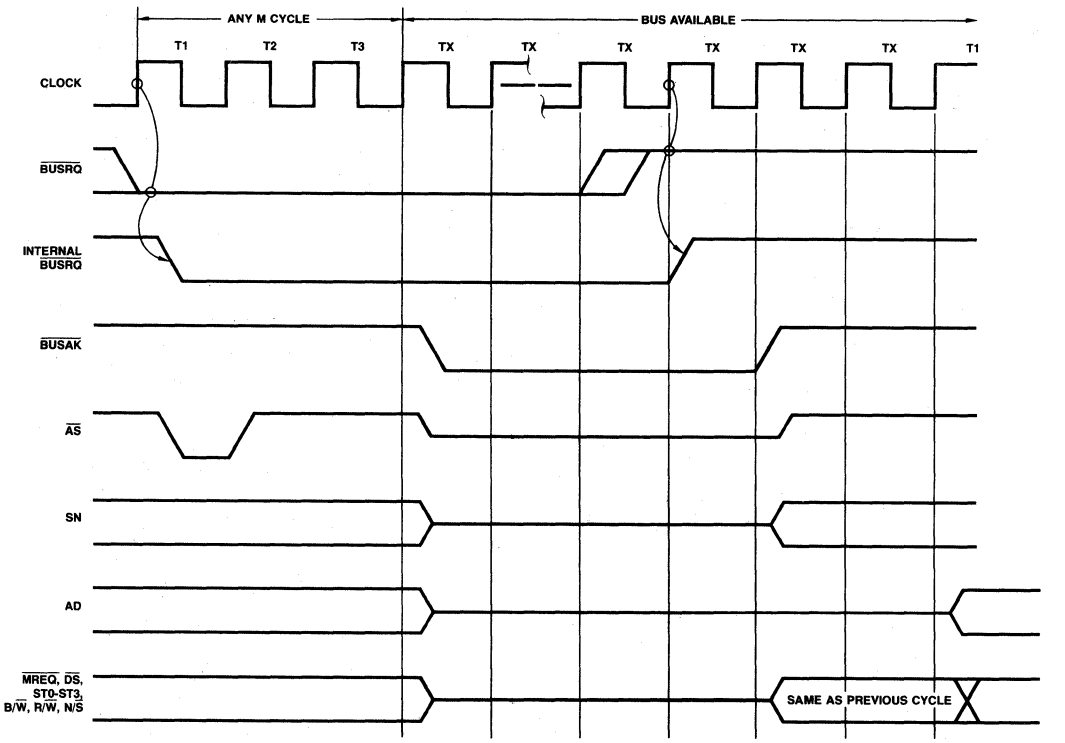


Figure 10. Bus Request/Acknowledge Cycle. MOS-251

It was mentioned that $\overline{BUSRQ}$ will be honored during any machine cycle with one exception. This exception is during the execution of TSET/TSETB instructions. $\overline{BUSRQ}$ will not be honored once execution of these instructions has started.

SINGLE STEPPING

The $\overline{STOP}$ input of the AmZ8001 facilitates one instruction at a time or single step operation. Figure 11 illustrates $\overline{STOP}$ input timing. The $\overline{STOP}$ input is sampled on the HIGH to LOW transition of the clock input that immediately precedes an IF1 cycle. If the $\overline{STOP}$ is found LOW, AmZ8001 introduces a memory refresh cycle after T3. Moreover, $\overline{STOP}$ input will be sampled again at T3 in the refresh cycle. If $\overline{STOP}$ is LOW one more refresh cycle will follow the previous refresh cycle. The $\overline{STOP}$ will be sampled during T3 of the refresh cycle also. One additional refresh cycle will be added every time $\overline{STOP}$ input is sampled LOW. After completing the last refresh cycle which will occur after $\overline{STOP}$ is HIGH, the CPU will insert two dummy states T4 and T5 to complete the IF1 cycle and resume its normal operations for executing the instruction. See appropriate sections on memory transactions and memory refresh. It should be noted that refresh cycles will occur even if the refresh facility is disabled during single stepping.

MULTIMICROPROCESSOR FACILITIES

The AmZ8001 is provided with hardware and software facilities to support multiple microprocessor systems. The $\overline{\mu O}$ and $\overline{\mu I}$ signals of the AmZ8001 are used in conjunction with the MBIT, MREQ, MRES and MSET instructions for this purpose. The $\overline{\mu O}$ output can be activated LOW by using appropriate instruction to signal a request from the AmZ8001 for a resource. The $\overline{\mu I}$ input is tested by the AmZ8001 before activating the $\overline{\mu O}$ output. LOW at the $\overline{\mu I}$ input indicates that the resource is busy. The AmZ8001 can examine the $\overline{\mu I}$ input after activating the $\overline{\mu O}$ output LOW. The $\overline{\mu I}$ will be tested again to see if the requested resource became available. For detailed information on the Multimicroprocessor facilities, AmZ8001/AmZ8002 Processor Interface Manual (Publication No. AM-PUB089) should be consulted.

INITIALIZATION

A LOW on the $\overline{Reset}$ input starts the CPU initialization. The initialization sequence is shown in Figure 12. Within five clock periods after the HIGH to LOW level change of the $\overline{Reset}$ input the following will occur:

a) AD0-AD15 bus will be in the HIGH impedance state
b) $\overline{AS}$, $\overline{DS}$, $\overline{MREQ}$, $\overline{BUSAK}$ and $\overline{\mu O}$ outputs will be HIGH
c) ST0-ST3 outputs will be LOW
d) Refresh will be disabled
e) R/$\overline{W}$, B/$\overline{W}$ and N/$\overline{S}$ outputs are not affected. For a power on reset the state of these outputs is not specified.
f) SN0-SN6 outputs will be LOW.

After the $\overline{Reset}$ input returns HIGH and remains HIGH for three clock periods, three 16-bit memory read operations will be performed as follows from segment C. Note that the N/$\overline{S}$ output will be LOW and ST0-ST3 outputs will reflect IFN code.

a) The contents of the memory location 0002 will be read. This information will be loaded into the FCW of the AmZ8001.
b) The contents of the memory location 0004 will be read. This information will be loaded into the program counter segment number.
c) The contents of the memory location 0006 will be read. This information will be loaded into the program counter offset.

This completes initialization sequence and an IF1 cycle will follow to fetch the first instruction to begin program execution. See the section on memory transactions for timing.

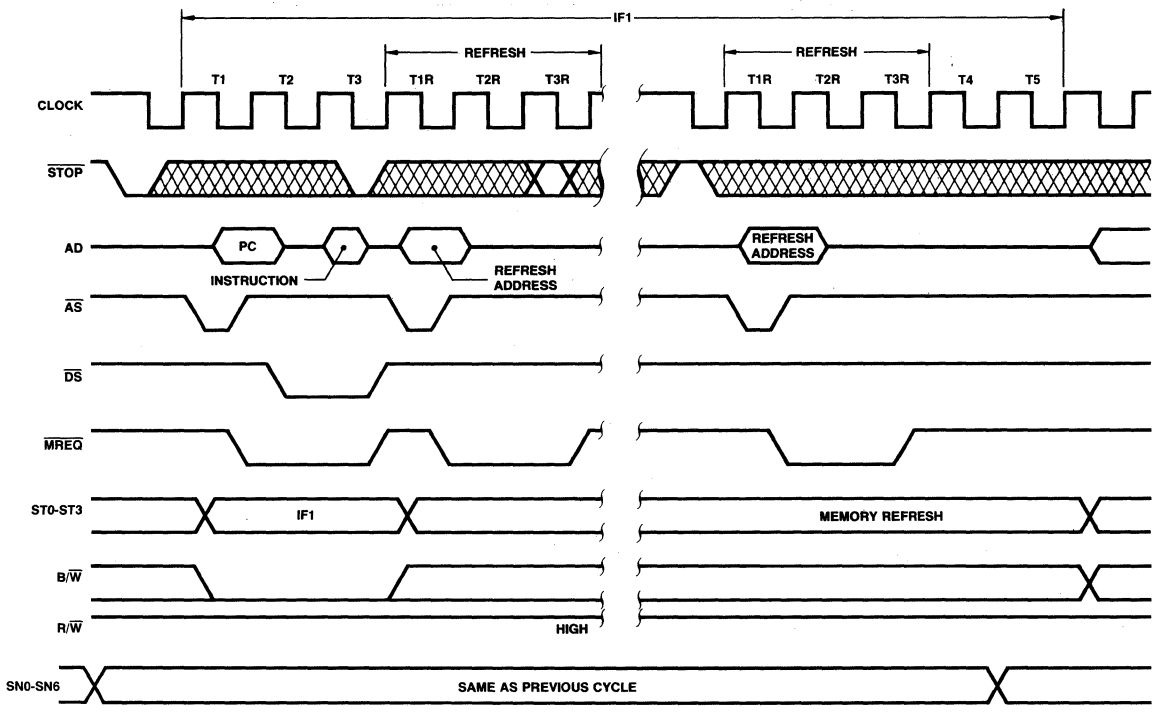


Figure 11. Single Step Timing. MOS-252

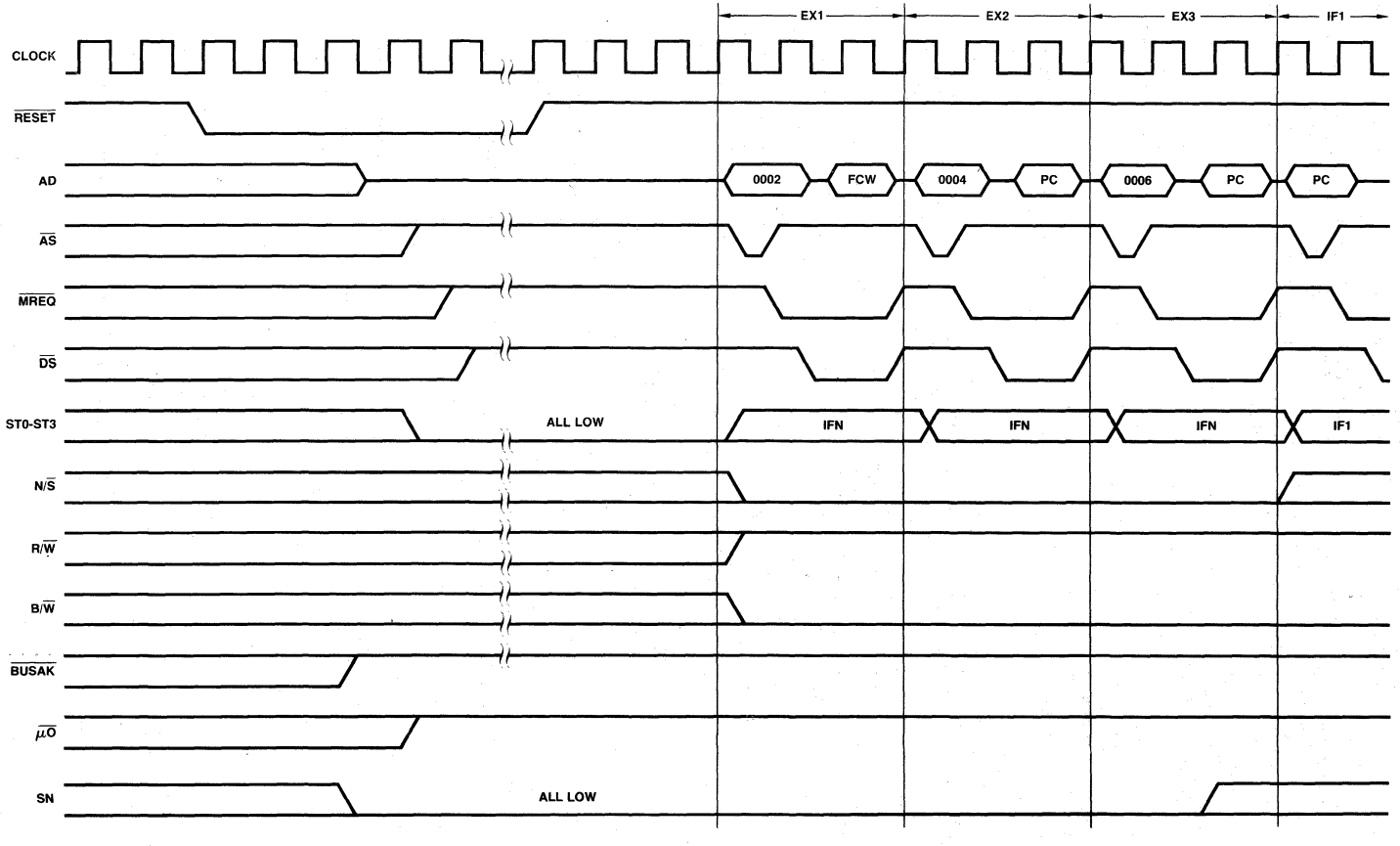


Figure 12. Reset Sequence.

AmZ8001 INSTRUCTION SET

LOAD AND EXCHANGE

Mnemonics	Operands	Addr. Modes	Operation
CLR CLRB	dst	R IR DA X	Clear dst ← 0
EX EXB	R, src	R IR DA X	Exchange R ← src
LD LDB LDL	R, src	R IM IM IR DA X BA BX	Load into Register R ← src
LD LDB LDL	dst, R	IR DA X BA BX	Load into Memory (Store) dst ← R
LD LDB	dst, IM	IR DA X	Load Immediate into Memory dst ← IM
LDA	R, src	DA X BA BX	Load Address R ← source address
LDAR	R, src	RA	Load Address Relative R ← source address
LDK	R, src	IM	Load Constant R ← n (n = 0 . . . 15)
LDM	R, src, n	IR DA X	Load Multiple R ← src (n consecutive words) (n = 1 . . . 16)
LDM	dst, R, n	IR DA X	Load Multiple (Store Multiple) dst ← R (n consecutive words) (n = 1 . . . 16)
LDR LDRB LDRL	R, src	RA	Load Relative R ← src (range −32768 . . . +32767)
LDR LDRB LDRL	dst, R	RA	Load Relative (Store Relative) dst ← R (range −32768 . . . +32767)
POP POPL	dst, R	R IR DA X	Pop dst ← IR Autoincrement contents of R
PUSH PUSHL	IR, src	R IM IR DA X	Push Autodecrement contents of R IR ← src

ARITHMETIC

Mnemonics	Operands	Addr. Modes	Operation
ADC ADCB	R, src	R	Add with Carry R ← R + src + carry
ADD ADDB ADDL	R, src	R IM IR DA X	Add R ← R + src
CP CPB CPL	R, src	R IM IR DA X	Compare with Register R − src
CP CPB	dst, IM	IR DA X	Compare with Immediate dst − IM
DAB	dst	R	Decimal Adjust
DEC DECB	dst, n	R IR DA X	Decrement by n dst ← dst − n (n = 1 . . . 16)
DIV DIVL	R, src	R IM IR DA X	Divide (signed) Word: $R_{n+1} \leftarrow R_{n,n+1} \div$ src $R_n \leftarrow$ remainder Long Word: $R_{n+2,n+3} \leftarrow R_{n...n+3} \div$ src $R_{n,n+1} \leftarrow$ remainder
EXTS EXTSB EXTSL	dst	R	Extend Sign Extend sign of low order half of st through high order half of dst
INC INCB	dst, n	R IR DA X	Increment by n dst ← dst + n (n = 1 . . . 16)
MULT MULTL	R, src	R IM IR DA X	Multiply (signed) Word: $R_{n,n+1} \leftarrow R_{n+1} \cdot$ src Long Word: $R_{n...n+3} \leftarrow R_{n+2,n+3} \cdot$ src *Plus seven cycles for each 1 in the multiplicand
NEG NEGB	dst	R IR DA X	Negate dst ← 0 − dst
SBC SBCB	R, src	R	Subtract with Carry R ← R − src − carry
SUB SUBB SUBL	R,src	R IM IR DA X	Subtract R ← R − src

LOGICAL

Mnemonics	Operands	Addr. Modes	Operation
AND ANDB	R, src	R IM IR DA X	AND R ← R AND src
COM COMB	dst	R IM IR DA X	Complement dst ← NOT dst
OR ORB	R, src	R IM IR DA X	OR R ← R OR src
TEST TESTB TESTL	dst	R IR DA X	TEST dst OR 0
TCC TCCB	cc, dst	R	Test Condition Code Set LSB if cc is true
XOR XORB	R, src	R IM IR DA X	Exclusive OR R ← R XOR src

PROGRAM CONTROL

Mnemonics	Operands	Addr. Modes	Operation
CALL	dst	IR DA X	Call Subroutine Autodecrement SP @ SP ← PC PC ← dst
CALR	dst	RA	Call Relative Autodecrement SP @ SP ← PC PC ← PC + dst (range −4094 to +4096)
DJNZ DBJNZ	R, dst	RA	Decrement and Jump if Non-Zero R ← R − 1 IF R = 0: PC ← PC + dst (range −254 to 0)
IRET*	–	–	Interrupt Return PS ← @ SP Autoincrement SP
JP	cc, dst	IR IR DA X	Jump Conditional If cc is true: PC ← dst
JR	cc, dst	RA	Jump Conditional Relative If cc is true: PC ← PC + dst (range −256 to +254)
RET	cc	–	Return Conditional If cc is true: PC ← @ SP Autodecrement SP
SC	src	IM	System Call Autodecrement SP @ SP ← old PS Push instruction PS ← System Call PS

*Privileged instructions. Executed in system mode only.

BIT MANIPULATION

Mnemonics	Operand	Addr. Modes	Operation
BIT BITB	dst, b	R IR DA X	Test Bit Static Z flag ← NOT dst bit specified by b
BIT BITB	dst, R	R	Test Bit Dynamic Z flag ← NOT dst bit specified by contents of R
RES RESB	dst, b	R IR DA X	Reset Bit Static Reset dst bit specified by b
RES RESB	dst, R	R	Reset Bit Dynamic Reset dst bit specified by contents of R
SET SETB	dst, b	R IR DA X	Set Bit Static Set dst bit specified by b
SET SETB	dst, R	R	Set Bit Dynamic Set dst bit specified by contents of R
TSET TSETB	dst	R IR DA X	Test and Set S flag ← MSB of dst dst ← all 1s

ROTATE AND SHIFT

Mnemonics	Operand	Addr. Modes	Operation
RLDB	R, src	R	Rotate Digit Left
RRDB	R, src	R	Rotate Digit Right
RL RLB	dst, n	R R	Rotate Left by n bits (n = 1, 2)
RLC RLCB	dst, n	R R	Rotate Left through Carry by n bits (n = 1, 2)
RR RRB	dst, n	R R	Rotate Right by n bits (n = 1, 2)
RRC RRCB	dst, n	R R	Rotate Right through Carry by n bits (n = 1, 2)
SDA SDAB SDAL	dst, R	R	Shift Dynamic Arithmetic Shift dst left or right by contents of R
SDL SDLB SDLL	dst, R	R	Shift Dynamic Logical Shift dst left or right by contents of R
SLA SLAB SLAL	dst, n	R	Shift Left Arithmetic by n bits
SLL SLLB SLLL	dst, n	R	Shift Left Logical by n bits
SRA SRAB SRAL	dst, n	R	Shift Right Arithmetic by n bits
SRL SRLB SRLL	dst, n	R	Shift Right Logical by n bits

BLOCK TRANSFER AND STRING MANIPULATION

Mnemonics	Operands	Addr. Modes	Operation
CPD CPDB	R_X, src, R_Y, cc	IR	Compare and Decrement R_X − src Autodecrement src address $R_Y \leftarrow R_Y - 1$
CPDR CPDRB	R_X, src, R_Y, cc	IR	Compare, Decrement and Repeat R_X − src Autodecrement src address $R_Y \leftarrow R_Y - 1$ Repeat until cc is true or $R_Y = 0$
CPI CPIB	R_X, src, R_Y, cc	IR	Compare and Increment R_X − src Autoincrement src address $R_Y \leftarrow R_Y - 1$
CPIR CPIRB	R_X, src, R_Y, cc	IR	Compare, Increment and Repeat R_X − src Autoincrement src address $R_Y \leftarrow R_Y - 1$ Repeat until cc is true or $R_Y = 0$
CPSD CPSDB	dst, src, R, cc	IR	Compare String and Decrement dst − src Autodecrement dst and src addresses R ← R − 1
CPSDR CPSDRB	dst, src, R, cc	IR	Compare String, Decr. and Repeat dst − src Autodecrement dst and src addresses R ← R − 1 Repeat until cc is true or R = 0
CPSI CPSIB	dst, src, R, cc	IR	Compare String and Increment dst − src Autoincrement dst and src addresses R ← R − 1
CPSIR CPSIRB	dst, src, R, cc	IR	Compare String, Incr. and Repeat dst − src Autoincrement dst and src addresses R ← R − 1 Repeat until cc is true or R = 0
LDD LDDB	dst, src, R	IR	Load and Decrement dst ← src Autodecrement dst and src addresses R ← R − 1
LDDR LDDRB	dst, src, R	IR	Load, Decrement and Repeat dst ← src Autodecrement dst and src addresses R ← R − 1 Repeat until R = 0

BLOCK TRANSFER AND STRING MANIPULATION (Cont.)

Mnemonics	Operands	Addr. Modes	Operation
LDI LDIB	dst, src, R	IR	Load and Increment dst ← src Autoincrement dst and src addresses R ← R − 1
LDIR LDIRB	dst, src, R	IR	Load, Increment and Repeat dst ← src Autoincrement dst and src addresses R ← R − 1 Repeat until R = 0
TRDB	dst, src, R	IR	Translate and Decrement dst ← src (dst) Autodecrement dst address R ← R − 1
TRDRB	dst, src, R	IR	Translate, Decrement and Repeat dst ← src (dst) Autodecrement dst address R ← R − 1 Repeat until R = 0
TRIB	dst, src, R	IR	Translate and Increment dst ← src (dst) Autoincrement dst address R ← R − 1
TRIRB	dst, src, R	IR	Translate, Increment and Repeat dst ← src (dst) Autoincrement dst address R ← R − 1 Repeat until R = 0
TRTDB	src 1, src 2, R	IR	Translate and Test, Decrement RH1 ← src 2 (src 1) Autodecrement src 1 address R ← R − 1
TRTDRB	src 1, src 2, R	IR	Translate and Test, Decrement and Repeat RH1 ← src 2 (src 1) Autodecrement src 1 address R ← R − 1 Repeat until R = 0 or RH1 = 0
TRTIB	src 1, src 2, R	IR	Translate and Test, Increment RH1 ← src 2 (src 1) Autoincrement src 1 address R ← R − 1
TRTIRB	src 1, src 2, R	IR	Translate and Test, Increment and Repeat RH1 ← src 2 (src 1) Autoincrement src 1 address R ← R − 1 Repeat until R = 0 or RH1 = 0

INPUT/OUTPUT

Mnemonics	Operands	Addr. Modes	Operation
IN* INB*	R, src	IR DA	Input R ← src
IND* INDB*	dst, src, R	IR	Input and Decrement dst ← src Autodecrement dst address R ← R − 1
INDR* INDRB*	dst, src, R	IR	Input, Decrement and Repeat dst ← src Autodecrement dst address R ← R − 1 Repeat until R = 0
INI* INIB*	dst, src, R	IR	Input and Increment dst ← src Autoincrement dst address R ← R − 1
INIR* INIRB*	dst, src, R	IR	Input, Increment and Repeat dst ← src Autoincrement dst address R ← R − 1 Repeat until R = 0
OUT* OUTB*	dst, R	IR DA	Output dst ← R
OUTD* OUTDB*	dst, src, R	IR	Output and Decrement dst ← src Autodecrement src address R ← R − 1
OTDR* OTDRB*	dst, src, R	IR	Output, Decrement and Repeat dst ← src Autodecrement src address R ← R − 1 Repeat until R = 0
OUTI* OUTIB*	dst, src, R	IR	Output and Increment dst ← src Autoincrement src address R ← R − 1
OTIR* OTIRB*	dst, src, R	IR	Ouput, Increment and Repeat dst ← src Autoincrement src address R ← R − 1 Repeat until R = 0
SIN* SINB*	R, src	DA	Special Input R ← src
SIND* SINDB*	dst, src, R	IR	Special Input and Decrement dst ← src Autodecrement dst address R ← R − 1
SINDR* SINDRB*	dst, src, R	IR	Special Input, Decr. and Repeat dst ← src Autodecrement dst address R ← R − 1 Repeat until R = 0
SINI* SINIB*	dst, src, R	IR	Special Input and Increment dst ← src Autoincrement dst address R ← R − 1
SINIR* SINIRB*	dst, src, R	IR	Special Input, Incr. and Repeat dst ← src Autoincrement dst address R ← R − 1 Repeat until R = 0

INPUT/OUTPUT (Cont.)

Mnemonics	Operands	Addr. Modes	Operation
SOUT* SOUTB*	dst, src	DA	Special Output dst ← src
SOUTD* SOUTDB*	dst, src, R	IR	Special Output and Decrement dst ← src Autodecrement src address R ← R − 1
SOTDR* SOTDRB*	dst, src, R	IR	Special Output, Decr. and Repeat dst ← src Autodecrement src address R ← R − 1 Repeat until R = 0
SOUTI* SOUTIB*	dst, src, R	IR	Special Output and Increment dst ← src Autoincrement src address R ← R − 1
SOTIR* SOTIRB*	dst, src, R	R	Special Output, Incr. and Repeat dst ← src Autoincrement src address R ← R − 1 Repeat until R = 0

CPU CONTROL

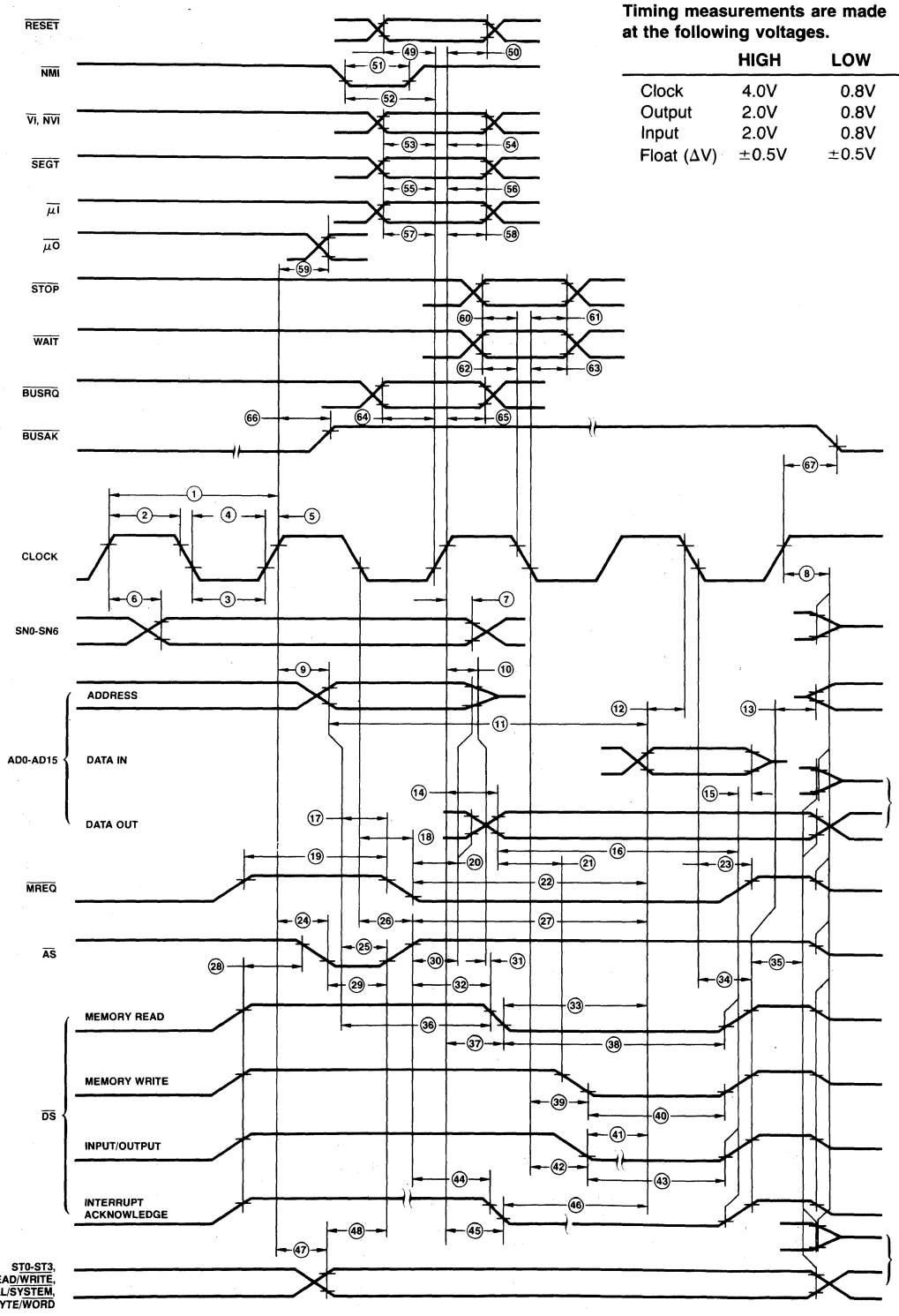
Mnemonics	Operands	Addr. Modes	Operation
COMFLG	flags	–	Complement Flag (Any combination of C, Z, S, P/V)
DI*	int	–	Disable Interrupt (Any combination of NVI, VI)
EI*	int	–	Enable Interrupt (Any combination of NVI, VI)
HALT*	–	–	HALT
LDCTL*	CTLR, src	R	Load into Control Register CTLR ← src
LDCTL*	dst, CTLR	R	Load from Control Register dst ← CTLR
LDCTLB	FLGR, src	R	Load into Flag Byte Register FLGR ← src
LDCTLB	dst, FLGR	R	Load from Flag Byte Register dst ← FLGR
LDPS*	src	IR DA X	Load Program Status PS ← src
MBIT*	–	–	Test Multi-Micro Bit Set S if $\overline{\mu I}$ is High; reset S if $\overline{\mu I}$ is Low.
MREQ*	dst	R	Multi-Micro Request
MRES*	–	–	Multi-Micro Reset
MSET*	–	–	Multi-Micro Set
NOP	–	–	No Operation
RESFLG	flag	–	Reset Flag (Any combination of C, Z, S, P/V)
SETFLG	flag	–	Set Flag (Any combination of C, Z, S, P/V)

*Privileged instructions. Executed in system mode only.

AC TIMING DIAGRAM

Timing measurements are made at the following voltages.

	HIGH	LOW
Clock	4.0V	0.8V
Output	2.0V	0.8V
Input	2.0V	0.8V
Float (ΔV)	±0.5V	±0.5V



This composite timing diagram does not show actual timing sequences. Refer to this diagram only for the detailed timing relationships of individual edges. Use the preceding illustrations as an explanation of the various timing sequences.

MOS-255

SWITCHING CHARACTERISTICS over operating range — AmZ8001DC

Number	Parameter	Description	Min	Max	Units
1	TcC	Clock Cycle Time	250	2000	ns
2	TwCh	Clock Width (High)	105	2000	ns
3	TwCl	Clock Width (Low)	105	2000	ns
4	TfC	Clock Fall Time		20	ns
5	TrC	Clock Rise Time		20	ns
6	TdC(SNv)	Clock ↑ to Segment Number Valid (50pF Load)		130	ns
7	TdC(SNn)	Clock ↑ to Segment Number Not Valid	20		ns
8	TdC(Bz)	Clock ↑ to Bus Float		65	ns
9	TdC(A)	Clock ↑ to Address Valid		100	ns
10	TdC(Az)	Clock ↑ to Address Float		65	ns
11	TdA(DI)	Address Valid to Data In Required Valid	400		ns
12	TsDI(C)	Data In to Clock ↓ Set-up Time	70		ns
13	TdDS(A)	$\overline{DS}$ ↑ to Address Active	80		ns
14	TdC(DO)	Clock ↑ to Data Out Valid		100	ns
15	ThDI(DS)	Data In to $\overline{DS}$ ↑ Hold Time	0		ns
16	TdDO(DS)	Data Out Valid to $\overline{DS}$ ↑ Delay	230		ns
17	TdA(MR)	Address Valid to $\overline{MREQ}$ ↓ Delay	55		ns
18	TdC(MR)	Clock ↓ to $\overline{MREQ}$ ↓ Delay		80	ns
19	TwMRh	$\overline{MREQ}$ Width (High)	190		ns
20	TdMR(A)	$\overline{MREQ}$ ↓ to Address Not Active	70		ns
21	TdDO(DSW)	Data Out Valid to DS ↓ (Write) Delay	55		ns
22	TdMR(DI)	$\overline{MREQ}$ ↓ to Data In Required Valid	330		ns
23	TdC(MR)	Clock ↓ to $\overline{MREQ}$ ↑ Delay		80	ns
24	TdC(ASf)	Clock ↑ to $\overline{AS}$ ↓ Delay		80	ns
25	TdA(AS)	Address Valid to $\overline{AS}$ ↑ Delay	55		ns
26	TdC(ASr)	Clock ↓ to $\overline{AS}$ ↑ Delay		90	ns
27	TdAS(DI)	$\overline{AS}$ ↑ to Data In Required Valid	290		ns
28	TdDS(AS)	$\overline{DS}$ ↑ to $\overline{AS}$ ↓ Delay	70		ns
29	TwAS	$\overline{AS}$ Width (Low)	80		ns
30	TdAS(A)	$\overline{AS}$ ↑ to Address Not Active Delay	60		ns
31	TdAz(DSR)	Address Float to $\overline{DS}$ (Read) ↓ Delay	0		ns
32	TdAS(DSR)	$\overline{AS}$ ↑ to $\overline{DS}$ (Read) ↓ Delay	70		ns
33	TdDSR(DI)	$\overline{DS}$ (Read) ↓ to Data In Required Valid	155		ns
34	TdC(DSr)	Clock ↓ to $\overline{DS}$ ↑ Delay		70	ns
35	TdDS(DO)	$\overline{DS}$ ↑ to Data Out and STATUS Not Valid	80		ns
36	TdA(DSR)	Address Valid to $\overline{DS}$ (Read) ↓ Delay	120		ns
37	TdC(DSR)	Clock ↑ to $\overline{DS}$ (Read) ↓ Delay		120	ns
38	TwDSR	$\overline{DS}$ (Read) Width (Low)	275		ns
39	TdC(DSW)	Clock ↓ to $\overline{DS}$ (Write) ↓ Delay		95	ns
40	TwDSW	$\overline{DS}$ (Write) Width (Low)	160		ns
41	TdDSI(DI)	$\overline{DS}$ (Input) ↓ to Data In Required Valid	315		ns
42	TdC(DSf)	Clock ↓ to $\overline{DS}$ (I/O) ↓ Delay		120	ns
43	TwDS	$\overline{DS}$ (I/O) Width (Low)	400		ns
44	TdAS(DSA)	$\overline{AS}$ ↑ to $\overline{DS}$ (Acknowledge) ↓ Delay	960		ns
45	TdC(DSA)	Clock ↑ to $\overline{DS}$ (Acknowledge) ↓ Delay		120	ns
46	TdDSA(DI)	$\overline{DS}$ (Acknowledge) ↓ to Data In Required Delay	420		ns
47	TdC(S)	Clock ↑ to Status Valid Delay		110	ns
48	TdS(AS)	Status Valid to $\overline{AS}$ ↑ Delay	40		ns

SWITCHING CHARACTERISTICS (Cont.)

AmZ8001DC

Number	Parameter	Description	Min	Max	Units
49	TsR(C)	$\overline{\text{RESET}}$ to Clock ↑ Set-up Time	180		ns
50	ThR(C)	$\overline{\text{RESET}}$ to Clock ↑ Hold Time	0		ns
51	TwNMI	$\overline{\text{NMI}}$ Width (Low)	100		ns
52	TsNMI(C)	$\overline{\text{NMI}}$ to Clock ↑ Set-up Time	140		ns
53	TsVI(C)	$\overline{\text{VI}}$, N$\overline{\text{VI}}$ to Clock ↑ Set-up Time	110		ns
54	ThVI(C)	$\overline{\text{VI}}$, N$\overline{\text{VI}}$ to Clock ↑ Hold Time	0		ns
55	TsSGT(C)	$\overline{\text{SEGT}}$ to Clock ↑ Set-up Time	70		ns
56	ThSGT(C)	$\overline{\text{SEGT}}$ to Clock ↑ Hold Time	0		ns
57	TsμI(C)	μI to Clock ↑ Set-up Time	180		ns
58	ThμI(C)	μI to Clock ↑ Hold Time	0		ns
59	TdC(μO)	Clock ↑ to μO Delay		120	ns
60	TsSTP(C)	$\overline{\text{STOP}}$ to Clock ↓ Set-up Time	140		ns
61	ThSTP(C)	$\overline{\text{STOP}}$ to Clock ↓ Hold Time	0		ns
62	TsWT(C)	$\overline{\text{WAIT}}$ to Clock ↓ Set-up Time	70		ns
63	ThWT(C)	$\overline{\text{WAIT}}$ to Clock ↓ Hold Time	0		ns
64	TsBRQ(C)	$\overline{\text{BUSRQ}}$ to Clock ↑ Set-up Time	90		ns
65	ThBRQ(C)	$\overline{\text{BUSRQ}}$ to Clock ↑ Hold Time	0		ns
66	TdC(BAKr)	Clock ↑ to $\overline{\text{BUSAK}}$ ↑ Delay		100	ns
67	TdC(BAKf)	Clock ↑ to $\overline{\text{BUSAK}}$ ↓ Delay		100	ns

MAXIMUM RATINGS above which useful life may be impaired.

Voltages on all inputs and outputs with respect to GND	−0.3V to +7.0V
Operating Ambient Temperature	0 to +70°C
Storage Temperature	−65 to +150°C

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

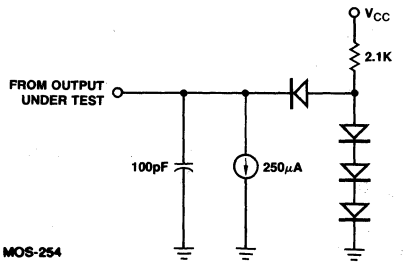
ELECTRICAL CHARACTERISTICS over operating range (Note 1)

Parameter	Description	Test Conditions	Min	Max	Units
V_{CH}	Clock Input High Voltage	Driven by External Clock Generator	$V_{CC}-0.4$	$V_{CC}+0.3$	Volts
V_{CL}	Clock Input Low Voltage	Driven by External Clock Generator	−0.3	0.45	Volts
V_{IH}	Input High Voltage		2.0	$V_{CC}+0.3$	Volts
V_{IL}	Input Low Voltage		−0.3	0.8	Volts
V_{OH}	Output High Voltage	$I_{OH} = -250\mu A$	2.4		Volts
V_{OL}	Output Low Voltage	$I_{OL} = +2.0mA$		0.4	Volts
I_{IL}	Input Leakage	$0.4 \leq V_{IN} \leq +2.4V$		±10	μA
I_{OL}	Output Leakage	$0.4 \leq V_{OUT} \leq +2.4V$		±10	μA
I_{CC}	V_{CC} Supply Current			300	mA

Standard Test Conditions

The characteristics below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND. Positive current flows into the referenced pin. Standard conditions are as follows:

$+4.75V \leq V_{CC} \leq +5.25V$
GND = 0V
$0°C \leq T_A \leq +70°C$



All AC parameters assume a load capacitance of 100pF max, except for parameter 6, TdC(SNv) (50pF max). Timing references between two output signals assume a load difference of 50pF max.

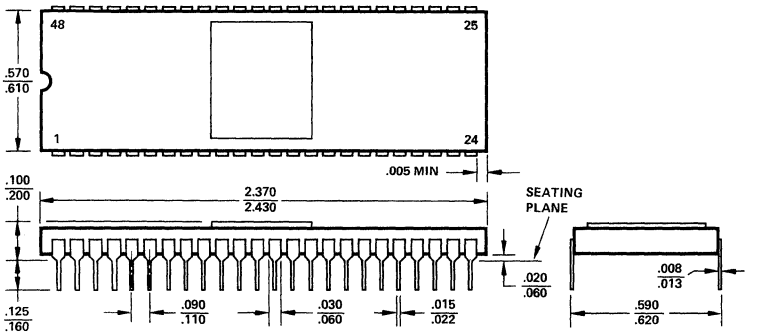
For more information, refer to these AMD publications:

Processor Instruction Set (AM-PUB086).
Describes each instruction in detail. 250 pp.

Processor Interface Manual (AM-PUB089).
Describes hardware interfacing for interrupts and I/O, including the Am9511A Arithmetic Processor, the Am9517A DMA Controller, and the Am9519 Interrupt Controller. 81 pp.

PHYSICAL DIMENSIONS
Dual-In-Line

48-Pin Ceramic



For more information, refer to these AMD publications:

Processor Instruction Set (AM-PUB086).
Describes each instruction in detail. 250 pp.

Processor Interface Manual (AM-PUB089).
Describes hardware interfacing for interrupts and I/O, including the Am9511A Arithmetic Processor, the Am9517A DMA Controller, and the Am9519 Interrupt Controller. 81 pp.

AmZ8002
16-Bit Microprocessor

PRELIMINARY DATA

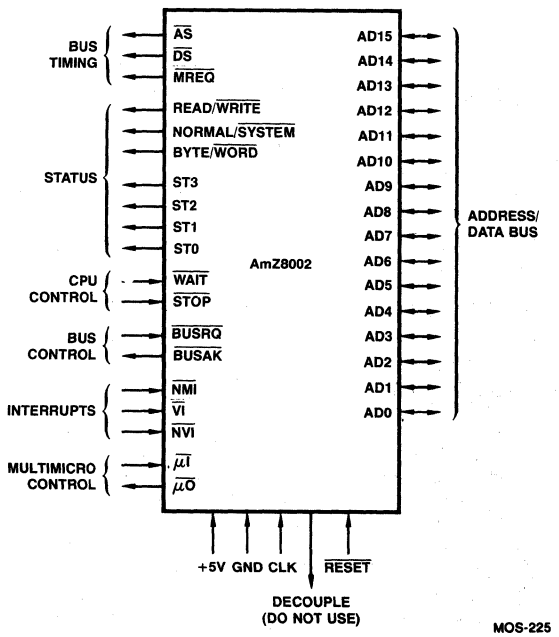
DISTINCTIVE CHARACTERISTICS

- Sixteen general purpose registers
- Direct addressing up to 64KB memory
- Software compatible with AmZ8001 microprocessor
- Powerful instructions with flexible addressing modes
- Privileged/Non-Privileged mode of operation
- Sophisticated interrupt structure
- On-chip memory refresh facility
- TTL compatible inputs and outputs
- Single phase clock
- Single +5V power supply
- 40-pin package

GENERAL DESCRIPTION

The AmZ8002 is a general-purpose 16-bit CPU belonging to the AmZ8000 family of microprocessors. Its architecture is centered around sixteen 16-bit general registers. The CPU deals with 16-bit address spaces and hence can address directly 64 Kilobytes of memory. Facilities are provided to maintain three distinct address spaces – code, data and stack. The AmZ8002 implements a powerful instruction set with flexible addressing modes. These instructions operate on several data types – bit, byte, word (16-bit), long word (32-bit), byte string and word string. The CPU can execute instructions in one of two modes – System and Normal. Sometimes these modes are also known as Privileged and Non-Privileged, respectively. The CPU also contains an on-chip memory refresh facility. The AmZ8002 is software compatible with the AmZ8001 microprocessor. The AmZ8002 is fabricated using silicon-gate N-MOS technology and is packaged in a 40-pin DIP. The AmZ8002 requires a single +5 power supply and a single phase clock for its operation.

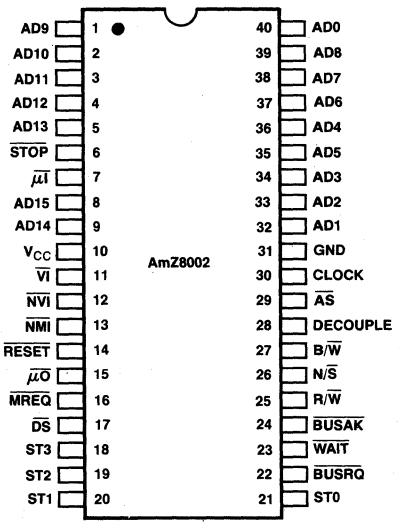
LOGIC SYMBOL



MOS-225

CONNECTION DIAGRAM

Top View



Note: Pin 1 is marked for orientation.

MOS-226

ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency 4MHz
Hermetic DIP	0°C ≤ T_A ≤ 70°C	AmZ8002DC

INTERFACE SIGNAL DESCRIPTION

V_{CC}: +5V Power Supply

V_{SS}: Ground

AD0-AD15: Address/Data Bus (Bidirectional, 3-State)

This 16-bit multiplexed address/data bus is used for all I/O and memory transactions. HIGH on the bus corresponds to 1 and LOW corresponds to 0. AD0 is the least significant bit position with AD15 the most significant. The $\overline{AS}$ output and $\overline{DS}$ output will indicate whether the bus is used for address or data. The status output lines ST0-ST3 will indicate the type of transaction; memory or I/O.

$\overline{AS}$: Address Strobe (Output, 3-State)

LOW on this output indicates that the AD0-AD15 bus contains address information. The address information is stable by the time of the LOW-to-HIGH transition of the $\overline{AS}$ output (see timing diagrams). The status outputs ST0-ST3 indicate whether the bus contains a memory address or I/O address.

$\overline{DS}$: Data Strobe (Output, 3-State)

LOW on this output indicates that the AD0-AD15 bus is being used for data transfer. The R/$\overline{W}$ output indicates the direction of data transfer – read (or in) means data into the CPU and write (or out) means data from the CPU. During a read operation, data can be gated on to the bus when $\overline{DS}$ goes LOW. A LOW-to-HIGH transition on the $\overline{DS}$ output indicates that the CPU has accepted the data (see timing diagram). During a write operation, LOW on the $\overline{DS}$ output indicates that data is setup on the bus. Data will be removed sometime after the LOW-to-HIGH transition of the $\overline{DS}$ output (see timing diagram).

R/$\overline{W}$: Read/Write (Output, 3-State)

This output indicates the direction of data flow on the AD0-AD15 bus. HIGH indicates a read operation, i.e., data into the CPU and LOW indicates a write operation, i.e., data from the CPU. This output is activated at the same time as $\overline{AS}$ going LOW and remains stable for the duration of the whole transaction (see timing diagram).

B/$\overline{W}$: Byte/Word (Output, 3-State)

This output indicates the type of data transferred on the AD0-AD15 bus. HIGH indicates byte (8-bit) and LOW indicates word (16-bit) transfer. This output is activated at the same stage as $\overline{AS}$ going LOW and remains valid for the duration of the whole transaction (see timing diagram). The address generated by the CPU is always a byte address. However, the memory is organized as 16-bit words. All instructions and word operands are word aligned and are addressed by even addresses. Thus, for all word transactions with the memory the least significant address bit will be zero. When addressing the memory for byte transactions, the least significant address bit determines which byte of the memory word is needed; even address specifies the most significant byte and odd address specifies the least significant byte. In the case of I/O transactions, the address information on the AD0-AD15 bus refers to an I/O port and B/$\overline{W}$ determines whether a data word or data byte will be transacted. During I/O byte transactions, the least significant address bit A0 determines which half of the AD0-AD15 bus will be used for the I/O transactions. The ST0-ST3 outputs will indicate whether the current transaction is for memory, normal I/O or special I/O.

ST0-ST3: Status (Outputs, 3-State)

These four outputs contain information regarding the current transaction in a coded form. The status line codes are shown in the following table:

ST3	ST2	ST1	ST0	
L	L	L	L	Internal Operation
L	L	L	H	Memory Refresh
L	L	H	L	Normal I/O Transaction
L	L	H	H	Special I/O Transaction
L	H	L	L	Reserved
L	H	L	H	Non-Maskable Interrupt Acknowledge
L	H	H	L	Non-Vectored Interrupt Acknowledge
L	H	H	H	Vectored Interrupt Acknowledge
H	L	L	L	Memory Transaction for Operand
H	L	L	H	Memory Transaction for Stack
H	L	H	L	Reserved
H	L	H	H	Reserved
H	H	L	L	Memory Transaction for Instruction Fetch (Subsequent Word)
H	H	L	H	Memory Transaction for Instruction Fetch (First Word)
H	H	H	L	Reserved
H	H	H	H	Reserved

$\overline{WAIT}$: Wait (Input)

LOW on this input indicates to the CPU that memory or I/O is not ready for the data transfer and hence the current transaction should be stretched. The $\overline{WAIT}$ input is sampled by the CPU at certain instances during the transaction (see timing diagram). If $\overline{WAIT}$ input is LOW at these instances, the CPU will go into wait state to prolong the transaction. The wait state will repeat until the $\overline{WAIT}$ input is HIGH at the sampling instant.

N/$\overline{S}$: Normal/System Mode (Output, 3-State)

HIGH on this output indicates that the CPU is operating in Normal Mode and LOW indicates operation in System Mode. This output is derived from the Flag Control Word (FCW) register. The FCW register is described under the program status information section of this document.

$\overline{MREQ}$: Memory Request (Output, 3-State)

LOW on this output indicates that a CPU transaction with memory is taking place.

$\overline{BUSRQ}$: Bus Request (Input)

LOW on this input indicates to the CPU that another device (such as DMA) is requesting to take control of the bus. The $\overline{BUSRQ}$ input can be driven LOW anytime. The CPU synchronizes this input internally. The CPU responds by activating $\overline{BUSAK}$ output LOW to indicate that the bus has been relinquished. Relinquishing the bus means that the AD0-AD15, $\overline{AS}$, $\overline{DS}$, B/$\overline{W}$, R/$\overline{W}$, N/$\overline{S}$, ST0-ST3 and $\overline{MREQ}$ outputs will be in the high impedance state. The requesting device should control these lines in an identical fashion to the CPU to accomplish transactions. The $\overline{BUSRQ}$ input must remain LOW as long as needed to perform all the transactions and the CPU will keep the $\overline{BUSAK}$ output LOW. After completing the transactions, the device must disable the AD0-AD15, $\overline{AS}$, $\overline{DS}$, B/$\overline{W}$, R/$\overline{W}$, N/$\overline{S}$, ST0-ST3 and $\overline{MREQ}$ into the high impedance state and stop driving the $\overline{BUSRQ}$ input LOW. The CPU will make $\overline{BUSAK}$ output HIGH sometime later and take back the bus control.

$\overline{BUSAK}$: Bus Acknowledge (Output)

LOW on this output indicates that the CPU has relinquished the bus in response to a bus request.

$\overline{NMI}$: Non-Maskable Interrupt (Input)

HIGH-to-LOW transition on this input constitutes non-maskable interrupt request. The CPU will respond with the non-maskable Interrupt Acknowledge on the ST0-ST3 outputs and will enter an interrupt sequence. The transition on the $\overline{NMI}$ can occur anytime. Of the three kinds of interrupts available, the non-maskable interrupt has the highest priority.

$\overline{VI}$: Vectored Interrupt (Input)

LOW on this input constitutes vectored interrupt request. Vectored interrupt is next lower to the non-maskable interrupt in priority. The VIE bit in the Flag and Control Word register must be 1 for the vectored interrupt to be honored. The CPU will respond with Vectored Interrupt Acknowledge code on the ST0-ST3 outputs and will begin the interrupt sequence. The $\overline{VI}$ input can be driven LOW any time and should be held LOW until acknowledged.

$\overline{NVI}$: Non-Vectored Interrupt (Input)

LOW on this input constitutes non-vectored interrupt request. Non-vectored has the lowest priority of the three types of interrupts. The NVIE bit in the Flag and Control Word register must be 1 for this request to be honored. The CPU will respond with Non-Vectored Interrupt Acknowledge code on the $\overline{ST0}$-ST3 outputs and will begin the interrupt sequence. The $\overline{NVI}$ input can be driven LOW anytime and should be held LOW until acknowledged.

$\overline{\mu I}$: Micro-In (Input)

This input participates in the resource request daisy chain. See the section on multimicroprocessor support facilities in this document.

$\overline{\mu O}$: Micro-Out (Output)

This output participates in the resource request daisy chain. See the section on multimicroprocessor support facilities in this document.

$\overline{RESET}$: Reset (Input)

LOW on this input initiates a reset sequence in the CPU. See the section on Initialization for details on reset sequence.

CLK: Clock (Input)

All CPU operations are controlled from the signal fed into this input. See DC Characteristics for clock voltage level requirements.

DECOUPLE: (Output)

Output from the on-chip substrate bias generator. Do not use.

$\overline{STOP}$: Stop (Input)

This active LOW input facilitates one instruction at a time operation. See the section on single stepping.

PROCESSOR ORGANIZATION

The following is a brief discussion of the AmZ8002 CPU. For detailed information, see the AmZ8001/AmZ8002 Processor Instruction Set Manual (Publication No. AM-PUB086).

General Purpose Registers

The CPU is organized around sixteen 16-bit general purpose registers R0 through R15 as shown in Figure 1. For byte operations, the first eight registers (R0 through R7) can also be addressed as sixteen 8-bit registers designated as RL0, RH0 and so on to RL7 and RH7. The sixteen registers can also be grouped in pairs RR0, RR2 and so on to RR14 to form eight long word (32-bit) registers. Similarly, the sixteen registers can be grouped in quadruples RQ0, RQ4, RQ8 and RQ12 to form four 64-bit registers.

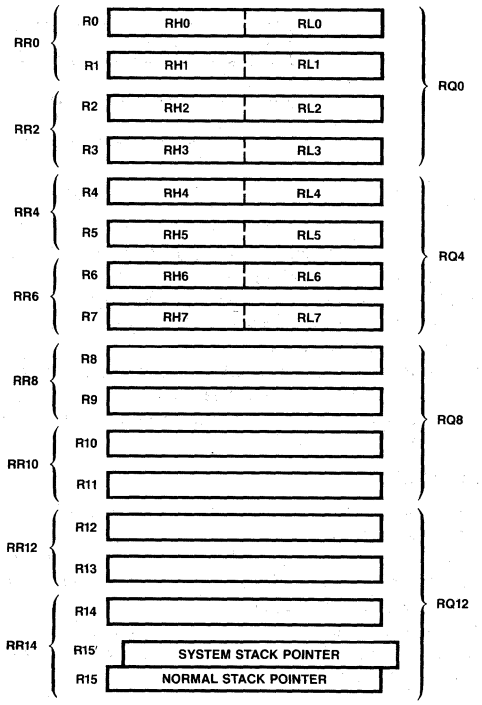


Figure 1. AmZ8002 General Registers.

MOS-227

STACK POINTER

The AmZ8002 architecture allows stacks to be maintained in the memory. Any general purpose register except R0 can be used as a stack pointer in stack manipulating instructions such as PUSH and POP. However, certain instructions such as subroutine call and return make implicit use of the register R15 as the stack pointer. Two implicit stacks are maintained – normal stack using R15 as the stack pointer and system stack using R15′ as the system stack pointer (see Figure 1). If the CPU is operating in the Normal Mode, R15 is active, and if the CPU is in System Mode R15′ will be used instead of R15. The implied stack pointer is a part of the general registers and hence can be manipulated using the instructions available for register operations.

PROCESSOR STATUS

The CPU status consists of the 16-bit Program Counter (PC) and the 16-bit Flag and Control Word (FCW) register (see Figure 2). The following is a brief description of the FCW bits.

S/$\overline{N}$: System/Normal – 1 indicates System Mode and 0 indicates Normal Mode.

VIE: Vectored Interrupt Enable – 1 indicates that Vectored Interrupt requests will be honored.

NVIE: Non-Vectored Interrupt Enable – 1 indicates that non-vectored interrupt requests will be honored.

C: Carry – 1 indicates that a carry has occurred from the most significant bit position when performing arithmetic operations.

Z: Zero – 1 indicates that the result of an operation is zero.

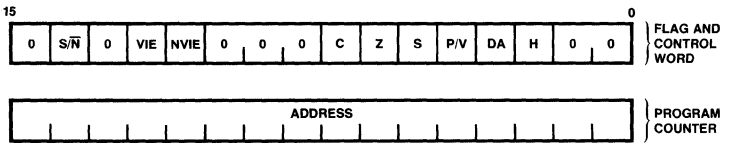


Figure 2. AmZ8002 Processor Status.

MOS-226

S: Sign – 1 indicates that the result of an operation is negative i.e., most significant bit is one.

P/V: Parity/Overflow – 1 indicates that there was an overflow during arithmetic operations. For logical operations this bit indicates parity of the result.

DA: Decimal Adjust – Records byte arithmetic operations.

H: Half Carry – 1 indicates that there was a carry from the most significant bit of the lower digit during byte arithmetic.

DATA TYPES

The AmZ8002 instructions operate on bits, digits (4 bits), bytes (8 bits), words (16 bits), long words (32 bits), byte strings and word strings type operands. Bits can be set, reset or tested. Digits are used to facilitate BCD arithmetic operations. Bytes are used for characters and small integers. Words are used for integer values and addresses while long words are used for large integer values. All operands except strings can reside either in memory or general registers. Strings can reside in memory only.

INTERRUPT AND TRAP STRUCTURE

Interrupt is defined as an external asynchronous event requiring program interruption. For example, interruption is caused by a peripheral needing service. Traps are synchronous events resulting from execution of certain instructions under some defined circumstances. Both interrupts and traps are handled in a similar manner by the AmZ8002.

The AmZ8002 supports three types of interrupts in order of descending priority – non-maskable, vectored and non-vectored. The vectored and non-vectored interrupts can be disabled by appropriate control bits in the FCW. The AmZ8002 has three traps – system call, unimplemented opcode and privileged instruction. The traps have higher priority than interrupts.

When an interrupt or trap occurs, the current program status is automatically pushed on to the system stack. The program status consists of processor status (i.e., PC and FCW) plus a 16-bit identifier. The identifier contains the reason, source and other coded information relating to the interrupt or trap.

After saving the current program status, the new processor status is automatically loaded from the new program status area located in the memory. This area is designated by the New Program Status Area Pointer (NPSAP) register. See AM-PUB086 publication for further details.

ADDRESSING MODES

Information contained in the AmZ8002 instructions consists of the operation to be performed, the operand type and the location of the operands. Operand locations are designated by general register addresses, memory addresses or I/O addresses. The addressing mode of a given instruction defines the address space referenced and the method to compute the operand address. Addressing modes are explicitly specified or implied in an instruction. Figure 3 illustrates the eight explicit addressing modes: Register (R), Immediate (IM), Indirect Register (IR), Direct Address (DA), Indexed (X), Relative Address (RA), Base Address (BA) and Base Indexed (BX).

INPUT/OUTPUT

A set of I/O instructions are provided to accomplish byte or word transfers between the AmZ8002 and I/O devices. I/O devices are addressed using 16-bit I/O port addresses and I/O address space is not a part of the memory address space. Two types of I/O instructions are provided; each with its own 16-bit address space. I/O instructions include a comprehensive set of In, Out and Block transfers.

CPU TIMING

The AmZ8002 accomplishes instruction execution by stepping through a pre-determined sequence of machine cycles, such as memory read, memory write, etc. Each machine cycle requires between three and ten clock cycles. Bus Requests by DMA devices are granted at machine cycle boundaries. No machine cycle is longer than ten clock cycles; thus assuring fast response to a Bus Request (assuming no extra wait states). The start of a machine cycle is always marked by a LOW pulse on the $\overline{AS}$ output. The status output lines ST0-ST3 indicate the nature of the current cycle in a coded form.

STATUS LINE CODES

Status line coding was listed in the table shown under ST0-ST3 outputs in the Interface Signal Description. The following is a detailed description of the status codes.

Internal Operation:

This status code indicates that the AmZ8002 is going through a machine cycle for its internal operation. Figure 4 depicts an internal operation cycle. It consists of three clock periods identified as T1, T2 and T3. The $\overline{AS}$ output will be activated with a LOW pulse by the AmZ8002 to mark the start of a machine cycle. The ST0-ST3 will reflect the code for the internal operation. The $\overline{MREQ}$, $\overline{DS}$ and R/$\overline{W}$ outputs will be HIGH. The N/$\overline{S}$ output will remain at the same level as in the previous machine cycle. The AmZ8002 will ignore the $\overline{WAIT}$ input during the internal operation cycle. The CPU will drive the AD0-AD15 bus with unspecified information during T1. However, the bus will go into high impedance during T2 and remain in that state for the remainder of the cycle. The B/$\overline{W}$ output is also activated by the CPU with unspecified information.

Memory Refresh:

This status code indicates that AmZ8002 is accessing the memory to refresh. The refresh cycle consists of three clock periods as depicted in Figure 5. The CPU will activate the $\overline{AS}$ output with a LOW pulse to mark the beginning of a machine cycle and ST0-ST3 outputs will reflect the refresh cycle code. The least significant 9 lines of the AD0-AD15 bus contain the refresh address. Because the memory is word organized, the AD0 will always be LOW. The most significant 7 bus lines are not specified. The $\overline{DS}$ output will remain HIGH for the entire cycle while R/$\overline{W}$, B/$\overline{W}$ and N/$\overline{S}$ outputs will remain at the same level as in the machine cycle prior to refresh. The AD0-AD15 bus will go into high impedance state during T2 period and remain there for

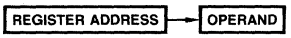
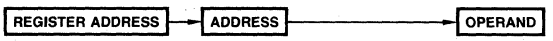
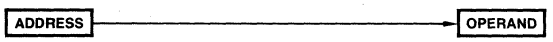
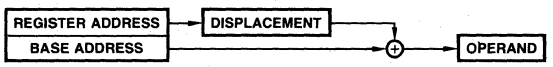
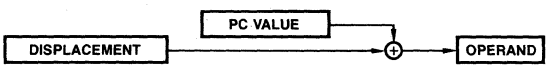
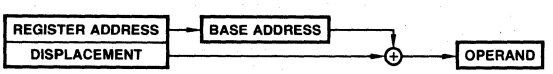
Mode	Operand Addressing			Operand Value
	In the Instruction	In a Register	In Memory	
Register				The content of the register.
Immediate				In the instruction
Indirect Register				The content of the location whose address is in the register.
Direct Address				The content of the location whose address is in the instruction.
Index				The content of the location whose address is the address in the instruction, offset by the content of the working register.
Relative Address				The content of the location whose address is the content of the program counter, offset by the displacement in the instruction.
Base Address				The content of the location whose address is the address in the register, offset by the displacement in the instruction.
Base Index				The content of the location whose address is the address in the register, offset by the displacement in the register.

Figure 3. Addressing Modes.

MOS-229

the remainder of the cycle. The AmZ8002 will activate the $\overline{\text{MREQ}}$ output LOW during the refresh cycle. It should be noted that $\overline{\text{WAIT}}$ input is ignored by the CPU for refresh operations.

I/O Transactions:

There are two status line codes used for I/O transaction cycles. The AmZ8002 provides two separate I/O spaces and two types of instructions called Normal I/O and Special I/O. Each I/O space is addressed by a 16-bit address called port address. The timing for both types of I/O transactions is essentially identical. A typical I/O cycle consists of four clock periods T1, T2, TWA and T3 as shown in Figure 6. The TWA is the wait state; insertion of one wait state for an I/O cycle is always automatic. Additional wait cycles can be inserted by LOW on the $\overline{\text{WAIT}}$ input. The $\overline{\text{WAIT}}$ input is sampled during every TW state. If this input is LOW, one more wait state will be inserted. Insertion of wait states continues until $\overline{\text{WAIT}}$ input is HIGH. T3 state will follow the last wait state to complete the I/O cycle.

During I/O cycles the ST0-ST3 outputs will reflect appropriate code depending on the type of instruction being executed (Normal I/O or Special I/O). $\overline{\text{AS}}$ output will be pulsed LOW to mark the beginning of the cycle. The CPU drives the AD0-AD15 bus with the 16-bit port address specified by the current instruction. The N/$\overline{\text{S}}$ output will be LOW indicating that CPU is operating in the system mode. It should be recalled that the N/$\overline{\text{S}}$ output is derived from the appropriate bit in the FCW register. All I/O instructions are privileged instructions and will be allowed to

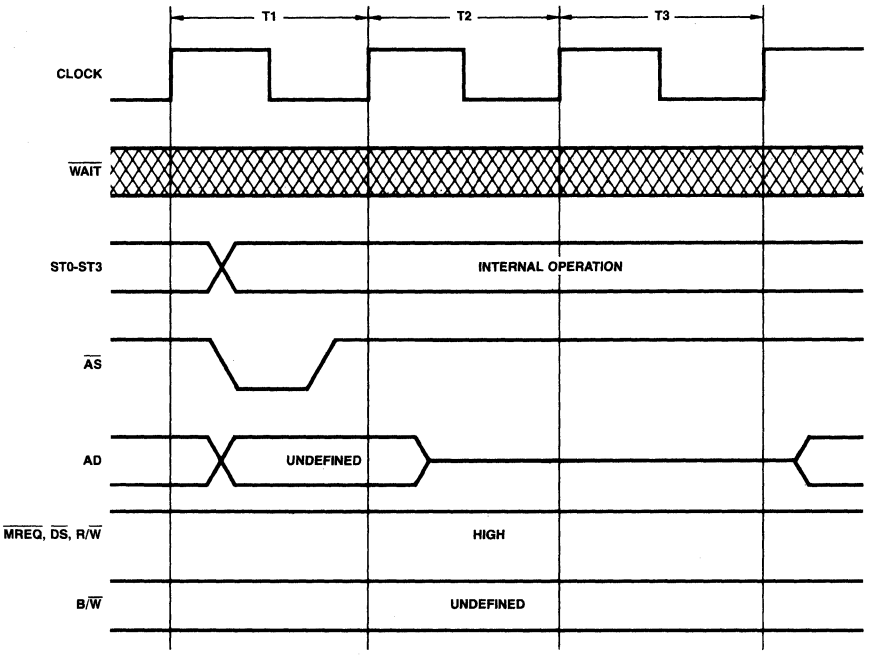


Figure 4. Internal Operation Cycle.

MOS-230

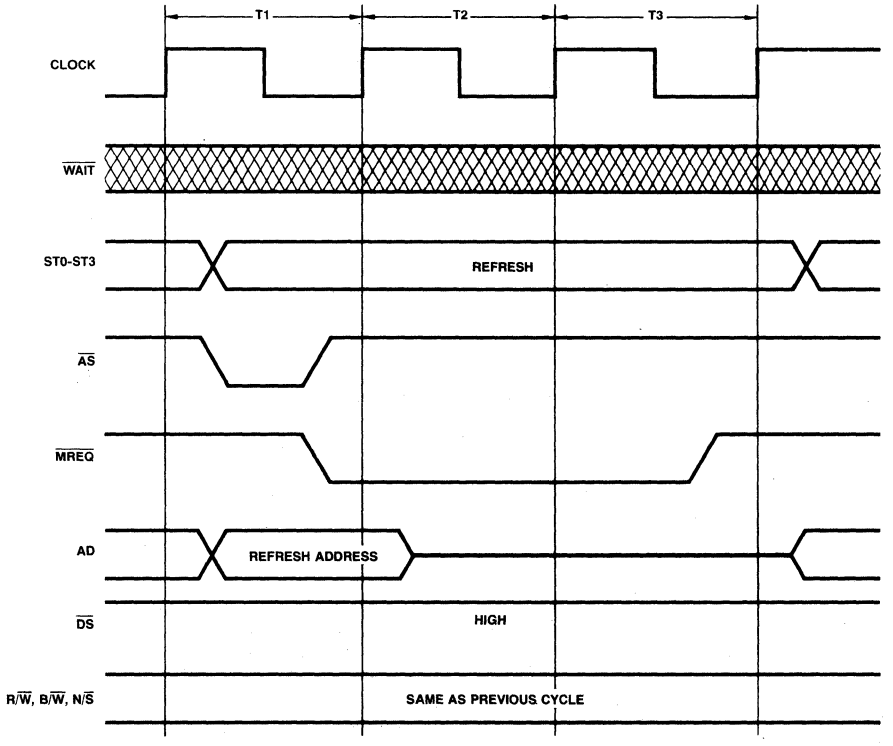


Figure 5. Refresh Cycle.

MOS-231

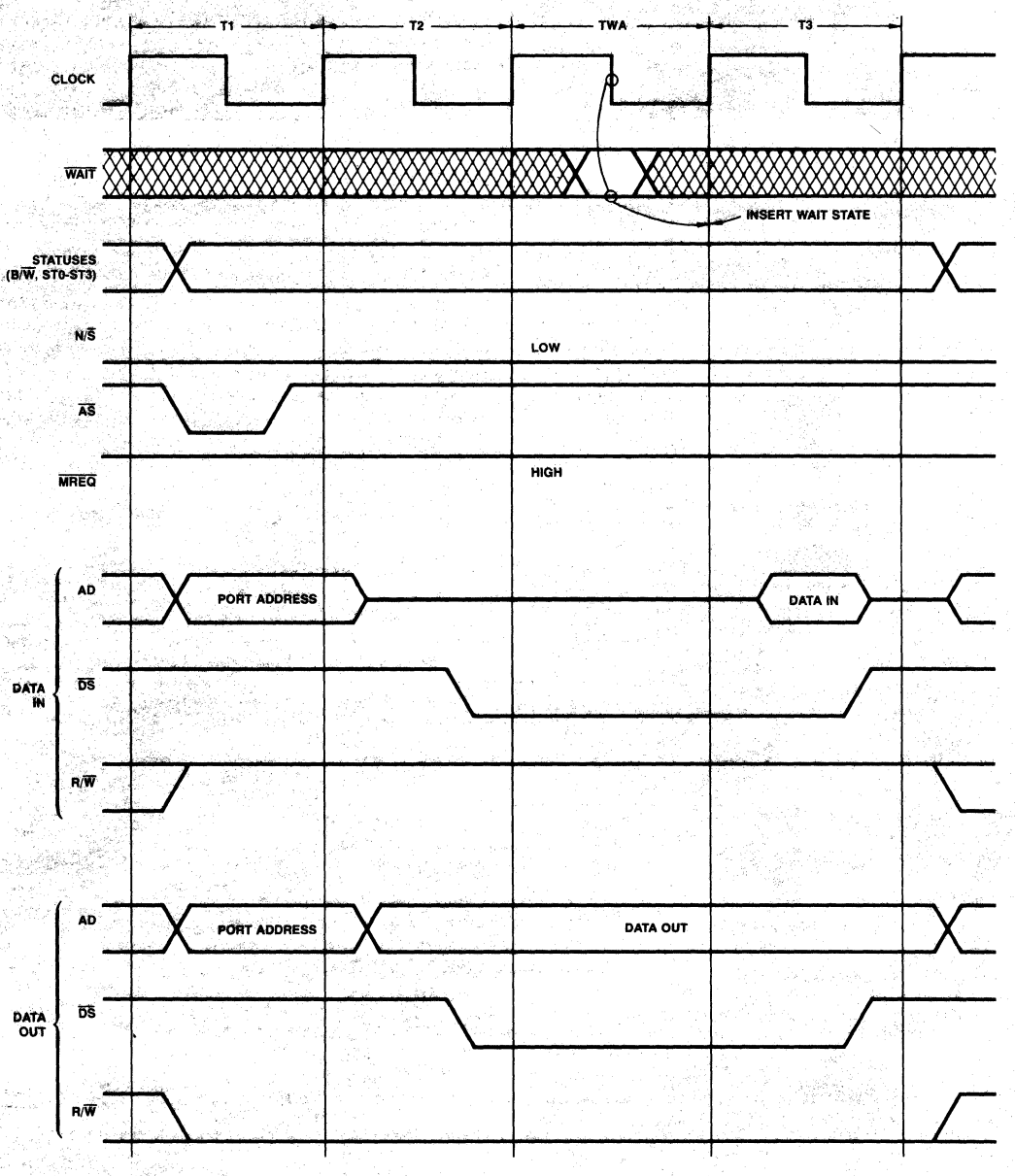


Figure 6. AmZ8002 I/O Cycle.

MOS-232

execute only if the FCW specifies system mode operation. The MREQ output will be HIGH. The AmZ8002 I/O instructions provide both word or byte transactions. The B/W output will be HIGH or LOW depending whether the instruction specifies a byte or word transfer.

Two kinds of I/O transfers should be considered: Data In means reading from the device and Data Out means writing into the device. For In operations, the R/W output will be HIGH. The AD0-AD15 bus will go into high impedance state during T2. During byte input instructions, the CPU reads either the even or odd half of the Data Bus, dependent upon the port address. If the port address is even, the most significant half of the Data Bus is read. If the port address is odd, the least significant half of the Data Bus is read. During word input instructions, the CPU reads all 16 bits of the Data Bus. The AmZ8002 will drive the DS output LOW to signal to the device that data can be gated on to the bus. The CPU will accept the data during T3 and DS output will go HIGH signalling the end of an I/O transaction.

For Data Out, the R/W output will be LOW. The AmZ8002 will provide data on the AD0-AD15 bus and activates the DS output LOW during T2. During byte output instructions, the CPU duplicates the byte data onto both the high and low halves of the Data Bus and external logic, using A0, enables the appropriate byte port. During word output instructions the CPU outputs data onto all 16 bits of the Data Bus. The DS output goes HIGH during T3 and the cycle is complete.

Memory Transactions:

There are four status line codes that indicate a memory transaction:

a) Memory transaction to read or write an operand
b) Memory transaction to read from or write into the stack
c) Memory transaction to fetch the first word of an instruction (sometimes called IF1)
d) Memory transaction to fetch the subsequent word of an instruction (sometimes called IFN).

It can be appreciated that all the above transactions essentially fall into two categories: memory read and memory write. In the case of IF1 and IFN cycles, the memory will be read at the address supplied by the program counter. All AmZ8002 instructions are multiples of 16-bit words. Words are always addressed by an even address. Thus IF1 and IFN cycles involve performing a memory read for words. On the other hand, a memory transaction for operand and stack operation could be a read or write. Moreover, an operand could be a word or a byte. For stack operation involving the implied stack pointer the address will be supplied by the R15 (or R15'). For operand transactions, the memory address will come from several sources depending on the instruction and the addressing mode. Memory transaction cycle timing is shown in Figure 7. It typically consists of three clock periods T1, T2 and T3. Wait states (TW) can be inserted between T2 and T3 by activating the $\overline{WAIT}$ input LOW. The $\overline{WAIT}$ input will be sampled during T2 and during every subsequent TW. The ST0-ST3 outputs will reflect the appropriate code for the current cycle early in T1 and the $\overline{AS}$ output will be pulsed LOW to mark the beginning of the cycle. The N/$\overline{S}$ output will indicate whether the normal or system address space will be used for the current cycle. As shown in the figure the $\overline{MREQ}$ output will go LOW during T1 to indicate a memory operation.

Consider a read operation first. The R/$\overline{W}$ output will be HIGH. The AmZ8002 will drive the AD0-AD15 with the appropriate address early in T1. During T2, the bus will go into high impedance state and $\overline{DS}$ output will be activated LOW by the CPU. The data can be gated on to the bus when $\overline{DS}$ is LOW. During T1 the B/$\overline{W}$ will also be activated to indicate byte or word will be transacted. The AmZ8002 memory is word organized and words are addressed by even addresses. However, when addressing bytes, the memory address may be odd or even; an even address for

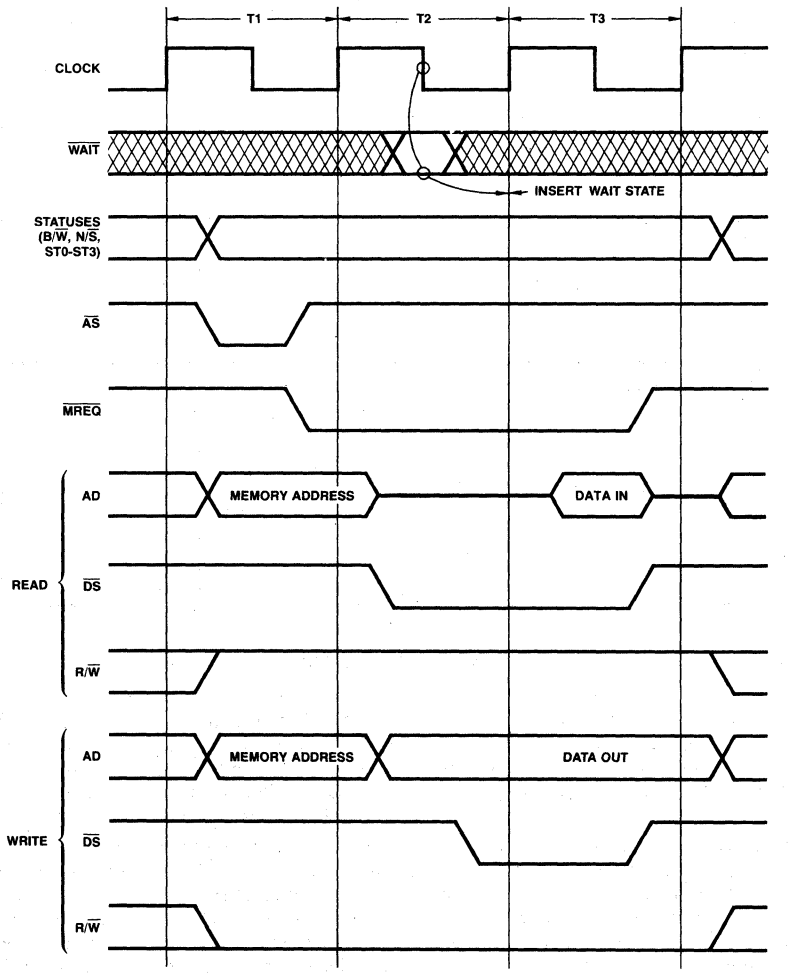


Figure 7. Memory Transactions.

MOS-233

most significant byte of a word and the next odd address for the least significant byte of that word. When reading a byte from the memory, the least significant address bit can be ignored and the whole word containing the desired byte is gated on to the bus. The CPU will pick the appropriate byte automatically. The AmZ8002 will drive the $\overline{DS}$ output HIGH indicating data acceptance.

Consider the write operation next. The R/$\overline{W}$ output will be LOW. The AmZ8002 removes the address and gates out the data to be written on the bus and activates the $\overline{DS}$ output LOW during T2. If the data to be written is a byte then the same byte will be on both halves of the bus. The $\overline{DS}$ output will go HIGH during T3 signifying completion of the cycle.

Interrupt Acknowledge:

There are three status line codes devoted to interrupt acknowledgement. These correspond to non-maskable, vectored and non-vectored interrupts. The Interrupt Acknowledge cycle is illustrated in Figure 8. The $\overline{NMI}$ input of the AmZ8002 is edge detected i.e., a HIGH to LOW input level change is stored in an internal latch. Similar internal storage is not provided for the $\overline{VI}$ and $\overline{NVI}$ inputs. For $\overline{VI}$ and $\overline{NVI}$ inputs to cause an interruption, the corresponding interrupt enable bits in the FCW must be 1. For the following discussion, both the VIE and NVIE bits in the FCW are assumed to be 1.

As shown in the figure, the $\overline{VI}$ input, $\overline{NVI}$ input and the internal $\overline{NMI}$ latch output are sampled during T3 of the last machine cycle of an instruction.

A LOW on these signals triggers the corresponding interrupt acknowledge sequence described below. The AmZ8002 executes a dummy IF1 cycle prior to entering the actual acknowledge cycle (see memory transactions for IF1 cycle description). During this dummy IF1 cycle, the program counter is not updated; instead the implied system stack pointer (R15′) will be decremented. Following the dummy IF1 cycle is the actual interrupt acknowledge cycle.

The interrupt acknowledge cycle typically consists of 10 clock periods; T1 through T5 and five automatic TW (wait) states. As usual, the $\overline{AS}$ output will be pulsed LOW during T1 to mark the beginning of a cycle. The ST0-ST3 outputs will reflect the appropriate interrupt acknowledge code, the $\overline{MREQ}$ output will be HIGH, the N/$\overline{S}$ output remains the same as in the preceding cycle, the R/$\overline{W}$ output will be HIGH and the B/W output will be LOW. The AmZ8002 will drive the AD0-AD15 bus with unspecified information during T1 and the bus will go into the high impedance state during T2. Three TWA states will automatically follow T2. The WAIT input will be sampled during the third TWA state.

If LOW, an extra TW state will be inserted and the $\overline{WAIT}$ will be sampled again during TW. Such insertion of TW states continues until the $\overline{WAIT}$ input is HIGH. After the last TW state, the $\overline{DS}$ output will go LOW and two more automatic wait states follow. The interrupting device can gate up to a 16-bit identifier on to the bus when the $\overline{DS}$ output is LOW. The $\overline{WAIT}$ input will be sampled again during the last TWA state. If the $\overline{WAIT}$ input is LOW one TW state will be inserted and the $\overline{WAIT}$ will be sampled during TW. Such TW insertion continues until the $\overline{WAIT}$

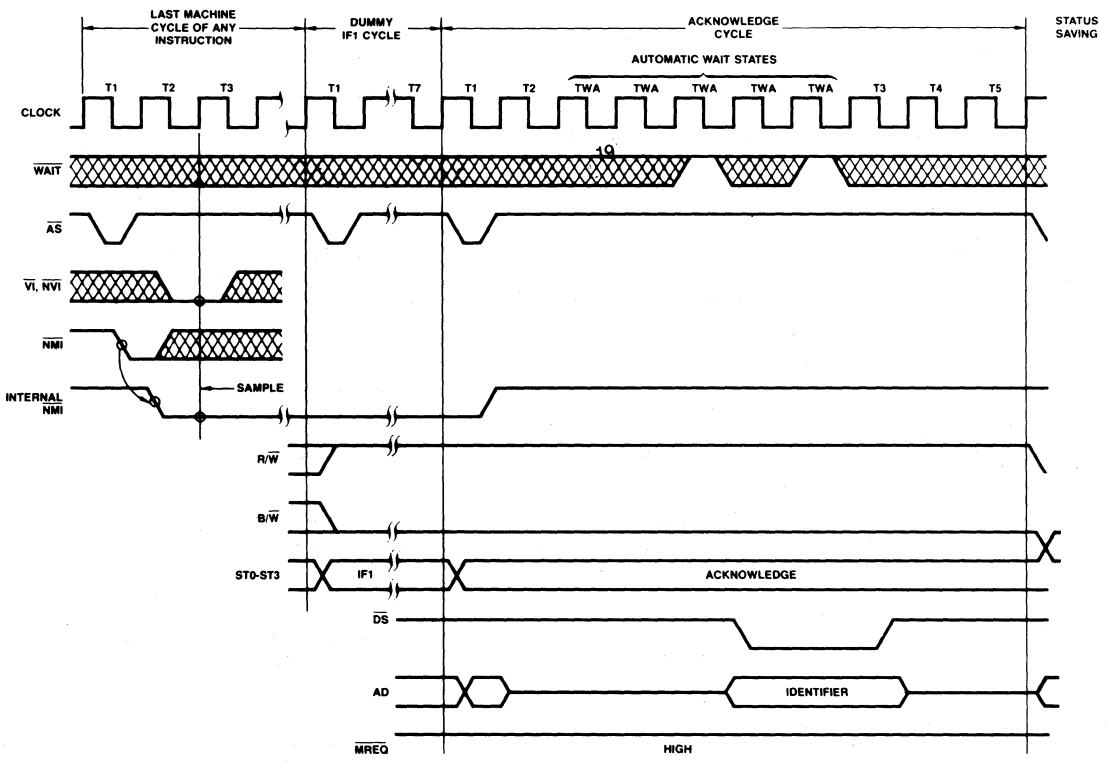


Figure 8. Interrupt Acknowledge Cycle.

MOS-234

input is HIGH. After completing the last TW state T3 will be entered and the $\overline{DS}$ output will go HIGH. The interrupting device should remove the identifier and cease driving the bus. T4 and T5 states will follow T3 to complete the cycle. Following the interrupt acknowledge cycle will be memory transaction cycles. to save the status on the stack. Note that the N/$\overline{S}$ output will be automatically LOW during status saving.

The internal $\overline{NMI}$ latch will be reset to the initial state at $\overline{AS}$ going HIGH in the interrupt acknowledge cycle. The $\overline{VI}$ and $\overline{NVI}$ inputs should be kept LOW until this time also.

Status Saving Sequence:

The machine cycles following the interrupt acknowledge cycle push the old status information on the system stack in the following order: the 16-bit program counter; the flag and control word; and finally the interrupt/trap indentifier. Subsequent machine cycles fetch the new program status from the new program status area, and then branch to the interrupt/service routine.

BUS REQUEST/BUS ACKNOWLEDGE TIMING:

A LOW on the $\overline{BUSRQ}$ input is an indication to the AmZ8002 that another device (such as DMA) is requesting control of the bus. The $\overline{BUSRQ}$ input is synchronized internally at T1 of any machine cycle. (See below for exception.) The $\overline{BUSAK}$ will go LOW after the last clock period of the machine cycle. The LOW on the $\overline{BUSAK}$ output indicates acknowledgement. When $\overline{BUSAK}$ is LOW the following outputs will go into the high impedance state; AD0-AD15, $\overline{AS}$, $\overline{DS}$, $\overline{MREQ}$, ST0-ST3, B/$\overline{W}$, R/$\overline{W}$ and N/$\overline{S}$. The $\overline{BUSRQ}$ must be held LOW until all transactions are completed. When $\overline{BUSRQ}$ goes HIGH, it is synchronized internally, the $\overline{BUSAK}$ output will go HIGH and normal CPU operation will resume. Figure 9 illustrates the $\overline{BUSRQ}$/$\overline{BUSAK}$ timing.

It was mentioned that $\overline{BUSRQ}$ will be honored during any machine cycle with one exception. This exception is during the execution of TSET/TSETB instructions. $\overline{BUSRQ}$ will not be honored once execution of these instructions has started.

SINGLE STEPPING

The $\overline{STOP}$ input of the AmZ8002 facilitates one instruction at a time or single step operation. Figure 10 illustrates $\overline{STOP}$ input timing. The $\overline{STOP}$ input is sampled on the HIGH to LOW transition of the clock input that immediately precedes an IF1 cycle. If the $\overline{STOP}$ is found LOW, AmZ8002 introduces a memory refresh cycle after T3. Moreover, $\overline{STOP}$ input will be sampled again at T3. If $\overline{STOP}$ is LOW one more refresh cycle will follow the previous refresh cycle. The $\overline{STOP}$ will be sampled during T3 of the refresh cycle also. One additional refresh cycle will be added every time $\overline{STOP}$ input is sampled LOW. After completing the last refresh cycle which will occur after $\overline{STOP}$ is HIGH, the CPU will insert two dummy states T4 and T5 to complete the IF1 cycle and resume its normal operations for executing the instruction. See appropriate sections on memory transactions and memory refresh.

It should be noted that refresh cycles will occur in the stop mode even if the refresh facility is disabled in the refresh register.

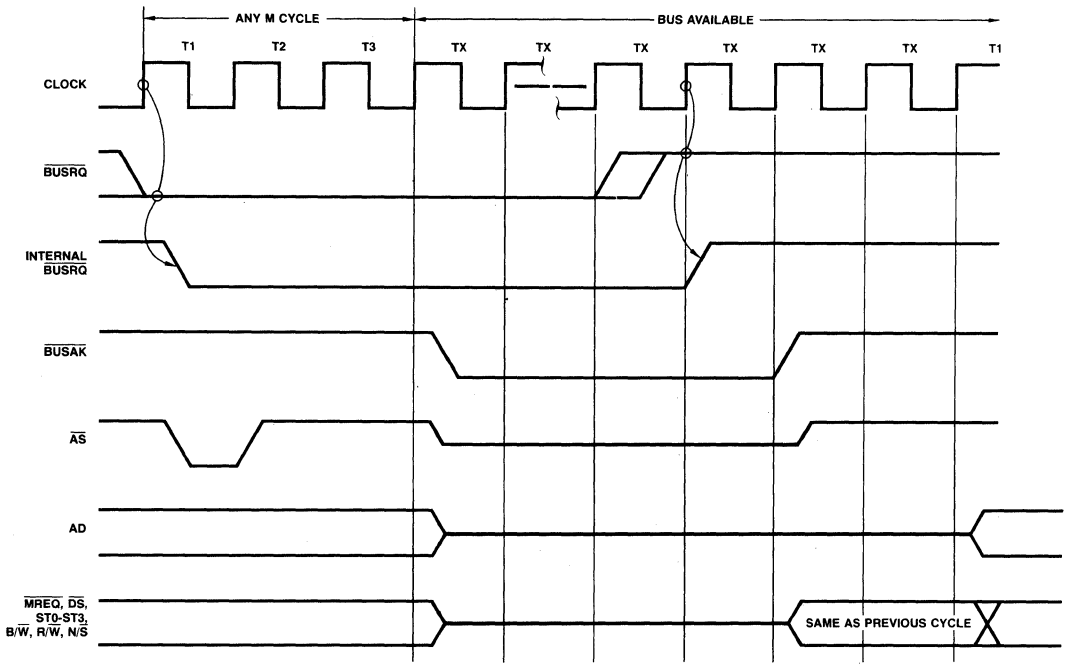


Figure 9. Bus Request/Acknowledge Cycle.

MOS-235

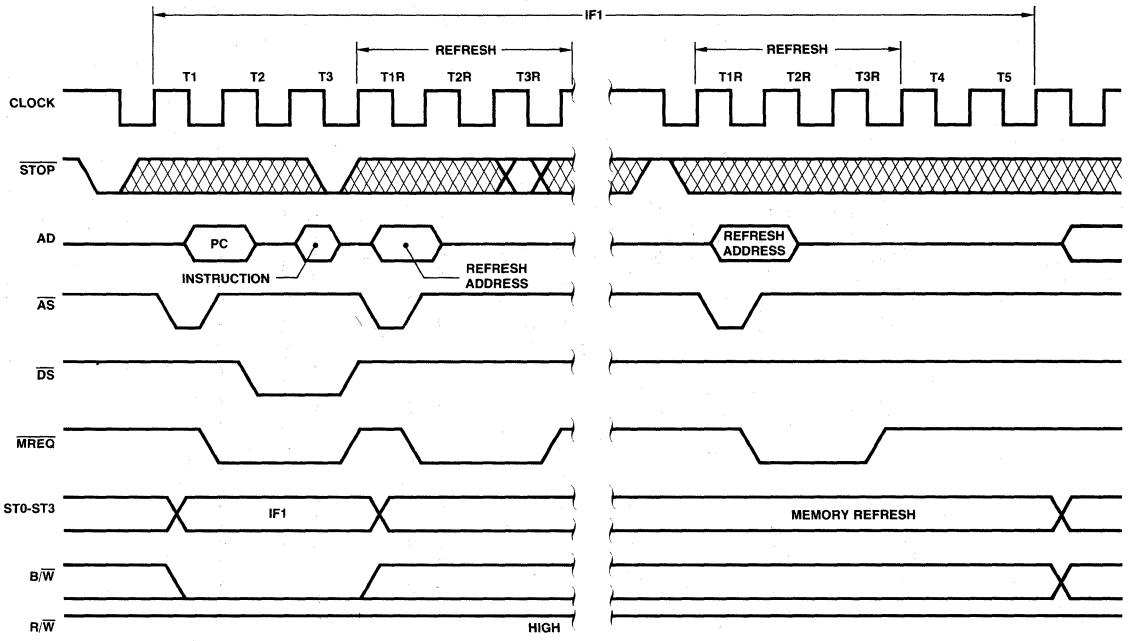


Figure 10. Single Step Timing.

MOS-236

MULTIMICROPROCESSOR FACILITIES

The AmZ8002 is provided with hardware and software facilities to support multiple microprocessor systems. The $\overline{\mu O}$ and $\overline{\mu I}$ signals of the AmZ8002 are used in conjunction with the MBIT, MREQ, MRES and MSET instructions for this purpose. The $\overline{\mu O}$ output can be activated LOW by using an appropriate instruction to signal a request from the AmZ8002 for a resource. The $\overline{\mu I}$ input is tested by the AmZ8002 before activating the $\overline{\mu O}$ output. LOW at the $\overline{\mu I}$ input at this time indicates that the resource is busy. The AmZ8002 can examine the $\overline{\mu I}$ input after activating the $\overline{\mu O}$ output LOW. The $\overline{\mu I}$ will be tested again to see if the requested resource became available. For detailed information on the Multimicroprocessor facilities the AmZ8001/AmZ8002 Processor Interface Manual (Publication No. AM-PUB089) should be consulted.

INITIALIZATION

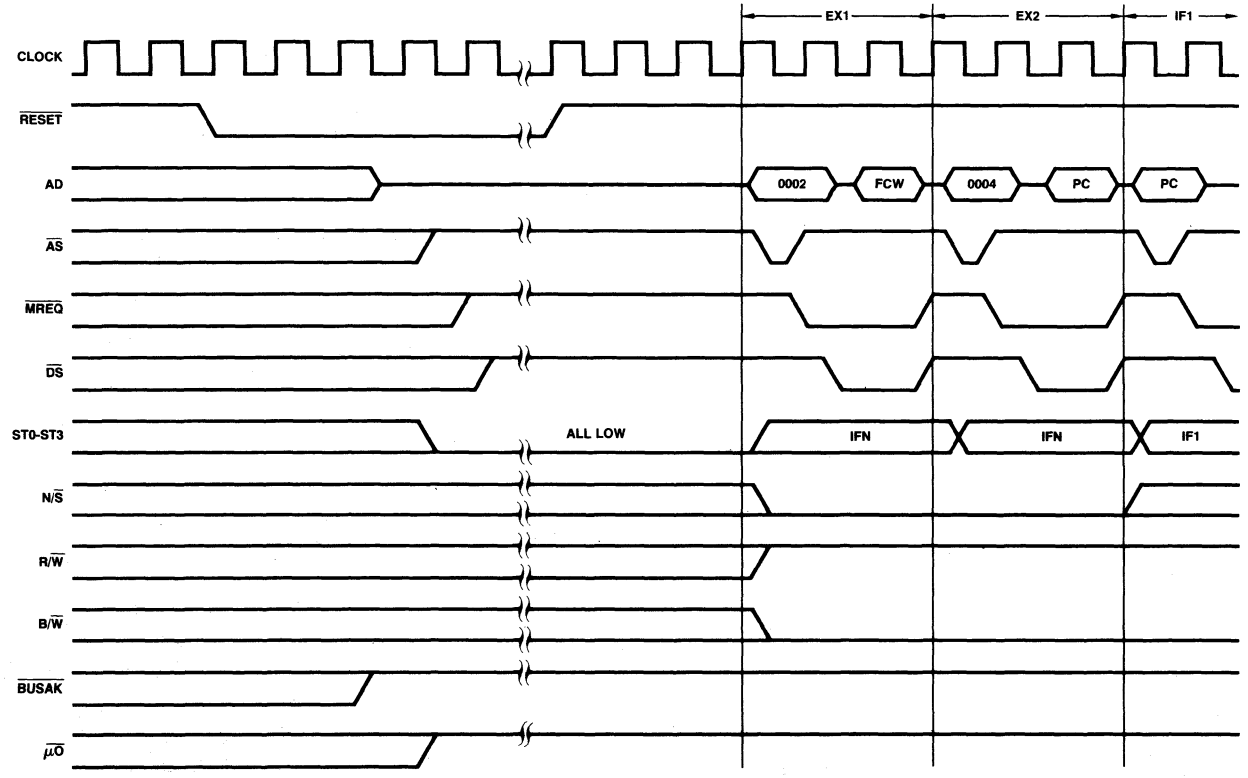
A LOW on the $\overline{\text{Reset}}$ input starts the CPU initialization. The initialization sequence is shown in Figure 11. Within five clock periods after the HIGH to LOW level change of the $\overline{\text{Reset}}$ input the following will occur:

a) AD0-AD15 bus will be in the HIGH impedance state
b) $\overline{\text{AS}}$, $\overline{\text{DS}}$, $\overline{\text{MREQ}}$, $\overline{\text{BUSAK}}$ and $\overline{\mu O}$ outputs will be HIGH
c) ST0-ST3 outputs will be LOW
d) Refresh will be disabled
e) R/$\overline{\text{W}}$, B/$\overline{\text{W}}$ and N/$\overline{\text{S}}$ outputs are not affected. For a power on reset the state of these outputs is not specified.

After the $\overline{\text{Reset}}$ input returns HIGH and remains HIGH for three clock periods, two 16-bit memory read operations will be performed as follows. Note that the N/$\overline{\text{S}}$ output will be LOW and ST0-ST3 outputs will reflect IFN code.

a) The contents of the memory location 0002 will be read. This information will be loaded into the FCW of the AmZ8002.
b) The contents of the memory location 0004 will be read. This information will be loaded into the AmZ8002 program counter.

This completes initialization sequence and an IF1 cycle will follow to fetch the first instruction to begin program execution. See the section on memory transactions for timing.



MOS-237

Figure 11. Reset Sequence.

AmZ8002 INSTRUCTION SET

LOAD AND EXCHANGE

Mnemonics	Operands	Addr. Modes	Operation
CLR CLRB	dst	R IR DA X	Clear dst ← 0
EX EXB	R, src	R IR DA X	Exchange R ← src
LD LDB LDL	R, src	R IM IM IR DA X BA BX	Load into Register R ← src
LD LDB LDL	dst, R	IR DA X BA BX	Load into Memory (Store) dst ← R
LD LDB	dst, IM	IR DA X	Load Immediate into Memory dst ← IM
LDA	R, src	DA X BA BX	Load Address R ← source address
LDAR	R, src	RA	Load Address Relative R ← source address
LDK	R, src	IM	Load Constant R ← n (n = 0 . . . 15)
LDM	R, src, n	IR DA X	Load Multiple R ← src (n consecutive words) (n = 1 . . . 16)
LDM	dst, R, n	IR DA X	Load Multiple (Store Multiple) dst ← R (n consecutive words) (n = 1 . . . 16)
LDR LDRB LDRL	R, src	RA	Load Relative R ← src (range −32768 . . . +32767)
LDR LDRB LDRL	dst, R	RA	Load Relative (Store Relative) dst ← R (range −32768 . . . +32767)
POP POPL	dst, R	R IR DA X	Pop dst ← IR Autoincrement contents of R
PUSH PUSHL	IR, src	R IM IR DA X	Push Autodecrement contents of R IR ← src

ARITHMETIC

Mnemonics	Operands	Addr. Modes	Operation
ADC ADCB	R, src	R	Add with Carry R ← R + src + carry
ADD ADDB ADDL	R, src	R IM IR DA X	Add R ← R + src
CP CPB CPL	R, src	R IM IR DA X	Compare with Register R − src
CP CPB	dst, IM	IR DA X	Compare with Immediate dst − IM
DAB	dst	R	Decimal Adjust
DEC DECB	dst, n	R IR DA X	Decrement by n dst ← dst − n (n = 1 . . . 16)
DIV DIVL	R, src	R IM IR DA X	Divide (signed) Word: $R_{n+1} \leftarrow R_{n,n+1} \div$ src $R_n \leftarrow$ remainder Long Word: $R_{n+2,n+3} \leftarrow R_{n...n+3} \div$ src $R_{n,n+1} \leftarrow$ remainder
EXTS EXTSB EXTSL	dst	R	Extend Sign Extend sign of low order half of st through high order half of dst
INC INCB	dst, n	R IR DA X	Increment by n dst ← dst + n (n = 1 . . . 16)
MULT MULTL	R, src	R IM IR DA X	Multiply (signed) Word: $R_{n,n+1} \leftarrow R_{n+1} \cdot$ src Long Word: $R_{n...n+3} \leftarrow R_{n+2,n+3} \cdot$ src *Plus seven cycles for each 1 in the multiplicand
NEG NEGB	dst	R IR DA X	Negate dst ← 0 − dst
SBC SBCB	R, src	R	Subtract with Carry R ← R − src − carry
SUB SUBB SUBL	R,src	R IM IR DA X	Subtract R ← R − src

LOGICAL

Mnemonics	Operands	Addr. Modes	Operation
AND ANDB	R, src	R IM IR DA X	AND R ← R AND src
COM COMB	dst	R IM IR DA X	Complement dst ← NOT dst
OR ORB	R, src	R IM IR DA X	OR R ← R OR src
TEST TESTB TESTL	dst	R IR DA X	TEST dst OR 0
TCC TCCB	cc, dst	R	Test Condition Code Set LSB if cc is true
XOR XORB	R, src	R IM IR DA X	Exclusive OR R ← R XOR src

PROGRAM CONTROL

Mnemonics	Operands	Addr. Modes	Operation
CALL	dst	IR DA X	Call Subroutine Autodecrement SP @ SP ← PC PC ← dst
CALR	dst	RA	Call Relative Autodecrement SP @ SP ← PC PC ← PC + dst (range −4094 to +4096)
DJNZ DBJNZ	R, dst	RA	Decrement and Jump if Non-Zero R ← R − 1 IF R = 0: PC ← PC + dst (range −254 to 0)
IRET*	–	–	Interrupt Return PS ← @ SP Autoincrement SP
JP	cc, dst	IR IR DA X	Jump Conditional If cc is true: PC ← dst
JR	cc, dst	RA	Jump Conditional Relative If cc is true: PC ← PC + dst (range −256 to +254)
RET	cc	–	Return Conditional If cc is true: PC ← @ SP Autodecrement SP
SC	src	IM	System Call Autodecrement SP @ SP ← old PS Push instruction PS ← System Call PS

*Privileged instructions. Executed in system mode only.

BIT MANIPULATION

Mnemonics	Operand	Addr. Modes	Operation
BIT BITB	dst, b	R IR DA X	Test Bit Static Z flag ← NOT dst bit specified by b
BIT BITB	dst, R	R	Test Bit Dynamic Z flag ← NOT dst bit specified by contents of R
RES RESB	dst, b	R IR DA X	Reset Bit Static Reset dst bit specified by b
RES RESB	dst, R	R	Reset Bit Dynamic Reset dst bit specified by contents of R
SET SETB	dst, b	R IR DA X	Set Bit Static Set dst bit specified by b
SET SETB	dst, R	R	Set Bit Dynamic Set dst bit specified by contents of R
TSET TSETB	dst	R IR DA X	Test and Set S flag ← MSB of dst dst ← all 1s

ROTATE AND SHIFT

Mnemonics	Operand	Addr. Modes	Operation
RLDB	R, src	R	Rotate Digit Left
RRDB	R, src	R	Rotate Digit Right
RL RLB	dst, n	R R	Rotate Left by n bits (n = 1, 2)
RLC RLCB	dst, n	R R	Rotate Left through Carry by n bits (n = 1, 2)
RR RRB	dst, n	R R	Rotate Right by n bits (n = 1, 2)
RRC RRCB	dst, n	R R	Rotate Right through Carry by n bits (n = 1, 2)
SDA SDAB SDAL	dst, R	R	Shift Dynamic Arithmetic Shift dst left or right by contents of R
SDL SDLB SDLL	dst, R	R	Shift Dynamic Logical Shift dst left or right by contents of R
SLA SLAB SLAL	dst, n	R	Shift Left Arithmetic by n bits
SLL SLLB SLLL	dst, n	R	Shift Left Logical by n bits
SRA SRAB SRAL	dst, n	R	Shift Right Arithmetic by n bits
SRL SRLB SRLL	dst, n	R	Shift Right Logical by n bits

BLOCK TRANSFER AND STRING MANIPULATION

Mnemonics	Operands	Addr. Modes	Operation
CPD CPDB	R_X, src, R_Y, cc	IR	Compare and Decrement R_X − src Autodecrement src address $R_Y \leftarrow R_Y - 1$
CPDR CPDRB	R_X, src, R_Y, cc	IR	Compare, Decrement and Repeat R_X − src Autodecrement src address $R_Y \leftarrow R_Y - 1$ Repeat until cc is true or $R_Y = 0$
CPI CPIB	R_X, src, R_Y, cc	IR	Compare and Increment R_X − src Autoincrement src address $R_Y \leftarrow R_Y - 1$
CPIR CPIRB	R_X, src, R_Y, cc	IR	Compare, Increment and Repeat R_X − src Autoincrement src address $R_Y \leftarrow R_Y - 1$ Repeat until cc is true or $R_Y = 0$
CPSD CPSDB	dst, src, R, cc	IR	Compare String and Decrement dst − src Autodecrement dst and src addresses R ← R − 1
CPSDR CPSDRB	dst, src, R, cc	IR	Compare String, Decr. and Repeat dst − src Autodecrement dst and src addresses R ← R − 1 Repeat until cc is true or R = 0
CPSI CPSIB	dst, src, R, cc	IR	Compare String and Increment dst − src Autoincrement dst and src addresses R ← R − 1
CPSIR CPSIRB	dst, src, R, cc	IR	Compare String, Incr. and Repeat dst − src Autoincrement dst and src addresses R ← R − 1 Repeat until cc is true or R = 0
LDD LDDB	dst, src, R	IR	Load and Decrement dst ← src Autodecrement dst and src addresses R ← R − 1
LDDR LDDRB	dst, src, R	IR	Load, Decrement and Repeat dst ← src Autodecrement dst and src addresses R ← R − 1 Repeat until R = 0

BLOCK TRANSFER AND STRING MANIPULATION (Cont.)

Mnemonics	Operands	Addr. Modes	Operation
LDI LDIB	dst, src, R	IR	Load and Increment dst ← src Autoincrement dst and src addresses R ← R − 1
LDIR LDIRB	dst, src, R	IR	Load, Increment and Repeat dst ← src Autoincrement dst and src addresses R ← R − 1 Repeat until R = 0
TRDB	dst, src, R	IR	Translate and Decrement dst ← src (dst) Autodecrement dst address R ← R − 1
TRDRB	dst, src, R	IR	Translate, Decrement and Repeat dst ← src (dst) Autodecrement dst address R ← R − 1 Repeat until R = 0
TRIB	dst, src, R	IR	Translate and Increment dst ← src (dst) Autoincrement dst address R ← R − 1
TRIRB	dst, src, R	IR	Translate, Increment and Repeat dst ← src (dst) Autoincrement dst address R ← R − 1 Repeat until R = 0
TRTDB	src 1, src 2, R	IR	Translate and Test, Decrement RH1 ← src 2 (src 1) Autodecrement src 1 address R ← R − 1
TRTDRB	src 1, src 2, R	IR	Translate and Test, Decrement and Repeat RH1 ← src 2 (src 1) Autodecrement src 1 address R ← R − 1 Repeat until R = 0 or RH1 = 0
TRTIB	src 1, src 2, R	IR	Translate and Test, Increment RH1 ← src 2 (src 1) Autoincrement src 1 address R ← R − 1
TRTIRB	src 1, src 2, R	IR	Translate and Test, Increment and Repeat RH1 ← src 2 (src 1) Autoincrement src 1 address R ← R − 1 Repeat until R = 0 or RH1 = 0

INPUT/OUTPUT

Mnemonics	Operands	Addr. Modes	Operation
IN* INB*	R, src	IR DA	Input R ← src
IND* INDB*	dst, src, R	IR	Input and Decrement dst ← src Autodecrement dst address R ← R − 1
INDR* INDRB*	dst, src, R	IR	Input, Decrement and Repeat dst ← src Autodecrement dst address R ← R − 1 Repeat until R = 0
INI* INIB*	dst, src, R	IR	Input and Increment dst ← src Autoincrement dst address R ← R − 1
INIR* INIRB*	dst, src, R	IR	Input, Increment and Repeat dst ← src Autoincrement dst address R ← R − 1 Repeat until R = 0
OUT* OUTB*	dst, R	IR DA	Output dst ← R
OUTD* OUTDB*	dst, src, R	IR	Output and Decrement dst ← src Autodecrement src address R ← R − 1
OTDR* OTDRB*	dst, src, R	IR	Output, Decrement and Repeat dst ← src Autodecrement src address R ← R − 1 Repeat until R = 0
OUTI* OUTIB*	dst, src, R	IR	Output and Increment dst ← src Autoincrement src address R ← R − 1
OTIR* OTIRB*	dst, src, R	IR	Ouput, Increment and Repeat dst ← src Autoincrement src address R ← R − 1 Repeat until R = 0
SIN* SINB*	R, src	DA	Special Input R ← src
SIND* SINDB*	dst, src, R	IR	Special Input and Decrement dst ← src Autodecrement dst address R ← R − 1
SINDR* SINDRB*	dst, src, R	IR	Special Input, Decr. and Repeat dst ← src Autodecrement dst address R ← R − 1 Repeat until R = 0
SINI* SINIB*	dst, src, R	IR	Special Input and Increment dst ← src Autoincrement dst address R ← R − 1
SINIR* SINIRB*	dst, src, R	IR	Special Input, Incr. and Repeat dst ← src Autoincrement dst address R ← R − 1 Repeat until R = 0

INPUT/OUTPUT (Cont.)

Mnemonics	Operands	Addr. Modes	Operation
SOUT* SOUTB*	dst, src	DA	Special Output dst ← src
SOUTD* SOUTDB*	dst, src, R	IR	Special Output and Decrement dst ← src Autodecrement src address R ← R − 1
SOTDR* SOTDRB*	dst, src, R	IR	Special Output, Decr. and Repeat dst ← src Autodecrement src address R ← R − 1 Repeat until R = 0
SOUTI* SOUTIB*	dst, src, R	IR	Special Output and Increment dst ← src Autoincrement src address R ← R − 1
SOTIR* SOTIRB*	dst, src, R	R	Special Output, Incr. and Repeat dst ← src Autoincrement src address R ← R − 1 Repeat until R = 0

CPU CONTROL

Mnemonics	Operands	Addr. Modes	Operation
COMFLG	flags	–	Complement Flag (Any combination of C, Z, S, P/V)
DI*	int	–	Disable Interrupt (Any combination of NVI, VI)
EI*	int	–	Enable Interrupt (Any combination of NVI, VI)
HALT*	–	–	HALT
LDCTL*	CTLR, src	R	Load into Control Register CTLR ← src
LDCTL*	dst, CTLR	R	Load from Control Register dst ← CTLR
LDCTLB	FLGR, src	R	Load into Flag Byte Register FLGR ← src
LDCTLB	dst, FLGR	R	Load from Flag Byte Register dst ← FLGR
LDPS*	src	IR DA X	Load Program Status PS ← src
MBIT*	–	–	Test Multi-Micro Bit Set S if $\overline{\mu I}$ is High; reset S if $\overline{\mu I}$ is Low.
MREQ*	dst	R	Multi-Micro Request
MRES*	–	–	Multi-Micro Reset
MSET*	–	–	Multi-Micro Set
NOP	–	–	No Operation
RESFLG	flag	–	Reset Flag (Any combination of C, Z, S, P/V)
SETFLG	flag	–	Set Flag (Any combination of C, Z, S, P/V)

*Privileged instructions. Executed in system mode only.

MAXIMUM RATINGS above which useful life may be impaired

Voltages on all inputs and outputs with respect to GND	−0.3 to +7.0V
Ambient Temperature under bias	0 to 70°C
Storage Temperature	−65 to +150°C

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

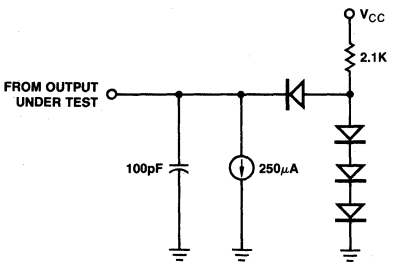
ELECTRICAL CHARACTERISTICS over operating range (Note 1)

			AmZ8002DC		
Parameter	**Description**	**Test Conditions**	**Min**	**Max**	**Units**
V_{CH}	Clock Input High Voltage	Driven by External Clock Generator	$V_{CC}-0.4$	$V_{CC}+0.3$	Volts
V_{CL}	Clock Input Low Voltage	Driven by External Clock Generator	−0.3	0.45	Volts
V_{IH}	Input High Voltage		2.0	$V_{CC}+0.3$	Volts
V_{IL}	Input Low Voltage		−0.3	0.8	Volts
V_{OH}	Output High Voltage	$I_{OH} = -250\mu A$	2.4		Volts
V_{OL}	Output Low Voltage	$I_{OL} = +2.0mA$		0.4	Volts
I_{IL}	Input Leakage	$0.4 \leq V_{IN} \leq +2.4V$		±10	μA
I_{OL}	Output Leakage	$0.4 \leq V_{OUT} \leq +2.4V$		±10	μA
I_{CC}	V_{CC} Supply Current			300	mA

Standard Test Conditions

The characteristics below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND. Positive current flows into the referenced pin. Standard conditions are as follows:

$+4.75V \leq V_{CC} \leq +5.25V$
GND = 0V
$0°C \leq T_A \leq +70°C$



MOS-238

All AC parameters assume a load capacitance of 100pF max. Timing references between two output signals assume a load difference of 50pF max.

SWITCHING CHARACTERISTICS over operating range

Number	Parameter	Description	AmZ8002DC Min	AmZ8002DC Max	Units
1	TcC	Clock Cycle Time	250	2000	ns
2	TwCh	Clock Width (High)	105	2000	ns
3	TwCl	Clock Width (Low)	105	2000	ns
4	TfC	Clock Fall Time		20	ns
5	TrC	Clock Rise Time		20	ns
6					
7					
8	TdC(Bz)	Clock ↑ to Bus Float		65	ns
9	TdC(A)	Clock ↑ to Address Valid		100	ns
10	TdC(Az)	Clock ↑ to Address Float		65	ns
11	TdA(DI)	Address Valid to Data In Required Valid	400		ns
12	TsDI(C)	Data In to Clock ↓ Set-up Time	70		ns
13	TdDS(A)	$\overline{DS}$ ↑ to Address Active	80		ns
14	TdC(DO)	Clock ↑ to Data Out Valid		100	ns
15	ThDI(DS)	Data In to $\overline{DS}$ ↑ Hold Time	0		ns
16	TdDO(DS)	Data Out Valid to $\overline{DS}$ ↑ Delay	230		ns
17	TdA(MR)	Address Valid to $\overline{MREQ}$ ↓ Delay	55		ns
18	TdC(MR)	Clock ↓ to $\overline{MREQ}$ ↓ Delay		80	ns
19	TwMRh	$\overline{MREQ}$ Width (High)	190		ns
20	TdMR(A)	$\overline{MREQ}$ ↓ to Address Not Active	70		ns
21	TdDO(DSW)	Data Out Valid to $\overline{DS}$ ↓ (Write) Delay	55		ns
22	TdMR(DI)	$\overline{MREQ}$ ↓ to Data In Required Valid	330		ns
23	TdC(MR)	Clock ↓ to $\overline{MREQ}$ ↑ Delay		80	ns
24	TdC(ASf)	Clock ↑ to $\overline{AS}$ ↓ Delay		80	ns
25	TdA(AS)	Address Valid to $\overline{AS}$ ↑ Delay	55		ns
26	TdC(ASr)	Clock ↓ to $\overline{AS}$ ↑ Delay		90	ns
27	TdAS(DI)	$\overline{AS}$ ↑ to Data In Required Valid	290		ns
28	TdDS(AS)	$\overline{DS}$ ↑ to $\overline{AS}$ ↓ Delay	70		ns
29	TwAS	$\overline{AS}$ Width (Low)	80		ns
30	TdAS(A)	$\overline{AS}$ ↑ to Address Not Active Delay	60		ns
31	TdAz(DSR)	Address Float to $\overline{DS}$ (Read) ↓ Delay	0		ns
32	TdAS(DSR)	$\overline{AS}$ ↑ to $\overline{DS}$ (Read) ↓ Delay	70		ns
33	TdDSR(DI)	$\overline{DS}$ (Read) ↓ to Data In Required Valid	155		ns
34	TdC(DSr)	Clock ↓ to $\overline{DS}$ ↑ Delay		70	ns
35	TdDS(DO)	$\overline{DS}$ ↑ to Data Out and STATUS Not Valid	80		ns
36	TdA(DSR)	Address Valid to $\overline{DS}$ (Read) ↓ Delay	120		ns
37	TdC(DSR)	Clock ↑ to $\overline{DS}$ (Read) ↓ Delay		120	ns
38	TwDSR	$\overline{DS}$ (Read) Width (Low)	275		ns
39	TdC(DSW)	Clock ↓ to $\overline{DS}$ (Write) ↓ Delay		95	ns
40	TwDSW	$\overline{DS}$ (Write) Width (Low)	160		ns
41	TdDSI(DI)	$\overline{DS}$ (Input) ↓ to Data In Required Valid	315		ns
42	TdC(DSf)	Clock ↓ to $\overline{DS}$ (I/O) ↓ Delay		120	ns
43	TwDS	$\overline{DS}$ (I/O) Width (Low)	400		ns
44	TdAS(DSA)	$\overline{AS}$ ↑ to $\overline{DS}$ (Acknowledge) ↓ Delay	960		ns
45	TdC(DSA)	Clock ↑ to $\overline{DS}$ (Acknowledge) ↓ Delay		120	ns
46	TdDSA(DI)	$\overline{DS}$ (Acknowledge) ↓ to Data In Required Delay	420		ns
47	TdC(S)	Clock ↑ to Status Valid Delay		110	ns
48	TdS(AS)	Status Valid to $\overline{AS}$ ↑ Delay	40		ns

SWITCHING CHARACTERISTICS (Cont.)

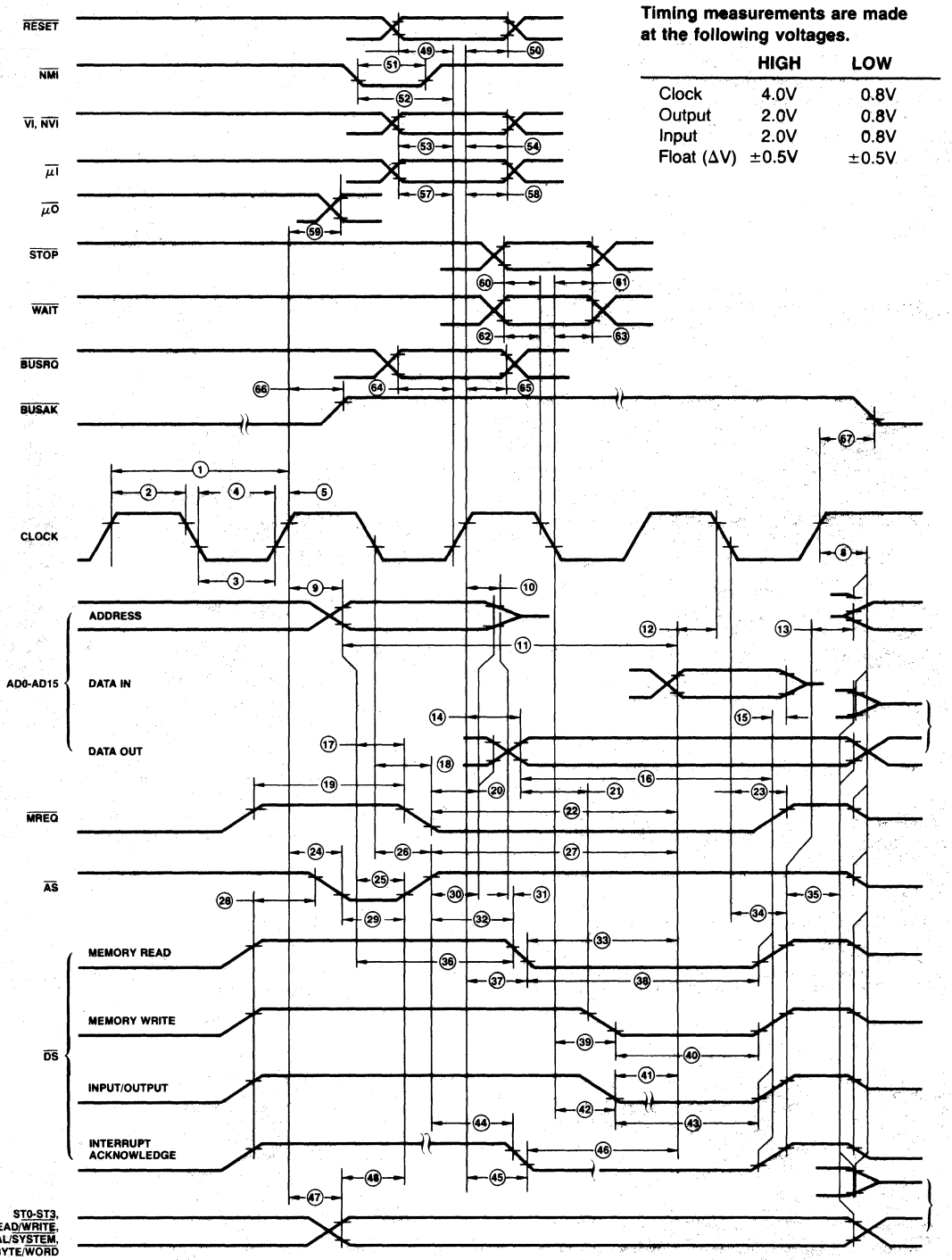
			AmZ8002DC		
Number	**Parameter**	**Description**	**Min**	**Max**	**Units**
49	TsR(C)	$\overline{\text{RESET}}$ to Clock ↑ Set-up Time	180		ns
50	ThR(C)	$\overline{\text{RESET}}$ to Clock ↑ Hold Time	0		ns
51	TwNMI	$\overline{\text{NMI}}$ Width (Low)	100		ns
52	TsNMI(C)	$\overline{\text{NMI}}$ to Clock ↑ Set-up Time	140		ns
53	TsVI(C)	$\overline{\text{VI}}$, $\overline{\text{NVI}}$ to Clock ↑ Set-up Time	110		ns
54	ThVI(C)	$\overline{\text{VI}}$, $\overline{\text{NVI}}$ to Clock ↑ Hold Time	0		ns
55					
56					
57	TsμI(C)	$\overline{\mu\text{I}}$ to Clock ↑ Set-up Time	180		ns
58	ThμI(C)	$\overline{\mu\text{I}}$ to Clock ↑ Hold Time	0		ns
59	TdC(μO)	Clock ↑ to $\overline{\mu\text{O}}$ Delay		120	ns
60	TsSTP(C)	$\overline{\text{STOP}}$ to Clock ↓ Set-up Time	140		ns
61	ThSTP(C)	$\overline{\text{STOP}}$ to Clock ↓ Hold Time	0		ns
62	TsWT(C)	$\overline{\text{WAIT}}$ to Clock ↓ Set-up Time	70		ns
63	ThWT(C)	$\overline{\text{WAIT}}$ to Clock ↓ Hold Time	0		ns
64	TsBRQ(C)	$\overline{\text{BUSRQ}}$ to Clock ↑ Set-up Time	90		ns
65	ThBRQ(C)	$\overline{\text{BUSRQ}}$ to Clock ↑ Hold Time	0		ns
66	TdC(BAKr)	Clock ↑ to $\overline{\text{BUSAK}}$ ↑ Delay		100	ns
67	TdC(BAKf)	Clock ↑ to $\overline{\text{BUSAK}}$ ↓ Delay		100	ns

For more information, refer to these AMD publications:

Processor Instruction Set (AM-PUB086).
Describes each instruction in detail. 250 pp.

Processor Interface Manual (AM-PUB089).
Describes hardware interfacing for interrupts and I/O, including the Am9511A Arithmetic Processor, the Am9517A DMA Controller, and the Am9519 Interrupt Controller. 81 pp.

AC TIMING DIAGRAM



Timing measurements are made at the following voltages.

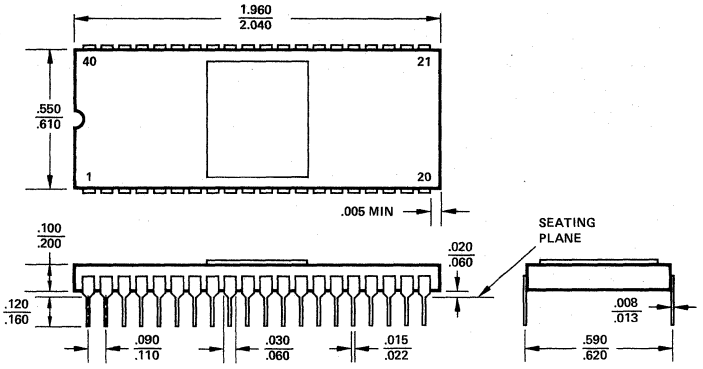
	HIGH	LOW
Clock	4.0V	0.8V
Output	2.0V	0.8V
Input	2.0V	0.8V
Float (ΔV)	±0.5V	±0.5V

This composite timing diagram does not show actual timing sequences. Refer to this diagram only for the detailed timing relationships of individual edges. Use the preceding illustrations as an explanation of the various timing sequences.

MOS-239

PHYSICAL DIMENSIONS
Dual-In-Line

40-Pin Ceramic



Memories

Am2716

2048 x 8-Bit UV Erasable PROM

DISTINCTIVE CHARACTERISTICS

- Direct replacement for Intel 2716
- Interchangeable with Am9218 – 16K ROM
- Single +5V power supply
- Fast access time – 450ns
- Low power dissipation
  - –525mW active
  - –132mW standby
- Fully static operation – no clocks
- Three-state outputs
- TTL compatible inputs/outputs
- 100% MIL-STD-883 reliability assurance testing

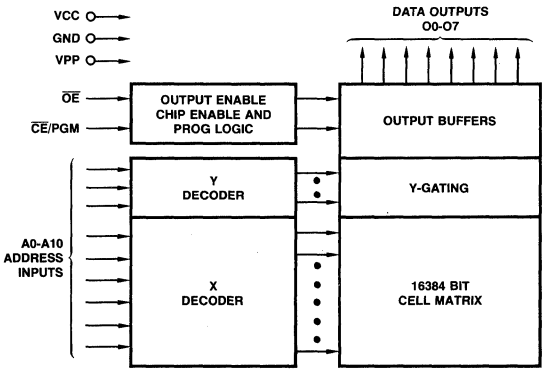
GENERAL DESCRIPTION

The Am2716 is a 16384-bit ultraviolet erasable and programmable read-only memory. It is organized as 2048 words by 8 bits per word, operates from a single +5V supply, has a static standby mode, and features fast single address location programming.

Because the Am2716 operates from a single +5V supply, it is ideal for use in microprocessor systems. All programming signals are TTL levels, requiring a single pulse. For programming outside of the system, existing EPROM programmers may be used. Locations may be programmed singly, in blocks, or at random. Total programming time for all bits is 100 seconds.

4

BLOCK DIAGRAM

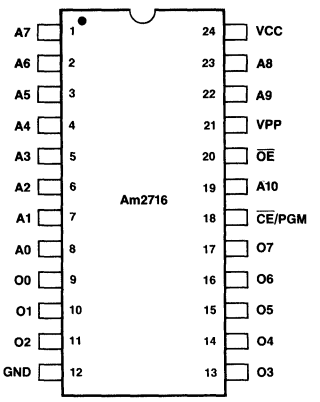


MOS-199

MODE SELECTION

Mode \ Pins	CE/PGM (18)	OE (20)	VPP (21)	VCC (24)	Outputs (9-11, 13-17)
Read	VIL	VIL	+5	+5	DOUT
Standby	VIH	Don't Care	+5	+5	High Z
Program	Pulsed VIL to VIH	VIH	+25	+5	DIN
Program Verify	VIL	VIL	+25	+5	DOUT
Program Inhibit	VIL	VIH	+25	+5	High Z

PIN CONFIGURATION



MOS-200

A0-A9: Addresses
O0-O7: Outputs
CE/PGM: Chip Enable/Program
OE: Output Enable

ORDERING INFORMATION

Package Type	Ambient Temperature Specification	Order Number
Hermetic DIP Transparent Window	$0°C \leq T_A \leq +70°C$	AM2716DC

MAXIMUM RATINGS above which the useful life may be impaired

Storage Temperature	−65°C to +125°C
Ambient Temperature Under Bias	−10°C to +80°C
Voltage on all inputs/outputs (except VPP) with respect to GND	+6V to −0.3V
Voltage on VPP During Program with Respect to GND	+26.5V to −0.3V

READ OPERATION

DC CHARACTERISTICS

0°C ≤ T_A ≤ +70°C, VCC (Notes 1, 2) = +5V ±5%, VPP (Note 2) = VCC

Parameters	Description	Test Conditions	Min	Max	Units
ILI	Input Load Current	VIN = 5.25V		10	μA
ILO	Output Leakage Current	VOUT = 5.25V		10	μA
IPP1 (Note 2)	VPP Current	VPP = 5.25V		5	mA
ICC1 (Note 2)	VCC Current (Standby)	$\overline{CE}$ = VIH, $\overline{OE}$ = VIL		25	mA
ICC2 (Note 2)	VCC Current (Active)	$\overline{OE}$ = $\overline{CE}$ = VIL		100	mA
VIL	Input Low Voltage		−0.1	0.8	Volts
VIH	Input High Voltage		2.0	VCC+1	Volts
VOL	Output Low Voltage	IOL = 2.1mA		0.45	Volts
VOH	Output High Voltage	IOH = −400μA	2.4		Volts

AC CHARACTERISTICS

0°C ≤ T_A ≤ +70°C, VCC (Notes 1, 2) = +5V ±5%, VPP (Note 2) = VCC

Parameters	Description	Test Conditions		Min	Max	Units
tACC	Address to Output Delay	Output Load: 1 TTL gate and CL = 100pF Input Rise and Fall Times: ≤20ns Input Pulse Levels: 0.8V to 2.2V Timing Measurement Reference Level: Inputs: 1V and 2V Outputs: 0.8V and 2V	$\overline{CE}$ = $\overline{OE}$ = VIL		450	ns
tCE	$\overline{CE}$ to Output Delay		$\overline{OE}$ = VIL		450	ns
tOE	Output Enable to Output Delay		$\overline{CE}$ = VIL		120	ns
tDF	Output Enable High to Output Float		$\overline{CE}$ = VIL	0	100	ns
tOH	Output Hold from Addresses, $\overline{CE}$ or $\overline{OE}$ Whichever Occurred First		$\overline{CE}$ = $\overline{OE}$ = VIL	0		ns

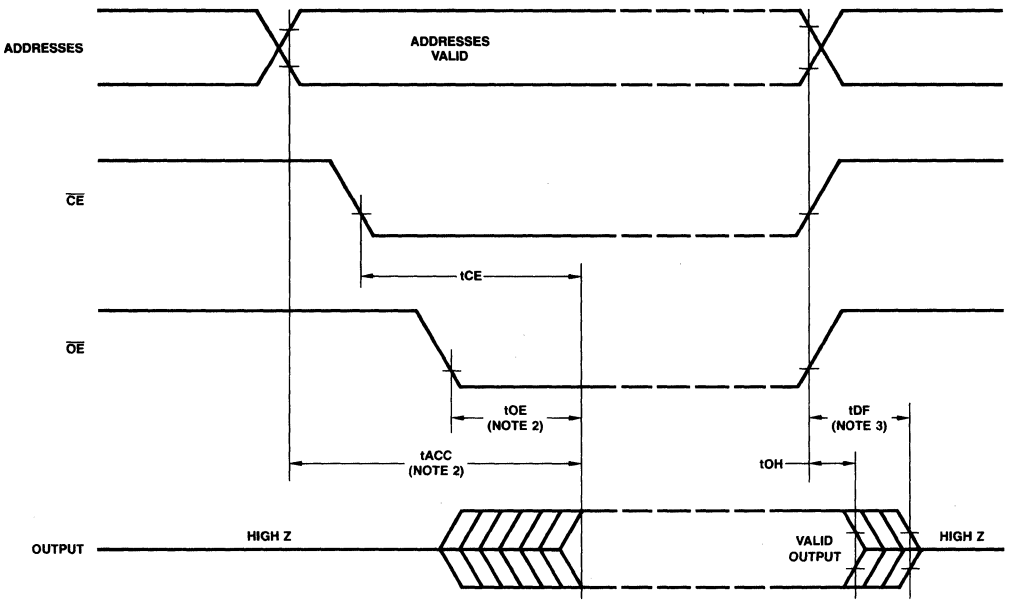
CAPACITANCE (Note 3)

T_A = +25°C, f = 1MHz

Parameters	Description	Test Conditions	Typ	Max	Units
CIN	Input Capacitance	VIN = 0V	4	6	pF
COUT	Output Capacitance	VOUT = 0V	8	12	pF

Notes: 1. VCC must be applied simultaneously or before VPP and removed simultaneously or after VPP.
2. VPP may be connected directly to VCC except during programming. The supply current would then be the sum of ICC and IPP1.
3. This parameter is only sampled and is not 100% tested.

AC WAVEFORMS (Note 1)



MOS-201

Notes: 1. VCC must be applied simultaneously or before VPP and removed simultaneously or after VPP.
2. OE may be delayed up to tACC − tOE after the falling edge of CE without impact on tACC.
3. tDF is specified from OE or CE, whichever occurs first.

PROGRAM OPERATION

DC PROGRAMMING CHARACTERISTICS

T_A = +25°C ±5°C, VCC (Note 1) = 5V ±5%, VPP (Notes 1, 2) = 25V ±1V

Parameters	Description	Test Conditions	Min	Max	Units
ILI	Input Current	VIN = 5.25V/0.45V		10	μA
IPP1	VPP Supply Current	$\overline{CE}$/PGM = VIL		5	mA
IPP2	VPP Supply Current During Programming Pulse	$\overline{CE}$/PGM = VIH		30	mA
ICC	VCC Supply Current			100	mA
VIL	Input Low Level		−0.1	0.8	Volts
VIH	Input High Level		2.0	VCC+1	Volts

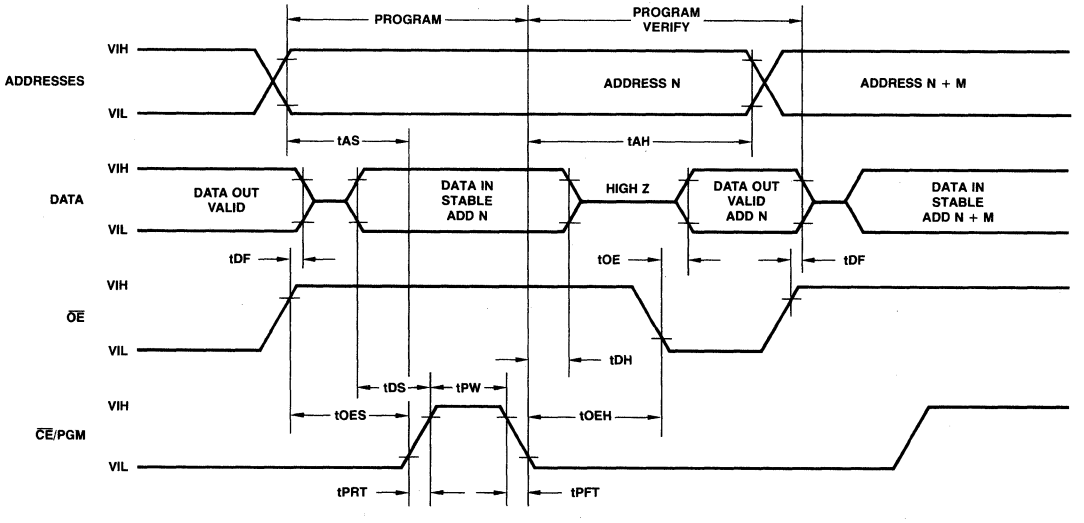
Notes: 1. VCC must be applied simultaneously or before VPP and removed simultaneously or after VPP.
2. VPP must not be greater than 26 volts including overshoot. Permanent device damage may occur if the device is taken out of or put into the socket when VPP = 25 volts is applied. Also, during $\overline{OE}$ = $\overline{CE}$/PGM = VIH, VPP must not be switched from 5 volts to 25 volts or vice versa.

AC PROGRAMMING CHARACTERISTICS

T_A = +25°C ±5°C, VCC (Note 1) = 5V ±5%, VPP (Notes 1, 2) = 25V ±1V

Parameters	Description	Test Conditions	Min	Max	Units
tAS	Address Set-up Time	Input tR and tF (10% to 90%) = 20ns Input Signal Levels = 0.8V to 2.2V Input Timing Reference Level = 1V and 2V Output Timing Reference Level = 0.8V and 2V	2		μs
tOES	Output Enable Set-up Time		2		μs
tDS	Data Set-up Time		2		μs
tAH	Address Hold Time		2		μs
tOEH	Output Enable Hold Time		2		μs
tDH	Data Hold Time		2		μs
tDF	Output Disable to Output Float Delay ($\overline{CE}$/PGM = VIL)		0	120	ns
tOE	Output Enable to Output Delay ($\overline{CE}$/PGM = VIL)		–	120	ns
tPW	Program Pulse Width		45	55	ms
tPRT	Program Pulse Rise Time		5	–	ns
tPFT	Program Pulse Fall Time		5	–	ns

PROGRAMMING WAVEFORMS



MOS-202

ERASING THE Am2716

In order to clear all locations of their programmed contents, it is necessary to expose the Am2716 to an ultraviolet light source. A dosage of 15 Wseconds/cm^2 is required to completely erase an Am2716. This dosage can be obtained by exposure to an ultraviolet lamp [(wavelength of 2537 Angstroms (Å)] with intensity of 12000μW/cm^2 for 15 to 20 minutes. The Am2716 should be about one inch from the source and all filters should be removed from the UV light source prior to erasure.

It is important to note that the Am2716, and similar devices, will erase with light sources having wavelengths shorter than 4000 Angstroms. Although erasure times will be much longer than with UV sources at 2537Å, nevertheless the exposure to florescent light and sunlight will eventually erase the Am2716, and exposure to them should be prevented to realize maximum system reliability. If used in such an environment, the package windows should be covered by an opaque label or substance.

PROGRAMMING THE Am2716

Upon delivery, or after each erasure the Am2716 has all 16384 bits in the "1", or high state. "0s" are loaded into the Am2716 through the procedure of programming.

The programming mode is entered when +25V is applied to the VPP pin and when $\overline{OE}$ is at VIH. The address to be programmed is applied to the proper address pins. 8-bit patterns are placed on the respective data output pins. The voltage levels should be standard TTL levels. When both the address and data are stable, a 50msec, TTL high level pulse is applied to the $\overline{CE}$/PGM input to accomplish the programming.

The procedure can be done manually, address by address, randomly, or automatically via the proper circuitry. All that is required is that one 50msec program pulse be applied at each address to be programmed. It is necessary that this program pulse width not exceed 55msec. Therefore, applying a DC level to the $\overline{CE}$/PGM input is prohibited when programming.

READ MODE

The Am2716 has two control functions, both of which must be logically satisfied in order to obtain data at the outputs. Chip Enable ($\overline{CE}$) is the power control and should be used for device selection. Output Enable ($\overline{OE}$) is the output control and should be used to gate data to the output pins, independent of device selection. Assuming that addresses are stable, address access time (tACC) is equal to the delay from $\overline{CE}$ to output (tCE). Data is available at the outputs 120ns (tOE) after the falling edge of $\overline{OE}$, assuming that $\overline{CE}$ has been low and addresses have been stable for at least tACC − tOE.

STANDBY MODE

The Am2716 has a standby mode which reduces the active power dissipation by 75%, from 525mW to 132mW. The Am2716 is placed in the standby mode by applying a TTL high signal to the $\overline{CE}$ input. When in standby mode, the outputs are in a high impedance state, independent of the $\overline{OE}$ input.

OUTPUT OR-TIEING

To accommodate multiple memory connections, a 2-line control function is provided to allow for:

1. Low memory power dissipation
2. Assurance that output bus contention will not occur.

It is recommended that $\overline{CE}$ be decoded and used as the primary device selecting function, while $\overline{OE}$ be made a common connection to all devices in the array and connected to the READ line from the system control bus. This assures that all deselected memory devices are in their low-power standby mode and that the output pins are only active when data is desired from a particular memory device.

PROGRAM INHIBIT

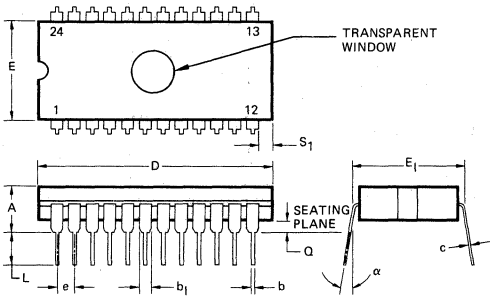
Programming of multiple Am2716s in parallel with different data is also easily accomplished. Except for $\overline{CE}$/PGM, all like inputs (including $\overline{OE}$) of the parallel Am2716s may be common. A TTL level program pulse applied to an Am2716's $\overline{CE}$/PGM input with VPP at 25V will program that Am2716. A low level $\overline{CE}$/PGM input inhibits the other Am2716 from being programmed.

PROGRAM VERIFY

A verify should be performed on the programmed bits to determine that they were correctly programmed. The verify may be performed with VPP at 25V. Except during programming and program verify, VPP must be at 5V.

PHYSICAL DIMENSIONS

24-Lead Hermetic Cerdip



Reference Symbol	Min.	Max.
A	.150	.225
b	.016	.020
b_1	.045	.065
c	.009	.011
D	1.235	1.280
E	0.51	0.55
E_1	0.60	0.63
e	.090	.110
L	0.12	0.15
Q	.015	.060
S_1*	0.01	
α	3°	13°

*From edge of end lead.

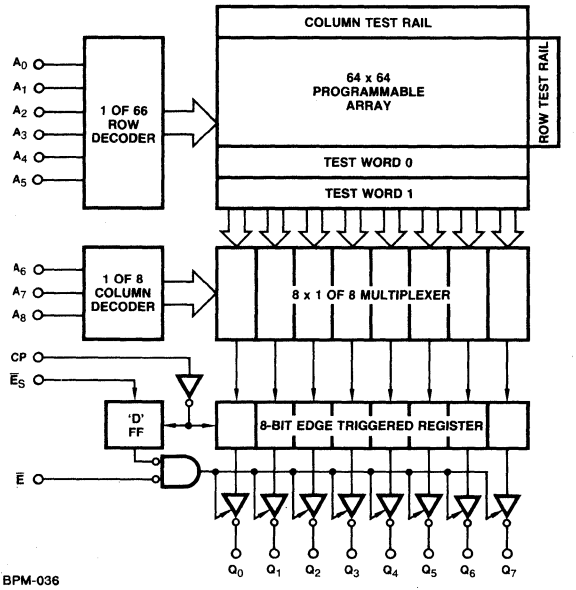
Am27S26 • Am27S27

4096-Bit Generic Series Bipolar PROM

DISTINCTIVE CHARACTERISTICS

- On chip edge triggered registers – Ideal for pipelined microprogrammed systems
- Versatile synchronous and asynchronous enables for simplified word expansion
- Predetermined OFF outputs on power-up
- Fast 50ns address setup and 20ns clock to output times
- Excellent performance over the military range
- Performance pretested with N^2 patterns
- Space saving 22 pin package
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Common Generic PROM Series characteristics and programming procedures

BLOCK DIAGRAM



BPM-036

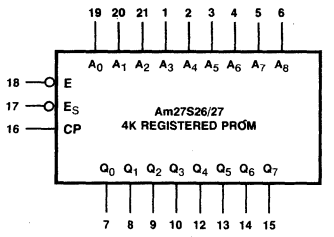
FUNCTIONAL DESCRIPTION

The Am27S26 and Am27S27 are electrically programmable Schottky TTL read only memories incorporating true D-type, master-slave data registers on chip. These devices feature the versatile 512 word by 8 bit organization and are available in both the open collector Am27S26 and three-state Am27S27 output versions. Designed to optimize system performance, these devices also substantially reduce the cost of pipelined microprogrammed system and other designs wherein accessed PROM data is temporarily stored in a register. The Am27S26 and Am27S27 also offer maximal flexibility for memory expansion and data bus control by providing both synchronous and asynchronous output enables.

When V_{CC} power is first applied, the synchronous enable ($\overline{E}_S$) flip-flop will be in the set condition causing the outputs, Q_0-Q_7, to be in the OFF or high impedance state, eliminating the need for a register clear input. Reading data is accomplished by first applying the binary word address to the address inputs, A_0-A_8, and a logic LOW to the synchronous output enable, $\overline{E}_S$. During the address setup time, stored data is accessed and loaded into the master flip-flops of the data register. Since the synchronous enable setup time is less than the address setup requirement, additional decoding delays may occur in the enable path without reducing memory performance. Upon the next LOW-to-HIGH transition of the clock, CP, data is transferred to the slave flip-flops which drive the output buffers. Providing the asynchronous enable, $\overline{E}$, is also LOW, stored data will appear on the outputs, Q_0-Q_7. If $\overline{E}_S$ is HIGH when the positive clock edge occurs, outputs go to the OFF or high impedance state. The outputs may be disabled at any time by switching $\overline{E}$ to a HIGH level. Following the positive clock edge, the address and synchronous enable inputs are free to change; changes do not affect the outputs until another positive clock edge occurs. This unique feature allows the PROM decoders and sense amplifiers to access the next location while previously addressed data remains stable on the outputs. For less complex applications either enable may be effectively eliminated by tying it to ground.

The on-chip, edge triggered register simplifies system timing since the PROM clock may be derived directly from system clock without introducing dangerous race conditions. Other register timing requirements are similar to those of standard Schottky registers and are easily implemented.

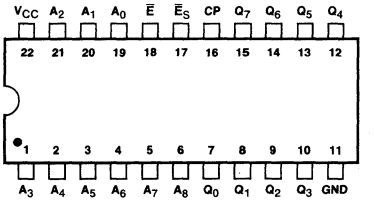
LOGIC SYMBOL



V_{CC} = Pin 22
GND = Pin 11

BPM-037

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BPM-038

GENERIC SERIES CHARACTERISTICS

The Am27S26 and Am27S27 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test words are pre-programmed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliability. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential (Pin 22 to Pin 11) Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +V_{CC} max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5V to +5.5V
DC Input Current	−30mA to +5mA

OPERATING RANGE

COM'L	AM27S26XC, AM27S27XC	T_A = 0°C to +75°C	V_{CC} = 5.0V ±5%
MIL	AM27S26XM, AM27S27XM	T_C = −55°C to +125°C	V_{CC} = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions			Min.	Typ. (Note 1)	Max.	Units
V_{OH} (Am27S27 only)	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −2.0mA V_{IN} = V_{IH} or V_{IL}			2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}				0.38	0.50	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs			2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs					0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.45V				−0.010	−0.250	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V					25	μA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V					1.0	mA
I_{SC} (Am27S27 only)	Output Short Circuit Current	V_{CC} = MAX., V_{OUT} = 0.0V (Note 2)			−20	−40	−90	mA
I_{CC}	Power Supply Current	All inputs = GND V_{CC} = MAX.				130	185	mA
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA					−1.2	Volts
I_{CEX}	Output Leakage Current	V_{CC} = MAX. $\overline{VE}$ = 2.4V		V_O = 4.5V			100	μA
			Am27S27 Only	V_O = 2.4V			40	
				V_O = 0.4V			−40	
C_{IN}	Input Capacitance	V_{IN} = 2.0V @ f = 1MHz (Note 3)				5		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0V @ f = 1MHz (Note 3)				12		

Notes: 1. Typical limits are at V_{CC} = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but are periodically sampled.

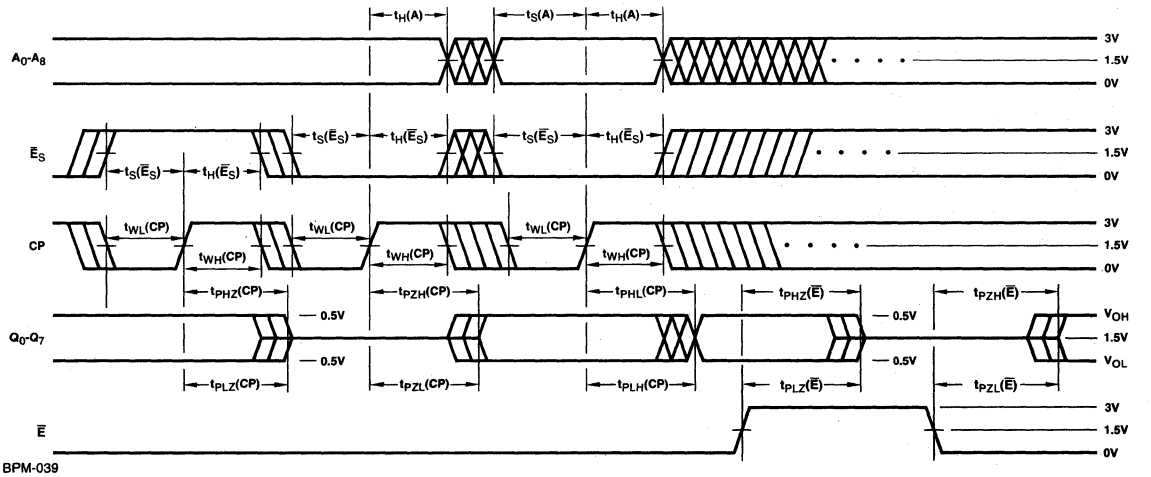
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ 5V 25°C	COM'L Min	COM'L Max	MIL Min	MIL Max	Units
$t_S(A)$	Address to CP (HIGH) Setup Time	C_L = 30pF S_1 closed. (See AC Test Load below)	40	55		65		ns
$t_H(A)$	Address to CP (HIGH) Hold Time		−15	0		0		ns
$t_{PHL}(CP)$ $t_{PLH}(CP)$	Delay from CP (HIGH) to Output (HIGH or LOW)		15		27		30	ns
$t_{WH}(CP)$ $t_{WL}(CP)$	CP Width (HIGH or LOW)		10	30		40		ns
$t_S(\overline{E}_S)$	$\overline{E}_S$ to CP (HIGH) Setup Time		10	25		30		ns
$t_H(\overline{E}_S)$	$\overline{E}_S$ to CP (HIGH) Hold Time		−10	0		0		ns
$t_{PZL}(CP)$ $t_{PZH}(CP)$	Delay from CP (HIGH) to Active Output (HIGH or LOW) (Note 1)	C_L = 30pF S_1 closed for t_{PZL} and open for t_{PZH}	15		35		45	ns
$t_{PZL}(\overline{E})$ $t_{PZH}(\overline{E})$	Delay from $\overline{E}$ (LOW) to Active Output (HIGH or LOW) (Note 1)		15		40		45	ns
$t_{PLZ}(CP)$ $t_{PHZ}(CP)$	Delay from CP (HIGH) to Inactive Output (OFF or High Impedance) (Note 1)	C_L = 5pF (Note 2) S_1 closed for t_{PLZ} and open for t_{PHZ}	15		35		45	ns
$t_{PLZ}(\overline{E})$ $t_{PHZ}(\overline{E})$	Delay from $\overline{E}$ (HIGH) to Inactive Output (OFF or High Impedance) (Note 1)		10		30		40	ns

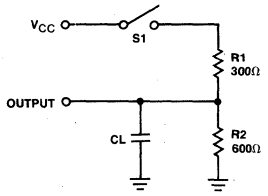
Notes:
1. t_{PHZ} and t_{PZH} apply to the three-state Am27S27 only.
2. t_{PHZ} and t_{PLZ} are measured to the V_{OH} − 0.5V and V_{OL} + 0.5V output levels respectively. All other switching parameters are tested from and to the 1.5V threshold levels.
3. Tests are performed with input 10% to 90% rise and fall times of 5ns or less.

SWITCHING WAVEFORMS



BPM-039

AC TEST LOAD



BPM-040

KEY TO TIMING DIAGRAM

WAVEFORM	INPUTS	OUTPUTS
	MUST BE STEADY	WILL BE STEADY
	MAY CHANGE FROM H TO L	WILL BE CHANGING FROM H TO L
	MAY CHANGE FROM L TO H	WILL BE CHANGING FROM L TO H

WAVEFORM	INPUTS	OUTPUTS
	DON'T CARE; ANY CHANGE PERMITTED	CHANGING; STATE UNKNOWN
	DOES NOT APPLY	CENTER LINE IS HIGH IMPEDANCE "OFF" STATE

PROGRAMMING

The Am27S26 and Am27S27 are manufactured with a conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to the memory output after the $\overline{E}$ input is at a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{E}$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip is enabled and the CP input is clocked. Each data verification attempt must be preceded by a positive going (LOW-to-HIGH) clock edge to load the array data into the on-chip register. The output level is then sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycles. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{E}$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be continuously applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including V_{CC} should be removed for a period of 5 seconds after which programming may be resumed.

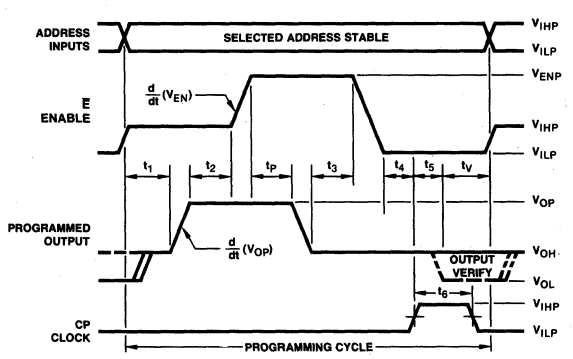
When all programming has been completed, the data content of the memory should be verified by clocking and reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min.	Max.	Units
V_{CCP}	V_{CC} During Programming	5.0	5.5	V
V_{IHP}	Input HIGH Level During Programming	2.4	5.5	V
V_{ILP}	Input LOW Level During Programming	0.0	0.45	V
V_{ENP}	$\overline{E}$ Voltage During Programming	14.5	15.5	V
V_{OP}	Output Voltage During Programming	19.5	20.5	V
V_{ONP}	Voltage on Outputs Not to be Programmed	0	V_{CCP}+0.3	V
I_{ONP}	Current into Outputs Not to be Programmed		20	mA
$d(V_{OP})/dt$	Rate of Output Voltage Change	20	250	V/μsec
$d(V_{EN})/dt$	Rate of $\overline{E}$ Voltage Change	100	1000	V/μsec
t_P	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5.0	15	msec

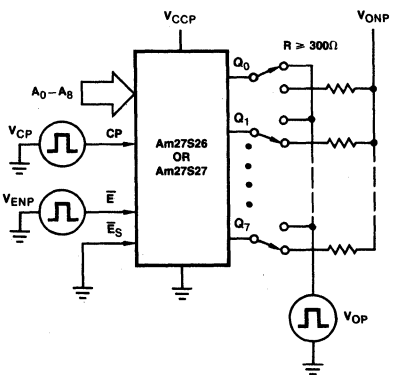
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e., not to the midpoints.
2. Delays t_1 through t_6 must be greater than 100 ns; maximum delays of 1 μsec are recommended to minimize heating during programming.
3. During t_V, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to V_{ONP} through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-041

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-042

APPLYING THE Am27S26 AND Am27S27 IN BIPOLAR MICROCOMPUTERS

With the advent of the Am2901 and Am2903 4-bit microprocessor slices, the Am2910 bipolar microprogram sequencer and the Am27S26/27 registered PROM, the design engineer can upgrade the performance of existing systems or implement new systems taking advantage of the latest state-of-the-art technology in Low-Power Schottky integrated circuits. These devices, however, utilize a new concept in machine design not familiar to many design engineers. This technique is called microprogramming.

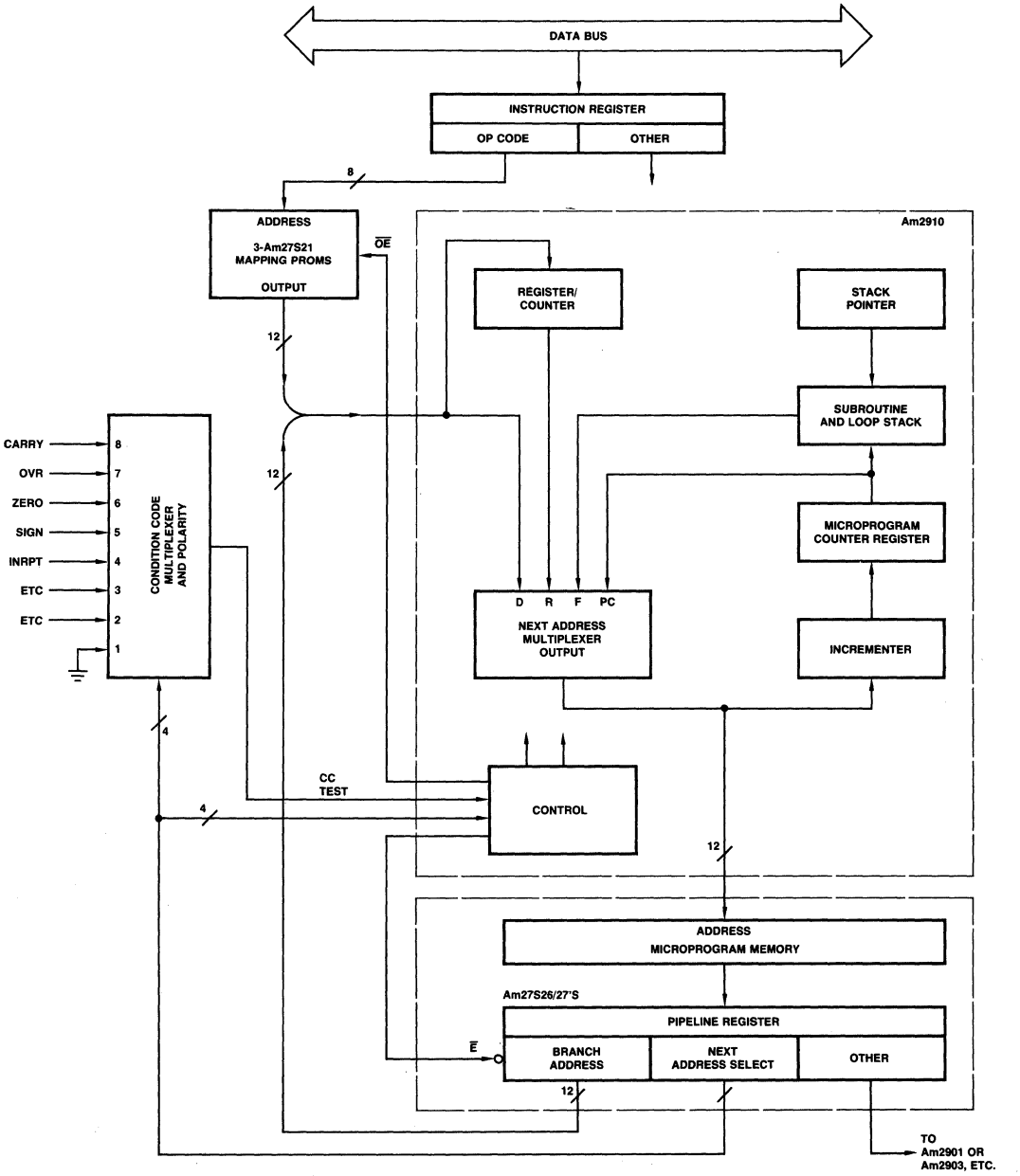


Fig. 1. A Typical Computer Control Unit using the Am27S26/27. BPM-044

APPLYING THE Am27S26 and Am27S27 IN BIPOLAR MICROCOMPUTERS (Cont.)

Basically, a microprogrammed machine is one in which a coherent sequence of microinstructions is used to execute various commands required by the machine. If the machine is a computer, each sequence of microinstructions can be made to execute a machine instruction. All of the little elemental tasks performed by the machine in executing the machine instruction are called microinstructions. The storage area for these microinstructions is usually called the microprogram memory.

A microinstruction usually has two primary parts. These are: (1) the definition and control of all elemental micro-operations to be carried out and (2) the definition and control of the address of the next microinstruction to be executed.

The definition of the various micro-operations to be carried out usually includes such things as ALU source operand selection, ALU function, ALU destination, carry control, shift control, interrupt control, data-in and data-out control, and so forth. The definition of the next microinstruction function usually includes identifying the source selection of the next microinstruction address and, in some cases, supplying the actual value of that microinstruction address.

Microprogrammed machines are usually distinquished from non-microprogrammed machines in the following manner. Older, non-microprogrammed machines implemented the control function by using combinations of gates and flip-flops connected in a somewhat random fashion in order to generate the required timing and control signals for the machine. Microprogrammed machines, on the other hand, are normally considered highly ordered, particularly with regard to the control function field, and use high speed PROMs for control definition. In its simplest definition, a microprogram control unit consists of the microprogram memory and the structure required to determine the address of the next microinstruction

The microprogram memory control unit block diagram of Figure 1 is easily implemented using the Am2910 and the Am27S26/27 registered PROM's. This architecture provides a structured state machine design capable of executing many highly sophisticated next address control instructions. The Am2910 contains a next address multiplexer that provides four different inputs from which the address of the next microinstruction can be selected. These are the direct input (D), the register input (R), the program counter (PC), and the file (F). The starting address decoder (mapping PROM) output and the Am27S26/27's pipeline register output are connected together at the D input to the Am2910 and are operated in the three-state mode.

The architecture of Figure 1 shows an instruction register capable of being loaded with a machine instruction word from the data bus. The op code portion of the instruction is decoded using a mapping PROM to arrive at a starting address for the microinstruction sequence required to execute the machine instruction. When the microprogram memory address is to be the first microinstruction of the machine instruction sequence, the Am2910 next address control selects the multiplexer D input and enables the three-state output from the mapping PROM. When the current microinstruction being executed is selecting the next microinstruction address as a JUMP function, the JUMP address will be available at the multiplexer D input. This is accomplished by having the Am2910 select the next address multiplexer D input and also enabling the three-state output of the pipeline register branch address field. The register enable input to the Am2910 can be grounded so that this register will load the value at the Am2910 D input. The value at D is clocked into the Am2910's register (R) at the end of the current microcycle, which makes the D value of this microcycle available as the R value of the next microcycle. Thus, by using the branch address field of two sequential microinstructions, a conditional JUMP-TO-ONE-OF-TWO-SUBROUTINES or a conditional JUMP-TO-ONE-OF-TWO-BRANCH-ADDRESSES can be executed by either selecting the D input or the R input of the next address multiplexer.

When sequencing through continuous microinstructions in the Am27S26/27 microprogram memory, the program counter in the Am2910 is used. Here, the control logic simply selects the PC input of the next address multiplexer. In addition, most of these instructions enable the three-state outputs of the Am27S26/27 pipeline register associated with the branch address field, which allows the register within the Am2910 to be loaded. The 5 x 12 stack in the Am2910 is used for looping and subroutining in microprogram operations. Up to five levels of subroutines or loops can be nested. Also, loops and subroutines can be intermixed as long as the five word depth of the stack is not exceeded.

The expansion scheme for increasing the depth of Am27S26/27's is shown in Figure 2. Note that no speed degradation results when devices are cascaded. This is because the decode of the Am74S139 is in parallel with the PROM access time.

In order to provide an overall view of a typical Am2900 Bipolar Microprocessor, the block diagram of Figure 3 is presented. Here, the computer control unit (CCU) using the Am2910 and Am27S26/27 registered PROM's is depicted. In addition, the typical connection scheme for the Am2903 Bipolar Microprocessor slices is shown. The four Am2903 devices in the block diagram form a typical 16-bit architecture. Also shown in Figure 3 is the general connection for the Am2914 Priority Interrupt Controller and the Am2920 as a Memory Address Register.

The block diagram also shows the Am2917 as the bus interface unit. Note that the Am2917 can interface directly with a data bus and the drive levels and receive levels are such that the instruction register can be built using standard Low-Power Schottky devices. This is possible because the receiver threshold on the Am2917 is designed identically to the thresholds on standard power Schottky and the Low-Power Schottky devices.

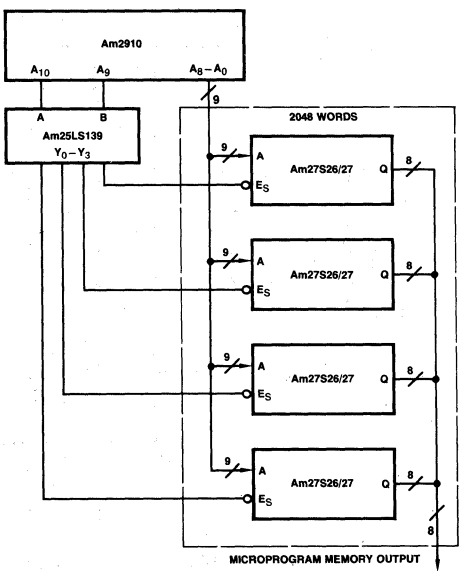


Fig. 2. Word Expansion Scheme for the Am27S26 and Am27S27.

BPM-045

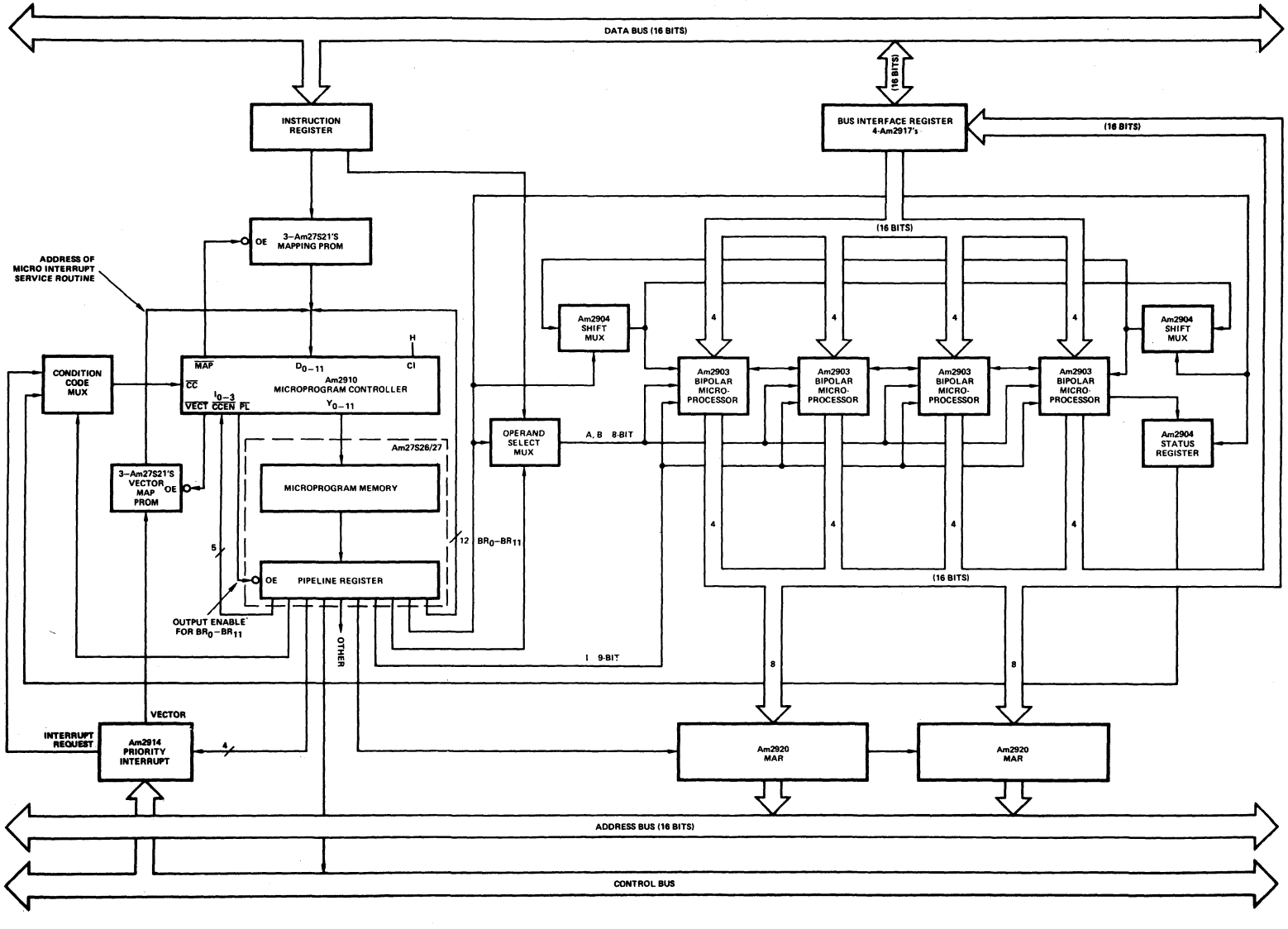
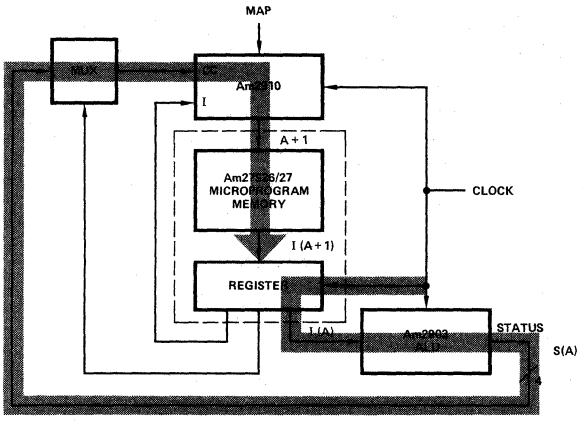


Fig. 3. Typical Bipolar Microcomputer using the Am27S26/27.

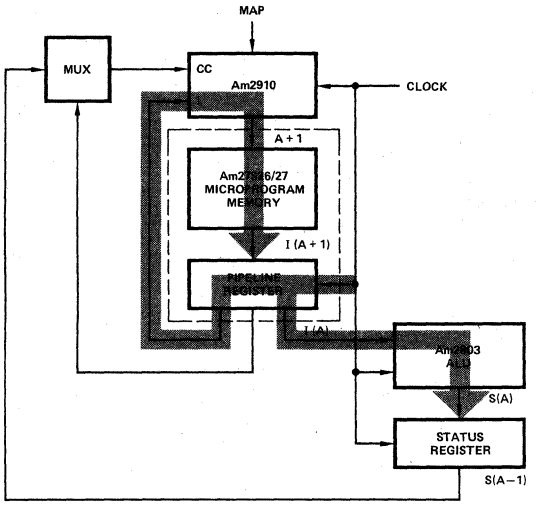
BPM-046

USING THE Am27S26/27 IN A PIPELINED ARCHITECTURE



A Register at the Microprogram Memory output contains the microinstruction being executed. The microprogram memory and Am2903 delay are in series. Conditional branches are executed on same cycle as the ALU operation generating the condition.

BPM-047



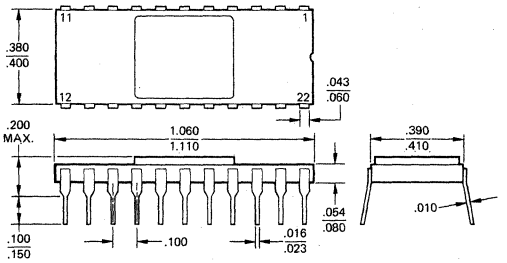
One level pipeline provides better speed than the architecture to the left. The Microprogram Memory and the Am2903 array are in parallel speed paths instead of in series. This is the recommended architecture for Am2900 designs. Note that the Am27S26/27 reduces the parts count of the microprogram memory/pipeline by a factor of two.

BPM-048

PHYSICAL DIMENSIONS

Dual In-Line

22-Pin Hermetic



ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Open Collectors		
Hermetic DIP	0°C to +75°C	AM27S26DC
Hermetic DIP	−55°C to +125°C	AM27S26DM
Three-State Outputs		
Hermetic DIP	0°C to +75°C	AM27S27DC
Hermetic DIP	−55°C to +125°C	AM27S27DM

Am27S28 • Am27S29

4096-Bit Generic Series Bipolar PROM

DISTINCTIVE CHARACTERISTICS

- High Speed – 55ns max commercial range access time
- Excellent performance over full MIL and commercial ranges
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Fast chip select
- Access time tested with N^2 patterns
- Pin for pin replacements for industry standard products
- Common Generic PROM series electrical characteristics and simple programming procedures.

GENERIC SERIES CHARACTERISTICS

The Am27S28 and Am27S29 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test rows are pre-programmed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliabilty. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

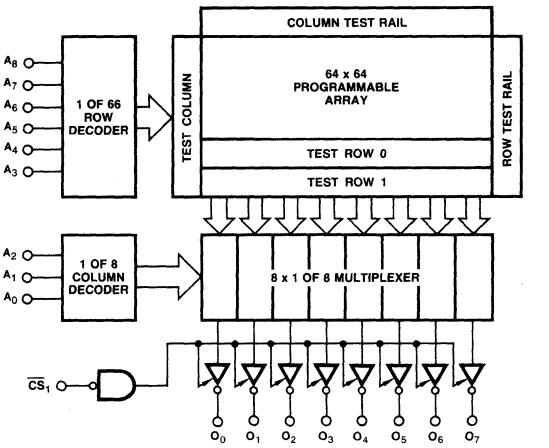
ORDERING INFORMATION

Package Type	Temperature Range	Order Number
	Open Collectors	
Hermetic DIP	0°C to +75°C	AM27S28DC
Hermetic DIP	−55°C to +125°C	AM27S28DM
	Three-State Outputs	
Hermetic DIP	0°C to +75°C	AM27S29DC
Hermetic DIP	−55°C to +125°C	AM27S29DM

FUNCTIONAL DESCRIPTION

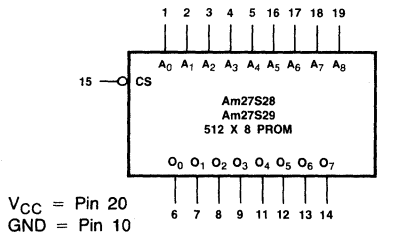
The Am27S28 and Am27S29 are high speed electrically programmable Schottky read only memories. Organized in the industry standard 512 x 8 configuration, they are available in both open collector Am27S28 and three-state Am27S29 output versions. After programming, stored information is read on outputs O_0-O_7 by applying unique binary addresses to A_0-A_8 and holding the chip select input, $\overline{CS}$, at a logic LOW. If the chip select input goes to a logic HIGH, O_0-O_7 go to the off or high impedance state.

BLOCK DIAGRAM



BPM-083

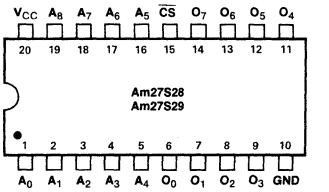
LOGIC SYMBOL



V_{CC} = Pin 20
GND = Pin 10

BPM-084

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BPM-085

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential (Pin 20 to Pin 10) Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +V_{CC} max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5V to +5.5V
DC Input Current	−30mA to +5mA

OPERATING RANGE

COM'L	Am27S28XC, Am27S29XC	T_A = 0°C to +75°C	V_{CC} = 5.0V ±5%
MIL	Am27S28XM, Am27S29XM	T_A = −55°C to +125°C	V_{CC} = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions	Min.	Typ. (Note 1)	Max.	Units
V_{OH} (Am27S29 only)	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −2.0mA V_{IN} = V_{IH} or V_{IL}	2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}			0.50	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs	2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs			0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.45V		−0.010	−0.250	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V			25	μA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V			1.0	mA
I_{SC} (Am27S29 only)	Output Short Circuit Current	V_{CC} = MAX., V_{OUT} = 0.0V (Note 2)	−20	−40	−90	mA
I_{CC}	Power Supply Current	All inputs = GND V_{CC} = MAX.		105	160	mA
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA			−1.2	Volts
I_{CEX}	Output Leakage Current	V_{CC} = MAX., $\overline{V_{CS}}$ = 2.4V; Am27S29 only; V_O = 4.5V			40	μA
		V_{CC} = MAX., $\overline{V_{CS}}$ = 2.4V; Am27S29 only; V_O = 2.4V			40	μA
		V_{CC} = MAX., $\overline{V_{CS}}$ = 2.4V; Am27S29 only; V_O = 0.4V			−40	μA
C_{IN}	Input Capacitance	V_{IN} = 2.0V @ f = 1 MHz (Note 3)		4		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0V @ f = 1 MHz (Note 3)		8		pF

Notes: 1. Typical limits are at V_{CC} = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but periodically sampled.

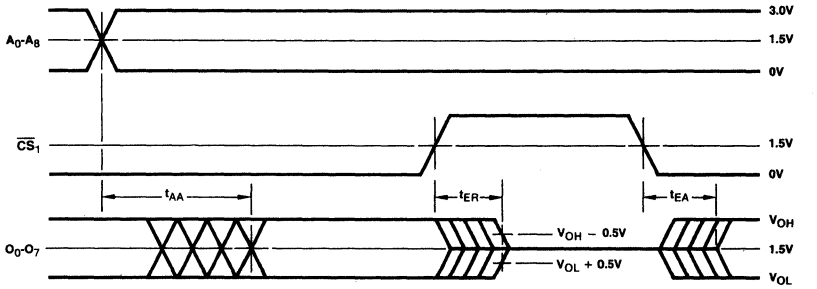
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ 5V 25°C	Max COM'L	Max MIL	Units
t_{AA}	Address Access Time	AC Test Load (See Notes 1-3)	35	55	70	ns
t_{EA}	Enable Access Time		15	25	30	ns
t_{ER}	Enable Recovery Time		15	25	30	ns

Notes:
1. t_{AA} is tested with switch S_1 closed and C_L = 30pF.
2. For open collector outputs, t_{EA} and t_{ER} are tested with S_1 closed to the 1.5V output level. C_L = 30pF.
3. For three state outputs, t_{EA} is tested with C_L = 30pF to the 1.5V level; S_1 is open for high impedance to HIGH tests and closed for high impedance to LOW tests. t_{ER} is tested with C_L = 5pF. HIGH to high impedance tests are made with S_1 open to an output voltage of V_{OH} − 0.5V; LOW to high impedance tests are made with S_1 closed to the V_{OL} + 0.5V level.

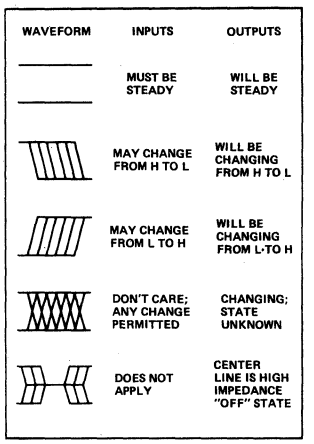
SWITCHING WAVEFORMS



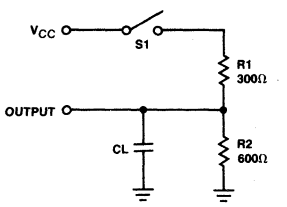
Note: Level on output while CS is HIGH is determined externally.

BPM-086

KEY TO TIMING DIAGRAM



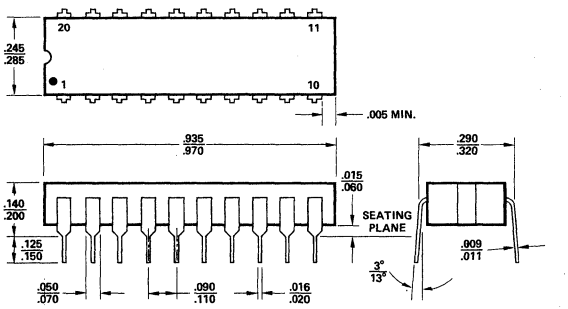
AC TEST LOAD



BPM-087

PHYSICAL DIMENSIONS
Dual-In-Line

20-Pin Hermetic



PROGRAMMING

The Am27S28 and Am27S29 are manufactured with a conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to the memory output after the $\overline{CS}$ input is at a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{CS}$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip is enabled and the output level is sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycles. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{CS}$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be continuously applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including V_{CC} should be removed for a period of 5 seconds after which programming may be resumed.

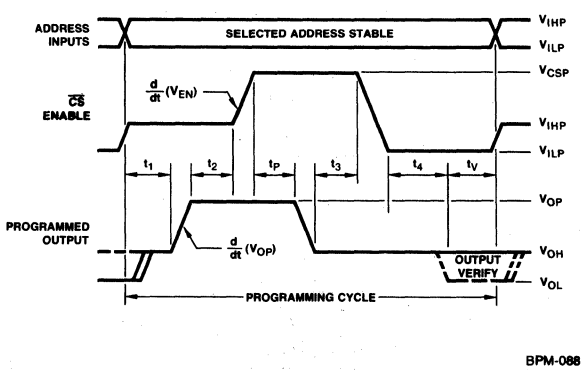
When all programming has been completed, the data content of the memory should be verified by sequentially reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min.	Max.	Units
V_{CCP}	V_{CC} During Programming	5.0	5.5	Volts
V_{IHP}	Input HIGH Level During Programming	2.4	5.5	Volts
V_{ILP}	Input LOW Level During Programming	0.0	0.45	Volts
V_{CSP}	$\overline{CS}$ Voltage During Programming	14.5	15.5	Volts
V_{OP}	Output Voltage During Programming	19.5	20.5	Volts
V_{ONP}	Voltage on Outputs Not to be Programmed	0.0	V_{CCP}+0.3	Volts
I_{ONP}	Current into Outputs Not to be Programmed		20	mA
$d(V_{OP})/dt$	Rate of Output Voltage Change	20	250	V/μsec
$d(V_{EN})/dt$	Rate of $\overline{CS}$ Voltage Change	100	1000	v/μsec
t_p	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5.0	15	msec

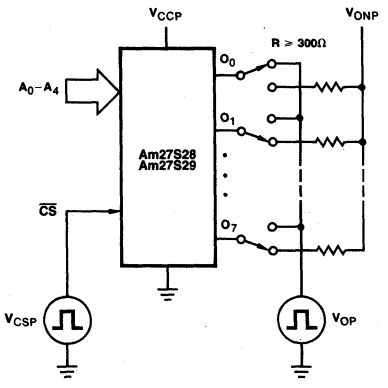
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e., not to the midpoints.
2. Delays t_1 through t_4 must be greater than 100ns; maximum delays of 1μsec are recommended to minimize heating during programming.
3. During t_V, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to V_{ONP} through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-088

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-089

Am27S30 • Am27S31

4096-Bit Generic Series Bipolar PROM

DISTINCTIVE CHARACTERISTICS

- High Speed – 55 ns max commercial range access time
- Excellent performance over full MIL and commercial ranges
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Fast chip select
- Access time tested with N^2 patterns
- Pin for pin replacements for industry standard products
- Common Generic PROM series electrical characteristics and simple programming procedures.

GENERIC SERIES CHARACTERISTICS

The Am27S30 and Am27S31 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test rows are preprogrammed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliabilty. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

ORDERING INFORMATION

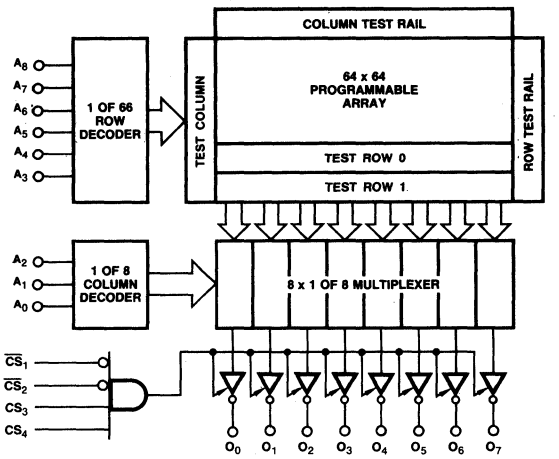
Package Type	Temperature Range	Order Number
	Open Collectors	
Hermetic DIP	0°C to +75°C	AM27S30DC
Hermetic DIP	−55°C to +125°C	Am27S30DM
	Three-State Outputs	
Hermetic DIP	0°C to +75°C	AM27S31DC
Hermetic DIP	−55°C to +125°C	AM27S31DM

FUNCTIONAL DESCRIPTION

The Am27S30 and Am27S31 are high speed electrically programmable Schottky read only memories. Organized in the industry standard 512 x 8 configuration, they are available in both open collector Am27S30 and three-state Am27S31 output versions. After programming, stored information is read on outputs O_0-O_7 by applying unique binary addresses to A_0-A_8 and holding $\overline{CS}_1$ and $\overline{CS}_2$ LOW and CS_3 and CS_4 HIGH. All other valid input conditions on $\overline{CS}_1$, $\overline{CS}_2$, CS_3 and CS_4 place O_0-O_7 into the OFF or high impedance state.

4

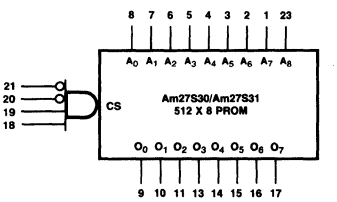
BLOCK DIAGRAM



BPM-114

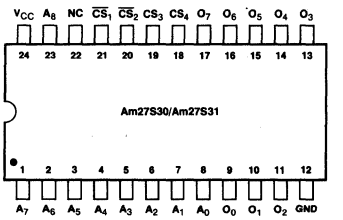
LOGIC SYMBOL

V_{CC} = Pin 24
GND = Pin 12
(Pin 22 Open)



BPM-115

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BPM-116

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential (Pin 24 to Pin 12) Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +V_{CC} max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5V to +5.5V
DC Input Current	−30mA to +5mA

OPERATING RANGE

COM'L	Am27S30XC, Am27S31XC	T_A = 0°C to +75°C	V_{CC} = 5.0V ±5%
MIL	Am27S30XM, Am27S31XM	T_A = −55°C to +125°C	V_{CC} = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions	Min.	Typ. (Note 1)	Max.	Units
V_{OH} (Am27S31 only)	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −2.0mA V_{IN} = V_{IH} or V_{IL}	2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}			0.50	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs	2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs			0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.45V		−0.010	−0.250	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V			25	µA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V			1.0	mA
I_{SC} (Am27S31 only)	Output Short Circuit Current	V_{CC} = MAX., V_{OUT} = 0.0V (Note 2)	−20	−40	−90	mA
I_{CC}	Power Supply Current	All inputs = GND V_{CC} = MAX.		115	175	mA
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA			−1.2	Volts
I_{CEX}	Output Leakage Current	V_{CC} = MAX, $V_{\overline{CS1}}$ = 2.4V; Am27S31 only; V_O = 4.5V			40	µA
		V_{CC} = MAX, $V_{\overline{CS1}}$ = 2.4V; Am27S31 only; V_O = 2.4V			40	µA
		V_{CC} = MAX, $V_{\overline{CS1}}$ = 2.4V; Am27S31 only; V_O = 0.4V			−40	µA
C_{IN}	Input Capacitance	V_{IN} = 2.0V @ f = 1 MHz (Note 3)		4		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0V @ f = 1 MHz (Note 3)		8		pF

Notes: 1. Typical limits are at V_{CC} = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but periodically sampled.

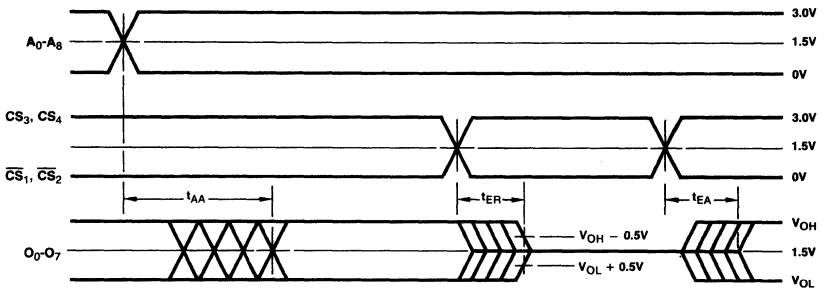
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ 5V 25°C	Max COM'L	Max MIL	Units
t_{AA}	Address Access Time	AC Test Load (See Notes 1-3)	35	55	70	ns
t_{EA}	Enable Access Time		15	25	30	ns
t_{ER}	Enable Recovery Time		15	25	30	ns

Notes:
1. t_{AA} is tested with switch S_1 closed and C_L = 30pF.
2. For open collector outputs, t_{EA} and t_{ER} are tested with S_1 closed to the 1.5V output level. C_L = 30pF.
3. For three state outputs, t_{EA} is tested with C_L = 30pF to the 1.5V level; S_1 is open for high impedance to HIGH tests and closed for high impedance to LOW tests. t_{ER} is tested with C_L = 5pF. HIGH to high impedance tests are made with S_1 open to an output voltage of V_{OH} – 0.5V; LOW to high impedance tests are made with S_1 closed to the V_{OL} + 0.5V level.

SWITCHING WAVEFORMS



Note: Level on output while chip is disabled is determined externally.

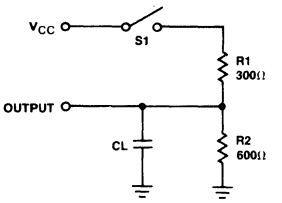
BPM-117

KEY TO TIMING DIAGRAM

WAVEFORM	INPUTS	OUTPUTS
——— / ———	MUST BE STEADY	WILL BE STEADY
\	MAY CHANGE FROM H TO L	WILL BE CHANGING FROM H TO L
/	MAY CHANGE FROM L TO H	WILL BE CHANGING FROM L TO H

WAVEFORM	INPUTS	OUTPUTS
⋈	DON'T CARE; ANY CHANGE PERMITTED	CHANGING; STATE UNKNOWN
⋈	DOES NOT APPLY	CENTER LINE IS HIGH IMPEDANCE "OFF" STATE

AC TEST LOAD



BPM-118

PROGRAMMING

The Am27S30 and Am27S31 are manufactured with a conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to the memory output after the $\overline{CS}_1$ input is at a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{CS}_1$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip is enabled and the output level is sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycles. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{CS}_1$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be continuously applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including V_{CC} should be removed for a period of 5 seconds after which programming may be resumed.

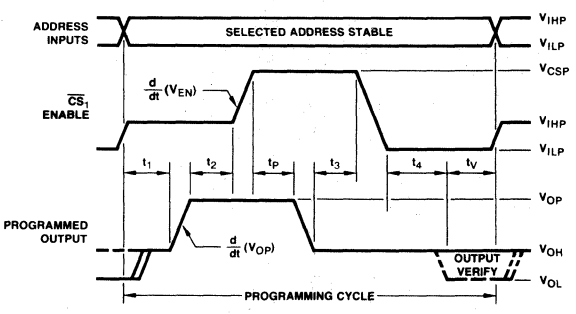
When all programming has been completed, the data content of the memory should be verified by sequentially reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min.	Max.	Units
V_{CCP}	V_{CC} During Programming	5.0	5.5	Volts
V_{IHP}	Input HIGH Level During Programming	2.4	5.5	Volts
V_{ILP}	Input LOW Level During Programming	0.0	0.45	Volts
V_{CSP}	$\overline{CS}_1$ Voltage During Programming	14.5	15.5	Volts
V_{OP}	Output Voltage During Programming	19.5	20.5	Volts
V_{ONP}	Voltage on Outputs Not to be Programmed	0.0	V_{CCP}+0.3	Volts
I_{ONP}	Current into Outputs Not to be Programmed		20	mA
$d(V_{OP})/dt$	Rate of Output Voltage Change	20	250	V/μsec
$d(V_{EN})/dt$	Rate of $\overline{CS}_1$ Voltage Change	100	1000	v/μsec
t_p	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5.0	15	msec

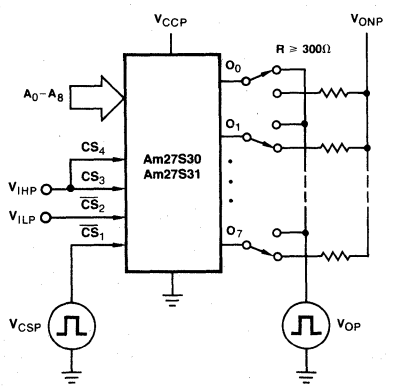
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e., not to the midpoints.
2. Delays t_1 through t_4 must be greater than 100ns; maximum delays of 1μsec are recommended to minimize heating during programming.
3. During t_V, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to V_{ONP} through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-119

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-120

Am27S32 • Am27S33

4096-Bit Generic Series Bipolar PROM

DISTINCTIVE CHARACTERISTICS

- High Speed – 55ns max commercial range access time
- Excellent performance over full MIL and commercial ranges
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Fast chip select
- Access time tested with N^2 patterns
- Pin for pin replacements for industry standard products
- Common Generic PROM series electrical characteristics and simple programming procedures.

GENERIC SERIES CHARACTERISTICS

The Am27S32 and Am27S33 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test words are pre-programmed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliabilty. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

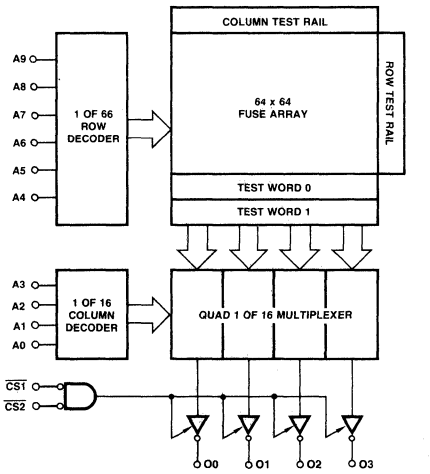
ORDERING INFORMATION

Package Type	Temperature Range	Order Number
	Open Collectors	
Hermetic DIP	0°C to +75°C	AM27S32DC
Hermetic DIP	−55°C to +125°C	AM27S32DM
	Three-State Outputs	
Hermetic DIP	0°C to +75°C	AM27S33DC
Hermetic DIP	−55°C to 125°C	AM27S33DM

FUNCTIONAL DESCRIPTION

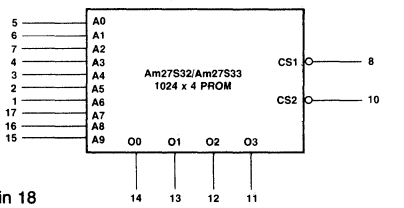
The Am27S32 and Am27S33 are high speed electrically programmable Schottky read only memories. Organized in the industry standard 1024 x 4 configuration, they are available in both open collector Am27S32 and three-state Am27S33 output versions. After programming, stored information is read on outputs O0–O3 by applying unique binary addresses to A0–A9 and holding the chip select inputs, $\overline{CS1}$ and $\overline{CS2}$, LOW. If either chip select input goes to a logic HIGH, O0–O3 go to the off or high impedance state.

BLOCK DIAGRAM



BPM-090

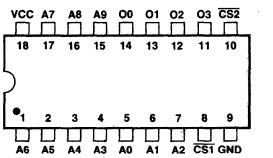
LOGIC SYMBOL



VCC = Pin 18
GND = Pin 9

BPM-091

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BPM-092

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential (Pin 18 to Pin 9) Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +VCC max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5V to +5.5V
DC Input Current	−30mA to +5mA

OPERATING RANGE

COM'L	Am27S32XC, Am27S33XC	T_A = 0°C to +75°C	VCC = 5.0V ±5%
MIL	Am27S32XM, Am27S33XM	T_A = −55°C to +125°C	VCC = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions			Min.	Typ. (Note 1)	Max.	Units
VOH (Am27S33 only)	Output HIGH Voltage	VCC = MIN., IOH = −2.0mA VIN = VIH or VIL			2.4			Volts
VOL	Output LOW Voltage	VCC = MIN., IOL = 16mA VIN = VIH or VIL					0.45	Volts
VIH	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs			2.0			Volts
VIL	Input LOW Level	Guaranteed input logical LOW voltage for all inputs					0.8	Volts
IIL	Input LOW Current	VCC = MAX., VIN = 0.45V				−0.020	−0.250	mA
IIH	Input HIGH Current	VCC = MAX., VIN = 2.7V					25	μA
II	Input HIGH Current	VCC = MAX., VIN = 5.5V					1.0	mA
ISC (Am27S33 only)	Output Short Circuit Current	VCC = MAX., VOUT = 0.0V (Note 2)			−20	−40	−90	mA
ICC	Power Supply Current	All inputs = GND VCC = MAX.		COM'L		105	140	mA
				MIL		105	145	
VI	Input Clamp Voltage	VCC = MIN., IIN = −18mA					−1.2	Volts
ICEX	Output Leakage Current	VCC = MAX. $\overline{VCS1}$ = 2.4V		VO = 4.5V			40	μA
			Am27S33 only	VO = 2.4V			40	
				VO = 0.4V			−40	
CIN	Input Capacitance	VIN = 2.0V @ f = 1MHz (Note 3)				5		pF
COUT	Output Capacitance	VOUT = 2.0V @ f = 1MHz (Note 3)				12		

Notes: 1. Typical limits are at VCC = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but are periodically sampled.

SWITCHING CHARACTERISTICS OVER OPERATING RANGE

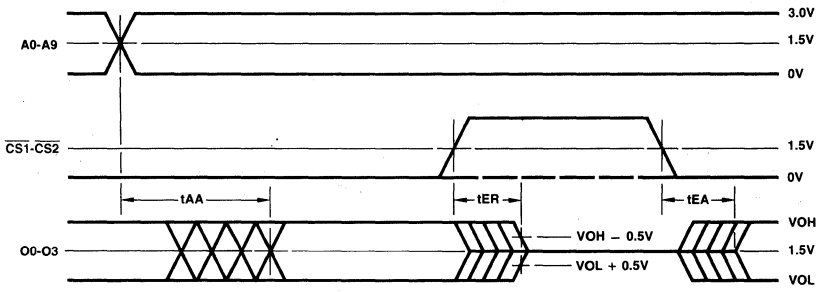
PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ 5V 25°C	Max COM'L	Max MIL	Units
t_{AA}	Address Access Time	AC Test Load (See Notes 1-3)	38	55	70	ns
t_{EA}	Enable Access Time		10	25	30	ns
t_{ER}	Enable Recovery Time		10	25	30	ns

Notes: 1. tAA is tested with switch S1 closed and CL = 30pF.
2. For open collector outputs, tEA and tER are tested with S1 closed to the 1.5V output level. CL = 30pF.
3. For three state outputs, tEA is tested with CL = 30pF to the 1.5V level; S1 is open for high impedance to HIGH tests and closed for high impedance to LOW tests. tER is tested with CL = 5pF. HIGH to high impedance tests are made with S1 open to an output voltage of VOH − 0.5V; LOW to high impedance tests are made with S1 closed to the VOL + 0.5V level.

4

SWITCHING CHARACTERISTICS



Note: Level on output while either $\overline{CS}$ is HIGH is determined externally.

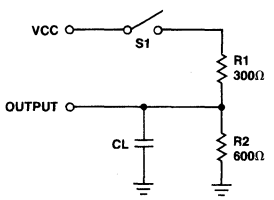
BPM-093

KEY TO TIMING DIAGRAM

WAVEFORM	INPUTS	OUTPUTS
———	MUST BE STEADY	WILL BE STEADY
▨	MAY CHANGE FROM H TO L	WILL BE CHANGING FROM H TO L
▨	MAY CHANGE FROM L TO H	WILL BE CHANGING FROM L TO H

WAVEFORM	INPUTS	OUTPUTS
▨	DON'T CARE; ANYCHANGE PERMITTED	CHANGING; STATE UNKNOWN
▨	DOES NOT APPLY	CENTER LINE IS HIGH IMPEDANCE "OFF" STATE

AC TEST LOAD



BPM-094

PROGRAMMING

The Am27S32 and Am27S33 are manufactured with conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. To program the device, the fusible links are selectively opened.

The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to one memory output after the $\overline{CS}_1$ input is at a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{CS}_1$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip enabled, and the output level sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycle. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{CS}_1$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including VCC should be removed for a period of 5 seconds after which programming may be resumed.

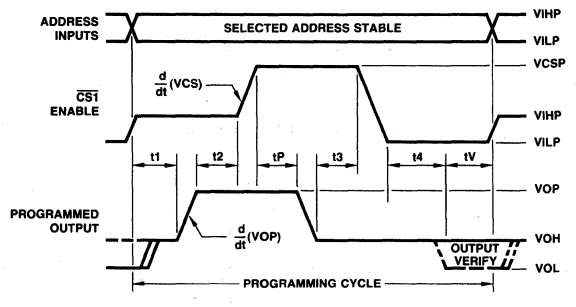
When all programming has been completed, the data content of the memory should be verified by sequentially reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min.	Max.	Units
VCCP	VCC During Programming	5.0	5.5	Volts
VIHP	Input HIGH Level During Programming	2.4	5.5	Volts
VILP	Input LOW Level During Programming	0.0	0.45	Volts
VCSP	$\overline{CS}_1$ Voltage During Programming	14.5	15.5	Volts
VOP	Output Voltage During Programming	19.5	20.5	Volts
VONP	Voltage on Outputs. Not to be Programmed	0	VCCP+0.3	Volts
IONP	Current into Outputs Not to be Programmed		20	mA
d(VOP)/dt	Rate of Output Voltage Change	20	250	V/μsec
d(VCS)/dt	Rate of $\overline{CS}_1$ Voltage Change	100	1000	V/μsec
tP	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5	15	msec

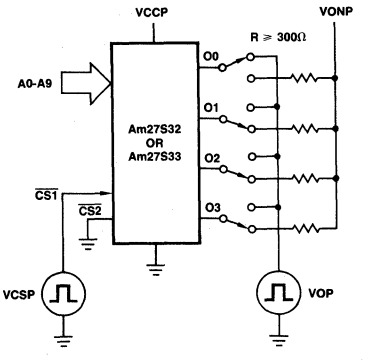
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e. not to the midpoints.
2. Delays t1, t2, t3 and t4 must be greater than 100 ns; maximum delays of 1 μsec are recommended to minimize heating during programming.
3. During tv, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to VONP through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-095

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-096

Am27S180 • Am27S181

8192-Bit Generic Series Bipolar PROM

PRELIMINARY DATA

DISTINCTIVE CHARACTERISTICS

- High Speed – 60ns max commercial range access time
- Excellent performance over full MIL and commercial ranges
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Fast chip select
- Access time tested with N^2 patterns
- Pin for pin replacements for industry standard products
- Common Generic PROM series electrical characteristics and simple programming procedures.

GENERIC SERIES CHARACTERISTICS

The Am27S180 and Am27S181 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test words are pre-programmed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliabilty. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

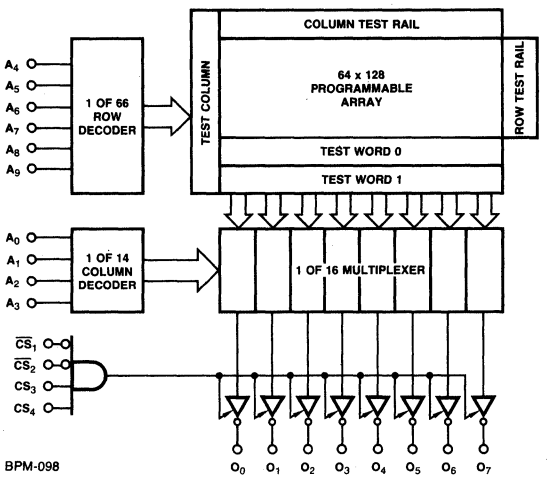
ORDERING INFORMATION

Package Type	Temperature Range	Order Number
	Open Collectors	
Hermetic DIP	0°C to +75°C	AM27S180DC
Hermetic DIP	−55°C to +125°C	AM27S180DM
	Three-State Outputs	
Hermetic DIP	0°C to +75°C	AM27S181DC
Hermetic DIP	−55°C to +125°C	AM27S181DM

FUNCTIONAL DESCRIPTION

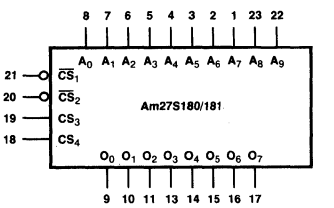
The Am27S180 and Am27S181 are high speed electrically programmable Schottky read only memories. Organized in the industry standard 1024 x 8 configuration, they are available in both open collector Am27S180 and three-state Am27S181 output versions. After programming, stored information is read on outputs O_0-O_7 by applying unique binary addresses to A_0-A_9 and enabling the chip ($\overline{CS}_1$, $\overline{CS}_2$, low and CS_3, CS_4 high). Changes of chip select input levels disables the outputs causing them to go to the off or high impedance state.

BLOCK DIAGRAM



BPM-098

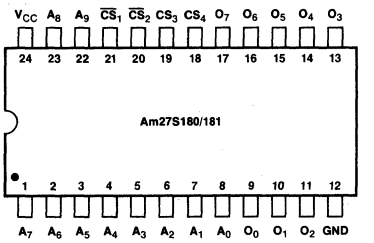
LOGIC DIAGRAM



BPM-099

V_{CC} = Pin 24
GND = Pin 12

CONNECTION DIAGRAM
Top View



BPM-100

Note: Pin 1 is marked for orientation.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Temperature (Ambient) Under Bias	−55 to +125°C
Supply Voltage to Ground Potential (Pin 24 to Pin 12) Continuous	−0.5 to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +V_{CC} max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5 to +5.5V
DC Input Current	−30 to +5mA

OPERATING RANGE

COM'L	Am27S180XC, Am27S181XC	T_A = 0 to 75°C	V_{CC} = 5.0V ±5%
MIL	Am27S180XM, Am27S181XM	T_C = −55 to +125°C	V_{CC} = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions	Min.	Typ. (Note 1)	Max.	Units
V_{OH} (Am27S181 only)	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −2.0mA V_{IN} = V_{IH} or V_{IL}	2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}		0.38	0.50	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs	2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs			0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.45V		−0.010	−0.250	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V			25	μA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V			1.0	mA
I_{SC} (Am27S181 only)	Output Short Circuit Current	V_{CC} = MAX., V_{OUT} = 0.0V (Note 2)	−20	−40	−90	mA
I_{CC}	Power Supply Current	All inputs = GND V_{CC} = MAX.		120	185	mA
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA			−1.2	Volts
I_{CEX}	Output Leakage Current	V_{CC} = MAX., $V_{\overline{CS}1,2}$ = 2.4V, $V_{CS3,4}$ = 0.4V; Am27S181 Only; V_O = 4.5V			40	μA
		V_{CC} = MAX., $V_{\overline{CS}1,2}$ = 2.4V, $V_{CS3,4}$ = 0.4V; Am27S181 Only; V_O = 2.4V			40	μA
		V_{CC} = MAX., $V_{\overline{CS}1,2}$ = 2.4V, $V_{CS3,4}$ = 0.4V; Am27S181 Only; V_O = 0.4V			−40	μA
C_{IN}	Input Capacitance	V_{IN} = 2.0V @ f = 1MHz (Note 3)		5		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0V @ f = 1MHz (Note 3)		12		pF

Notes: 1. Typical limits are at V_{CC} = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but are periodically sampled.

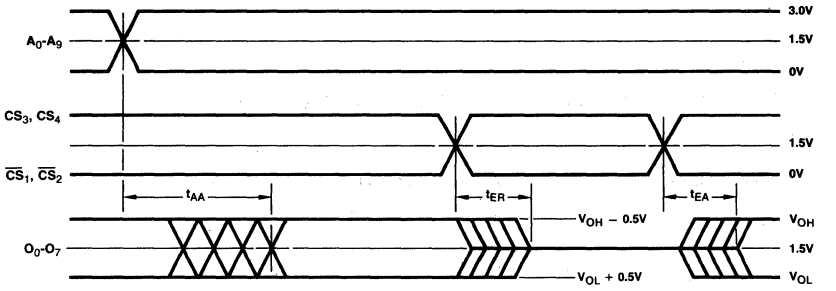
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ 5V 25°C	Max COM'L	Max MIL	Units
t_{AA}	Address Access Time	AC Test Load (See Notes 1-3)	40	60	80	ns
t_{EA}	Enable Access Time		20	40	50	ns
t_{ER}	Enable Recovery Time		20	40	50	ns

Notes: 1. t_{AA} is tested with switch S_1 closed and C_L = 30pF.
2. For open collector outputs, t_{EA} and t_{ER} are tested with S_1 closed to the 1.5V output level. C_L = 30pF.
3. For three state outputs, t_{EA} is tested with C_L = 30pF to the 1.5V level; S_1 is open for high impedance to HIGH tests and closed for high impedance to LOW tests. t_{ER} is tested with C_L = 5pF. HIGH to high impedance tests are made with S_1 open to an output voltage of V_{OH} − 0.5V; LOW to high impedance tests are made with S_1 closed to the V_{OL} + 0.5V level.

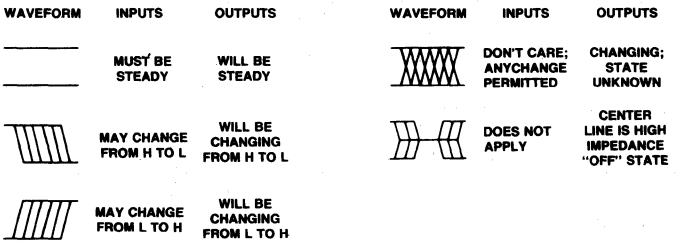
SWITCHING CHARACTERISTICS



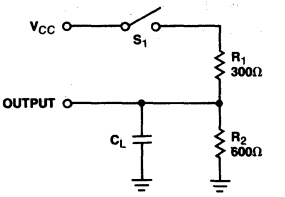
Note: Level on output while chip is disabled is determined externally.

BPM-101

KEY TO TIMING DIAGRAM



AC TEST LOAD



BPM-102

PROGRAMMING

The Am27S180 and Am27S181 are manufactured with a conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. To program the device, the fusible links are selectively opened.

The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to one memory output after the $\overline{CS}_1$ input is at a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{CS}_1$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip enabled, and the output level sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycles. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{CS}_1$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including V_{CC} should be removed for a period of 5 seconds after which programming may be resumed.

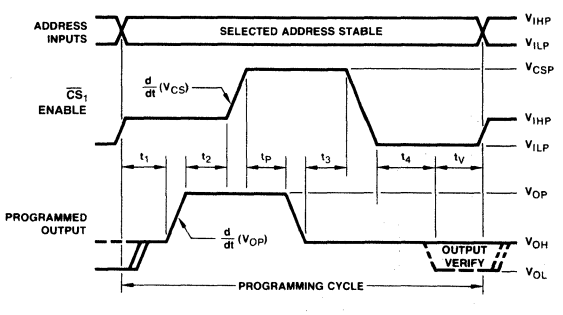
When all programming has been completed, the data content of the memory should be verified by sequentially reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min.	Max.	Units
V_{CCP}	V_{CC} During Programming	5.0	5.5	Volts
V_{IHP}	Input HIGH Level During Programming	2.4	5.5	Volts
V_{ILP}	Input LOW Level During Programming	0.0	0.45	Volts
V_{CSP}	$\overline{CS}_1$ Voltage During Programming	14.5	15.5	Volts
V_{OP}	Output Voltage During Programming	19.5	20.5	Volts
V_{ONP}	Voltage on Outputs Not to be Programmed	0	V_{CCP}+0.3	Volts
I_{ONP}	Current into Outputs Not to be Programmed		20	mA
$d(V_{OP})/dt$	Rate of Output Voltage Change	20	250	V/μsec
$d(V_{CS})/dt$	Rate of $\overline{CS}_1$, Voltage Change	100	1000	V/μsec
t_P	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5.0	15	msec

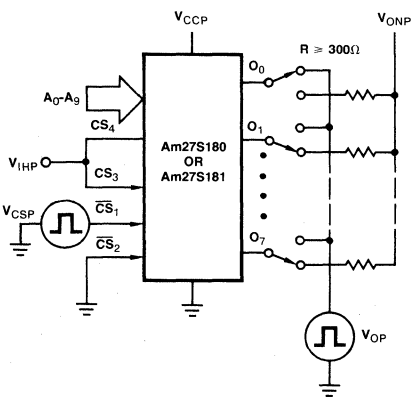
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e., not to the midpoints.
2. Delays t_1, t_2, t_3 and t_4 must be greater than 100 ns; maximum delays of 1 μsec are recommended to minimize heating during programming.
3. During t_V, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to V_{ONP} through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-103

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-104

Am9016

16,384 x 1 Dynamic R/W Random Access Memory

DISTINCTIVE CHARACTERISTICS

- High density 16k x 1 organization
- Direct replacement for MK4116
- Low maximum power dissipation — 462mW active, 20mW standby
- High speed operation — 150ns access, 320ns cycle
- ±10% tolerance on standard +12, +5, −5 voltages
- TTL compatible interface signals
- Three-state output
- RAS only, RMW and Page mode clocking options
- 128 cycle refreshing
- Unlatched data output
- Standard 16-pin, .3 inch wide dual in-line package
- Double poly N-channel silicon gate MOS technology
- 100% MIL-STD-883 reliability assurance testing

GENERAL DESCRIPTION

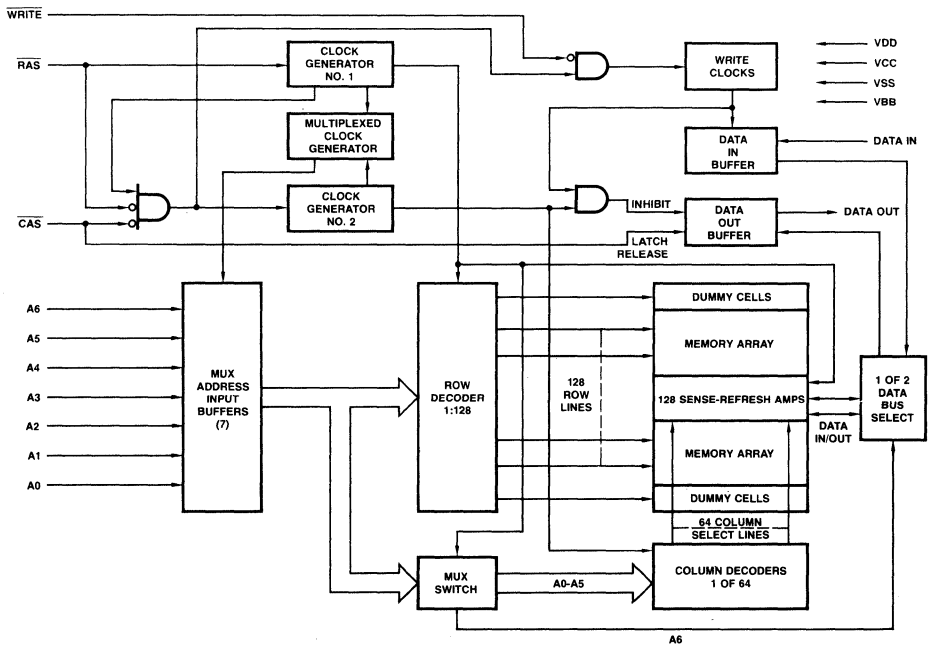
The Am9016 is a high speed, 16k-bit, dynamic, read/write random access memory. It is organized as 16,384 words by 1 bit per word and is packaged in a standard 16-pin DIP. The basic memory element is a single transistor cell that stores charge on a small capacitor. This mechanism requires periodic refreshing of the memory cells to maintain stored information.

All input signals, including the two clocks, are TTL compatible. The Row Address Strobe ($\overline{RAS}$) loads the row address and the Column Address Strobe ($\overline{CAS}$) loads the column address. The row and column address signals share 7 input lines. Active cycles are initiated when $\overline{RAS}$ goes low, and standby mode is entered when $\overline{RAS}$ goes high. In addition to normal read and write cycles, other types of operations are available to improve versatility, performance, and power dissipation.

The three-state output buffer turns on when the column access time has elapsed and turns off after $\overline{CAS}$ goes high. Input and output data are the same polarity.

4

BLOCK DIAGRAM

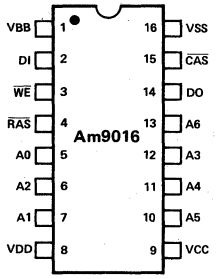


MOS-190

ORDERING INFORMATION

Ambient Temperature	Package Type	Access Time			
		300ns	250ns	200ns	150ns
0°C ≤ T_A ≤ +70°C	Hermetic DIP	AM9016CDC	AM9016DDC	AM9016EDC	AM9016FDC
	Molded DIP	AM9016CPC	AM9016DPC	AM9016EPC	AM9016FPC

CONNECTION DIAGRAM



Top View

Pin 1 is marked for orientation.

A0 – A6	ADDRESS INPUTS
$\overline{CAS}$	COLUMN ADDRESS STROBE
DI	DATA IN
DO	DATA OUT
$\overline{RAS}$	ROW ADDRESS STROBE
VDD	POWER (+12V)
VCC	POWER (+5V)
VSS	GROUND
VBB	POWER (–5V)
$\overline{WE}$	WRITE ENABLE

MOS-191

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	–65°C to +150°C
Ambient Temperature Under Bias	0°C to +70°C
Voltage on Any Pin Relative to VBB	–0.5V to +20V
VDD and VCC Supply Voltages with Respect to VSS	–1.0V to +15.0V
VBB – VSS (VDD – VSS > 0V)	0V
Power Dissipation	1.0W
Short Circuit Output Current	50mA

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulation of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling, and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Ambient Temperature	VDD	VCC	VSS	VBB
$0°C \leq T_A \leq +70°C$	+12V ±10%	+5V ±10%	0	–5.0V ±10%

ELECTRICAL CHARACTERISTICS over operating range (Notes 1, 11)

				Am9016X			
Parameters	**Description**		**Test Conditions**	**Min.**	**Typ.**	**Max.**	**Units**
VOH	Output HIGH Voltage		IOH = –5.0mA	2.4		VCC	Volts
VOL	Output LOW Voltage		IOL = 4.2mA	VSS		0.40	Volts
VIH	Input HIGH Voltage for Address, Data In			2.4		7.0	Volts
VIHC	Input HIGH Voltage for $\overline{CAS}$, $\overline{RAS}$, $\overline{WE}$			2.7		7.0	Volts
VIL	Input LOW Voltage			–1.0		0.80	Volts
IIX	Input Load Current		VSS ≤ VI ≤ 7V	–10		10	µA
IOZ	Output Leakage Current		VSS ≤ VO ≤ VCC, Output OFF	–10		10	µA
ICC	VCC Supply Current		Output OFF (Note 4)	–10		10	µA
IBB	VBB Supply Current, Average		Standby, $\overline{RAS} \geq$ VIHC			100	µA
			Operating, Minimum Cycle Time			200	
IDD	VDD Supply Current, Average	Operating IDD1	$\overline{RAS}$ Cycling, $\overline{CAS}$ Cycling, Minimum Cycle Times			35	mA
		Page Mode IDD4	$\overline{RAS} \leq$ VIL, $\overline{CAS}$ Cycling, Minimum Cycle Times			27	
		$\overline{RAS}$ Only Refresh IDD3	$\overline{RAS}$ Cycling, $\overline{CAS} \geq$ VIHC, Minimum Cycle Times			27	
		Standby IDD2	$\overline{RAS} \geq$ VIHC			1.5	
CI	Input Capacitance	$\overline{RAS}$, $\overline{CAS}$, $\overline{WE}$	Inputs at 0V, f = 1MHz, Nominal Supply Voltages			10	pF
		Address, Data In				5.0	
CO	Output Capacitance		Output OFF			7.0	

SWITCHING CHARACTERISTICS over operating range (Notes 2, 3, 5, 10)

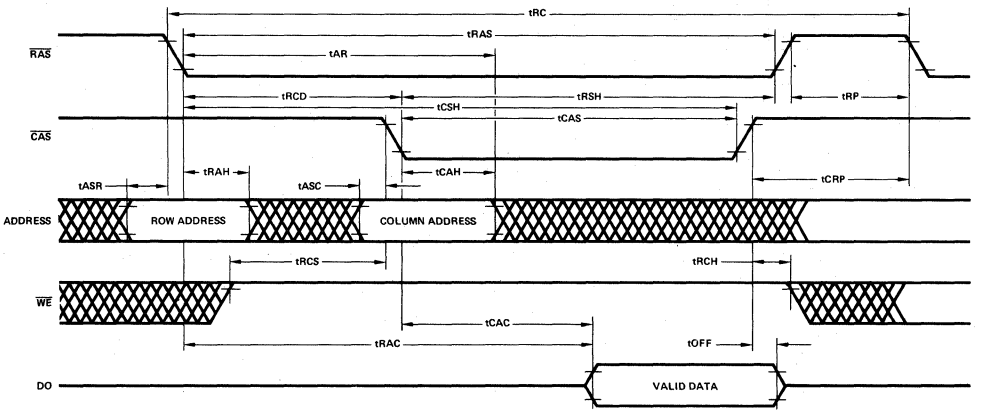
Parameters	Description	Am9016C		Am9016D		Am9016E		Am9016F		Units
		Min	Max	Min	Max	Min	Max	Min	Max	
tAR	$\overline{RAS}$ LOW to Column Address Hold Time	200		160		120		95		ns
tASC	Column Address Set-up Time	−10		−10		−10		−10		ns
tASR	Row Address Set-up Time	0		0		0		0		ns
tCAC	Access Time from $\overline{CAS}$ (Note 6)		185		165		135		100	ns
tCAH	$\overline{CAS}$ LOW to Column Address Hold Time	85		75		55		45		ns
tCAS	$\overline{CAS}$ Pulse Width	185	10,000	165	10,000	135	10,000	100	10,000	ns
tCP	Page Mode $\overline{CAS}$ Precharge Time	100		100		80		60		ns
tCRP	$\overline{CAS}$ to $\overline{RAS}$ Precharge Time	−20		−20		−20		−20		ns
tCSH	$\overline{CAS}$ Hold Time	300		250		200		150		ns
tCWD	$\overline{CAS}$ LOW to $\overline{WE}$ LOW Delay (Note 9)	145		125		95		70		ns
tCWL	$\overline{WE}$ LOW to $\overline{CAS}$ HIGH Set-up Time	100		85		70		50		ns
tDH	$\overline{CAS}$ LOW or $\overline{WE}$ LOW to Data In Valid Hold Time (Note 7)	85		75		55		45		ns
tDHR	$\overline{RAS}$ LOW to Data In Valid Hold Time	200		160		120		95		ns
tDS	Data In Stable to $\overline{CAS}$ LOW or $\overline{WE}$ LOW Set-up Time (Note 7)	0		0		0		0		ns
tOFF	$\overline{CAS}$ HIGH to Output OFF Delay	0	60	0	60	0	50	0	40	ns
tPC	Page Mode Cycle Time	295		275		225		170		ns
tRAC	Access Time from $\overline{RAS}$ (Note 6)		300		250		200		150	ns
tRAH	$\overline{RAS}$ LOW to Row Address Hold Time	45		35		25		20		ns
tRAS	$\overline{RAS}$ Pulse Width	300	10,000	250	10,000	200	10,000	150	10,000	ns
tRC	Random Read or Write Cycle Time	460		410		375		320		ns
tRCD	$\overline{RAS}$ LOW to $\overline{CAS}$ LOW Delay (Note 6)	35	115	35	85	25	65	20	50	ns
tRCH	Read Hold Time	0		0		0		0		ns
tRCS	Read Set-up Time	0		0		0		0		ns
tREF	Refresh Interval		2		2		2		2	ms
tRMW	Read Modify Write Cycle Time	600		500		405		320		ns
tRP	RAS Precharge Time	150		150		120		100		ns
tRSH	$\overline{CAS}$ LOW to $\overline{RAS}$ HIGH Delay	185		165		135		100		ns
tRWC	Read/Write Cycle Time	525		425		375		320		ns
tRWD	$\overline{RAS}$ LOW to $\overline{WE}$ LOW Delay (Note 9)	260		210		160		120		ns
tRWL	$\overline{WE}$ LOW to $\overline{RAS}$ HIGH Set-up Time	100		85		70		50		ns
tT	Transition Time	3	50	3	50	3	50	3	35	ns
tWCH	Write Hold Time	85		75		55		45		ns
tWCR	$\overline{RAS}$ LOW to Write Hold Time	200		160		120		95		ns
tWCS	$\overline{WE}$ LOW to $\overline{CAS}$ LOW Set-up Time (Note 9)	−20		−20		−20		−20		ns
tWP	Write Pulse Width	85		75		55		45		ns

NOTES

1. Typical values are for T_A = 25°C, nominal supply voltages and nominal processing parameters.
2. Signal transition times are assumed to be 5ns. Transition times are measured between specified high and low logic levels.
3. Timing reference levels for both input and output signals are the specified worst-case logic levels.
4. VCC is used in the output buffer only. ICC will therefore depend only on leakage current and output loading. When the output is ON and at a logic high level, VCC is connected to the Data Out pin through an equivalent resistance of approximately 135Ω. In standby mode VCC may be reduced to zero without affecting stored data or refresh operations.
5. Output loading is two standard TTL loads plus 100pF capacitance.
6. Both $\overline{RAS}$ and $\overline{CAS}$ must be low to read data. Access timing will depend on the relative positions of their falling edges. When tRCD is less than the maximum value shown, access time depends on $\overline{RAS}$ and tRAC governs. When tRCD is more than the maximum value shown access time depends on $\overline{CAS}$ and tCAC governs. The maximum value listed for tRCD is shown for reference purposes only and does not restrict operation of the part.
7. Timing reference points for data input set-up and hold times will depend on what type of write cycle is being performed and will be the later falling edge of $\overline{CAS}$ or $\overline{WE}$.
8. At least eight initialization cycles that exercise $\overline{RAS}$ should be performed after power-up and before valid operations are begun.
9. The tWCS, tRWD and tCWD parameters are shown for reference purposes only and do not restrict the operating flexibility of the part. When the falling edge of $\overline{WE}$ follows the falling edge of $\overline{CAS}$ by at most tWCS, the data output buffer will remain off for the whole cycle and an "early write" cycle is defined. When the falling edge of $\overline{WE}$ follows the falling edges of $\overline{RAS}$ and $\overline{CAS}$ by at least tRWD and tCWD respectively, the Data Out from the addressed cell will be valid at the access time and a "read/write" cycle is defined. The falling edge of $\overline{WE}$ may also occur at intermediate positions, but the condition and validity of the Data Out signal will not be known.
10. Switching characteristics are listed in alphabetical order.
11. All voltages referenced to VSS.

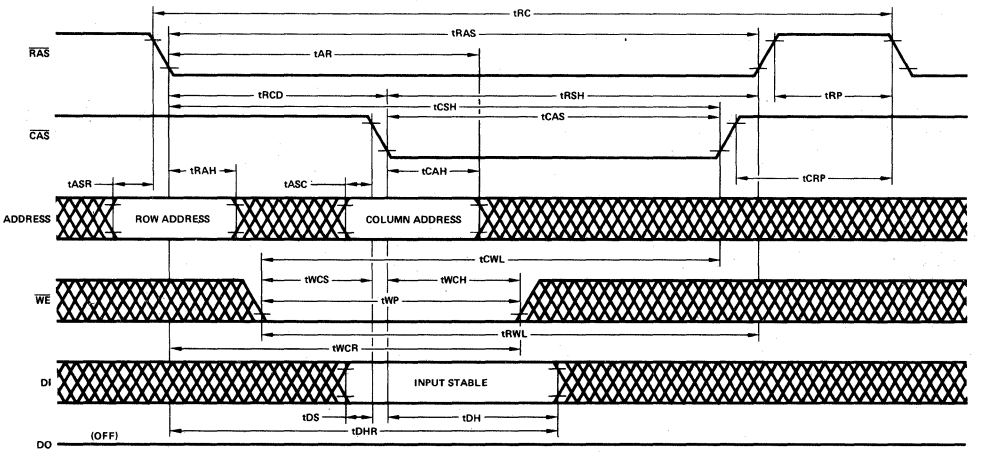
SWITCHING WAVEFORMS

READ CYCLE



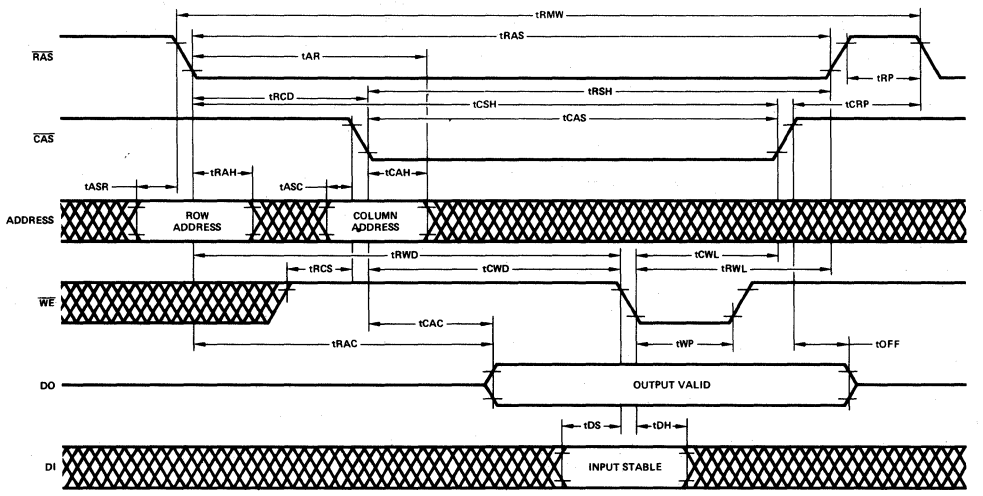
MOS-192

WRITE CYCLE (EARLY WRITE)



MOS-193

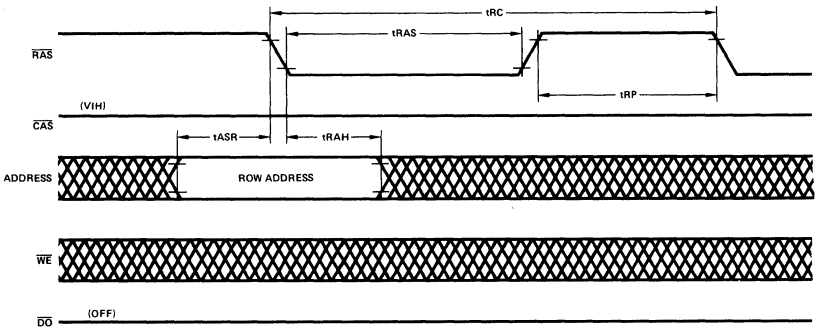
READ-WRITE/READ-MODIFY-WRITE CYCLE



MOS-194

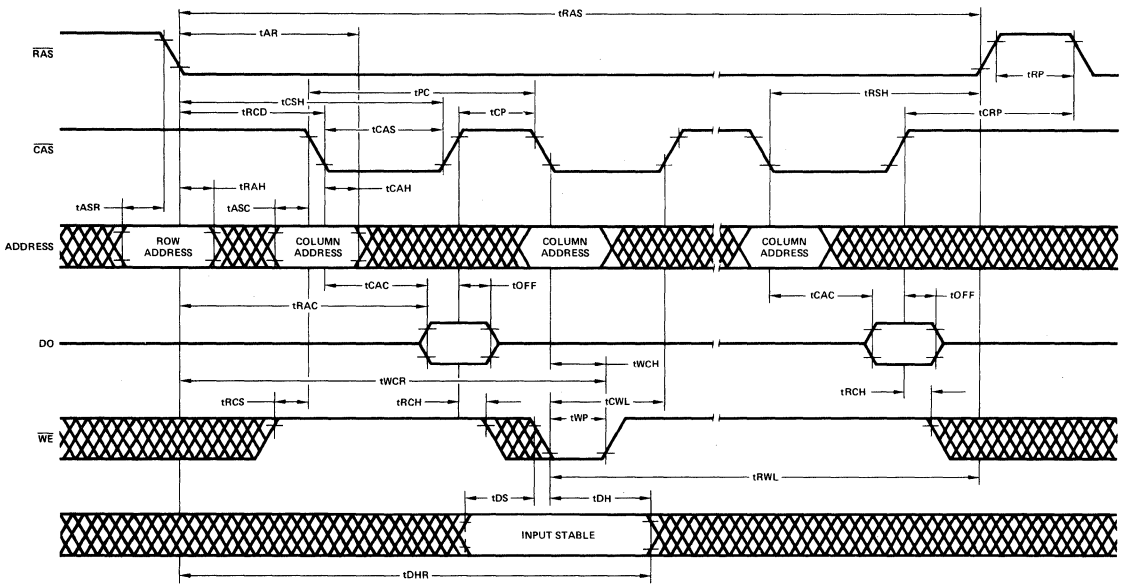
SWITCHING WAVEFORMS (Cont.)

$\overline{RAS}$ ONLY REFRESH CYCLE



MOS-195

PAGE MODE CYCLE



MOS-196

APPLICATION INFORMATION

The Am9016 electrical connections are such that if power is applied with the device installed upside down it will be permanently damaged. Precautions should be taken to avoid this mishap.

OPERATING CYCLES

Random read operations from any location hold the $\overline{WE}$ line high and follow this sequence of events:

1) The row address is applied to the address inputs and $\overline{RAS}$ is switched low.
2) After the row address hold time has elapsed, the column address is applied to the address inputs and $\overline{CAS}$ is switched low.
3) Following the access time, the output will turn on and valid read data will be present. The data will remain valid as long as $\overline{CAS}$ is low.
4) $\overline{CAS}$ and $\overline{RAS}$ are then switched high to end the operation. A new cycle cannot begin until the precharge period has elapsed.

Random write operations follow the same sequence of events, except that the $\overline{WE}$ line is low for some portion of the cycle. If the data to be written is available early in the cycle, it will usually be convenient to simply have $\overline{WE}$ low for the whole write operation.

Sequential Read and Write operations at the same location can be designed to save time because re-addressing is not necessary. A read/write cycle holds $\overline{WE}$ high until a valid read is established and then strobes new data in with the falling edge of $\overline{WE}$.

After the power is first applied to the device, the internal circuit requires execution of at least eight initialization cycles which exercise $\overline{RAS}$ before valid memory accesses are begun.

ADDRESSING

14 address bits are required to select one location out of the 16,384 cells in the memory. Two groups of 7 bits each are multiplexed onto the 7 address lines and latched into the internal address registers. Two negative-going external clocks are used to control the multiplexing. The Row Address Strobe ($\overline{RAS}$) enters the row address bits and the Column Address Strobe ($\overline{CAS}$) enters the column address bits.

When $\overline{RAS}$ is inactive, the memory enters its low power standby mode. Once the row address has been latched, it need not be changed for successive operations within the same row, allowing high-speed page-mode operations.

Page-mode operations first establish the row address and then maintain $\overline{RAS}$ low while $\overline{CAS}$ is repetitively cycled and designated operations are performed. Any column address within the selected row may be accessed in any sequence. The maximum time that $\overline{RAS}$ can remain low is the factor limiting the number of page-mode operations that can be performed.

Multiplexed addressing does not introduce extra delays in the access path. By inserting the row address first and the column address second, the memory takes advantage of the fact that the delay path through the memory is shorter for column addresses. The column address does not propagate through the cell matrix as the row address does and it can therefore arrive somewhat later than the row address without impacting the access time.

REFRESH

The Am9016 is a dynamic memory and each cell must be refreshed at least once every refresh interval in order to maintain the cell contents. Any operation that accesses a row serves to refresh all 128 cells in the row. Thus the refresh requirement is met by accessing all 128 rows at least once every refresh interval. This may be accomplished, in some applications, in the course of performing normal operations. Alternatively, special refresh operations may be initiated. These special operations could be simply additional conventional accesses or they could be "$\overline{RAS}$-only" cycles. Since only the rows need to be addressed, $\overline{CAS}$ may be held high while $\overline{RAS}$ is cycled and the appropriate row addresses are input. Power required for refreshing is minimized and simplified control circuitry will often be possible.

DATA INPUT/OUTPUT

Data is written into a selected cell by the combination of $\overline{WE}$ and $\overline{CAS}$ while $\overline{RAS}$ is low. The later negative transition of $\overline{WE}$ or $\overline{CAS}$ strobes the data into the internal register. In a write cycle, if the $\overline{WE}$ input is brought low prior to $\overline{CAS}$, the data is strobed by $\overline{CAS}$, and the set-up and hold times are referenced to $\overline{CAS}$. If the cycle is a read/write cycle then the data set-up and hold times are referenced to the negative edge of $\overline{WE}$.

In the read cycle the data is read by maintaining $\overline{WE}$ in the high state throughout the portion of the memory cycle in which $\overline{CAS}$ is low. The selected valid data will appear at the output within the specified access time.

DATA OUTPUT CONTROL

Any time $\overline{CAS}$ is high the data output will be off. The output contains either one or zero during read cycle after the access time has elapsed. Data remains valid from the access time until $\overline{CAS}$ is returned to the high state. The output data is the same polarity as the input data.

The user can control the output state during write operations by controlling the placement of the $\overline{WE}$ signal. In the "early write" cycle (see note 9) the output is at a high impedance state throughout the entire cycle.

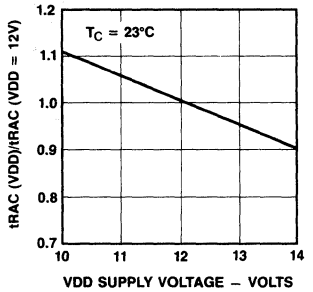
POWER CONSIDERATIONS

$\overline{RAS}$ and/or $\overline{CAS}$ can be decoded and used as a chip select signal for the Am9016 but overall system power is minimized if $\overline{RAS}$ is used for this purpose. The devices which do not receive $\overline{RAS}$ will be in low power standby mode regardless of the state of $\overline{CAS}$.

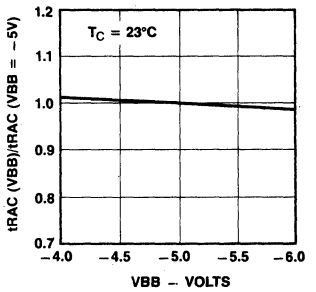
At all times the Absolute Maximum Rating Conditions must be observed. During power supply sequencing VBB should never be more positive than VSS when power is applied to VDD.

TYPICAL CHARACTERISTICS

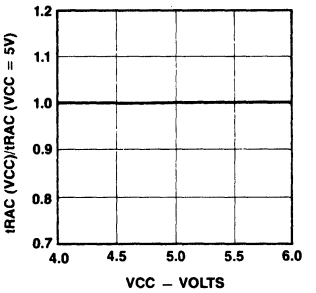
Typical Access Time (Normalized) tRAC Versus VDD



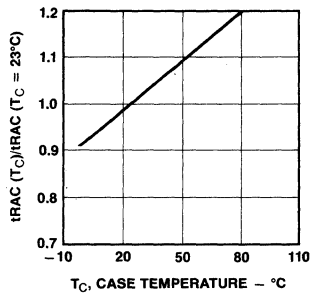
Typical Access Time (Normalized) tRAC Versus VBB



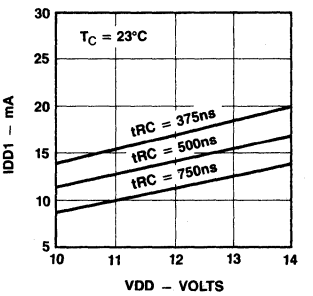
Typical Access Time (Normalized) tRAC Versus VCC



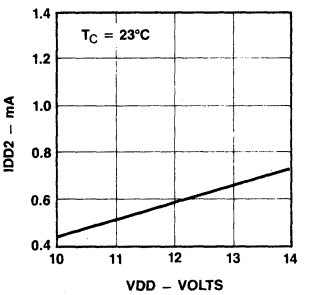
Typical Access Time (Normalized) tRAC Versus Case Temperature



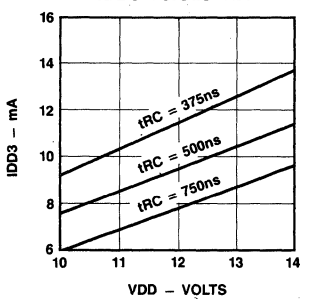
Typical Operating Current IDD1 Versus VDD



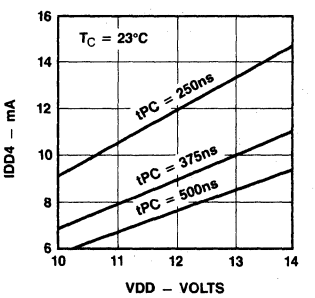
Typical Standby Current IDD2 Versus VDD



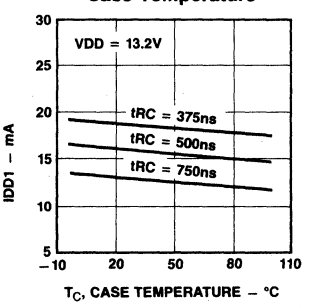
Typical Refresh Current IDD3 Versus VDD



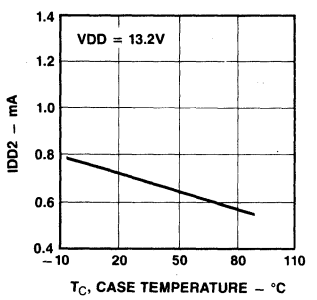
Typical Page Mode Current IDD4 Versus VDD



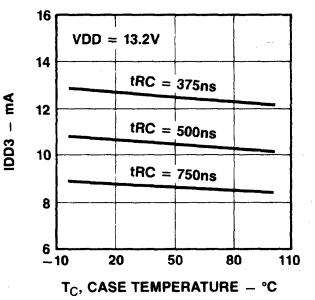
Typical Operating Current IDD1 Versus Case Temperature



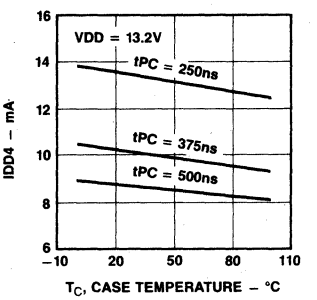
Typical Standby Current IDD2 Versus Case Temperature



Typical Refresh Current IDD3 Versus Case Temperature



Typical Page Mode Current IDD4 Versus Case Temperature

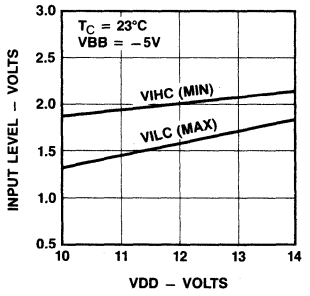


MOS-197

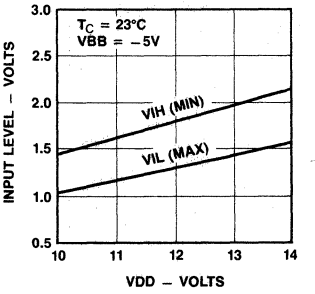
4

TYPICAL CHARACTERISTICS (Cont.)

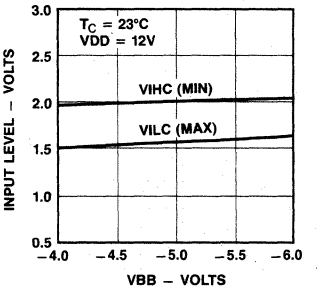
Input Voltage Levels Versus VDD



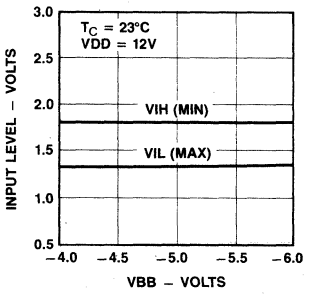
Input Voltage Levels Versus VDD



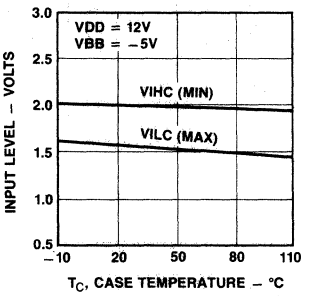
Input Voltage Levels Versus VBB



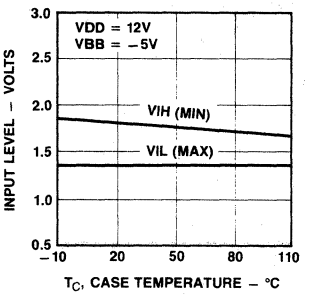
Input Voltages Levels Versus VBB



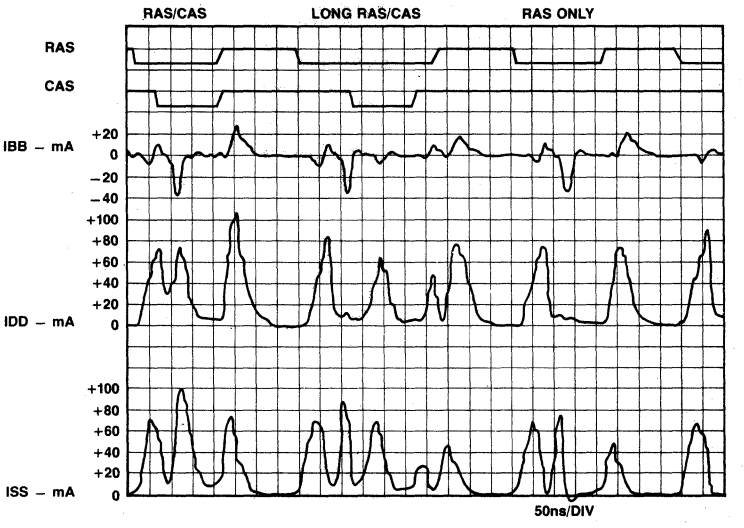
Input Voltage Levels Versus Case Temperature



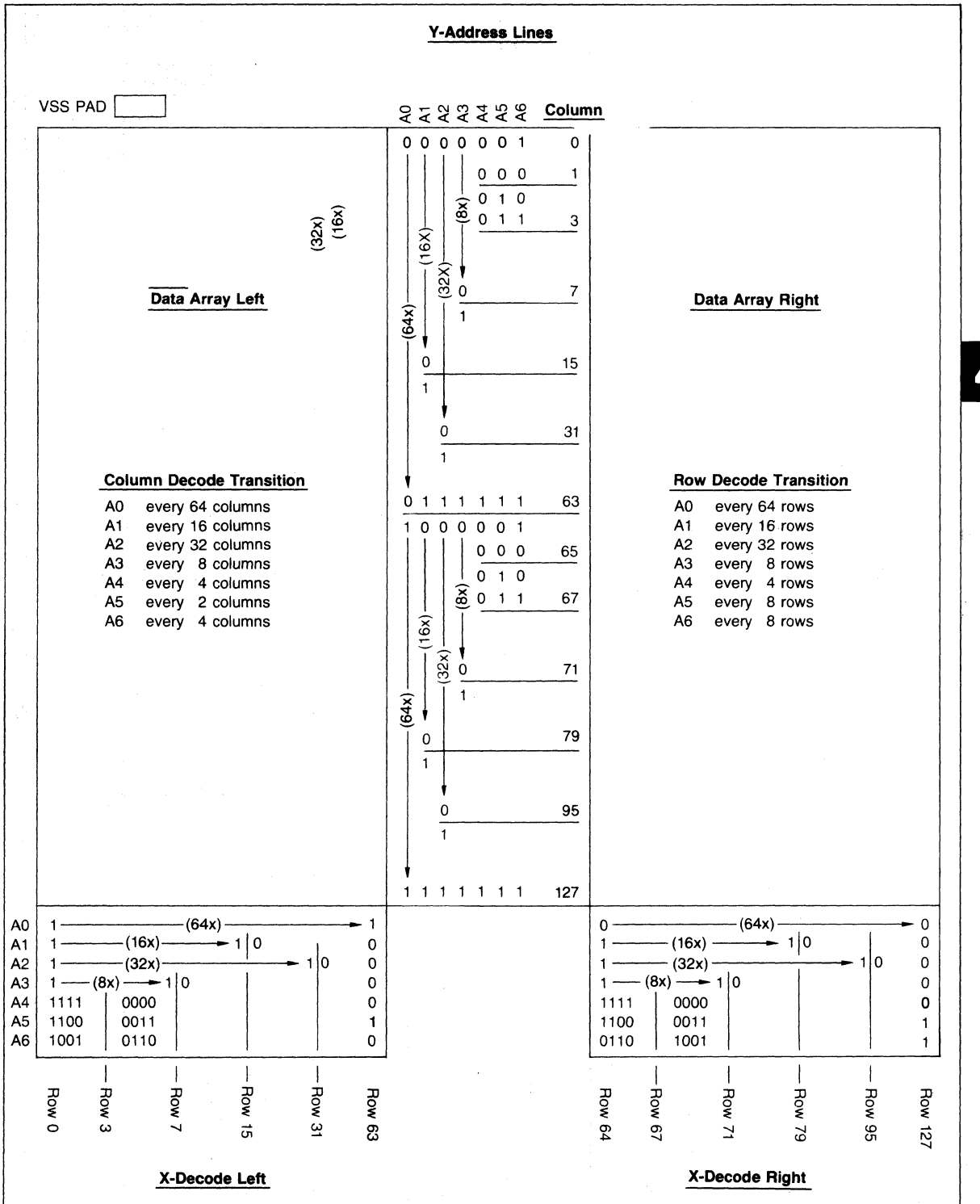
Input Voltage Levels Versus Case Temperature



TYPICAL CURRENT WAVEFORMS



MOS-198



TOPOLOGICAL BIT MAP

Am9044 • Am9244

4096 x 1 Static R/W Random Access Memory

DISTINCTIVE CHARACTERISTICS

- **LOW OPERATING POWER (MAX)**
  Am9044/Am9244 385mW (70mA)
  Am90L44/Am92L44 275mW (50mA)
- **LOW STANDBY POWER (MAX)**
  Am92L44 110mW (20mA)
- Access times down to 200ns (max)
- Military temperature range available to 250ns (max)
- Am9044 is a direct plug-in replacement for 4044
- Am9244 pin and function compatible with Am9044 and 4044 plus $\overline{CS}$ power down feature
- Fully static – no clocking
- Identical access and cycle time
- High output drive – 4.0mA sink current @ 0.4V
- TTL identical interface logic levels
- 100% MIL-STD-883 reliability assurance testing

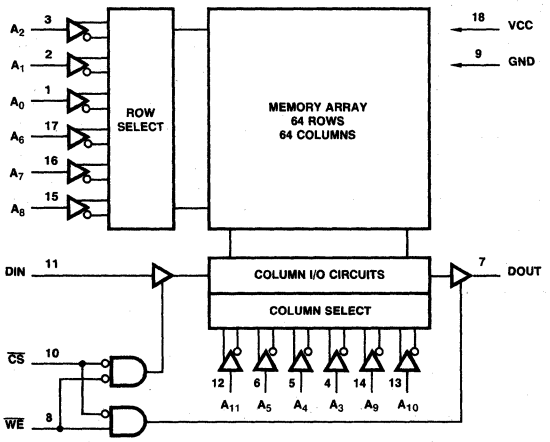
GENERAL DESCRIPTION

The Am9044 and Am9244 are high performance, static, N-Channel, read/write, random access memories organized as 4096 x 1. Operation is from a single 5V supply, and all input/output levels are identical to standard TTL specifications. Low power versions of both devices are available with power savings of about 30%. The Am9044 and Am9244 are the same except that the Am9244 offers an automatic $\overline{CS}$ power down feature.

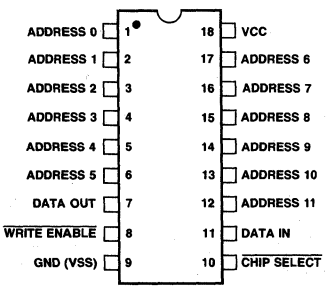
The Am9244 remains in a low power standby mode as long as $\overline{CS}$ remains high, thus reducing its power requirements. The Am9244 power decreases from 385mW to 165mW in the standby mode, and the Am92L44 from 275mW to 110mW. The $\overline{CS}$ input does not affect the power dissipation of the Am9044.

Data readout is not destructive and the same polarity as data input. $\overline{CS}$ provides for easy selection of an individual package when the outputs are OR-tied. The outputs of 4.0mA for Am9244 and Am9044 provide increased short circuit current for improved compacitive drive.

BLOCK DIAGRAM



CONNECTION DIAGRAM



Top View
Pin 1 is marked for orientation.

ORDERING INFORMATION

Ambient Temperature	Package Type	ICC Current Level	Access Times							
			Am9044				Am9244			
			450ns	300ns	250ns	200ns	450ns	300ns	250ns	200ns
0°C ≤ T_A ≤ +70°C	Plastic	70mA	AM9044BPC	AM9044CPC	AM9044DPC	AM9044EPC	AM9244BPC	AM9244CPC	AM9244DPC	AM9244EPC
		50mA	AM90L44BPC	AM90L44CPC	AM90L44DPC		AM92L44BPC	AM92L44CPC	AM92L44DPC	
	Hermetic	70mA	AM9044BDC	AM9044CDC	AM9044DDC	AM9044EDC	AM9244BDC	AM9244CDC	AM9244DDC	AM9244EDC
		50mA	AM90L44BDC	AM90L44CDC	AM90L44DDC		AM92L44BDC	AM92L44CDC	AM92L44DDC	
−55°C ≤ T_A ≤ +125°C	Hermetic	90mA	AM9044BDM	AM9044CDM	AM9044DDM		AM9244BDM	AM9244CDM	AM9244DDM	
		60mA	AM90L44BDM	AM90L44CDM			AM92L44BDM	AM92L44CDM		

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.0W
DC Output Current	10mA

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VSS	VCC	Part Number	Ambient Temperature	VSS	VCC
Am9044DC/PC Am90L44DC/PC Am9244DC/PC Am92L44DC/PC	$0°C \leq T_A \leq +70°C$	0V	+5.0V ±10%	Am9044DM Am90L44DM Am9244DM Am92L44DM	$-55°C \leq T_A \leq +125°C$	0V	+5.0V ±10%

ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions			Am9244XX Am92L44XX Min.	Typ.	Max.	Am9044XX Am90L44XX Min.	Typ.	Max.	Units
IOH	Output High Current	VOH = 2.4V	VCC = 4.5V	70°C	−1.0			−1.0			mA
		VOH = 2.4V	VCC = 4.5V	125°C	−.4			−.4			
IOL	Output Low Current	VOL = 0.4V	T_A = +70°C		4.0			4.0			mA
			T_A = +125°C		3.2			3.2			
VIH	Input High Voltage				2.0		VCC	2.0		VCC	Volts
VIL	Input Low Voltage				−0.5		0.8	−0.5		0.8	Volts
IIX	Input Load Current	VSS ≤ VI ≤ VCC					10			10	μA
IOZ	Output Leakage Current	0.4V ≤ VO ≤ VCC Output Disabled	T_A = +125°C		−50		50	−50		50	μA
			T_A = +70°C		−10		10	−10		10	
CI	Input Capacitance (Note 1)	Test Frequency = 1.0MHz T_A = 25°C, All pins at 0V				3.0	5.0		3.0	5.0	pF
CI/O	I/O Capacitance (Note 1)					5.0	6.0		5.0	6.0	

ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions		Am92L44 Typ.	Am92L44 Max.	Am9244 Typ.	Am9244 Max.	Am90L44 Typ.	Am90L44 Max.	Am9044 Typ.	Am9044 Max.	Units
ICC	VCC Operating Supply Current	Max. VCC $\overline{CS}$ ≤ VIL for Am9244/92L44	T_A = 0°C		50		70		50		70	mA
			T_A = −55°C		60		80		60		80	
IPD	Automatic $\overline{CS}$ Power Down Current	Max. V_{CC} ($\overline{CS} \geq V_{IH}$)	T_A = 0°C		20		30		–		–	mA
			T_A = −55°C		22		33		–		–	

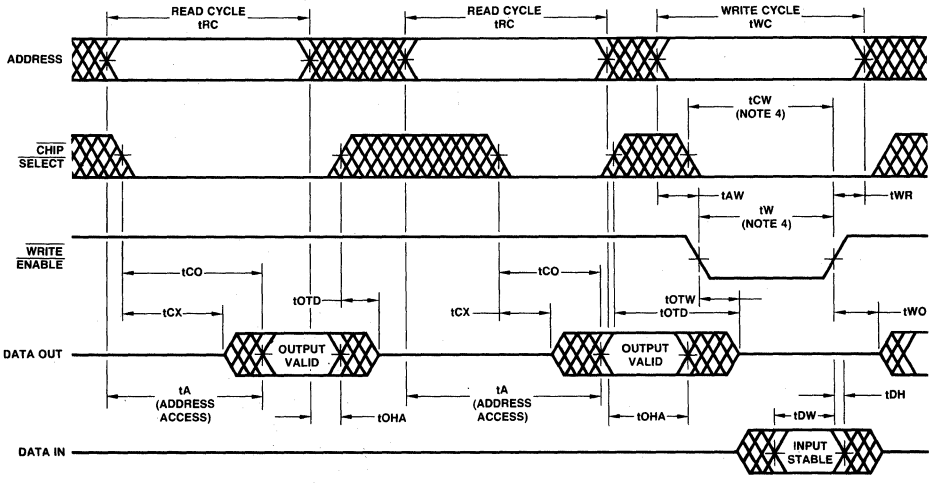
Notes:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. For test purposes, not more than one output at a time should be shorted. Short circuit test duration should not exceed 30 seconds.
3. Test conditions assume signal transition times of 10ns or less, timing reference levels of 1.5V and output loading of one standard TTL gate plus 100pF.
4. The internal write time of the memory is defined by the overlap of $\overline{CS}$ low and $\overline{WE}$ low. Both signals must be low to initiate a write and either signal can terminate a write by going high. The data input setup and hold timing should be referenced to the rising edge of the signal that terminates the write.
5. Chip Select access time (t_{CO}) is longer for the Am9244 than for the Am9044. The specified address access time will be valid only when Chip Select is low soon enough for t_{CO} to elapse.

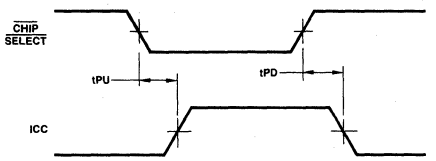
SWITCHING CHARACTERISTICS over operating range (Note 3)

Parameter	Description		Am9044B Am9244B Min.	Am9044B Am9244B Max.	Am9044C Am9244C Min.	Am9044C Am9244C Max.	Am9044D Am9244D Min.	Am9044D Am9244D Max.	Am9044E Am9244E Min.	Am9044E Am9244E Max.	Units
Read Cycle											
tRC	Address Valid to Address Do Not Care Time (Read Cycle Time)		450		300		250		200		ns
tA	Address Valid to Data Out Valid Delay (Address Access Time)			450		300		250		200	ns
tCO	Chip Select Low to Data Out Valid (Note 5)	Am9044		100		100		70		70	ns
		Am9244		450		300		250		200	ns
tCX	Chip Select Low to Data Out On		20		20		20		20		ns
tOTD	Chip Select High to Data Out Off			100		80		60		60	ns
tOHA	Address Unknown to Data Out Unknown Time		20		20		20		20		ns
Write Cycle											
tWC	Address Valid to Address Do Not Care Time (Write Cycle Time)		450		300		250		200		ns
tW	Write Enable Low to Write Enable High Time (Note 4)	Am9044	200		150		100		100		ns
		Am9244	250		200		150		150		ns
tWR	Write Enable High to Address Do Not Care Time		0		0		0		0		ns
tOTW	Write Enable Low to Data Out Off Delay			100		80		60		60	ns
tDW	Data In Valid to Write Enable High Time		200		150		100		100		ns
tDH	Write Enable Low to Data In Do Not Care Time		0		0		0		0		ns
tAW	Address Valid to Write Enable Low Time		0		0		0		0		ns
tPD	Chip Select High to Power Low Delay (Am9244 only)			200		150		100		100	ns
tPU	Chip Select Low to Power High Delay (Am9244 only)		0		0		0		0		ns
tCW	Chip Select Low to Write Enable High Time (Note 4)	Am9044	200		150		100		100		ns
		Am9244	250		200		150		150		ns
tWO	Write Enable High To Output Turn On			100		100		70		70	ns

SWITCHING WAVEFORMS

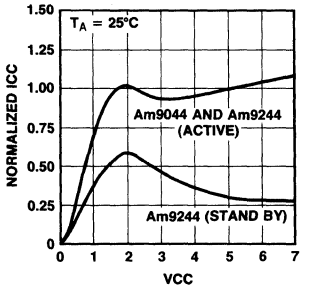


POWER DOWN WAVEFORM (Am9244 ONLY)

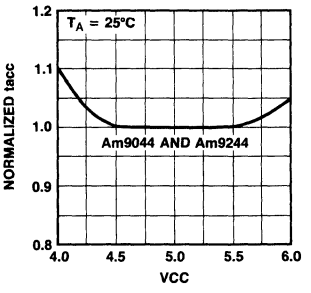


TYPICAL CHARACTERISTICS

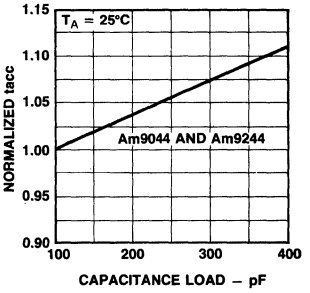
Typical ICC Versus VCC Characteristics



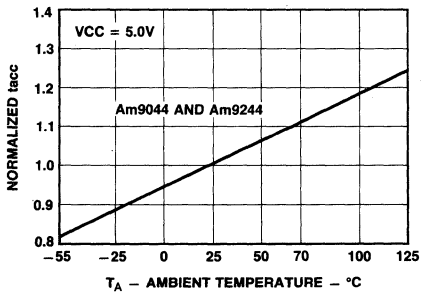
Typical tacc Versus VCC Characteristics



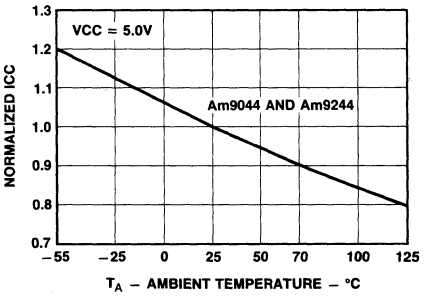
Typical C Load Versus Normalized tacc Characteristics



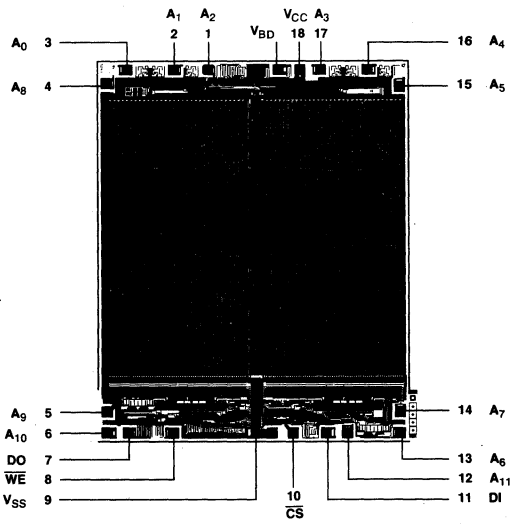
Normalized tacc Versus Ambient Temperature



Normalized ICC Versus Ambient Temperature



BIT MAP



Address Designators	
External	Internal
A0	A2
A1	A1
A2	A0
A3	A8
A4	A9
A5	A10
A6	A3
A7	A4
A8	A5
A9	A7
A10	A6
A11	A11

Figure 1. Bit Mapping Information.

Am9114 • Am9124

1024 x 4 Static R/W Random Access Memory

DISTINCTIVE CHARACTERISTICS

- **LOW OPERATING POWER (MAX)**
  Am9124/Am9114 368mW (70mA)
  Am91L24/Am91L14 262mW (50mA)
- **LOW STANDBY POWER (MAX)**
  Am9124 158mW (30mA)
  Am91L24 105mW (20mA)
- Access times down to 200ns (max)
- Military temperature range available to 300ns (max)
- Am9114 is a direct plug-in replacement for 2114
- Am9124 pin and function compatible with Am9114 and 2114, plus $\overline{CS}$ power down feature
- Fully static – no clocking
- Identical access and cycle time
- High output drive –
  4.0mA sink current @ 0.4V – 9124
  3.2mA sink current @ 0.4V – 9114
- TTL identical input/output levels
- 100% MIL-STD-883 reliability assurance testing

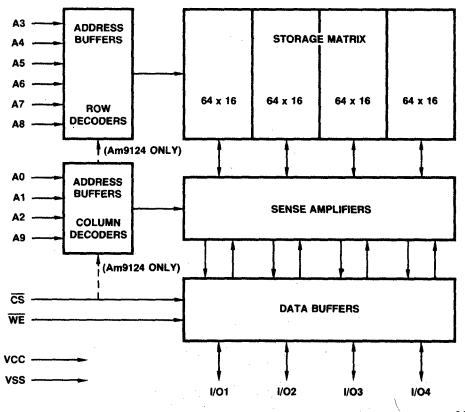
GENERAL DESCRIPTION

The Am9114 and Am9124 are high performance, static, N-Channel, read/write, random access memories organized as 1024 x 4. Operation is from a single 5V supply, and all input/output levels are identical to standard TTL specifications. Low power versions of both devices are available with power savings of over 30%. The Am9114 and Am9124 are the same except that the Am9124 offers an automatic $\overline{CS}$ power down feature.

The Am9124 remains in a low power standby mode as long as $\overline{CS}$ remains high, thus reducing its power requirements. The Am9124 power decreases from 368mW to 158mW in the standby mode, and the Am91L24 from 262mW to 105mW. The $\overline{CS}$ input does not affect the power dissipation of the Am9114. (See Figure 1, page 4).

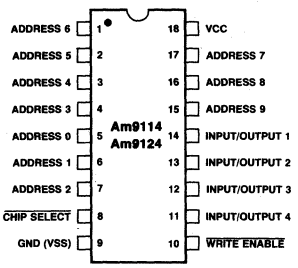
Data readout is not destructive and the same polarity as data input. $\overline{CS}$ provides for easy selection of an individual package when the outputs are OR-tied. The outputs of 4.0mA for Am9124 and 3.2mA for Am9114 provides increased short circuit current for improved capacitive drive.

BLOCK DIAGRAM



MOS-066

CONNECTION DIAGRAM



Top View
Pin 1 is marked for orientation.

MOS-067

ORDERING INFORMATION

Ambient Temperature	Package Type	ICC Current Level	Access Times					
			Am9114			Am9124 (Power Down Option)		
			450ns	300ns	200ns	450ns	300ns	200ns
$0°C \leq T_A \leq 70°C$	Plastic	70mA	Am9114BPC	Am9114CPC	Am9114EPC	Am9124BPC	Am9124CPC	Am9124EPC
		50mA	Am91L14BPC	Am91L14CPC		Am91L24BPC	Am91L24CPC	
	Hermetic	70mA	Am9114BDC	Am9114CDC	Am9114EDC	Am9124BDC	Am9124CDC	Am9124EDC
		50mA	Am91L14BDC	Am91L14CDC		Am91L24BDC	Am91L24CDC	
$-55°C \leq T_A \leq +125°C$	Hermetic	80mA	Am9114BDM	Am9114CDM		Am9124BDM	Am9124CDM	
		60mA	Am91L14BDM	Am91L14CDM		Am91L24BDM	Am91L24CDM	

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
V_{CC} with Respect to V_{SS}	−0.5V to +7.0V
All Signal Voltages with Respect to V_{SS}	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.0W
DC Output Current	10mA

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	V_{SS}	V_{CC}	Part Number	Ambient Temperature	V_{SS}	V_{CC}
Am9114DC/PC Am91L14DC/PC Am9124DC/PC Am91L24DC/PC	$0°C \leq T_A \leq +70°C$	0V	+5.0V ± 5%	Am9114DM Am91L14DM Am9124DM Am91L24DM	$-55°C \leq T_A \leq +125°C$	0V	+5.0V ± 10%

4

ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions		Am9124XX Am91L24XX Min.	Typ.	Max.	Am9114XX Am91L14XX Min.	Typ.	Max.	Units
I_{OH}	Output High Current	V_{OH} = 2.4V	V_{CC} = 4.75V	−1.4			−1.0			mA
		V_{OH} = 2.2V	V_{CC} = 4.5V	−1.0			−1.0			
I_{OL}	Output Low Current	V_{OL} = 0.4V	T_A = +70°C	4.0			3.2			mA
			T_A = +125°C	3.2			2.4			
V_{IH}	Input High Voltage			2.0		V_{CC}	2.0		V_{CC}	Volts
V_{IL}	Input Low Voltage			−0.5		0.8	−0.5		0.8	Volts
I_{IX}	Input Load Current	$V_{SS} \leq V_I \leq V_{CC}$				10			10	µA
I_{OZ}	Output Leakage Current	$0.4V \leq V_O \leq V_{CC}$ Output Disabled	T_A = +125°C	−50		50	−50		50	µA
			T_A = +70°C	−10		10	−10		10	
I_{OS}	Output Short Circuit Current	(Note 2)	0°C to +70°C			95			75	mA
			−55°C to +125°C			115			75	
CI	Input Capacitance (Note 1)	Test Frequency = 1.0MHz			3.0	5.0		3.0	5.0	pF
CI/O	I/O Capacitance (Note 1)	T_A = 25°C, All pins at 0V			5.0	6.0		5.0	6.0	

ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions		Am91L24 Typ.	Am91L24 Max.	Am9124 Typ.	Am9124 Max.	Am91L14 Typ.	Am91L14 Max.	Am9114 Typ.	Am9114 Max.	Units
I_{CC}	V_{CC} Operating Supply Current	Max. V_{CC}, $\overline{CS} \leq V_{IL}$ for Am9124/91L24	T_A = 25°C	40		60		40		60		mA
			T_A = 0°C		50		70		50		70	
			T_A = −55°C		60		80		60		80	
I_{PD}	Automatic $\overline{CS}$ Power Down Current	Max. V_{CC} ($\overline{CS} \geq V_{IH}$)	T_A = 25°C	15		24			–		–	mA
			T_A = 0°C		20		30		–		–	
			T_A = −55°C		22		33		–		–	

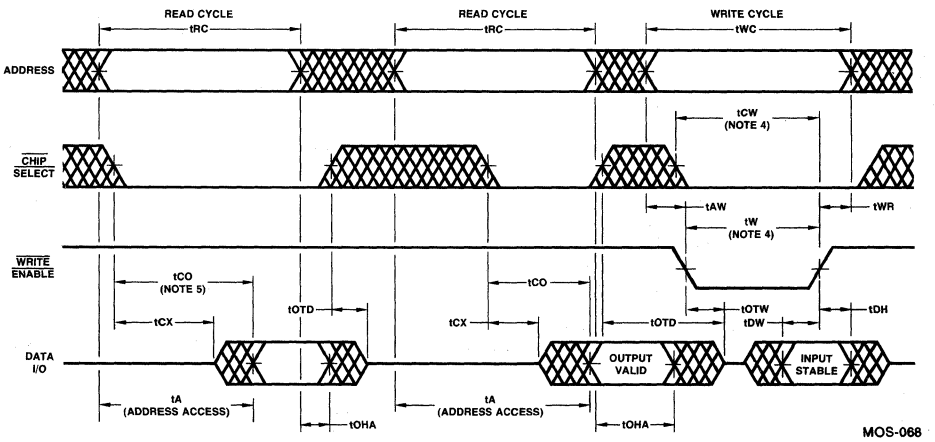
Notes:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. For test purposes, not more than one output at a time should be shorted. Short circuit test duration should not exceed 30 seconds.
3. Test conditions assume signal transition times of 10ns or less, timing reference levels of 1.5V and output loading of one standard TTL gate plus 100pF.
4. The internal write time of the memory is defined by the overlap of $\overline{CS}$ low and $\overline{WE}$ low. Both signals must be low to initiate a write and either signal can terminate a write by going high. The data input setup and hold timing should be referenced to the rising edge of the signal that terminates the write.
5. Chip Select access time (t_{CO}) is longer for the Am9124 than for the Am9114. The specified address access time will be valid only when Chip Select is low soon enough for t_{CO} to elapse.

SWITCHING CHARACTERISTICS over operating range (Note 3)

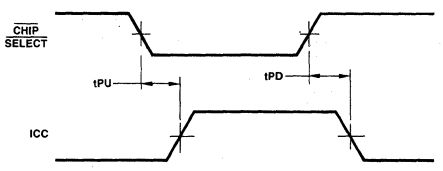
Parameter	Description		Am9114B Am9124B Min.	Am9114B Am9124B Max.	Am9114C Am9124C Min.	Am9114C Am9124C Max.	Am9114E Am9124E Min.	Am9114E Am9124E Max.	Unit
Read Cycle									
tRC	Address Valid to Address Do Not Care Time (Read Cycle Time)		450		300		200		ns
tA	Address Valid to Data Out Valid Delay (Address Access Time)			450		300		200	
tCO	Chip Select Low to Data Out Valid (Note 5)	Am9114		120		100		70	
		Am9124		420		280		185	
tCX	Chip Select Low to Data Out On		20		20		20		
tOTD	Chip Select High to Data Out Off			100		80		60	
tOHA	Address Unknown to Data Out Unknown Time		50		50		50		
Write Cycle									
tWC	Address Valid to Address Do Not Care Time (Write Cycle Time)		450		300		200		ns
tW	Write Enable Low to Write Enable High Time (Note 4)	Am9114	200		150		120		
		Am9124	250		200		150		
tWR	Write Enable High to Address Do Not Care Time		0		0		0		
tOTW	Write Enable Low to Data Out Off Delay			100		80		60	
tDW	Data In Valid to Write Enable High Time		200		150		120		
tDH	Write Enable Low to Data In Do Not Care Time		0		0		0		
tAW	Address Valid to Write Enable Low Time		0		0		0		
tPD	Chip Select High to Power Low Delay (Am9124 only)			200		150		100	
tPU	Chip Select Low to Power High Delay (Am9124 only)		0		0		0		
tCW	Chip Select Low to Write Enable High Time (Note 4)	Am9114	200		150		120		
		Am9124	250		200		150		

SWITCHING WAVEFORMS



MOS-068

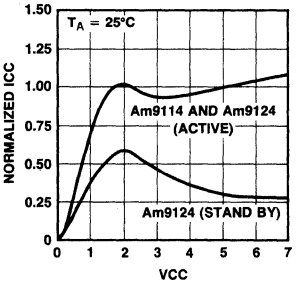
POWER DOWN WAVEFORM (Am9124 ONLY)



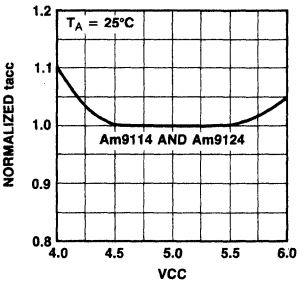
MOS-069

TYPICAL CHARACTERISTICS

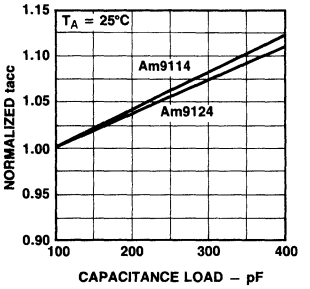
Typical ICC Versus VCC Characteristics



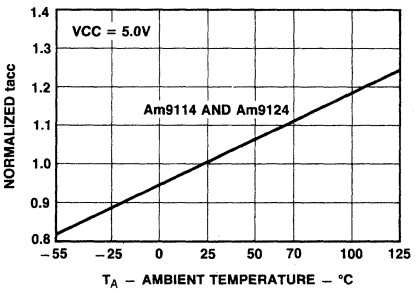
Typical tacc Versus VCC Characteristics



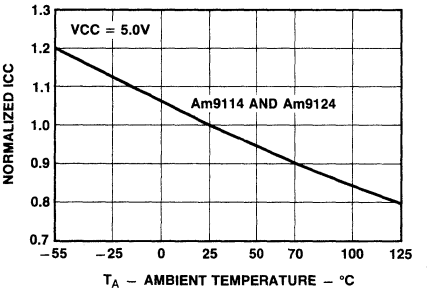
Typical C Load Versus Normalized tacc Characteristics



Normalized tacc Versus Ambient Temperature



Normalized ICC Versus Ambient Temperature



Configuration	Part Number	Worst Case Current (mA at 0°C) 100% Duty Cycle	50% Duty Cycle
2K x 8	9114	280	280
	91L14	200	200
	9124	200	160
	91L24	140	110
4K x 12	9114	840	840
	91L14	600	600
	9124	480	420
	91L24	330	285
8K x 16	9114	2240	2240
	91L14	1600	1600
	9124	1120	1040
	91L24	760	700

Figure 1. Supply Current Advantages of Am9124.

BIT MAP

Address Designators	
External	Internal
A0	A9
A1	A8
A2	A7
A3	A0
A4	A1
A5	A2
A6	A3
A7	A4
A8	A5
A9	A6

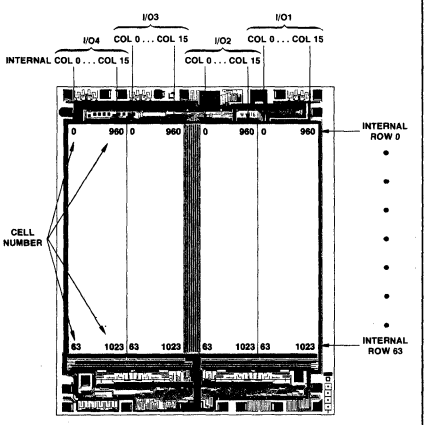


Figure 2. Bit Mapping Information.

Am9218/8316E
2048 x 8 Read Only Memory

DISTINCTIVE CHARACTERISTICS

- 2048 x 8 organization
- Plug-in replacement for 8316E
- Access times as fast as 350 ns
- Fully capacitive inputs — simplified driving
- 3 fully programmable Chip Selects — increased flexibility
- Logic voltage levels compatible with TTL
- Three-state output buffers — simplified expansion
- Drives two full TTL loads
- Single supply voltage — +5.0V
- Low power dissipation
- N-channel silicon gate MOS technology
- 100% MIL-STD-883 reliability assurance testing

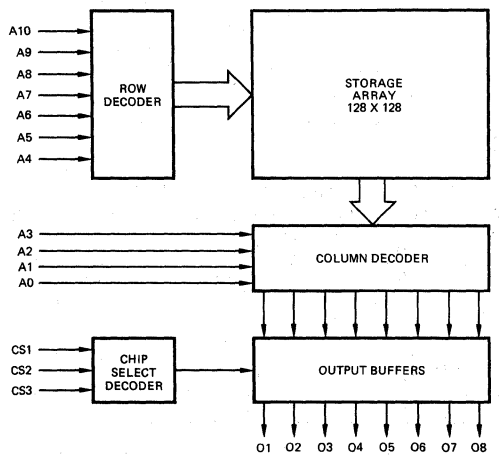
FUNCTIONAL DESCRIPTION

The Am9218 devices are high performance, 16384-bit, static, mask programmed, read only memories. Each memory is implemented as 2048 words by 8 bits per word. This organization simplifies the design of small memory systems and permits incremental memory sizes as small as 2048 words. The fast access times provided allow the ROM to service high performance microcomputer applications without stalling the processor.

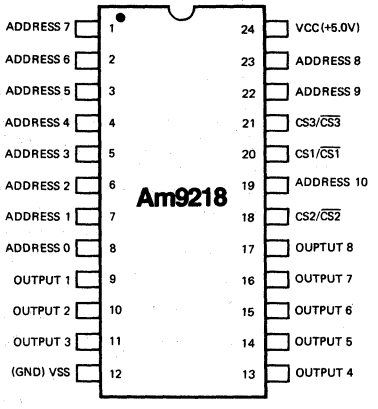
Three programmable Chip Select input signals are provided to control the output buffers. Each Chip Select polarity may be specified by the customer thus allowing the addressing of 8 memory chips without external gating. The outputs of unselected chips are turned off and assume a high impedance state. This permits wire-ORing with additional Am9218 devices and other three-state components.

These memories are fully static and require no clock signals of any kind. A selected chip will output data from a location specified by whatever address is present on the address input lines. Input and output voltage levels are compatible with TTL specifications.

BLOCK DIAGRAM



CONNECTION DIAGRAM



Top View
Pin 1 is marked for orientation.

ORDERING INFORMATION

Package Type	Ambient Temperature Specifications	Access Time	
		450ns	350ns
Molded	$0°C \leq T_A \leq 70°C$	AM9218BPC P8316E	AM9218CPC
Cerdip	$0°C \leq T_A \leq 70°C$	AM9218BCC	AM9218CCC
Side-Brazed Ceramic	$0°C \leq T_A \leq 70°C$	AM9218BDC C8316E	AM9218CDC
	$-55°C \leq T_A \leq +125°C$	AM9218BDM	

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	+7.0V
DC Voltage Applied to Outputs	−0.5V to +7.0V
DC Input Voltage	−0.5V to +7.0V
Power Dissipation	1.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

ELECTRICAL CHARACTERISTICS

Am9218BDC, Am9218CDC, C8316A: $T_A = 0°C$ to $+70°C$, VCC = 5.0V ±5%

Parameters	Description	Test Conditions		Am9218XDC Min.	Am9218XDC Max.	C8316E Min.	C8316E Max.	Units
VOH	Output HIGH Voltage	9218	IOH = −200μA	2.4				Volts
		8316E	IOH = −100μA			2.4		
VOL	Output LOW Voltage	9218	IOL = 3.2mA		0.4			Volts
		8316E	IOL = 2.1mA				0.4	
VIH	Input HIGH Voltage			2.0	VCC + 1.0	2.0	VCC + 1.0	Volts
VIL	Input LOW Voltage			−0.5	0.8	−0.5	0.8	Volts
ILO	Output Leakage Current	Chip Disabled			10		10	μA
ILI	Input Leakage Current				10		10	μA
ICC	VCC Supply Current				70		95	mA

ELECTRICAL CHARACTERISTICS

Am9218BDM: $T_A = -55°C$ to $+125°C$, VCC = 5.0V ±10%

Parameters	Description	Test Conditions	Am9218B Min.	Am9218B Max.	Units
VOH	Output HIGH Voltage	IOH = −200μA	2.2		Volts
VOL	Output LOW Voltage	IOL = 3.2mA		0.45	Volts
VIH	Input HIGH Voltage		2.0	VCC + 1.0	Volts
VIL	Input LOW Voltage		−0.5	0.8	Volts
ILO	Output Leakage Current	Chip Disabled		10	μA
ILI	Input Leakage Current			10	μA
ICC	VCC Supply Current			80	mA

SWITCHING CHARACTERISTICS OVER OPERATING RANGE

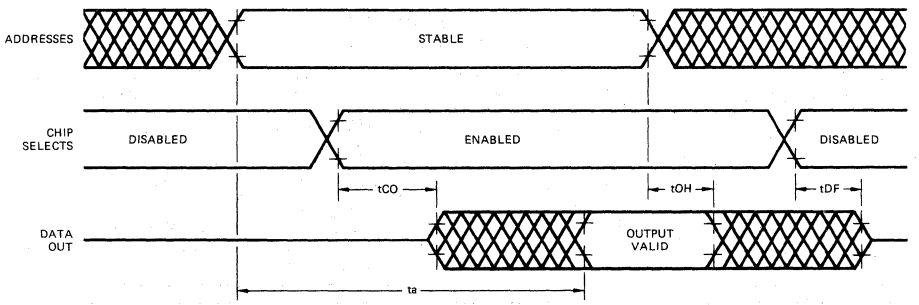
Am9218XDC/C8316E: $T_A = 0°C$ to $+70°C$, VCC = 5.0V ± 5%

Am9218BDM: $T_A = -55°C$ to $+125°C$, VCC = 5.0V ± 10%

Parameters	Description	Test Conditions	Am9218B Min.	Am9218B Max.	Am9218C Min.	Am9218C Max.	8316E Min.	8316E Max.	Units
ta	Address to Output Access Time	tr = tf = 20ns, Output Load: one standard TTL gate plus 100pF (Note 1)		450		350		450	ns
tCO	Chip Select to Output ON Delay			150		130		250	ns
tOH	Previous Read Data Valid with Respect to Address Change		20		20		–		ns
tDF	Chip Select to Output OFF Delay			150		130		250	ns
CI	Input Capacitance	$T_A = 25°C$, f = 1.0MHz, All pins at 0V		7.0		7.0		7.0	pF
CO	Output Capacitance			7.0		7.0		7.0	pF

Notes: 1. Timing reference levels: High = 2.0V, Low = 0.8V.

SWITCHING WAVEFORMS



Am9232 • Am9233

4096 X 8 Read Only Memory

PRELIMINARY DATA

DISTINCTIVE CHARACTERISTICS

- 4096 X 8 organization
- No clocks or refresh required
- Access time selected to 300ns
- Fully capacitive inputs – simplified driving
- 2 mask programmable chip selects – increased flexibility
- Logic voltage levels compatible with TTL
- Three state output buffers – simplified expansion
- Drives two TTL loads
- Single +5 volt power supply
- Two different pinouts for universal application
- Low power dissipation
- 100% MIL-STD-883 reliability assurance testing
- Non-connect option on chip selects.

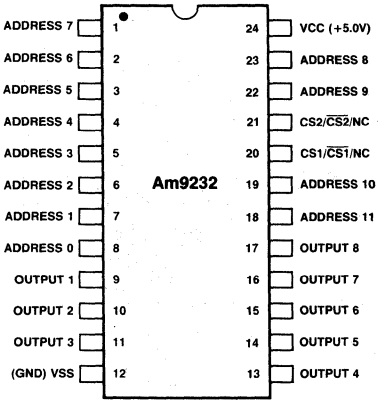
FUNCTIONAL DESCRIPTION

The Am9232/33 devices are high performance, 32,768-bit, static, mask programmed, read only memories. Each memory is implemented as 4096 words by 8 bits per word. This organization simplifies the design of small memory systems and permits incremental memory sizes of 4096 words. The fast access times provided allow the ROM to service high performance microcomputer applications without stalling the processor.

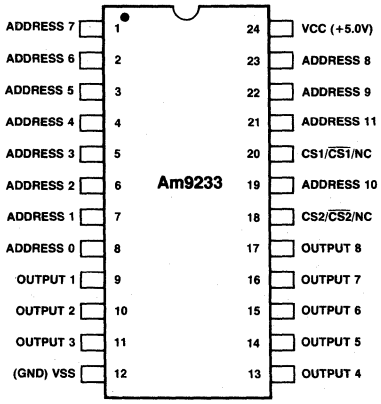
Two programmable Chip Select input signals are provided to control the output buffers. Each Chip Select polarity may be specified by the customer thus allowing the addressing of 4 memory chips without external gating. The outputs of unselected chips are turned off and assume a high impedance state. This permits wire-ORing with additional Am9232/33 devices and other three-state components.

These memories are fully static and require no clock signals of any kind. A selected chip will output data from a location specified by the address present on the address input lines. Input and output voltage levels are compatible with TTL specifications.

CONNECTION DIAGRAMS
Top Views



MOS-103



MOS-104

Note: Pin 1 is marked for orientation.

ORDERING INFORMATION

Package Type	Ambient Temperature Specifications	Access Time	
		450ns	300ns
Molded	$0°C \leq T_A \leq +70°C$	AM9232/33BPC	AM9232/33CPC
Cerdip	$0°C \leq T_A \leq +70°C$	AM9232/33BCC	AM9232/33CCC
Side-Brazed Ceramic	$-55°C \leq T_A \leq +125°C$	AM9232/33BDM	
	$0°C \leq T_A \leq +70°C$	AM9232/33BDC	AM9232/33CDC

MAXIMUM RATINGS beyond which the useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	+7.0V
DC Voltage Applied to Outputs	−0.5V to +7.0V
DC Input Voltage	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VCC	VSS
Am9232DC/PC/CC	0°C ≤ T_A ≤ +70°C	+5.0V ±5%	0V
Am9232/33DM	−55°C ≤ T_A ≤ +125°C	+5.0V ±10%	0V

4

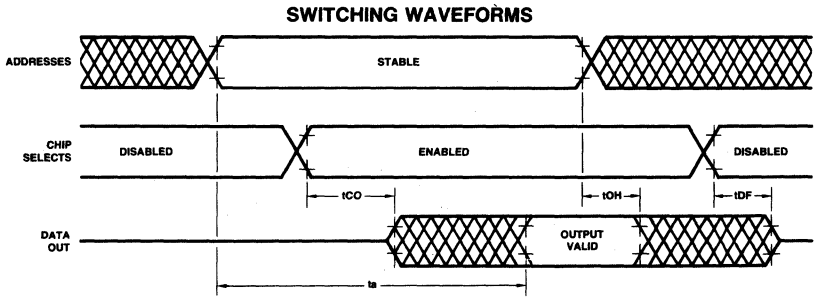
ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions		Am9232/Am9233 Min.	Max.	Unit
VOH	Output HIGH Voltage	IOH = −200μA	VCC = 4.75	2.4		Volts
			VCC = 4.50	2.2		
VOL	Output LOW Voltage	IOL = 3.2mA			0.4	Volts
VIH	Input HIGH Voltage			2.0	VCC+1.0	Volts
VIL	Input LOW Voltage			−0.5	0.8	Volts
I_{LI}	Input Load Current	VSS ≤ VI ≤ VCC			10	μA
IOZ	Output Leakage Current	VSS ≤ VO ≤ VCC Chip Disabled	+70°C		10	μA
			+125°C (DM)		50	
ICC	VCC Supply Current		0°C		80	mA
			−55°C (DM)		100	
CI	Input Capacitance	T_A = 25°C, f = 1.0MHz All pins at 0V			7.0	pF
CO	Output Capacitance				7.0	pF

SWITCHING CHARACTERISTICS over operating range

Parameter	Description	Test Conditions	Am9232/33B Min.	Am9232/33B Max.	Am9232/33C Min.	Am9232/33C Max.	Unit
ta	Address to Output Access Time	tr = tf = 20ns Output Load: one standard TTL gate plus 100pF (Note 1)		450		300	ns
tCO	Chip Select to Output ON Delay			150		120	ns
tOH	Previous Read Data Valid with Respect to Address Change		20		20		ns
tDF	Chip Select to Output OFF Delay			150		120	ns

SWITCHING WAVEFORMS



MOS-105

Am93412 • Am93422

TTL 1024-Bit Random Access Memories

DISTINCTIVE CHARACTERISTICS

- Fully decoded 256-word x 4-bit RAMs
- High speed: Address access time typically 30ns
- Internal ECL circuitry for optimum speed/power performance over voltage and temperature
- Output preconditioned during write to eliminate write recovery glitch
- Available with 3-state outputs (Am93422) or with open collector outputs (Am93412)
- 100% reliability assurance testing in compliance with MIL-STD-883
- Electrically tested and optically inspected die for the assemblers of hybrid products
- Plug-in replacement for Fairchild 93412 and 93422
- Power dissipation decreases with increasing temperature

FUNCTIONAL DESCRIPTION

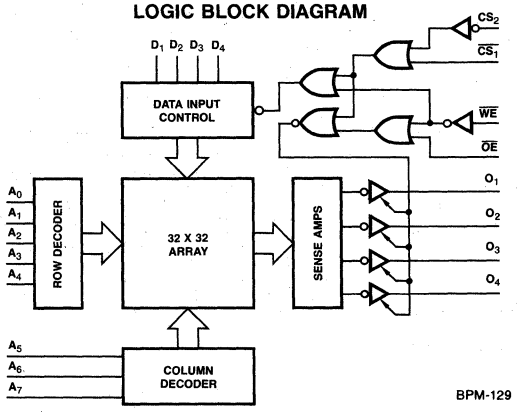
The Am93412 and Am93422 are 1024-bit RAMs built using Schottky diode clamped transistors in conjunction with internal ECL circuitry and are ideal for use in high-speed control and buffer memory applications. Each memory is organized as a fully decoded 256-word memory of four bits per word. Easy memory expansion is provided by an active LOW chip select one ($\overline{CS_1}$) or an active HIGH chip select two (CS_2) and open collector OR tieable outputs (Am93412) or 3-state outputs (Am93422).

An active LOW write line ($\overline{WE}$) controls the writing/reading operation of the memory. When the chip select one ($\overline{CS_1}$) and write line ($\overline{WE}$) are LOW and chip select two (CS_2) is HIGH, the information on data inputs (D_X) is written into the addressed memory word and preconditions the output circuitry so that true data is present at the outputs when the write cycle is complete. This preconditioning operation insures minimum write recovery times by eliminating the "write recovery glitch".

Reading is performed with the chip select one ($\overline{CS_1}$) LOW and the chip select two (CS_2) HIGH and the write line ($\overline{WE}$) HIGH. The information stored in the addressed word is read out on the non-inverting outputs (O_X).

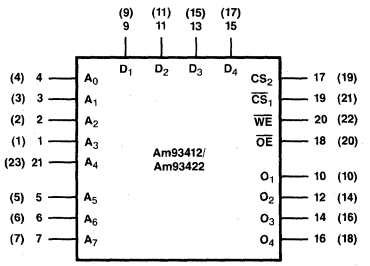
During the writing operation, chip select one ($\overline{CS_1}$) is HIGH, chip select two (CS_2) is LOW or output enable ($\overline{OE}$) is HIGH. The outputs of the memory go to an inactive high-impedance state.

LOGIC BLOCK DIAGRAM



BPM-129

LOGIC SYMBOL

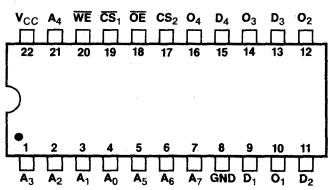


BPM-130

CONNECTION DIAGRAMS

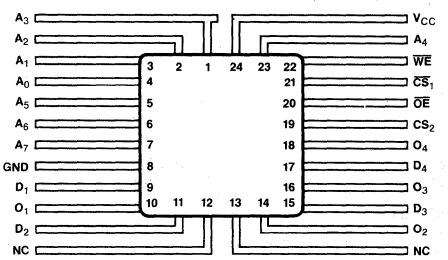
Top Views

DIP



BPM-131

Flat Pack



Note: Pin 1 is marked for orientation.

BPM-132

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Temperature (Ambient) Under Bias	−55 to +125°C
Supply Voltage to Ground Potential (Pin 22 to Pin 11)	−0.5 to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to V_{CC} max
DC Input Voltage	−0.5 to +5.5V
Output Current, Into Outputs (Low)	20mA
DC Input Current	−30 to +5.0mA

OPERATING RANGE

Part No.	V_{CC}	Ambient Temperature
Commercial Grade Am93412PC, DC Am93422PC, DC	5.0V ±5%	0 to +75°C
Military Grade Am93412DM, FM Am93422DM, FM	5.0V ±10%	−55 to +125°C

4

DC ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

Parameters	Description	Test Conditions		Min	Typ (Note 1)	Max	Units
V_{OH} (Am93422 only)	Output HIGH Voltage	V_{CC} = MIN, V_{IN} = V_{IH} or V_{IL}	I_{OH} = −5.2mA	2.4	3.6		Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN, V_{IN} = V_{IH} or V_{IL}	I_{OL} = 8.0mA		0.350	0.45	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs		2.1	1.6		Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs			1.5	0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX, V_{IN} = 0.40V			−180	−400	μA
I_{IH}	Input HIGH Current	V_{CC} = MAX, V_{IN} = 4.5V			1	40	μA
I_{SC} (Am93422 only)	Output Short Circuit Current	V_{CC} = MAX, V_{OUT} = 0.0V				−70	mA
I_{CC}	Power Supply Current	All inputs = GND, V_{CC} = MAX.	$T_A \geq 75°C$		100	130	mA
			T_A = 0°C			155	
			T_A = −55°C			170	
V_C	Input Clamp Voltage	V_{CC} = MIN, I_{IN} = −16mA			−0.850	−1.5	Volts
I_{CEX}	Output Leakage Current	V_{OUT} = 2.4V	Am93412/422		0	100	μA
		V_{OUT} = 0.5V, V_{CC} = MAX	Am93422	−50	0		
C_{IN}	Input Pin Capacitance	See Note 2			4		pF
C_{OUT}	Output Pin Capacitance	See Note 2			7		pF

Notes: 1. Typical characteristics are at V_{CC} = 5.0V and T_A = 25°C.
2. Input and output capacitance measured on a sample basis using pulse technique.

FUNCTION TABLE

Inputs					Output	
CS_2	$\overline{CS_1}$	$\overline{WE}$	$\overline{OE}$	D_X	O_X	Mode
L	X	X	X	X	*HIGH Z	Not Select
X	H	X	X	X	*HIGH Z	Not Select
H	L	H	H	X	*HIGH Z	Output Disable
H	L	H	L	X	*Data	Read Data
H	L	L	X	L	*HIGH Z	Write "0"
H	L	L	X	H	*HIGH Z	Write "1"

H = High Voltage Level L = Low Voltage Level X = Don't Care

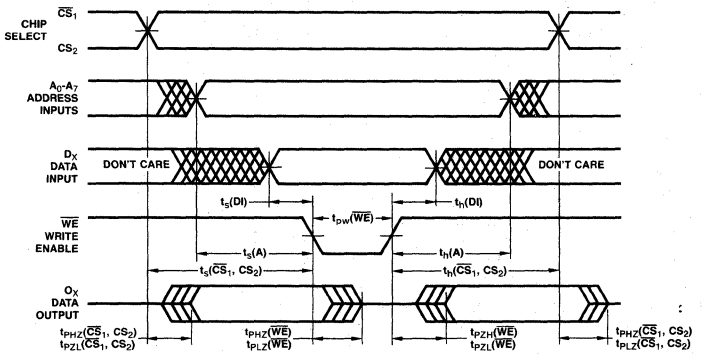
*HIGH Z implies outputs are disabled or off. This condition is defined as a high impedance state for the Am93422 and as an output high level for the Am93412.

SWITCHING CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

Parameters	Description		Test Conditions	Typ (Note 1)	Am93412/422PC Am93412/422DC Min	Am93412/422PC Am93412/422DC Max	Am93412/422DM Am934 ?/422FM Min	Am93412/422DM Am934 ?/422FM Max	Units
$t_{PLH}(A)$ $t_{PHL}(A)$	Delay from Address to Output (Address Access Time)	See Fig. 2	See Figures 3 and 4 and Notes 3, 4 and 5 (below)	30		45		60	ns
$t_{PZH}(\overline{CS_1}),(CS_2)$ $t_{PZL}(\overline{CS_1}),(CS_2)$	Delay from Chip Select to Active Output and Correct Data	See Fig. 2		15		30		45	ns
$t_{PZH}(\overline{WE})$ $t_{PZL}(\overline{WE})$	Delay from Write Enable to Active Output and Correct Data (Write Recovery)	See Fig. 1		15		30		45	ns
$t_{PZH}(\overline{OE})$ $t_{PZL}(\overline{OE})$	Delay from Output Enable to Active Output and Correct Data	See Fig. 2		15		30		45	ns
$t_s(A)$	Set-up Time Address (Prior to Initiation of Write)	See Fig. 1		−10	10		10		ns
$t_h(A)$	Hold Time Address (After Termination of Write)	See Fig. 1		−10	5		10		ns
$t_s(DI)$	Set-up Time Data Input (Prior to Initiation of Write)	See Fig. 1		−10	5		5		ns
$t_h(DI)$	Hold Time Data Input (After Termination of Write)	See Fig. 1		−10	5		5		ns
$t_s(\overline{CS_1}),(CS_2)$	Set-up Time Chip Select (Prior to Initiation of Write)	See Fig. 1		−10	5		5		ns
$t_h(\overline{CS_1}),(CS_2)$	Hold Time Chip Select (After Termination of Write)	See Fig. 1		−10	5		5		ns
$t_{pw}(\overline{WE})$	Min Write Enable Pulse Width to Insure Write	See Fig. 1		15	30		40		ns
$t_{PHZ}(\overline{CS_1}),(CS_2)$ $t_{PLZ}(\overline{CS_1}),(CS_2)$	Delay from Chip Select to Inactive Output (HIGH Z)	See Fig. 2		15		30		45	ns
$t_{PHZ}(\overline{WE})$ $t_{PLZ}(\overline{WE})$	Delay from Write Enable to Inactive Output (HIGH-Z)	See Fig. 1		15		30		45	ns
$t_{PHZ}(\overline{OE})$ $t_{PLZ}(\overline{OE})$	Delay from Output Enable to Inactive Output (HIGH-Z)	See Fig. 2		15		30		45	ns

Notes:
1. Typical characteristics are at V_{CC} = 5.0V and T_A = 25°C.
2. Input and output capacitance measured on a sample basis using pulse technique.
3. $t_{PLH}(A)$ and $t_{PHL}(A)$ are tested with S_1 closed and C_L = 30pF with both input and output timing referenced to 1.5V.
4. For open collector Am93412, all delays from Write Enable ($\overline{WE}$) or Chip Select ($\overline{CS_1}$, CS_2) inputs to the Data Output (D_{OUT}) ($t_{PLZ}(\overline{WE})$, $t_{PLZ}(\overline{CS_1}$, $CS_2)$, $t_{PLZ}(\overline{OE})$, $t_{PZL}(\overline{WE})$ and $t_{PZL}(\overline{CS_1}, CS_2)$), $t_{PZL}(\overline{OE})$ are measured with S_1 closed and C_L = 30pF; and with both the input and output timing referenced to 1.5V.
5. For 3-state output Am93422, $t_{PZH}(\overline{WE})$ and $t_{PZH}(\overline{CS_1}, CS_2)$, $t_{PZH}(\overline{OE})$ are measured with S_1 open, C_L = 30pF and with both the input and output timing referenced to 1.5V. $t_{PZL}(\overline{WE})$ and $t_{PZL}(\overline{CS_1}, CS_2)$, $t_{PZL}(\overline{OE})$ are measured with S_1 closed, C_L = 30pF and with both the input and output timing referenced to 1.5V. $t_{PHZ}(\overline{WE})$ and $t_{PHZ}(\overline{CS_1}, CS_2)$, $t_{PHZ}(\overline{OE})$ are measured with S_1 open and $C_L \leq$ 5pF and are measured between the 1.5V level on the input to the V_{OH} − 500mV level on the output. $t_{PLZ}(\overline{WE})$ and $t_{PLZ}(\overline{CS_1}, CS_2)$, $t_{PLZ}(\overline{OE})$ are measured with S_1 closed and $C_L \leq$ 5pF and are measured between the 1.5V level on the input and the V_{OL} + 500mV level on the output.

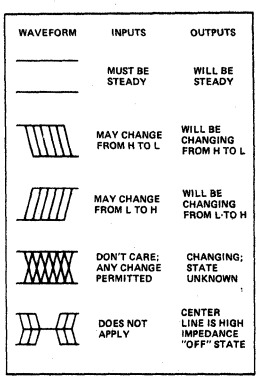
SWITCHING WAVEFORMS
WRITE MODE (WITH $\overline{OE}$ = LOW)



BPM-133

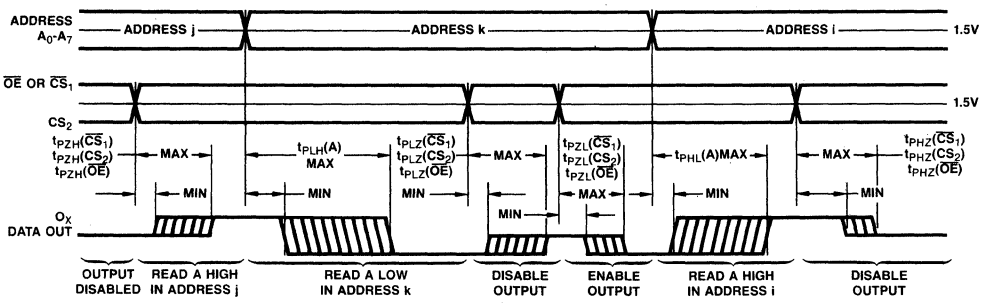
Figure 1.

KEY TO TIMING DIAGRAM



SWITCHING WAVEFORMS (Cont.)

READ MODE



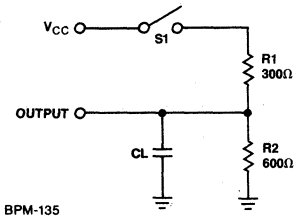
BPM-134

Switching delays from address, output enable input and chip select inputs to the data output. For the Am93422 disabled output is "OFF", represented by a single center line. For the Am93412, a disabled output is HIGH.

Figure 2.

AC TEST LOAD AND WAVEFORM

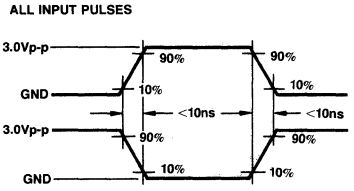
AC TEST LOAD



BPM-135

Figure 3.

INPUT PULSES



BPM-136

Figure 4.

See Notes 3, 4 and 5 of Switching Characteristics.

ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Open Collector Outputs		
Molded DIP	0 to 75°C	AM93412PC
Hermetic DIP	0 to 75°C	AM93412DC
Hermetic DIP	−55 to +125°C	AM93412DM
Hermetic Flat Pack	−55 to +125°C	AM93412FM
3-State Outputs		
Molded DIP	0 to 75°C	AM93422PC
Hermetic DIP	0 to 75°C	AM93422DC
Hermetic DIP	−55 to +125°C	AM93422DM
Hermetic Flat Pack	−55 to +125°C	AM93422FM

Am93415 • Am93425

TTL 1024-Bit Random Access Memories

DISTINCTIVE CHARACTERISTICS

- Fully decoded 1024-word x 1-bit RAMs
- High speed: Address access time typically 30ns
- Internal ECL circuitry for optimum speed/power performance over voltage and temperature
- Output preconditioned during write to eliminate write recovery glitch
- Available with 3-state outputs (Am93425) or with open collector outputs (Am93415)
- 100% reliability assurance testing in compliance with MIL-STD-883
- Electrically tested and optically inspected die for the assemblers of hybrid products
- Plug in replacement for Fairchild 93415 and 93425.
- Power dissipation decreases with increasing temperature

FUNCTIONAL DESCRIPTION

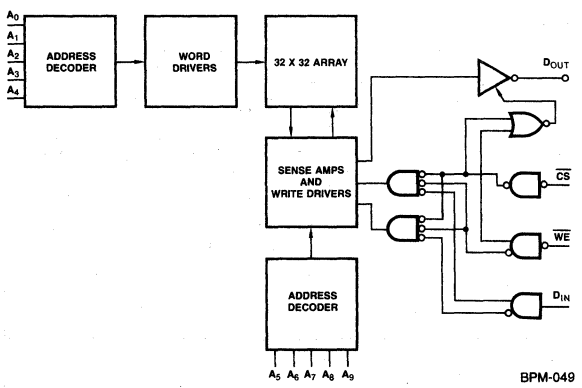
The Am93415 and Am93425 are 1024-bit RAMs built using Schottky diode clamped transistors in conjunction with internal ECL circuitry and are ideal for use in high-speed control and buffer memory applications. Each memory is organized as a fully decoded 1024-word memory of 1 bit per word. Easy memory expansion is provided by an active LOW chip select ($\overline{CS}$) input and open collector OR tieable outputs (Am93415) or 3-state outputs (Am93425). Chip selection for large memory systems can be controlled by active LOW output decoders such as the Am74S138.

An active LOW write line ($\overline{WE}$) controls the writing/reading operation of the memory. When the chip select and write lines are LOW the information on the data input (D_{IN}) is written into the addressed memory word and preconditions the output circuitry so that true data is present at the outputs when the write cycle is complete. This preconditioning operation insures minimum write recovery times by eliminating the "write recovery glitch".

Reading is performed with the chip select line LOW and the write line HIGH. The information stored in the addressed word is read out on the non-inverting output (D_{OUT}).

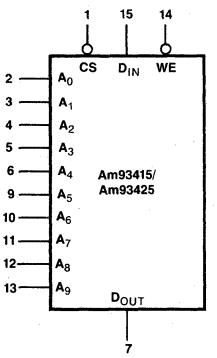
During the writing operation or when the chip select line is HIGH the output of the memory goes to an inactive high impedance state.

LOGIC BLOCK DIAGRAM



BPM-049

LOGIC SYMBOL



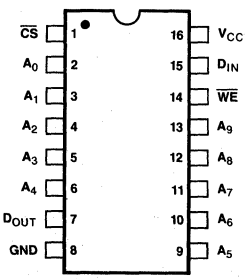
V_{CC} = Pin 16
GND = Pin 8

BPM-050

ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Open Collector Outputs		
Molded DIP	0 to 75°C	AM93415PC
Hermetic DIP	0 to 75°C	AM93415DC
Hermetic DIP	−55 to +125°C	AM93415DM
Hermetic Flat Pak	−55 to +125°C	AM93415FM
3-State Outputs		
Molded DIP	0 to 75°C	AM93425PC
Hermetic DIP	0 to 75°C	AM93425DC
Hermetic DIP	−55 to +125°C	AM93425DM
Hermetic Flat Pak	−55 to +125°C	AM93425FM

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BPM-051

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential (Pin 16 to Pin 8)	−0.5V to +7V
DC Voltage Applied to Outputs for High Output State	−0.5V to V_{CC} max.
DC Input Voltage	−0.5V to +5.5V
Output Current, Into Outputs (Low)	20mA
DC Input Current	−30mA to +5.0mA

OPERATING RANGE

Part No.	V_{CC}	Ambient Temperature
Commercial Grade Am93415PC, DC Am93425PC, DC	5.0V ±5%	0°C to +75°C
Military Grade Am93415DM, FM Am93425DM, FM	5.0V ±10%	−55°C to +125°C

4

DC ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

Parameters	Description	Test Conditions			Min	Typ (Note 1)	Max	Units
V_{OH} (Am93425 only)	Output HIGH Voltage	V_{CC} = MIN, V_{IN} = V_{IH} or V_{IL}	I_{OH} = −10.3mA	COM'L	2.4	3.6		Volts
			I_{OH} = −5.2	MIL				
V_{OL}	Output LOW Voltage	V_{CC} = MIN, V_{IN} = V_{IH} or V_{IL}	I_{OL} = 16mA			0.350	0.45	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs			2.1	1.6		Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs				1.5	0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX, V_{IN} = 0.40V				−180	−400	μA
I_{IH}	Input HIGH Current	V_{CC} = MAX, V_{IN} = 4.5V				1	40	μA
I_{SC} (Am93425 only)	Output Short Circuit Current	V_{CC} = MAX, V_{OUT} = 0.0V					−100	mA
I_{CC}	Power Supply Current	All inputs = GND, V_{CC} = MAX.		$T_A \geq 75°C$		95	130	mA
				T_A = 0°C			155	
				T_A = −55°C			170	
V_C	Input Clamp Voltage	V_{CC} = MIN, I_{IN} = −16mA				−0.850	−1.5	Volts
I_{CEX}	Output Leakage Current	$V_{\overline{CS}}$ = V_{IH} or $V_{\overline{WE}}$ = V_{IL} V_{OUT} = 2.4V		Am93415/425		0	100	μA
		$V_{\overline{CS}}$ = V_{IH} or $V_{\overline{WE}}$ = V_{IL} V_{OUT} = 0.5V, V_{CC} = MAX		Am93425	−50	0		
C_{IN}	Input Pin Capacitance	See Note 2				4		pF
C_{OUT}	Output Pin Capacitance	See Note 2				7		pF

Notes: 1. Typical characteristics are at V_{CC} = 5.0V and T_A = 25°C.
2. Input and output capacitance measured on a sample basis using pulse technique.

FUNCTION TABLE

Inputs			Output	
$\overline{CS}$	$\overline{WE}$	D_{IN}	D_{OUT}	Mode
H	X	X	*HIGH Z	Not Selected
L	L	L	*HIGH Z	Write "0"
L	L	H	*HIGH Z	Write "1"
L	H	X	Selected Data	Read

H = High Voltage Level L = Low Voltage Level X = Don't Care

*HIGH Z implies outputs are disabled or off. This condition is defined as a high impedance state for the Am93425 and as an output high level for the Am93415.

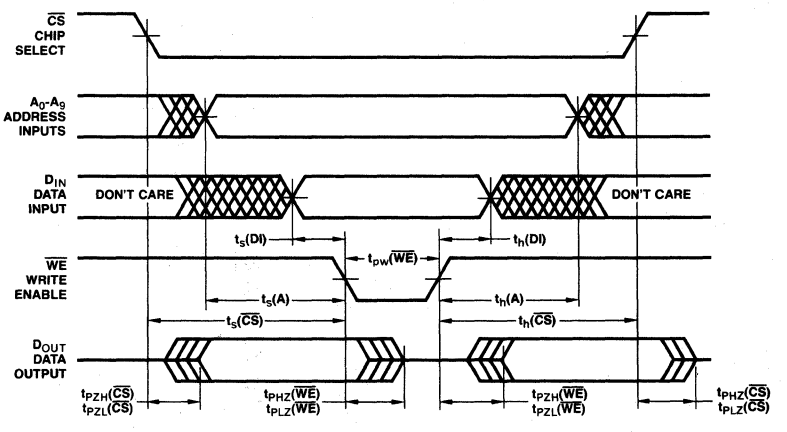
SWITCHING CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

Parameters	Description		Test Conditions	Typ (Note 1)	Am93415/425PC Am93415/425DC Min	Max	Am93415/425DM Am93415/425FM Min	Max	Units
$t_{PLH}(A)$ $t_{PHL}(A)$	Delay from Address to Output (Address Access Time)	See Fig. 2	See Figures 3 and 4 and Notes 3, 4 and 5 (below)	30		45		60	ns
$t_{PZH}(\overline{CS})$ $t_{PZL}(\overline{CS})$	Delay from Chip Select to Active Output and Correct Data	See Fig. 2		15		35		45	ns
$t_{PZH}(\overline{WE})$ $t_{PZL}(\overline{WE})$	Delay from Write Enable to Active Output and Correct Data (Write Recovery)	See Fig. 1		15		40		50	ns
$t_s(A)$	Set-up Time Address (Prior to Initiation of Write)	See Fig. 1		−10	10		15		ns
$t_h(A)$	Hold Time Address (After Termination of Write)	See Fig. 1		−10	5		5		ns
$t_s(DI)$	Set-up Time Data Input (Prior to Initiation of Write)	See Fig. 1		−10	5		5		ns
$t_h(DI)$	Hold Time Data Input (After Termination of Write)	See Fig. 1		−10	5		5		ns
$t_s(\overline{CS})$	Set-up Time Chip Select (Prior to Initiation of Write)	See Fig. 1		−10	5		5		ns
$t_h(\overline{CS})$	Hold Time Chip Select (After Termination of Write)	See Fig. 1		−10	5		5		ns
$t_{pw}(\overline{WE})$	Min Write Enable Pulse Width to Insure Write	See Fig. 1		15	30		40		ns
$t_{PHZ}(\overline{CS})$ $t_{PLZ}(\overline{CS})$	Delay from Chip Select to Inactive Output (HIGH Z)	See Fig. 2		15		35		50	ns
$t_{PHZ}(\overline{WE})$ $t_{PLZ}(\overline{WE})$	Delay from write Enable to Inactive Output (HIGH-Z)	See Fig. 1		15		35		45	ns

Notes:
1. Typical characteristics are at V_{CC} = 5.0V and T_A = 25°C.
2. Input and output capacitance measured on a sample basis using pulse technique.
3. $t_{PLH}(A)$ and $t_{PHL}(A)$ are tested with S_1 closed and C_L = 30pF with both input and output timing referenced to 1.5V.
4. For open collector Am93415, all delays from Write Enable ($\overline{WE}$) or Chip Select ($\overline{CE}$) inputs to the Data Output (D_{OUT}) ($t_{PLZ}(\overline{WE})$, $t_{PLZ}(\overline{CS})$, $t_{PZL}(\overline{WE})$ and $t_{PZL}(\overline{CS})$) are measured with S_1 closed and C_L = 30pF; and with both the input and output timing referenced to 1.5V.
5. For 3-state output Am93425, $t_{PZH}(\overline{WE})$ and $t_{PZH}(\overline{CS})$ are measured with S_1 open, C_L = 30pF and with both the input and output timing referenced to 1.5V. $t_{PZL}(\overline{WE})$ and $t_{PZL}(\overline{CS})$ are measured with S_1 closed, C_L = 30pF and with both the input and output timing referenced to 1.5V. $t_{PHZ}(\overline{WE})$ and $t_{PHZ}(\overline{CS})$ are measured with S_1 open and $C_L \leq$ 5pF and are measured between the 1.5V level on the input to the V_{OH} − 500mV level on the output. $t_{PLZ}(\overline{WE})$ and $t_{PLZ}(\overline{CS})$ are measured with S_1 closed and $C_L \leq$ 5pF and are measured between the 1.5V level on the input and the V_{OL} + 500mV level on the output.

SWITCHING WAVEFORMS

WRITE MODE



KEY TO TIMING DIAGRAM

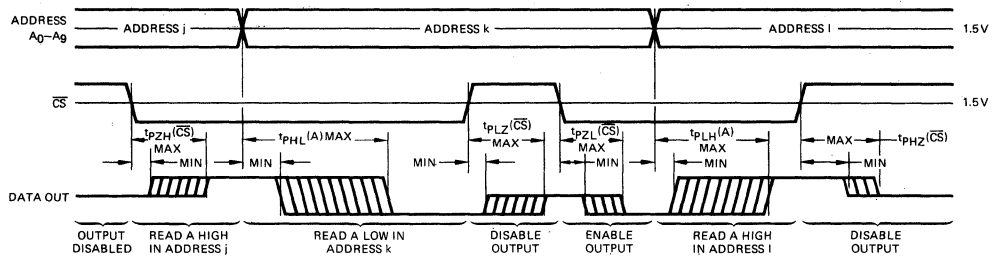
WAVEFORM	INPUTS	OUTPUTS
	MUST BE STEADY	WILL BE STEADY
	MAY CHANGE FROM H TO L	WILL BE CHANGING FROM H TO L
	MAY CHANGE FROM L TO H	WILL BE CHANGING FROM L TO H
	DON'T CARE; ANY CHANGE PERMITTED	CHANGING; STATE UNKNOWN
	DOES NOT APPLY	CENTER LINE IS HIGH IMPEDANCE "OFF" STATE

Figure 1.

BPM-052

SWITCHING WAVEFORMS (Cont.)

READ MODE



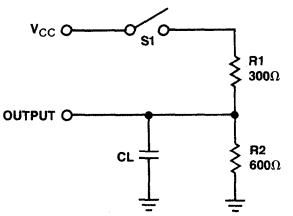
Switching delays from address and chip select inputs to the data output. For the Am93425 disabled output is "OFF", represented by a single center line. For the Am93415, a disabled output is HIGH.

Figure 2.

BPM-053

AC TEST LOAD AND WAVEFORM

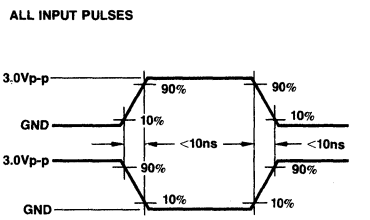
AC TEST LOAD



BPM-054

Figure 3.

INPUT PULSES



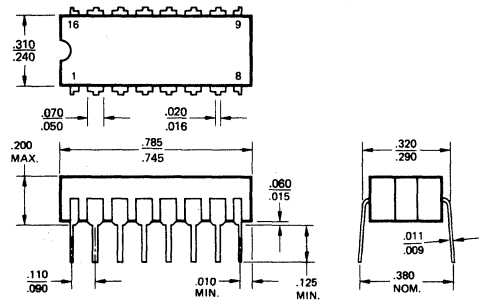
BPM-055

Figure 4.

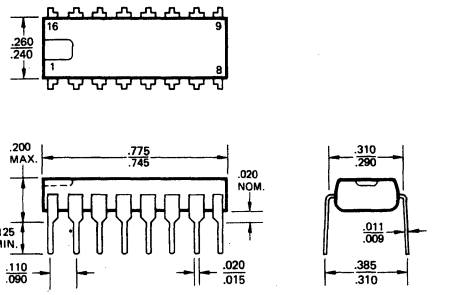
See Notes 3, 4 and 5 of Switching Characteristics.

PHYSICAL DIMENSIONS
Dual-In-Line

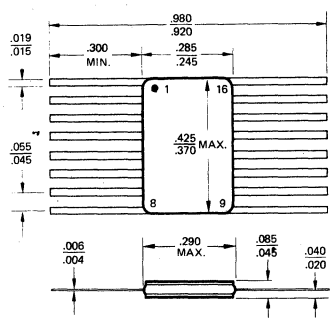
Ceramic



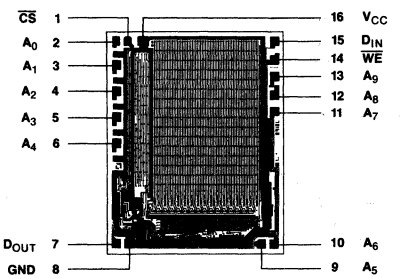
Molded



Flat Package



Metallization and Pad Layout



DIE SIZE 0.096" X 0.119"

Memory Management Products

Am2960

Cascadable 16-Bit Error Detection and Correction Unit

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

- Modified Hamming Code
  Detects multiple errors and corrects single bit errors in a parallel data word. Ideal for use in dynamic memory systems.
- Expandable
  One Am2960 provides EDC on 16-bit data words.
  Two Am2960s provide EDC on 32-bit data words.
  Four Am2960s provide EDC on 64-bit data words.
- Syndromes provided
  The Am2960 makes available the syndrome bits when an error occurs, so the location of memory faults can be logged.
- Microprocessor compatible
  The Am2960 is designed to work with Z8000 microprocessor systems as well as high performance 2900 designs.
- Advanced circuit and process technologies
  Newest 2900 LSI techniques provide very high performance.
  Data-in to error detection typically 30ns
  Data-in to correct data out typically 50ns
- Built-in Diagnostics
  Extra logic on the chip provides diagnostic functions to be used during device test and for system diagnostics.

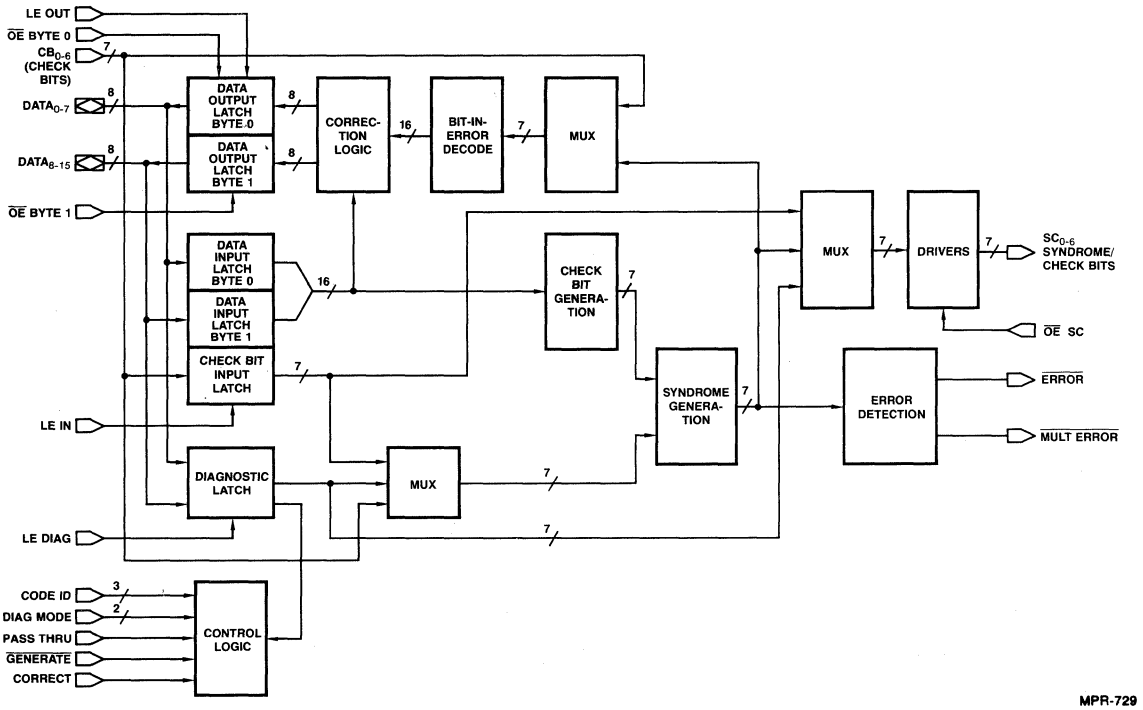
GENERAL DESCRIPTION

The Am2960 Error Detection and Correction Unit (EDC) contains the logic necessary to generate check bits on a 16-bit data field according to a modified Hamming Code, and to correct the data word when check bits are supplied. Operating on data read from memory, the Am2960 will correct any single bit error and will detect all double and some triple bit errors. For 16-bit words, 6 check bits are used. The Am2960 is expandable to operate on 32-bit words (7 check bits) and 64-bit words (8 check bits). In all configurations, the device makes the error syndrome available on separate outputs for data logging.

The Am2960 also features two diagnostic modes, in which diagnostic data can be forced into portions of the chip to simplify device testing and to execute system diagnostic functions. The product is supplied in a 48 lead hermetic DIP package.

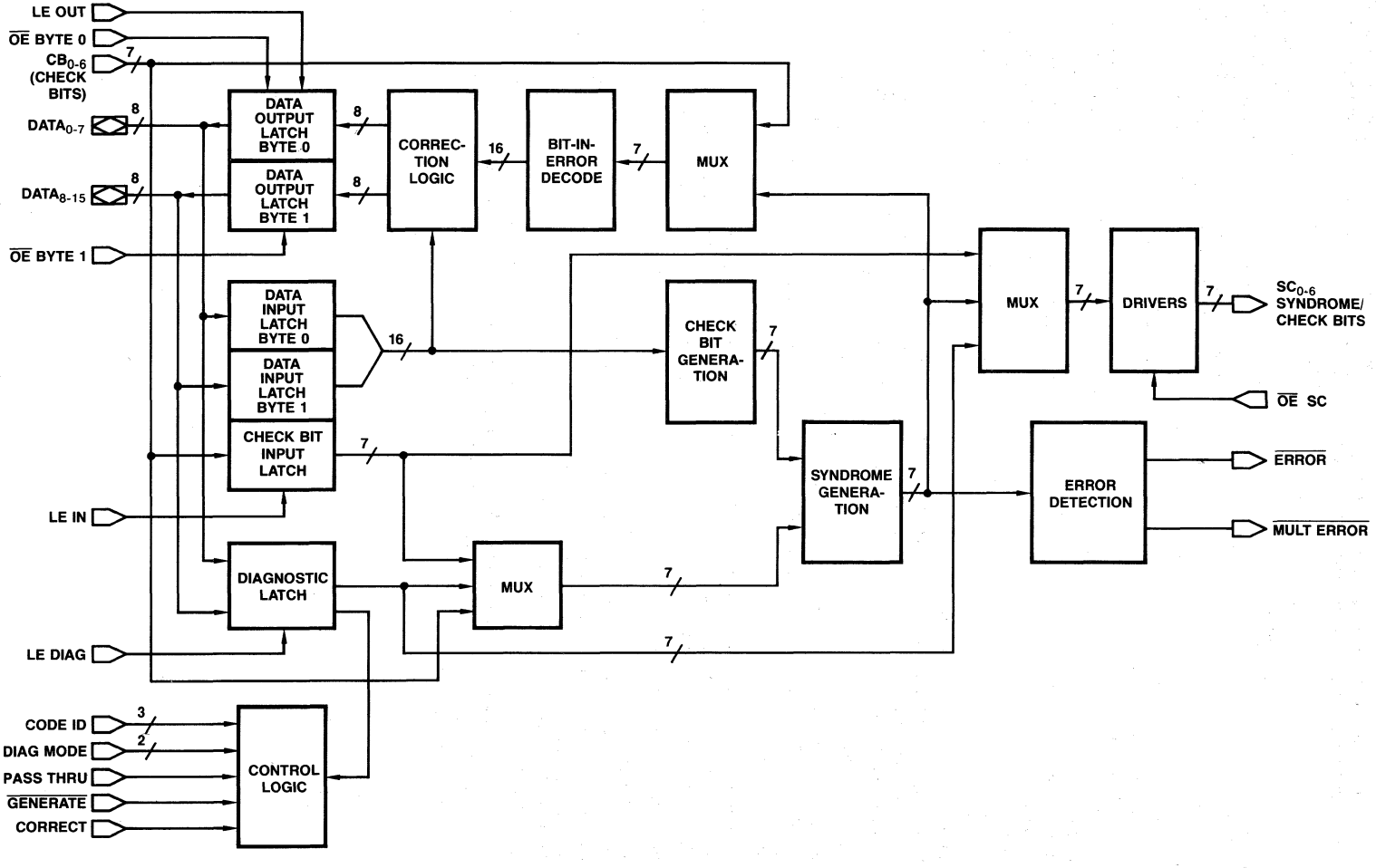
5

BLOCK DIAGRAM



MPR-729

BLOCK DIAGRAM



EDC Architecture

The EDC Unit is a powerful 16-bit cascadable slice used for check bit generation, error detection, error correction and diagnostics.

As shown in the block diagram, the device consists of the following:

- Data Input Latch
- Check Bit Input Latch
- Check Bit Generation Logic
- Syndrome Generation Logic
- Error Detection Logic
- Error Correction Logic
- Data Output Latch
- Diagnostic Latch
- Control Logic

Data Input Latch

16 bits of data are loaded from the bidirectional DATA lines under control of the Latch Enable input, LE IN. Depending on the control mode the input data is either used for check bit generation or error detection/correction.

Check Bit Input Latch

Seven check bits are loaded under control of LE IN. Check bits are used in the Error Detection and Error Correction modes.

Check Bit Generation Logic

This block generates the appropriate check bits for the 16 bits of data in the Data Input Latch. The check bits are generated according to a modified Hamming code.

Syndrome Generation Logic

In both Error Detection and Error Correction modes, this logic block compares the check bits read from memory against a newly generated set of check bits produced for the data read in from memory. If both sets of check bits match, then there are no errors. If there is a mismatch, then one or more of the data or check bits is in error.

The syndrome bits are produced by an exclusive-OR of the two sets of check bits. If the two sets of check bits are identical (meaning there are no errors) the syndrome bits will be all zeroes. If there are errors, the syndrome bits can be decoded to determine the number of errors and the bit-in-error.

Error Detection Logic

This section decodes the syndrome bits generated by the Syndrome Generation Logic. If there are no errors in either the input data or check bits, the $\overline{\text{ERROR}}$ and $\overline{\text{MULTI ERROR}}$ outputs remain HIGH. If one or more errors are detected, $\overline{\text{ERROR}}$ goes LOW. If two or more errors are detected, both $\overline{\text{ERROR}}$ and $\overline{\text{MULTI ERROR}}$ go LOW.

Error Correction Logic

For single errors, the Error Correction Logic complements (corrects) the single data bit in error. This corrected data is loadable into the Data Output Latch, which can then be read onto the bidirectional data lines. If the single error is one of the check bits, the correction logic does not place corrected check bits on the syndrome/check bit outputs. If the corrected check bits are needed the EDC must be switched to Generate Mode.

Data Output Latch

The Data Output Latch is used for storing the result of an error correction operation. The latch is loaded from the correction logic under control of the Data Output Latch Enable, LE OUT. The Data Output Latch may also be loaded directly from the Data Input Latch under control of the PASS THRU control input.

The Data Output Latch is split into two 8-bit (byte) latches which may be enabled independently for reading onto the bidirectional data lines.

Diagnostic Latch

This is a 16-bit latch loadable from the bidirectional data lines under control of the Diagnostic Latch Enable, LE DIAG. The Diagnostic Latch contains check bit information in one byte and control information in the other byte. The Diagnostic Latch is used for driving the device when in Internal Control Mode, or for supplying check bits when in one of the Diagnostic Modes.

Control Logic

The control logic determines the specific mode the device operates in. Normally the control logic is driven by external control inputs. However, in Internal Control Mode, the control signals are instead read from the Diagnostic Latch.

PIN DEFINITIONS

$DATA_{0-15}$ 16 bidirectional data lines. They provide input to the Data Input Latch and Diagnostic Latch, and receive output from the Data Output Latch. $DATA_0$ is the least significant bit; $DATA_{15}$ the most significant.

CB_{0-6} Seven Check Bit input lines. The check bit lines are used to input check bits for error detection. Also used to input syndrome bits for error correction in 32 and 64-bit configurations.

LE IN Latch Enable – Data Input Latch. Controls latching of the input data. When HIGH the Data Input Latch and Check Bit Input Latch follow the input data and input check bits. When LOW, the Data Input Latch and Check Bit Input Latch are latched to their previous state.

$\overline{\text{GENERATE}}$ Generate Check Bits input. When this input is LOW the EDC is in the Check Bit Generate Mode. When HIGH the EDC is in the Detect Mode or Correct Mode.

In the Generate Mode the circuit generates the check bits or partial check bits specific to the data in the Data Input Latch. The generated check bits are placed on the SC outputs.

In the Detect or Correct Modes the EDC detects single and multiple errors, and generates syndrome bits based upon the contents of the Data Input Latch and Check Bit Input Latch. In Correct Mode, single-bit errors are also automatically corrected – corrected data is placed at the inputs of the Data Output Latch. The syndrome result is placed on the SC outputs and indicates in a coded form the number of errors and the bit-in-error.

SC_{0-6} Syndrome/Check Bit outputs. These seven lines hold the check/partial-check bits when the EDC is in Generate Mode, and will hold the syndrome/partial syndrome bits when the device is in Detect or Correct Modes. These are 3-state outputs.

$\overline{\text{OE SC}}$ Output Enable – Syndrome/Check Bits. When LOW, the 3-state output lines SC_{0-6} are enabled. When HIGH, the SC outputs are in the high impedance state.

$\overline{\text{ERROR}}$ Error Detected output. When the EDC is in Detect or Correct Mode, this output will go LOW if one or more syndrome bits are asserted, meaning there are one or more bit errors in the data or check bits. If no syndrome bits are asserted, there are no errors detected and the output will be HIGH. In Generate Mode, $\overline{\text{ERROR}}$ is forced HIGH. (In a 64-bit configuration, $\overline{\text{ERROR}}$ must be externally implemented.)

$\overline{\text{MULTI ERROR}}$ Multiple Errors Detected output. When the EDC is in Detect or Correct Mode, this output if LOW indicates that there are two or more bit errors that have been detected. If HIGH this indicates that either one or no errors have been detected. In Generate Mode, $\overline{\text{MULTI ERROR}}$ is forced HIGH. (In a 64-bit configuration, $\overline{\text{MULTI ERROR}}$ must be externally implemented.)

CORRECT Correct input. When HIGH this signal allows the correction network to correct any single-bit error in the Data Input Latch (by complementing the bit-in-error) before putting it onto the Data Output Latch. When LOW the EDC will drive data directly from the Data Input Latch to the Data Output Latch without correction.

LE OUT Latch Enable – Data Output Latch. Controls the latching of the Data Output Latch. When LOW the Data Output Latch is latched to its previous state. When HIGH the Data Output Latch follows the output of the Data Input Latch as modified by the correction logic network. In Correct Mode, single-bit errors are corrected by the network before loading into the Data Output Latch. In Detect Mode, the contents of the Data Input Latch are passed through the correction network unchanged into the Data Output Latch. The inputs to the Data Output Latch are unspecified if the EDC is in Generate Mode.

$\overline{\text{OE}}$ BYTE 0, $\overline{\text{OE}}$ BYTE 1 Output Enable – Bytes 0 and 1, Data Output Latch. These lines control the 3-state outputs for each of the two bytes of the Data Output Latch. When LOW these lines enable the Data Output Latch and when HIGH these lines force the Data Output Latch into the high impedance state. The two enable lines can be separately activated to enable only one byte of the Data Output Latch at a time.

PASS THRU Pass Thru input. This line when HIGH forces the contents of the Check Bit Input Latch onto the Syndrome/Check Bit outputs (SC_{0-6}) and the unmodified contents of the Data Input Latch onto the inputs of the Data Output Latch.

DIAG $MODE_{0-1}$ Diagnostic Mode Select. These two lines control the initialization and diagnostic operation of the EDC.

CODE ID_{0-2} Code Identification inputs. These three bits identify the size of the total data word to be processed and which 16-bit slice of larger data words a particular EDC is processing. The three allowable data word sizes are 16, 32 and 64 bits and their respective modified Hamming codes are designated 16/22, 32/39 and 64/72. Special CODE ID input 001 (ID_2, ID_1, ID_0) is also used to instruct the EDC that the signals CODE ID_{0-2}, DIAG $MODE_{0-1}$, CORRECT and PASS THRU are to be taken from the Diagnostic Latch, rather than from the input control lines.

LE DIAG Latch Enable – Diagnostic Latch. When HIGH the Diagnostic Latch follows the 16-bit data on the input lines. When LOW the outputs of the Diagnostic Latch are latched to their previous states. The Diagnostic Latch holds diagnostic check bits, and internal control signals for CODE ID_{0-2}, DIAG $MODE_{0-1}$, CORRECT and PASS THRU.

FUNCTIONAL DESCRIPTION

The EDC contains the logic necessary to generate check bits on a 16-bit data field according to a modified Hamming code. Operating on data read from memory, the EDC will correct any single-bit error, and will detect all double and some triple-bit errors. The Am2960 may be configured to operate on 16-bit data words (with 6 check bits), 32-bit data words (with 7 check bits) and 64-bit data words (with 8 check bits). In all configurations, the device makes the error syndrome bits available on separate outputs for error data logging.

Code and Byte Specification

The EDC may be configured in several different ways and operates differently in each configuration. It is necessary to indicate to the device what size data word is involved and which bytes of the data word it is processing. This is done with input lines CODE ID_{0-2}, as shown in Table I. The three modified Hamming codes referred to in Table I are:

- 16/22 code – 16 data bits
  – 6 check bits
  – 22 bits in total.
- 32/39 code – 32 data bits
  – 7 check bits
  – 39 bits in total.
- 64/72 code – 64 data bits
  – 8 check bits
  – 72 bits in total.

CODE ID input 001 (ID_2, ID_1, ID_0) is a special code used to operate the device in Internal Control Mode (described later in this section).

TABLE I. HAMMING CODE AND SLICE IDENTIFICATION.

CODE ID_2	CODE ID_1	CODE ID_0	Hamming Code and Slice Selected
0	0	0	Code 16/22
0	0	1	Internal Control Mode
0	1	0	Code 32/39, Bytes 0 and 1
0	1	1	Code 32/39, Bytes 2 and 3
1	0	0	Code 64/72, Bytes 0 and 1
1	0	1	Code 64/72, Bytes 2 and 3
1	1	0	Code 64/72, Bytes 4 and 5
1	1	1	Code 64/72, Bytes 6 and 7

Control Mode Selection

The device control lines are $\overline{\text{GENERATE}}$, CORRECT, PASS THRU, DIAG $MODE_{0-1}$ and CODE ID_{0-2}. Table II indicates the control modes selected by various combinations of the control line inputs.

Diagnostics

Table III shows specifically how DIAG $MODE_{0-1}$ select between normal operation, initialization and one of two diagnostic modes.

The Diagnostic Modes allow the user to operate the EDC under software control in order to verify proper functioning of the device.

Check and Syndrome Bit Labeling

The check bits generated in the EDC are designated as follows:

- 16-bit configuration – CX C0, C1, C2, C4, C8;
- 32-bit configuration – CX, C0, C1, C2, C4, C8, C16;
- 64-bit configuration – CX, C0, C1, C2, C4, C8, C16, C32.

Syndrome bits are similarly labeled SX through S32. There are only 6 syndrome bits in the 16-bit configuration, 7 for 32 bits and 8 syndrome bits in the 64-bit configuration.

FUNCTIONAL DESCRIPTION – 16-BIT DATA WORD CONFIGURATION

The 16-bit format consists of 16 data bits, 6 check bits and is referred to as 16/22 code (see Figure 1).

The 16-bit configuration is shown in Figure 2.

Generate Mode

In this mode check bits will be generated that correspond to the contents of the Data Input Latch. The check bits generated are placed on the outputs SC_{0-5} (SC_6 is unspecified for 16-bit operation).

Check bits are generated according to a modified Hamming code. Details of the code for check bit generation are contained in Table IV. Each check bit is generated as either an XOR or XNOR of eight of the 16 data bits as indicated in the table. The XOR function results in an even parity check bit, the XNOR is an odd parity check bit.

Detect Mode

In this mode the device examines the contents of the Data Input Latch against the Check Bit Input Latch, and will detect all single-bit errors, all double-bit errors and some triple-bit errors. If one or more errors are detected, $\overline{\text{ERROR}}$ goes LOW. If two or more errors are detected, $\overline{\text{MULTI ERROR}}$ goes LOW. Both error indicators are HIGH if there are no errors.

Also available on device outputs SC_{0-5} are the syndrome bits generated by the error detection step. The syndrome bits may be decoded to determine if a bit error was detected and, for single-bit errors, which of the data or check bits is in error. Table V gives the chart for decoding the syndrome bits generated by the 16-bit configuration (as an example, if the syndrome bits SX/S0/S1/S2/S4/S8 were 101001 this would be decoded to indicate that there is a single-bit error at data bit 9). If no error is detected the syndrome bits will all be zeroes.

In Detect Mode, the contents of the Data Input Latch are driven directly to the inputs of the Data Output Latch without correction.

Correct Mode

In this mode, the EDC functions the same as in Detect Mode except that the correction network is allowed to correct (complement) any single-bit error of the Data Input Latch before putting it onto the inputs of the Data Output Latch. If multiple errors are detected, the output of the correction network is unspecified. If the single-bit error is a check bit there is no automatic correction. If check bit correction is desired, this can be done by placing the device in Generate Mode to produce a correct check bit sequence for the data in the Data Input Latch.

Pass Thru Mode

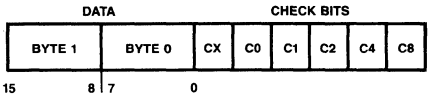
In this mode, the unmodified contents of the Data Input Latch are placed on the inputs of the Data Output Latch and the contents of the Check Bit Input Latch are placed on outputs SC_{0-5}.

TABLE II. EDC CONTROL MODE SELECTION.

$\overline{\text{GENERATE}}$	CORRECT	PASS THRU	DIAG $\text{MODE}_{0\text{-}1}$ (DM_1, DM_0)	CODE $\text{ID}_{0\text{-}2}$ (ID_2, ID_1, ID_0)	Control Mode Selected
LOW	LOW	LOW	00	Not 001	Generate
LOW	LOW	LOW	01	Not 001	Generate Using Diagnostic Latch
LOW	LOW	LOW	10	Not 001	Generate
LOW	LOW	LOW	11	Not 001	Initialize
LOW	LOW	HIGH	00	Not 001	Pass Thru
LOW	LOW	HIGH	01	Not 001	Pass Thru
LOW	LOW	HIGH	10	Not 001	Pass Thru
LOW	LOW	HIGH	11	Not 001	Undefined
LOW	HIGH	LOW	00	Not 001	Generate
LOW	HIGH	LOW	01	Not 001	Generate Using Diagnostic Latch
LOW	HIGH	LOW	10	Not 001	Generate
LOW	HIGH	LOW	11	Not 001	Initialize
LOW	HIGH	HIGH	00	Not 001	Pass Thru
LOW	HIGH	HIGH	01	Not 001	Pass Thru
LOW	HIGH	HIGH	10	Not 001	Pass Thru
LOW	HIGH	HIGH	11	Not 001	Undefined
HIGH	LOW	LOW	00	Not 001	Detect
HIGH	LOW	LOW	01	Not 001	Detect
HIGH	LOW	LOW	10	Not 001	Detect Using Diagnostic Latch
HIGH	LOW	LOW	11	Not 001	Initialize
HIGH	LOW	HIGH	00	Not 001	Pass Thru
HIGH	LOW	HIGH	01	Not 001	Pass Thru
HIGH	LOW	HIGH	10	Not 001	Pass Thru
HIGH	LOW	HIGH	11	Not 001	Undefined
HIGH	HIGH	LOW	00	Not 001	Correct
HIGH	HIGH	LOW	01	Not 001	Correct
HIGH	HIGH	LOW	10	Not 001	Correct Using Diagnostic Latch
HIGH	HIGH	LOW	11	Not 001	Initialize
HIGH	HIGH	HIGH	00	Not 001	Pass Thru
HIGH	HIGH	HIGH	01	Not 001	Pass Thru
HIGH	HIGH	HIGH	10	Not 001	Pass Thru
HIGH	HIGH	HIGH	11	Not 001	Undefined
Any	Any	Any	Any	001	Internal Control Using Diagnostic Latch

TABLE III. DIAGNOSTIC MODE CONTROL.

DIAG MODE_1	DIAG MODE_0	Diagnostic Mode Selected
0	0	**Non-diagnostic mode.** The EDC functions normally in all modes.
0	1	**Diagnostic Mode A.** The contents of the Diagnostic Latch are substituted for the normally generated check bits when in the Generate Mode. The EDC functions normally in the Detect or Correct modes.
1	0	**Diagnostic Mode B.** In the Detect or Correct Mode, the contents of the Diagnostic Latch are substituted for the check bits normally read from the Check Bit Input Latch. The EDC functions normally in the Generate Mode.
1	1	**Initialize.** The inputs of the Data Output Latch are forced to zeroes and the check bits generated correspond to the all-zero data.



Uses Modified Hamming Code 16/22
- 16 data bits
- 6 check bits
- 22 bits in total

Figure 1. 16 Bit Data Format.

MPR-730

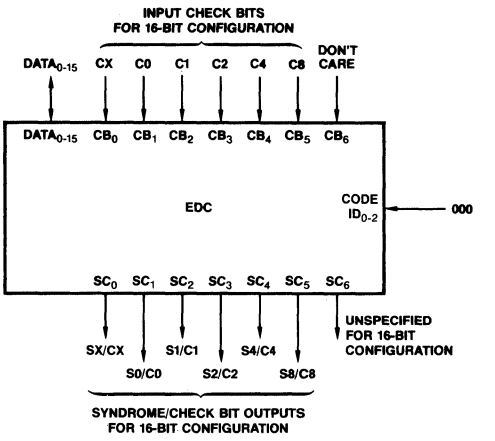


Figure 2. 16 Bit Configuration.

MPR-731

5

TABLE IV. 16-BIT MODIFIED HAMMING CODE – CHECK BIT ENCODE CHART.

Generated Check Bits	Parity	Participating Data Bits															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CX	Even (XOR)		X	X	X		X			X	X		X			X	
C0	Even (XOR)	X	X	X		X		X		X		X		X			
C1	Odd (XNOR)	X			X	X			X		X	X			X		X
C2	Odd (XNOR)	X	X				X	X	X				X	X	X		
C4	Even (XOR)			X	X	X	X	X	X							X	X
C8	Even (XOR)									X	X	X	X	X	X	X	X

The check bit is generated as either an XOR or XNOR of the eight data bits noted by an "X" in the table.

TABLE V. SYNDROME DECODE TO BIT-IN-ERROR.

Syndrome Bits SX	S0	S1	S8	S4	S2								
			0	0	0	1	0	1	0	1	0	1	1
			0	0	1	1	0	0	1	1	1	1	1
			0	0	0	0	1	1	1	1			
0	0	0		*	C8	C4	T	C2	T	T	M		
0	0	1		C1	T	T	15	T	13	7	T		
0	1	0		C0	T	T	M	T	12	6	T		
0	1	1		T	10	4	T	0	T	T	M		
1	0	0		CX	T	T	14	T	11	5	T		
1	0	1		T	9	3	T	M	T	T	M		
1	1	0		T	8	2	T	1	T	T	M		
1	1	1		M	T	T	M	T	M	M	T		

* – no errors detected
Number – the location of the single bit-in-error
T – two errors detected
M – three or more errors detected

TABLE VI. DIAGNOSTIC LATCH LOADING – 16-BIT FORMAT.

Data Bit	Internal Function
0	Diagnostic Check Bit X
1	Diagnostic Check Bit 0
2	Diagnostic Check Bit 1
3	Diagnostic Check Bit 2
4	Diagnostic Check Bit 4
5	Diagnostic Check Bit 8
6, 7	Don't Care
8	CODE ID 0
9	CODE ID 1
10	CODE ID 2
11	DIAG MODE 0
12	DIAG MODE 1
13	CORRECT
14	PASS THRU
15	Don't Care

Diagnostic Latch

The Diagnostic Latch serves both for diagnostic uses and internal control uses. It is loaded from the DATA lines under the control of LE DIAG. Table VI shows the loading definitions for the DATA lines.

Generate Using Diagnostic Latch (Diagnostic Mode A)

Detect Using Diagnostic Latch (Diagnostic Mode B)

Correct Using Diagnostic Latch (Diagnostic Mode B)

These are special diagnostic modes selected by DIAG $MODE_{0-1}$ where either normal check bit inputs or outputs are substituted for by check bits loaded into the Diagnostic Latch. See Table III for details.

Internal Control Mode

This mode is selected by CODE ID_{0-2} input 001 (ID_2, ID_1, ID_0).

When in Internal Control Mode, the EDC takes the CODE ID_{0-2}, DIAG $MODE_{0-1}$, CORRECT and PASS THRU control signals from the internal Diagnostic Latch rather than from the external input lines.

Table VI gives the format for loading the Diagnostic Latch.

FUNCTIONAL DESCRIPTION – 32-BIT DATA WORD CONFIGURATION

The 32-bit format consists of 32 data bits, 7 check bits and is referred to as 32/39 code (see Figure 3).

The 32-bit configuration is shown in Figure 4.

The upper EDC (Slice 0/1) handles the least significant bytes 0 and 1 – the external DATA lines 0 to 15 are connected to the same numbered inputs of the upper device. The lower EDC (Slice 2/3) handles the most significant bytes 2 and 3 – the external DATA lines for bits 16 to 31 are connected to inputs $DATA_0$ through $DATA_{15}$ respectively.

The valid syndrome and check bit outputs are those of Slice 2/3 as shown in the diagram. In Correct Mode these must be read into Slice 0/1 via the CB inputs, thus requiring external buffering and output enabling of the check bit lines as shown. The $\overline{OE}$ SC signal can be used to control enabling of check bit inputs – when syndrome outputs are enabled, the external check bit inputs will be disabled.

The valid $\overline{ERROR}$ and $\overline{MULTI\ ERROR}$ outputs are those of the Slice 2/3. The $\overline{ERROR}$ and $\overline{MULTI\ ERROR}$ outputs of Slice 0/1 are unspecified. All of the latch enables and control signals must be input to both of the devices.

Generate Mode

In this mode check bits will be generated that correspond to the contents of the Data Input Latch. The check bits generated are placed on the outputs SC_{0-6} of Slice 2/3.

Check bits are generated according to a modified Hamming code. Details of the code for check bit generation are contained in Table VII. Check bits are generated as either an XOR or XNOR of 16 of the 32 data bits as indicated in the table. The XOR function results in an even parity check bit, the XNOR in an odd parity check bit.

Detect Mode

In this mode the device examines the contents of the Data Input Latch against the Check Bit Input Latch, and will detect all single-bit errors, all double-bit errors and some triple-bit errors. If one or more errors are detected, $\overline{ERROR}$ goes LOW. If two or more errors are detected, $\overline{MULTI\ ERROR}$ goes LOW. Both error indicators are HIGH if there are no errors. The valid ERROR and MULTI ERROR signals are those of Slice 2/3 – those of Slice 0/1 are undefined.

Also available on Slice 2/3 outputs SC_{0-6} are the syndrome bits generated by the error detection step. The syndrome bits may be decoded to determine if a bit error was detected and, for single-bit errors, which of the data or check bits is in error. Table VIII gives the chart for decoding the syndrome bits generated for the 32-bit configuration (as an example, if the syndrome bits SX/S0/S1/S2/S4/S8/S16 were 0010011 this would be decoded to indicate that there is a single-bit error at data bit 25). If no error is detected the syndrome bits will be all zeroes.

In Detect Mode, the contents of the Data Input Latch are driven directly to the inputs of the Data Output Latch without corrections.

Correct Mode

In this mode, the EDC functions the same as in Detect Mode except that the correction network is allowed to correct (complement) any single-bit error of the Data Input Latch before putting it onto the inputs of the Data Output Latch. If multiple errors are detected, the output of the correction network is unspecified. If the single-bit error is a check bit there is no automatic correction – if desired this would be done by placing the device in Generate Mode to produce a correct check bit sequence for the data in the Data Input Latch.

For data correction, both Slices 0/1 and 2/3 require access to the syndrome bits on Slice 2/3's outputs SC_{0-6}. Slice 2/3 has access to these syndrome bits through internal data paths, but for Slice 0/1 they must be read through the inputs CB_{0-6}. The device connections for this are shown in Figure 4. When in Correct Mode the SC outputs must be enabled so that they are available for reading in through the CB inputs.

Pass Thru Mode

In this mode, the unmodified contents of the Data Input Latch are placed on the inputs of the Data Output Latch and the contents of the Check Bit Input Latch are placed on outputs SC_{0-6} of Slice 2/3.

Uses Modified Hamming Code 32/39
- 32 data bits
- 7 check bits
- 39 bits in total

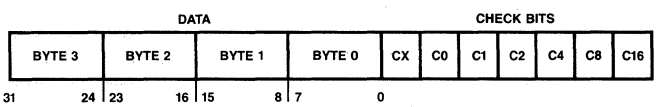


Figure 3. 32 Bit Data Format.

MPR-732

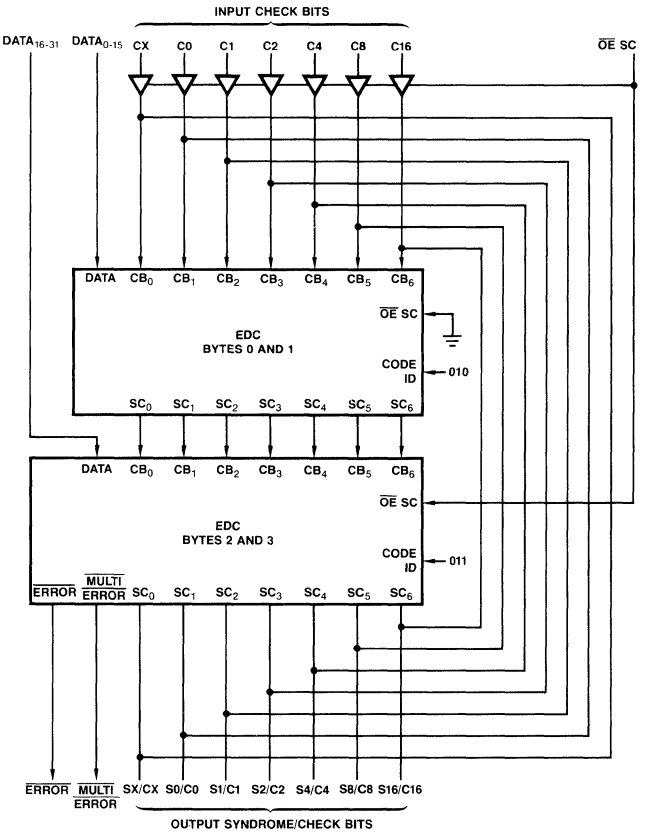


Figure 4. 32 Bit Configuration.

MPR-733

5

TABLE VII. 32-BIT MODIFIED HAMMING CODE – CHECK BIT ENCODE CHART.

Generated Check Bits	Parity	Participating Data Bits															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CX	Even (XOR)	X				X		X	X	X	X		X			X	
C0	Even (XOR)	X	X	X		X		X		X		X		X			
C1	Odd (XNOR)	X			X	X			X		X	X			X		X
C2	Odd (XNOR)	X	X				X	X	X				X	X	X		
C4	Even (XOR)			X	X	X	X	X	X							X	X
C8	Even (XOR)									X	X	X	X	X	X	X	X
C16	Even (XOR)	X	X	X	X	X	X	X	X								

Generated Check Bits	Parity	Participating Data Bits															
		16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
CX	Even (XOR)		X	X	X		X					X		X	X		X
C0	Even (XOR)	X	X	X		X		X		X		X		X			
C1	Odd (XNOR)	X			X	X			X		X	X			X		X
C2	Odd (XNOR)	X	X				X	X	X				X	X	X		
C4	Even (XOR)			X	X	X	X	X	X							X	X
C8	Even (XOR)									X	X	X	X	X	X	X	X
C16	Even (XOR)									X	X	X	X	X	X	X	X

The check bit is generated as either an XOR or XNOR of the sixteen data bits noted by an "X" in the table.

TABLE VIII. SYNDROME DECODE TO BIT-IN-ERROR.

Syndrome Bits SX	S0	S1	S2	S16	0	1	0	1	0	1	0	1
				S8	0	0	1	1	0	0	1	1
				S4	0	0	0	0	1	1	1	1
0	0	0	0		*	C16	C8	T	C4	T	T	30
0	0	0	1		C2	T	T	27	T	5	M	T
0	0	1	0		C1	T	T	25	T	3	15	T
0	0	1	1		T	M	13	T	23	T	T	M
0	1	0	0		C0	T	T	24	T	2	M	T
0	1	0	1		T	1	12	T	22	T	T	M
0	1	1	0		T	M	10	T	20	T	T	M
0	1	1	1		16	T	T	M	T	M	M	T
1	0	0	0		CX	T	T	M	T	M	14	T
1	0	0	1		T	M	11	T	21	T	T	M
1	0	1	0		T	M	9	T	19	T	T	31
1	0	1	1		M	T	T	29	T	7	M	T
1	1	0	0		T	M	8	T	18	T	T	M
1	1	0	1		17	T	T	28	T	6	M	T
1	1	1	0		M	T	T	26	T	4	M	T
1	1	1	1		T	0	M	T	M	T	T	M

* – no errors detected
Numbers – number of the single bit-in-error
T – two errors detected
M – three or more errors detected

Diagnostic Latch

The Diagnostic Latch serves both for diagnostic uses and internal control uses. It is loaded from the DATA lines under the control of LE DIAG. Table IX shows the loading definitions for the DATA lines.

Generate Using Diagnostic Latch (Diagnostic Mode A)

Detect Using Diagnostic Latch (Diagnostic Mode B)

Correct Using Diagnostic Latch (Diagnostic Mode B)

These are special diagnostic modes selected by DIAG $MODE_{0\text{-}1}$ where either normal check bit inputs or outputs are substituted for by check bits from the Diagnostic Latch. See Table III for details.

Internal Control Mode

This mode is selected by CODE $ID_{0\text{-}2}$, input 001 (ID_2, ID_1, ID_0).

When in Internal Control Mode the device takes the CODE $ID_{0\text{-}2}$, DIAG $MODE_{0\text{-}1}$, CORRECT and PASS THRU signals from the internal Diagnostic Latch rather than from the external control lines.

Table IX gives the format for loading the Diagnostic Latch.

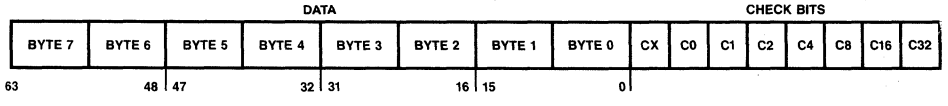
FUNCTIONAL DESCRIPTION – 64-BIT DATA WORD CONFIGURATION

The 64-bit format consists of 64 data bits, 8 check bits and is referred to as 64/72 code (see Figure 5).

The configuration to process 64-bit format is shown in Figure 6. In this configuration a portion of the syndrome generation and error detection is implemented externally of the EDCs in MSI. For error correction the syndrome bits generated must be read back into all four EDCs through the CB inputs. This necessitates the check bit buffering shown in the connection diagram of Figure 6. The $\overline{OE}$ SC signal can control the check bit enabling – when syndrome bit outputs are enabled the external check bit lines will be disabled so that the syndrome bits may be read onto the CB inputs.

TABLE IX. DIAGNOSTIC LATCH LOADING – 32-BIT FORMAT.

Data Bit	Internal Function
0	Diagnostic Check Bit X
1	Diagnostic Check Bit 0
2	Diagnostic Check Bit 1
3	Diagnostic Check Bit 2
4	Diagnostic Check Bit 4
5	Diagnostic Check Bit 8
6	Diagnostic Check Bit 16
7	Don't Care
8	Slice 0/1 – CODE ID 0
9	Slice 0/1 – CODE ID 1
10	Slice 0/1 – CODE ID 2
11	Slice 0/1 – DIAG MODE 0
12	Slice 0/1 – DIAG MODE 1
13	Slice 0/1 – CORRECT
14	Slice 0/1 – PASS THRU
15	Don't Care
16-23	Don't Care
24	Slice 2/3 – CODE ID 0
25	Slice 2/3 – CODE ID 1
26	Slice 2/3 – CODE ID 2
27	Slice 2/3 – DIAG MODE 0
28	Slice 2/3 – DIAG MODE 1
29	Slice 2/3 – CORRECT
30	Slice 2/3 – PASS THRU
31	Don't Care



Uses Modified Hamming Code 64/72
– 64 data bits
– 8 check bits
– 72 bits in total

Figure 5. 64 Bit Data Format.

MPR-734

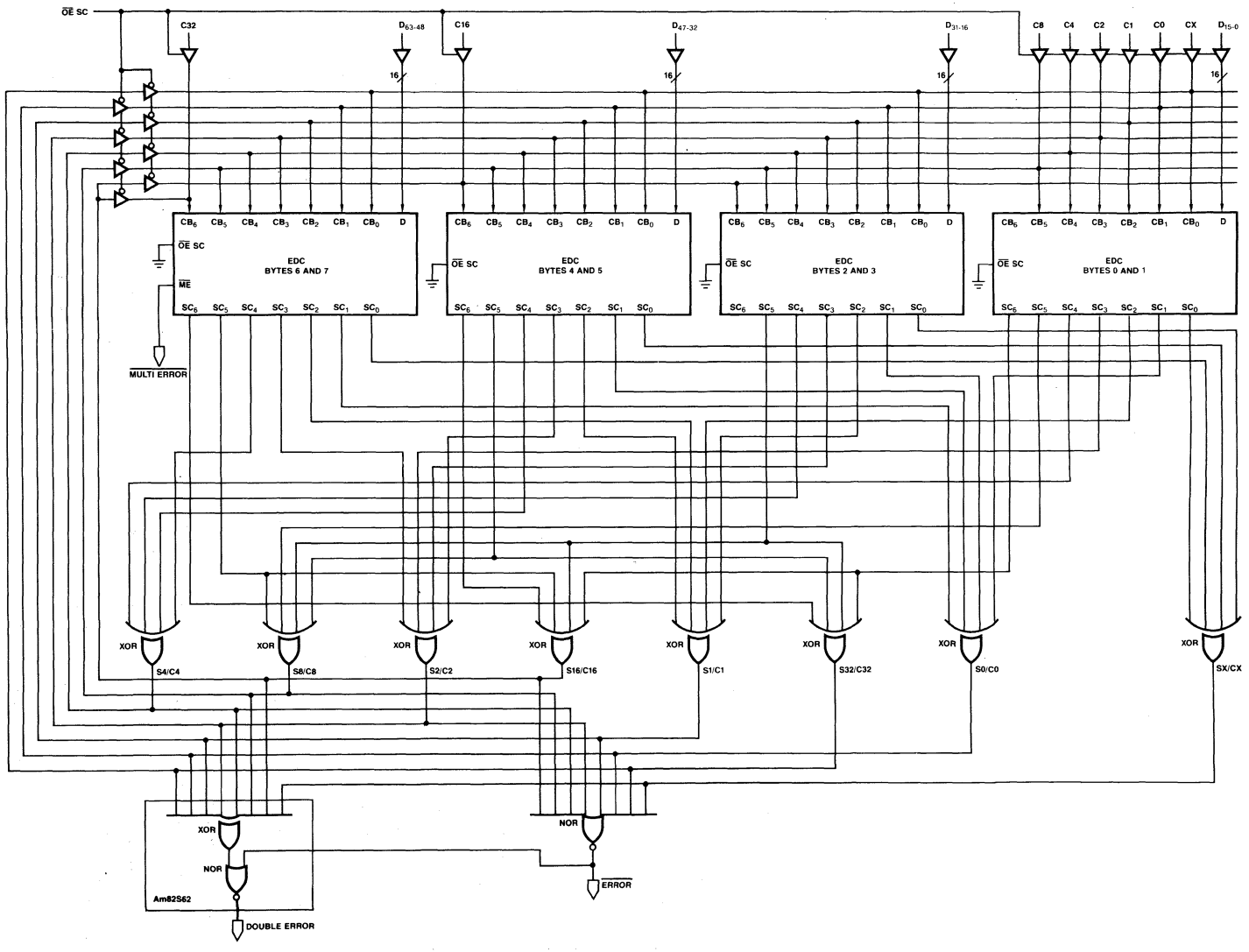


Figure 6. Am2960 – 64 Bit Data Configuration.

MPR-735

Input CB_6 of Slice 0/1 is unused and need not be connected. Output SC_6 of Slice 2/3 is unspecified for the 64-bit configuration.

The error detection signals for the 64-bit configuration differ from the 16 and 32-bit configurations. The $\overline{\text{ERROR}}$ signal functions the same: it is LOW if one or more errors are detected, and HIGH if no errors are detected. The DOUBLE ERROR signal is HIGH if and only if a double-bit error is detected – it is LOW otherwise. All of the $\overline{\text{MULTI ERROR}}$ outputs of the four devices are valid. $\overline{\text{MULTI ERROR}}$ is LOW when three or more errors are detected; it is HIGH if either zero, one or two errors are detected. This is a different meaning for $\overline{\text{MULTI ERROR}}$ than in other configurations.

Generate Mode

In this mode check bits will be generated that correspond to the contents of the Data Input Latch. The check bits generated appear at the outputs of the XOR gates as indicated in Figure 6.

Check bits are generated according to a modified Hamming code. Details of the code for check bit generation are contained in Table X. Check bits are generated as either an XOR or XNOR

TABLE X. 64-BIT MODIFIED HAMMING CODE – CHECK BIT ENCODE.

Generated Check Bits	Parity	Participating Data Bits															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CX	Even (XOR)		X	X	X		X			X	X		X			X	
C0	Even (XOR)	X	X	X		X		X		X		X		X			
C1	Odd (XNOR)	X			X	X			X		X	X			X		X
C2	Odd (XNOR)	X	X				X	X	X				X	X	X		
C4	Even (XOR)			X	X	X	X	X	X							X	X
C8	Even (XOR)									X	X	X	X	X	X	X	X
C16	Even (XOR)	X	X	X	X	X	X	X	X								
C32	Even (XOR)	X	X	X	X	X	X	X	X								

Generated Check Bits	Parity	Participating Data Bits															
		16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
CX	Even (XOR)		X	X	X		X			X	X		X			X	
C0	Even (XOR)	X	X	X		X		X		X		X		X			
C1	Odd (XNOR)	X			X	X			X		X	X			X		X
C2	Odd (XNOR)	X	X				X	X	X				X	X	X		
C4	Even (XOR)			X	X	X	X	X	X							X	X
C8	Even (XOR)									X	X	X	X	X	X	X	X
C16	Even (XOR)									X	X	X	X	X	X	X	X
C32	Even (XOR)									X	X	X	X	X	X	X	X

Generated Check Bits	Parity	Participating Data Bits															
		32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
CX	Even (XOR)	X				X		X	X			X		X	X		X
C0	Even (XOR)	X	X	X		X		X		X		X		X			
C1	Odd (XNOR)	X			X	X			X		X	X			X		X
C2	Odd (XNOR)	X	X				X	X	X				X	X	X		
C4	Even (XOR)			X	X	X	X	X	X							X	X
C8	Even (XOR)									X	X	X	X	X	X	X	X
C16	Even (XOR)	X	X	X	X	X	X	X	X								
C32	Even (XOR)									X	X	X	X	X	X	X	X

Generated Check Bits	Parity	Participating Data Bits															
		48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
CX	Even (XOR)	X				X		X	X			X		X	X		X
C0	Even (XOR)	X	X	X		X		X		X		X		X			
C1	Odd (XNOR)	X			X	X			X		X	X			X		X
C2	Odd (XNOR)	X	X				X	X	X				X	X	X		
C4	Even (XOR)			X	X	X	X	X	X							X	X
C8	Even (XOR)									X	X	X	X	X	X	X	X
C16	Even (XOR)									X	X	X	X	X	X	X	X
C32	Even (XOR)	X	X	X	X	X	X	X	X								

The check bit is generated as either an XOR or XNOR of the 32 data bits noted by an "X" in the table.

of 32 of the 64 bits as indicated in the table. The XOR function results in an even parity check bit, the XNOR in an odd parity check bit.

Detect Mode

In this mode the device examines the contents of the Data Input Latch against the Check Bit Input Latch, and will detect all single-bit errors, all double-bit errors and some triple-bit errors. If one or more errors are detected, $\overline{\text{ERROR}}$ goes LOW. If exactly two errors are detected, DOUBLE ERROR goes HIGH. If three or more errors are detected, $\overline{\text{MULTI ERROR}}$ goes LOW – the $\overline{\text{MULTI ERROR}}$ output of any of the four EDCs may be used.

Available as XOR gate outputs are the generated syndrome bits (see Figure 6). The syndrome bits may be decoded to determine if a bit error was detected and, for single-bit errors, which of the data or check bits is in error. Table XI gives the chart for encoding the syndrome bits generated for the 64-bit configuration (as an example, if the syndrome bits SX/S0/S1/S2/S4/S8/S16/S32 were 00100101 this would be decoded to indicate that there is a single-bit error at data bit 41). If no error is detected the syndrome bits will all be zeroes.

In Detect Mode the contents of the Data Input Latch are driven directly to the inputs of the Data Output Latch without corrections.

Correct Mode

In this mode, the EDC functions the same as in Detect Mode except that the correction network is allowed to correct (complement) any single-bit error of the Data Input Latch before putting it onto the inputs of the Data Output Latch. If multiple errors are detected, the output of the correction network is unspecified. If the single bit error is a check bit there is no automatic correction. Check bit correction can be done by placing the device in generate mode to produce a correct check bit sequence for the data in the Data Input Latch.

To perform the correction step, all four slices require access to the syndrome bits which are generated externally of the devices. This access is provided by reading the syndrome bits in through the CB inputs. The device connections for this are shown in Figure 6. When in Correct Mode the SC outputs must be enabled so that the syndrome bits are available at the CB inputs.

Pass Thru Mode

In this mode, the unmodified contents of the Data Input Latch are placed on the inputs of the Data Output Latch, and the contents of the Check Bit Input Latch are passed through the external XOR network and appear unmodified at the XOR gate outputs labeled CX to C32 (see Figure 6).

5

TABLE XI. SYNDROME DECODE TO BIT-IN-ERROR.

Syndrome Bits				S32	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
				S16	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
				S8	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
				S4	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
SX	S0	S1	S2																	
0	0	0	0		*	C32	C16	T	C8	T	T	M	C4	T	T	M	T	46	62	T
0	0	0	1		C2	T	T	M	T	43	59	T	T	53	37	T	M	T	T	M
0	0	1	0		C1	T	T	M	T	41	57	T	T	51	35	T	15	T	T	31
0	0	1	1		T	M	M	T	13	T	T	29	23	T	T	7	T	M	M	T
0	1	0	0		C0	T	T	M	T	40	56	T	T	50	34	T	M	T	T	M
0	1	0	1		T	49	33	T	12	T	T	28	22	T	T	6	T	M	M	T
0	1	1	0		T	M	M	T	10	T	T	26	20	T	T	4	T	M	M	T
0	1	1	1		16	T	T	0	T	M	M	T	T	M	M	T	M	T	T	M
1	0	0	0		CX	T	T	M	T	M	M	T	T	M	M	T	14	T	T	30
1	0	0	1		T	M	M	T	11	T	T	27	21	T	T	5	T	M	M	T
1	0	1	0		T	M	M	T	9	T	T	25	19	T	T	3	T	47	63	T
1	0	1	1		M	T	T	M	T	45	61	T	T	55	39	T	M	T	T	M
1	1	0	0		T	M	M	T	8	T	T	24	18	T	T	2	T	M	M	T
1	1	0	1		17	T	T	1	T	44	60	T	T	54	38	T	M	T	T	M
1	1	1	0		M	T	T	M	T	42	58	T	T	52	36	T	M	T	T	M
1	1	1	1		T	48	32	T	M	T	T	M	M	T	T	M	T	M	M	T

* – no errors detected
Number – the number of the single bit-in-error
T – two errors detected
M – more than two errors detected

Diagnostic Latch

The Diagnostic Latch serves both for diagnostic uses and internal control uses. It is loaded from the DATA lines under the control of LE DIAG. Table XII shows the loading definitions for the DATA lines.

Generate Using Diagnostic Latch (Diagnostic Mode A)

Detect Using Diagnostic Latch (Diagnostic Mode B)

Correct Using Diagnostic Latch (Diagnostic Mode B)

These are special diagnostic modes selected by DIAG $MODE_{0\text{-}1}$ where either normal check bit inputs or outputs are substituted for by check bits from the Diagnostic Latch. See Table III for details.

Internal Control Mode

This mode is selected by CODE $ID_{0\text{-}2}$, input 001 (ID_2, ID_1, ID_0).

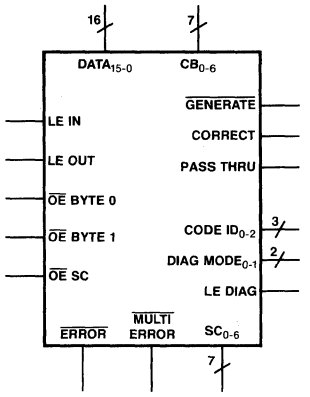
When in Internal Control Mode the EDC takes the CODE $ID_{0\text{-}2}$, DIAG $MODE_{0\text{-}1}$, CORRECT and PASS THRU signals from the internal Diagnostic Latch rather than from the external control lines. Table XII gives format for loading the Diagnostic Latch.

TABLE XII. DIAGNOSTIC LATCH LOADING – 64-BIT FORMAT.

Data Bit	Internal Function
0	Diagnostic Check Bit X
1	Diagnostic Check Bit 0
2	Diagnostic Check Bit 1
3	Diagnostic Check Bit 2
4	Diagnostic Check Bit 4
5	Diagnostic Check Bit 8
6, 7	Don't Care
8	Slice 0/1 – CODE ID 0
9	Slice 0/1 – CODE ID 1
10	Slice 0/1 – CODE ID 2
11	Slice 0/1 – DIAG MODE 0
12	Slice 0/1 – DIAG MODE 1
13	Slice 0/1 – CORRECT
14	Slice 0/1 – PASS THRU
15	Don't Care
16-23	Don't Care
24	Slice 2/3 – CODE ID 0
25	Slice 2/3 – CODE ID 1
26	Slice 2/3 – CODE ID 2
27	Slice 2/3 – DIAG MODE 0
28	Slice 2/3 – DIAG MODE 1
29	Slice 2/3 – CORRECT
30	Slice 2/3 – PASS THRU

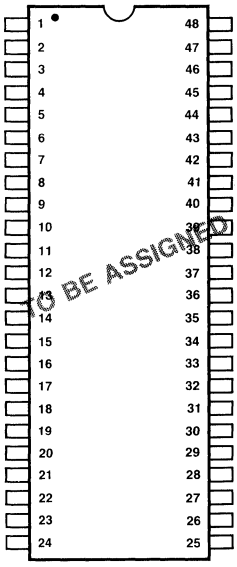
Data Bit	Internal Function
31	Don't Care
32-37	Don't Care
38	Diagnostic Check Bit 16
39	Don't Care
40	Slice 4/5 – CODE ID 0
41	Slice 4/5 – CODE ID 1
42	Slice 4/5 – CODE ID 2
43	Slice 4/5 – DIAG MODE 0
44	Slice 4/5 – DIAG MODE 1
45	Slice 4/5 – CORRECT
46	Slice 4/5 – PASS THRU
47	Don't Care
48-54	Don't Care
55	Diagnostic Check Bit 32
56	Slice 6/7 – CODE ID 0
57	Slice 6/7 – CODE ID 1
58	Slice 6/7 – CODE ID 2
59	Slice 6/7 – DIAG MODE 0
60	Slice 6/7 – DIAG MODE 1
61	Slice 6/7 – CORRECT
62	Slice 6/7 – PASS THRU
63	Don't Care

LOGIC SYMBOL



MPR-736

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

MPR-737

5

APPLICATIONS

The EDC unit may be used in two different ways depending upon whether the design objective is to minimize the total time for memory operations or to minimize system complexity.

Check-Only Configuration

Figure 7 shows the EDC unit in a Check-Only configuration. This method minimizes the delay needed on memory operations for EDC.

On writes to memory the EDC unit generates check-bits. On reads from memory, the EDC unit monitors the data and check bits. If an error is detected, the CPU is interrupted – the CPU then issues the appropriate control signals to the EDC unit to correct the data, write back into memory and/or perform diagnostics and error logging.

Correct-Always Configuration

Figure 8 illustrates the use of EDC unit in a Correct-Always configuration. This method has the advantage of being less complex than the Check-Only configuration. Memory operations are slower in this configuration since on reads from memory, data is always passed through the EDC unit for correction. This method does not necessarily slow down total system speed if the CPU cycle-time is the constraint on system speed.

In the Correct-Always configuration, check bits are generated by the EDC on writes into memory. On reads from memory, the check and data bits are loaded into the EDC unit – after a delay, corrected data is available at the EDC unit's outputs and is placed on the system bus. If the data was not correctable, then the $\overline{\text{MULTI ERROR}}$ output will go active and may be used to interrupt the CPU.

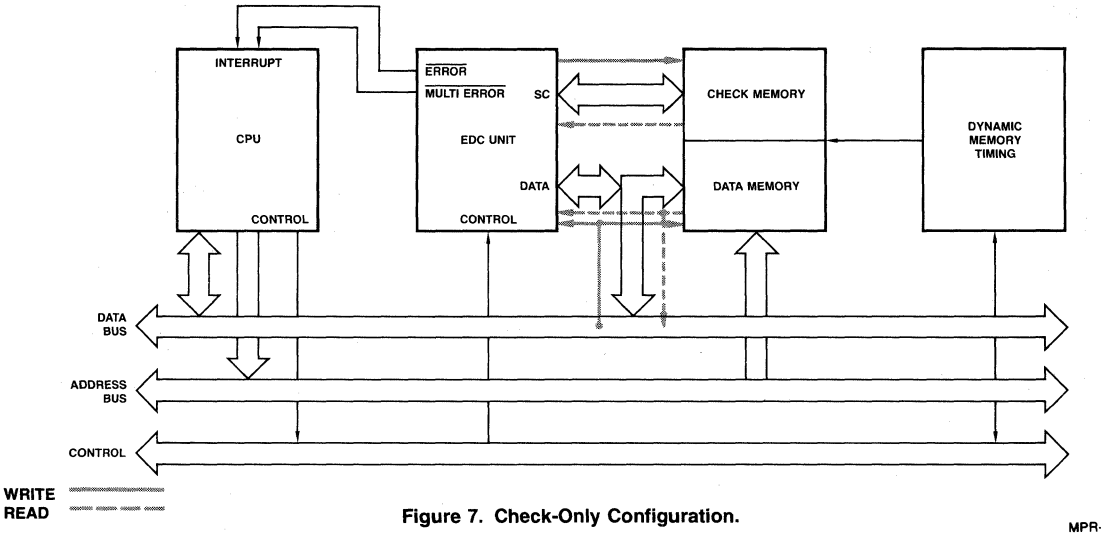


Figure 7. Check-Only Configuration.

MPR-738

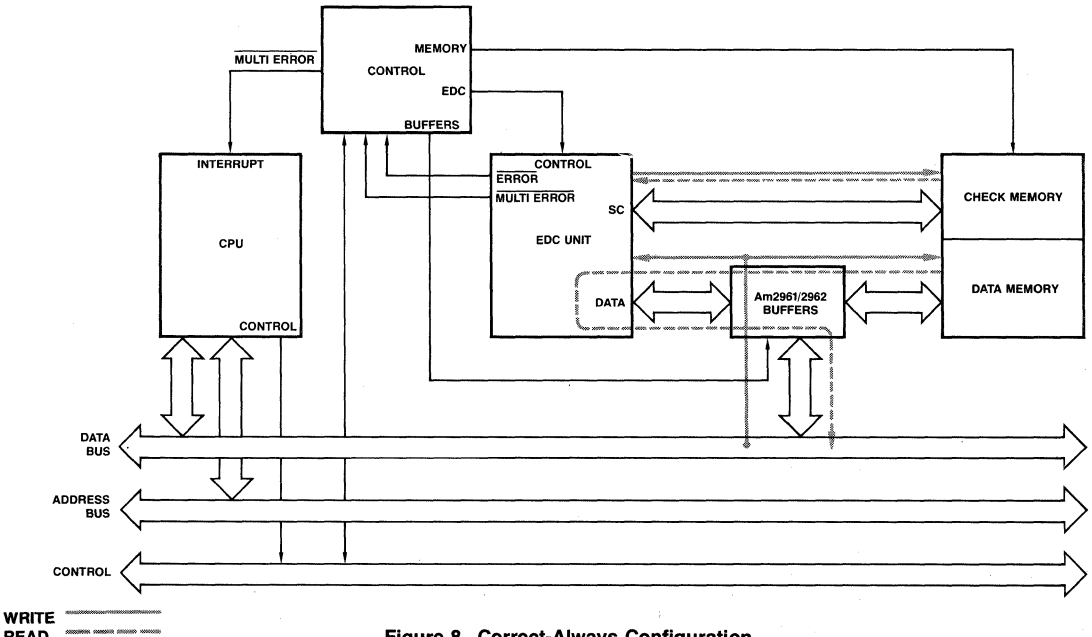


Figure 8. Correct-Always Configuration.

MPR-739

Am2961 • Am2962

4-Bit Error Correction Multiple Bus Buffers

DISTINCTIVE CHARACTERISTICS

- Quad high-speed LSI bus-transceiver
- Provides complete data path interface between the Am2960 Error Detection and Correction Unit, the system data bus and dynamic RAM memory
- Three-state 24mA output to data bus
- Three-state data output to memory
- Inverting data bus for Am2961 and non-inverting for Am2962
- Data bus latches allow operation with multiplexed buses
- Advanced low-power Schottky processing
- Space saving 24-pin 0.3" package
- 100% product assurance screening to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

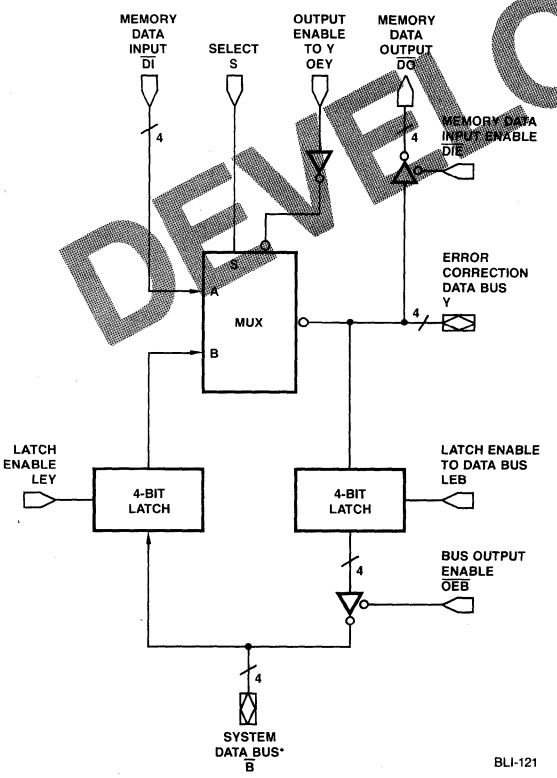
The Am2961 and Am2962 are high-performance, low-power Schottky multiple bus buffers that provide the complete data path interface between the Am2960 Error Detection and Correction Unit, dynamic RAM memory and the system data bus. The Am2961 provides an inverting data path between the data bus (B_i) and the Am2960 error correction data input (Y_i) and the Am2962 provides a non-inverting configuration (B_i to Y_i). Both devices provide inverting data paths between the Am2960 and memory data bus thereby optimizing internal data path speeds.

The Am2961 and Am2962 are 4-bit devices. Four devices are used to interface each 16-bit Am2960 Error Detection and Correction Unit with dynamic memory. The system can easily be expanded to 32 or more bits for wider memory applications. The 4-bit configuration allows enabling the appropriate devices two-at-a-time for intermixed word or byte, read and write in 16-bit systems with error correction.

Data latches between the error correction data bus and the system data bus facilitate byte writing in memory systems wider than 8-bits. They also provide a data holding capability during single-step system operation.

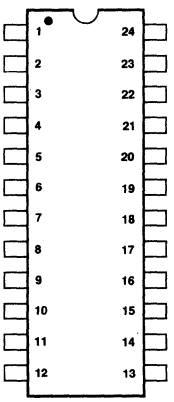
IN DEVELOPMENT

LOGIC DIAGRAM



*Am2962 is the same function but non-inverting to the system data bus, B.

CONNECTION DIAGRAM
Top View



24 pin slim (0.3")

BLI-122

Note: Pin 1 is marked for orientation.

Am2964

Dynamic Memory Controller

IN DEVELOPMENT

DISTINCTIVE CHARACTERISTICS

- Dynamic Memory Controller for 16K and 64K MOS dynamic RAMs
- 8-Bit Refresh Counter for refresh address generation, has clear input and terminal count output
- Refresh Counter terminal count selectable at 256 or 128
- Latch Input $\overline{RAS}$ Decoder provides 4 $\overline{RAS}$ outputs, all active during refresh
- Dual 8-Bit Address Latches plus separate $\overline{RAS}$ Decoder Latches
- Grouping functions on a common chip minimizes speed differential/skew between address, $\overline{RAS}$ and $\overline{CAS}$ outputs
- 3-Port, 8-Bit Address Multiplexer with Schottky speed
- Burst mode, distributed refresh or transparent refresh mode determined by user
- Non-inverting address, $\overline{RAS}$ and $\overline{CAS}$ paths
- 100% product assurance screening to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

The Am2964 Dynamic Memory Controller replaces several MSI devices by grouping several unique functions. Two 8-bit latches capture and hold the memory address. These latches and a clearable, 8-bit refresh counter feed into an 8-bit, 3-input, Schottky speed MUX, for output to the dynamic RAM address lines.

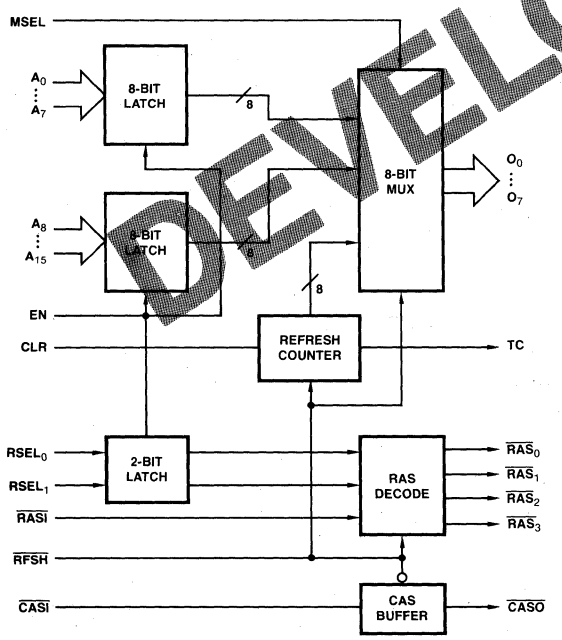
The same silicon chip also includes a special $\overline{RAS}$ decoder and $\overline{CAS}$ buffer. Placing these functions on the same chip minimizes the time skew between output functions which would otherwise be separate MSI chips, and therefore, allows a faster memory cycle time by the amount of skew eliminated.

Pulsing the active LOW refresh line $\overline{RFSH}$, switches the MUX to the counter output, inhibits $\overline{CAS}$, and forces all four $\overline{RAS}$ decoder outputs active simultaneously. The counter is advanced at the end of the refresh cycle – the LOW-to-HIGH transition of $\overline{RFSH}$. Various refresh modes can be accommodated for 16K or 64K RAMs and for a wide variety of processor configurations.

A_{15} is a dual function input which controls the refresh counter's range. For 64K RAMs it is an address input. For 16K RAMs it can be pulled to +12V through 1KΩ to terminate the refresh count at 128 instead of 256.

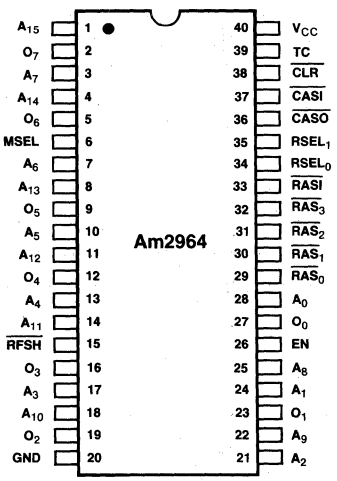
IN DEVELOPMENT

LOGIC DIAGRAM



BLI-123

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BLI-124

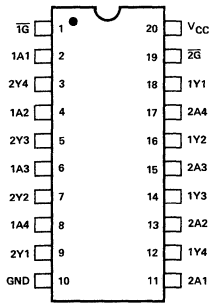
Am2965 • Am2966

Octal Dynamic Memory Drivers with Three-State Outputs

DISTINCTIVE CHARACTERISTICS

- **Controlled rise and fall characteristics**
  Internal resistors provide symmetrical drive to HIGH and LOW states, eliminating need for external series resistor.
- **Output swings designed to drive 16K and 64K RAMs**
  V_{OH} guaranteed at V_{CC} −1.15V. Undershoot going LOW guaranteed at less than 0.5V.
- **Large capacitive drive capability**
  35mA min source or sink current at 2.0V. Propagation delays specified for 50pF and 500pF loads.
- **Pin-compatible with 'S240 and 'S244**
  Non-inverting Am2966 replaces 74S244; inverting Am2965 replaces 74S240. Faster than 'S240/244 under equivalent load.
- **No-glitch outputs**
  Outputs forced into OFF state during power up and down.

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BLI-125

FUNCTIONAL DESCRIPTION

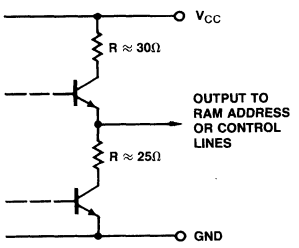
The Am2965 and Am2966 are designed and specified to drive the capacitive input characteristics of the address and control lines of MOS dynamic RAMs. The unique design of the lower output driver includes a collector resistor to control undershoot on the HIGH-to-LOW transition. The upper output driver pulls up to V_{CC} − 1.15V to be compatible with MOS memory and is designed to have a rise time symmetrical with the lower output's controlled fall time. This allows optimization of Dynamic RAM performance.

The Am2965 and Am2966 are pin-compatible with the popular 'S240 and 'S244 with identical 3-state output enable controls. The Am2965 has inverting drivers and the Am2966 has non-inverting drivers.

The inclusion of an internal resistor in the lower output driver eliminates the requirement for an external series resistor, therefore reducing package count and the board area required. The internal resistor controls the output fall and undershoot without slowing the output rise.

These devices are designed for use with the Am2964 Dynamic Memory Controller where large dynamic memories with highly capacitive input lines require additional buffering. Driving eight address lines or four $\overline{RAS}$ and four $\overline{CAS}$ lines with drivers on the same silicon chip also provides a significant performance advantage by minimizing skew between drivers. Each device has specified skew between drivers to improve the memory access worst case timing over the min and max t_{PD} difference of unspecified devices.

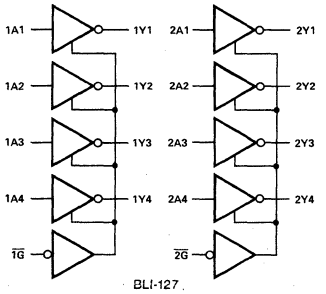
TYPICAL OUTPUT DRIVER



BLI-126

LOGIC DIAGRAMS

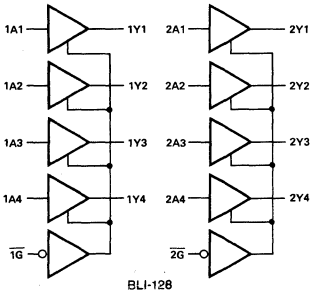
Am2965



BLI-127

Inputs		Outputs
$\overline{G}$	A	Y
H	X	Z
L	H	L
L	L	H

Am2966



BLI-128

Inputs		Outputs
$\overline{G}$	A	Y
H	X	Z
L	L	L
L	H	H

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Temperature (Ambient) Under Bias	−55 to +125°C
Supply Voltage to Ground Potential Continuous	−0.5 to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to +V_{CC} max
DC Input Voltage	−0.5 to +7.0V
DC Output Current, into Outputs	30mA
DC Input Current	−30 to +5.0mA

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

Am2965/66XC, DC, PC	T_A = 0 to 70°C	V_{CC} = 5.0V ±10%	(MIN = 4.50V	MAX = 5.50V)
Am2965/66XM, DM	T_A = −55 to +125°C	V_{CC} = 5.0V ±10%	(MIN = 4.50V	MAX = 5.50V)
Am2965/66FM	T_C = −55 to +125°C	V_{CC} = 5.0V ±10%	(MIN = 4.50V	MAX = 5.50V)

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description		Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Units
V_{OH}	Output High Voltage		V_{CC} = MIN, V_{IN} = V_{IH} or V_{IL}	I_{OH} = −1mA	V_{CC}−1.15	V_{CC}−0.7V		Volts
V_{OL}	Output LOW Voltage		V_{CC} = MIN, V_{IN} = V_{IH} or V_{IL}	I_{OL} = 1mA			0.5	Volts
				I_{OL} = 12mA			0.8	
V_{IH}	Input HIGH Level		Guaranteed input logical HIGH voltage for all inputs		2.0			Volts
V_{IL}	Input LOW Level		Guaranteed input logical LOW voltage for all inputs				0.8	Volts
V_I	Input Clamp Voltage		V_{CC} = MIN, I_{IN} = −18mA				−1.2	Volts
I_{IL}	Input LOW Current		V_{CC} = MAX, V_{IN} = 0.4V				−200	μA
I_{IH}	Input HIGH Current		V_{CC} = MAX, V_{IN} = 2.7V				20	μA
I_I	Input HIGH Current		V_{CC} = MAX, V_{IN} = 7.0V				0.1	mA
I_{OZH}	Off-State Current		V_O = 2.7V				100	μA
I_{OZL}	Off-State Current		V_O = 0.4V				−200	μA
I_{OL}	Output Sink Current		V_{OL} = 2.0V		35			mA
I_{OH}	Output Source Current		V_{OH} = 2.0V		−35			mA
I_{SC}	Output Short Circuit Current (Note 3)		V_{CC} = MAX		−60 (see I_{OH})		−200	mA
I_{CC}	Supply Current	Am2965	All Outputs HIGH	V_{CC} = MAX Outputs Open		24	50	mA
			All Outputs LOW			86	125	
			All Outputs Hi-Z			86	125	
		Am2966	All Outputs HIGH	V_{CC} = MAX Outputs Open		53	75	
			All Outputs LOW			92	130	
			All Outputs Hi-Z			116	150	

Notes: 1. For conditions shown as MIN or MAX, use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

Am2965 • Am2966
SWITCHING CHARACTERISTICS
(T_A = +25°C, V_{CC} = 5.0V)

Parameters	Description	Test Conditions		Min	Typ	Max	Units
t_{PLH}	Propagation Delay Time from LOW-to-HIGH Output	Figure 1 Test Circuit Figure 3 Voltage Levels and Waveforms	C_L = 0pF		6	(Note 4)	ns
			C_L = 50pF	6	9	15	
			C_L = 500pF	15	22	35	
t_{PHL}	Propagation Delay Time from HIGH-to-LOW Output		C_L = 0pF		4	(Note 4)	ns
			C_L = 50pF	6	12	20	
			C_L = 500pF	20	30	45	
t_{PLZ}	Output Disable Time from LOW, HIGH	Figures 2 and 4, S = 1			13	20	ns
t_{PHZ}		Figures 2 and 4, S = 2			8	12	
t_{PZL}	Output Enable Time from LOW, HIGH	Figures 2 and 4, S = 1			13	20	ns
t_{PZH}		Figures 2 and 4, S = 2			13	20	
t_{SKEW}	Output-to-Output Skew	Figures 1 and 3, C_L = 50pF			±0.5	±3.0 (Note 5)	ns
V_{ONP}	Output Voltage Undershoot	Figures 1 and 3, C_L = 50pF			0	−0.5	Volts

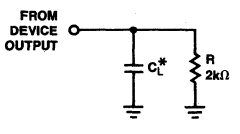
5

SWITCHING CHARACTERISTICS
OVER OPERATING RANGE (Note 6)

Parameters	Description	Test Conditions		COM'L T_A = 0 to 70°C V_{CC} = 5.0V ±10% Min	COM'L Max	MIL (Note 7) T_A = −55 to +125°C V_{CC} = 5.0V ±10% Min	MIL Max	Units
t_{PLH}	Propagation Delay Time LOW-to-HIGH Output	Figures 1 and 3	C_L = 50pF	4	20	4	20	ns
			C_L = 500pF	13	40	13	40	
t_{PHL}	Propagation Delay Time HIGH-to-LOW Output	Figures 1 and 3	C_L = 50pF	4	24	4	24	ns
			C_L = 500pF	17	50	17	50	
t_{PLZ}	Output Disable Time from LOW, HIGH	Figures 2 and 4	S = 1		24		24	ns
t_{PHZ}			S = 2		16		16	
t_{PZL}	Output Enable Time from LOW, HIGH	Figures 2 and 4	S = 1		28		28	ns
t_{PZH}			S = 2		28		28	
V_{ONP}	Output Voltage Undershoot	Figures 1 and 3, C_L = 50pF			−0.5		−0.5	Volts

Notes: 4. Typical time shown for reference only – not tested.
5. Time Skew specification is guaranteed by design but not tested.
6. AC performance over the operating temperature range is guaranteed by testing defined in Group A, Subgroup 9.
7. T_C = −55 to +125°C for Flatpak versions.

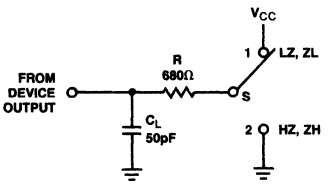
SWITCHING TEST CIRCUITS



BLI-129

*t_{pd} specified at C = 50 and 500pF.

Figure 1. Capacitive Load Switching.

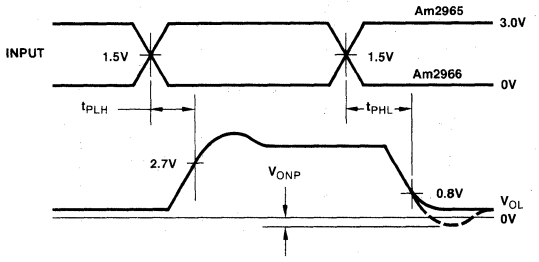


BLI-130

Figure 2. Three-State Enable/Disable.

TYPICAL SWITCHING CHARACTERISTICS

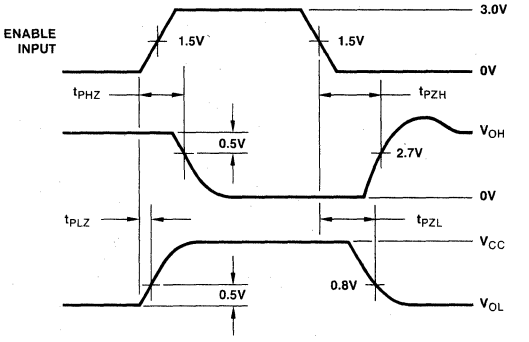
VOLTAGE WAVEFORMS



BLI-131

$t_r = t_f = 2.5ns$
$f = 2.5MHz$
$t_{pw} = 200ns$

Figure 3. Output Drive Levels.



BLI-132

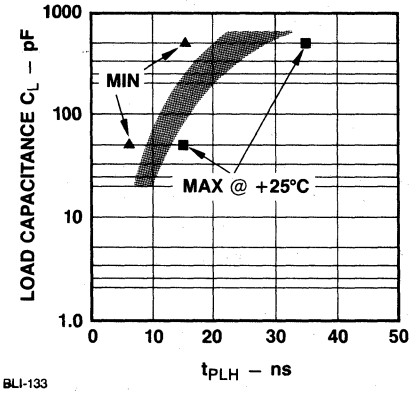
$t_r = t_f = 2.5ns$
$f = 1MHz$
$t_{pw} = 800ns$

Figure 4. Three-State Control Levels.

The RAM Driver symmetrical output design offers significant improvement over a standard Schottky output by providing a balanced drive output impedance (≈33Ω both HIGH and LOW), and by pulling up to MOS V_{OH} levels (V_{CC} – 1.15V). External resistors, not required with the RAM Driver, protect standard Schottky drivers from error causing undershoot but also slow the output rise by adding to the internal R.

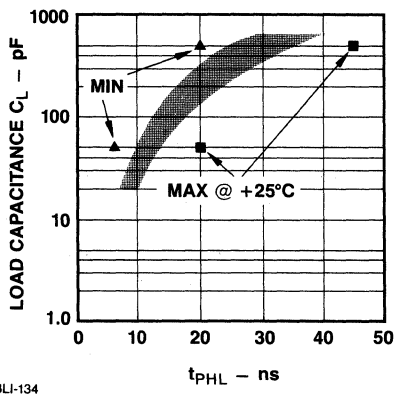
The RAM Driver is optimized to drive LOW at maximum speed based on safe undershoot control and to drive HIGH with a symmetrical speed characteristic. This is an optimum approach because the dominant RAM loading characteristic is input capacitance.

The curves shown below provide performance characteristics typical of both the inverting (Am2965) and non-inverting (Am2966) RAM Drivers.



BLI-133

Figure 5. t_{PLH} for V_{OH} = 2.7 Volts vs. C_L.

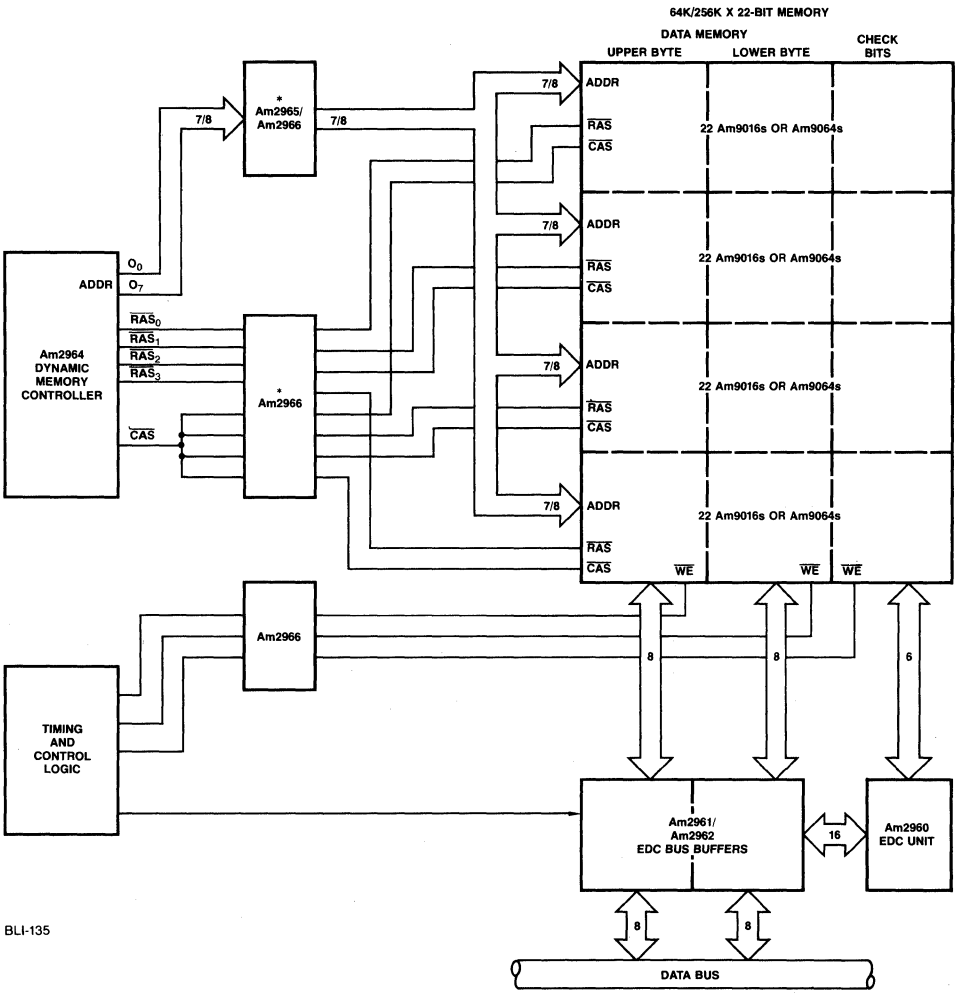


BLI-134

Figure 6. t_{PHL} for V_{OL} = 0.8 Volts vs. C_L.

The curves above depict the typical tPLH and tPHL for the RAM Driver outputs as a function of load capacitance. The minimums and maximums are shown for worst case design. The typical band is provided as a guide for intermediate capacitive loads.

APPLICATION

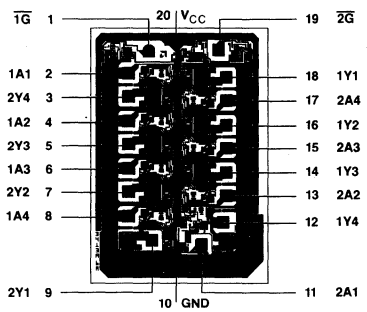


*Address and $\overline{RAS}$/$\overline{CAS}$ drivers each drive 22 RAM inputs at each output. Timing skew is minimized by using one device for address lines and one device for $\overline{RAS}$/$\overline{CAS}$, spreading the $\overline{CAS}$ loading over four drivers to equalize the capacitive load on each driver.

DYNAMIC MEMORY CONTROL WITH ERROR DETECTION AND CORRECTION

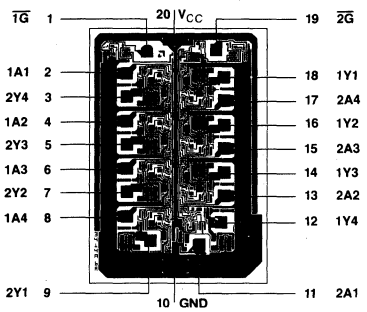
Metallization and Pad Layouts

Am2965



DIE SIZE 0.094" X 0.060"

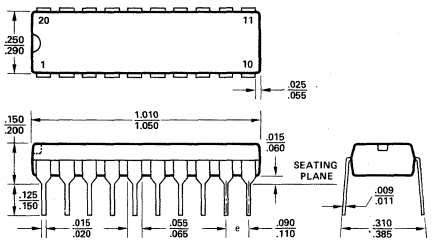
Am2966



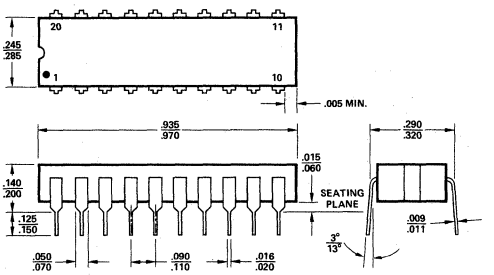
DIE SIZE 0.094" X 0.066"

PHYSICAL DIMENSIONS
Dual-In-Line

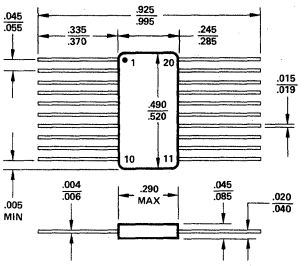
20-Pin Molded DIP



20-Pin Cerdip



20-Pin Flatpack



ORDERING INFORMATION

Order the part number according to the table below to obtain the desired package, temperature range, and screening level.

Am2965 Order Number	Am2966 Order Number	Package Type	Temperature Range	Screening Level
AM2965PC	AM2966PC	P-20	C	C-1
AM2965DC	AM2966DC	D-20	C	C-1
AM2965DCB	AM2966DCB	D-20	C	B-1
AM2965DM	AM2966DM	D-20	M	C-3
AM2965DMB	AM2966DMB	D-20	M	B-3
AM2965FM	AM2966FM	F-20	M	C-3
AM2965FMB	AM2966FMB	F-20	M	B-3
AM2965XC	AM2966XC	Dice	C	Visual inspection to MIL-STD-883 Method 2010B.
AM2965XM	AM2966XM	Dice	M	

Notes: 1. P = Molded DIP, D = Hermetic DIP, F = Flatpak. Number following letter is number of leads. See Appendix B for detailed outline. Where Appendix B contains several dash numbers, any of the variations of the package may be used unless othewise specified.
2. C = 0 to 70°C, V_{CC} = 4.50V to 5.50V, M = −55 to +125°C, V_{CC} = 4.50V to 5.50V.
3. See Appendix A for details of screening. Levels C-1 and C-3 conform to MIL-STD-883, Class C. Level B-3 conforms to MIL-STD-883, Class B.

AmZ8163

Dynamic Memory Timing, Refresh and EDC Controller

DISTINCTIVE CHARACTERISTICS

- Complete AmZ8000 CPU to dynamic RAM contol interface
- $\overline{RAS}/\overline{CAS}$ Sequencer to eliminate delay lines
- Memory request/refresh arbitration
- Controls for Word/Byte read or write
- Complete EDC data path and mode controls
- Refresh interval timer independent of CPU
- Refresh control during Single Step or Halt modes
- EDC error flag latches for error logging under software control
- Also, complete control for 8-Bit MOS μP

FUNCTIONAL DESCRIPTION

The AmZ8163 is a high speed bus interface controller forming an integral part of the AmZ8000 memory support chip set using dynamic MOS RAMs with Error Detection and Correction (EDC). The complete chip set includes the AmZ8127 Clock Generator and Controller, the AmZ8164 Dynamic Memory Controller, the AmZ8161/2 EDC Bus Buffers, the AmZ8160 EDC Unit and optional AmZ8165/6 RAM Drivers.

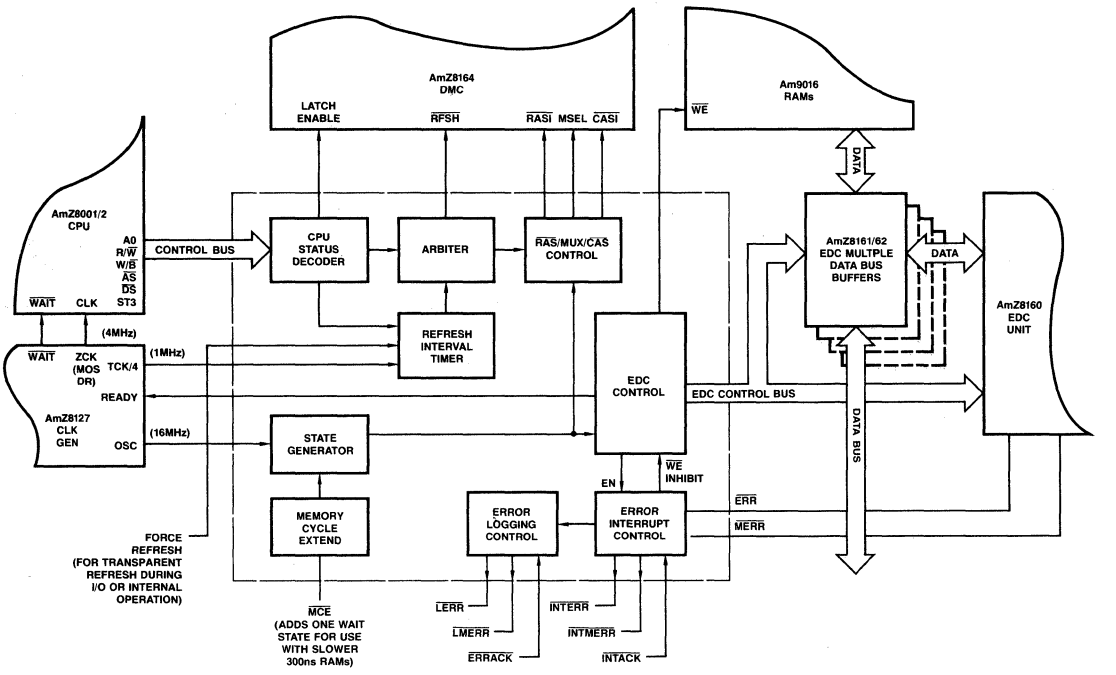
The AmZ8163 provides all of the control interface functions including $\overline{RAS}$/Address MUX/$\overline{CAS}$ timing (without delay lines), refresh timing, memory request/refresh arbitration and all EDC enables and controls. The enable controls are configured for both word and byte operations including the data controls for byte write with error correction. The AmZ8163 generates bus and operating mode controls for the AmZ8160 EDC Unit.

The AmZ8163 uses the AmZ8127 16MHz (4 x CLK) output to generate $\overline{RAS}$/Address MUX/$\overline{CAS}$ timing. An internal refresh interval timer generates the memory refresh request independent of the CPU to guarantee the proper refresh timing under all combinations of CPU and DMA memory requests.

5

FUNCTION DIAGRAM

REFRESH AND EDC CONTROLLER



BLI-167

High-Performance Controller Products

6

Am29116

16-Bit Bipolar Microprocessor

ADVANCE INFORMATION

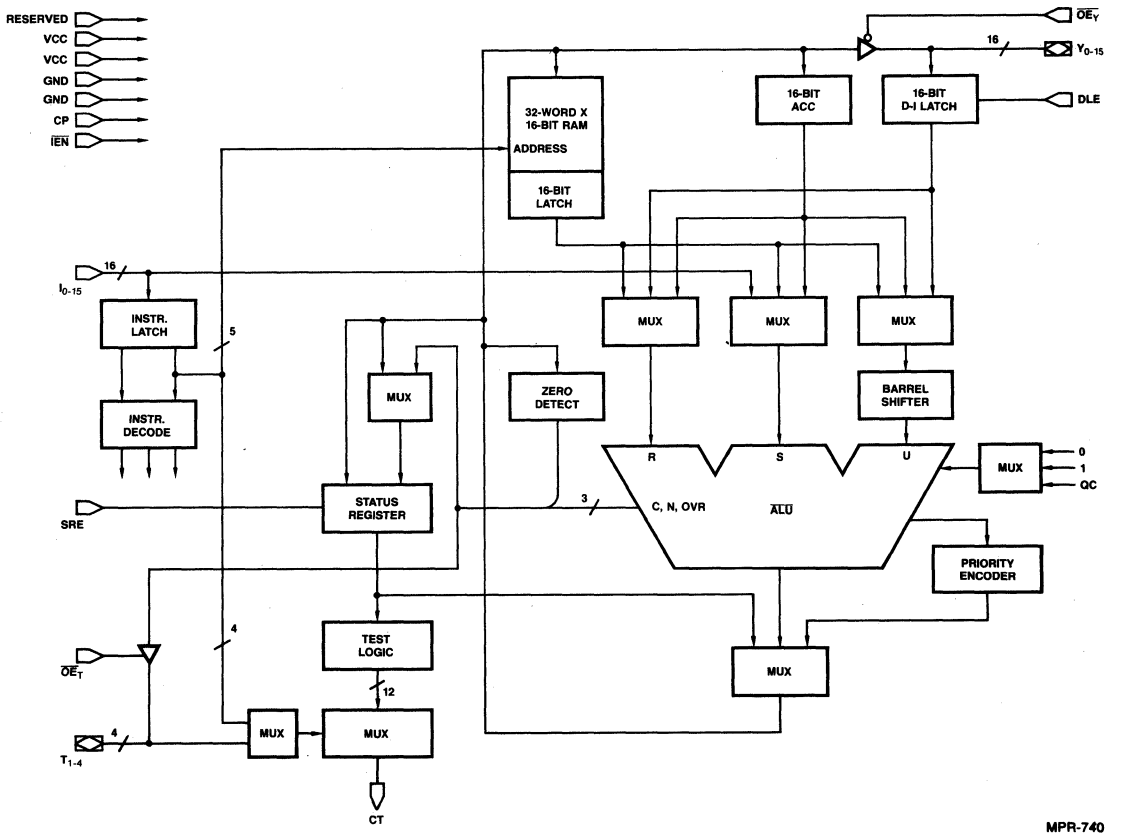
DISTINCTIVE CHARACTERISTICS

- **Designed for Controller Applications**
  Instruction set designed for high performance peripheral controllers, communications controllers, industrial controllers and digital modems . . . but general purpose, too. Excellent solution for applications requiring speed and bit-manipulation power.
- **FAST**
  Design objective of 100ns maximum microcycle time for all instructions. Allows a 10MHz clock rate.
- **Powerful Instruction Set**
  All instructions executable in single cycle on full 16-bit word or on 8-bit byte:
  - Add, Subtract
  - N-bit Rotate
  - Shift-Up/Shift-Down
  - Set-Bit/Reset-Bit
  - Add/Subtract 2^N
  - Rotate & Merge
  - Rotate & Compare
  - CRC Generation
  - Priority Encode
- **Powerful Data Manipulation**
  Full 16-bit data path. 32 registers on chip. Direct data input for immediate mode instructions.
- **48-Pin Dual-in-Line Package**

Note: Am2900 High Performance Controller Products Family.

Refer to the following page for more information on the Am2900 High Performance Controller Products Family.

BLOCK DIAGRAM



Am2940
DMA Address Generator

DISTINCTIVE CHARACTERISTICS

- DMA Address Generation
  Generates memory address, word count and DONE signal for DMA transfer operation.
- Expandable Eight-bit Slice
  Any number of Am2940's can be cascaded to form larger memory addresses – three devices address 16 megawords.
- Repeat Data Transfer Capability
  Initial memory address and word count are saved so that the data transfer can be repeated.
- Programmable Control Modes
  Provides four types of DMA transfer control plus memory address increment/decrement.
- High Speed, Bipolar LSI
  Advanced Low-Power Schottky TTL technology provides typical CLOCK to DONE propagation delay of 50ns and 24mA output current sink capability.
- Microprogrammable
  Executes 8 different instructions.

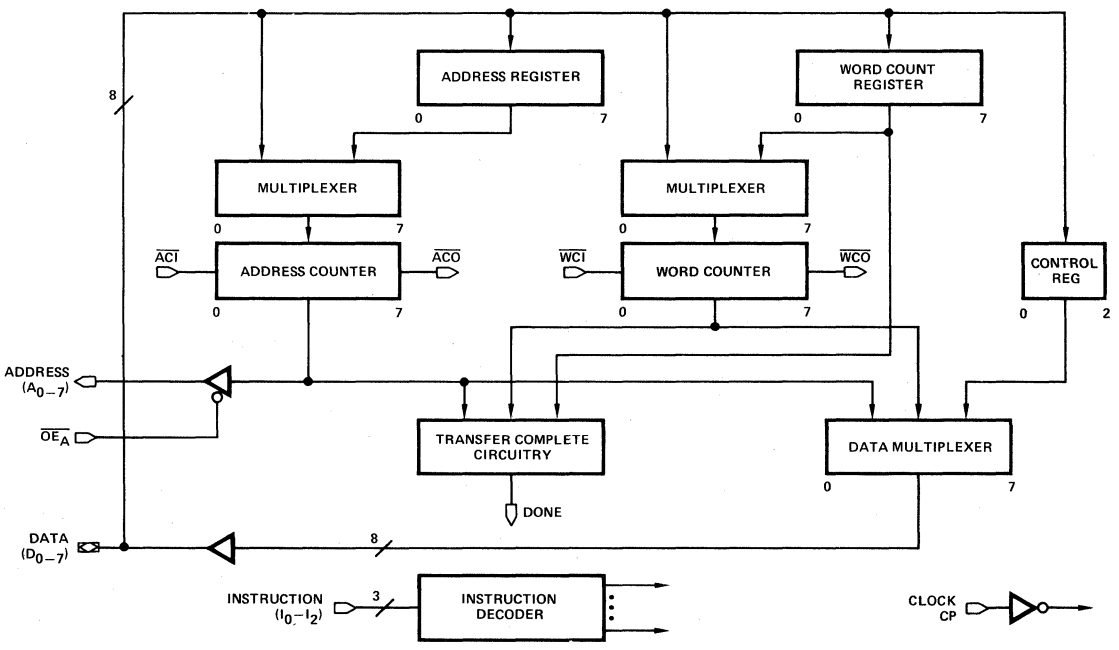
GENERAL DESCRIPTION

The Am2940, a 28-pin member of Advanced Micro Devices Am2900 family of Low-Power Schottky bipolar LSI chips, is a high-speed, cascadable, eight-bit wide Direct Memory Access Address Generator slice. Any number of Am2940's can be cascaded to form larger addresses.

The primary function of the device is to generate sequential memory addresses for use in the sequential transfer of data to or from a memory. It also maintains a data word count and generates a DONE signal when a programmable terminal count has been reached. The device is designed for use in peripheral controllers with DMA capability or in any other system which transfers data to or from sequential locations of a memory.

The Am2940 can be programmed to increment or decrement the memory address in any of four control modes, and executes eight different instructions. The initial address and word count are saved internally by the Am2940 so that they can be restored later in order to repeat the data transfer operation.

BLOCK DIAGRAM



Am2940 DMA Address Generator

MPR-226

Am2940 ARCHITECTURE

As shown in the Block Diagram, the Am2940 consists of the following:

- A three-bit Control Register.
- An eight-bit Address Counter with input multiplexer.
- An eight-bit Address Register.
- An eight-bit Word Counter with input multiplexer.
- An eight-bit Word Count Register.
- Transfer complete circuitry.
- An eight-bit wide data multiplexer with three-state output buffers.
- Three-state address output buffers with external output enable control.
- An instruction decoder.

Control Register

Under instruction control, the Control Register can be loaded or read from the bidirectional DATA lines D_0-D_7. Control Register bits 0 and 1 determine the Am2940 Control Mode, and bit 2 determines whether the Address Counter increments or decrements. Figure 1 defines the Control Register format.

Address Counter

The Address Counter, which provides the current memory address, is an eight-bit, binary, up/down counter with full look-ahead carry generation. The Address Carry input ($\overline{ACI}$) and Address Carry Output ($\overline{ACO}$) allow cascading to accommodate larger addresses. Under instruction control, the Address Counter can be enabled, disabled, and loaded from the DATA inputs, D_0-D_7, or the Address Register. When enabled and the $\overline{ACI}$ input is LOW, the Address Counter increments/decrements on the LOW to HIGH transition of the CLOCK input, CP. The Address Counter output can be enabled onto the three-state ADDRESS outputs A_0-A_7 under control of the Output Enable input, $\overline{OE}_A$.

Address Register

The eight-bit Address Register saves the initial address so that it can be restored later in order to repeat a transfer operation. When the LOAD ADDRESS instruction is executed, the Address Register and Address Counter are simultaneously loaded from the DATA inputs, D_0-D_7.

Word Counter and Word Count Register

The Word Counter and Word Count Register, which maintain and save a word count, are similar in structure and operation to the Address Counter and Address Register, with the exception that the Word Counter increments in Control Modes 1 and 3, decrements in Control Mode 0, and is disabled in Control Mode 2. The LOAD WORD COUNT instruction simultaneously loads the Word Counter and Word Count Register.

Transfer Complete Circuitry

The Transfer Complete Circuitry is a combinational logic network which detects the completion of the data transfer operation in three Control Modes and generates the DONE output signal. The DONE signal is a open-collector output, which can be dot-anded between chips.

Data Multiplexer

The Data Multiplexer is an eight-bit wide, 3-input multiplexer which allows the Address Counter, Word Counter, and Control Register to be read at the DATA lines, D_0-D_7. The Data Multiplexer and three-state Data output buffers are instruction controlled.

Address Output Buffers

The three-state Address Output Buffers allow the Address Counter output to be enabled onto the ADDRESS lines, A_0-A_7, under external control. When the Output Enable input, $\overline{OE}_A$, is LOW, the Address output buffers are enabled; when $\overline{OE}_A$ is HIGH, the ADDRESS lines are in the high-impedance state. The address and Data Output Buffers can sink 24mA output current over the commercial operating range.

Instruction Decoder

The Instruction Decoder generates required internal control signals as a function of the INSTRUCTION inputs, I_0-I_2 and Control Register bits 0 and 1.

Clock

The CLOCK input, CP is used to clock the Address Register, Address Counter, Word Count Register, Word Counter, and Control Register, all on the LOW to HIGH transition of the CP signal.

Control Register

CR_2	CR_1	CR_0

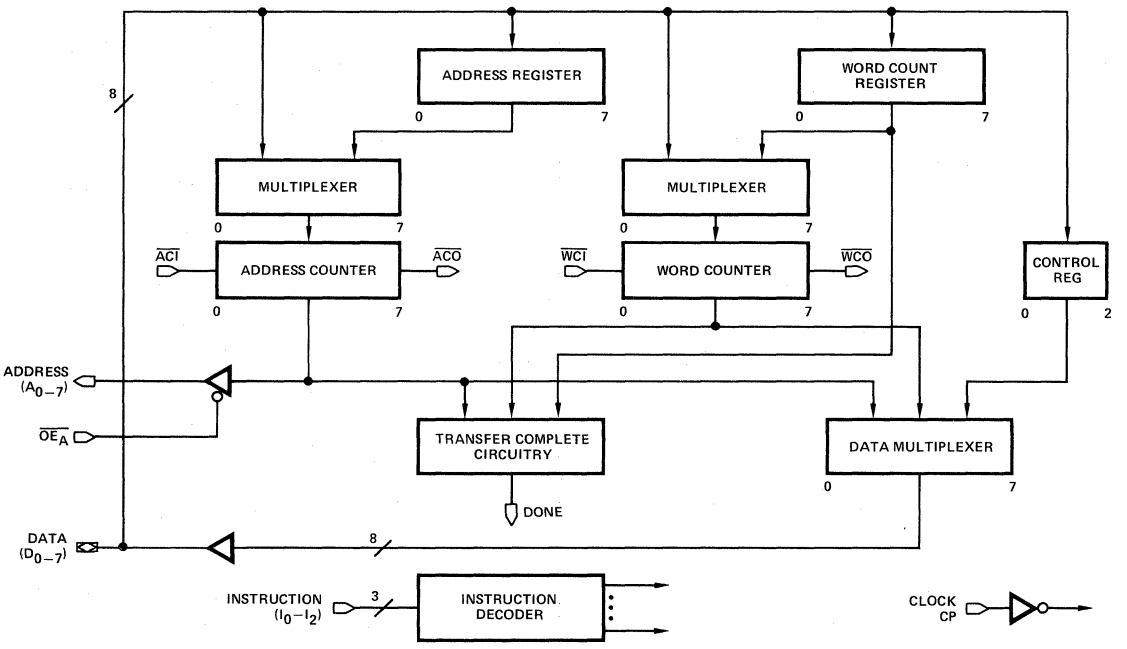
CR_1	CR_0	Control Mode Number	Control Mode Type	Word Counter	DONE Output Signal	
					$\overline{WCI}$ = LOW	$\overline{WCI}$ = HIGH
L	L	0	Word Count Equals Zero	Decrement	HIGH when Word Counter = 1	HIGH when Word Counter = 0
L	H	1	Word Count Compare	Increment	HIGH when Word Counter + 1 = Word Count Reg.	HIGH when Word Counter = Word Count Reg.
H	L	2	Address Compare	Hold	HIGH when Word Counter = Address Counter	
H	H	3	Word Counter Carry Out	Increment	Always LOW	

CR_2	Address Counter
L	Increment
H	Decrement

H = HIGH
L = LOW

Figure 1. Control Register Format Definition.

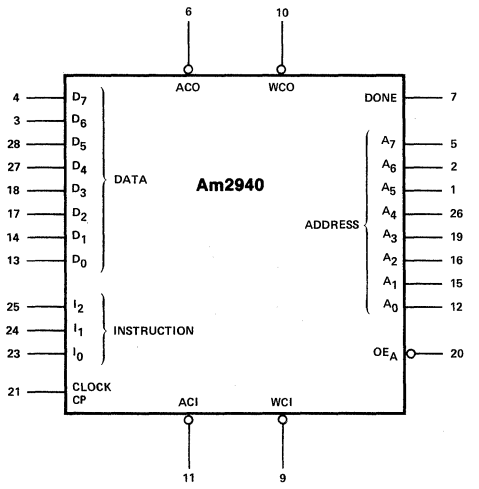
BLOCK DIAGRAM



Am2940 DMA Address Generator

MPR-226

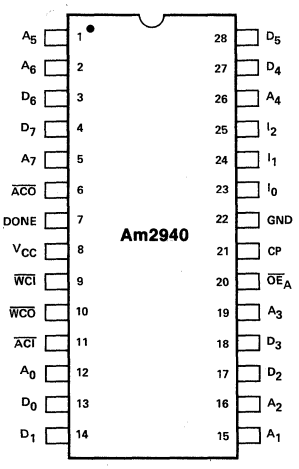
LOGIC SYMBOL



MPR-227

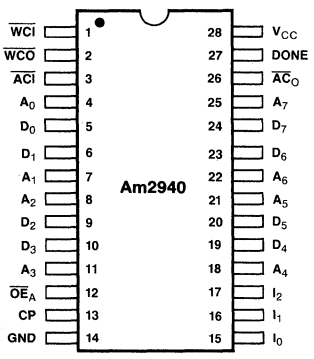
CONNECTION DIAGRAMS
Top Views

DIP



MPR-228

Flat Pack



MPR-586

Note: Pin 1 is marked for orientation

Am2940 CONTROL MODES

Control Mode 0 – Word Count Equals Zero Mode

In this mode, the LOAD WORD COUNT instruction loads the word count into the Word Count Register and Word Counter. When the Word Counter is enabled and the Word Counter Carry-in, $\overline{WCI}$, is LOW, the Word Counter increments on the LOW to HIGH transition of the clock input, CP. Figure 1 specifies when the DONE signal is generated.

Control Mode 1 – Word Count Compare Mode

In this mode the LOAD WORD COUNT instruction loads the word count into the Word Count Register and clears the Word Counter. When the Word Counter is enabled and the Word Counter Carry-in $\overline{WCI}$, is LOW, the Word Counter increments on the LOW to HIGH transition of the clock input, CP. Figure 1 specifies when the DONE signal is generated.

Control Mode 2 – Address Compare Mode

In this mode, only an initial and final memory address need be specified. The initial Memory Address is loaded into the Address Register and Address Counter and the final memory address is loaded into the Word Count Register and Word Counter. The Word Counter is always disabled in this mode and serves as a holding register for the final memory address. When the Address Counter is enabled and the $\overline{ACI}$ input is LOW, the Address Counter increments or decrements (depending on Control Register bit 2) on the LOW to HIGH transition of the CLOCK input, CP. The Transfer Complete Circuitry compares the Address Counter with the Word Counter and generates the DONE signal during the last word transfer, i.e. when the Address Counter equals the Word Counter.

Control Mode 3 – Word Counter Carry Out Mode

For this mode of operation, the user can load the Word Count Register and Word Counter with the two's complement of the number of data words to be transferred. When the Word Counter is enabled and the $\overline{WCI}$ input is LOW, the Word Counter increments on the LOW to HIGH transition of the CLOCK input, CP. A Word Counter Carry Out signal, $\overline{WCO}$, indicates the last data word is being transferred. The DONE signal is not required in this mode and, therefore, is always LOW.

Am2940 INSTRUCTIONS

The Am2940 instruction set consists of eight instructions. Six instructions load and read the Address Counter, Word Counter and Control Register, one instruction enables the Address and Word counters, and one instruction reinitializes the Address and Word Counters. The function of the REINITIALIZE COUNTERS, LOAD WORD COUNT, and ENABLE COUNTERS instructions varies with the Control Mode being utilized. Table 1 defines the Am2940 Instructions as a function of Instruction inputs I_0-I_2 and the four Am2940 Control Modes.

The WRITE CONTROL REGISTER instruction writes DATA input D_0-D_2 into the Control Register; DATA inputs D_3-D_7 are "don't care" inputs for this instruction. The READ CONTROL REGISTER instruction gates the Control Register outputs to DATA lines, D_0-D_2. DATA lines D_3-D_7 are in the HIGH state during this instruction.

The Word Counter can be read using the READ WORD COUNTER instruction, which gates the Word Counter outputs to DATA lines D_0-D_7. The LOAD WORD COUNT instruction is Control Mode dependent. In Control Modes 0, 2, and 3, DATA inputs D_0-D_7 are written into both the Word Count Register and Word Counter. In Control Mode 1, DATA inputs D_0-D_7 are written into the Word Count Register and the Word Counter is cleared.

The READ ADDRESS COUNTER instruction gates the Address Counter outputs to DATA lines D_0-D_7, and the LOAD ADDRESS instruction writes DATA inputs D_0-D_7 into both the Address Register and Address Counter.

In Control Modes 0, 1, and 3, the ENABLE COUNTERS instruction enables both the Address and Word Counters; in Control Mode 2, the Address Counter is enabled and the Word Counter holds its contents. When enabled and the carry input is active, the counters increment on the LOW to HIGH transition of the CLOCK input, CP. Thus, with this instruction applied, counting can be controlled by the carry inputs.

The REINITIALIZE COUNTERS instruction also is Control Mode dependent. In Control Modes 0, 2, and 3, the contents of the Address Register and Word Count Register are transferred to the respective Address Counter and Word Counter; in Control Mode 1, the content of the Address Register is transferred to the Address Counter and the Word Counter is cleared. The REINITIALIZE COUNTERS instruction allows a data transfer operation to be repeated without reloading the address and word count from the DATA lines.

6

TABLE I. Am2940 INSTRUCTIONS

I_2 I_1 I_0	Octal Code	Function	Mnemonic	Control Mode	Word Reg.	Word Counter	Address Reg.	Address Counter	Control Register	Data D_0-D_7
L L L	0	WRITE CONTROL REGISTER	WRCR	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	D_0-D_2→CR	INPUT
L L H	1	READ CONTROL REGISTER	RDCR	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	CR→D_0-D_2 (Note 1)
L H L	2	READ WORD COUNTER	RDWC	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	WC→D
L H H	3	READ ADDRESS COUNTER	RDAC	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	AC→D
H L L	4	REINITIALIZE COUNTERS	REIN	0, 2, 3	HOLD	WCR→WC	HOLD	AR→AC	HOLD	Z
				1	HOLD	ZERO→WC	HOLD	AR→AC	HOLD	Z
H L H	5	LOAD ADDRESS	LDAD	0, 1, 2, 3	HOLD	HOLD	D→AR	D→AC	HOLD	INPUT
H H L	6	LOAD WORD COUNT	LDWC	0, 2, 3	D→WR	D→WC	HOLD	HOLD	HOLD	INPUT
				1	D→WR	ZERO→WC	HOLD	HOLD	HOLD	INPUT
H H H	7	ENABLE COUNTERS	ENCT	0, 1, 3	HOLD	ENABLE COUNT	HOLD	ENABLE COUNT	HOLD	Z
				2	HOLD	HOLD	HOLD	ENABLE COUNT	HOLD	Z

CR = Control Reg.
AR = Address Reg.
AC = Address Counter
WCR = Word Count Reg.
WC = Word Counter
D = Data
L = LOW
H = HIGH
Z = High Impedance

Note 1:
Data Bits D_3-D_7 are high during this instruction.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5V to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to V_{CC} max.
DC Input Voltage	−0.5V to +5.5V
DC Output Current, Into Outputs	30mA
DC Input Current	−30mA to +5.0mA

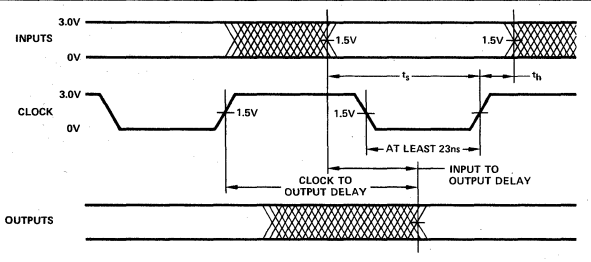
OPERATING RANGE

P/N	Range	Temperature	V_{CC}	
Am2940PC, DC	COM'L	T_A = 0°C to +70°C	V_{CC} = 5.0V ±5%	(MIN. = 4.75V MAX. = 5.25V)
Am2940DM, FM	MIL	T_C = −55° to +125°C	V_{CC} = 5.0V ±10%	(MIN. = 4.50V MAX. = 5.50V)

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)			Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN., V_{IN} = V_{IH} or V_{IL}		MIL I_{OH} = −1.0mA COM'L I_{OH} = −2.6mA	2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., V_{IN} = V_{IH} or V_{IL} (Note 5)	WCO, ACO	MIL I_{OL} = 8.0mA COM'L I_{OL} = 12mA			0.5	Volts
			A_{0-7}, D_{0-7} DONE	MIL I_{OL} = 16mA COM'L I_{OL} = 24mA				
V_{IH}	Input HIGH Level (Note 4)	Guaranteed Input Logical HIGH voltage for all inputs			2.0			Volts
V_{IL}	Input LOW Level (Note 4)	Guaranteed Input Logical LOW voltage for all inputs					0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA					−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.5V		D_{0-7} All Others			−0.15 −0.8	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V		D_{0-7} All Others			150 40	µA
I_{CEX}	Output Leakage on DONE	V_{CC} = MAX., V_O = 5.5V					250	µA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V					1.0	mA
I_{SC}	Output Short Circuit Current (Note 3)	V_{CC} = MAX. + 0.5V, V_O = 0.5V			−30		−85	mA
I_{OZL} I_{OZH}	Output OFF Current	V_{CC} = MAX. $\overline{OE}$ = 2.4V	V_{OUT} = 0.5V	A_{0-7} D_{0-7}			−50 −150	µA
			V_{OUT} = 2.4V	A_{0-7} D_{0-7}			50 150	
I_{CC}	Power Supply Current	V_{CC} = MAX.		T_A = 25°C		170	275	mA
			Am2940PC, DC	T_A = 0°C to +70°C			290	
				T_A = +70°C			235	
			Am2940DM, FM	T_C = −55°C to +125°C			315	
				T_C = +125°C			225	

Notes:
1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. These input levels provide no guaranteed noise immunity and should only be tested in a static-, noise-free environment.
5. I_{OL} limit on A_i and D_i (i = 0 to 7) applies to either output individually, but not both at the same time. The sum of the loading on A_i plus D_i is limited to 24mA MIL or 32mA COM'L.



See Tables A for t_s and t_h for various inputs. See Tables B for combinational delays from clock and other inputs to outputs.

Figure 2. Switching Waveforms.

MPR-728

SWITCHING CHARACTERISTICS

The tables below define the Am2940 switching characteristics. Tables A are set-up and hold times relative to the clock LOW-to-HIGH transition. Tables B are combinational delays. Tables C are clock requirements. All measurements are made at 1.5V with input levels at 0V or 3V. All values are in ns with C_L = 50pF except output disable times ($\overline{OE}$ to A and I to D) which are specified for a 5pF load. All times are in ns.

I. TYPICAL ROOM TEMPERATURE CHARACTERISTICS (T_A = 25°C, V_{CC} = 5.0V, C_L = 50pF)

A. Set-up and Hold Times (Relative to clock LOW-to-HIGH transition)

Input	t_s	t_h
D_{0-7}	13	3
I_{012}	33	2
$\overline{ACI}$	15	2
$\overline{WCI}$ (Note 1)	15	1

B. Combinational Delays

Input	$\overline{ACO}$	$\overline{WCO}$	A_{0-7}	DONE	D_{0-7}
$\overline{ACI}$	12	–	–	–	–
$\overline{WCI}$ (Note 2)	–	12	–	27	–
I_{0-2}	–	–	–	–	21
CP (Note 3)	35	35	35	50	–

C. Clock Requirements

Minimum Clock LOW Time	20	ns
Minimum Clock HIGH Time	25	ns
Maximum Clock Frequency	22	MHz

D. Enable/Disable Times

From	To	Disable	Enable	
I_{012}	D_{0-7}	25	19	ns
$\overline{OE}$	A_{0-7}	19	13	ns

II. GUARANTEED ROOM TEMPERATURE CHARACTERISTICS (T_A = 25°C, V_{CC} = 5.0V, C_L = 50pF)

A. Set-up and Hold Times (Relative to clock LOW-to-HIGH transition)

Input	t_s	t_h
D_{0-7}	21	4
I_{012}	41	3
$\overline{ACI}$	27	3
$\overline{WCI}$ (Note 1)	27	3

B. Combinational Delays

Input	$\overline{ACO}$	$\overline{WCO}$	A_{0-7}	DONE	D_{0-7}
$\overline{ACI}$	18	–	–	–	–
$\overline{WCI}$ (Note 2)	–	18	–	41	–
I_{0-2}	–	–	–	–	34
CP (Note 3)	50	50	48	77	–

C. Clock Requirements

Minimum Clock LOW Time	23	ns
Minimum Clock HIGH Time	30	ns
Maximum Clock Frequency	18	MHz

D. Enable/Disable Times

From	To	Disable	Enable	
I_{012}	D_{0-7}	30	30	ns
$\overline{OE}$	A_{0-7}	23	23	ns

III. GUARANTEED CHARACTERISTICS OVER COMMERCIAL OPERATING RANGE

Am2940PC, DC (T_A = 0°C to +70°C, V_{CC} = 4.75V to 5.25V, C_L = 50pF)

A. Set-up and Hold Times (Relative to clock LOW-to-HIGH transition)

Input	t_s	t_h
D_{0-7}	24	5
I_{012}	46	4
$\overline{ACI}$	30	4
$\overline{WCI}$ (Note 1)	30	3

B. Combinational Delays

Input	$\overline{ACO}$	$\overline{WCO}$	A_{0-7}	DONE	D_{0-7}
$\overline{ACI}$	20	–	–	–	–
$\overline{WCI}$ (Note 2)	–	20	–	46	–
I_{0-2}	–	–	–	–	37
CP (Note 3)	58	58	54	85	–

C. Clock Requirements

Minimum Clock LOW Time	23	ns
Minimum Clock HIGH Time	34	ns
Maximum Clock Frequency	17	MHz

D. Enable/Disable Times

From	To	Disable	Enable	
I_{012}	D_{0-7}	35	35	ns
$\overline{OE}$	A_{0-7}	25	25	ns

IV. GUARANTEED CHARACTERISTICS OVER MILITARY OPERATING RANGE

Am2940DM, FM (T_C = −55°C to +125°C, V_{CC} = 4.5V to 5.5V, C_L = 50pF)

A. Set-up and Hold Times (Relative to clock LOW-to-HIGH transition)

Input	t_s	t_h
D_{0-7}	27	6
I_{012}	49	5
$\overline{ACI}$	34	5
$\overline{WCI}$ (Note 1)	34	5

B. Combinational Delays

Input	$\overline{ACO}$	$\overline{WCO}$	A_{0-7}	DONE	D_{0-7}
$\overline{ACI}$	21	–	–	–	–
$\overline{WCI}$ (Note 2)	–	21	–	54	–
I_0-I_2	–	–	–	–	41
CP (Note 3)	64	64	62	88	–

C. Clock Requirements

Minimum Clock LOW Time	23	ns
Minimum Clock HIGH Time	35	ns
Maximum Clock Frequency	16	MHz

D. Enable/Disable Times

From	To	Disable	Enable	
I_{012}	D_{0-7}	42	42	ns
$\overline{OE}$	A_{0-7}	30	30	ns

Notes: 1. Control modes 0, 1, and 3 only.
2. $\overline{WCI}$ to Done occurs only in control modes 0 and 1.
3. CP to Done occurs only in control modes 0, 1, and 2.

APPLICATIONS

The Am2940 is designed for use in peripheral controllers with DMA capability or in any other system which transfers data to or from sequential locations of a memory. One or more Am2940's can be used in each peripheral controller of a distributed DMA system to provide the memory address and word count required for DMA operation.

Figure 3 shows a block diagram of an example microprogrammed DMA peripheral controller. The Am2910 Microprogram Sequencer, Microprogram Memory, and the Microinstruction Register form the microprogram control portion of this peripheral controller. The Am2940 generates the memory address and maintains the word count required for DMA operation. An internal three-state bus provides the communication path between the Microinstruction Register, the Am2917 Data Transceivers, the Am2940, the Am2901A Microprocessor, and the Device Interface Circuitry.

The Am2940 interconnections are shown in detail in Figure 4. Two Am2940's are cascaded to generate a sixteen-bit address. The Am2940 ADDRESS and DATA output current sink capability is 24mA over the commercial operating range. This allows the Am2940's to drive the System Address Bus and Internal Three-State Bus directly, thereby eliminating the need for separate bus drivers. Three-bits in the Microinstruction Register provide the Am2940 Instruction Inputs, I_0-I_2. The microprogram clock is used to clock the Am2940's and, when the ENABLE COUNTERS instruction is applied, address and word counting is controlled by the CNT bit of the Microinstruction Register.

Asynchronous interface control circuitry generates System Bus control signals and enables the Am2940 Address onto the System Address Bus at the appropriate time. The open-collector DONE outputs are dot-anded and used as a test input to the Am2910 Microprogram Sequencer.

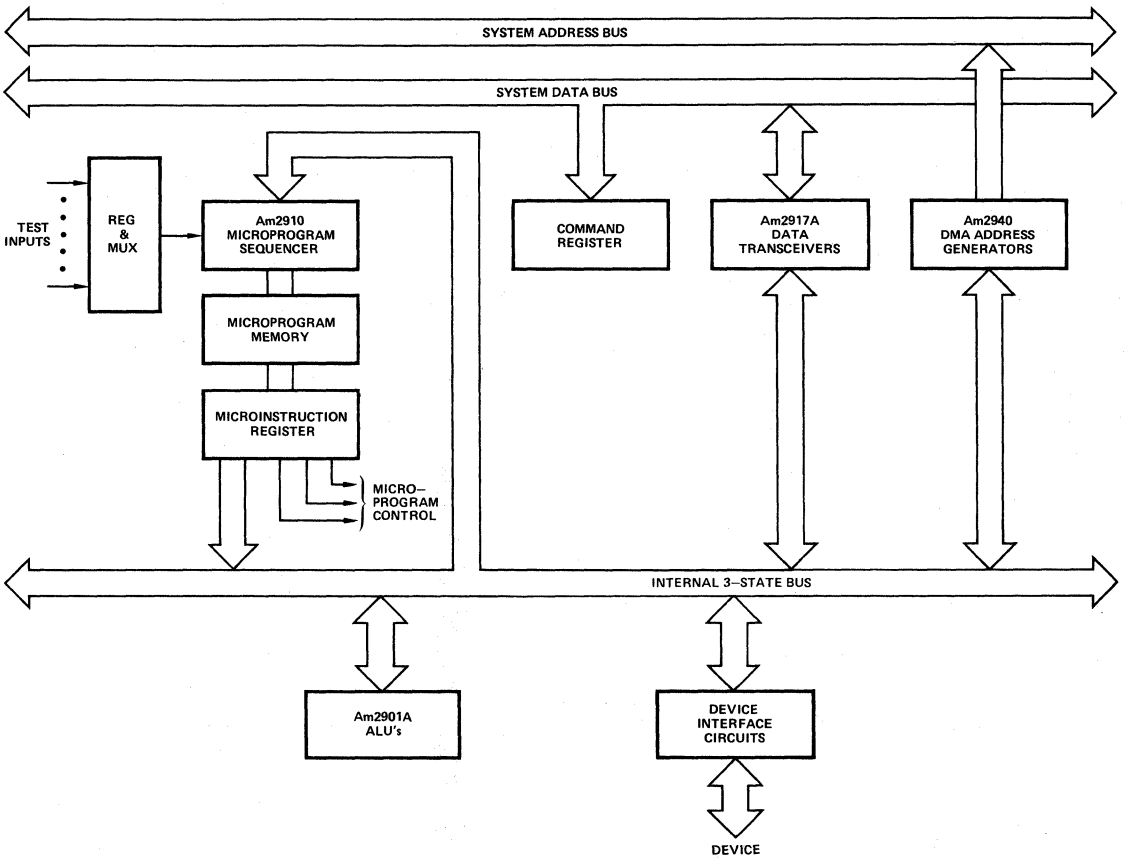


Figure 3. DMA Peripheral Controller Block Diagram.

MPR-229

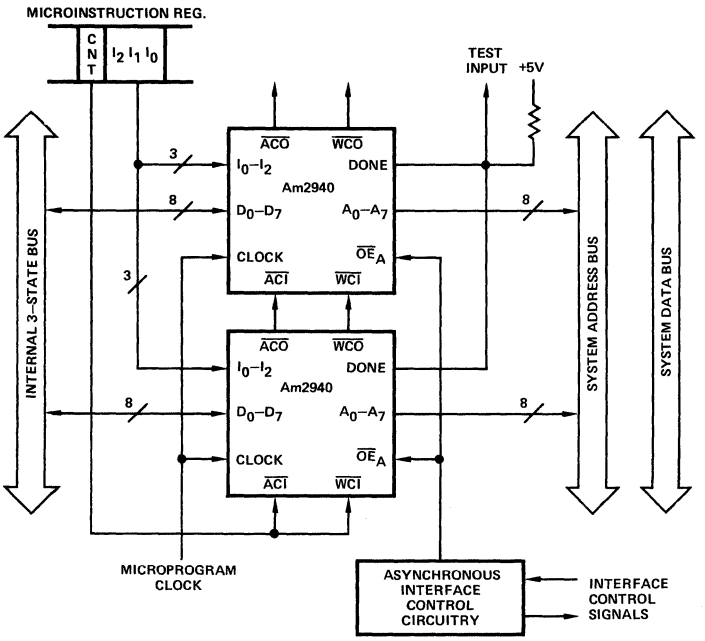


Figure 4. Am2940 Interconnections.

MPR-230

6

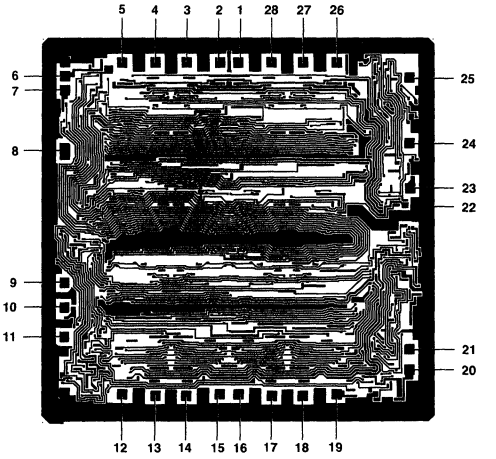
ORDERING INFORMATION

Order the part number according to the table below to obtain the desired package, temperature range, and screening level.

Order Number	Package Type (Note 1)	Operating Range (Note 2)	Screening Level (Note 3)
AM2940PC	P-28	C	C-1
AM2940DC	D-28	C	C-1
AM2940DC-B	D-28	C	B-2 (Note 4)
AM2940DM	D-28	M	C-3
AM2940DM-B	D-28	M	B-3
AM2940FM	F-28-2	M	C-3
AM2940FM-B	F-28-2	M	B-3
AM2940XC	Dice	C	Visual inspection to MIL-STD-883 Method 2010B.
AM2940XM	Dice	M	

Notes:
1. P = Molded DIP, D = Hermetic DIP, F = Flat Pak. Number following letter is number of leads. See Appendix B for detailed outline. Where Appendix B contains several dash numbers, any of the variations of the package may be used unless otherwise specified.
2. C = 0°C to +70°C, V_{CC} = 4.75V to 5.25V.
   M = −55°C to +125°C, V_{CC} = 4.50V to 5.50V.
3. See Appendix A for details of screening. Levels C-1 and C-3 conform to MIL-STD-883, Class C. Level B-3 conforms to MIL-STD-883, Class B.
4. 96 hour burn-in.

METALLIZATION AND PAD LAYOUT



Note: Numbers refer to DIP pin connections.

DIE SIZE: 0.178" X 0.181"

Am2942

Programmable Timer/Counter DMA Address Generator

DISTINCTIVE CHARACTERISTICS

- 22-pin version of Am2940 –
  Provides multiplexed Address and Data lines plus additional Instruction Input and Instruction Enable pins.
- Can be used as either DMA Address Generator or Programmable Timer Counter.
- Executes 16 instructions –
  Eight DMA instructions plus eight Timer/Counter instructions
- Provides two independent programmable 8-bit up/down counters in a 22-pin package –
  Counters can be cascaded to form single-chip 16-bit up/down counter.
- Reinitialize capability –
  Counters can be reinitialized from on-chip registers.
- Expandable eight-bit slice –
  Any number of Am2942s can be cascaded. Three devices provide a 48 bit counter.
- Programmable control modes –
  Provide four types of control.
- High speed bipolar LSI –
  Advanced Low-Power Schottky TTL technology provides typical count frequency of 25MHz and 24mA output current sink capability.

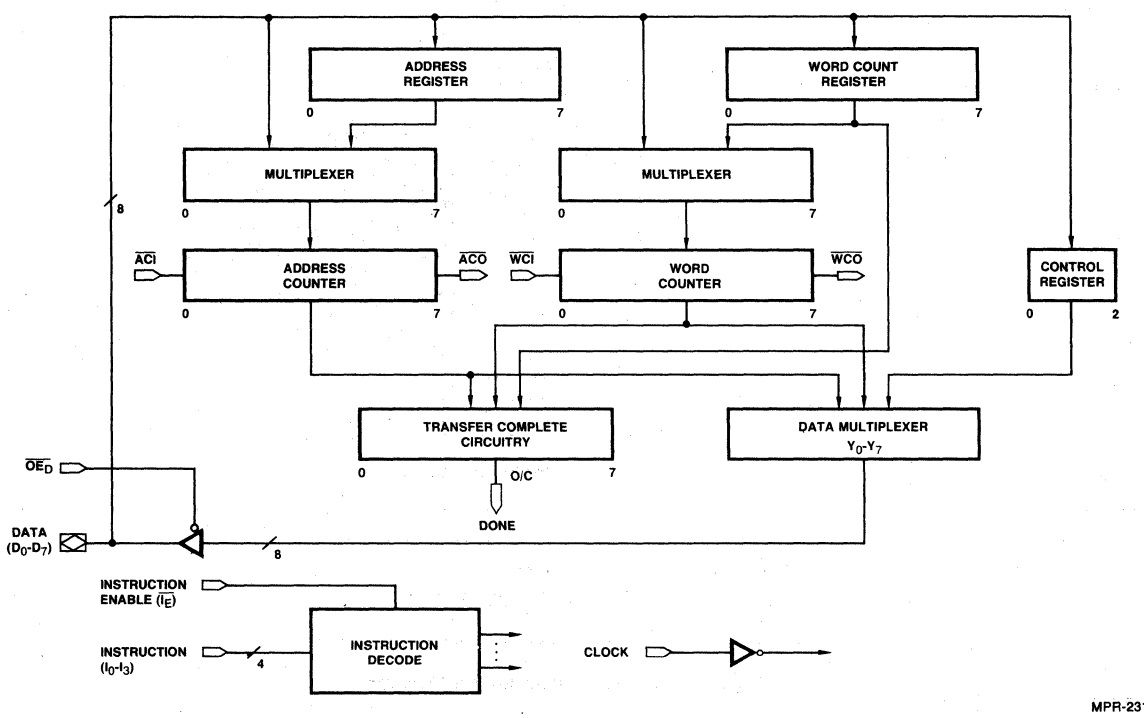
GENERAL DESCRIPTION

The Am2942, a 22-pin version of the Am2940, can be used as a high-speed DMA Address Generator or Programmable Timer/Counter. It provides multiplexed Address and Data lines, for use with a common bus, and additional Instruction Input and Instruction Enable pins. The Am2942 executes 16 instructions; eight are the same as the Am2940 instructions, and eight instructions facilitate the use of the Am2942 as a Programmable Timer/Counter. The Instruction Enable input allows the sharing of the Am2942 instruction field with other devices.

When used as a Timer/Counter, the Am2942 provides two independent, programmable, eight-bit, up-down counters in a 22-pin package. The two on-chip counters can be cascaded to form a single chip, 16-bit counter. Also, any number of chips can be cascaded – for example three cascaded Am2942s form a 48-bit timer/counter.

Reinitialization instructions provide the capability to reinitialize the counters from on-chip registers. Am2942 Programmable Control Modes, identical to those of the Am2940, offer four different types of programmable control.

BLOCK DIAGRAM



Am2942 ARCHITECTURE

As shown in the Block Diagram, the Am2942 consists of the following:

- A three-bit Control Register.
- An eight-bit Address Counter with input multiplexer.
- An eight-bit Address Register.
- An eight-bit Word Counter with input multiplexer.
- An eight-bit Word Count Register.
- Transfer complete circuitry.
- An eight-bit wide data multiplexer with three-state output buffers.
- An instruction decoder.

Control Register

Under instruction control, the Control Register can be loaded or read from the bidirectional DATA lines D_0-D_7. Control Register bits 0 and 1 determine the Am2942 Control Mode, and bit 2 determines whether the Address Counter increments or decrements. Figure 1 defines the Control Register format.

Address Counter

The Address Counter, which provides the current memory address, is an eight-bit, binary, up/down counter with full look-ahead carry generation. The Address Carry input ($\overline{ACI}$) and Address Carry Output ($\overline{ACO}$) allow cascading to accommodate larger addresses. Under instruction control, the Address Counter can be enabled, disabled, and loaded from the DATA inputs, D_0-D_7, or the Address Register. When enabled and the $\overline{ACI}$ input is LOW, the Address Counter increments/decrements on the LOW to HIGH transition of the CLOCK input, CP.

Address Register

The eight-bit Address Register saves the initial address so that it can be restored later in order to repeat a transfer operation. When the LOAD ADDRESS instruction is executed, the Address Register and Address Counter are simultaneously loaded from the DATA inputs, D_0-D_7.

Word Counter and Word Count Register

The Word Counter and Word Count Register, which maintain and save a word count, are similar in structure and operation to the Address Counter and Address Register, with the exception that the Word Counter increments in Control Modes 1 and 3, and decrements in Control Modes 0 and 2. The LOAD WORD COUNT instruction simultaneously loads the Word Counter and Word Count Register.

Transfer Complete Circuitry

The Transfer Complete Circuitry is a combinational logic network which detects the completion of the data transfer operation in three Control Modes and generates the DONE output signal. The DONE signal is a open-collector output, which can be dot-anded between chips.

Data Multiplexer

The Data Multiplexer is an eight-bit wide, three-input multiplexer which allows the Address Counter, Word Counter and Control Register to be read at DATA lines D_0-D_7. The Data Multiplexer output, Y_0-Y_7, is enabled onto DATA lines D_{0-7} if and only if the Output Enable input, $\overline{OE_D}$, is LOW. (Refer to Figure 2.)

Instruction Decoder

The Instruction Decoder generates required internal control signals as a function of the INSTRUCTION inputs, I_0-I_3 Control Register bits 0 and 1, and the INSTRUCTION ENABLE input, $\overline{I_E}$.

Clock

The CLOCK input, CP is used to clock the Address Register, Address Counter, Word Count Register, Word Counter, and Control Register, all on the LOW to HIGH transition of the CP signal.

Control Register

CR_2	CR_1	CR_0

CR_1	CR_0	Control Mode Number	Control Mode Type	Word Counter	DONE Output Signal	
					$\overline{WCI}$ = LOW	$\overline{WCI}$ = HIGH
L	L	0	Word Count Equals Zero	Decrement	HIGH when Word Counter = 1	HIGH when Word Counter = 0
L	H	1	Word Count Compare	Increment	HIGH when Word Counter + 1 = Word Count Reg.	HIGH when Word Counter = Word Count Reg.
H	L	2	Address Compare	Decrement	HIGH when Word Counter = Address Counter	
H	H	3	Word Counter Carry Out	Increment	Always LOW	

CR_2	Address Counter
L	Increment
H	Decrement

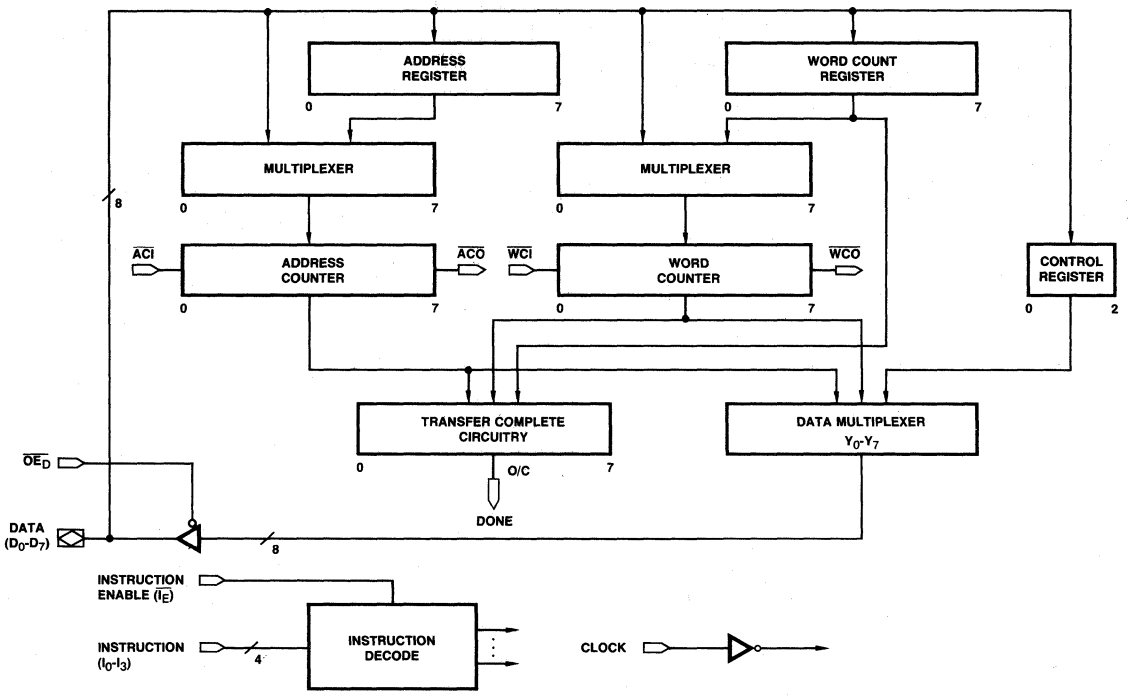
H = HIGH
L = LOW

Figure 1. Control Register Format Definition.

$\overline{OE_D}$	D_0-D_7
L	DATA MULTIPLEXER OUTPUT, Y_0-Y_7
H	HIGH Z

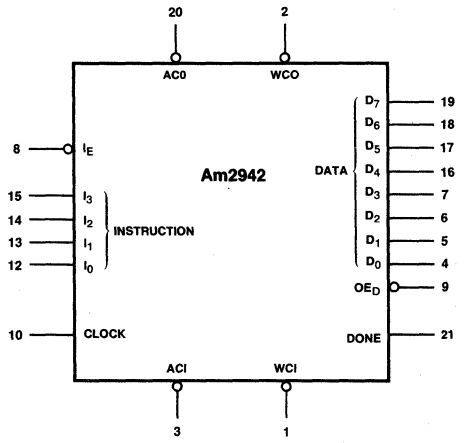
Figure 2. Data Bus Output Enable Function.

BLOCK DIAGRAM



MPR-231

LOGIC SYMBOL

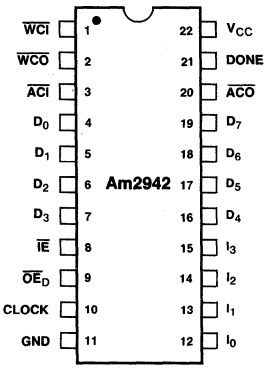


V_{CC} = Pin 22
GND = Pin 11

MPR-232

CONNECTION DIAGRAM
Top View

DIP



Note: Pin 1 is marked for orientation.

MPR-233

Am2942 CONTROL MODES

Control Mode 0 – Word Count Equals Zero Mode

In this mode, the LOAD WORD COUNT instruction loads the word count into the Word Count Register and Word Counter. When the Word Counter is enabled and the Word Counter Carry-in, $\overline{WCI}$, is LOW, the Word Counter decrements on the LOW to HIGH transition of the CLOCK input, CP. Figure 1 specifies when the DONE signal is generated in this mode.

Control Mode 1 – Word Count Compare Mode

In this mode the LOAD WORD COUNT instruction loads the word count into the Word Count Register and clears the Word Counter. When the Word Counter is enabled and the Word Counter Carry-in, $\overline{WCI}$, is LOW, the Word Counter increments on the LOW to HIGH transition of the clock input, CP. Figure 1 specifies when the DONE signal is generated.

Control Mode 2 – Address Compare Mode

In this mode, only an initial and final memory address need to be specified. The initial Memory Address is loaded into the Address Register and Address Counter and the final memory address is loaded into the Word Count Register and Word Counter. The Word Counter is always disabled in this mode and serves as a holding register for the final memory address. When the Address Counter is enabled and the $\overline{ACI}$ input is LOW, the Address Counter increments or decrements (depending on Control Register bit 2) on the LOW to HIGH transition of the CLOCK input, CP. The Transfer Complete Circuitry compares the Address Counter with the Word Counter and generates the DONE signal during the last word transfer, i.e. when the Address Counter equals the Word Counter.

Control Mode 3 – Word Counter Carry Out Mode

For this mode of operation, the user can load the Word Count Register and Word Counter with the two's complement of the number of data words to be transferred. When the Word Counter is enabled and the $\overline{WCI}$ input is LOW, the Word Counter increments on the LOW to HIGH transition of the CLOCK input, CP. A Word Counter Carry Out signal, $\overline{WCO}$, indicates the last data word is being transferred. The DONE signal is not required in this mode and, therefore, is always LOW.

Am2942 INSTRUCTIONS

The Am2942 instruction set consists of sixteen instructions. Eight are DMA Instructions and are similar to the Am2940 instructions. The remaining eight instructions are designed to facilitate the use of the Am2942 as a Programmable Timer/Counter. Figures 3 and 4 define the Am2942 Instructions.

Instructions 0-7 are DMA instructions. The WRITE CONTROL REGISTER instruction writes DATA input D_0-D_2 into the Control Register; DATA inputs D_3-D_7 are "don't care" inputs for this instruction. The READ CONTROL REGISTER instruction gates the Control Register to Data Multiplexer outputs Y_0-Y_2. Outputs Y_3-Y_7 are HIGH during this instruction.

The Word Counter can be read using the READ WORD COUNTER instruction, which gates the Word Counter to Data Multiplexer outputs, Y_0-Y_7. The LOAD WORD COUNT instruction is Control Mode dependent. In Control Modes 0, 2, and 3, DATA inputs D_0-D_7 are written into both the Word Count Register and Word Counter. In Control Mode 1, DATA inputs D_0-D_7 are written into the Word Count Register and the Word Counter is cleared.

The READ ADDRESS COUNTER instruction gates the Address Counter to Data Multiplexer outputs, Y_0-Y_7, and the LOAD ADDRESS instruction writes DATA inputs D_0-D_7 into both the Address Register and Address Counter.

In Control Modes 0, 1, and 3, the ENABLE COUNTERS instruction enables both the Address and Word Counters; in Control Mode 2, the Address Counter is enabled and the Word Counter holds its contents. When enabled and the carry input is active, the counters increment on the LOW to HIGH transition of the CLOCK input, CP. Thus, with this instruction applied, counting can be controlled by the carry inputs.

The REINITIALIZE COUNTERS instruction also is Control Mode dependent. In Control Modes 0, 2, and 3, the contents of the Address Register and Word Count Register are transferred to the respective Address Counter and Word Counter; in Control Mode 1, the content of the Address Register is transferred to the Address Counter and the Word Counter is cleared. The REINITIALIZE COUNTERS instruction allows a data transfer operation to be repeated without reloading the address and word count from the DATA lines.

When $\overline{I_E}$ is HIGH, Instruction inputs, I_0-I_2, are disabled. If I_3 is LOW, the function performed is identical to that of the ENABLE COUNTERS instruction. Thus, counting can be controlled by the carry inputs with the ENABLE COUNTERS instruction applied or with Instruction Inputs I_0-I_2 disabled.

Instructions 8-F facilitate the use of the Am2942 as a Programmable Timer/Counter. They differ from instructions 0-7 in that they provide independent control of the Address Counter, Word Counter and Control Register.

The WRITE CONTROL REGISTER, T/C instruction writes DATA input D_0-D_2 into the Control Register. DATA inputs D_3-D_7 are "don't care" inputs for this instruction. The Address and Word Counters are enabled, and the Control Register contents appear at the Data Multiplexer output.

The REINITIALIZE ADDRESS COUNTER instruction allows the independent reinitialization of the Address Counter. The Word Counter is enabled and the contents of the Address Counter appear at the Data Multiplexer output.

$\overline{I_E}$	I_3	I_2	I_1	I_0	HEX CODE		
0	0	0	0	0	0	WRITE CONTROL REGISTER	DMA INSTRUCTIONS
0	0	0	0	1	1	READ CONTROL REGISTER	
0	0	0	1	0	2	READ WORD COUNTER	
0	0	0	1	1	3	READ ADDRESS COUNTER	
0	0	1	0	0	4	REINITIALIZE COUNTERS	
0	0	1	0	1	5	LOAD ADDRESS	
0	0	1	1	0	6	LOAD WORD COUNT	
0	0	1	1	1	7	ENABLE COUNTERS	
1	0	X	X	X	0-7	INSTRUCTION DISABLE	
0	1	0	0	0	8	WRITE CONTROL REGISTER, T/C	TIMER/COUNTER INSTRUCTIONS
0	1	0	0	1	9	REINITIALIZE ADDRESS COUNTER	
0	1	0	1	0	A	READ WORD COUNTER, T/C	
0	1	0	1	1	B	READ ADDRESS COUNTER, T/C	
0	1	1	0	0	C	REINITIALIZE ADDRESS & WORD COUNTERS	
0	1	1	0	1	D	LOAD ADDRESS, T/C	
0	1	1	1	0	E	LOAD WORD COUNT, T/C	
0	1	1	1	1	F	REINITIALIZE WORD COUNTER	
1	1	X	X	X	8-F	INSTRUCTION DISABLE, T/C	

0 = LOW 1 = HIGH X = DON'T CARE

Notes: 1. When I_3 is tied LOW, the Am2942 acts as a DMA circuit: When I_3 is tied HIGH, the Am2942 acts as a Timer/Counter circuit.
2. Am2942 instructions 0 through 7 are the same as Am2940 instructions.

Figure 3. Am2942 Instructions.

The Word Counter can be read, using the READ WORD COUNTER, T/C instruction. Both counters are enabled when this instruction is executed.

When the READ ADDRESS COUNTER, T/C instruction is executed, both counters are enabled and the address counter contents appear at the Data Multiplexer output.

The REINITIALIZE ADDRESS and WORD COUNTERS instruction provides the capability to reinitialize both counters at the same time. The Address Counter contents appear at the Data Multiplexer output.

DATA inputs D_0-D_7 are loaded into both the Address Register and Counter when the LOAD ADDRESS, T/C instruction is executed. The Word Counter is enabled and its contents appear at the Data Multiplexer output.

The LOAD WORD COUNT, T/C instruction is identical to the LOAD WORD COUNT instruction with the exception that Address Counter is enabled.

The Word Counter can be independently reinitialized using the REINITIALIZE WORD COUNTER instruction. The Address Counter is enabled and the Word Counter contents appear at the Data Multiplexer output.

When the $\overline{I_E}$ input is HIGH, Instruction inputs, I_0-I_2, are disabled. The function performed when I_3 is HIGH is identical to that performed when I_3 is LOW, with the exception that the Word Counter contents appear at the Data Multiplexer output.

$\overline{I_E}$	$I_3I_2I_1I_0$ (Hex)	Function	Mnemonic	Control Mode	Word Reg.	Word Counter	Adr. Reg.	Adr. Counter	Control Reg.	Data Multiplexer Output
L	0	WRITE CONTROL REGISTER	WRCR	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	$D_{0-2} \rightarrow$ CR	FORCED HIGH
L	1	READ CONTROL REGISTER	RDCR	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	CONTROL REG.
L	2	READ WORD COUNTER	RDWC	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	WORD COUNTER
L	3	READ ADDRESS COUNTER	RDAC	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	ADR. COUNTER
L	4	REINITIALIZE COUNTERS	REIN	0, 2, 3	HOLD	WR → WC	HOLD	AR → AC	HOLD	ADR. CNTR.
				1	HOLD	ZERO → WC	HOLD	AR → AC	HOLD	ADR. CNTR.
L	5	LOAD ADDRESS	LDAD	0, 1, 2, 3	HOLD	HOLD	D → AR	D → AC	HOLD	WORD COUNTER
L	6	LOAD WORD COUNT	LDWC	0, 2, 3	D → WR	D → WC	HOLD	HOLD	HOLD	FORCED HIGH
				1	D → WR	ZERO → WC	HOLD	HOLD	HOLD	FORCED HIGH
L	7	ENABLE COUNTERS	ENCT	0, 1, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	ADR. CNTR.
				2	HOLD	HOLD	HOLD	ENABLE	HOLD	ADR. CNTR.
H	0-7	INSTRUCTION DISABLE	–	0, 1, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	ADR. CNTR.
				2	HOLD	HOLD	HOLD	ENABLE	HOLD	ADR. CNTR.
L	8	WRITE CONTROL REGISTER, T/C	WCRT	0, 1, 2, 3	HOLD	ENABLE	HOLD	ENABLE	$D_{0-2} \rightarrow$ CR	CONTROL REG.
L	9	REINITIALIZE ADR. COUNTER	REAC	0, 1, 2, 3	HOLD	ENABLE	HOLD	AR → AC	HOLD	ADR. COUNTER
L	A	READ WORD COUNTER, TC	RWCT	0, 1, 2, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	WORD COUNTER
L	B	READ ADDRESS COUNTER, T/C	RACT	0, 1, 2, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	ADR. COUNTER
L	C	REINITIALIZE ADDRESS AND WORD COUNTERS	RAWC	0, 2, 3	HOLD	WR → WC	HOLD	AR → AC	HOLD	ADR. CNTR.
				1	HOLD	ZERO → WC	HOLD	AR → AC	HOLD	ADR. CNTR.
L	D	LOAD ADDRESS, T/C	LDAT	0, 1, 2, 3	HOLD	ENABLE	D → AR	D → AC	HOLD	WORD COUNTER
L	E	LOAD WORD COUNT, T/C	LWCT	0, 2, 3	D → WR	D → WC	HOLD	ENABLE	HOLD	FORCED HIGH
				1	D → WR	ZERO → WC	HOLD	ENABLE	HOLD	FORCED HIGH
L	F	REINITIALIZE WORD COUNTER	REWC	0, 2, 3	HOLD	WR → WC	HOLD	ENABLE	HOLD	WD. CNTR.
				1	HOLD	ZERO → WC	HOLD	ENABLE	HOLD	WD. CNTR.
H	8-F	INSTRUCTION DISABLE, T/C	–	0, 1, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	WD. CNTR
				2	HOLD	HOLD	HOLD	ENABLE	HOLD	WD. CNTR.

WR = WORD REGISTER
WC = WORD COUNTER
AR = ADDRESS REGISTER
AC = ADDRESS COUNTER
CR = CONTROL REGISTER
D = DATA

Figure 4. Am2942 Function Table.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5V to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to V_{CC} max.
DC Input Voltage	−0.5V to +5.5V
DC Output Current, Into Outputs	30mA
DC Input Current	−30mA to +5.0mA

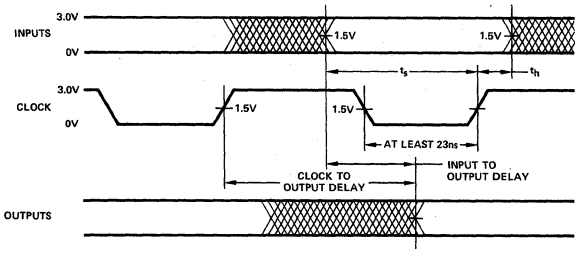
OPERATING RANGE

P/N	Range	Temperature	V_{CC}	
Am2942PC, DC	COM'L	T_A = 0°C to +70°C	V_{CC} = 5.0V ±5%	(MIN. = 4.75V MAX. = 5.25V)
Am2942DM, FM	MIL	T_C = −55° to +125°C	V_{CC} = 5.0V ±10%	(MIN. = 4.50V MAX. = 5.50V)

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)			Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN., V_{IN} = V_{IH} or V_{IL}		MIL I_{OH} = −1.0mA COM'L I_{OH} = −2.6mA	2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., V_{IN} = V_{IH} or V_{IL}	WCO, ACO	MIL I_{OL} = 8.0mA COM'L I_{OL} = 12mA			0.5	Volts
			D_{0-7}, DONE	MIL I_{OL} = 16mA COM'L I_{OL} = 24mA				
V_{IH}	Input HIGH Level (Note 4)	Guaranteed Input Logical HIGH voltage for all inputs			2.0			Volts
V_{IL}	Input LOW Level (Note 4)	Guaranteed Input Logical LOW voltage for all inputs					0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA					−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.5V		D_{0-7} All Others			−0.15 −0.8	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V		D_{0-7} All Others			150 40	µA
I_{CEX}	Output Leakage on DONE	V_{CC} = MAX., V_O = 5.5V					250	µA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V					1.0	mA
I_{SC}	Output Short Circuit Current (Note 3)	V_{CC} = MAX. + 0.5V, V_O = 0.5V			−30		−85	mA
I_{OZL} I_{OZH}	Output OFF Current	V_{CC} = MAX. $\overline{OE}$ = 2.4V	V_{OUT} = 0.5V V_{OUT} = 2.4V	D_{0-7} D_{0-7}			−150 150	µA
I_{CC}	Power Supply Current	V_{CC} = MAX.	Am2942PC, DC	T_A = 25°C T_A = 0°C to +70°C T_A = +70°C		155	250 265 220	mA
			Am2942DM, FM	T_C = −55°C to +125°C T_C = +125°C			285 205	

Notes:
1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. These input levels provide no guaranteed noise immunity and should only be tested in a static-, noise-free environment.



See Tables A for t_s and t_h for various inputs. See Tables B for combinational delays from clock and other inputs to outputs.

Figure 5. Switching Waveforms.

MPR-728

SWITCHING CHARACTERISTICS

The tables below define the Am2942 switching characteristics. Tables A are set-up and hold times relative to the clock LOW-to-HIGH transition. Tables B are combinational delays. Tables C are clock requirements. All measurements are made at 1.5V with input levels at 0V or 3V. All values are in ns with C_L = 50pF except output disable times (I to D) which are specified for a 5pF load. All times are in ns.

I. TYPICAL ROOM TEMPERATURE CHARACTERISTICS (T_A = 25°C, V_{CC} = 5.0V, C_L = 50pF)

A. Set-up and Hold Times (Relative to clock LOW-to-HIGH transition)

Input	t_s	t_h
D_{0-7}	13	3
I_{012}	33	2
$\overline{ACI}$	15	2
$\overline{WCI}$	15	1

B. Combinational Delays

Input	$\overline{ACO}$	$\overline{WCO}$	DONE	D_{0-7}
$\overline{ACI}$	12	–	–	–
$\overline{WCI}$ (Note 1)	–	12	27	–
I_{0-2}	–	–	–	21
CP (Note 2)	35	35	50	37

C. Clock Requirements

Minimum Clock LOW Time	20	ns
Minimum Clock HIGH Time	30	ns
Maximum Clock Frequency	28	MHz

D. Enable/Disable Times

From	To	Disable	Enable	
$\overline{OE}$	D_{0-7}	19	13	ns

II. GUARANTEED ROOM TEMPERATURE CHARACTERISTICS (T_A = 25°C, V_{CC} = 5.0V, C_L = 50pF)

A. Set-up and Hold Times (Relative to clock LOW-to-HIGH transition)

Input	t_s	t_h
D_{0-7}	21	4
I_{012}	41	3
$\overline{ACI}$	27	3
$\overline{WCI}$	27	3

B. Combinational Delays

Input	$\overline{ACO}$	$\overline{WCO}$	DONE	D_{0-7}
$\overline{ACI}$	18	–	–	–
$\overline{WCI}$ (Note 1)	–	18	41	–
I_{0-2}	–	–	–	34
CP (Note 2)	50	50	77	53

C. Clock Requirements

Minimum Clock LOW Time	23	ns
Minimum Clock HIGH Time	30	ns
Maximum Clock Frequency	22	MHz

D. Enable/Disable Times

From	To	Disable	Enable	
$\overline{OE}$	D_{0-7}	23	23	ns

III. GUARANTEED CHARACTERISTICS OVER COMMERCIAL OPERATING RANGE

Am2942PC, DC (T_A = 0°C to +70°C, V_{CC} = 4.75V to 5.25V, C_L = 50pF)

A. Set-up and Hold Times (Relative to clock LOW-to-HIGH transition)

Input	t_s	t_h
D_{0-7}	24	6
I_{012}	46	5
$\overline{ACI}$	30	4
$\overline{WCI}$	30	3

B. Combinational Delays

Input	$\overline{ACO}$	$\overline{WCO}$	DONE	D_{0-7}
$\overline{ACI}$	20	–	–	–
$\overline{WCI}$ (Note 1)	–	20	46	–
I_{0-2}	–	–	–	37
CP (Note 2)	58	58	85	59

C. Clock Requirements

Minimum Clock LOW Time	23	ns
Minimum Clock HIGH Time	34	ns
Maximum Clock Frequency	20	MHz

D. Enable/Disable Times

From	To	Disable	Enable	
$\overline{OE}$	D_{0-7}	25	25	ns

IV. GUARANTEED CHARACTERISTICS OVER MILITARY OPERATING RANGE

Am2942DM, FM (T_C = −55°C to +125°C, V_{CC} = 4.5V to 5.5V, C_L = 50pF)

A. Set-up and Hold Times (Relative to clock LOW-to-HIGH transition)

Input	t_s	t_h
D_{0-7}	27	7
I_{012}	49	5
$\overline{ACI}$	34	5
$\overline{WCI}$	34	5

B. Combinational Delays

Input	$\overline{ACO}$	$\overline{WCO}$	DONE	D_{0-7}
$\overline{ACI}$	21	–	–	–
$\overline{WCI}$ (Note 1)	–	21	54	–
I_{0-2}	–	–	–	41
CP (Note 2)	64	64	88	68

C. Clock Requirements

Minimum Clock LOW Time	23	ns
Minimum Clock HIGH Time	35	ns
Maximum Clock Frequency	15	MHz

D. Enable/Disable Times

From	To	Disable	Enable	
$\overline{OE}$	D_{0-7}	30	30	ns

Notes: 1. $\overline{WCI}$ to Done occurs only in control modes 0 and 1.
2. CP to Done occurs only in control modes 0, 1, and 2.

APPLICATIONS

Figure 6 shows an Am2942 used as two independent, programmable eight-bit timer/counters. In this example, an Am2910 Microprogram Sequencer provides an address to Am27S27 512 x 8 Registered PROMs. The on-chip PROM output register is used as the Microinstruction Register.

The Am2942 Instruction input, I_3, is tied HIGH to select the eight Timer/Counter instructions. The $\overline{I_E}$, I_0-I_2, and $\overline{OE_D}$ inputs are provided by the microinstruction, and the D_0-D_7 data lines are connected to a common Data Bus. GATE WC and GATE AC are separate enable controls for the respective Word Counter and Address Counter. The DONE, $\overline{ACO}$ and $\overline{WCO}$ output signals indicate that a pre-programmed time or count has been reached.

Figure 7 shows an Am2942 used as a single 16-bit programmable timer/counter. In this example, the Word Counter carry-out, $\overline{WCO}$, is connected to the Address Counter carry-in, $\overline{ACI}$, to form a single 16-bit counter which is enabled by the GATE signal.

Figure 8 shows two Am2942s cascaded to form a 32-bit programmable timer/counter. The two Word Counters form the low order 16 bits, and the two Address Counters form the high order bits. This allows the timer/counter to be loaded and read 16 bits at a time.

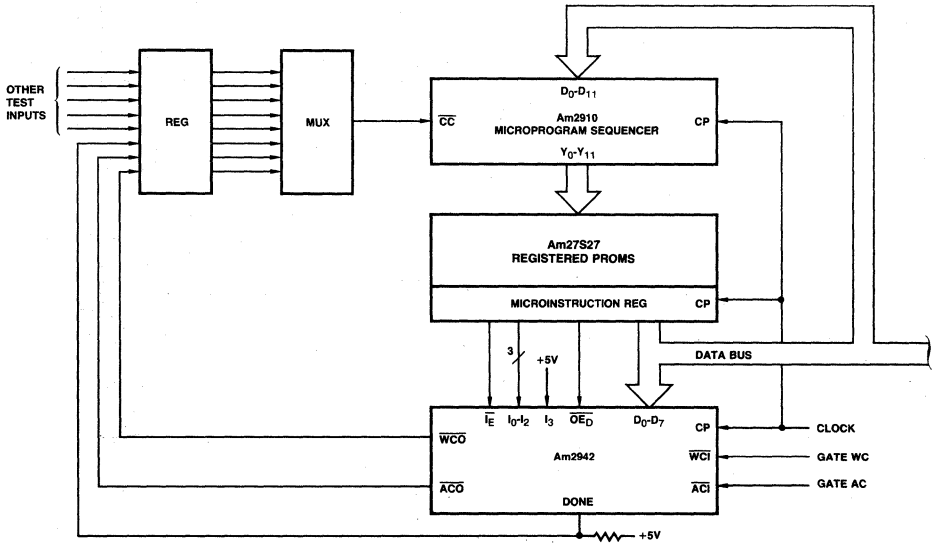


Figure 6. Two 8-Bit Programmable Counters/Timers in a 22-Pin Package.

MPR-234

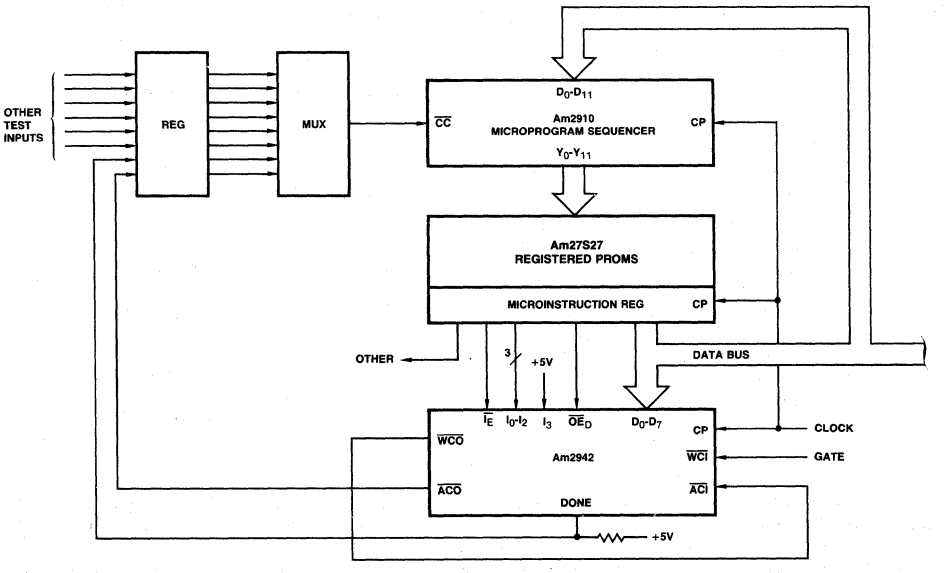


Figure 7. 16-Bit Programmable Counter/Timer Using a Single Am2942.

MPR-235

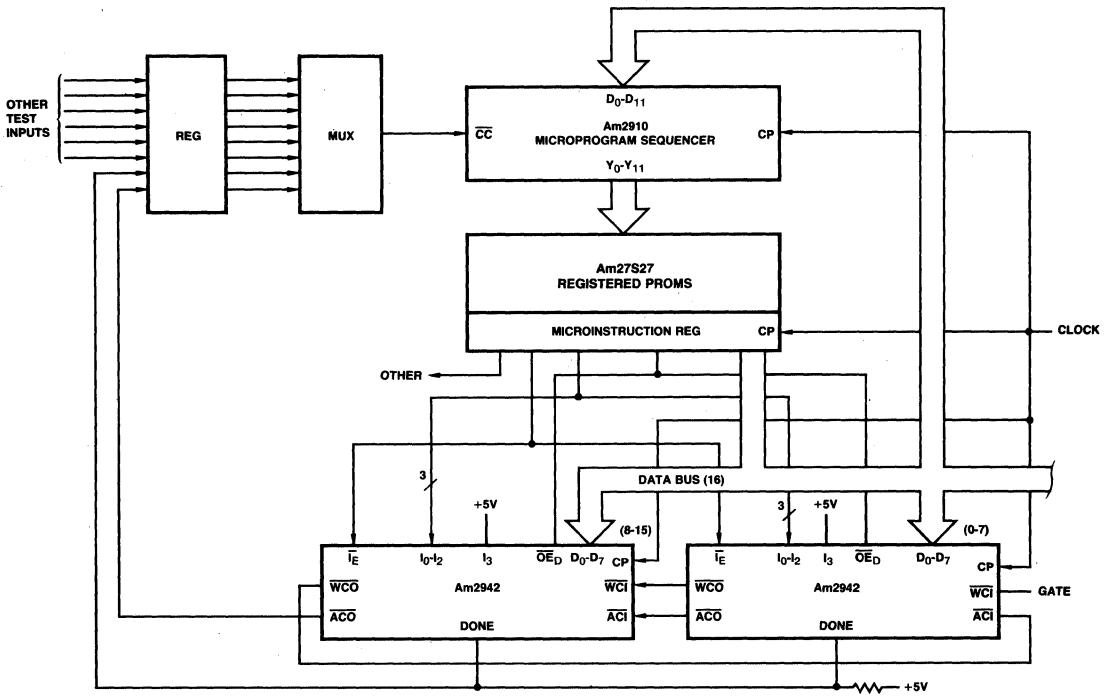


Figure 8. 32-Bit Programmable Counter/Timer Using Two Am2942s.

MPR-236

ORDERING INFORMATION

Order the part number according to the table below to obtain the desired package, temperature range, and screening level.

Order Number	Package Type (Note 1)	Operating Range (Note 2)	Screening Level (Note 3)
AM2942PC	P-22	C	C-1
AM2942DC	D-22	C	C-1
AM2942DC-B	D-22	C	B-2 (Note 4)
AM2942DM	D-22	M	C-3
AM2942DM-B	D-22	M	B-3
AM2942FM	F-22	M	C-3
AM2942FM-B	F-22	M	B-3
AM2942XC	Dice	C	Visual inspection to MIL-STD-883 Method 2010B.
AM2942XM	Dice	M	

Notes: 1. P = Molded DIP, D = Hermetic DIP, F = Flat Pak. Number following letter is number of leads. See Appendix B for detailed outline. Where Appendix B contains several dash numbers, any of the variations of the package may be used unless otherwise specified.
2. C = 0°C to +70°C, V_{CC} = 4.75V to 5.25V, M = − 55°C to +125°C, V_{CC} = 4.50V to 5.50V.
3. See Appendix A for details of screening. Levels C-1 and C-3 conform to MIL-STD-883, Class C. Level B-3 conforms to MIL-STD-883, Class B.
4. 96 hour burn-in.

Am2950 • Am2951

Eight-Bit Bidirectional I/O Ports

DISTINCTIVE CHARACTERISTICS

- Eight-Bit, Bidirectional I/O Port with Handshake – Two eight-bit, back-to-back registers store data moving in both directions between two bidirectional busses.
- Register Full/Empty Flags – On-chip flag flip-flops provide data transfer handshaking signals.
- Separate Clock, Clock Enable and Three-State Output Enable for Each Register.
- Separate, Edge-Sensitive Clear Control for Each Flag Flip-Flop.
- Inverting and Non-Inverting Versions – The Am2950 provides non-inverting data outputs. The Am2951 provides inverting data outputs.
- 24mA Output Current Sink Capability.
- 100% Reliability Assurance Testing in Compliance with MIL-STD-883.

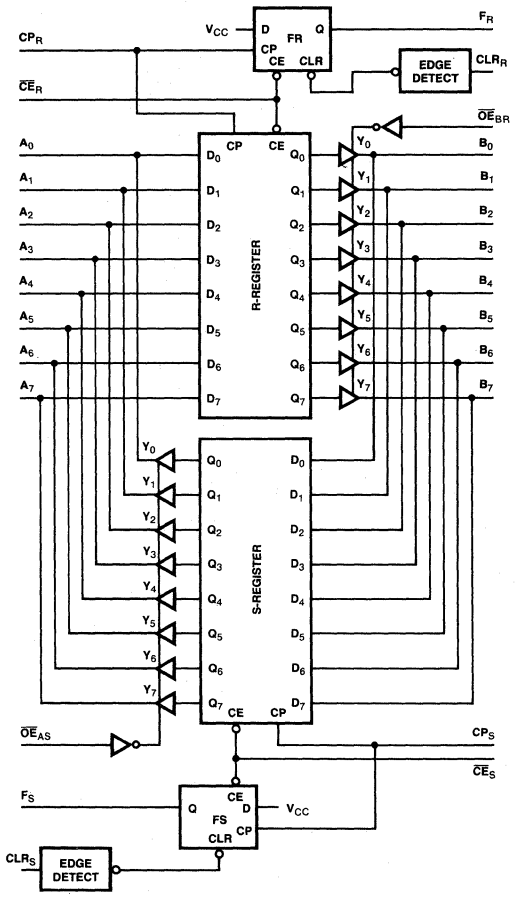
GENERAL DESCRIPTION

The Am2950 and Am2951, members of Advanced Micro Devices Am2900 Family, are designed for use as parallel data I/O ports. Two eight-bit, back to back registers store data moving in both directions between two bidirectional, 3-state busses. On chip flag flip-flops, set automatically when a register is loaded, provide the handshaking signals required for demand-response data transfer.

Considerable flexibility is designed into the Am2950 • Am2951. Separate clock, clock enable and three-state output enable signals are provided for each register, and edge-sensitive clear inputs are provided for each flag flip-flop. A number of these circuits can be used for wider I/O ports. Both inverting and non-inverting versions are available.

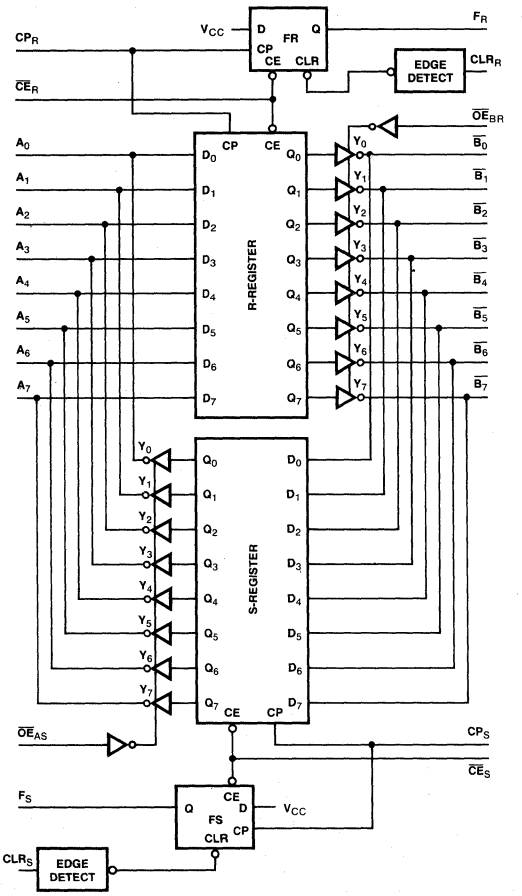
Twenty-four mA output current sink capability, sufficient for most three-state busses, is provided by the Am2950 • Am2951.

Am2950 BLOCK DIAGRAM



MPR-573

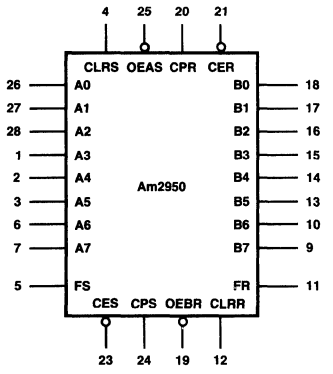
Am2951 BLOCK DIAGRAM



MPR-574

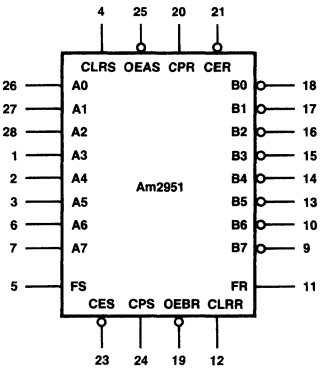
LOGIC SYMBOLS

Am2950



MPR-575

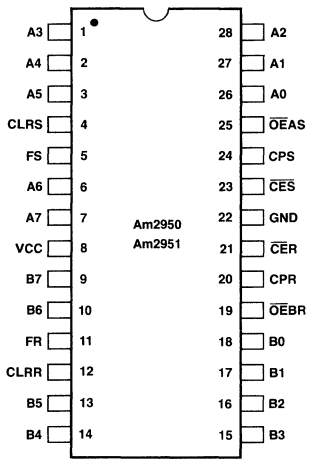
Am2951



MPR-576

CONNECTION DIAGRAMS
Top Views

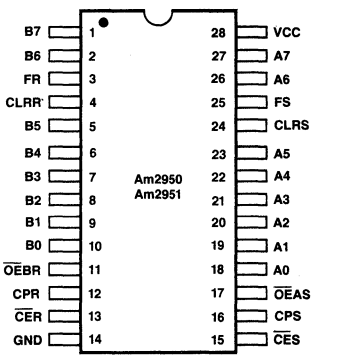
DIP



Note: Pin 1 is marked for orientation.
Bi is inverted on Am2951.

MPR-577

FLAT PACK



MPR-585

DEFINITION OF FUNCTIONAL TERMS

A0-7 Eight bidirectional lines carrying the R Register inputs or S Register outputs.

B0-7 Eight bidirectional lines carrying the S Register inputs or R Register outputs.

CPR The clock for the R Register and FR Flip-Flop. When $\overline{CER}$ is LOW, data is entered into the R Register and the FR Flip-Flop is set on the LOW to HIGH transition of the CPR signal.

$\overline{CER}$ The Clock Enable for the R Register and FR Flip-Flop. When $\overline{CER}$ is LOW, data is entered into the R Register and the FR Flip-Flop is set on the LOW to HIGH transition of the CPR signal. When $\overline{CER}$ is HIGH, The R Register and FR Flip-Flop hold their contents, regardless of CPR signal transitions.

$\overline{OEBR}$ The Output Enable for the R Register. When $\overline{OEBR}$ is LOW, The R Register three-state outputs are enabled onto the B0-7 lines. When $\overline{OEBR}$ is HIGH, the R Register outputs are in the high-impedance state.

FR The FR Flip-Flop output.

CLRR The clear control for the FR Flip-Flop. The FR Flip-Flop is cleared on the LOW to HIGH transition of CLRR signal.

CPS The clock for the S Register and FS Flip-Flop. When $\overline{CES}$ is LOW, data is entered into the S Register and the FS Flip-Flop is set on the LOW to HIGH transtion of the CPS signal.

$\overline{CES}$ The clock enable for the S Register and FS Flip-Flop. When $\overline{CES}$ is LOW, data is entered into the S Register and the FS Flip-Flop is set on the LOW to HIGH transition of the CPS signal. When $\overline{CES}$ is HIGH, the S Register and FS Flip-Flop hold their contents, regardless of CPS signal transitions.

$\overline{OEAS}$ The output enable for the S Register. When $\overline{OEAS}$ is LOW, the S Register three-state outputs are enabled onto the A0-7 lines. When OEAS is HIGH, the S Register outputs are in the high-impedance state.

FS The FS Flip-Flop output.

CLRS The clear control for the FS Flip-Flop. The FS Flip-Flop is cleared on the LOW to HIGH transition of CLRS signal.

REGISTER FUNCTION TABLE
(Applies to R or S Register)

Inputs			Internal	
D	CP	$\overline{CE}$	Q	Function
X	X	H	NC	Hold Data
L	↑	L	L	Load Data
H	↑	L	H	

OUTPUT CONTROL

	Internal	Y-Outputs		
$\overline{OE}$	Q	Am2950	Am2951	Function
H	X	Z	Z	Disable Outputs
L	L	L	H	Enable Outputs
L	H	H	L	

FLAG FLIP-FLOP FUNCTION TABLE
(Applies to R or S Flag Flip-Flop)

Inputs				
$\overline{CE}$	CP	CLR	F-Output	Function
H	X	↓	NC	Hold Flag
X	X	↑	L	Clear Flag
L	↑	↓	H	Set Flag

H = HIGH
L = LOW
X = Don't Care
Z = High Impedance
NC = NO CHANGE
↑ = LOW-to-HIGH Transition
↓ = NO LOW-to-HIGH Transition

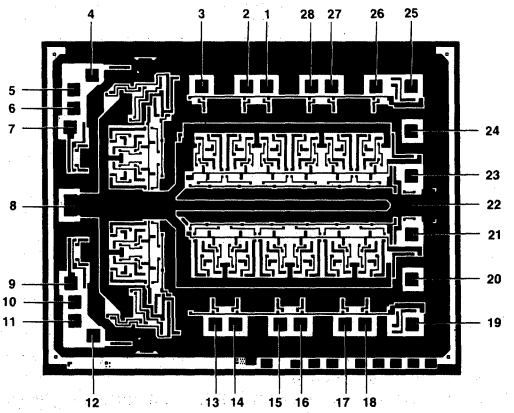
ORDERING INFORMATION

Order the part number according to the table below to obtain the desired package, temperature range, and screening level.

Am2950 Order Number	Am2951 Order Number	Package Type (Note 1)	Operating Range (Note 2)	Screening Level (Note 3)
AM2950PC	AM2951PC	P-28	C	C-1
AM2950DC	AM2951DC	D-28	C	C-1
AM2950DC-B	AM2951DC-B	D-28	C	B-2 (Note 4)
AM2950DM	AM2951DM	D-28	M	C-3
AM2950DM-B	AM2951DM-B	D-28	M	B-3
AM2950FM	AM2951FM	F-28-2	M	C-3
AM2950FM-B	AM2951FM-B	F-28-2	M	B-3
AM2950XC	AM2951XC	Dice	C	Visual inspection to MIL-STD-883 Method 2010B.
AM2950XM	AM2951XM	Dice	M	

Notes:
1. P = Molded DIP, D = Hermetic DIP, F = Flat Pak. Number following letter is number of leads. See Appendix B for detailed outline. Where Appendix B contains several dash numbers, any of the variations of the package may be used unless otherwise specified.
2. C = 0°C to +70°C, V_{CC} = 4.75V to 5.25V, M = − 55°C to +125°C, V_{CC} = 4.50V to 5.50V.
3. See Appendix A for details of screening. Levels C-1 and C-3 conform to MIL-STD-883, Class C. Level B-3 conforms to MIL-STD-883, Class B.
4. 96 hour burn-in.

METALLIZATION AND PAD LAYOUT



Numbers refer to DIP pin connection
DIE SIZE 0.107" X 0.138"

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to +VCC max.
DC Input Voltage	−0.5V to +5.5V
DC Output Current, Into Outputs	30mA
DC Input Current	−30mA to +5.0mA

OPERATING RANGE

Part Number	Range	Temperature	VCC
Am2950/51PC, DC	COM'L	T_A = 0°C to +70°C	VCC = 5.0V ±5% (MIN. = 4.75V, MAX. =5.25V)
Am2950/51DM, FM	MIL	TC = −55°C to +125°C	VCC = 5.0V ±10% (MIN. = 4.50V, MAX. = 5.50V)

Am2950, Am2951
DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)			Min.	Typ. (Note 2)	Max.	Units
VOH	Output HIGH Voltage	VCC = MIN. VIN = VIH or VIL	FR, FS	IOH = −1mA	2.4	3.4		Volts
			A0-7, B0-7	MIL, IOH = −2mA	2.4	3.4		
				COM'L, IOH = −6.5mA	2.4	3.4		
VOL	Output LOW Voltage (Note 5)	VCC = MIN. VIN = VIH or VIL	FR, FS	IOL = 12mA			0.5	Volts
			A0-7, B0-7	MIL IOL = 16mA			0.5	
				COM'L IOL = 24mA			0.5	
VIH	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs			2.0			Volts
VIL	Input LOW Level	Guaranteed input logical LOW voltage for all inputs					0.8	Volts
VI	Input Clamp Voltage	VCC = MIN., IIN = −18mA					−1.5	Volts
IIL	Input LOW Current	VCC = MAX., VIN = 0.5V		A0-7, B0-7			−250	μA
				CLRR, CLRS			−2.0	mA
				Others			−360	μA
IIH	Input HIGH Current	VCC = MAX., VIN = 2.7V		A0-7, B0-7			70	μA
				CLRR, CLRS			100	
				Others			20	
II	Input HIGH Current	VCC = MAX., VIN = 5.5V					1.0	mA
IO	Output Off-state Leakage Current	VCC = MAX.	A0-7, B0-7	V0 = 2.4V			70	μA
				V0 = 0.4V			−250	
ISC	Output Short Circuit Current (Note 3)	VCC =MAX.			−30		−85	mA
ICC	Power Supply Current (Notes 4, 6)	VCC = MAX.		T_A = 25°C		156	263	mA
			COM'L	T_A = 0°C to +70°C			275	
				T_A = +70°C			228	
			MIL	T_C = −55°C to +125°C			309	
				T_C = +125°C			202	

Notes:
1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at VCC = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. ICC is measured with all inputs at 4.5V and all outputs open.
5. The sum of IOL into Ai and Bi for each i must not exceed 32mA COM'L, 24mA MIL at a given point in time.
6. Worst case ICC is at minimum temperature.

SWITCHING CHARACTERISTICS

The tables below define the Am 2950 • Am2951 switching characteristics. Tables A are set-up and hold times relative to a clock LOW-to-HIGH transition. Tables B are propagation delays. Tables C are recovery times. Tables D are pulse-width requirements. Tables E are enable/disable times. All measurements are made at 1.5V with input levels at 0V or 3V. All values are in ns with RL on Ai and Bi = 220Ω and RL on FS and FR = 300Ω. CL = 50pF except output disable times which are specified at CL = 5pF.

TYPICAL ROOM TEMPERATURE CHARACTERISTICS (T_A = 25°C, VCC = 5.0V)

A. Set-up and Hold Times (With respect to clock LOW-to-HIGH transition.)

Input	With Respect To	ts	th
A0-7	CPS ↓	2	0
B0-7	CPR ↓	2	0
$\overline{CES}$	CPS ↓	9	0
$\overline{CER}$	CPR ↓	9	0

B. Propagation Delays

Input	A0-7	B0-7	FS	FR
CPS ↓	15	–	13	–
CPR ↓	–	15	–	13
CLRS ↓	–	–	11	–
CLRR ↓	–	–	–	11

C. Recovery Times

From	To	tREC
CLRS ↓	CPS ↓	17
CLRR ↓	CPR ↓	17

D. Pulse-Width Requirements

Input	Min. LOW Pulse Width	Min. HIGH Pulse Width
CPS	15	15
CPR	15	15
CLRS	15	15
CLRR	15	15

E. Enable/Disable Times

From	To	Disable	Enable
$\overline{OE}$AS	A0-7	15	16
$\overline{OE}$BR	B0-7	15	16

GUARANTEED ROOM TEMPERATURE CHARACTERISTICS (T_A = 25°C, VCC = 5.0V)

A. Set-up and Hold Times.

Input	With Respect To	ts	th
A0-7	CPS ↓		
B0-7	CPR ↓		
$\overline{CES}$	CPS ↓		
$\overline{CER}$	CPR ↓		

B. Propagation Delays

Input	A0-7	B0-7	FS	FR
CPS ↓		–		–
CPR ↓	–		–	
CLRS ↓	–	–		–
CLRR ↓	–	–	–	

C. Recovery Times

From	To	tREC
CLRS ↓	CPS ↓	
CLRR ↓	CPR ↓	

D. Pulse-Width Requirements

Input	Min. LOW Pulse Width	Min. HIGH Pulse Width
CPS		
CPR		
CLRS		
CLRR		

E. Enable/Disable Times

From	To	Disable	Enable
$\overline{OE}$AS	A0-7		
$\overline{OE}$BR	B0-7		

GUARANTEED CHARACTERISTICS OVER COMMERCIAL OPERATING RANGE
(T_A = 0°C to +70°C, VCC = 4.75 to 5.25V)

PRELIMINARY DATA

A. Set-up and Hold Times.

Input	With Respect To	ts	th
A0-7	CPS ↓	7	5
B0-7	CPR ↓	7	5
$\overline{CES}$	CPS ↓	15	4
$\overline{CER}$	CPR ↓	15	4

B. Propagation Delays

Input	A0-7	B0-7	FS	FR
CPS ↓	26	–	20	–
CPR ↓	–	26	–	20
CLRS ↓	–	–	22	–
CLRR ↓	–	–	–	22

C. Recovery Times

From	To	tREC
CLRS ↓	CPS ↓	31
CLRR ↓	CPR ↓	31

D. Pulse-Width Requirements

Input	Min. LOW Pulse Width	Min. HIGH Pulse Width
CPS	20	20
CPR	20	20
CLRS	20	20
CLRR	20	20

E. Enable/Disable Times

From	To	Disable	Enable
$\overline{OE}$AS	A0-7	22	27
$\overline{OE}$BR	B0-7	22	27

6

GUARANTEED CHARACTERISTICS OVER MILITARY OPERATING RANGE
(T_C = −55°C to +125°C, VCC = 4.5 to 5.5V)

PRELIMINARY DATA

A. Set-up and Hold Times.

Input	With Respect To	ts	th
A0-7	CPS ↓	11	8
B0-7	CPR ↓	11	8
$\overline{CES}$	CPS ↓	15	4
$\overline{CER}$	CPR ↓	15	4

B. Propagation Delays

Input	A0-7	B0-7	FS	FR
CPS ↓	28	–	20	–
CPR ↓	–	28	–	20
CLRS ↓	–	–	22	–
CLRR ↓	–	–	–	22

C. Recovery Times

From	To	tREC
CLRS ↓	CPS ↓	34
CLRR ↓	CPR ↓	34

D. Pulse-Width Requirements

Input	Min. LOW Pulse Width	Min. HIGH Pulse Width
CPS	20	20
CPR	20	20
CLRS	20	20
CLRR	20	20

E. Enable/Disable Times

From	To	Disable	Enable
$\overline{OE}$AS	A0-7	24	28
$\overline{OE}$BR	B0-7	24	28

APPLICATIONS

The Am2950 • Am2951 provides data transfer handshaking signals as well as eight-bit, bidirectional data storage. Its flexibility allows it to be used in any type of computer system, including Am2900, 8080, 8085, 8086, Z80, and Z8000 systems.

Figure 1 shows an Am2950 used to store data moving in both directions between a bidirectional system data bus and a bidirectional peripheral data bus. The on-chip Flag flip-flops provide the data in, data out handshaking signals required for data transfer and interrupt request generation.

Figure 2 shows a multiple I/O port system using Am2950's. Two Am2950's are used at each port to interface the 16-bit system data bus. The Am2950 flags are used to generate I/O interrupt requests.

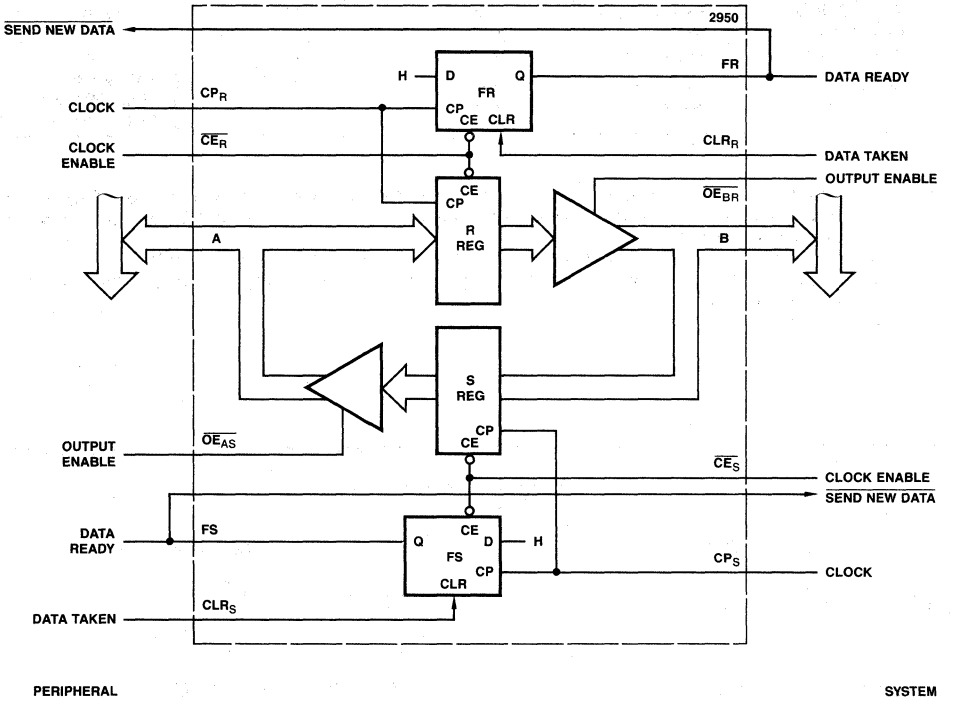


Figure 1. A Bidirectional I/O Port with Handshaking Using the Am2950.

MPR-578

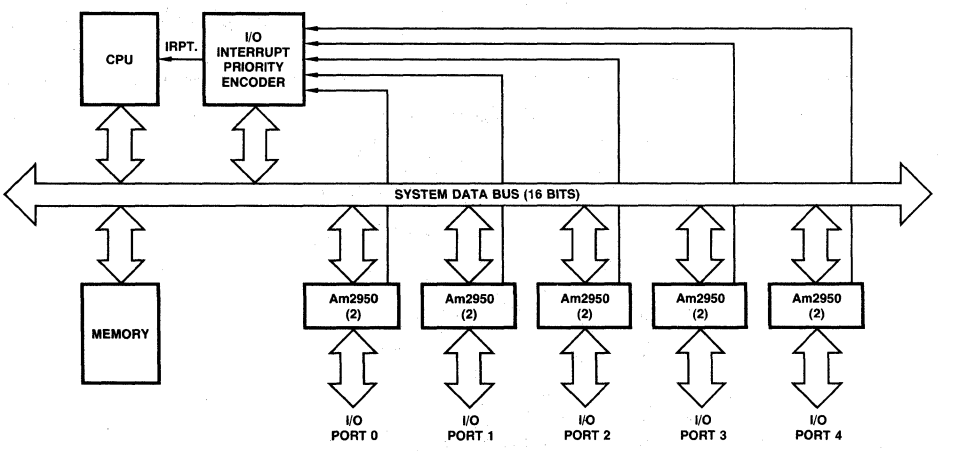


Figure 2. Multiple I/O Port System.

MPR-579

ORDERING INFORMATION

Order the part number according to the table below to obtain the desired package, temperature range, and screening level.

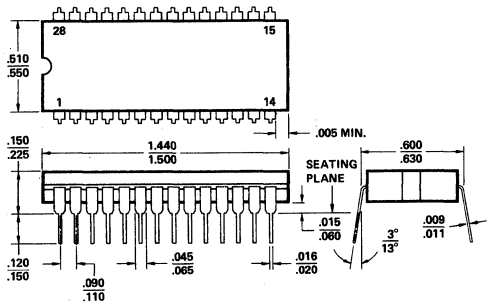
Am2950 Order Number	Am2951 Order Number	Package Type (Note 1)	Operating Range (Note 2)	Screening Level (Note 3)
AM2950PC	AM2951PC	P-28	C	C-1
AM2950DC	AM2951DC	D-28	C	C-1
AM2950DC-B	AM2951DC-B	D-28	C	B-2 (Note 4)
AM2950DM	AM2951DM	D-28	M	C-3
AM2950DM-B	AM2951DM-B	D-28	M	B-3
AM2950FM	AM2951FM	F-28-2	M	C-3
AM2950FM-B	AM2951FM-B	F-28-2	M	B-3
AM2950XC	AM2951XC	Dice	C	Visual inspection to MIL-STD-883 Method 2010B.
AM2950XM	AM2951XM	Dice	M	

Notes: 1. P = Molded DIP, D = Hermetic DIP, F = Flat Pak. Number following letter is number of leads. See Appendix B for detailed outline. Where Appendix B contains several dash numbers, any of the variations of the package may be used unless otherwise specified.
2. C = 0°C to +70°C, V_{CC} = 4.75V to 5.25V, M = − 55°C to +125°C, V_{CC} = 4.50V to 5.50V.
3. See Appendix A for details of screening. Levels C-1 and C-3 conform to MIL-STD-883, Class C. Level B-3 conforms to MIL-STD-883, Class B.
4. 96 hour burn-in.

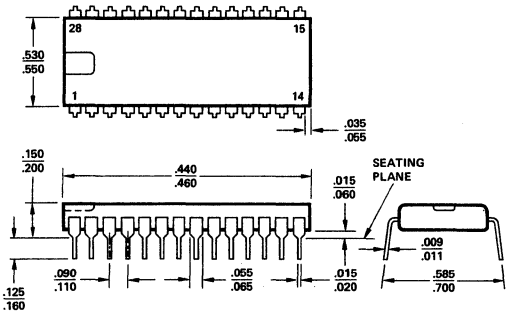
6

Physical Dimensions
Dual-In-Line

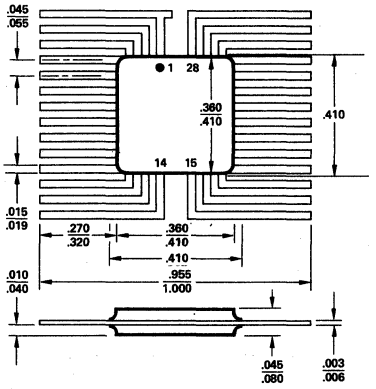
28-Pin Hermetic



28-Pin Molded



28-Pin Flat-Package



High-Performance Multipliers

7

Am25S557 • Am25S558

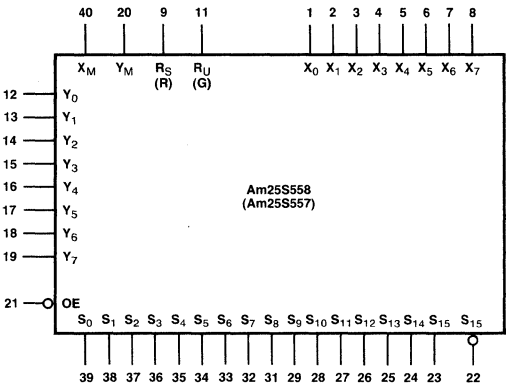
Eight-Bit by Eight-Bit Combinatorial Multiplier

PRELIMINARY

DISTINCTIVE CHARACTERISTICS

- Multiplies two 8-bit numbers – 16-bit output
- Combinatorial – no clocks required
- Full 8 x 8 multiply in 45ns typ.
- Cascades to 16 x 16 in 110ns typ.
- Expandable to multiples of 8 bits
- MSB and $\overline{\text{MSB}}$ outputs for easy expansion
- Unsigned, two's complement or mixed operands
- Implements common rounding algorithms with additional logic
- Three-state outputs
- Transparent 16-bit latch in Am25S557
- Industry standard pin-outs
- 100% product assurance screening to MIL-STD-883 requirements

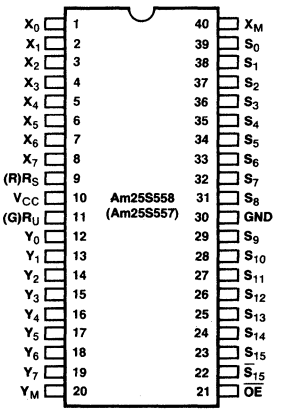
LOGIC SYMBOL



V_{CC} = Pin 10
GND = Pin 30

BLI-035

CONNECTION DIAGRAM



Pin assignments shown are for Am25S558. G and R shown in parentheses are pin assignments for Am25S557.

BLI-036

FUNCTIONAL DESCRIPTION

The Am25S557 and Am25S558 are high-speed, combinatorial, 8 x 8-bit multipliers. Both use an array of full adders to form and add partial products in a single unclocked operation, resulting in a 16-bit parallel output product.

Mode control inputs X_M and Y_M allow the multiplier to accept either unsigned or two's complement numbers from either respective input to provide an unsigned or signed output. The mode control lines are held LOW for unsigned input words and HIGH for two's complement.

The Am25S557 and Am25S558 are easily expandable to longer work lengths. Both S_{15} and $\overline{S}_{15}$ are available to allow expansion in either signed or unsigned modes without external inverters. In the 16-bit by 16-bit configuration (32-bit output) the typical multiply time is 110ns.

Both configurations offer three-state output flexibility and the Am25S557 adds a 16-bit transparent latch between the multiplier array and the three-state output buffers (including $\overline{S}_{15}$).

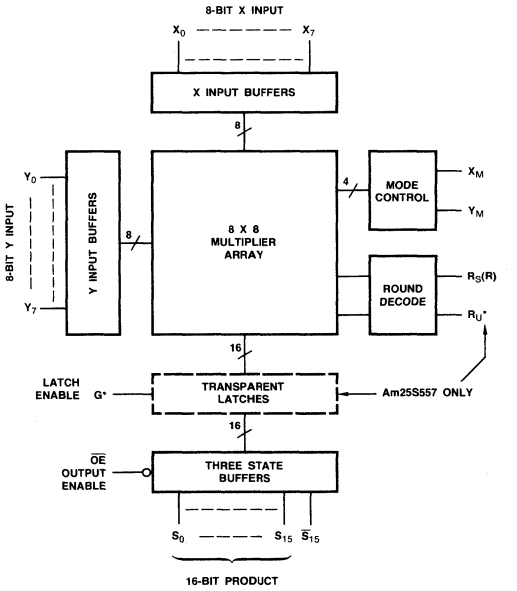
Rounding provisions for 8-bit truncated output configurations are particularly optimized for maximum flexibility. The Am25S557 internally develops proper rounding for either signed or unsigned numbers by combining rounding input R with X_M, Y_M, $\overline{X}_M$ and $\overline{Y}_M$ as follows:

$R_U = \overline{X}_M \cdot \overline{Y}_M \cdot R$ = Unsigned Rounding input to 2^7 adder.

$R_S = (X_M + Y_M)\, R$ = Signed Rounding input to 2^6 adder.

Since the Am25S558 does not require the use of pin 9 for the latch enable input, (G), R_S and R_U are brought out separately.

LOGIC DIAGRAM



*Pin 11 is G for Am25S557 and R_U for Am25S558.

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L T_A = 0°C to +70°C V_{CC} = 5.0V ±5% (MIN. = 4.75V MAX. = 5.25V)
MIL T_C = −55°C to +125°C V_{CC} = 5.0V ±10% (MIN. = 4.50V MAX. = 5.50V)

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)			Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN. V_{IN} = V_{IH} or V_{IL}	V_{IL} = 0.8V V_{IH} = 2.0V	I_{OH} = −2.0mA	2.4	3.0		Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN. V_{IN} = V_{IH} or V_{IL}	V_{IL} = 0.8V V_{IH} = 2.0V I_{OL} = 8.0mA			0.3	0.5	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs			2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs		MIL			0.8	Volts
				COM'L			0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA					−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.5V					−1.0	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.4V					100	µA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V					1	mA
I_O	Off-State (High-Impedance) Output Current	V_{CC} = MAX.	V_O = 0.5V				−100	µA
			V_O = 2.4V				+100	
I_{SC}	Output Short Circuit Current (Note 3)	V_{CC} = MAX.			−20		−90	mA
I_{CC}	Power Supply Current (Note 4)	V_{CC} = MAX.					280	mA

Notes:
1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. Test with pin 21 at 4.5V, all other input pins at GND, all outputs open. Am25S557 conditions the same except initialize with G (pin 11) at 4.5V, then GND.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to +V_{CC} max.
DC Input Voltage	−0.5V to +5.5V
DC Output Current, Into Outputs	30mA
DC Input Current	−30mA to +5.0mA

Am25S557 SWITCHING CHARACTERISTICS OVER OPERATING RANGE*

Parameters	Description	Am25S COM'L T_A = 0 to 70°C V_{CC} = 5V ±5%			Am25S MIL T_A = −55 to +125°C V_{CC} = 5V ±10%			Units	Test Conditions
		Min	Typ	Max	Min	Typ	Max		
t_{PD}	X_i, Y_i to S_0 to S_{14}		45			55		ns	C_L = 30pF R_L = 560Ω (See test figures)
t_{PD}	X_i, Y_i to S_{15} or $\overline{S}_{15}$		50			60		ns	
t_s	X_i, Y_i to G Set-up Time		35			45		ns	
t_h	X_i, Y_i to G Hold Time		−10			−10		ns	
t_{PD}	G to S_i		40			45		ns	
t_{PW}	Latch Enable Pulse Width		15			25		ns	
t_{PHZ}	$\overline{OE}$ to S_i		15			20		ns	
t_{PLZ}	$\overline{OE}$ to S_i		20			25		ns	
t_{PZH}	$\overline{OE}$ to S_i		25			25		ns	
t_{PZL}	$\overline{OE}$ to S_i		20			25		ns	

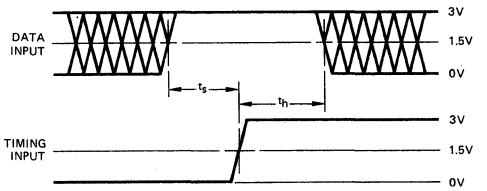
*AC performance over the operating temperature range is guaranteed by testing defined in Group A, subgroup 9.

Am25S558
SWITCHING CHARACTERISTICS OVER OPERATING RANGE*

		Am25S COM'L T_A = 0 to 70°C V_{CC} = 5V ±5%			Am25S MIL T_A = −55 to +125°C V_{CC} = 5V ±10%				
Parameters	**Description**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Units**	**Test Conditions**
t_{PD}	X_i, Y_i to S_0 to S_6		40			50		ns	C_L = 30pF, R_L = 560Ω (See test figures)
t_{PD}	X_i, Y_i to S_7 to S_{14}		40			50		ns	
t_{PD}	X_i, Y_i to S_{15} or $\overline{S}_{15}$		45			55		ns	
t_{PHZ}	$\overline{OE}$ to S_i		15			20		ns	
t_{PLZ}	$\overline{OE}$ to S_i		20			25		ns	
t_{PZH}	$\overline{OE}$ to S_i		25			25		ns	
t_{PZL}	$\overline{OE}$ to S_i		20			25		ns	

*AC performance over the operating temperature range is guaranteed by testing defined in Group A, subgroup 9.

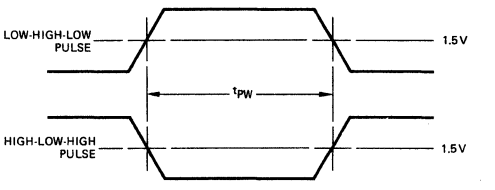
SET-UP AND HOLD TIMES



BLI-057

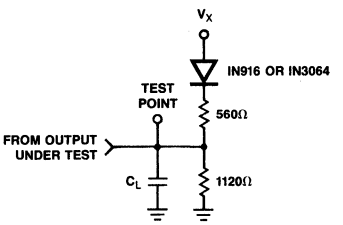
Notes: 1. Diagram shown for HIGH data only. Output transition may be opposite sense.
2. Cross hatched area is don't care condition.

PULSE WIDTH



BLI-058

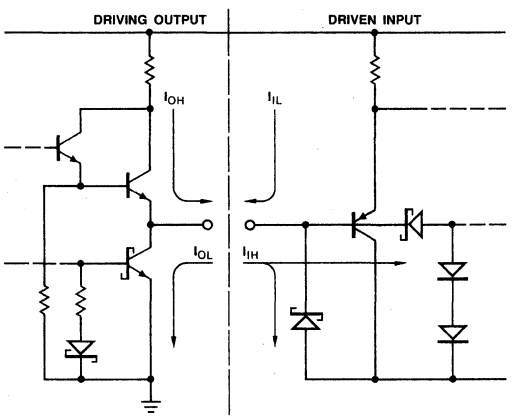
LOAD TEST CIRCUIT



BLI-059

C_L Includes probe and jig capacitance.

Am25S557 • Am25S558
INPUT/OUTPUT CURRENT INTERFACE CONDITIONS



BLI-038

TEST WAVEFORMS

Test	V_X	Output Waveform – Measurement Level
All t_{PD}s	5.0V	V_{OH}, V_{OL}, 1.5V
t_{PHZ}	0.0V	V_{OH}, 0.5V, 0.0V
t_{PLZ}	5.0V	2.8V, 0.5V, V_{OL}
t_{PZH}	0.0V	V_{OH}, 1.5V, 0.0V
t_{PZL}	5.0V	2.8V, 1.5V, V_{OL}

DEFINITION OF TERMS

X_0-X_7	Multiplicand 8-bit data inputs
Y_0-Y_7	Multiplier 8-bit data inputs
X_M, Y_M	Mode control inputs for each data word; LOW for unsigned data and HIGH for two's complement data
S_0-S_{15}	Product 16-bit output
$\overline{S_{15}}$	Inverted MSB for expansion
R_S, R_U	Rounding inputs for signed and unsigned data, respectively (Am25S558 only)
G	Transparent Latch Enable (Am25S557 only)
$\overline{OE}$	Three-state enable for S_0-S_{15} outputs
R	Rounding input for signed or unsigned data (combined internally with X_M, Y_M in Am25S557 only)

MODE CONTROL INPUTS

Operating Mode	Input Data		Mode Control Inputs	
	X_0-X_7	Y_0-Y_7	X_M	Y_M
UNSIGNED	UNSIGNED	UNSIGNED	L	L
MIXED	UNSIGNED	2's COMP	L	H
	2's COMP	UNSIGNED	H	L
SIGNED	2's COMP	2's COMP	H	H

ROUNDING INPUTS

Am25S557

Inputs			Adds	
X_M	Y_M	R	2^7	2^6
L	L	H	YES	NO
L	H	H	NO	YES
H	L	H	NO	YES
H	H	H	NO	YES
X	X	L	NO	NO

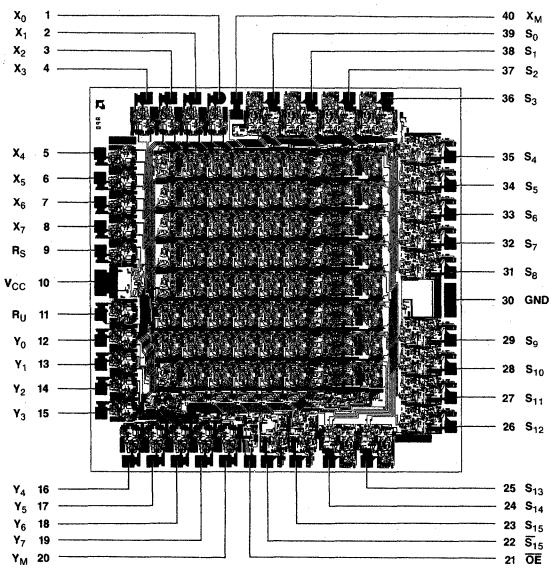
Am25S558

Inputs		Adds		Normally Used With	
R_U	R_S	2^7	2^6	X_M	Y_M
L	L	NO	NO	X	X
L	H	NO	YES	$X_M + Y_M = H$	
H	L	YES	NO	L	L
H	H	YES	YES	*	*

* Most rounding applications require a HIGH level for R_U or R_S, but not both.

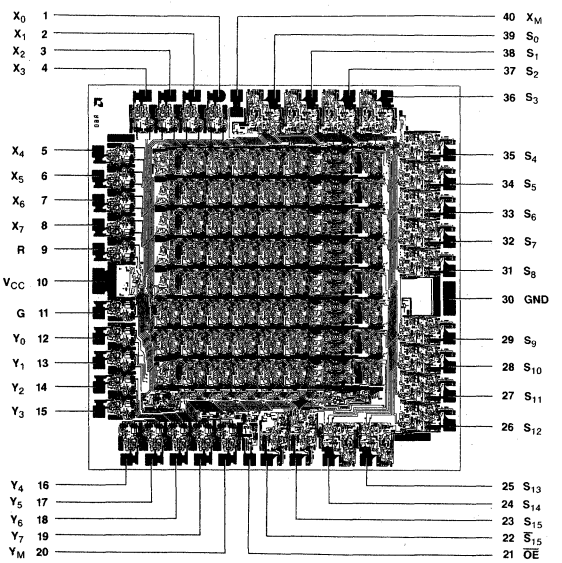
Metallization and Pad Layouts

Am25S557



DIE SIZE 0.171" X 0.165"

Am25S558



DIE SIZE 0.171" X 0.165"

APPLICATION

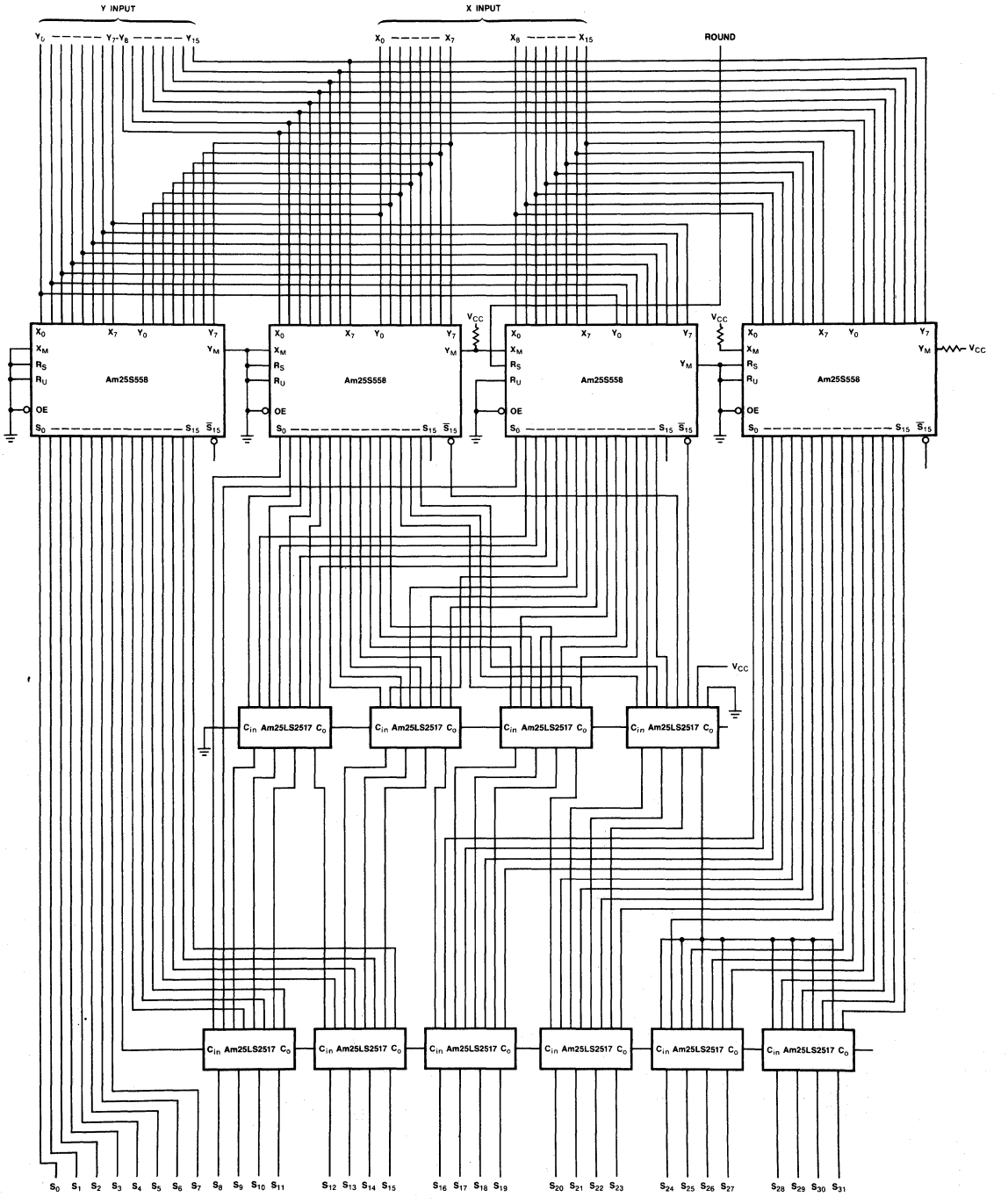
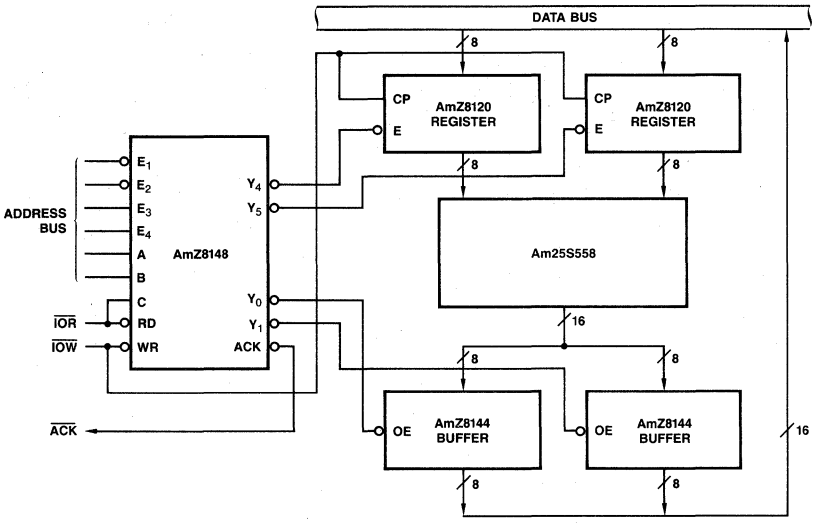


Figure 2. High-Speed 16 x 16 2's Complement Multiplication.

BLI-040

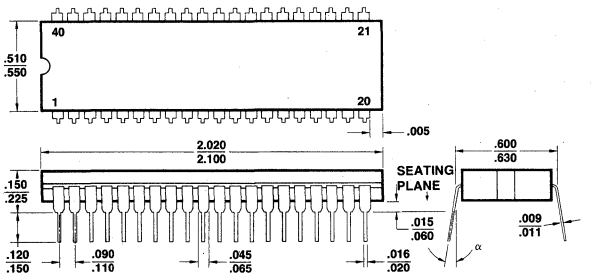
I/O MAPPED INTERFACE WITH MOS MICROPROCESSOR



BLI-060

PHYSICAL DIMENSIONS
Dual-In-Line

40-Pin Cerdip



ORDERING INFORMATION

Package Type	Temperature Range	Am25S557 Order Number	Am25S558 Order Number
Hermetic DIP	0°C to +70°C	AM25S557DC	AM25S558DC
	−55°C to +125°C	AM25S557DM	AM25S558DM

AmZ8000
Peripheral/Support Products

THE AmZ8000 COMPONENT FAMILY

The following 19 circuits – available now or by the end of 1980 – are LSI devices providing support and peripheral functions for the AmZ8000 Family.

AmZ8010 MMU

Memory Management Unit. This circuit interfaces between the segmented CPU and physical memory. It translates segment numbers and offsets from the CPU into physical memory addresses through a programmable table. It also provides facilities for preventing unauthorized access to a segment.

AmZ8030 SCC

Serial Communications Controller. A two-channel serial I/O port capable of handling a variety of synchronous and asynchronous formats, including SDLC, HDLC and Bisync. The device includes CRC-16 and CCITT block frame checking and separate modem controls for two full-duplex channels.

AmZ8036 CIO

Counter/Timer Parallel I/O Port. Includes two general-purpose double-buffered 8-bit I/O ports as well as three independent 16-bit counters. Each bit of the parallel ports can be separately programmed for input or output.

AmZ8038 FIO

FIFO Input/Output Interface. A general purpose 8-bit I/O port which includes a 128-word FIFO in the data path. Devices can be cascaded to form wider words or a deeper stack.

AmZ8052 CRT

CRT Controller. A raster-scan CRT controller with a host of features for sophisticated display systems. This device includes control for horizontal and vertical split screens, superscripts and subscripts, simple line drawing capability and linked list processing.

AmZ8065 BEP

Burst Error Processor. Implements IBM (48- and 56-bit) polynomials, CDC and DEC polynomials. Detects burst errors and permits corrections of errors up to 11 bits in length.

AmZ8068 DCP

Data Ciphering Processor. Implements the NBS encryption algorithm. It interfaces directly to the AmZ8000 CPU bus and handles data rates up to 1M byte/sec.

AmZ8073 STC

System Timing Controller. The STC includes five independent 16-bit counters that can be programmed to count up or down in binary or BCD from a number of different clock sources. The device provides for frequency synthesis, digital one-shots, time-of-day, coincidence alarms and much more.

AmZ8160 EDC

Error Detection and Correction Circuit. Provides polynomial error detection and correction on parallel 16-bit words with six extra bits. Detects all double errors and corrects all single errors in less than 100nsec.

AmZ8161/2

Quad data buffers in a "T" configuration for interface between the data bus and the memory. It provides a connection to the EDC circuit for data correction on the fly.

AmZ8164 DMC

Dynamic Memory Controller. Provides address latches, refresh address counter and RAS decode for dynamic RAMs.

AmZ8165/6

High-drive quad buffers with matched impedance in HIGH and LOW states, to drive high-capacitance loads with controlled rise and fall times.

AmZ8140/44

Octal three-state buffers. Inverting and nonverting versions.

AmZ8103/4 AmZ8107/8

Octal bus transceivers. Similar to 8304. Noninverting and inverting versions.

AmZ8120

Octal edge-triggered register with a reset, clock enable and three-state outputs.

AmZ8133/8173

Octal latch with three-state outputs. Both inverting and non-inverting versions.

AmZ8121

Equality comparator. Signals equality between two 8-bit words. Used for address comparison.

AmZ8136

Eight-bit decoder with control storage. On-chip register stores inputs to decoder so encoded chip-select field can be strobed in.

AmZ8148

Address Decoder with Acknowledge. A one-of-eight decoder with multiple enable inputs and an acknowledge output.

AmZ8127

AmZ8000 Clock Generator

Advanced Information

DISTINCTIVE CHARACTERISTICS

- **High-drive high-level clock output**
  Special output provides clock signal matched to requirements of AmZ8000 CPU, MMU and DMA devices.
- **Four TTL-level clocks**
  Generates synchronized TTL compatible clocks at 16MHz, 2MHz and 1MHz to drive memory circuits and LSI peripheral devices. An additional TTL clock is synchronized with the high-level clock for registers, latches and other peripherals.
- **Synchronized WAIT state and time-out controls**
  On-chip logic generates WAIT signal under control of Halt, Single-step and Ready signals. Automatic time-out of peripheral wait requests.

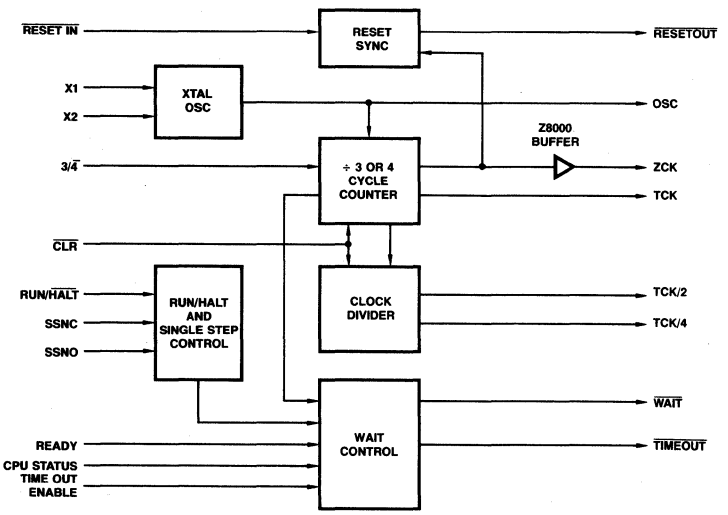
FUNCTIONAL DESCRIPTION

The AmZ8127 Clock Generator and Controller provides the clock oscillator, frequency dividers and clock drivers for the complete array of AmZ8000 CPUs, peripherals and memory system configurations. In addition to the special 4MHz output driver for the AmZ8001 and AmZ8002 CPUs, a standard buffered TTL 16MHz oscillator output is provided for dynamic memory timing and control. The AmZ8127 forms an integral part of the dynamic memory support chip set including the AmZ8163 EDC and Refresh Controller, AmZ8164 Dynamic Memory Controller, AmZ8160 Error Detection and Correction Unit and AmZ8161/AmZ8162 EDC Bus Buffers. The oscillator is designed to operate with a 16MHz crystal or with external 16MHz drive. The AmZ8127 uses an internal divide-by-4 to provide 4MHz clock drive to the AmZ8001/AmZ8002 CPU. Additional dividers generate synchronous buffered 2MHz and 1MHz clock outputs for use by peripheral devices. The clock divider counters are clearable to allow synchronizing the multiple clock outputs.

The controller functions include $\overline{\text{RESET}}$, RUN/$\overline{\text{HALT}}$, SINGLE-STEP, READY and a READY TIMEOUT counter which limits a peripheral's wait request to 16 clock cycles. The CPU's $\overline{\text{WAIT}}$ input is controlled by RUN/$\overline{\text{HALT}}$, SINGLE-STEP and READY. A $\overline{\text{HALT}}$ command to the AmZ8127 drives the $\overline{\text{WAIT}}$ output LOW causing the CPU to add wait states (TW to TW). The READY input is used by peripherals to request wait states. The active HIGH input TIMEOUT ENABLE is used to force TIMEOUT and $\overline{\text{WAIT}}$ to HIGH 16 clock cycles after a peripheral has requested a wait but fails to release the request. The CPU status lines ST1, ST2 and ST3 are decoded in the AmZ8127 to disable the TIME-OUT counter during CPU "Internal Operations" and during refresh.

BLOCK DIAGRAM

CLOCK GENERATOR



BLI-168

High-Performance Microprocessor Support Products (Am2900 Family)

9

Am2925

Clock Generator and Microcycle Length Controller

ADVANCED INFORMATION

DISTINCTIVE CHARACTERISTICS

- Single chip clock generator and driver
- Four different clock output waveforms for Am2900 and other bipolar and MOS systems
- Crystal controlled for stable system operation
- Oscillator to 31MHz – oscillator output for external system timing
- Clock halt, single-step and wait controls
- Variable cycle lengths – 1 of 8 different cycle lengths may be programmed
- Slim 0.3" 24-pin package
- 100% product assurance screening to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

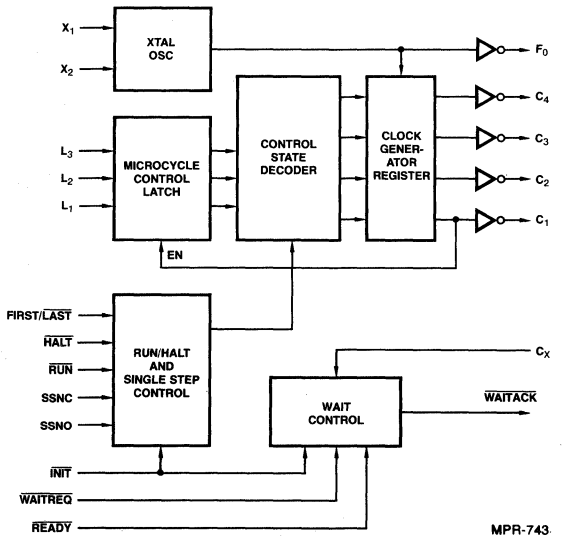
The Am2925 is a single-chip general purpose clock generator/driver. It is controlled by a crystal, selected by the designer, and is microprogrammable to meet a variety of system speed requirements. The Am2925 generates four different clock output waveforms tailored to meet the needs of Am2900 and other bipolar and MOS microprocessor based systems. Also, variable cycle lengths may be generated under microprogram control. One of eight different cycle lengths may be microprogrammed using the Cycle Length inputs L_1, L_2, and L_3.

The Am2925 oscillator runs at frequencies up to 31MHz. A buffered oscillator output is provided for external system timing.

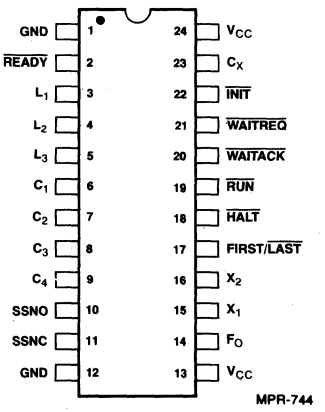
Clock halt, single-step and wait controls are provided for the Am2925. The HALT REQ input halts the clocks; the clocks resume when the HALT REQ input is deactivated. The SINGLE-STEP input, which operates only when the clocks are halted, generates the clocks for a single cycle. The WAIT REQ input stops the clocks and puts the Am2925 in a "wait" state. In this state, the clocks remain stopped until an asynchronous READY input signal is received. The WAIT ACK output indicates when the Am2925 is in the "wait" state. The WAIT REQ and READY inputs are pulse sensitive and are overridden by the HALT REQ input.

One of eight cycle lengths may be microprogrammed using the L_1, L_2, and L_3 inputs. There are four clock output waveforms for each of the 8 possible cycle lengths.

LOGIC DIAGRAM



CONNECTION DIAGRAM
Top View



24-pin slim (0.3")

Note: Pin 1 is marked for orientation.

The Am2925 is scheduled for introduction in 2Q '80.

Am2927 • Am2928

Quad Three-State Bus Transceivers With Clock Enable

DISTINCTIVE CHARACTERISTICS

- Quad high-speed LSI bus-transceivers
- Three-state bus driver and receiver outputs
- D-type register on drivers
- Latch output on Am2927
- Registered output on Am2928
- Output data to input wrap around gating
- Input register to output transfer gating with or without driving data bus
- Clock enabled registers
- Bus driver outputs can sink 48mA at 0.5V max.
- Three-state receiver outputs sink 20mA at 0.5V max.
- 3.5V minimum V_{OH} for direct interface to MOS microprocessors
- Advanced low-power Schottky processing
- 100% reliability assurance testing in compliance with MIL-STD-883

FUNCTIONAL DESCRIPTION

The Am2927 and Am2928 are high-performance, low-power Schottky, quad bus transceivers intended for use in bipolar or MOS microprocessor system applications.

Both devices feature register enable lines which function as clock enables without introducing gate delay in the clock inputs. The four transceivers share common enables, clock, select and three-state control lines.

The Am2927 consists of four D-type edge-triggered flip-flops. Each flip-flop output is connected to a three-state data bus driver and separately to the input of a corresponding receiver latch input. The receiver latch can select input from the driver or the data bus. The select line determines the source of input data for the bus driver choosing between input data or data recirculated from the receiver output. The receiver output also has a three-state output buffer.

The combination of the select input, S, the driver input enable, $\overline{ENDR}$, and the receiver latch enable, $\overline{RLE}$, provide seven different data path operating modes not available in other transceivers. For example, transmitted data can be stored in the receiver for subsequent retransmission. Also, received data can be output to the system and simultaneously fed back to the driver input.

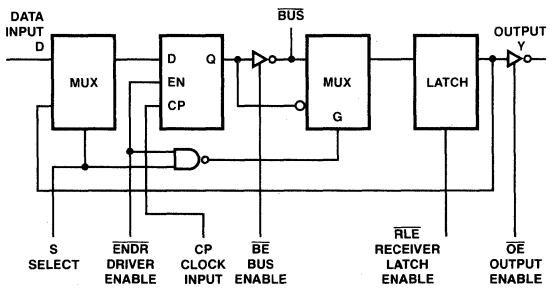
The Am2928 is similar to the Am2927, but with a D-type edge-triggered register in the receiver and a receiver enable, $\overline{ENREC}$, which functions as a common clock enable.

Data from each D input is inverted at the bus output. Likewise, data at the bus input is inverted at the receiver output.

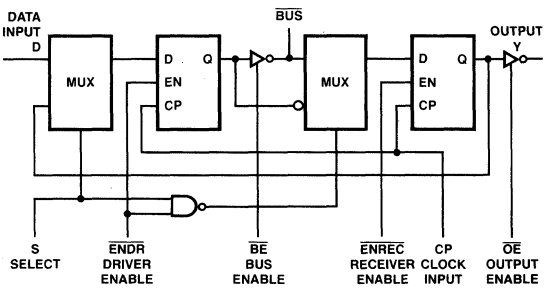
All three-state controls and enable lines are active low (the Am2927 receiver latch is transparent when $\overline{RLE}$ is LOW). The select input, S, determines whether the enabled driver input accepts data from the data input, D, or from the corresponding receiver output, Y. Similarly, the select line determines whether the receiver accepts input data from the data bus, or the driver output.

SIMPLIFIED LOGIC DIAGRAMS

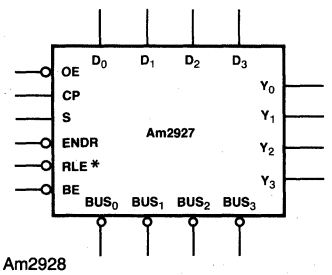
Am2927



Am2928



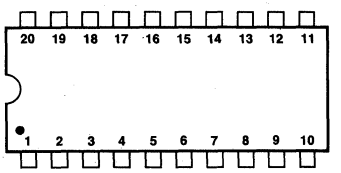
LOGIC SYMBOL



*ENREC for Am2928

BLI-089

CONNECTION DIAGRAM
Top View

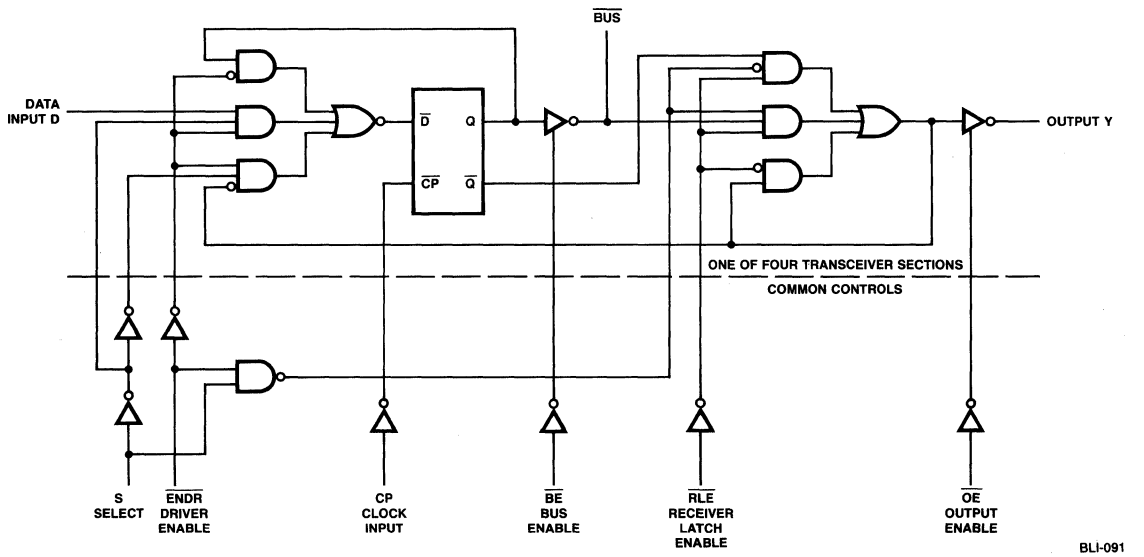


Note: Pin 1 is marked for orientation.

BLI-090

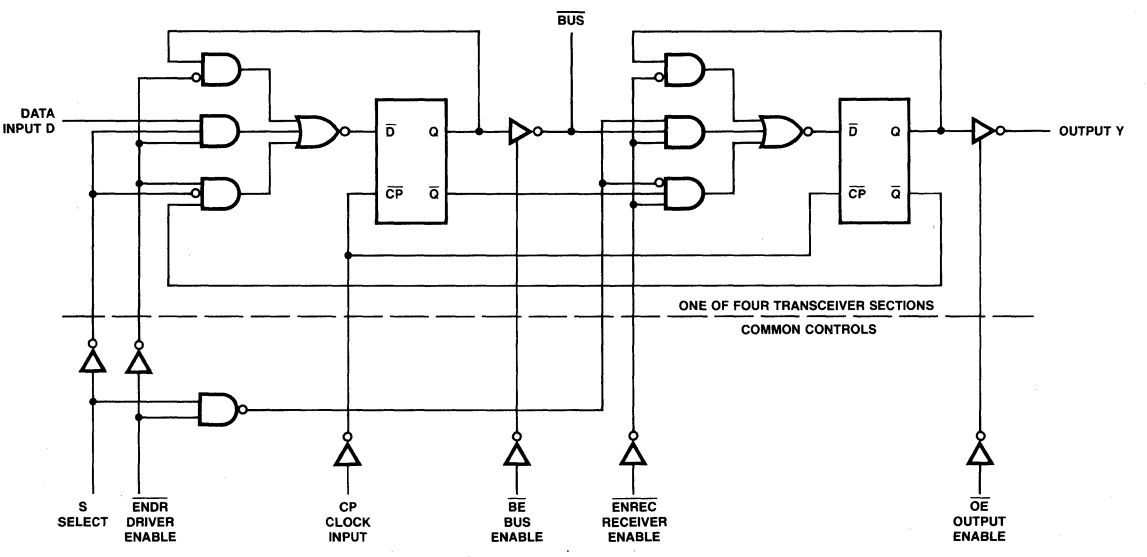
DETAILED LOGIC DIAGRAMS

Am2927



BLI-091

Am2928



BLI-092

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Temperature (Ambient) Under Bias	−55 to +125°C
Supply Voltage to Ground Potential	−0.5 to +7V
DC Voltage Applied to Outputs for HIGH Output State	−0.5 V to +V_{CC} max.
DC Input Voltage	−0.5 to +5.5V
DC Output Current, Into Outputs (Except BUS)	30 mA
DC Output Current, Into Bus	100 mA
DC Input Current	−30 to +5.0mA

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L	T_A = 0°C to +70°C	V_{CC} = 5.0V ±5%	(MIN. = 4.75V MAX. = 5.25V)
MIL	T_A = −55°C to +125°C	V_{CC} = 5.0V ±10%	(MIN. = 4.50V MAX. = 5.50V)

BUS INPUT/OUTPUT CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameters	Description	Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Units
V_{OL}	Bus Output LOW Voltage	V_{CC} = MIN.	I_{OL} = 24mA			0.4	Volts
			I_{OL} = 48mA			0.5	
V_{OH}	Bus Output HIGH Voltage	V_{CC} = MIN.	COM'L, I_{OH} = −20mA	2.4			Volts
			MIL, I_{OH} = −15mA	2.4			
V_{IH}	Receiver Input HIGH Threshold	Bus Enable = 2.4V		2.0			Volts
V_{IL}	Receiver Input LOW Threshold	Bus Enable = 2.4V				0.8	Volts
I_{OFF}	Bus Leakage Current (Power Off)	V_{CC} = 0V, V_O = 4.5V				100	µA
I_O	Bus Leakage Current (HIGH Impedance)	V_{CC} = MAX Bus Enable = 2.4V	V_O = 0.4V			−200	µA
			V_O = 2.4V			50	
			V_O = 4.5V			100	
I_{SC}	Bus Output Short Circuit Current	V_{CC} = MAX, V_O = 0V		−50		−255	mA
C_B	Bus Capacitance (Note 4)	V_{CC} = 0V			8		pF

Notes: 1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. This parameter is typical of device characterization data and is not tested in production.

PRELIMINARY

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L	T_A = 0°C to +70°C	V_{CC} = 5.0V ±5%	(MIN. = 4.75V MAX. = 5.25V)
MIL	T_A = −55°C to +125°C	V_{CC} = 5.0V ±10%	(MIN. = 4.50V MAX. = 5.50V)

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Units
V_{OH}	Receiver Output HIGH Voltage	V_{CC} = MIN. V_{IN} = V_{IH} or V_{IL}	MIL, I_{OH} = −2.0mA	2.4	3.4		Volts
			COM'L, I_{OH} = −6.5mA	2.4	3.4		
		V_{CC} = 5.0V	I_{OH} = −100µA	3.5			
V_{OL}	Output LOW Voltage (Except Bus)	V_{CC} = MIN V_{IN} = V_{IL} or V_{IH}	I_{OL} = 20mA			0.5	Volts
V_{IH}	Input HIGH Level (Except Bus)	Guaranteed input logical HIGH voltage for all inputs		2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs	MIL			0.8	Volts
			COM'L			0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA				−1.2	Volts
I_{IL}	Input LOW Current (Except Bus)	V_{CC} = MAX., V_{IN} = 0.5V				−2.0	mA
I_{IH}	Input HIGH Current (Except Bus)	V_{CC} = MAX., V_{IN} = 2.7V	S, ENDR			100	µA
			All other inputs			50	
I_I	Input HIGH Current (Except Bus)	V_{CC} = MAX., V_{IN} = 7.0V	S, ENDR			2.0	mA
			All other inputs			1.0	
I_O	Off-State Output Current (Receiver Output)	V_{CC} = MAX.	V_O = 2.4V			50	µA
			V_O = 0.5V			−50	
I_{SC}	Output Short Circuit Current (Except Bus)	V_{CC} = MAX.	Receiver	−40		−100	mA
I_{CC}	Power Supply Current	V_{CC} = MAX.					mA

Am2927
SWITCHING CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameters	Description	Test Conditions	Am2927XM Min	Am2927XM Typ	Am2927XM Max	Am2927XC Min	Am2927XC Typ	Am2927XC Max	Units
t_{PLH}	Driver Clock, CP, to $\overline{BUS}$	C_L (BUS) = 50pF, R_L (BUS) = 130Ω					18		ns
t_{PHL}							18		
t_{ZH} • t_{ZL}	Bus Enable, $\overline{BE}$, to $\overline{BUS}$						14		ns
t_{HZ} • t_{LZ}		C_L = 5pF					12		
t_{PW}	Min. Clock Pulse Width (HIGH or LOW)					10			ns
t_{PLH}	$\overline{BUS}$ to Receiver Output (Latch Enabled)						16		ns
t_{PHL}							16		
t_{PLH}	Latch Enable, $\overline{RLE}$, to Receiver Output						18		ns
t_{PHL}							18		
t_{ZH} • t_{ZL}	Output Enable, $\overline{OE}$, to Output	C_L = 15pF, R_L = 2.0kΩ					16		ns
t_{HZ} • t_{LZ}		C_L = 5pF, R_L = 2.0kΩ					14		
t_s	Driver Enable, $\overline{ENDR}$, to Clock					8			ns
t_h						2			
t_s	Select, S, to Clock ($\overline{RLE}$ = HIGH)					10			ns
t_h						0			
t_s	Select, S, to Clock ($\overline{RLE}$ = LOW)					13			ns
t_h						0			
t_s	Data Inputs, D, to Clock					7			ns
t_h						2			
t_s	$\overline{BUS}$ to Latch Enable, $\overline{RLE}$					10			ns
t_h						3			

Notes: 1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

PRELIMINARY

Am2928
SWITCHING CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE

Parameters	Description	Test Conditions	Am2928XM Min	Am2928XM Typ	Am2928XM Max	Am2928XC Min	Am2928XC Typ	Am2928XC Max	Units
t_{PLH}	Clock, CP, to $\overline{BUS}$	C_L (BUS) = 50pF, R_L (BUS) = 130Ω					18		ns
t_{PHL}							18		
t_{ZH} • t_{ZL}	Bus Enable, $\overline{BE}$, to $\overline{BUS}$						14		ns
t_{HZ} • t_{LZ}		C_L = 5pF					12		
t_{PLH}	Clock, CP, to Output, Y	C_L = 15pF, R_L = 2.0kΩ					18		ns
t_{PHL}							18		
t_{PW}	Min. Clock Pulse Width (HIGH or LOW)					10			ns
t_{ZH} • t_{ZL}	Output Enable, $\overline{OE}$, to Output, Y	C_L = 15pF, R_L = 2.0kΩ					16		ns
t_{HZ} • t_{LZ}		C_L = 5pF, R_L = 2.0kΩ					14		
t_s	Driver Enable, $\overline{ENDR}$, to Clock					8			ns
t_h						2			
t_s	$\overline{BUS}$ to Clock (Receiver Register)					7			ns
t_h						2			
t_s	Receiver Enable, $\overline{ENREC}$, to Clock					8			ns
t_h						2			
t_s	S to Clock					10			ns
t_h						0			
t_s	Data Inputs, D, to Clock (Driver Register)					7			ns
t_h						2			

Notes: 1. For conditions shown an MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

DEFINITION OF FUNCTIONAL TERMS

CP — Clock Pulse to internal registers enters data on the LOW-to-HIGH transition.

$\overline{BE}$ — Bus Enable. When Bus Enable is LOW the four drivers drive the $\overline{BUS}$ outputs.

$\overline{BUS}_0$, $\overline{BUS}_1$, $\overline{BUS}_2$, $\overline{BUS}_3$ — The four driver outputs and receiver inputs.

D_0, D_1, D_2, D_3 — The four driver data inputs inverting from D to $\overline{BUS}$.

Y_0, Y_1, Y_2, Y_3 — The four receiver data outputs inverting from $\overline{BUS}$ to Y.

S — Select input controls data path modes in conjunction with $\overline{ENDR}$ and $\overline{RLE}$ (or $\overline{ENREC}$).

$\overline{OE}$ — Output Enable. When Output Enable is LOW the four receiver outputs Y are active.

$\overline{ENDR}$ — Driver Enable. Common clock enable for the input register. Allows the data on the D inputs to be loaded into the driver register on the clock LOW-to-HIGH transition.

$\overline{RLE}$ — Receiver Latch Enable (Am2927 only). When Receiver Latch Enable is LOW, the four receiver latches are transparent. The latches hold received data when $\overline{RLE}$ is HIGH.

$\overline{ENREC}$ — Receiver Enable (Am2928 only). Common clock enable for the receiver register. Allows the $\overline{BUS}$ driver or previous receiver data to enter the receiver register on the rising edge of the clock.

Am2927 FUNCTION TABLES

Driver Register Control

$\overline{ENDR}$	S	$\overline{RLE}$	Driver Register
H	X	X	Hold Previous Data
L	L	X	Load from D Input
L	H	L	Load from $\overline{BUS}$
L	H	H	Load Latched Receiver Data

Receiver Latch Control

$\overline{ENDR}$	S	$\overline{RLE}$	Receiver Output
X	X	H	Data Latched
H	H	L	Driver Register Output at Y Output (Latch Transparent)
X	L	L	Bus Data at Y Output (Latch Transparent)
L	X	L	

Am2928 FUNCTION TABLES

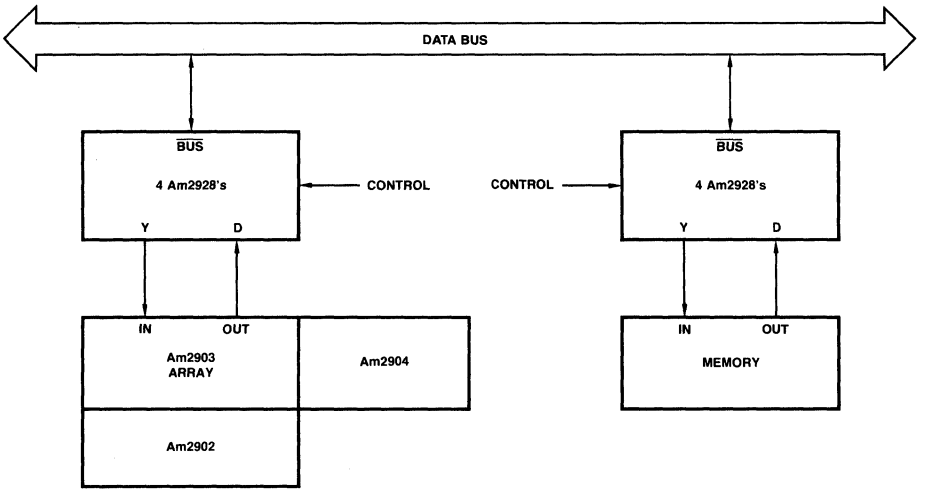
Driver Register Control

$\overline{ENDR}$	S	Driver Register
H	X	Hold Previous Data
L	L	Load from D Input
L	H	Load from Receiver Register

Receiver Register Control

$\overline{ENDR}$	S	$\overline{ENREC}$	Receiver Output
X	X	H	Hold Previous Data
H	H	L	Load from Driver Register
X	L	L	Load from $\overline{BUS}$
L	X	L	

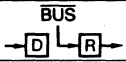
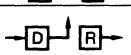
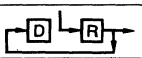
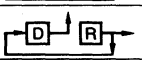
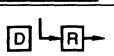
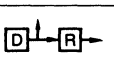
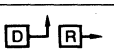
APPLICATION



The Am2927 and Am2928 can be used to provide Data Bus, Address Bus and Control Bus Interface in a high-speed bipolar microprocessor system.

BLI-093

Am2927 AND Am2928 FUNCTION TABLE

Driver Input From	Receiver Input From	Control Input Condition			Signal Flow	$\overline{BE}$
		S	$\overline{ENDR}$	*		
D Input	BUS	L	L	L		H
	(No Load)	L	L	H		L
Receiver	BUS	H	L	L		H
	(No Load)	H	L	H		L
(No Load)	BUS	L	H	L		H
	Driver	H	H	L		X
	(No Load)	X	H	H		L

*$\overline{RLE}$ for Am2927 (asynchronous) or $\overline{ENREC}$ for Am2928 (⌐).

Support and Peripheral Products

Am6012

12-Bit High-Speed Multiplying D/A Converter

Distinctive Characteristics

- All grades 12-bit monotonic over temperature
- Differential nonlinearity to ±.012% (13 bits) max. over temperature
- Trimless design is inherently monotonic
- Fast settling output current: 250nsec
- Full scale current 4mA
- High output impedance and compliance: −5 to +10V
- Differential current outputs
- Low cost
- High-speed multiplying capability
- Direct interface to TTL, CMOS, ECL, HTL, NMOS
- Performance unchanged over supply range
- Low power consumption: 230mW
- R_{OUT}, C_{OUT} independent of logic code

GENERAL DESCRIPTION

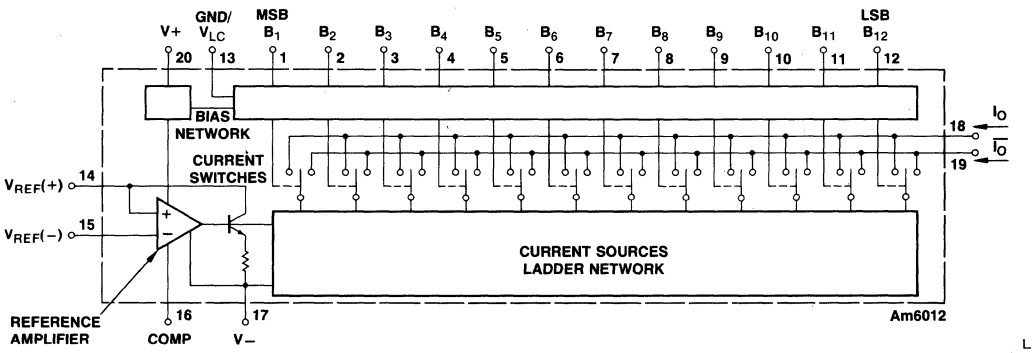
The Am6012 series of 12-bit monolithic multiplying Digital to Analog Converters represent a new level of high speed and accuracy coupled with low cost. The Am6012 is the first 12-bit D/A Converter ever built using standard processing without the requirements of thin film resistors and/or active trimming of individual devices. The Am6012 uses sophisticated new circuit design concepts that give inherent monotonicity without requiring ultra precision internal components.

The Am6012 design guarantees a more uniform step size than is possible with standard binarily weighted DAC's. This ±1/2 LSB differential nonlinearity is desirable in many applications where local linearity is critical. The uniform step size allows finer resolution of levels and in most applications is more useful than conformance to an ideal straight line from zero to full scale.

The Am6012 has high voltage compliance, high impedance dual complementary outputs which increase its versatility and enable differential operation to effectively double the peak to peak output swing. These outputs can be used directly without op amps in many applications. The dual complementary outputs can also be connected in A/D converter applications to present a constant load current and significantly reduce switching transients and increase system throughput. Output full scale current is specified at 4mA, allowing use of smaller load resistors to minimize the output RC delay which usually dominates settling time at the 12-bit level.

The Am6012 series guarantees full 12-bit monotonicity for all grades and differential nonlinearity as tight as ±.012% (13 bits) over the entire temperature range. Device performance is essentially independent of power supply voltage. The devices work over a wide operating range of +5, −12 volts to ±18 volts.

FUNCTIONAL DIAGRAM

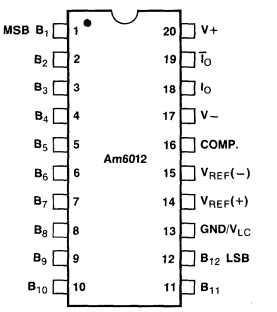


LIC-846

ORDERING INFORMATION

Order Number	Temperature Range	Differential Nonlinearity
AM6012ADM	−55°C to +125°C	±.012%
AM6012DM	−55°C to +125°C	±.025%
AM6012ADC	0°C to +70°C	±.012%
AM6012DC	0°C to +70°C	±.025%

CONNECTION DIAGRAM – Top View



Note:
Pin 1 is marked for orientation.

LIC-847

MAXIMUM RATINGS above which useful life may be impaired

Operating Temperature		Power Supply Voltage	±18V
Am6012ADM, Am6012DM	−55°C to +125°C	Logic Inputs	−5V to +18V
Am6012ADC, Am6012DC	0°C to +70°C	Analog Current Outputs	−8V to +12V
Storage Temperature	−65°C to +125°C	Reference Inputs V_{14}, V_{15}	V− to V+
Lead Temperature (Soldering, 60 sec)	300°C	Reference Input Differential Voltage (V_{14} to V_{15})	±18V
		Reference Input Current (I_{14})	1.25mA

ELECTRICAL CHARACTERISTICS

These specifications apply for $V_+ = +15V$, $V_- = -15V$, $I_{REF} = 1.0mA$, over the operating temperature range unless otherwise specified.

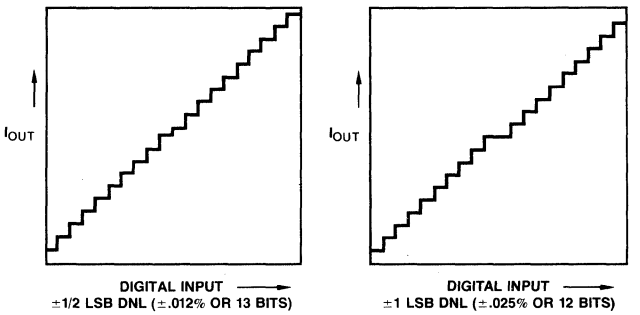
				Am6012A			Am6012			
Parameter	**Description**		**Test Conditions**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Units**
	Resolution			12	12	12	12	12	12	Bits
	Monotonicity			12	12	12	12	12	12	Bits
D.N.L.	Differential Nonlinearity		Deviation from ideal step size	–	–	±.012	–	–	±.025	%FS
				13	–	–	12	–	–	Bits
N.L.	Nonlinearity		Deviation from ideal straight line	–	–	±.05	–	–	±.05	%FS
I_{FS}	Full Scale Current		V_{REF} = 10.000V $R_{14} = R_{15}$ = 10.000kΩ T_A = 25°C	3.967	3.999	4.031	3.935	3.999	4.063	mA
TCI_{FS}	Full Scale Tempco			–	±5	±20	–	±10	±40	ppm/°C
				–	±.0005	±.002		±.001	±.004	%FS/°C
V_{OC}	Output Voltage Compliance		D.N.L. Specification guaranteed over compliance range R_{OUT} > 10 megohms typ.	−5	–	+10	−5	–	+10	Volts
I_{FSS}	Full Scale Symmetry		$I_{FS} - \overline{I_{FS}}$	–	±0.2	±1.0	–	±0.4	±2.0	μA
I_{ZS}	Zero Scale Current			–	–	0.10	–	–	0.10	μA
t_S	Settling Time		To ±1/2 LSB, all bits ON or OFF, T_A = 25°C	–	250	500	–	250	500	nsec
t_{PLH} t_{PHL}	Propagation Delay – all bits		50% to 50%	–	25	50	–	25	50	nsec
C_{OUT}	Output Capacitance			–	20	–	–	20	–	pF
V_{IL}	Logic Input Levels	Logic "0"		–	–	0.8	–	–	0.8	Volts
V_{IH}		Logic "1"		2.0	–	–	2.0	–	–	
I_{IN}	Logic Input Current		V_{IN} = −5 to +18V	–	–	40	–	–	40	μA
V_{IS}	Logic Input Swing		V− = −15V	−5	–	+18	−5	–	+18	Volts
I_{REF}	Reference Current Range			0.2	1.0	1.1	0.2	1.0	1.1	mA
I_{15}	Reference Bias Current			0	−0.5	−2.0	0	−0.5	−2.0	μA
dI/dt	Reference Input Slew Rate		$R_{14(eq)}$ = 800Ω CC = 0pF	4.0	8.0	–	4.0	8.0	–	mA/μs
$PSSI_{FS+}$	Power Supply Sensitivity		V+ = +13.5V to +16.5V, V− = −15V	–	±.00005	±.001	–	±0.0005	±.001	%FS/%
$PSSI_{FS-}$			V− = −13.5V to −16.5V, V+ = +15V	–	±.00025	±.001	–	±.00025	±.001	
V+	Power Supply Range		V_{OUT} = 0V	4.5	–	18	4.5	–	18	Volts
V−				−18	–	−10.8	−18	–	−10.8	
I+	Power Supply Current		V+ = +5V, V− = −15V	–	5.7	8.5	–	5.7	8.5	mA
I−				–	−13.7	−18.0	–	−13.7	−18.0	
I+			V+ = +15V, V− = −15V		5.7	8.5	–	5.7	8.5	
I−				–	−13.7	−18.0	–	−13.7	−18.0	
P_D	Power Dissipation		V+ = +5V, V− = −15V	–	234	312	–	234	312	mW
			V+ = +15V, V− = −15V	–	291	397	–	291	397	

ACCURACY SPECIFICATIONS

The design of the Am6012 emphasizes differential linearity which is a measure of the uniformity of each step in the transfer characteristic. The circuit design, described in greater detail on page 6, requires resistor matching and tracking tolerances of 8 times lower than that of previous designs to achieve and maintain monotonicity over temperature. This advantage has been used in the Am6012A to provide 13-bit differential nonlinearity over temperature, a level of performance not generally available in previous designs, even when using thin film resistors.

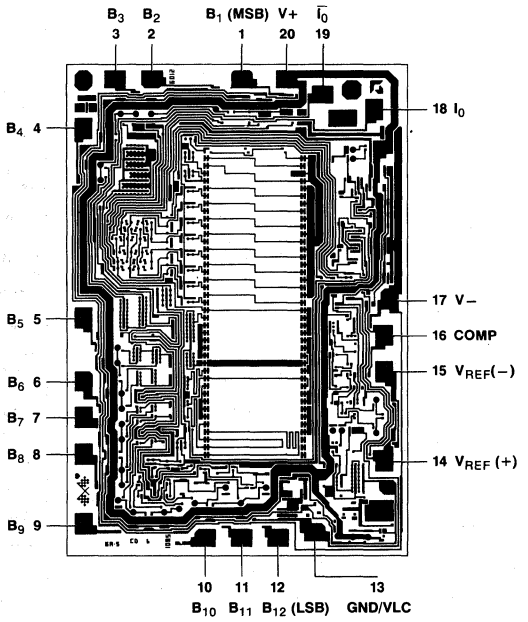
The figures illustrate that ±1/2 LSB (13-bit) differential nonlinearity guarantees a converter with 4096 distinct output levels. ±1 LSB D.N.L. guarantees monotonicity, so that when the input code is increased the output will not decrease. Note that nonlinearity, or deviation from an ideal straight line through zero and full scale, cannot be visually determined from the figures. In most applications, 12-bit resolution and differential linearity are more important than linearity. This is especially true in video and graphics, where the human eye has difficulty discerning nonlinearity of less than 5%.

DIFFERENTIAL NONLINEARITY
WORST CASE AT TEMPERATURE EXTREME



LIC-848

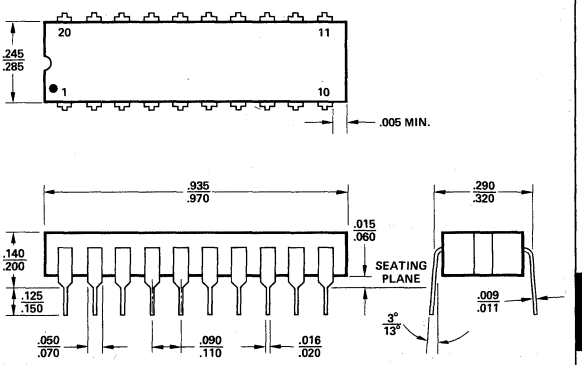
Metallization and Pad Layout



DIE SIZE 0.093" X 0.134"

PHYSICAL DIMENSIONS
Dual-In-Line

20-Pin Hermetic

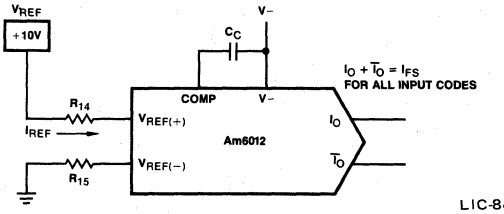


APPLICATION HINTS:

1. Reference current and reference resistor.

There is a 1 to 4 scale factor between the reference current (I_{REF}) and the full scale output current (I_{FS}). If V_{REF} = +10V and I_{FS} = 4mA, the value of the R_{14} is:

$$R_{14} = \frac{4 \times 10\ \text{Volt}}{4\text{mA}} = 10\text{k}\Omega \qquad R_{14} = R_{15}$$



LIC-849

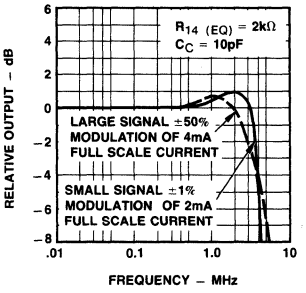
2. Reference amplifier compensation.

For AC reference applications, a minimum value compensation capacitor (C_C) is normally used. The value of this capacitor depends on the equivalent resistance at pin 14. The values to maximize bandwidth without oscillation are as follows:

MINIMUM SIZE COMPENSATION CAPACITOR (I_{FS} = 4mA, I_{REF} = 1.0mA)

R_{14} (EQ) (kΩ)	C_C (pF)
10	50
5	25
2	10
1	5
.5	0

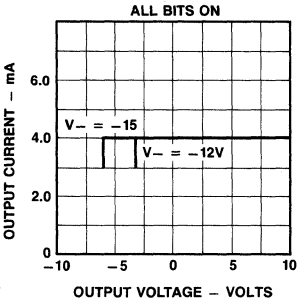
Reference Amplifier Frequency Response



Note: A 0.01μF capacitor is recommended for fixed reference operation.

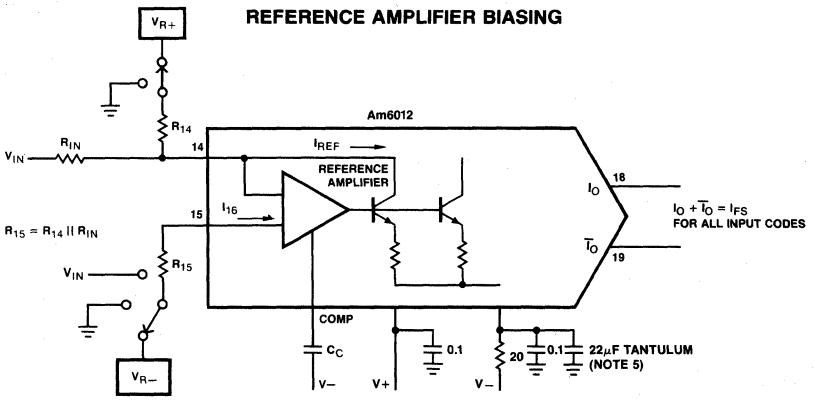
LIC-850

Output Voltage Compliance



LIC-851

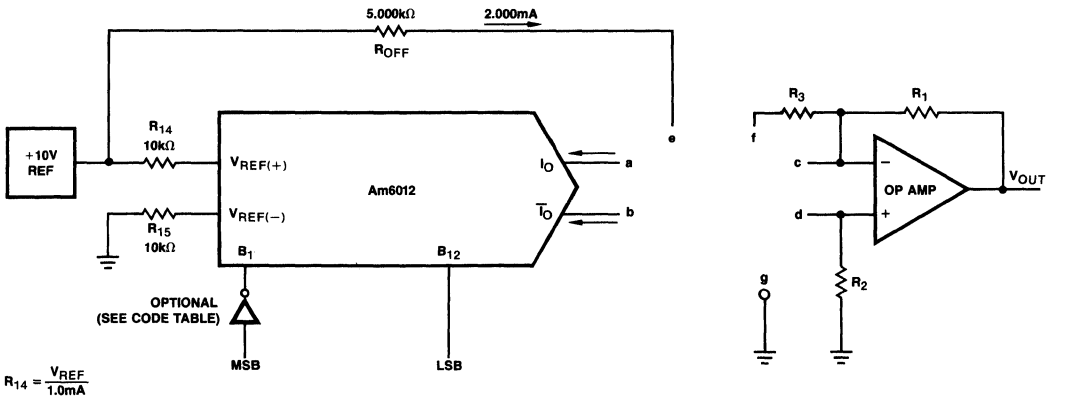
REFERENCE AMPLIFIER BIASING



LIC-852

Reference Configuration	R_{14}	R_{15}	R_{IN}	C_C	I_{REF}
Positive Reference	V_{R+}	0V	N/C	.01μF	V_{R+}/R_{14}
Negative Reference	0V	V_{R-}	N/C	.01μF	$-V_{R-}/R_{14}$
Lo Impedance Bipolar Reference	V_{R+}	0V	V_{IN}	(Note 1)	$(V_{R+}/R_{14}) + (V_{IN}/R_{IN})$ (Note 2)
Hi Impedance Bipolar Reference	V_{R+}	V_{IN}	N/C	(Note 1)	$(V_{R+} - V_{IN})/R_{14}$ (Note 3)
Pulsed Reference (Note 4)	V_{R+}	0V	V_{IN}	No Cap	$(V_{R+}/R_{14}) + (V_{IN}/R_{IN})$

Notes:
1. The compensation capacitor is a function of the impedance seen at the $+V_{REF}$ input and must be at least C = 5pF x $R_{14(eq)}$ in kΩ. For R_{14} < 800Ω no capacitor is necessary.
2. For negative values of V_{IN}, V_{R+}/R_{14} must be greater than $-V_{IN}$ Max/R_{IN} so that the amplifier is not turned off.
3. For positive values of V_{IN}, V_{R+} must be greater than V_{IN} Max so the amplifier is not turned off.
4. For pulsed operation, V_{R+} provides a DC offset and may be set to zero in some cases. The impedance at pin 14 should be 800Ω or less.
5. For optimum settling time, decouple V– with 20Ω and bypass with 22μF tantulum capacitor.



CODE FORMAT		CONNECTIONS	OUTPUT SCALE	MSB B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	LSB B12	I_o (mA)	$\overline{I_o}$ (mA)	V_{OUT}
UNIPOLAR	Straight binary; one polarity with true input code, true zero output.	a-c b-g R1 = R2 = 2.5K	Positive full scale	1	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	9.9976
			Positive full scale − LSB	1	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9951
			Zero scale	0	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	.0000
	Complementary binary; one polarity with complementary input code, true zero output.	a-g b-c R1 = R2 = 2.5K	Positive full scale	0	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	9.9976
			Positive full scale − LSB	0	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	9.9951
			Zero scale	1	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	.0000
SYM-METRICAL OFFSET	Straight offset binary; offset half scale, symmetrical about zero, no true zero output.	a-c b-d f-g R1 = R3 = 2.5K R2 = 1.25K	Positive full scale	1	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	9.9976
			Positive full scale − LSB	1	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9927
			(+) Zero scale	1	0	0	0	0	0	0	0	0	0	0	0	2.000	1.999	.0024
			(−) Zero scale	0	1	1	1	1	1	1	1	1	1	1	1	1.999	2.000	−.0024
			Negative full scale − LSB	0	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	−9.9927
			Negative full scale	0	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	−9.9976
	1's complement; offset half scale, symmetrical about zero, no true zero output MSB complemented (need inverter at B1).	a-c b-d f-g R1 = R3 = 2.5K R2 = 1.25K	Positive full scale	0	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	9.9976
			Positive full scale − LSB	0	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9927
			(+) Zero scale	0	0	0	0	0	0	0	0	0	0	0	0	2.000	1.999	.0024
			(−) Zero scale	1	1	1	1	1	1	1	1	1	1	1	1	1.999	2.000	−.0024
			Negative full scale − LSB	1	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	−9.9927
			Negative full scale	1	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	−9.9976
OFFSET WITH TRUE ZERO	Offset binary; offset half scale, true zero output.	e-a-c b-g R1 = R2 = 5K	Positive full scale	1	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	9.9951
			Positive full scale − LSB	1	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9902
			+ LSB	1	0	0	0	0	0	0	0	0	0	0	1	2.001	1.998	.0049
			Zero Scale	1	0	0	0	0	0	0	0	0	0	0	0	2.000	1.999	.000
			− LSB	0	1	1	1	1	1	1	1	1	1	1	1	1.999	2.000	−.0049
			Negative full scale + LSB	0	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	−9.9951
			Negative full scale	0	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	−10.000
	2's complement; offset half scale true zero output MSB complemented (need inverter at B1).	e-a-c b-g R1 = R2 = 5K	Positive full scale	0	1	1	1	1	1	1	1	1	1	1	1	3.999	.006	9.9951
			Positive full scale − LSB	0	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9902
			+1 LSB	0	0	0	0	0	0	0	0	0	0	0	1	2.001	1.998	.0049
			Zero scale	0	0	0	0	0	0	0	0	0	0	0	0	2.000	1.999	.000
			−1 LSB	1	1	1	1	1	1	1	1	1	1	1	1	1.999	2.000	−0.049
			Negative full scale + LSB	1	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	−9.9951
			Negative full scale	1	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	−10.000

ADDITIONAL CODE MODIFICATIONS

1. Any of the offset binary codes may be complemented by reversing the output terminal pair.

SEGMENTED DAC DESIGN INFORMATION

The design of a 12-bit D/A converter has traditionally required precision thin film resistors, a trimming method, and a binarily weighted ladder network. The Am6012 is a 12-bit DAC which uses diffused resistors and requires no trimming, cutting, blowing, or zapping to guarantee monotonicity for all grades over the temperature range. A proprietary design technique, departing from the traditional R-2R approach used in virtually all high speed high resolution converters, provides inherent monotonicity and differential linearity as high as 13 bits. This guarantees a more uniform step size over the temperature range than available trimmed 12-bit converters. The converter's performance is immune to variations in temperature, time, process, and mechanical stress. The circuit also features differential high compliance current outputs, wide supply range, and a multiplying reference input.

In most converter applications, uniform step size is more important than conformance to an ideal straight line. Most 12-bit converters are used for high resolution rather than high linearity, since few transducers are more linear than ±0.1%. All classic binarily weighted converters require ±1/2 LSB (±.012%) linearity in order to guarantee monotonicity, which requires very tight resistor matching and tracking. This new circuit uses conventional bipolar processing to achieve high differential linearity and monotonicity without requiring correspondingly high linearity, or conformance to an ideal straight line.

One design approach which provides monotonicity without requiring high linearity is the MOS switch-resistor string. This circuit is actually a full complement to a current switched R-2R DAC since it is slower, has a voltage output, and if implemented at the 12-bit level would use 4096 low tolerance resistors rather than a minimum number of high tolerance resistors as in the R-2R network. Its lack of speed and density for 12 bits are its drawbacks.

The technique used in the Am6012 combines the advantages of both the R-2R and 2^nR approaches. It is inherently monotonic, fast, and uses untrimmed resistors which are actually fewer in number than the classic R-2R ladder.

In order to properly describe the new design technique, the standard R-2R ladder approach used in previous 12-bit DAC's will first be discussed. Figure 1 shows the twelve-bit currents which are used in all possible binary combinations to generate 4096 analog output levels. The resistor ladder tolerance is most critical for the major carry, where the 11 least significant bits turn off and the most significant bit turns on. If the MSB is more than 1μA low, or −.05%, the converter will be nonmonotonic. Table 1 shows the maximum tracking error which can be allowed over a 100°C range to maintain monotonicity, which is ±1 LSB D.N.L. Achieving ±1/2 LSB differential nonlinearity is especially difficult since it requires a tracking temperature coefficient of ±1.25 ppm/°C.

Figure 2 shows the transfer characteristic for the new technique, called the segmented DAC. The 4096 output levels are composed of 8 groups of 512 steps each. Each step group is generated by a 9-bit DAC, and each of the segment slopes is determined by one of 8 equal current sources, as shown in Figure 3. The resistors which determine monotonocity are in the 9-bit DAC. The major carry of the 9-bit DAC is repeated in each of the 8 segments, and requires eight times lower initial resistor accuracy and tracking to maintain a given differential nonlinearity over temperature.

The operation of the segmented DAC may be visualized by assuming an input code of all zeroes. The first segment current I_O is divided into 512 levels by the 9-bit multiplying DAC and fed to the output, I_{OUT}. As the input code increases, a new segment current is selected for each 512 counts. The previous segment is fed to output I_{OUT} where the new step group is added to it, thus ensuring monotonicity independent of segment resistor values. All higher order segments feed $\overline{I_{OUT}}$.

At each segment endpoint, monotonicity is assured because no critical resistor tolerances are involved. For example, at the midpoint of the transfer characteristic, as shown in Figure 2, $I_{4,0}$ is actually generated by the same segment resistor as $I_{3,511}$ and has been incremented by the remainder current of the 9-bit DAC.

In the segmented DAC, the precision of the 8 main resistors determines linearity only. The influence of each of these resistors on linearity is four times lower than that of the MSB resistor in an R-2R DAC. Hence, assuming the same resistor tolerances for both, the linearity of the segmented approach would actually be higher than that of an R-2R design.

The step generator or 9-bit DAC is composed of a master and a slave ladder. The slave ladder generates the four least significant bits from the remainder of the master ladder by active current splitting utilizing scaled emitters. This saves ladder resistors and greatly reduces the range of emitter scaling required in the 9-bit DAC. All current switches in the step generator are high speed fully differential switches which are capable of switching low currents at high speed. This allows the use of a binary scaled network all the way to the least significant bit which saves power and simplifies the circuitry.

Diffused resistors have advantages over thin film resistors beyond simple economy and bipolar process compatibility. The resistors are fabricated in single crystal rather than amorphous material which gives them better long term stability and tracking and much higher moisture resistance. They are diffused at 1000°C and so are resistant to changes in value due to thermal and chemical causes. Also, no burn-in is required for stability. The contact resistance between aluminum and silicon is more predictable than between aluminum and an amorphous thin film, and no sandwich metals are required to enhance or protect the contact or limit alloying. The initial match between two diffused resistors is similar to that of thin film since both are defined by photomasks and chemical etching. Since the resistors are not trimmed or altered after fabrication, their tracking and long term characteristics are not degraded.

TABLE 1
RESISTOR SPECIFICATIONS

Ladder Type	No. of Resistors	Initial Matching Required for ±1 LSB DNL (%)	Tracking Required for ±1 LSB DNL (ppm/°C)		Tracking Req'd. for ±1/2 LSB DNL (ppm/°C)
			0 Initial DNL	1/2 LSB Initial DNL	1/4 LSB Initial DNL
Straight R-2R	37	±.05	5	2.5	1.25
Segmented 3 Bits + 9 Bits	24	±.4	40	20	10

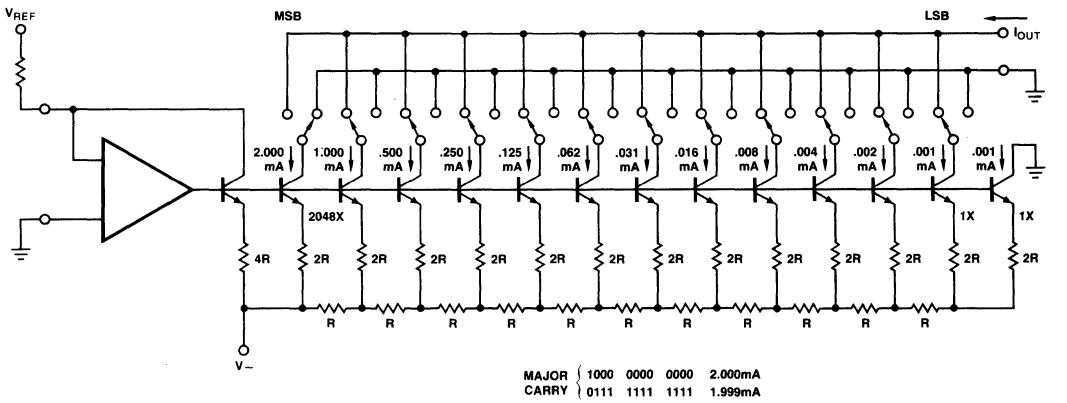


Figure 1. Traditional R-2R D/A Converter.

LIC-854

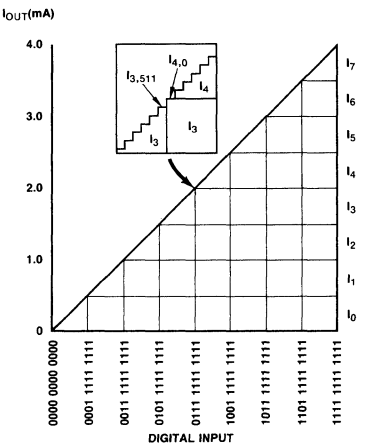


Figure 2. Transfer Characteristic of Segmented Design.

LIC-855

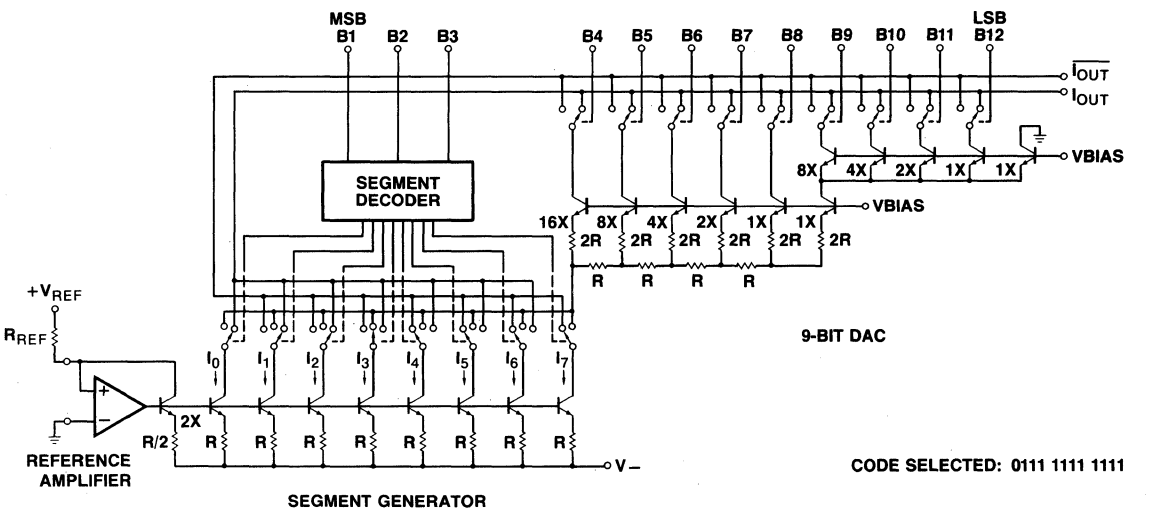
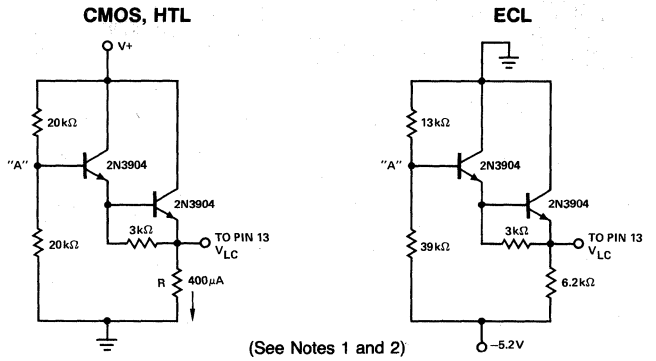


Figure 3. Segmented DAC Functional Diagram Used in Am6012.

LIC-856

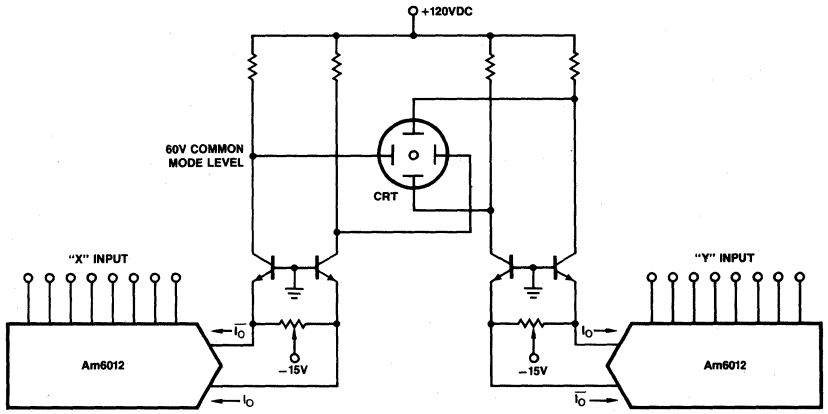


Notes:
1. Set the voltage "A" to the desired logic input switching theshold.
2. Allowable range of logic threshold is typically −5V to +13.5V when operating the DAC on ±15V supplies.

Interfacing Circuits for ECL, CMOS, HTL Logic Inputs

LIC-857

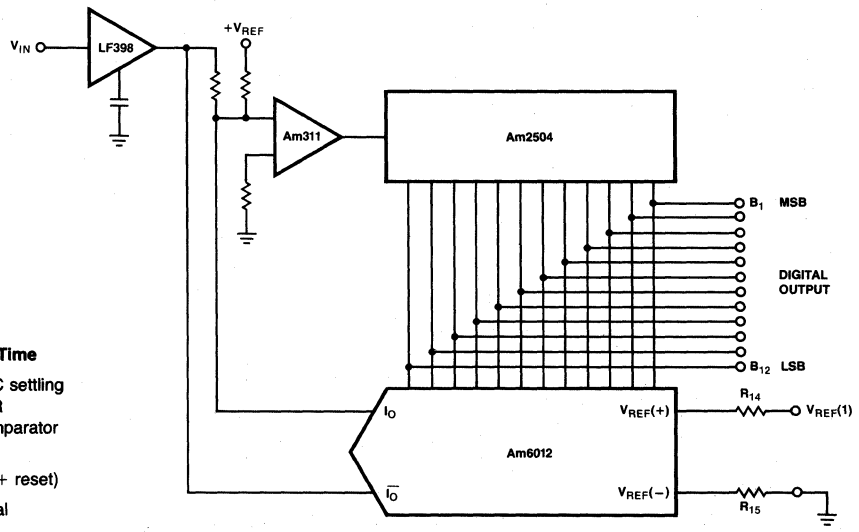
CRT DISPLAY DRIVER



Notes: 1. Full differential drive lowers power supply voltage.
2. Eliminates inverting amplifiers and transformers.
3. Independent beam centering controls.

LIC-858

HIGH-SPEED 12-BIT A/D CONVERTER



Conversion Time

300nsec DAC settling
50nsec SAR
300nsec Comparator
$\overline{650}$nsec
x13 (12 bits + reset)
$\overline{8.5}$μsec Total

LIC-859

Am6080

Microprocessor System Compatible 8-Bit High Speed Multiplying D/A Converter

DISTINCTIVE CHARACTERISTICS

- 8-Bit D/A with 8-Bit input data latch
- Compatible with most popular microprocessors including the AmZ8000 and the Am2900 Families
- Write, Chip Select and Data Enable logic on chip
- DAC appears as memory location to microprocessor
- MSB inversion under logic control
- Differential current output
- Choice of 6 coding formats
- Fast settling current output −160ns
- Nonlinearity to ±0.1% max over temperature range
- Full scale current pre-matched to ±1 LSB
- High output impedance and voltage compliance
- Low full scale current drift − ±5ppm/°C
- Wide range multiplying capability −2.0MHz bandwidth
- Direct interface to TTL, CMOS, NMOS
- High speed data latch − 80ns min write time

GENERAL DESCRIPTION

The Am6080 is a monolithic 8-bit multiplying Digital-to-Analog converter with an 8-bit data latch, chip select and other control signal lines which allow direct interface with microprocessor buses.

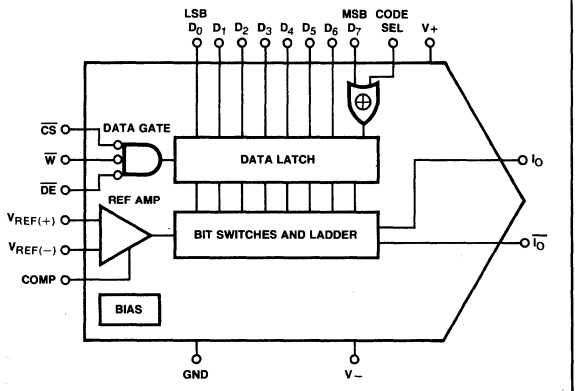
The converter allows a choice of 6 different coding formats. The most significant bit (D_7) can be inverted or non-inverted under the control of the code select input. The code control also provides a zero differential current output for 2's complement coding. A high voltage compliance, complementary current output pair is provided. The data latch is very high speed which makes the Am6080 capable of interfacing with high speed microprocessors.

Monotonic multiplying performance is maintained over a more than 40 to 1 reference current range. Matching within ±1 LSB between reference and full scale current eliminates the need for full scale trimming in most applications.

The Am6080 guarantees full 8-bit monotonicity. Nonlinearities as tight as 0.1% over the entire operating temperature range are available. Device performance is essentially unchanged over the full power supply voltage and temperature range.

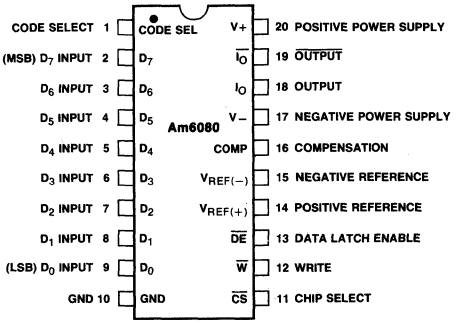
Applications for the Am6080 include microprocessor compatible data acquisition systems and data distribution systems, 8-bit A/D converters, servo-motor and pen drivers, waveform generators, programmable attenuators, analog meter drivers, programmable power supplies, CRT display drivers and high speed modems.

EQUIVALENT CIRCUIT



LIC-061

CONNECTION DIAGRAM
Top View



Pin 1 marked for orientation.

LIC-062

ORDERING INFORMATION

Package Type	Temperature Range	Nonlinearity	Order Number
Hermetic DIP	−55°C to +125°C	.1%	AM6080ADM
		.19%	AM6080DM
Hermetic DIP	0°C to +70°C	.1%	AM6080ADC
		.19%	AM6080DC
Molded DIP		.1%	AM6080APC
		.19%	AM6080PC

MAXIMUM RATINGS

Operating Temperature		Power Supply Voltage	±18V
Am6080ADM, Am6080DM	−55°C to +125°C	Logic Inputs	−5V to +18V
Am6080ADC, Am6080DC Am6080APC, Am6080PC	0°C to +70°C	Analog Current Outputs	−12V to +18V
		Reference Inputs ($V_{14}V_{15}$)	V− to V+
Storage Temperature	−65°C to +150°C	Reference Input Differential Voltage (V_{14} to V_{15})	±18V
Lead Temperature (Soldering, 60 sec)	300°C	Reference Input Current (I_{14})	1.25mA

ELECTRICAL CHARACTERISTICS

These specifications apply for $V_+ = +5V$, $V_- = -15V$, $I_{REF} = 0.5mA$, over the operating temperature range unless otherwise specified. Output characteristics refer to all outputs.

				Am6080A			Am6080			
Parameter	**Description**		**Conditions**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Unit**
	Resolution			8	8	8	8	8	8	bits
	Monotonicity			8	8	8	8	8	8	bits
D.N.L.	Differential Nonlinearity			–	–	±0.19	–	–	±0.39	%FS
N.L.	Nonlinearity			–	–	±0.1	–	–	±0.19	%FS
I_{FS}	Full Scale Current		$V_{REF} = 10.000V$ $R_{14} = R_{15} = 20.000k\Omega$ $T_A = 25°C$	1.984	1.992	2.000	1.976	1.992	2.008	mA
TCI_{FS}	Full Scale Tempco			–	±5	±20	–	±10	±40	ppm/°C
				–	.0005	±.002	–	.001	±.004	%FS/°C
V_{OC}	Output Voltage Compliance			−10	–	+18	−10	–	+18	Volts
I_{FSS}	Full Scale Symmetry		$I_{FS1} - \overline{I_{FS1}}$	–	±0.1	±1.0	–	±0.2	±2.0	μA
I_{ZS}	Zero Scale Current			–	0.01	0.4	–	0.01	0.8	μA
I_{RR}	Reference Current Range		V− = −5V	0	0.5	0.55	0	0.5	0.55	mA
			V− = −15V	0	0.5	1.1	0	0.5	1.1	
V_{IL}	Logic Input Levels	Logic "0"		–	–	0.8	–	–	0.8	Volts
V_{IH}		Logic "1"		2.0	–	–	2.0	–	–	
I_{IN}	Logic Input Current		V_{IN} = −5V to +18V	–	–	40	–	–	40	μA
V_{IS}	Logic Input Swing		V− = −15V	−5	–	+18	−5	–	+18	Volts
I_{15}	Reference Bias Current			–	−0.5	−2.0	–	−0.5	−2.0	μA
dI/dt	Reference Input Slew Rate		$R_{14(EQ)} = 800\Omega$ CC = 0pF	4.0	8.0	–	4.0	8.0	–	mA/μs
$PSSI_{FS+}$	Power Supply Sensitivity		V+ = +4.5V to +5.5V, V− = −15V	–	±0.0003	±0.01	–	±0.0005	±0.01	%FS
$PSSI_{FS-}$			V− = −13.5V to − 16.5V, V+ = +5V	–	±0.0005	±0.01	–	±0.0005	±0.01	
V+	Power Supply Range		$I_{REF} = 0.5mA$, $V_{OUT} = 0V$	4.5	–	18	4.5	–	18	Volts
V−				−18	–	−4.5	−18	–	−4.5	
I+	Power Supply Current		V+ = +5V, V− = −5V	–	9.8	14.7	–	9.8	14.7	mA
I−				–	−7.4	−9.9	–	−7.4	−9.9	
I+			V+ = +5V, V− = −15V	–	9.8	14.7	–	9.8	14.7	
I−				–	−7.4	−9.9	–	−7.4	−9.9	
I+			V+ = +15V, V− = −15V		9.8	14.7	–	9.8	14.7	
I−				–	−7.4	−9.9	–	−7.4	−9.9	
P_D	Power Dissipation		V+ = +5V, V− = −5V	–	86	123	–	86	123	mW
			V+ = +5V, V− = −15V	–	160	222	–	160	222	
			V+ = +15V, V− = −15V	–	258	369	–	258	369	

Am6080 FUNCTIONAL PIN DESCRIPTION

Symbol **Function**

D_0-D_7 D_0-D_7 are the input bits 1-8 to the input data latch. Data is transferred to the data latch when $\overline{CS}$, $\overline{DE}$, and $\overline{W}$ are active and is latched when any of the enable signals go inactive.

$\overline{CS}$ Chip Select – This active low input signal enables the Am6080. Writing into the data latch occurs only when the device is selected.

$\overline{W}$ Write – This active low control signal enables the data latch when the $\overline{CS}$ and $\overline{DE}$ inputs are active.

$\overline{DE}$ Data Latch Enable – This active low input is used to enable the data latch. The $\overline{CS}$, $\overline{DE}$, and $\overline{W}$ must be active in order to write into the data latch.

CODE SEL Code Select – When CODE SEL = 0, the MSB (D_7) is inverted and 1 LSB balance current is added to the $\overline{I_O}$ output.

$V_{REF(+)}$ **$V_{REF(-)}$** Positive and negative reference voltage to the reference bias amplifier. These differential inputs allow the use of positive, negative and bipolar references.

COMP Compensation – Frequency compensating terminal for the reference amplifier.

I_O, $\overline{I_O}$ These are high impedance complementary current outputs. The sum of these currents is always equal to I_{FS}

FUNCTION TABLES

DATA LATCH CONTROL

$\overline{CS}$	$\overline{W}$	$\overline{DE}$	Data Latch
0	0	0	Transparent
X	X	1	Latched
X	1	X	Latched
1	X	X	Latched

X = Don't Care

CODE SELECT

CODE SEL	Function
0	MSB Inverted (Note 1)
1	MSB Non-inverted

Note 1. LSB balance current is added to the $\overline{I_O}$ output.

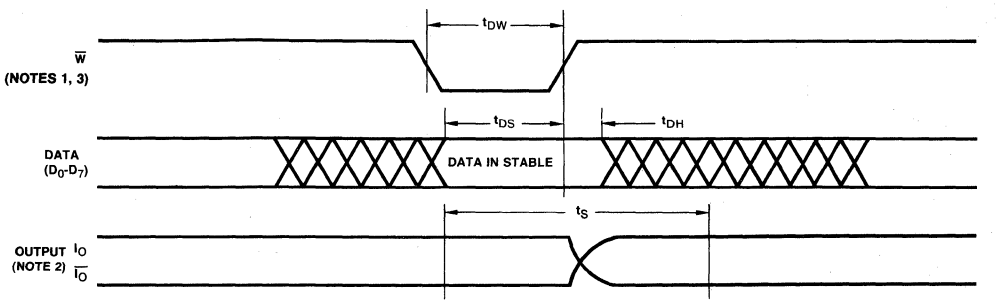
AC CHARACTERISTICS

$V_+ = +5V$, $V_- = -15V$, $I_{REF} = 0.5mA$, $R_L < 500\Omega$, $C_L < 15pF$ over the operating temperature range unless otherwise specified

Parameter	Description		Conditions	Commercial Temp. Grades Min.	Typ.	Max.	Military Temp. Grades Min.	Typ.	Max.	Unit
t_S	Settling Time, All Bits Switched		$T_A = 25°C$ Settling to ±½LSB		160			160		ns
t_{PLH}	Propagation Delay	Each bit	$T_A = 25°C$ 50% to 50%		80	160		80	160	ns
t_{PHL}		All bits switched			80	160		80	160	
t_{DH}	Data Hold Time		See timing diagram	10	−30		10	−30		ns
t_{DS}	Data Set Up Time		See timing diagram	80	35		100	35		ns
t_{DW}	Data Write Time		See timing diagram	80	35		100	35		ns

Notes: 1. t_{DW} is the overlap of $\overline{W}$ low, $\overline{CS}$ low, and $\overline{DE}$ low. All three signals must be low to enable the latch. Any signal going inactive latches the data.
2. t_S is measured with the latches open from the time the data becomes stable on the inputs to the time when the outputs are settled to within ±1/2 LSB. All bits switched on or off.
3. The internal time delays from $\overline{CS}$, $\overline{W}$ and $\overline{DE}$ inputs to the enabling of the latches are all equal.

TIMING DIAGRAM



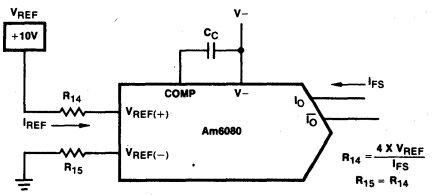
LIC-063

APPLICATION HINTS:

1. Reference current and reference resistor.

There is a 1 to 4 scale up between the reference current (I_{REF}) and the full scale output current (I_{FS}). If V_{REF} = +10V and I_{FS} = 2mA, the value of the R_{14} is:

$$R_{14} = \frac{4 \times 10\ \text{Volt}}{2\text{mA}} = 20\text{K}\Omega$$



LIC-064

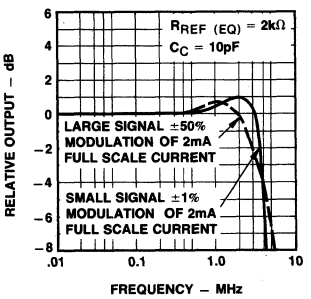
2. Reference amplifier compensation.

For AC reference applications, a minimum value compensation capacitor (C_C) is normally used. The value of this capacitor depends on R_{15}. The minimum values to maximize bandwidth without oscillation are as follows:

Table 2
Compensation Capacitor
(I_{FS} = 2mA, I_{REF} = 0.5mA)

R_{REF} (kΩ)	C_C (pF)
10	50
5	25
2	10
1	5
.5	0

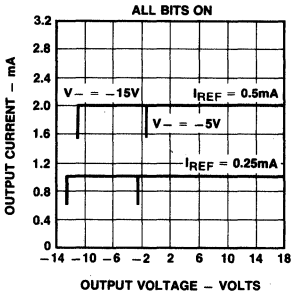
Reference Amplifier Frequency Response



A 0.01μF capacitor is recommended for the fixed reference operation.

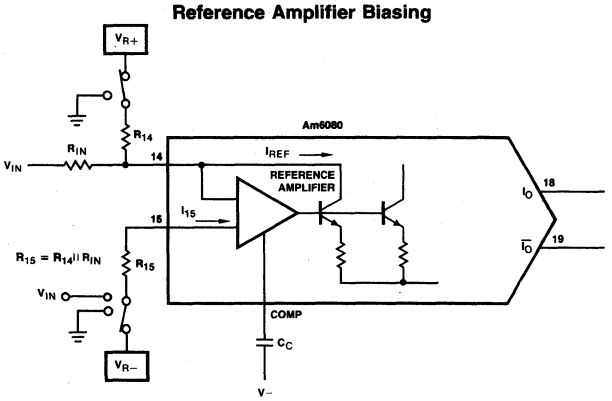
LIC-065

Output Voltage Compliance



LIC-066

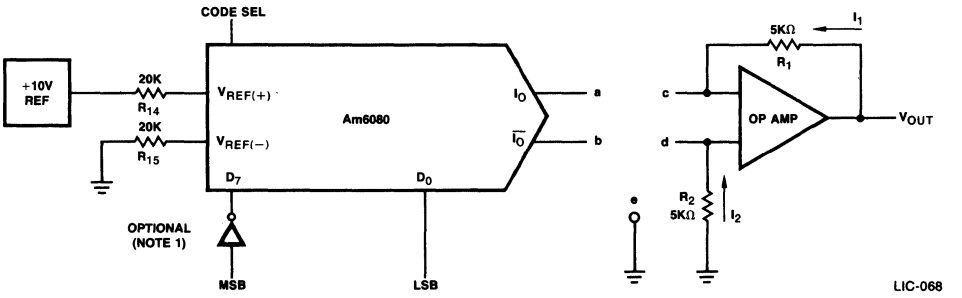
Reference Amplifier Biasing



LIC-067

Reference Configuration	R_{14}	R_{15}	R_{IN}	C_C	I_{REF}
Positive Reference	V_{R+}	0V	N/C	.01μF	V_{R+}/R_{14}
Negative Reference	0V	V_{R-}	N/C	.01μF	$-V_{R-}/R_{14}$
Lo Impedance Bipolar Reference	V_{R+}	0V	V_{IN}	(Note 1)	$(V_{R+}/R_{14}) + (V_{IN}/R_{IN})$ (Note 2)
Hi Impedance Bipolar Reference	V_{R+}	V_{IN}	N/C	(Note 1)	$(V_{R+} - V_{IN})/R_{14}$ (Note 3)
Pulsed Reference (Note 4)	V_{R+}	0V	V_{IN}	No Cap	$(V_{R+}/R_{14}) + (V_{IN}/R_{IN})$

Notes:
1. The compensation capacitor is a function of the impedance seen at the $+V_{REF}$ input and must be at least C = 5pFX $R_{14(eq)}$ in kΩ. For R_{14} < 800Ω no capacitor is necessary.
2. For negative values of V_{IN}, V_{R+}/R_{14} must be greater than $-V_{IN}$ Max/R_{IN} so that the amplifier is not turned off.
3. For positive values of V_{IN}, V_{R+} must be greater than V_{IN} Max so the amplifier is not turned off.
4. For pulsed operation, V_{R+} provides a DC offset and may be set to zero in some cases. The impedance at pin 14 should be 800Ω or less and an additional resistor may be connected from pin 14 to ground to lower the impedance.



CODE FORMAT		CODE SEL	CONNECTIONS	OUTPUT SCALE	OUT SEL	MSB D7	D6	D5	D4	D3	D2	D1	LSB D0	I_1 (mA)	I_2 (mA)	V_{OUT}
UNIPOLAR	Straight binary: one polarity with true input code, true zero output.	1	a-c b-e	Positive full scale	X	1	1	1	1	1	1	1	1	1.992	0	9.960
				Positive full scale − LSB	X	1	1	1	1	1	1	1	0	1.984	0	9.920
				Zero scale	X	0	0	0	0	0	0	0	0	.000	0	.000
	Complementary binary: one polarity with complementary input code, true zero output.	1	a-e b-c	Positive full scale	X	0	0	0	0	0	0	0	0	1.992	0	9.960
				Positive full scale − LSB	X	0	0	0	0	0	0	0	1	1.984	0	9.920
				Zero scale	X	1	1	1	1	1	1	1	1	.000	0	.000
SYMMETRICAL OFFSET	Straight offset binary: offset half scale, symmetrical about zero, no true zero output.	1	a-c b-d	Positive full scale	X	1	1	1	1	1	1	1	1	1.992	.000	9.960
				Positive full scale − LSB	X	1	1	1	1	1	1	1	0	1.984	.008	9.880
				(+) Zero scale	X	1	0	0	0	0	0	0	0	1.000	.992	.040
				(−) Zero scale	X	0	1	1	1	1	1	1	1	.992	1.000	−.040
				Negative full scale − LSB	X	0	0	0	0	0	0	0	1	.008	1.984	−9.880
				Negative full scale	X	0	0	0	0	0	0	0	0	.000	1.992	−9.960
	1's complement: offset half scale, symmetrical about zero, no true zero output MSB complemented. (need inverter at D_7)	1 (Note 1)	a-c b-d	Positive full scale	X	0	1	1	1	1	1	1	1	1.992	.000	9.960
				Positive full scale − LSB	X	0	1	1	1	1	1	1	0	1.984	.008	9.980
				(+) Zero scale	X	0	0	0	0	0	0	0	0	1.000	.992	.040
				(−) Zero scale	X	1	1	1	1	1	1	1	1	.992	1.000	−.040
				Negative full scale − LSB	X	1	0	0	0	0	0	0	1	.008	1.984	−9.880
				Negative full scale	X	1	0	0	0	0	0	0	0	.000	1.992	−9.960
OFFSET WITH TRUE ZERO	Offset binary: offset half scale, true zero output MSB complemented remainder add to I_O. (need inverter at D_7)	0 (Note 1)	a-c b-d	Positive full scale	X	1	1	1	1	1	1	1	1	1.992	.008	9.920
				Positive full scale − LSB	X	1	1	1	1	1	1	1	0	1.984	.016	9.840
				+ LSB	X	1	0	0	0	0	0	0	1	1.008	.992	.080
				Zero scale	X	1	0	0	0	0	0	0	0	1.000	1.000	.000
				− LSB	X	0	1	1	1	1	1	1	1	1.992	1.008	−.080
				Negative full scale + LSB	X	0	0	0	0	0	0	0	1	.008	1.992	−9.920
				Negative full scale	X	0	0	0	0	0	0	0	0	.000	2.000	−10.000
	2's complement: offset half scale true zero output MSB complemented.	0	a-c b-d	Positive full scale	X	0	1	1	1	1	1	1	1	1.992	.008	9.920
				Positive full scale − LSB	X	0	1	1	1	1	1	1	0	1.984	.016	9.840
				+1 LSB	X	0	0	0	0	0	0	0	1	1.008	.992	.080
				Zero scale	X	0	0	0	0	0	0	0	0	1.000	1.000	.000
				−1 LSB	X	1	1	1	1	1	1	1	1	.992	1.008	−.080
				Negative full scale + LSB	X	1	0	0	0	0	0	0	1	.008	1.992	−9.920
				Negative full scale	X	1	0	0	0	0	0	0	0	.000	2.000	−10.000

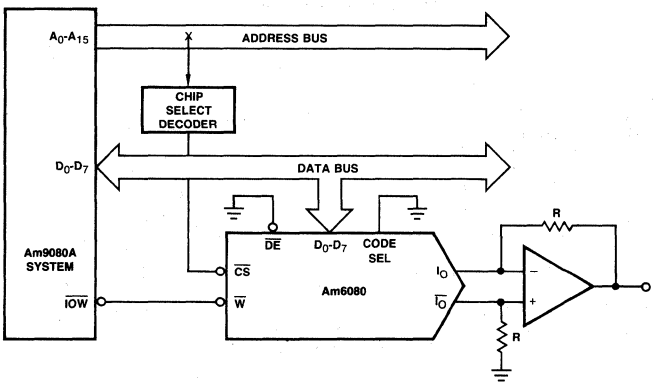
Note 1: An external inverter is necessary since the code select inverts the MSB and adds a 1 LSB balance current to $\overline{I_O}$. Only one of these features is desired for this code.

ADDITIONAL CODE MODIFICATIONS

1. Any of the offset binary codes may be complemented by reversing the output terminal pair.

SYSTEM APPLICATIONS

Am9080A DATA SYSTEM

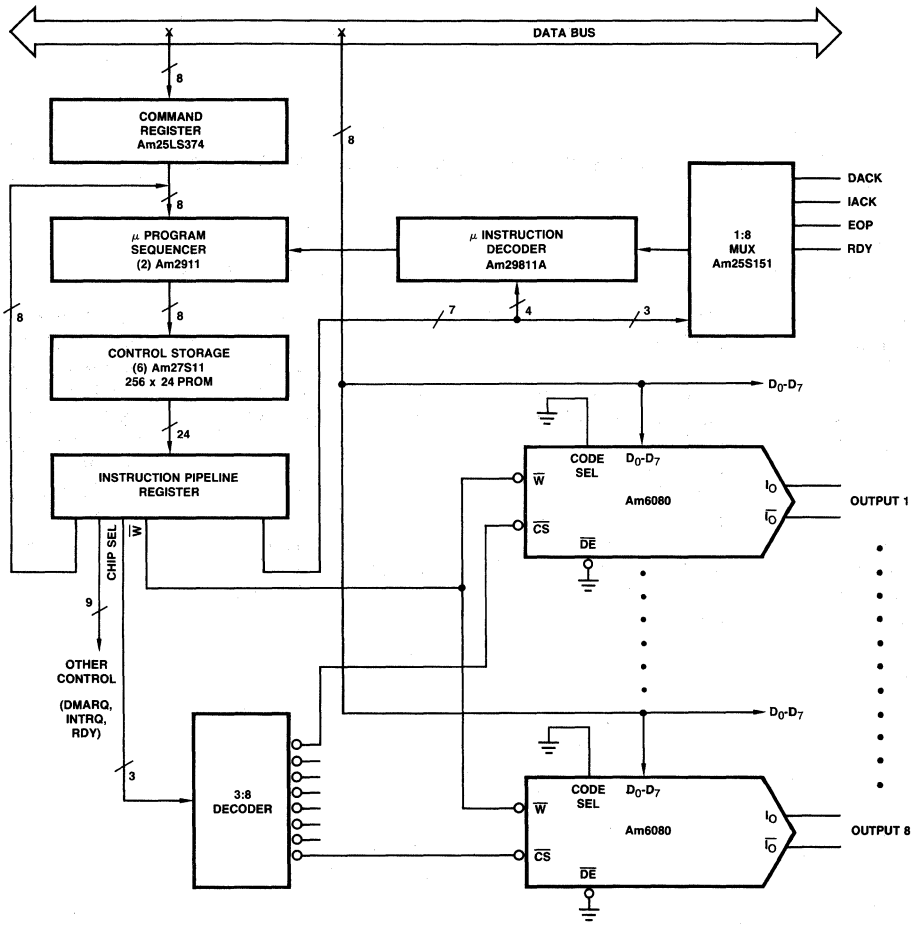


WRITING DATA INTO THE Am6080 (2's Complement)

```
PORT 1          :EQU OOH OUTPUT PORT ADDRESS
MOV A, M        :GET DATA FROM MEMORY
OUT 0 PORT1     :SEND DATA
```

LIC-069

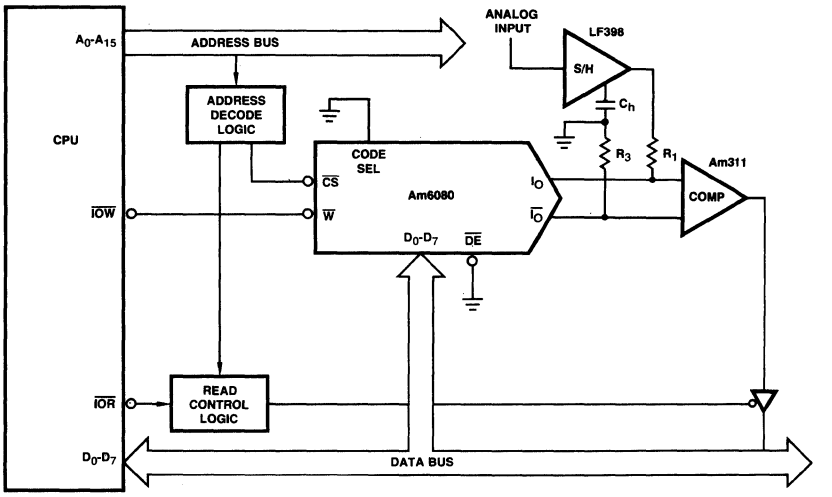
Am2900 DATA SYSTEM: MULTIPLE ANALOG INPUTS



LIC-070

SYSTEM APPLICATIONS (Cont.)

ANALOG/DIGITAL CONVERTER UNDER SOFTWARE CONTROL



LIC-071

Am9080A SOFTWARE FOR A/D CONVERSION USING Am6080.

```
SEQ     SOURCE STATEMENT
 0 PORT1   EQU 00H          ;6080 A/O PORT ADDRESS
 1 PORT3   EQU 02H          ;COMPARATOR ADDRESS
 2         ORG 3E50H
 3 START:  LXI SP,STAKS-16  ;INITIAL STAKS POINTER
 4 SAMPLE: CALL ADCON       ;CALL A/D CONVERSATION
 5         JMP SAMPLE       ;NEXT SAMPLE
 6 ADCON:  XRA A            :CLEAR ACC
 7         MOV D,A          ;CLEAR D REG
 8         STC              ;SET CARRY
 9         RAR              ;SET BIT 7 TO 1
10         MOV B,A          ;STORE TEST BIT AT B REGISTER
11 LOOP:   MOV E,A          ;STORE TEST WORD
12         OUT PORT1        ;OUTPUT TO A/D
```

```
SEQ     SOURCE STATEMENT
13         IN PORT3         ;INPUT FROM COMP
14         CRA A            ;SET SIGN FLAG
15         JM NEXT          ;IF SMALLER GO TO NEXT BIT
16         MOV D,E          ;SAVE RESULT
17 NEXT:   MOV A,B          ;GET NEXT TRIAL BIT
18         RAR              ;SHIFT RIGHT ONCE
19         RC               ;RETURN ON CARRY
20         MOV B,A          ;STORE TEST BIT
21         ADD D            ;ACCUMULATE RESULT
22         JMP LOOP         ;TRY NEXT BIT
23 STAKS:  DS 16
24         END START
```

Am6081

Microprocessor System Compatible 8-Bit High Speed Multiplying D/A Converter

DISTINCTIVE CHARACTERISTICS

- 8-Bit D/A with 8-Bit input data latch
- Compatible with most popular microprocessors including the AmZ8000 and the Am2900 Families
- Write, Chip Select and Data Enable logic on chip
- DAC appears as memory location to microprocessor
- MSB inversion under logic control
- Differential current outputs
- Output current mode multiplexer with logic selection
- 2-Bit status latch for output select and code select
- Choice of 8 coding formats
- Fast settling current output – 200ns
- Nonlinearity to ±0.1% max over temperature range
- Full scale current pre-matched to ±1 LSB
- High output impedance and voltage compliance
- Low full scale current drift – ±5ppm/°C
- Wide range multiplying capability –2.0MHz bandwidth
- Direct interface to TTL, CMOS, NMOS
- Output range selection with on chip multiplexer
- High speed data latch – 80ns min write time

GENERAL DESCRIPTION

The Am6081 is a monolithic 8-bit multiplying Digital-to-Analog converter with an 8-bit data latch, a 2-bit status latch, chip select and other control signal lines which allow direct interface with microprocessor buses.

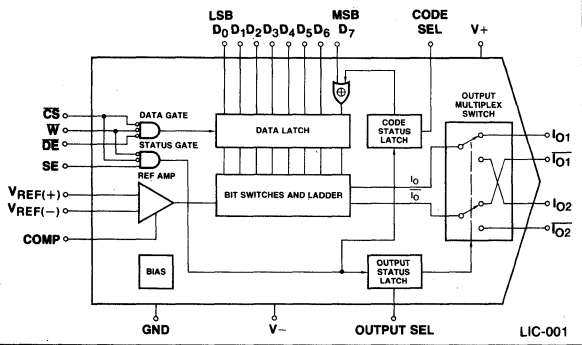
The converter allows a choice of 8 different coding formats. The most significant bit (D_7) can be inverted or non-inverted under the control of the code select input. The code control also provides a zero differential current output for 2's complement coding. A pair of high voltage compliance, dual complementary current output channels is provided and is selected by the output status command. The output multiplexer also allows analog bus connection of several converters, range or output load selection, and time-shared operation between D/A and A/D functions. The data and status latches are high speed which makes the Am6081 capable of interfacing with high speed microprocessors. The DE and SE control signals allow the data and status latches to be updated individually or simultaneously.

Monotonic multiplying performance is maintained over a more than 40 to 1 reference current range. Matching within ±1 LSB between reference and full scale current eliminates the need for full scale trimming in most applications.

The Am6081 guarantees full 8-bit monotonicity. Nonlinearities as tight as 0.1% over the entire operating temperature range are available. Device performance is essentially unchanged over the full power supply voltage and temperature range.

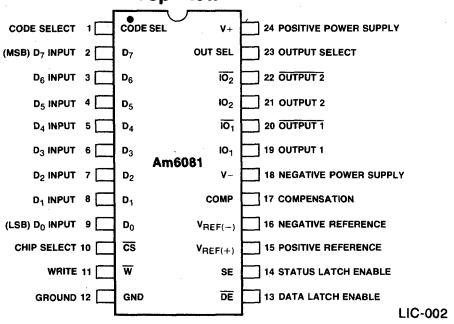
Applications for the Am6081 include microprocessor compatible data acquisition systems and data distribution systems, 8-bit A/D converters, servo-motor and pen drivers, waveform generators, programmable attenuators, analog meter drivers, programmable power supplies, CRT display drivers and high speed modems.

EQUIVALENT CIRCUIT



CONNECTION DIAGRAM

Top View



ORDERING INFORMATION

Package Type	Temperature Range	Nonlinearity	Order Number
Hermetic DIP	−55°C to +125°C	±.1%	AM6081ADM
		±.19%	AM6081DM
Hermetic DIP	0°C to +70°C	±.1%	AM6081ADC
		±.19%	AM6081DC
Molded DIP		±.1%	AM6081APC
		±.19%	AM6081PC

Am6081 FUNCTIONAL PIN DESCRIPTION

Symbol **Function**

$\overline{CS}$ Chip Select – This active low input signal enables the Am6081. Writing into the data or status latches occurs only when the device is selected.

$\overline{DE}$ Data Latch Enable – This active low input is used to enable the data latch. The $\overline{CS}$, $\overline{DE}$, and $\overline{W}$ must be active in order to write into the data latch.

SE Status Latch Enable – This active high input is used to enable the status latches. The $\overline{CS}$, SE, and $\overline{W}$ must be active in order to write into the status latches.

$\overline{W}$ Write – This active low control signal enables the data and status latches when the $\overline{CS}$, $\overline{DE}$, and SE inputs are active.

D_0-D_7 D_0-D_7 are the input bits 1-8 to the input data latch. Data is transferred to the data latch when $\overline{CS}$, $\overline{DE}$, and $\overline{W}$ are active and is latched when any of the enable signals go inactive.

CODE SEL Code Select – Input to the CODE SEL latch. The latch is transparent when $\overline{CS}$, SE and $\overline{W}$ are active and is latched when any of the above signals go inactive. When CODE SEL latch = 0, the MSB (D_7) is inverted and 1 LSB balance current is added to the $\overline{I_O}$ output.

OUT SEL Output Select – Input to the OUT SEL latch. The latch is transparent when $\overline{CS}$, SE and $\overline{W}$ are active and is latched when any of the above signals go inactive. When the OUT SEL latch is low, the channel 1 output pair (I_{O1}, $\overline{I_{O1}}$) is selected. When the OUT SEL latch is high, the channel 2 output pair (I_{O2}, $\overline{I_{O2}}$) is selected.

$V_{REF(+)}$ **$V_{REF(-)}$** Positive and negative reference voltage to the reference bias amplifier. These differential inputs allow the use of positive, negative and bipolar references.

COMP Compensation – Frequency compensating terminal for the reference amplifier.

I_{O1}, $\overline{I_{O1}}$ **I_{O2}, $\overline{I_{O2}}$** These high impedance current output pairs are selected by the output select latch. I_{O1} and I_{O2} are true outputs and $\overline{I_{O1}}$ and $\overline{I_{O2}}$ are complementary outputs.

FUNCTION TABLES

DATA LATCH CONTROL

$\overline{CS}$	$\overline{W}$	$\overline{DE}$	Data Latch
0	0	0	Transparent
X	X	1	Latched
X	1	X	Latched
1	X	X	Latched

STATUS LATCH CONTROL

$\overline{CS}$	$\overline{W}$	SE	CODE SEL and OUT SEL Latch
0	0	1	Transparent
X	X	0	Latched
X	1	X	Latched
1	X	X	Latched

CODE SELECT AND OUTPUT SELECT

CODE SEL	OUT SEL	Function
0	–	MSB Inverted (Note 1)
1	–	MSB Non-inverted
–	0	Output Channel 1
–	1	Output Channel 2

X = Don't Care

Note 1. 1 LSB balance current is added to the $\overline{I_O}$ output.

MAXIMUM RATINGS

Operating Temperature		Power Supply Voltage	±18V
Am6081ADM, Am6081DM	−55°C to +125°C	Logic Inputs	−5V to +18V
Am6081ADC, Am6081DC Am6081APC, Am6081PC	0°C to +70°C	Analog Current Outputs	−12V to +18V
		Reference Inputs (V_{15}, V_{16})	V− to V+
Storage Temperature	−65°C to +150°C	Reference Input Differential Voltage (V_{15} to V_{16})	±18V
Lead Temperature (Soldering, 60 sec)	300°C	Reference Input Current (I_{15})	1.25mA

GUARANTEED FUNCTIONAL SPECIFICATIONS

Resolution	8 bits
Monotonicity	8 bits

ELECTRICAL CHARACTERISTICS

These specifications apply for $V_+ = +5V$, $V_- = -15V$, $I_{REF} = 0.5mA$, over the operating temperature range unless otherwise specified. Output characteristics refer to all outputs.

Parameter	Description	Conditions	Am6081A Min.	Am6081A Typ.	Am6081A Max.	Am6081 Min.	Am6081 Typ.	Am6081 Max.	Unit
	Resolution	Straight coding/Sign Magnitude	8/9	8/9	8/9	8/9	8/9	8/9	bits
	Monotonicity	Straight coding/Sign Magnitude	8/9	8/9	8/9	8/9	8/9	8/9	bits
D.N.L.	Differential Nonlinearity		–	–	±0.19	–	–	±0.39	%FS
N.L.	Nonlinearity		–	–	±0.1	–	–	±0.19	%FS
I_{FS}	Full Scale Current	$V_{REF} = 10.000V$ $R_{15} = R_{16} = 20.000k\Omega$ $T_A = 25°C$	1.984	1.992	2.000	1.976	1.992	2.008	mA
TCI_{FS}	Full Scale Tempco		–	±5	±20	–	±10	±40	ppm/°C
			–	±.0005	±.002		±.001	±.004	%FS/°C
V_{OC}	Output Voltage Compliance		−10	–	+18	−10	–	+18	Volts
I_{FSS}	Full Scale Symmetry	$I_{FS1} - \overline{I_{FS1}}$ or $I_{FS2} - \overline{I_{FS2}}$	–	±0.1	±1.0	–	±0.2	±2.0	μA
I_{OSS}	Output Switch Symmetry	$I_{FS1} - I_{FS2}$ or $\overline{I_{FS1}} - \overline{I_{FS2}}$	–	±0.1	±1.0	–	±0.2	±2.0	μA
I_{ZS}	Zero Scale Current		–	0.01	0.4	–	0.01	0.8	μA
I_{DIS}	Output Disable Current	Output of mpx "Off" Channels	–	0.01	0.05	–	0.01	0.05	μA
I_{RR}	Reference Current Range	V− = −5V	0	0.5	0.55	0	0.5	0.55	mA
		V− = −15V	0	0.5	1.1	0	0.5	1.1	
V_{IL}	Logic Input Levels Logic "0"		–	–	0.8	–	–	0.8	Volts
V_{IH}	Logic Input Levels Logic "1"		2.0	–	–	2.0	–	–	
I_{IN}	Logic Input Current	V_{IN} = −5V to +18V	–	–	40	–	–	40	μA
V_{IS}	Logic Input Swing	V− = −15V	−5	–	+18	−5	–	+18	Volts
I_{16}	Reference Bias Current		–	−0.5	−2.0	–	−0.5	−2.0	μA
dI/dt	Reference Input Slew Rate	$R_{15(EQ)} = 800\Omega$ CC = 0pF	4.0	8.0	–	4.0	8.0	–	mA/μs
$PSSI_{FS+}$	Power Supply Sensitivity	V+ = +4.5V to +5.5V, V− = −15V	–	±0.0005	±0.01	–	±0.0005	±0.01	%FS
$PSSI_{FS-}$		V− = −13.5V to − 16.5V, V+ = +5V	–	±0.0005	±0.01	–	±0.0005	±0.01	
V+	Power Supply Range	$I_{REF} = 0.5mA$, $V_{OUT} = 0V$	4.5	–	18	4.5	–	18	Volts
V−			−18	–	−4.5	−18	–	−4.5	
I+	Power Supply Current	V+ = +5V, V− = −5V	–	9.8	14.7	–	9.8	14.7	mA
I−			–	−7.4	−9.9	–	−7.4	−9.9	
I+		V+ = +5V, V− = −15V	–	9.8	14.7	–	9.8	14.7	
I−			–	−7.4	−9.9	–	−7.4	−9.9	
I+		V+ = +15V, V− = −15V		9.8	14.7	–	9.8	14.7	
I−			–	−7.4	−9.9	–	−7.4	−9.9	
P_D	Power Dissipation	V+ = +5V, V− = −5V	–	86	123	–	86	123	mW
		V+ = +5V, V− = −15V	–	160	222	–	160	222	
		V+ = +15V, V− = −15V	–	258	369	–	258	369	

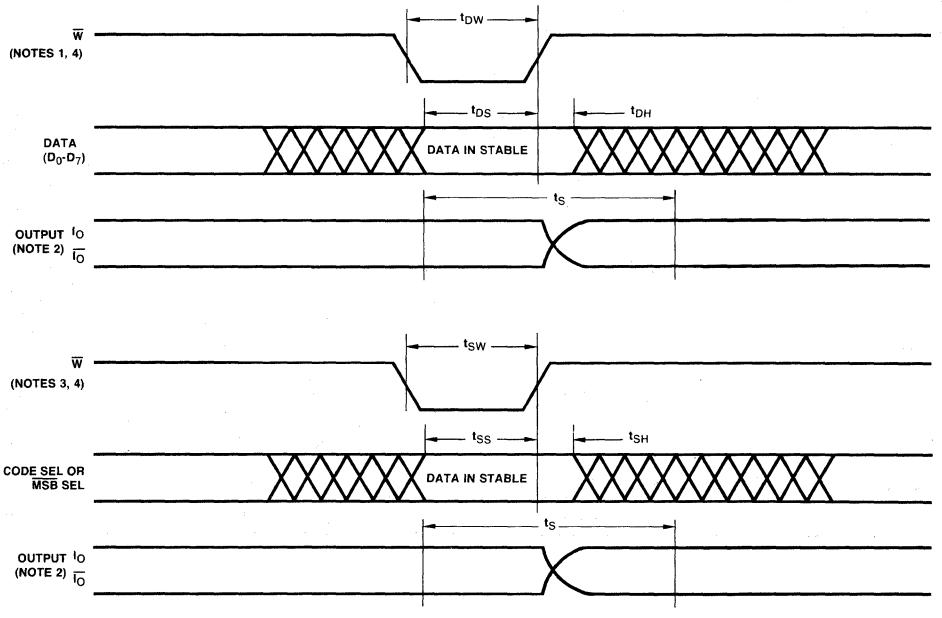
AC CHARACTERISTICS

$V_+ = +5V$, $V_- = -15V$, $I_{REF} = 0.5mA$, $R_L < 500\Omega$, $C_L < 15pF$ over the operating temperature range unless otherwise specified

Parameter	Description		Conditions	Commercial Temp. Grades			Military Temp. Grades			Unit
				Min.	Typ.	Max.	Min.	Typ.	Max.	
t_S	Settling Time, All Bits Switched		$T_A = 25°C$ Settling to ±½LSB		200			200		ns
t_{PLH}	Propagation Delay	Each bit	$T_A = 25°C$ 50% to 50%		90	180		90	180	ns
t_{PHL}		All bits switched			90	180		90	180	
t_{OS}	Output Switch Settling Time		$T_A = 25°C$ to ±1/2LSB of I_{FS}		250			250		ns
t_{OP}	Output Switch Propagation Delay		$T_A = 25°C$, 50% to 50%		150	300		150	300	ns
t_{DH}	Data Hold Time		See timing diagram	10	−30		10	−30		ns
t_{DS}	Data Set Up Time		See timing diagram	80	35		100	35		ns
t_{DW}	Data Write Time		See timing diagram	80	35		100	35		ns
t_{SH}	Status Hold Time		See timing diagram	10	−70		10	−70		ns
t_{SS}	Status Set Up Time		See timing diagram	200	100		230	100		ns
t_{SW}	Status Write Time		See timing diagram	200	100		230	100		ns

Notes:
1. t_{DW} is the overlap of $\overline{W}$ low, $\overline{CS}$ low, and $\overline{DE}$ low. All three signals must be low to enable the latch. Any signal going inactive latches the data.
2. t_S is measured with the latches open from the time the data becomes stable on the inputs to the time when the outputs are settled to within ±1/2 LSB. All bits switched on or off.
3. t_{SW} is the overlap of $\overline{W}$ low, $\overline{CS}$ low and SE high, all three signals must be active to enable the latch and any signal going inactive will latch the data.
4. The internal time delays from $\overline{CS}$, $\overline{W}$, SE and $\overline{DE}$ inputs to the enabling of the latches are all equal.

TIMING DIAGRAM



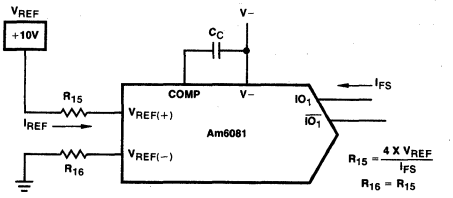
LIC-003

APPLICATION HINTS

1. Reference current and reference resistor

There is a 1 to 4 scale up between the reference current (I_{REF}) and the full scale output current (I_{FS}). If $V_{REF} = +10V$ and $I_{FS} = 2mA$, the value of the R_{15} is:

$$R_{15} = \frac{4 \times 10\ Volt}{2mA} = 20K\Omega$$



LIC-004

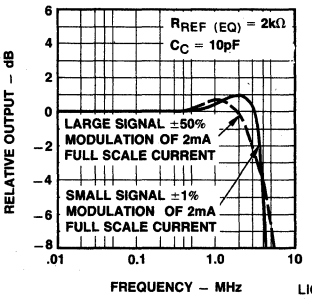
2. Reference amplifier compensation

For AC reference applications, a minimum value compensation capacitor (C_C) is normally used. The value of this capacitor depends on R_{15}. The minimum values to maximize bandwidth without oscillation are as follows:

Table 2
Compensation Capacitor
(I_{FS} = 2mA, I_{REF} = 0.5mA)

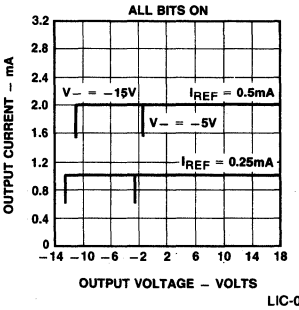
R_{REF} (kΩ)	C_C (pF)
20	100
10	50
5	25
2	10
1	5
.5	0

Reference Amplifier Frequency Response



LIC-005

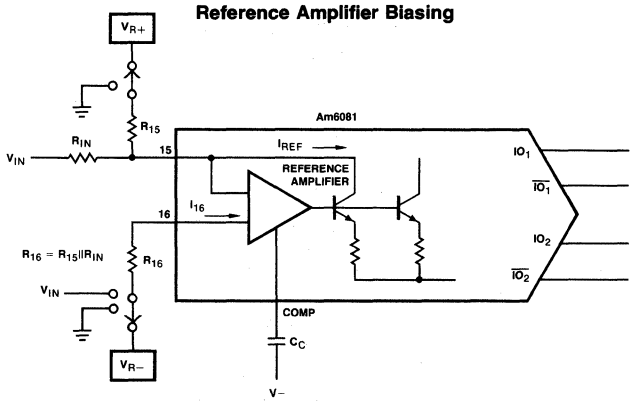
Output Voltage Compliance



LIC-006

A 0.01μF capacitor is recommended for the fixed reference operation.

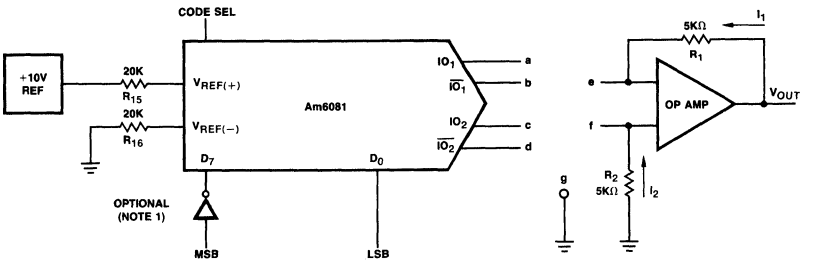
Reference Amplifier Biasing



LIC-007

Reference Configuration	R_{15}	R_{16}	R_{IN}	C_C	I_{REF}
Positive Reference	V_{R+}	0V	N/C	.01μF	V_{R+}/R_{15}
Negative Reference	0V	V_{R-}	N/C	.01μF	$-V_{R-}/R_{15}$
Lo Impedance Bipolar Reference	V_{R+}	0V	V_{IN}	(Note 1)	$(V_{R+}/R_{15}) + (V_{IN}/R_{IN})$ (Note 2)
Hi Impedance Bipolar Reference	V_{R+}	V_{IN}	N/C	(Note 1)	$(V_{R+} - V_{IN})/R_{15}$ (Note 3)
Pulsed Reference (Note 4)	V_{R+}	0V	V_{IN}	No Cap	$(V_{R+}/R_{15}) + (V_{IN}/R_{IN})$

Notes:
1. The compensation capacitor is a function of the impedance seen at the $+V_{REF}$ input and must be at least $C = 5pF \times R_{15(EQ)}$ (in kΩ). For $R_{15} < 800\Omega$ no capacitor is necessary.
2. For negative values of V_{IN}, V_{R+}/R_{15} must be greater than $-V_{IN}$ Max/R_{IN} so that the amplifier is not turned off.
3. For positive values of V_{IN}, V_{R+} must be greater than V_{IN} Max so the amplifier is not turned off.
4. For pulsed operation, V_{R+} provides a DC offset and may be set to zero in some cases. The impedance at pin 15 should be 800Ω or less and an additional resistor may be connected from pin 15 to ground to lower the impedance.



LIC-008

Note: Connect all unused outputs to ground.

CODE FORMAT		CODE SEL	OUT SEL	CON-NECTIONS	OUTPUT SCALE	OUT SEL	MSB D7	D6	D5	D4	D3	D2	D1	LSB D0	I_1 (mA)	I_2 (mA)	V_{OUT}
UNIPOLAR	Straight binary: one polarity with true input code, true zero output.	1	0	a-e b-g	Positive full scale	X	1	1	1	1	1	1	1	1	1.992	0	9.960
					Positive full scale − LSB	X	1	1	1	1	1	1	1	0	1.984	0	9.920
					Zero scale	X	0	0	0	0	0	0	0	0	.000	0	.000
			1	c-e d-g													
	Complementary binary: one polarity with complementary input code, true zero output.	1	0	a-g b-e	Positive full scale	X	0	0	0	0	0	0	0	0	1.992	0	9.960
					Positive full scale − LSB	X	0	0	0	0	0	0	0	1	1.984	0	9.920
					Zero scale	X	1	1	1	1	1	1	1	1	.000	0	.000
			1	c-g d-e													
SIGNED MAGNITUDE	Signed magnitude binary: 8 bits + sign reflected code, overlapping true zero output.	1		a-e c-f	Positive full scale	1	1	1	1	1	1	1	1	1	1.992	.000	9.960
					Positive full scale − LSB	1	1	1	1	1	1	1	1	0	1.984	.000	9.920
					(+) Zero scale	1	0	0	0	0	0	0	0	0	.000	.000	.000
					(−) Zero scale	0	0	0	0	0	0	0	0	0	.000	.000	.000
					Negative full scale − LSB	0	1	1	1	1	1	1	1	0	.000	1.984	−9.920
					Negative full scale	0	1	1	1	1	1	1	1	1	.000	1.992	−9.960
	Complementary signed magnitude: 8 bits + sign complementary reflected code, overlapping true zero output.	1		b-e d-f	Positive full scale	1	0	0	0	0	0	0	0	0	1.992	.000	9.960
					Positive full scale − LSB	1	0	0	0	0	0	0	0	1	1.984	.000	9.920
					(+) Zero scale	1	1	1	1	1	1	1	1	1	.000	.000	.000
					(−) Zero scale	0	1	1	1	1	1	1	1	1	.000	.000	.000
					Negative full scale − LSB	0	0	0	0	0	0	0	0	1	.000	1.984	−9.920
					Negative full scale	0	0	0	0	0	0	0	0	0	.000	1.992	−9.960
SYMMETRICAL OFFSET	Straight offset binary: offset half scale, symmetrical about zero, no true zero output.	1	0	a-e b-f	Positive full scale	X	1	1	1	1	1	1	1	1	1.992	.000	9.960
					Positive full scale − LSB	X	1	1	1	1	1	1	1	0	1.984	.008	9.880
					(+) Zero scale	X	1	0	0	0	0	0	0	0	1.000	.992	.040
					(−) Zero scale	X	0	1	1	1	1	1	1	1	.992	1.000	−.040
			1	c-e d-f	Negative full scale − LSB	X	0	0	0	0	0	0	0	1	.008	1.984	−9.880
					Negative full scale	X	0	0	0	0	0	0	0	0	.000	1.992	−9.960
	1's complement: offset half scale, symmetrical about zero, no true zero output MSB complemented. (need inverter at D_7)	1 (Note 1)	0	a-e b-f	Positive full scale	X	0	1	1	1	1	1	1	1	1.992	.000	9.960
					Positive full scale − LSB	X	0	1	1	1	1	1	1	0	1.984	.008	9.980
					(+) Zero scale	X	0	0	0	0	0	0	0	0	1.000	.992	.040
					(−) Zero scale	X	1	1	1	1	1	1	1	1	.992	1.000	−.040
			1	c-e d-f	Negative full scale − LSB	X	1	0	0	0	0	0	0	1	.008	1.984	−9.880
					Negative full scale	X	1	0	0	0	0	0	0	0	.000	1.992	−9.960
OFFSET WITH TRUE ZERO	Offset binary: offset half scale, true zero output MSB complemented remainder add to I_O. (need inverter at D_7)	0 (Note 1)	0	a-e b-f	Positive full scale	X	1	1	1	1	1	1	1	1	1.992	.008	9.920
					Positive full scale − LSB	X	1	1	1	1	1	1	1	0	1.984	.016	9.840
					+ LSB	X	1	0	0	0	0	0	0	1	1.008	.992	.080
					Zero scale	X	1	0	0	0	0	0	0	0	1.000	1.000	.000
			1	c-e d-f	− LSB	X	0	1	1	1	1	1	1	1	1.992	1.008	−.080
					Negative full scale + LSB	X	0	0	0	0	0	0	0	1	.008	1.992	−9.920
					Negative full scale	X	0	0	0	0	0	0	0	0	.000	2.000	−10.000
	2's complement: offset half scale true zero output MSB complemented.	0	0	a-e b-f	Positive full scale	X	0	1	1	1	1	1	1	1	1.992	.008	9.920
					Positive full scale − LSB	X	0	1	1	1	1	1	1	0	1.984	.016	9.840
					+1 LSB	X	0	0	0	0	0	0	0	1	1.008	.992	.080
					Zero scale	X	0	0	0	0	0	0	0	0	1.000	1.000	.000
			1	c-e d-f	−1 LSB	X	1	1	1	1	1	1	1	1	.992	1.008	−.080
					Negative full scale + LSB	X	1	0	0	0	0	0	0	1	.008	1.992	−9.920
					Negative full scale	X	1	0	0	0	0	0	0	0	.000	2.000	−10.000

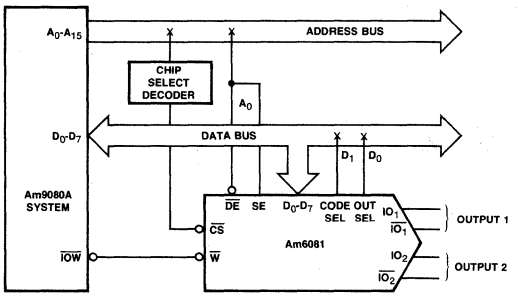
Note 1: An external inverter is necessary since the code select inverts the MSB and adds a 1 LSB balance current to $\overline{I_O}$. Only one of the two features is desired for these codes.

ADDITIONAL CODE MODIFICATIONS

1. Any of the offset binary codes may be complemented by reversing the output terminal pair.
2. The sign on any of the sign-magnitude codes may be changed by reversing the output terminal pair.
3. The polarity of the unipolar codes may be changed by driving the opposite side of the balanced load.

SYSTEM APPLICATIONS

Am9080A DATA SYSTEM: SEPARATE UPDATE OF DATA AND STATUS



SELECT OUTPUT PORT 1

MVI A, 2 : SET STATUS TO 0 (SELECT OUTPUT 1)
OUT 1 : SEND STATUS
MOV A, M : GET DATA FROM MEMORY
OUT 0 : SEND DATA

SELECT OUTPUT PORT 2

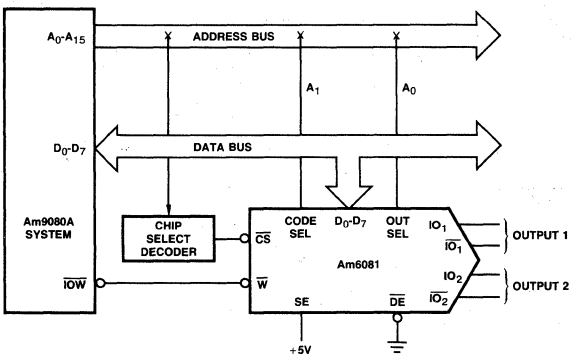
MVI A, 3 : SET STATUS TO 1 (SELECT OUTPUT 2)
OUT 1 : SEND STATUS
MOV A,M : GET DATA FROM MEMORY
OUT 0 : SEND DATA

SELECT OUTPUT PORT 2 AND 2's COMPLEMENT CODE

MVI A, 1 : SET STATUS TO 3 (OUTPUT 2, MSB COMP)
OUT 1 : SEND STATUS
MOV A, M : GET DATA FROM MEMORY
OUT 0 : SEND DATA

LIC-009

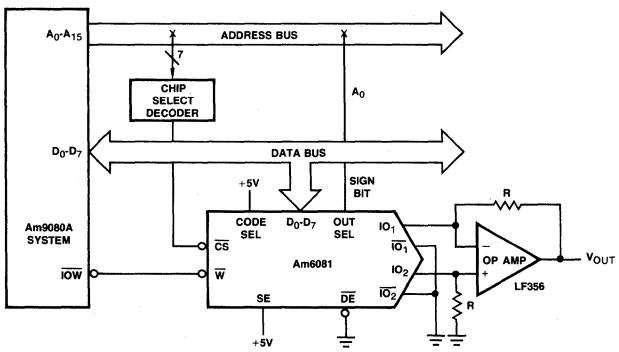
Am9080A DATA SYSTEM: SIMULTANEOUS UPDATE OF DATA AND STATUS



MOV A, M : GET DATA IN ACCUMULATOR
OUT 0 : OUTPUT DATA TO PORT 1, 2'S COMPLEMENT
OUT 1 : OUTPUT DATA TO PORT 2, 2'S COMPLEMENT
OUT 2 : OUTPUT DATA TO PORT 1, STRAIGHT BINARY
OUT 3 : OUTPUT DATA TO PORT 2, STRAIGHT BINARY

LIC-010

Am9080A DATA SYSTEM: 8-BIT PLUS SIGN CONVERSION

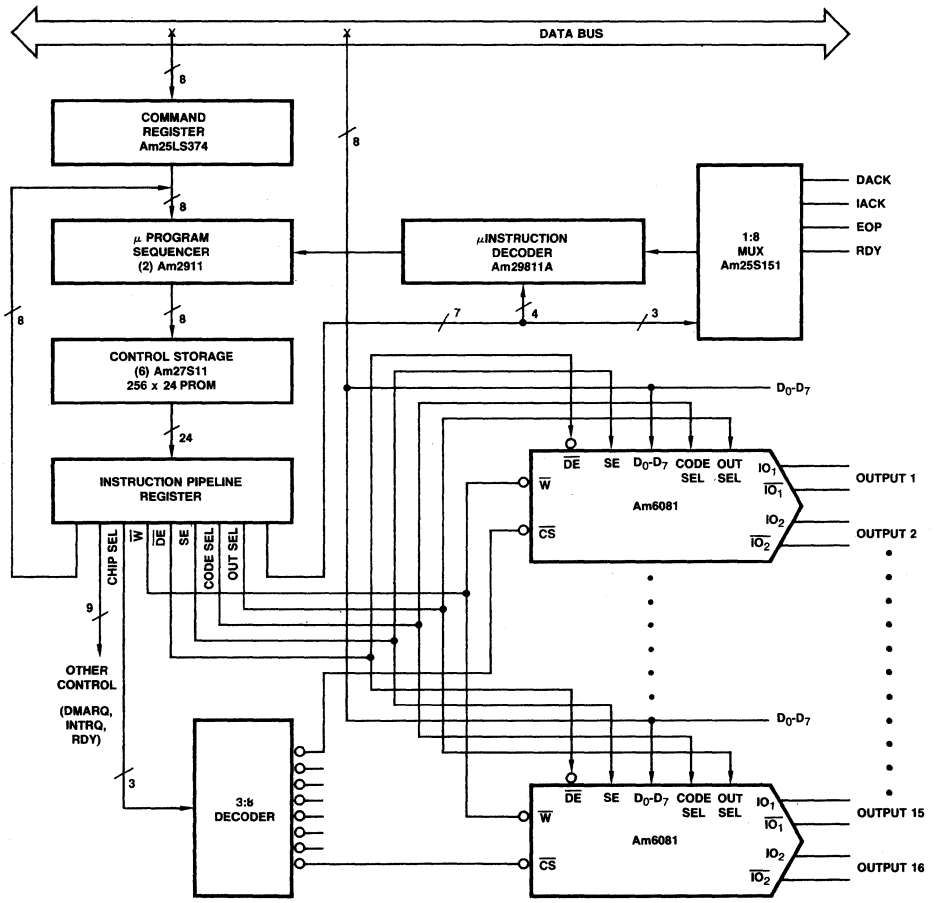


MOV A, M : LOAD MAGNITUDE (8-BITS)
OUT 0 : SEND POSITIVE OUTPUT
OUT 1 : SEND NEGATIVE OUTPUT

LIC-011

SYSTEM APPLICATIONS (Cont.)

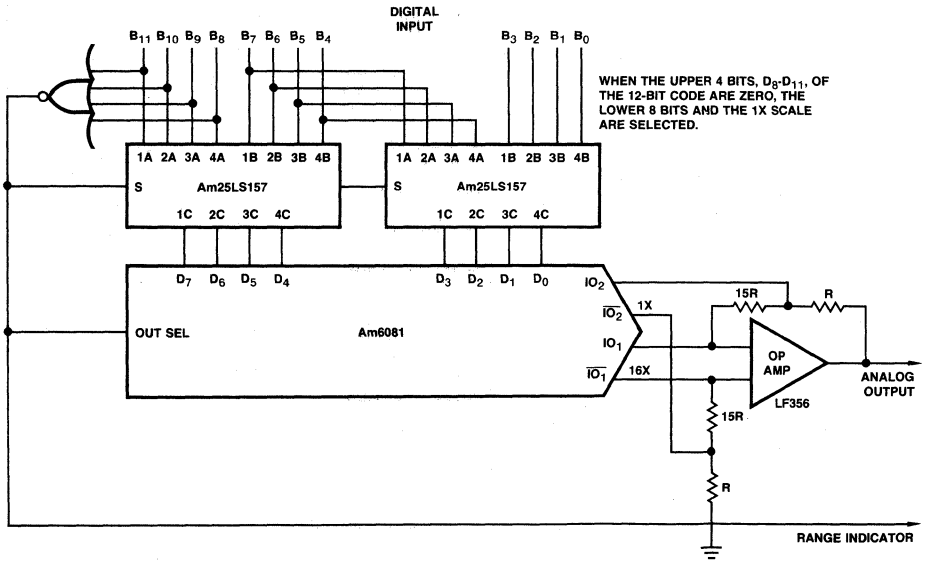
Am2900 DATA SYSTEM: MULTIPLE ANALOG OUTPUTS



LIC-012

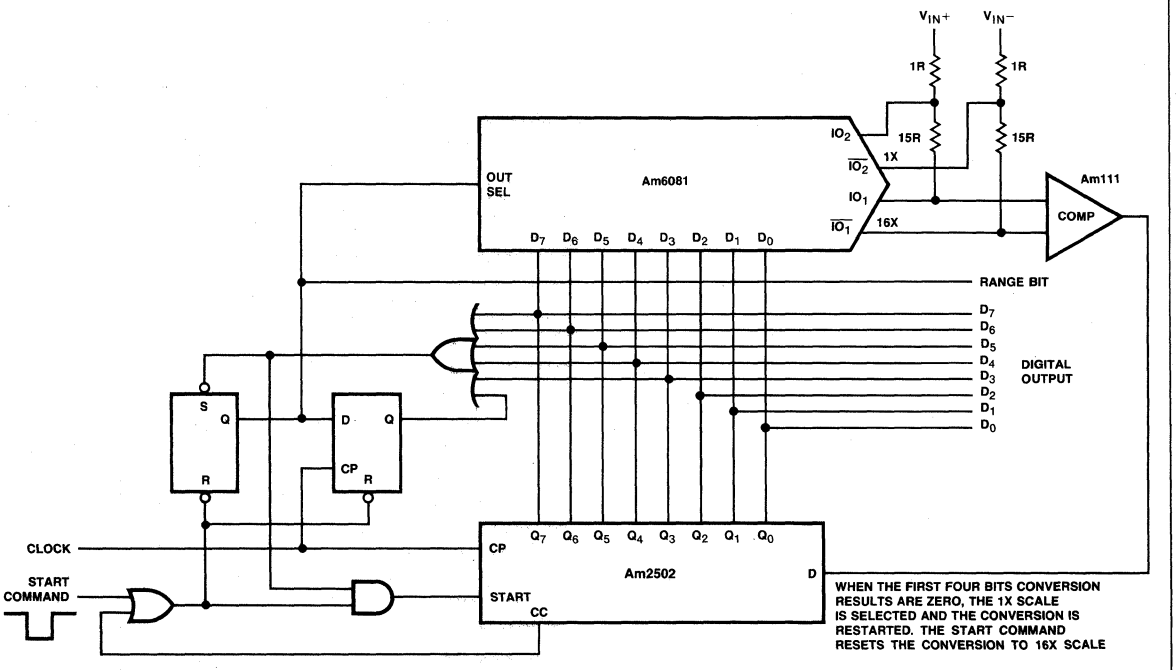
SYSTEM APPLICATIONS (Cont.)

D/A CONVERSION WITH 12-BIT DYNAMIC RANGE



LIC-013

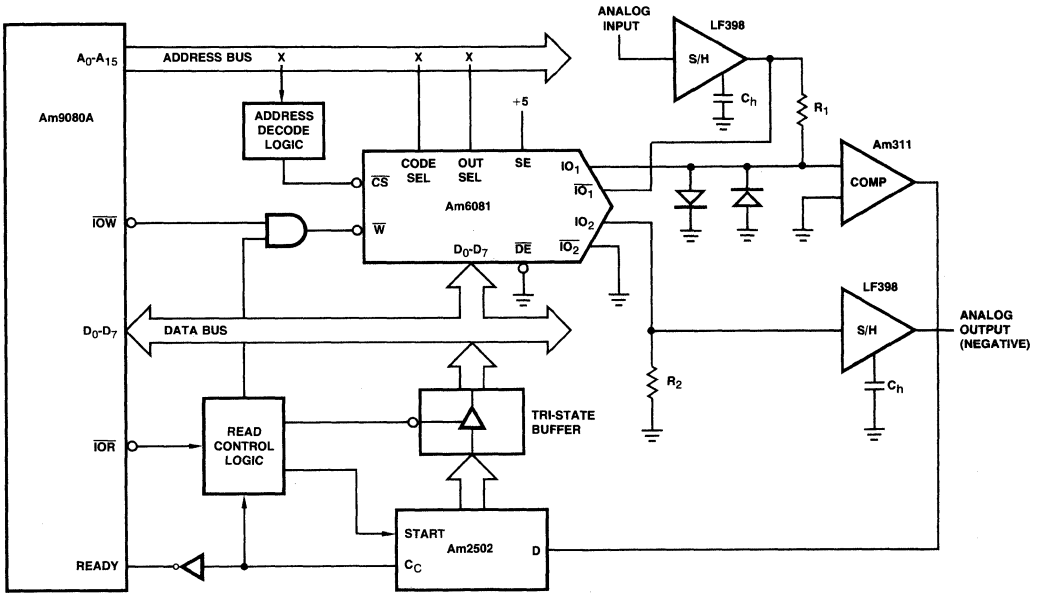
A/D CONVERSION WITH AUTO RANGING AND DIFFERENTIAL INPUT



LIC-014

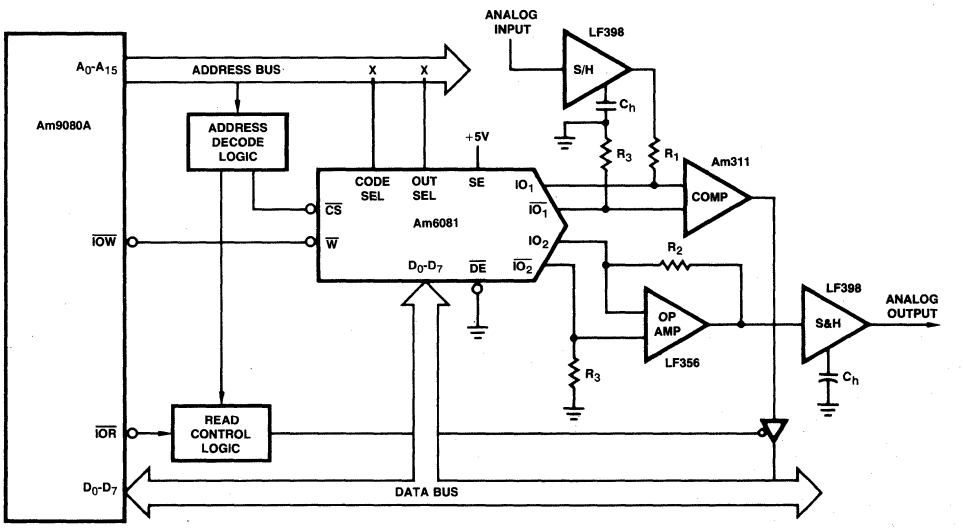
SYSTEM APPLICATIONS (Cont.)

ANALOG/DIGITAL TRANSCEIVER WITH HARDWARE CONTROLLED SUCCESSIVE APPROXIMATION A/D CONVERSION



LIC-015

ANALOG/DIGITAL TRANSCEIVER WITH SOFTWARE CONTROLLED A/D CONVERSION



LIC-016

Am9080A SOFTWARE FOR A/D AND D/A CONVERSION USING Am6081

```
SEQ     SOURCE STATEMENT
 0 PORT1   EQU 00H
 1 PORT3   EQU 02H
 2 PORT2   EQU 01H
 3         ORG 3E50H
 4 START:  LXI SP,STAKS-16   ;INITIAL STAKS POINTER
 5 SAMPLE: CALL ADCON        ;CALL A/D CONVERSATION
 6         CMA
 7         CALL DACON        ;CALL D/A CONVERSION
 8         JMP SAMPLE        ;NEXT SAMPLE
 9 ADCON:  XRA A             :CLEAR ACC
10         MOV D,A           ;CLEAR D REG
11         STC               ;SET CARRY
12         RAR               ;SET BIT 7 TO 1
13         MOV B,A           ;STORE TEST BIT AT B REGISTER
14 LOOP:   MOV E,A           ;STORE TEST WORD
15         CMA
16         OUT PORT1         ;OUTPUT TO A/D
17         IN PORT3          ;INPUT FROM COMP
```

```
SEQ     SOURCE STATEMENT
18         CMA
19         CRA A             ;SET SIGN FLAG
20         JM NEXT           ;IF SMALLER GO TO NEXT BIT
21         MOV D,E           ;SAVE RESULT
22 NEXT:   MOV A,B           ;GET NEXT TRIAL BIT
23         RAR               ;SHIFT RIGHT ONCE
24         RC                ;RETURN ON CARRY
25         MOV B,A           ;STORE TEST BIT
26         ADD D             ;ACCUMULATE RESULT
27         JMP LOOP          ;TRY NEXT BIT
28 DACON:  OUT PORT 2        ;OUTPUT TO D/A
29         MVI C,05H         ;LOAD C REG WITH TIME
30         DCR C             ;TIME DELAY
31         RZ                ;RETURN
32 FILT:   RET
33 STAKS:  DS 16
34         END START
```

ADVANCED MICRO DEVICES DATA CONVERSION PRODUCTS

Digital to Analog Converters

AmDAC-08	– 8-Bit High Speed Multiplying D/A Converter
Am1508/1408	– 8-Bit Multiplying D/A Converter
Am6070	– 8-Bit Companding D/A Converter for Control Systems (μ-law)
Am6071	– 8-Bit Companding D/A Converter for Control Systems (A-law)
Am6072	– 8-Bit Companding D/A Converter for Telecommunications (μ-law)
Am6073	– 8-Bit Companding D/A Converter for Telecommunicatons (A-law)
Am6080	– 8-Bit High Speed Multiplying D/A Converter System/Microprocessor Compatible
Am6081	– 8-Bit High Speed Multiplying D/A Converter System/Microprocessor Compatible
*Am6689	– 8-Bit, Ultra High Speed D/A Converter (ECL)
*Am6012	– 12-Bit High Speed Multiplying D/A Converter

Analog to Digital Converters

*Am6688	– 4-Bit Quantizer (Ultra High Speed A/D Converter)

Successive Approximation Registers

Am2502	– 8-Bit Successive Approximation Registers
Am2503	– 8-Bit Successive Approximation Registers
Am2504	– 12-Bit Successive Approximation Registers

Sample and Hold Amplifiers

LF198/398	–Monolithic Sample and Hold Amplifier
*Am6098	–Precision Sample and Hold Amplifier

Comparators

LM111/311	– Precision Voltage Comparator
LM119/319	– Dual Comparator
Am686	– High Speed Voltage Comparator

High Speed Operational Amplifiers

Am118/318	– High Speed Operational Amplifier
LF155/156/157	– JFET Input Operational Amplifiers
LF355/356/357	– JFET Input Operational Amplifiers

* To be announced.

Am9511A
Arithmetic Processor

DISTINCTIVE CHARACTERISTICS

- Replaces Am9511
- Fixed point 16 and 32 bit operations
- Floating point 32 bit operations
- Binary data formats
- Add, Subtract, Multiply and Divide
- Trigonometric and inverse trigonometric functions
- Square roots, logarithms, exponentiation
- Float to fixed and fixed to float conversions
- Stack-oriented operand storage
- DMA or programmed I/O data transfers
- End signal simplifies concurrent processing
- Synchronous/Asynchronous operations
- General purpose 8-bit data bus interface
- Standard 24 pin package
- +12 volt and +5 volt power supplies
- Advanced N-channel silicon gate MOS technology
- 100% MIL-STD-883 reliability assurance testing

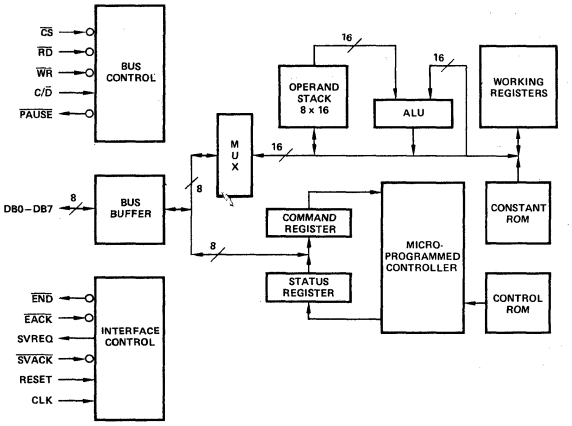
GENERAL DESCRIPTION

The Am9511A Arithmetic Processing Unit (APU) is a monolithic MOS/LSI device that provides high performance fixed and floating point arithmetic and a variety of floating point trigonometric and mathematical operations. It may be used to enhance the computational capability of a wide variety of processor-oriented systems.

All transfers, including operand, result, status and command information, take place over an 8-bit bidirectional data bus. Operands are pushed onto an internal stack and a command is issued to perform operations on the data in the stack. Results are then available to be retrieved from the stack, or additional commands may be entered.

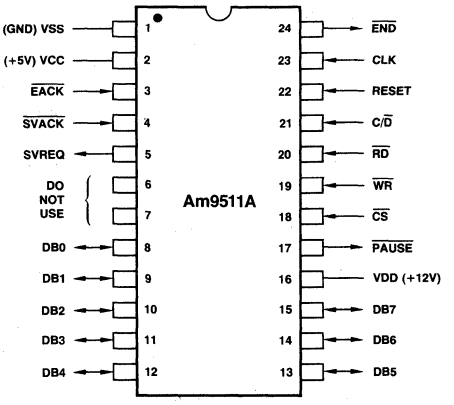
Transfers to and from the APU may be handled by the associated processor using conventional programmed I/O, or may be handled by a direct memory access controller for improved performance. Upon completion of each command, the APU issues an end of execution signal that may be used as an interrupt by the CPU to help coordinate program execution.

BLOCK DIAGRAM



MOS-046

CONNECTION DIAGRAM
Top View



Pin 1 is marked for orientation.

MOS-047

ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency	
		2MHz	3MHz
Hermetic DIP	0°C ≤ T_A ≤ +70°C	Am9511ADC	Am9511A-1DC
	−55°C ≤ T_A ≤ +125°C	Am9511ADM	Am9511A-1DM

10

INTERFACE SIGNAL DESCRIPTION

VCC: +5V Power Supply
VDD: +12V Power Supply
VSS: Ground

CLK (Clock, Input)

An external timing source connected to the CLK input provides the necessary clocking. The CLK input can be asynchronous to the $\overline{RD}$ and $\overline{WR}$ control signals.

RESET (Reset, Input)

A HIGH on this input causes initialization. Reset terminates any operation in progress, and clears the status register to zero. The internal stack pointer is initialized and the contents of the stack may be affected but the command register is not affected by the reset operation. After a reset the $\overline{END}$ output, and the SVREQ output will be LOW. For proper initialization, the RESET input must be HIGH for at least five CLK periods following stable power supply voltages and stable clock.

C/$\overline{D}$ (Command/Data Select, Input)

The C/$\overline{D}$ input together with the $\overline{RD}$ and $\overline{WR}$ inputs determines the type of transfer to be performed on the data bus as follows:

C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	Function
L	H	L	Push data byte into the stack
L	L	H	Pop data byte from the stack
H	H	L	Enter command byte from the data bus
H	L	H	Read Status
X	L	L	Undefined

L = LOW
H = HIGH
X = DON'T CARE

$\overline{END}$ (End of Execution, Output)

A LOW on this output indicates that execution of the current command is complete. This output will be cleared HIGH by activating the $\overline{EACK}$ input LOW or performing any read or write operation or device initialization using the RESET. If $\overline{EACK}$ is tied LOW, the $\overline{END}$ output will be a pulse (see $\overline{EACK}$ description). This is an open drain output and requires a pull up to +5V.

Reading the status register while a command execution is in progress is allowed. However any read or write operation clears the flip-flop that generates the $\overline{END}$ output. Thus such continuous reading could conflict with internal logic setting the $\overline{END}$ flip-flop at the completion of command execution.

$\overline{EACK}$ (End Acknowledge, Output)

This input when LOW makes the $\overline{END}$ output go LOW. As mentioned earlier HIGH on the $\overline{END}$ output signals completion of a command execution. The $\overline{END}$ output signal is derived from an internal flip-flop which is clocked at the completion of a command. This flip-flop is clocked to the reset state when $\overline{EACK}$ is LOW. Consequently, if the $\overline{EACK}$ is tied LOW, the $\overline{END}$ output will be a pulse that is approximately one CLK period wide.

SVREQ (Service Request, Output)

A HIGH on this output indicates completion of a command. In this sense this output is same as the $\overline{END}$ output. However, whether the SVREQ output will go HIGH at the completion of a command or not is determined by a service request bit in the command register. This bit must be 1 for SVREQ to go HIGH. The SVREQ can be cleared (i.e., go LOW) by activating the $\overline{SVACK}$ input LOW or initializing the device using the RESET. Also, the SVREQ will be automatically cleared after completion of any command that has the service request bit as 0.

$\overline{SVACK}$ (Service Acknowledge, Input)

A LOW on this input activates the reset input of the flip-flop generating the SVREQ output. If the $\overline{SVACK}$ input is permanently tied LOW, it will conflict with the internal setting of the flip-flop to generate the SVREQ output. Thus the SVREQ indication cannot be relied upon if the $\overline{SVACK}$ is tied LOW.

DB0-DB7 (Bidirectional Data Bus, Input/Output)

These eight bidirectional lines are used to transfer command, status and operand information between the device and the host processor. DB0 is the least significant and DB7 is the most significant bit position. HIGH on the data bus line corresponds to 1 and LOW corresponds to 0.

When pushing operands on the stack using the data bus, the least significant byte must be pushed first and most significant byte last. When popping the stack to read the result of an operation, the most significant byte will be available on the data bus first and the least significant byte will be the last. Moreover, for pushing operands and popping results, the number of transactions must be equal to the proper number of bytes appropriate for the chosen format. Otherwise, the internal byte pointer will not be aligned properly. The Am9511A single precision format requires 2 bytes, double precision and floating-point formats require 4 bytes.

$\overline{CS}$ (Chip Select, Input)

This input must be LOW to accomplish any read or write operation to the Am9511A.

To perform a write operation data is presented on DB0 through DB7 lines, C/$\overline{D}$ is driven to an appropriate level and the $\overline{CS}$ input is made LOW. However, actual writing into the Am9511A cannot start until $\overline{WR}$ is made LOW. After initiating the write operation by a $\overline{WR}$ HIGH to LOW transition, the $\overline{PAUSE}$ output will go LOW momentarily (TPPWW).

The $\overline{WR}$ input can go HIGH after $\overline{PAUSE}$ goes HIGH. The data lines, C/$\overline{D}$ input and the $\overline{CS}$ input can change when appropriate hold time requirements are satisfied. See write timing diagram for details.

To perform a read operation an appropriate logic level is established on the C/$\overline{D}$ input and $\overline{CS}$ is made LOW. The Read operation does not start until the $\overline{RD}$ input goes LOW. $\overline{PAUSE}$ will go LOW for a period of TPPWR. When $\overline{PAUSE}$ goes back HIGH again, it indicates that read operation is complete and the required information is available on the DB0 through DB7 lines. This information will remain on the data lines as long as $\overline{RD}$ input is LOW. The $\overline{RD}$ input can return HIGH anytime after $\overline{PAUSE}$ goes HIGH. The $\overline{CS}$ input and C/$\overline{D}$ inputs can change anytime after $\overline{RD}$ returns HIGH. See read timing diagram for details. The $\overline{CS}$ must have a HIGH to LOW transition for every READ or WRITE operation.

$\overline{RD}$ (Read, Input)

A LOW on this input is used to read information from an internal location and gate that information on to the data bus. The $\overline{CS}$ input must be LOW to accomplish the read operation. The C/$\overline{D}$ input determines what internal location is of interest. See C/$\overline{D}$, $\overline{CS}$ input descriptions and read timing diagram for details. If the END output was LOW, performing any read operation will make the $\overline{END}$ output go HIGH after the HIGH to LOW transition of the $\overline{RD}$ input (assuming $\overline{CS}$ is LOW).

$\overline{WR}$ (Write, Input)

A LOW on this input is used to transfer information from the data bus into an internal location. The $\overline{CS}$ must be LOW to accomplish the write operation. The C/$\overline{D}$ determines which internal location is to be written. See C/$\overline{D}$, $\overline{CS}$ input descriptions and write timing diagram for details.

If the $\overline{END}$ output was LOW, performing any write operation will make the $\overline{END}$ output go HIGH after the LOW to HIGH transition of the $\overline{WR}$ input (assuming $\overline{CS}$ is LOW).

$\overline{PAUSE}$ (Pause, Output)

This output is a handshake signal used while performing read or write transactions with the Am9511A. A LOW at this output indicates that the Am9511A has not yet completed its information transfer with the host over the data bus. During a read operation, after $\overline{CS}$ went LOW, the $\overline{PAUSE}$ will become LOW shortly (TRP) after $\overline{RD}$ goes LOW. $\overline{PAUSE}$ will return high only after the data bus contains valid output data. The $\overline{CS}$ and $\overline{RD}$ should remain LOW when $\overline{PAUSE}$ is LOW. The RD may go high anytime after $\overline{PAUSE}$ goes HIGH. During a write operation, after $\overline{CS}$ went LOW, the $\overline{PAUSE}$ will be LOW for a very short duration (TPPWN) after $\overline{WR}$ goes LOW. Since the minimum of TPPWW is 0, the $\overline{PAUSE}$ may not go LOW at all for fast devices. $\overline{WR}$ may go HIGH anytime after $\overline{PAUSE}$ goes HIGH.

FUNCTIONAL DESCRIPTION

Major functional units of the Am9511A are shown in the block diagram. The Am9511A employs a microprogram controlled stack oriented architecture with 16-bit wide data paths.

The Arithmetic Logic Unit (ALU) receives one of its operands from the Operand Stack. This stack is an 8-word by 16-bit 2-port memory with last in-first out (LIFO) attributes. The second operand to the ALU is supplied by the internal 16-bit bus. In addition to supplying the second operand, this bidirectional bus also carries the results from the output of the ALU when required. Writing into the Operand Stack takes place from this internal 16-bit bus when required. Also connected to this bus are the Constant ROM and Working Registers. The ROM provides the required constants to perform the mathematical operations (Chebyshev Algorithms) while the Working Registers provide storage for the intermediate values during command execution.

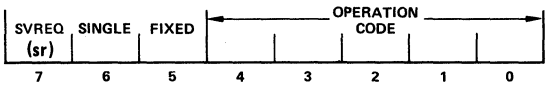
Communication between the external world and the Am9511A takes place on eight bidirectional input/output lines DB0 through DB7 (Data Bus). These signals are gated to the internal eight-bit bus through appropriate interface and buffer circuitry. Multiplexing facilities exist for bidirectional communication between the internal eight and sixteen-bit buses. The Status Register and Command Register are also accessible via the eight-bit bus.

The Am9511A operations are controlled by the microprogram contained in the Control ROM. The Program Counter supplies the microprogram addresses and can be partially loaded from the Command Register. Associated with the Program Counter is the Subroutine Stack where return addresses are held during subroutine calls in the microprogram. The Microinstruction Register holds the current microinstruction being executed. This register facilitates pipelined microprogram execution. The Instruction Decode logic generates various internal control signals needed for the Am9511A operation.

The Interface Control logic receives several external inputs and provides handshake related outputs to facilitate interfacing the Am9511A to microprocessors.

COMMAND FORMAT

Each command entered into the Am9511A consists of a single 8-bit byte having the format illustrated below:



Bits 0-4 select the operation to be performed as shown in the table. Bits 5-6 select the data format for the operation. If bit 5 is a 1, a fixed point data format is specified. If bit 5 is a 0, floating point format is specified. Bit 6 selects the precision of the data to be operated on by fixed point commands (if bit 5 = 0, bit 6 must be 0). If bit 6 is a 1, single-precision (16-bit) operands are indicated; if bit 6 is a 0, double-precision (32-bit) operands are indicated. Results are undefined for all illegal combinations of bits in the command byte. Bit 7 indicates whether a service request is to be issued after the command is executed. If bit 7 is a 1, the service request output (SVREQ) will go high at the conclusion of the command and will remain high until reset by a low level on the service acknowledge pin ($\overline{SVACK}$) or until completion of execution of a succeeding command where bit 7 is 0. Each command issued to the Am9511A requests post execution service based upon the state of bit 7 in the command byte. When bit 7 is a 0, SVREQ remains low.

COMMAND SUMMARY

Command Code								Command Mnemonic	Command Description
7	6	5	4	3	2	1	0		
FIXED-POINT 16-BIT									
sr	1	1	0	1	1	0	0	SADD	Add TOS to NOS. Result to NOS. Pop Stack.
sr	1	1	0	1	1	0	1	SSUB	Subtract TOS from NOS. Result to NOS. Pop Stack.
sr	1	1	0	1	1	1	0	SMUL	Multiply NOS by TOS. Lower half of result to NOS. Pop Stack.
sr	1	1	1	0	1	1	0	SMUU	Multiply NOS by TOS. Upper half of result to NOS. Pop Stack.
sr	1	1	0	1	1	1	1	SDIV	Divide NOS by TOS. Result to NOS. Pop Stack.
FIXED-POINT 32-BIT									
sr	0	1	0	1	1	0	0	DADD	Add TOS to NOS. Result to NOS. Pop Stack.
sr	0	1	0	1	1	0	1	DSUB	Subtract TOS from NOS. Result to NOS. Pop Stack.
sr	0	1	0	1	1	1	0	DMUL	Multiply NOS by TOS. Lower half of result to NOS. Pop Stack.
sr	0	1	1	0	1	1	0	DMUU	Multiply NOS by TOS. Upper half of result to NOS. Pop Stack.
sr	0	1	0	1	1	1	1	DDIV	Divide NOS by TOS. Result to NOS. Pop Stack.
FLOATING-POINT 32-BIT									
sr	0	0	1	0	0	0	0	FADD	Add TOS to NOS. Result to NOS. Pop Stack.
sr	0	0	1	0	0	0	1	FSUB	Subtract TOS from NOS. Result to NOS. Pop Stack.
sr	0	0	1	0	0	1	0	FMUL	Multiply NOS by TOS. Result to NOS. Pop Stack.
sr	0	0	1	0	0	1	1	FDIV	Divide NOS by TOS. Result to NOS. Pop Stack.
DERIVED FLOATING-POINT FUNCTIONS									
sr	0	0	0	0	0	0	1	SQRT	Square Root of TOS. Result in TOS.
sr	0	0	0	0	0	1	0	SIN	Sine of TOS. Result in TOS.
sr	0	0	0	0	0	1	1	COS	Cosine of TOS. Result in TOS.
sr	0	0	0	0	1	0	0	TAN	Tangent of TOS. Result in TOS.
sr	0	0	0	0	1	0	1	ASIN	Inverse Sine of TOS. Result in TOS.
sr	0	0	0	0	1	1	0	ACOS	Inverse Cosine of TOS. Result in TOS.
sr	0	0	0	0	1	1	1	ATAN	Inverse Tangent of TOS. Result in TOS.
sr	0	0	0	1	0	0	0	LOG	Common Logarithm (base 10) of TOS. Result in TOS.
sr	0	0	0	1	0	0	1	LN	Natural Logarithm (base e) of TOS. Result in TOS.
sr	0	0	0	1	0	1	0	EXP	Exponential (e^X) of TOS. Result in TOS.
sr	0	0	0	1	0	1	1	PWR	NOS raised to the power in TOS. Result in NOS. Pop Stack.
DATA MANIPULATION COMMANDS									
sr	0	0	0	0	0	0	0	NOP	No Operation
sr	0	0	1	1	1	1	1	FIXS	Convert TOS from floating point to 16-bit fixed point format.
sr	0	0	1	1	1	1	0	FIXD	Convert TOS from floating point to 32-bit fixed point format.
sr	0	0	1	1	1	0	1	FLTS	Convert TOS from 16-bit fixed point to floating point format.
sr	0	0	1	1	1	0	0	FLTD	Convert TOS from 32-bit fixed point to floating point format.
sr	1	1	1	0	1	0	0	CHSS	Change sign of 16-bit fixed point operand on TOS.
sr	0	1	1	0	1	0	0	CHSD	Change sign of 32-bit fixed point operand on TOS.
sr	0	0	1	0	1	0	1	CHSF	Change sign of floating point operand on TOS.
sr	1	1	1	0	1	1	1	PTOS	Push 16-bit fixed point operand on TOS to NOS (Copy)
sr	0	1	1	0	1	1	1	PTOD	Push 32-bit fixed point operand on TOS to NOS. (Copy)
sr	0	0	1	0	1	1	1	PTOF	Push floating point operand on TOS to NOS. (Copy)
sr	1	1	1	1	0	0	0	POPS	Pop 16-bit fixed point operand from TOS. NOS becomes TOS.
sr	0	1	1	1	0	0	0	POPD	Pop 32-bit fixed point operand from TOS. NOS becomes TOS.
sr	0	0	1	1	0	0	0	POPF	Pop floating point operand from TOS. NOS becomes TOS.
sr	1	1	1	1	0	0	1	XCHS	Exchange 16-bit fixed point operands TOS and NOS.
sr	0	1	1	1	0	0	1	XCHD	Exchange 32-bit fixed point operands TOS and NOS.
sr	0	0	1	1	0	0	1	XCHF	Exchange floating point operands TOS and NOS.
sr	0	0	1	1	0	1	0	PUPI	Push floating point constant "π" onto TOS. Previous TOS becomes NOS.

NOTES:

1. TOS means Top of Stack. NOS means Next on Stack.
2. AMD Application Brief "Algorithm Details for the Am9511A APU" provides detailed descriptions of each command function, including data ranges, accuracies, stack configurations, etc.
3. Many commands destroy one stack location (bottom of stack) during development of the result. The derived functions may destroy several stack locations. See Application Brief for details.
4. The trigonometric functions handle angles in radians, not degrees.
5. No remainder is available for the fixed-point divide functions.
6. Results will be undefined for any combination of command coding bits not specified in this table.

COMMAND INITIATION

After properly positioning the required operands on the stack, a command may be issued. The procedure for initiating a command execution is as follows:

1. Enter the appropriate command on the DB0-DB7 lines.
2. Establish HIGH on the C/$\overline{D}$ input.
3. Establish LOW on the $\overline{CS}$ input.
4. Establish LOW on the $\overline{WR}$ input after an appropriate set up time (see timing diagrams).
5. Sometime after the HIGH to LOW level transition of $\overline{WR}$ input, the $\overline{PAUSE}$ output will become LOW. After a delay of TPPWW, it will go HIGH to acknowledge the write operation. The WR input can return to HIGH anytime after $\overline{PAUSE}$ going HIGH. The DB0-DB7, C/$\overline{D}$ and $\overline{CS}$ inputs are allowed to change after the hold time requirements are satisfied (see timing diagram).

An attempt to issue a new command while the current command execution is in progress is allowed. Under these circumstances, the $\overline{PAUSE}$ output will not go HIGH until the current command execution is completed.

OPERAND ENTRY

The Am9511A commands operate on the operands located at the TOS and NOS and results are returned to the stack at NOS and then popped to TOS. The operands required for the Am9511A are one of three formats – single precision fixed-point (2 bytes), double precision fixed-point (4 bytes) or floating-point (4 bytes). The result of an operation has the same format as the operands except for float to fix or fix to float commands.

Operands are always entered into the stack least significant byte first and most significant byte last. The following procedure must be followed to enter operands onto the stack:

1. The lower significant operand byte is established on the DB0-DB7 lines.
2. A LOW is established on the C/$\overline{D}$ input to specify that data is to be entered into the stack.
3. The $\overline{CS}$ input is made LOW.
4. After appropriate set up time (see timing diagrams), the $\overline{WR}$ input is made LOW. The $\overline{PAUSE}$ output will become LOW.
5. Sometime after this event, the $\overline{PAUSE}$ will return HIGH to indicate that the write operation has been acknowledged.
6. Anytime after the $\overline{PAUSE}$ output goes HIGH the $\overline{WR}$ input can be made HIGH. The DB0-DB7, C/$\overline{D}$ and $\overline{CS}$ inputs can change after appropriate hold time requirements are satisfied (see timing diagrams).

The above procedure must be repeated until all bytes of the operand are pushed into the stack. It should be noted that for single precision fixed-point operands 2 bytes should be pushed and 4 bytes must be pushed for double precision fixed-point or floating-point. Not pushing all the bytes of a quantity will result in byte pointer misalignment.

The Am9511A stack can accommodate 8 single precision fixed-point quantities or 4 double precision fixed-point or floating-point quantities. Pushing more quantities than the capacity of the stack will result in loss of data which is usual with any LIFO stack.

DATA REMOVAL

Result from an operation will be available at the TOS. Results can be transferred from the stack to the data bus by reading the stack. When the stack is popped for results, the most significant byte is available first and the least significant byte last. A result is always of the same precision as the operands that produced it except for format conversion commands. Thus when the result is taken from the stack, the total number of bytes popped out should be appropriate with the precision – single precision results are 2 bytes and double precision and floating-point results are 4 bytes. The following procedure must be used for reading the result from the stack:

1. A LOW is established on the C/$\overline{D}$ input.
2. The $\overline{CS}$ input is made LOW.
3. After appropriate set up time (see timing diagrams), the $\overline{RD}$ input is made LOW. The $\overline{PAUSE}$ will become LOW.
4. Sometime after this, $\overline{PAUSE}$ will return HIGH indicating that the data is available on the DB0-DB7 lines. This data will remain on the DB0-DB7 lines as long as the $\overline{RD}$ input remains LOW.
5. Anytime after $\overline{PAUSE}$ goes HIGH, the $\overline{RD}$ input can return HIGH to complete transaction.
6. The $\overline{CS}$ and C/$\overline{D}$ inputs can change after appropriate hold time requirements are satisfied (see timing diagram).
7. Repeat this procedure until all bytes appropriate for the precision of the result are popped out.

Reading of the stack does not alter its data; it only adjusts the byte pointer. If more data is popped than the capacity of the stack, the internal byte pointer will wrap around and older data will be read again, consistent with the LIFO stack.

STATUS READ

The Am9511A status register can be read without any regard to whether a command is in progress or not. The only implication that has to be considered is the effect this might have on the END output discussed in the signal descriptions.

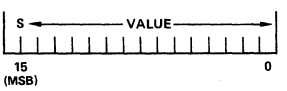
The following procedure must be followed to accomplish status register reading.

1. Establish HIGH on the C/$\overline{D}$ input.
2. Establish LOW on the $\overline{CS}$ input.
3. After appropriate set up time (see timing diagram) $\overline{RD}$ input is made LOW. The $\overline{PAUSE}$ will become LOW.
4. Sometime after the HIGH to LOW transition of $\overline{RD}$ input, the $\overline{PAUSE}$ will become HIGH indicating that status register contents are available on the DB0-DB7 lines. The status data will remain on DB0-DB7 as long as $\overline{RD}$ input is LOW.
5. The $\overline{RD}$ input can be returned HIGH anytime after $\overline{PAUSE}$ goes HIGH.
6. The C/$\overline{D}$ input and $\overline{CS}$ input can change after satisfying appropriate hold time requirements (see timing diagram).

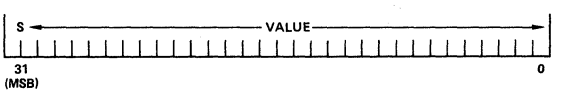
DATA FORMATS

The Am9511A Arithmetic Processing Unit handles operands in both fixed-point and floating-point formats. Fixed-point operands may be represented in either single (16-bit operands) or double precision (32-bit operands), and are always represented as binary, two's complement values.

16-BIT FIXED-POINT FORMAT



32-BIT FIXED-POINT FORMAT



The sign (positive or negative) of the operand is located in the most significant bit (MSB). Positive values are represented by a sign bit of zero (S = 0). Negative values are represented by the two's complement of the corresponding positive value with a sign bit equal to 1 (S = 1). The range of values that may be accommodated by each of these formats is −32,768 to +32,767 for single precision and −2,147,483,648 to +2,147,483,647 for double precision.

Floating point binary values are represented in a format that permits arithmetic to be performed in a fashion analogous to operations with decimal values expressed in scientific notation.

$$(5.83 \times 10^2)(8.16 \times 10^1) = (4.75728 \times 10^4)$$

In the decimal system, data may be expressed as values between 0 and 10 times 10 raised to a power that effectively shifts the implied decimal point right or left the number of places necessary to express the result in conventional form (e.g., 47,572.8). The value-portion of the data is called the mantissa. The exponent may be either negative or positive.

The concept of floating point notation has both a gain and a loss associated with it. The gain is the ability to represent the significant digits of data with values spanning a large dynamic range limited only by the capacity of the exponent field. For example, in decimal notation if the exponent field is two digits wide, and the mantissa is five digits, a range of values (positive or negative) from 1.0000×10^{-99} to $9.9999 \times 10^{+99}$ can be accommodated. The loss is that only the significant digits of the value can be represented. Thus there is no distinction in this representation between the values 123451 and 123452, for example, since each would be expressed as: 1.2345×10^5. The sixth digit has been discarded. In most applications where the dynamic range of values to be represented is large, the loss of significance, and hence accuracy of results, is a minor consideration. For greater precision a fixed point format could be chosen, although with a loss of potential dynamic range.

The Am9511 is a binary arithmetic processor and requires that floating point data be represented by a fractional mantissa value between .5 and 1 multiplied by 2 raised to an appropriate power. This is expressed as follows:

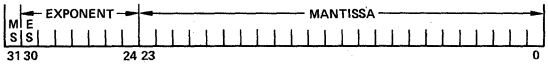
$$\text{value} = \text{mantissa} \times 2^{\text{exponent}}$$

For example, the value 100.5 expressed in this form is 0.11001001×2^7. The decimal equivalent of this value may be computed by summing the components (powers of two) of the mantissa and then multiplying by the exponent as shown below:

$$\begin{aligned}\text{value} &= (2^{-1} + 2^{-2} + 2^{-5} + 2^{-8}) \times 2^7 \\ &= (0.5 + 0.25 + 0.03125 + 0.00290625) \times 128 \\ &= 0.78515625 \times 128 \\ &= 100.5\end{aligned}$$

FLOATING POINT FORMAT

The format for floating-point values in the Am9511A is given below. The mantissa is expressed as a 24-bit (fractional) value; the exponent is expressed as an unbiased two's complement 7-bit value having a range of −64 to +63. The most significant bit is the sign of the mantissa (0 = positive, 1 = negative), for a total of 32 bits. The binary point is assumed to be to the left of the most significant mantissa bit (bit 23). All floating-point data values must be normalized. Bit 23 must be equal to 1, except for the value zero, which is represented by all zeros.



The range of values that can be represented in this format is $\pm(2.7 \times 10^{-20}$ to $9.2 \times 10^{18})$ and zero.

STATUS REGISTER

The Am9511A contains an eight bit status register with the following bit assignments:

BUSY	SIGN	ZERO	ERROR CODE				CARRY
7	6	5	4	3	2	1	0

BUSY: Indicates that Am9511A is currently executing a command (1 = Busy).

SIGN: Indicates that the value on the top of stack is negative (1 = Negative).

ZERO: Indicates that the value on the top of stack is zero (1 = Value is zero).

ERROR CODE: This field contains an indication of the validity of the result of the last operation. The error codes are:

- 0000 – No error
- 1000 – Divide by zero
- 0100 – Square root or log of negative number
- 1100 – Argument of inverse sine, cosine, or e^x too large
- XX10 – Underflow
- XX01 – Overflow

CARRY: Previous operation resulted in carry or borrow from most significant bit. (1 = Carry/Borrow, 0 = No Carry/No Borrow)

If the BUSY bit in the status register is a one, the other status bits are not defined; if zero, indicating not busy, the operation is complete and the other status bits are defined as given above.

Table 1.

Command Mnemonic	Hex Code (sr = 1)	Hex Code (sr = 0)	Execution Cycles	Summary Description
16-BIT FIXED-POINT OPERATIONS				
SADD	EC	6C	16-18	Add TOS to NOS. Result to NOS. Pop Stack.
SSUB	ED	6D	30-32	Subtract TOS from NOS. Result to NOS. Pop Stack.
SMUL	EE	6E	84-94	Multiply NOS by TOS. Lower result to NOS. Pop Stack.
SMUU	F6	76	80-98	Multiply NOS by TOS. Upper result to NOS. Pop Stack.
SDIV	EF	6F	84-94	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FIXED-POINT OPERATIONS				
DADD	AC	2C	20-22	Add TOS to NOS. Result to NOS. Pop Stack.
DSUB	AD	2D	38-40	Subtract TOS from NOS. Result to NOS. Pop Stack.
DMUL	AE	2E	194-210	Multiply NOS by TOS. Lower result to NOS. Pop Stack.
DMUU	B6	36	182-218	Multiply NOS by TOS. Upper result to NOS. Pop Stack.
DDIV	AF	2F	196-210	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FLOATING-POINT PRIMARY OPERATIONS				
FADD	90	10	54-368	Add TOS to NOS. Result to NOS. Pop Stack.
FSUB	91	11	70-370	Subtract TOS from NOS. Result to NOS. Pop Stack.
FMUL	92	12	146-168	Multiply NOS by TOS. Result to NOS. Pop Stack.
FDIV	93	13	154-184	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FLOATING-POINT DERIVED OPERATIONS				
SQRT	81	01	782-870	Square Root of TOS. Result to TOS.
SIN	82	02	3796-4808	Sine of TOS. Result to TOS.
COS	83	03	3840-4878	Cosine of TOS. Result to TOS.
TAN	84	04	4894-5886	Tangent of TOS. Result to TOS.
ASIN	85	05	6230-7938	Inverse Sine of TOS. Result to TOS.
ACOS	86	06	6304-8284	Inverse Cosine of TOS. Result to TOS.
ATAN	87	07	4992-6536	Inverse Tangent of TOS. Result to TOS.
LOG	88	08	4474-7132	Common Logarithm of TOS. Result to TOS.
LN	89	09	4298-6956	Natural Logarithm of TOS. Result to TOS.
EXP	8A	0A	3794-4878	e raised to power in TOS. Result to TOS.
PWR	8B	0B	8290-12032	NOS raised to power in TOS. Result to NOS. Pop Stack.
DATA AND STACK MANIPULATION OPERATIONS				
NOP	80	00	4	No Operation. Clear or set SVREQ.
FIXS	9F	1F	90-214	Convert TOS from floating point format to fixed point format.
FIXD	9E	1E	90-336	
FLTS	9D	1D	62-156	Convert TOS from fixed point format to floating point format.
FLTD	9C	1C	56-342	
CHSS	F4	74	22-24	Change sign of fixed point operand on TOS.
CHSD	B4	34	26-28	
CHSF	95	15	16-20	Change sign of floating point operand on TOS.
PTOS	F7	77	16	Push stack. Duplicate NOS in TOS.
PTOD	B7	37	20	
PTOF	97	17	20	
POPS	F8	78	10	Pop stack. Old NOS becomes new TOS. Old TOS rotates to bottom.
POPD	B8	38	12	
POPF	98	18	12	
XCHS	F9	79	18	Exchange TOS and NOS.
XCHD	B9	39	26	
XCHF	99	19	26	
PUPI	9A	1A	16	Push floating point constant π onto TOS. Previous TOS becomes NOS.

COMMAND DESCRIPTIONS

This section contains detailed descriptions of the APU commands. They are arranged in alphabetical order by command mnemonic. In the descriptions, TOS means Top Of Stack and NOS means Next On Stack.

All derived functions except Square Root use Chebyshev polynomial approximating algorithms. This approach is used to help minimize the internal microprogram, to minimize the maximum error values and to provide a relatively even distribution of errors over the data range. The basic arithmetic operations are used by the derived functions to compute the various Chebyshev terms. The basic operations may produce error codes in the status register as a result.

Execution times are listed in terms of clock cycles and may be converted into time values by multiplying by the clock period used. For example, an execution time of 44 clock cycles when running at a 3MHz rate translates to 14 microseconds (44 x 32μs = 14μs). Variations in execution cycles reflect the data dependency of the algorithms.

In some operations exponent overflow or underflow may be possible. When this occurs, the exponent returned in the result will be 128 greater or smaller than its true value.

Many of the functions use portions of the data stack as scratch storage during development of the results. Thus previous values in those stack locations will be lost. Scratch locations destroyed are listed in the command descriptions and shown with the crossed-out locations in the Stack Contents After diagram.

Table 1 is a summary of all the Am9511A commands. It shows the hex codes for each command, the mnemonic abbreviation, a brief description and the execution time in clock cycles. The commands are grouped by functional classes.

The command mnemonics in alphabetical order are shown below in Table 2.

Table 2.

Command Mnemonics in Alphabetical Order.

ACOS	ARCCOSINE
ASIN	ARCSINE
ATAN	ARCTANGENT
CHSD	CHANGE SIGN DOUBLE
CHSF	CHANGE SIGN FLOATING
CHSS	CHANGE SIGN SINGLE
COS	COSINE
DADD	DOUBLE ADD
DDIV	DOUBLE DIVIDE
DMUL	DOUBLE MULTIPLY LOWER
DMUU	DOUBLE MULTIPLY UPPER
DSUB	DOUBLE SUBTRACT
EXP	EXPONENTIATION (e^X)
FADD	FLOATING ADD
FDIV	FLOATING DIVIDE
FIXD	FIX DOUBLE
FIXS	FIX SINGLE
FLTD	FLOAT DOUBLE
FLTS	FLOAT SINGLE
FMUL	FLOATING MULTIPLY
FSUB	FLOATING SUBTRACT
LOG	COMMON LOGARITHM
LN	NATURAL LOGARITHM
NOP	NO OPERATION
POPD	POP STACK DOUBLE
POPF	POP STACK FLOATING
POPS	POP STACK SINGLE
PTOD	PUSH STACK DOUBLE
PTOF	PUSH STACK FLOATING
PTOS	PUSH STACK SINGLE
PUPI	PUSH π
PWR	POWER (X^Y)
SADD	SINGLE ADD
SDIV	SINGLE DIVIDE
SIN	SINE
SMUL	SINGLE MULTIPLY LOWER
SMUU	SINGLE MULTIPLY UPPER
SQRT	SQUARE ROOT
SSUB	SINGLE SUBTRACT
TAN	TANGENT
XCHD	EXCHANGE OPERANDS DOUBLE
XCHF	EXCHANGE OPERANDS FLOATING
XCHS	EXCHANGE OPERANDS SINGLE

ACOS

32-BIT FLOATING-POINT INVERSE COSINE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	1	1	0

Hex Coding: 86 with sr = 1
06 with sr = 0

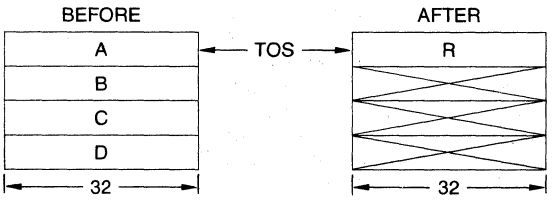
Execution Time: 6304 to 8284 clock cycles

Description:
The 32-bit floating-point operand A at the TOS is replaced by the 32-bit floating-point inverse cosine of A. The result R is a value in radians between 0 and π. Initial operands A, B, C and D are lost. ACOS will accept all input data values within the range of −1.0 to +1.0. Values outside this range will return an error code of 1100 in the status register.

Accuracy: ACOS exhibits a maximum relative error of 2.0 x 10^{-7} over the valid input data range.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



ATAN

32-BIT FLOATING-POINT INVERSE TANGENT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	1	1	1

Hex Coding: 87 with sr = 1
07 with sr = 0

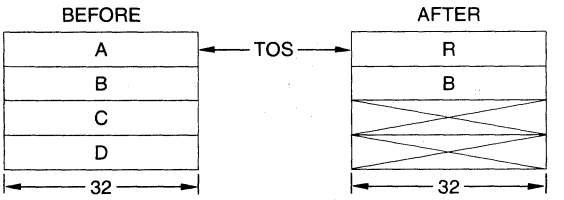
Execution Time: 4992 to 6536 clock cycles

Description:
The 32-bit floating-point operand A at the TOS is replaced by the 32-bit floating-point inverse tangent of A. The result R is a value in radians between $-\pi/2$ and $+\pi/2$. Initial operands A, C and D are lost. Operand B is unchanged.
ATAN will accept all input data values that can be represented in the floating point format.

Accuracy: ATAN exhibits a maximum relative error of 3.0 x 10^{-7} over the input data range.

Status Affected: Sign, Zero

STACK CONTENTS



ASIN

32-BIT FLOATING-POINT INVERSE SINE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	1	0	1

Hex Coding: 85 with sr = 1
05 with sr = 0

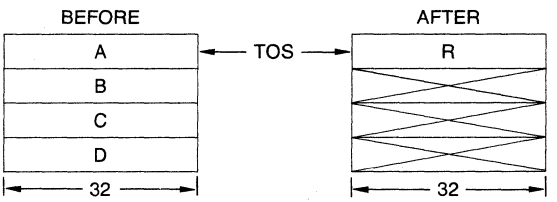
Execution Time: 6230 to 7938 clock cycles

Description:
The 32-bit floating-point operand A at the TOS is replaced by the 32-bit floating-point inverse sine of A. The result R is a value in radians between $-\pi/2$ and $+\pi/2$. Initial operands A, B, C and D are lost.
ASIN will accept all input data values within the range of −1.0 to +1.0. Values outside this range will return an error code of 1100 in the status register.

Accuracy: ASIN exhibits a maximum relative error of 4.0 x 10^{-7} over the valid input data range.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



CHSD

32-BIT FIXED-POINT SIGN CHANGE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	0	1	0	0

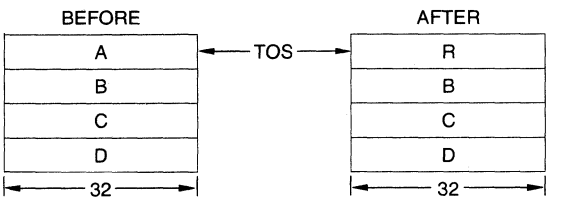
Hex Coding: B4 with sr = 1
34 with sr = 0

Execution Time: 26 to 28 clock cycles

Description:
The 32-bit fixed-point two's complement integer operand A at the TOS is subtracted from zero. The result R replaces A at the TOS. Other entries in the stack are not disturbed.
Overflow status will be set and the TOS will be returned unchanged when A is input as the most negative value possible in the format since no positive equivalent exists.

Status Affected: Sign, Zero, Error Field (overflow)

STACK CONTENTS



10

CHSF

32-BIT FLOATING-POINT SIGN CHANGE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	1	0	1

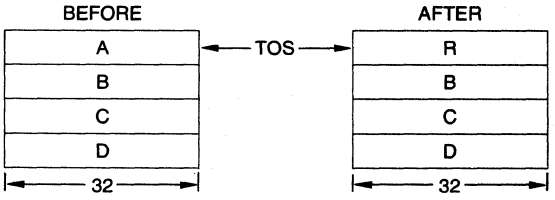
Hex Coding: 95 with sr = 1
15 with sr = 0

Execution Time: 16 to 20 clock cycles

Description:
The sign of the mantissa of the 32-bit floating-point operand A at the TOS is inverted. The result R replaces A at the TOS. Other stack entries are unchanged.
If A is input as zero (mantissa MSB = 0), no change is made.

Status Affected: Sign, Zero

STACK CONTENTS



CHSS

16-BIT FIXED-POINT SIGN CHANGE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	0	1	0	0

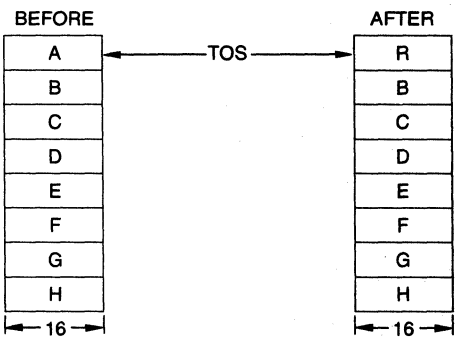
Hex Coding: F4 with sr = 1
74 with sr = 0

Execution Time: 22 to 24 clock cycles

Description:
16-bit fixed-point two's complement integer operand A at the TOS is subtracted from zero. The result R replaces A at the TOS. All other operands are unchanged.
Overflow status will be set and the TOS will be returned unchanged when A is input as the most negative value possible in the format since no positive equivalent exists.

Status Affected: Sign, Zero, Overflow

STACK CONTENTS



COS

32-BIT FLOATING-POINT COSINE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	0	1	1

Hex Coding: 83 with sr = 1
03 with sr = 0

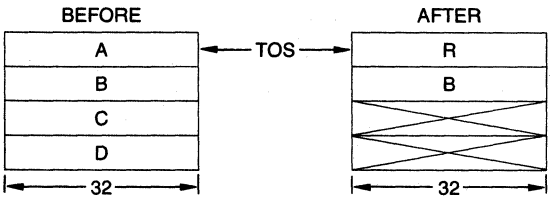
Execution Time: 3840 to 4878 clock cycles

Description:
The 32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point cosine of A. A is assumed to be in radians. Operands A, C and D are lost. B is unchanged.
The COS function can accept any input data value that can be represented in the data format. All input values are range reduced to fall within an interval of $-\pi/2$ to $+\pi/2$ radians.

Accuracy: COS exhibits a maximum relative error of 5.0 x 10^{-7} for all input data values in the range of -2π to $+2\pi$ radians.

Status Affected: Sign, Zero

STACK CONTENTS



DADD

32-BIT FIXED-POINT ADD

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	0	1	1	0	0

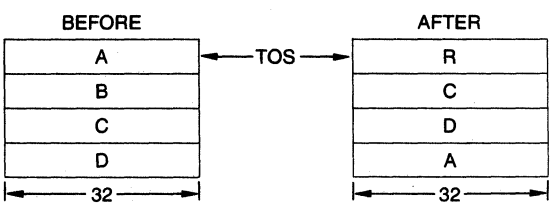
Hex Coding: AC with sr = 1
2C with sr = 0

Execution Time: 20 to 22 clock cycles

Description:
The 32-bit fixed-point two's complement integer operand A at the TOS is added to the 32-bit fixed-point two's complement integer operand B at the NOS. The result R replaces operand B and the Stack is moved up so that R occupies the TOS. Operand B is lost. Operands A, C and D are unchanged. If the addition generates a carry it is reported in the status register.
If the result is too large to be represented by the data format, the least significant 32 bits of the result are returned and overflow status is reported.

Status Affected: Sign, Zero, Carry, Error Field

STACK CONTENTS



DDIV

32-BIT FIXED-POINT DIVIDE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	0	1	1	1	1

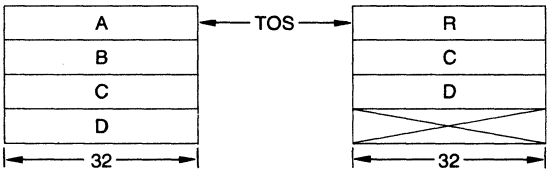
Hex Coding: AF with sr = 1
2F with sr = 0

Execution Time: 196 to 210 clock cycles when A ≠ 0
18 clock cycles when A = 0.

Description:
The 32-bit fixed-point two's complement integer operand B at NOS is divided by the 32-bit fixed-point two's complement integer operand A at the TOS. The 32-bit integer quotient R replaces B and the stack is moved up so that R occupies the TOS. No remainder is generated. Operands A and B are lost. Operands C and D are unchanged.
If A is zero, R is set equal to B and the divide-by-zero error status will be reported. If either A or B is the most negative value possible in the format, R will be meaningless and the overflow error status will be reported.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



DMUL

32-BIT FIXED-POINT MULTIPLY, LOWER

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	0	1	1	1	0

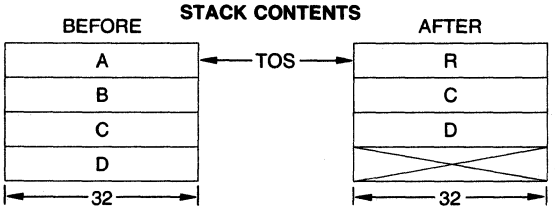
Hex Coding: AE with sr = 1
2E with sr = 0

Execution Time: 194 to 210 clock cycles

Description:
The 32-bit fixed-point two's complement integer operand A at the TOS is multiplied by the 32-bit fixed-point two's complement integer operand B at the NOS. The 32-bit least significant half of the product R replaces B and the stack is moved up so that R occupies the TOS. The most significant half of the product is lost. Operands A and B are lost. Operands C and D are unchanged.
The overflow status bit is set if the discarded upper half was non-zero. If either A or B is the most negative value that can be represented in the format, that value is returned as R and the overflow status is set.

Status Affected: Sign, Zero, Overflow

STACK CONTENTS



DMUU

32-BIT FIXED-POINT MULTIPLY, UPPER

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	0	1	1	0

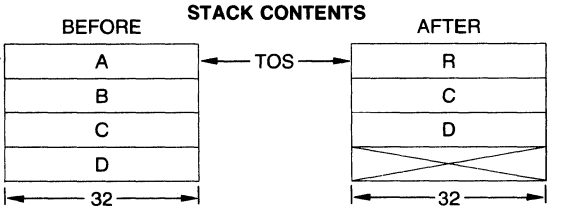
Hex Coding: B6 with sr = 1
36 with sr = 0

Execution Time: 182 to 218 clock cycles

Description:
The 32-bit fixed-point two's complement integer operand A at the TOS is multiplied by the 32-bit fixed-point two's complement integer operand B at the NOS. The 32-bit most significant half of the product R replaces B and the stack is moved up so that R occupies the TOS. The least significant half of the product is lost. Operands A and B are lost. Operands C and D are unchanged.
If A or B was the most negative value possible in the format, overflow status is set and R is meaningless.

Status Affected: Sign, Zero, Overflow

STACK CONTENTS



DSUB

32-BIT FIXED-POINT SUBTRACT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	0	1	1	0	1

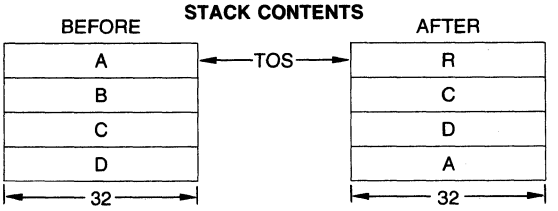
Hex Coding: AD with sr = 1
2D with sr = 0

Execution Time: 38 to 40 clock cycles

Description:
The 32-bit fixed-point two's complement operand A at the TOS is subtracted from the 32-bit fixed-point two's complement operand B at the NOS. The difference R replaces operand B and the stack is moved up so that R occupies the TOS. Operand B is lost. Operands A, C and D are unchanged.
If the subtraction generates a borrow it is reported in the carry status bit. If A is the most negative value that can be represented in the format the overflow status is set. If the result cannot be represented in the data format range, the overflow bit is set and the 32 least significant bits of the result are returned as R.

Status Affected: Sign, Zero, Carry, Overflow

STACK CONTENTS



EXP

32-BIT FLOATING-POINT e^X

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	1	0	1	0

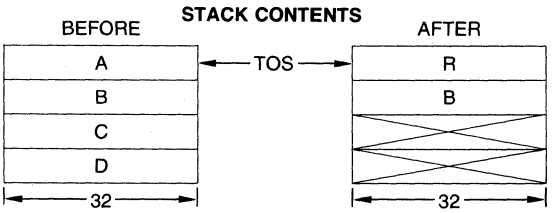
Hex Coding: 8A with sr = 1
0A with sr = 0

Execution Time: 3794 to 4878 clock cycles for $|A| \leq 1.0 \times 2^5$
34 clock cycles for $|A| > 1.0 \times 2^5$

Description:
The base of natural logarithms, e, is raised to an exponent value specified by the 32-bit floating-point operand A at the TOS. The result R of e^A replaces A. Operands A, C and D are lost. Operand B is unchanged.
EXP accepts all input data values within the range of $-1.0 \times 2^{+5}$ to $+1.0 \times 2^{+5}$. Input values outside this range will return a code of 1100 in the error field of the status register.

Accuracy: EXP exhibits a maximum relative error of 5.0×10^{-7} over the valid input data range.

Status Affected: Sign, Zero, Error Field



FADD

32-BIT FLOATING-POINT ADD

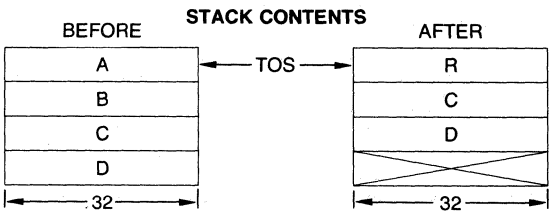
	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	0	0	0

Hex Coding: 90 with sr = 1
10 with sr = 0

Execution Time: 54 to 368 clock cycles for A ≠ 0
24 clock cycles for A = 0

Description:
32-bit floating-point operand A at the TOS is added to 32-bit floating-point operand B at the NOS. The result R replaces B and the stack is moved up so that R occupies the TOS. Operands A and B are lost. Operands C and D are unchanged.
Exponent alignment before the addition and normalization of the result accounts for the variation in execution time. Exponent overflow and underflow are reported in the status register, in which case the mantissa is correct and the exponent is offset by 128.

Status Affected: Sign, Zero, Error Field



FDIV

32-BIT FLOATING-POINT DIVIDE

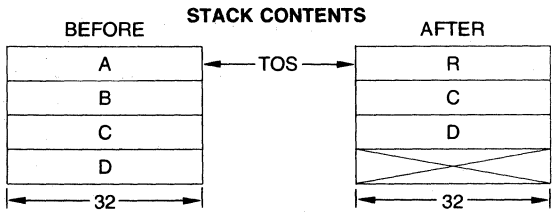
	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	0	1	1

Hex Coding: 93 with sr = 1
13 with sr = 0

Execution Time: 154 to 184 clock cycles for A ≠ 0
22 clock cycles for A = 0

Description:
32-bit floating-point operand B at NOS is divided by 32-bit floating-point operand A at the TOS. The result R replaces B and the stack is moved up so that R occupies the TOS. Operands A and B are lost. Operands C and D are unchanged.
If operand A is zero, R is set equal to B and the divide-by-zero error is reported in the status register. Exponent overflow or underflow is reported in the status register, in which case the mantissa portion of the result is correct and the exponent portion is offset by 128.

Status Affected: Sign, Zero, Error Field



FIXD

32-BIT FLOATING-POINT TO 32-BIT FIXED-POINT CONVERSION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	1	1	0

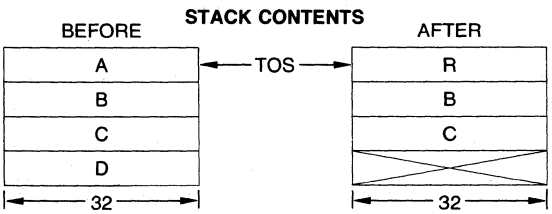
Hex Coding: 9E with sr = 1
1E with sr = 0

Execution Time: 90 to 336 clock cycles

Description:
32-bit floating-point operand A at the TOS is converted to a 32-bit fixed-point two's complement integer. The result R replaces A. Operands A and D are lost. Operands B and C are unchanged.

If the integer portion of A is larger than 31 bits when converted, the overflow status will be set and A will not be changed. Operand D, however, will still be lost.

Status Affected: Sign, Zero Overflow



FIXS

32-BIT FLOATING-POINT TO 16-BIT FIXED-POINT CONVERSION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	1	1	1

Hex Coding: 9F with sr = 1
1F with sr = 0

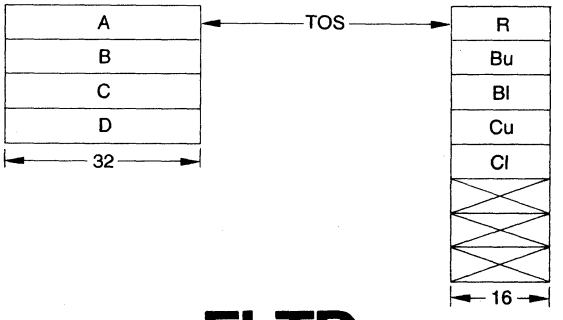
Execution Time: 90 to 214 clock cycles

Description:
32-bit floating-point operand A at the TOS is converted to a 16-bit fixed-point two's complement integer. The result R replaces the lower half of A and the stack is moved up by two bytes so that R occupies the TOS. Operands A and D are lost. Operands B and C are unchanged, but appear as upper (u) and lower (l) halves on the 16-bit wide stack if they are 32-bit operands.

If the integer portion of A is larger than 15 bits when converted, the overflow status will be set and A will not be changed. Operand D, however, will still be lost.

Status Affected: Sign, Zero, Overflow

STACK CONTENTS



FLTD

32-BIT FIXED-POINT TO 32-BIT FLOATING-POINT CONVERSION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	1	0	0

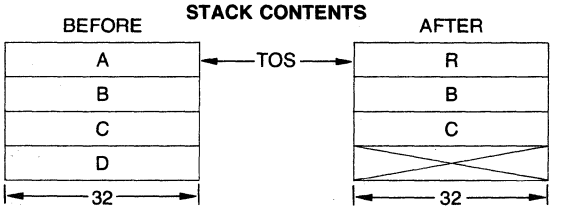
Hex Coding: 9C with sr = 1
1C with sr = 0

Execution Time: 56 to 342 clock cycles

Description:
32-bit fixed-point two's complement integer operand A at the TOS is converted to a 32-bit floating-point number. The result R replaces A at the TOS. Operands A and D are lost. Operands B and C are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



FLTS

16-BIT FIXED-POINT TO 32-BIT FLOATING-POINT CONVERSION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	1	0	1

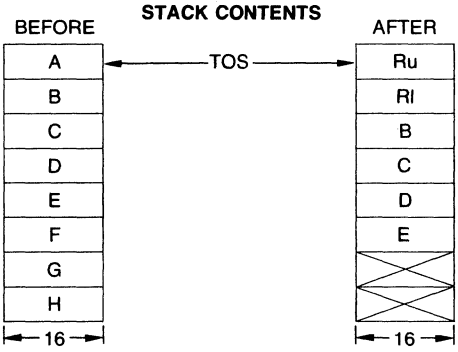
Hex Coding: 9D with sr = 1
1D with sr = 0

Execution Time: 62 to 156 clock cycles

Description:
16-bit fixed-point two's complement integer A at the TOS is converted to a 32-bit floating-point number. The lower half of the result R (Rl) replaces A, the upper half (Ru) replaces H and the stack is moved down so that Ru occupies the TOS. Operands A, F, G and H are lost. Operands B, C, D and E are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



FMUL

32-BIT FLOATING-POINT MULTIPLY

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	0	1	0

Hex Coding: 92 with sr = 1
12 with sr = 0

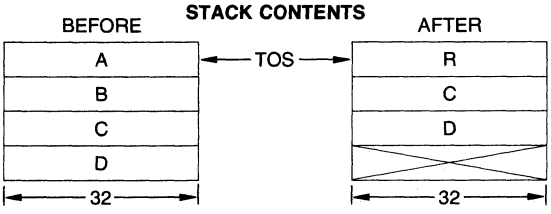
Execution Time: 146 to 168 clock cycles

Description:
32-bit floating-point operand A at the TOS is multiplied by the 32-bit floating-point operand B at the NOS. The normalized result R replaces B and the stack is moved up so that R occupies the TOS. Operands A and B are lost. Operands C and D are unchanged.

Exponent overflow or underflow is reported in the status register, in which case the mantissa portion of the result is correct and the exponent portion is offset by 128.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



FSUB

32-BIT FLOATING-POINT SUBTRACTION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	0	0	1

Hex Coding: 91 with sr = 1
11 with sr = 0

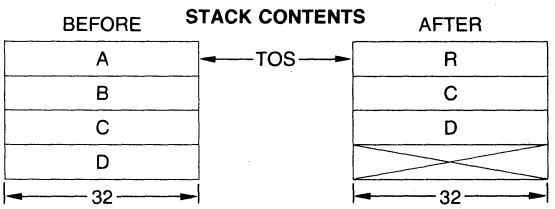
Execution Time: 70 to 370 clock cycles for A $\neq$ 0
26 clock cycles for A = 0

Description:
32-bit floating-point operand A at the TOS is subtracted from 32-bit floating-point operand B at the NOS. The normalized difference R replaces B and the stack is moved up so that R occupies the TOS. Operands A and B are lost. Operands C and D are unchanged.

Exponent alignment before the subtraction and normalization of the result account for the variation in execution time.

Exponent overflow or underflow is reported in the status register in which case the mantissa portion of the result is correct and the exponent portion is offset by 128.

Status Affected: Sign, Zero, Error Field (overflow)



LOG

32-BIT FLOATING-POINT COMMON LOGARITHM

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	1	0	0	0

Hex Coding: 88 with sr = 1
08 with sr = 0

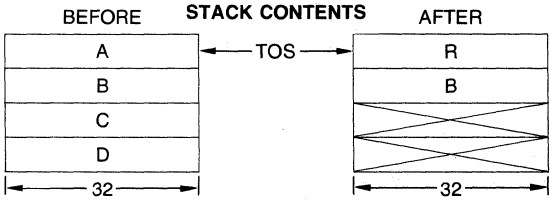
Execution Time: 4474 to 7132 clock cycles for A > 0
20 clock cycles for A $\leq$ 0

Description:
The 32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point common logarithm (base 10) of A. Operands A, C and D are lost. Operand B is unchanged.

The LOG function accepts any positive input data value that can be represented by the data format. If LOG of a non-positive value is attempted an error status of 0100 is returned.

Accuracy: LOG exhibits a maximum absolute error of 2.0×10^{-7} for the input range from 0.1 to 10, and a maximum relative error of 2.0×10^{-7} for positive values less than 0.1 or greater than 10.

Status Affected: Sign, Zero, Error Field



LN

32-BIT FLOATING-POINT NATURAL LOGARITHM

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	1	0	0	1

Hex Coding: 89 with sr = 1
09 with sr = 0

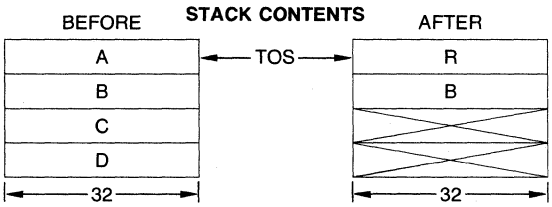
Execution Time: 4298 to 6956 clock cycles for A > 0
20 clock cycles for A$\leq$ 0

Description:
The 32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point natural logarithm (base e) of A. Operands A, C and D are lost. Operand B is unchanged.

The LN function accepts all positive input data values that can be represented by the data format. If LN of a non-positive number is attempted an error status of 0100 is returned.

Accuracy: LN exhibits a maximum absolute error of 2×10^{-7} for the input range from e^{-1} to e, and a maximum relative error of 2.0×10^{-7} for positive values less than e^{-1} or greater than e.

Status Affected: Sign, Zero, Error Field



NOP

NO OPERATION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	0	0	0

Hex Coding: 80 with sr = 1
00 with sr = 0

Execution Time: 4 clock cycles

Description:
The NOP command performs no internal data manipulations. It may be used to set or clear the service request interface line without changing the contents of the stack.

Status Affected: The status byte is cleared to all zeroes.

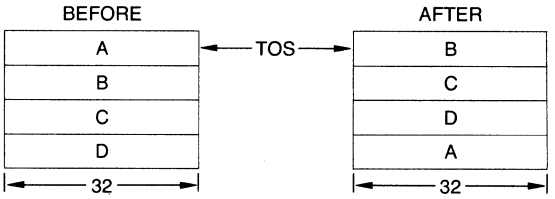
POPD

32-BIT
STACK POP

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	1	0	0	0

Hex Coding: B8 with sr = 1
38 with sr = 0
Execution Time: 12 clock cycles
Description:
The 32-bit stack is moved up so that the old NOS becomes the new TOS. The previous TOS rotates to the bottom of the stack. All operand values are unchanged. POPD and POPF execute the same operation.
Status Affected: Sign, Zero

STACK CONTENTS



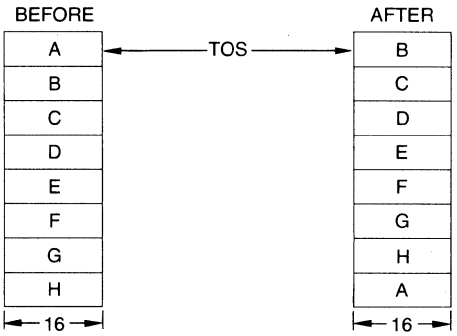
POPS

16-BIT
STACK POP

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	1	0	0	0

Hex Coding: F8 with sr = 1
78 with sr = 0
Execution Time: 10 clock cycles
Description:
The 16-bit stack is moved up so that the old NOS becomes the new TOS. The previous TOS rotates to the bottom of the stack. All operand values are unchanged.
Status Affected: Sign, Zero

STACK CONTENTS



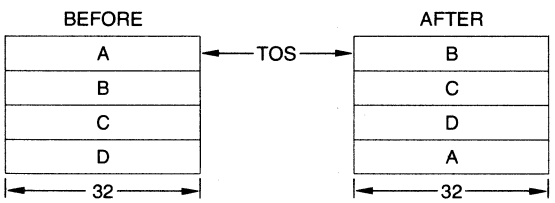
POPF

32-BIT
STACK POP

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	0	0	0

Hex Coding: 98 with sr = 1
18 with sr = 0
Execution Time: 12 clock cycles
Description:
The 32-bit stack is moved up so that the old NOS becomes the new TOS. The old TOS rotates to the bottom of the stack. All operand values are unchanged. POPF and POPD execute the same operation.
Status Affected: Sign, Zero

STACK CONTENTS



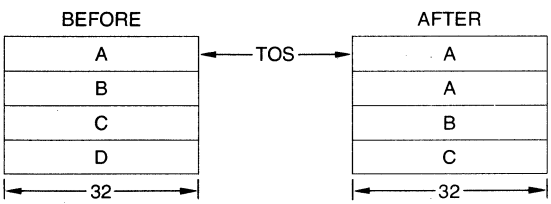
PTOD

PUSH 32-BIT
TOS ONTO STACK

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	0	1	1	1

Hex Coding: B7 with sr = 1
37 with sr = 0
Execution Time: 20 clock cycles
Description:
The 32-bit stack is moved down and the previous TOS is copied into the new TOS location. Operand D is lost. All other operand values are unchanged. PTOD and PTOF execute the same operation.
Status Affected: Sign, Zero

STACK CONTENTS



PTOF

PUSH 32-BIT TOS ONTO STACK

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	1	1	1

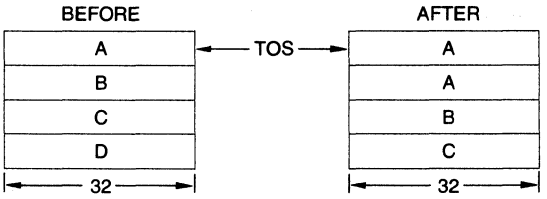
Hex Coding: 97 with sr = 1
17 with sr = 0

Execution Time: 20 clock cycles

Description:
The 32-bit stack is moved down and the previous TOS is copied into the new TOS location. Operand D is lost. All other operand values are unchanged. PTOF and PTOD execute the same operation.

Status Affected: Sign, Zero

STACK CONTENTS



PTOS

PUSH 16-BIT TOS ONTO STACK

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	0	1	1	1

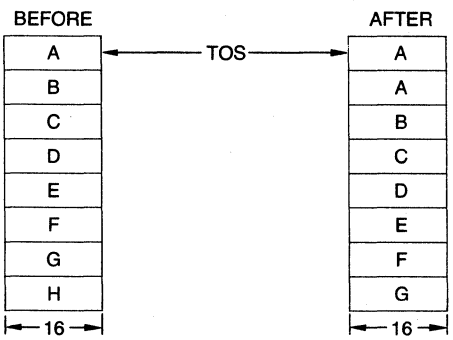
Hex Coding: F7 with sr = 1
77 with sr = 0

Execution Time: 16 clock cycles

Description:
The 16-bit stack is moved down and the previous TOS is copied into the new TOS location. Operand H is lost and all other operand values are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



PUPI

PUSH 32-BIT FLOATING-POINT π

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	0	1	0

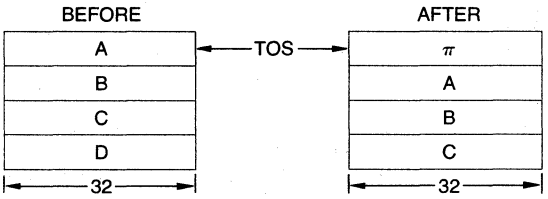
Hex Coding: 9A with sr = 1
1A with sr = 0

Execution Time: 16 clock cycles

Description:
The 32-bit stack is moved down so that the previous TOS occupies the new NOS location. 32-bit floating-point constant π is entered into the new TOS location. Operand D is lost. Operands A, B and C are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



PWR

32-BIT FLOATING-POINT X^Y

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	1	0	1	1

Hex Coding: 8B with sr = 1
0B with sr = 0

Execution Time: 8290 to 12032 clock cycles

Description:

32-bit floating-point operand B at the NOS is raised to the power specified by the 32-bit floating-point operand A at the TOS. The result R of B^A replaces B and the stack is moved up so that R occupies the TOS. Operands A, B, and D are lost. Operand C is unchanged.

The PWR function accepts all input data values that can be represented in the data format for operand A and all positive values for operand B. If operand B is non-positive an error status of 0100 will be returned. The EXP and LN functions are used to implement PWR using the relationship $B^A = EXP\ [A(LN\ B)]$. Thus if the term $[A(LN\ B)]$ is outside the range of $-1.0 \times 2^{+5}$ to $+1.0 \times 2^{+5}$ an error status of 1100 will be returned. Underflow and overflow conditions can occur.

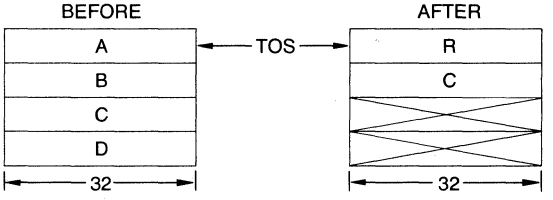
Accuracy: The error performance for PWR is a function of the LN and EXP performance as expressed by:

$|(\text{Relative Error})_{PWR}| = |(\text{Relative Error})_{EXP} + |A(\text{Absolute Error})_{LN}|$

The maximum relative error for PWR occurs when A is at its maximum value while $[A(LN\ B)]$ is near 1.0×2^5 and the EXP error is also at its maximum. For most practical applications the relative error for PWR will be less than 7.0×10^{-7}.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



SADD

16-BIT FIXED-POINT ADD

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	0	1	1	0	0

Hex Coding: EC with sr = 1
6C with sr = 0

Execution Time: 16 to 18 clock cycles

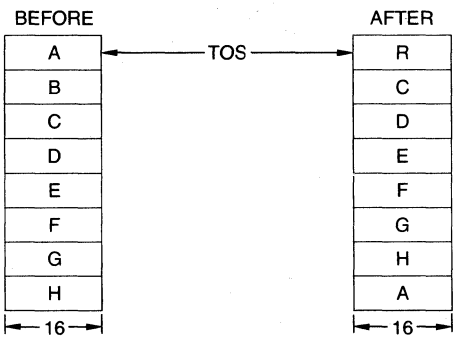
Description:

16-bit fixed-point two's complement integer operand A at the TOS is added to 16-bit fixed-point two's complement integer operand B at the NOS. The result R replaces B and the stack is moved up so that R occupies the TOS. Operand B is lost. All other operands are unchanged.

If the addition generates a carry bit it is reported in the status register. If an overflow occurs it is reported in the status register and the 16 least significant bits of the result are returned.

Status Affected: Sign, Zero, Carry, Error Field

STACK CONTENTS



10

SDIV

16-BIT FIXED-POINT DIVIDE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	0	1	1	1	1

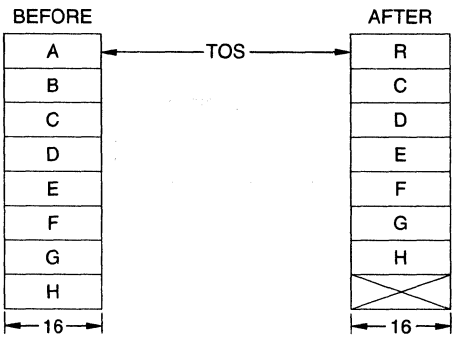
Hex Coding: EF with sr = 1
6F with sr = 0

Execution Time: 84 to 94 clock cycles for A ≠ 0
14 clock cycles for A = 0

Description:
16-bit fixed-point two's complement integer operand B at the NOS is divided by 16-bit fixed-point two's complement integer operand A at the TOS. The 16-bit integer quotient R replaces B and the stack is moved up so that R occupies the TOS. No remainder is generated. Operands A and B are lost. All other operands are unchanged.
If A is zero, R will be set equal to B and the divide-by-zero error status will be reported.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



SIN

32-BIT FLOATING-POINT SINE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	0	1	0

Hex Coding: 82 with sr = 1
02 with sr = 0

Execution Time: 3796 to 4808 clock cycles for $|A| > 2^{-12}$ radians
30 clock cycles for $|A| \leq 2^{-12}$ radians

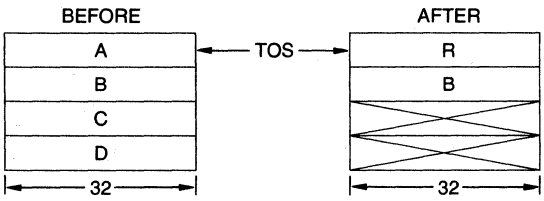
Description:
The 32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point sine of A. A is assumed to be in radians. Operands A, C and D are lost. Operand B is unchanged.

The SIN function will accept any input data value that can be represented by the data format. All input values are range reduced to fall within the interval $-\pi/2$ to $+\pi/2$ radians.

Accuracy: SIN exhibits a maximum relative error of 5.0×10^{-7} for input values in the range of -2π to $+2\pi$ radians.

Status Affected: Sign, Zero

STACK CONTENTS



SMUL

16-BIT FIXED-POINT MULTIPLY, LOWER

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	0	1	1	1	0

Hex Coding: EE with sr = 1
6E with sr = 0

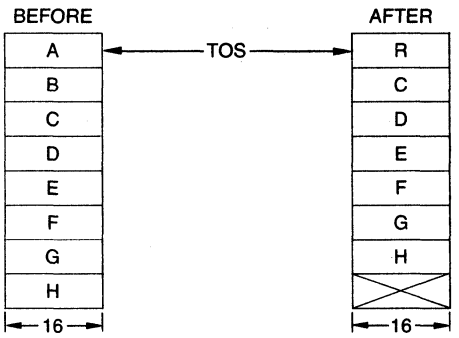
Execution Time: 84 to 94 clock cycles

Description:

16-bit fixed-point two's complement integer operand A at the TOS is multiplied by the 16-bit fixed-point two's complement integer operand B at the NOS. The 16-bit least significant half of the product R replaces B and the stack is moved up so that R occupies the TOS. The most significant half of the product is lost. Operands A and B are lost. All other operands are unchanged. The overflow status bit is set if the discarded upper half was non-zero. If either A or B is the most negative value that can be represented in the format, that value is returned as R and the overflow status is set.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



SMUU

16-BIT FIXED-POINT MULTIPLY, UPPER

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	0	1	1	0

Hex Coding: F6 with sr = 1
76 with sr. = 0

Execution Time: 80 to 98 clock cycles

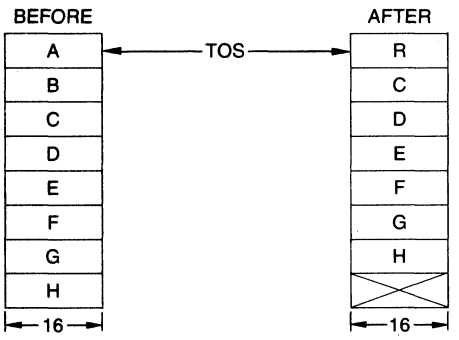
Description:

16-bit fixed-point two's complement integer operand A at the TOS is multiplied by the 16-bit fixed-point two's complement integer operand B at the NOS. The 16-bit most significant half of the product R replaces B and the stack is moved up so that R occupies the TOS. The least significant half of the product is lost. Operands A and B are lost. All other operands are unchanged.

If either A or B is the most negative value that can be represented in the format, that value is returned as R and the overflow status is set.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



SQRT

32-BIT FLOATING-POINT SQUARE ROOT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	0	0	1

Hex Coding: 81 with sr = 1
01 with sr = 0

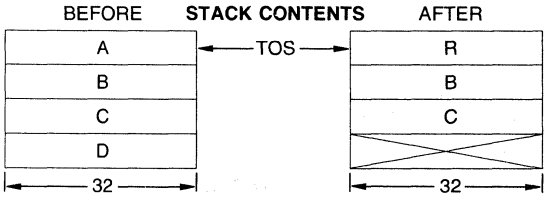
Execution Time: 782 to 870 clock cycles

Description:

32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point square root of A. Operands A and D are lost. Operands B and C are not changed.

SQRT will accept any non-negative input data value that can be represented by the data format. If A is negative an error code of 0100 will be returned in the status register.

Status Affected: Sign, Zero, Error Field



SSUB

16-BIT FIXED-POINT SUBTRACT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	0	1	1	0	1

Hex Coding: ED with sr = 1
6D with sr = 0

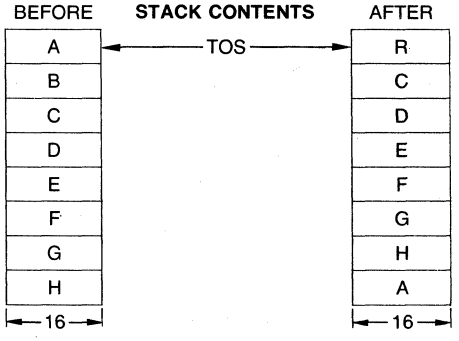
Execution Time: 30 to 32 clock cycles

Description:

16-bit fixed-point two's complement integer operand A at the TOS is subtracted from 16-bit fixed-point two's complement integer operand B at the NOS. The result R replaces B and the stack is moved up so that R occupies the TOS. Operand B is lost. All other operands are unchanged.

If the subtraction generates a borrow it is reported in the carry status bit. If A is the most negative value that can be represented in the format the overflow status is set. If the result cannot be represented in the format range, the overflow status is set and the 16 least significant bits of the result are returned as R.

Status Affected: Sign, Zero, Carry, Error Field



TAN

32-BIT FLOATING-POINT TANGENT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	1	0	0

Hex Coding: 84 with sr = 1
04 with sr = 0

Execution Time: 4894 to 5886 clock cycles for $|A| > 2^{-12}$ radians
30 clock cycles for $|A| \leq 2^{-12}$ radians

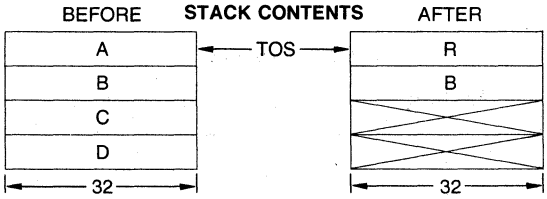
Description:

The 32-bit floating-point operand A at the TOS is replaced by the 32-bit floating-point tangent of A. Operand A is assumed to be in radians. A, C and D are lost. B is unchanged.

The TAN function will accept any input data value that can be represented in the data format. All input data values are range-reduced to fall within $-\pi/4$ to $+\pi/4$ radians. TAN is unbounded for input values near odd multiples of $\pi/2$ and in such cases the overflow bit is set in the status register. For angles smaller than 2^{-12} radians, TAN returns A as the tangent of A.

Accuracy: TAN exhibits a maximum relative error of 5.0×10^{-7} for input data values in the range of -2π to $+2\pi$ radians except for data values near odd multiples of $\pi/2$.

Status Affected: Sign, Zero, Error Field (overflow)



XCHD

EXCHANGE 32-BIT STACK OPERANDS

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	1	0	0	1

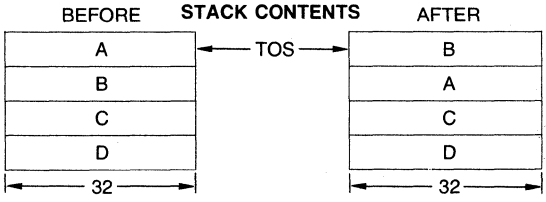
Hex Coding: B9 with sr = 1
39 with sr = 0

Execution Time: 26 clock cycles

Description:

32-bit operand A at the TOS and 32-bit operand B at the NOS are exchanged. After execution, B is at the TOS and A is at the NOS. All operands are unchanged. XCHD and XCHF execute the same operation.

Status Affected: Sign, Zero



XCHF

EXCHANGE 32-BIT STACK OPERANDS

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	0	0	1

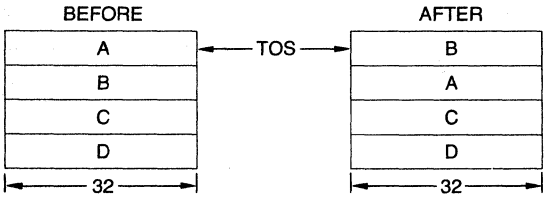
Hex Coding: 99 with sr = 1
19 with sr = 0

Execution Time: 26 clock cycles

Description:
32-bit operand A at the TOS and 32-bit operand B at the NOS are exchanged. After execution, B is at the TOS and A is at the NOS. All operands are unchanged. XCHD and XCHF execute the same operation.

Status Affected: Sign, Zero

STACK CONTENTS



XCHS

EXCHANGE 16-BIT STACK OPERANDS

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	1	0	0	1

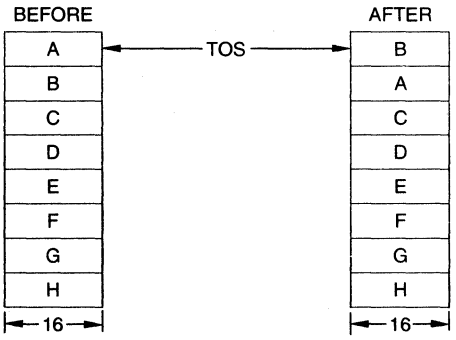
Hex Coding: F9 with sr = 1
79 with sr = 0

Execution Time: 18 clock cycles

Description:
16-bit operand A at the TOS and 16-bit operand B at the NOS are exchanged. After execution, B is at the TOS and A is at the NOS. All operand values are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



10

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VDD with Respect to VSS	−0.5V to +15.0V
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	2.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VSS	VCC	VDD
Am9511ADC	$0°C \leq T_A \leq +70°C$	0V	+5.0V ±5%	+12V ±5%
Am9511ADM	$-55°C \leq T_A \leq +125°C$	0V	+5.0V ±10%	+12V ±10%

ELECTRICAL CHARACTERISTICS Over Operating Range (Note 1)

Parameters	Description	Test Conditions	Min.	Typ.	Max.	Units
VOH	Output HIGH Voltage	IOH = −200μA	3.7			Volts
VOL	Output LOW Voltage	IOL = 3.2mA			0.4	Volts
VIH	Input HIGH Voltage		2.0		VCC	Volts
VIL	Input LOW Voltage		−0.5		0.8	Volts
IIX	Input Load Current	VSS ≤ VI ≤ VCC			±10	μA
IOZ	Data Bus Leakage	VO = 0.4V			10	μA
		VO = VCC			10	
ICC	VCC Supply Current	T_A = +25°C		50	90	mA
		T_A = 0°C			95	
		T_A = −55°C			100	
IDD	VDD Supply Current	T_A = +25°C		50	90	mA
		T_A = 0°C			95	
		T_A = −55°C			100	
CO	Output Capacitance	fc = 1.0MHz, Inputs = 0V		8	10	pF
CI	Input Capacitance			5	8	pF
CIO	I/O Capacitance			10	12	pF

SWITCHING CHARACTERISTICS over operating range (Notes 2, 3)

Parameters	Description		Am9511A Min.	Am9511A Max.	Am9511A-1 Min.	Am9511A-1 Max.	Units
TAPW	$\overline{EACK}$ LOW Pulse Width		100		75		ns
TCDR	C/$\overline{D}$ to $\overline{RD}$ LOW Set up Time		0		0		ns
TCDW	C/$\overline{D}$ to $\overline{WR}$ LOW Set up Time		0		0		ns
TCPH	Clock Pulse HIGH Width		200		140		ns
TCPL	Clock Pulse LOW Width		240		160		ns
TCSR	$\overline{CS}$ LOW to $\overline{RD}$ LOW Set up Time		0		0		ns
TCSW	$\overline{CS}$ LOW to $\overline{WR}$ LOW Set up Time		0		0		ns
TCY	Clock Period		480	5000	320	3300	ns
TDW	Data Bus Stable to $\overline{WR}$ HIGH Set up Time			150		100 (Note 9)	ns
TEAE	$\overline{EACK}$ LOW to $\overline{END}$ HIGH Delay			200		175	ns
TEPW	$\overline{END}$ LOW Pulse Width (Note 4)		400		300		ns
TOP	Data Bus Output Valid to $\overline{PAUSE}$ HIGH Delay		0		0		ns
TPPWR	$\overline{PAUSE}$ LOW Pulse Width Read (Note 5)	Data	3.5TCY+50	5.5TCY+300	3.5TCY+50	5.5TCY+200	ns
		Status	1.5TCY+50	3.5TCY+300	1.5TCY+50	3.5TCY+200	
TPPWW	$\overline{PAUSE}$ LOW Pulse Width Write (Note 8)			50		50	ns
TPR	$\overline{PAUSE}$ HIGH to $\overline{RD}$ HIGH Hold Time		0		0		ns
TPW	$\overline{PAUSE}$ HIGH to $\overline{WR}$ HIGH Hold Time		0		0		ns
TRCD	$\overline{RD}$ HIGH to C/$\overline{D}$ Hold Time		0		0		ns
TRCS	$\overline{RD}$ HIGH to $\overline{CS}$ HIGH Hold Time		0		0		ns
TRO	$\overline{RD}$ LOW to Data Bus ON Delay		50		50		ns
TRP	$\overline{RD}$ LOW to $\overline{PAUSE}$ LOW Delay (Note 6)			150		100 (Note 9)	ns
TRZ	$\overline{RD}$ HIGH to Data Bus OFF Delay		50	200	50	150	ns
TSAPW	$\overline{SVACK}$ LOW Pulse Width		100		75		ns
TSAR	$\overline{SVACK}$ LOW to SVREQ LOW Delay			300		200	ns
TWCD	$\overline{WR}$ HIGH to C/$\overline{D}$ Hold Time		60		30		ns
TWCS	$\overline{WR}$ HIGH to $\overline{CS}$ HIGH Hold Time		60		30		ns
TWD	$\overline{WR}$ HIGH to Data Bus Hold Time		20		20		ns
TWI	Write Inactive Time (Note 8)	Command	3TCY		3TCY		ns
		Data	4TCY		4TCY		
TWP	$\overline{WR}$ LOW to $\overline{PAUSE}$ LOW Delay (Note 6)			150		100 (Note 9)	ns

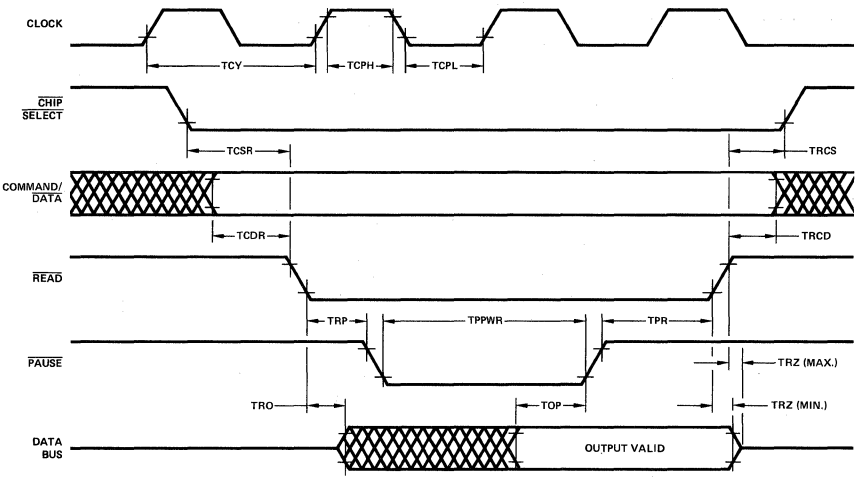
NOTES

1. Typical values are for $T_A = 25°C$, nominal supply voltages and nominal processing parameters.
2. Switching parameters are listed in alphabetical order.
3. Test conditions assume transition times of 20ns or less, output loading of one TTL gate plus 100pF and timing reference levels of 0.8V and 2.0V.
4. $\overline{END}$ low pulse width is specified for $\overline{EACK}$ tied to VSS. Otherwise TEAE applies.
5. Minimum values shown assume no previously entered command is being executed for the data access. If a previously entered command is being executed, $\overline{PAUSE}$ LOW Pulse Width is the time to complete execution plus the time shown. Status may be read at any time without exceeding the time shown.
6. $\overline{PAUSE}$ is pulled low for both command and data operations.
7. TEX is the execution time of the current command (see the Command Execution Times table).
8. $\overline{PAUSE}$ low pulse width is less than 50ns when writing into the data port or the control port as long as the duty cycle requirement (TWI) is observed and no previous command is being executed. TWI may be safely violated as long as the extended TPPWW that results is observed. If a previously entered command is being executed, $\overline{PAUSE}$ LOW Pulse Width is the time to complete execution plus the time shown.
9. 150ns for Am9511A-1DM.

10

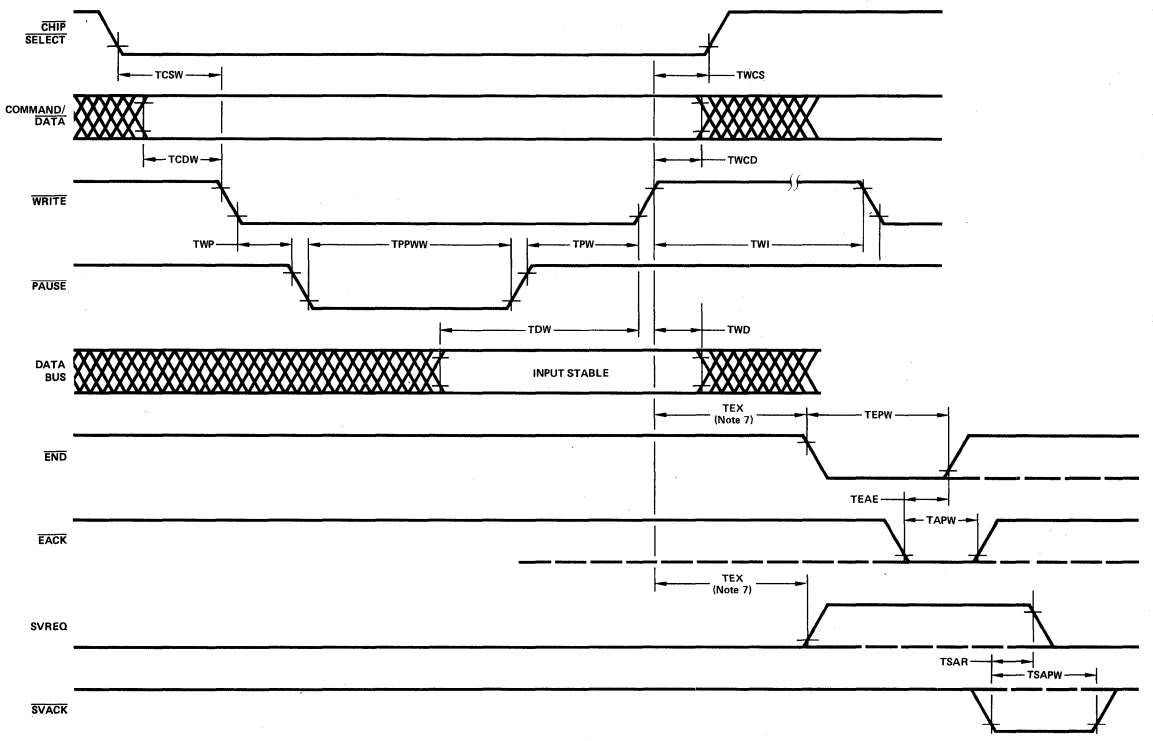
SWITCHING WAVEFORMS

READ OPERATIONS



MOS-048

WRITE OPERATIONS



MOS-049

APPLICATION INFORMATION

The diagram in Figure 2 shows the interface connections for the Am9511A APU with operand transfers handled by an Am9517 DMA controller, and CPU coordination handled by an Am9519 Interrupt Controller. The APU interrupts the CPU to indicate that a command has been completed. When the performance enhancements provided by the DMA and Interrupt operations are not required, the APU interface can be simplified as shown in Figure 1. The Am9511A APU is designed with a general purpose 8-bit data bus and interface control so that it can be conveniently used with any general 8-bit processor.

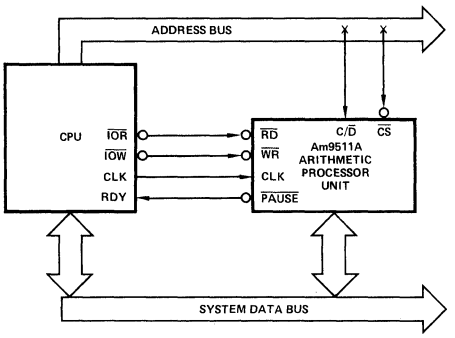
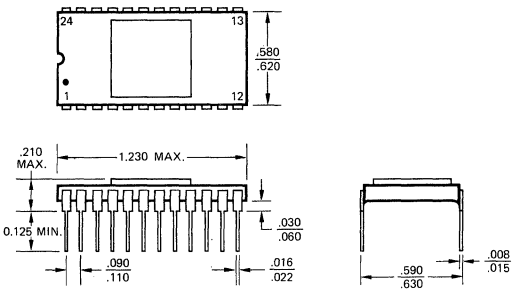


Figure 1. Am9511A Minimum Configuration Example.

MOS-050

PHYSICAL DIMENSIONS
Dual-In-Line

24-Pin Side-Brazed



10

Am9512
Floating-Point Processor

DISTINCTIVE CHARACTERISTICS

- Single (32-bit) and double (64-bit) precision capability
- Add, subtract, multiply and divide functions
- Compatible with proposed IEEE format
- Easy interfacing to microprocessors
- 8-bit data bus
- Standard 24-pin package
- 12V and 5V power supplies
- Stack oriented operand storage
- Direct memory access or programmed I/O Data Transfers
- End of execution signal
- Error interrupt
- All inputs and outputs TTL level compatible
- Advanced N-channel silicon gate MOS technology
- 100% MIL-STD-883 reliability assurance testing

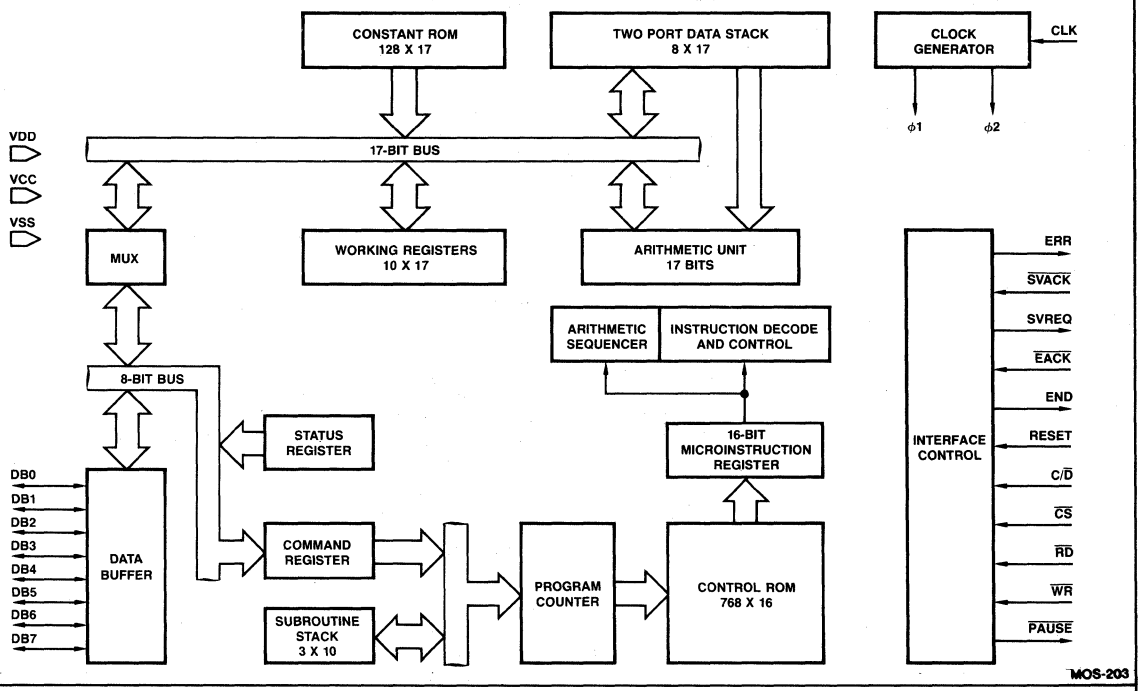
GENERAL DESCRIPTION

The Am9512 is a high performance floating-point processor unit (FPU). It provides single precision (32-bit) and double precision (64-bit) add, subtract, multiply and divide operations. It can be easily interfaced to enhance the computational capabilities of the host microprocessor.

The operand, result, status and command information transfers take place over an 8-bit bidirectional data bus. Operands are pushed onto an internal stack by the host processor and a command is issued to perform an operation on the data stack. The results of this operation are available to the host processor by popping the stack.

Information transfers between the Am9512 and the host processor can be handled by using programmed I/O or direct memory access techniques. After completing an operation, the Am9512 activates an "end of execution" signal that can be used to interrupt the host processor.

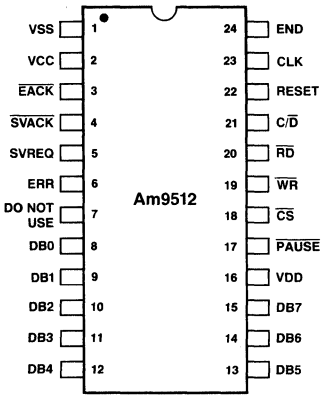
BLOCK DIAGRAM



ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency	
		2MHz	3MHz
Hermetic DIP	$0°C \leq T_A \leq 70°C$	AM9512DC	AM9512-1DC
	$-55°C \leq T_A \leq +125°C$	AM9512DM	AM9512-1DM

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

MOS-204

INTERFACE SIGNAL DESCRIPTION

VCC: +5V Power Supply

VDD: +12V Power Supply

VSS: Ground

CLK (Clock, Input)

An external timing source connected to the CLK input provides the necessary clocking.

RESET (Reset, Input)

A HIGH on this input causes initialization. Reset terminates any operation in progress, and clears the status register to zero. The internal stack pointer is initialized and the contents of the stack may be affected. After a reset the END output, the ERR output and the SVREQ output will be LOW. For proper initialization, RESET must be HIGH for at least five CLK periods following stable power supply voltages and stable clock.

C/$\overline{D}$ (Command/Data Select, Input)

The C/$\overline{D}$ input together with the $\overline{RD}$ and $\overline{WR}$ inputs determines the type of transfer to be performed on the data bus as follows:

C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	Function
L	H	L	Push data byte into the stack
L	L	H	Pop data byte from the stack
H	H	L	Enter command
H	L	H	Read Status
X	L	L	Undefined

L = LOW
H = HIGH
X = DON'T CARE

END (End of Execution, Output)

A HIGH on this output indicates that execution of the current command is complete. This output will be cleared LOW by activating the $\overline{EACK}$ input LOW or performing any read or write operation or device initialization using the RESET. If $\overline{EACK}$ is tied LOW, the END output will be a pulse (see $\overline{EACK}$ description).

Reading the status register while a command execution is in progress is allowed. However any read or write operation clears the flip-flop that generates the END output. Thus such continuous reading could conflict with internal logic setting of the END flip-flop at the end of command execution.

$\overline{EACK}$ (End Acknowledge, Input)

This input when LOW makes the END output go LOW. As mentioned earlier HIGH on the END output signals completion of a command execution. The END signal is derived from an internal flip-flop which is clocked at the completion of a command. This flip-flop is clocked to the reset state when $\overline{EACK}$ is LOW. Consequently, if $\overline{EACK}$ is tied LOW, the END output will be a pulse that is approximately one CLK period wide.

SVREQ (Service Request, Output)

A HIGH on this output indicates completion of a command. In this sense this output is the same as the END output. However, the SVREQ output will go HIGH at the completion of a command. This bit must be 1 for SVREQ to go HIGH. The SVREQ can be cleared (i.e., go LOW) by activating the $\overline{SVACK}$ input LOW or initializing the device using the RESET. Also, the SVREQ will be automatically cleared after completion of any command that has the service request bit as 0.

$\overline{SVACK}$ (Service Acknowledge, Input)

A LOW on this input clears SVREQ. If the $\overline{SVACK}$ input is permanently tied LOW, it will conflict with the internal setting of the SVREQ output. Thus the SVREQ indication cannot be relied upon if the $\overline{SVACK}$ is tied LOW.

DB0-DB7 (Data Bus, Input/Output)

These eight bidirectional lines are used to transfer command, status and operand information between the device and the host processor. DB0 is the least significant and DB7 is the most significant bit position. HIGH on a data bus line corresponds to 1 and LOW corresponds to 0.

When pushing operands on the stack using the data bus, the least significant byte must be pushed first and most significant byte last. When popping the stack to read the result of an operation, the most significant byte will be available on the data bus first and the least significant byte will be the last. Moreover, for pushing operands and popping results, the number of transactions must be equal to the proper number of bytes appropriate for the chosen format. Otherwise, the internal byte pointer will not be aligned properly. The Am9512 single precision format requires 4 bytes and double precision format requires 8 bytes.

ERR (Error, Output)

This output goes HIGH to indicate that the current command execution resulted in an error condition. The error conditions are: attempt to divide by zero, exponent overflow and exponent underflow. The ERR output is cleared LOW on read status register operation or upon RESET.

The ERR output is derived from the error bits in the status register. These error bits will be updated internally at an appropriate time during a command execution. Thus ERR output going HIGH may not correspond with the completion of a command. Reading of the status register can be performed while a command execution is in progress. However it should be noted that reading the status register clears the ERR output. Thus reading the status register while a command execution in progress may result in an internal conflict with the ERR output.

10

$\overline{CS}$ (Chip Select, Input)

This input must be LOW to accomplish any read or write operation to the Am9512.

To perform a write operation, appropriate data is presented on DB0 through DB7 lines, appropriate logic level on the C/$\overline{D}$ input and the $\overline{CS}$ input is made LOW. Whenever $\overline{WR}$ and $\overline{RD}$ inputs are both HIGH and $\overline{CS}$ is LOW, $\overline{PAUSE}$ goes LOW. However actual writing into the Am9512 cannot start until $\overline{WR}$ is made LOW. After initiating the write operation by the HIGH to LOW transition on the $\overline{WR}$ input, the $\overline{PAUSE}$ output will go HIGH indicating the write operation has been acknowledged. The $\overline{WR}$ input can go HIGH after $\overline{PAUSE}$ goes HIGH. The data lines, C/$\overline{D}$ input and the $\overline{CS}$ input can change when appropriate hold time requirements are satisfied. See write timing diagram for details.

To perform a read operation an appropriate logic level is established on the C/$\overline{D}$ input and $\overline{CS}$ is made LOW. The $\overline{PAUSE}$ output goes LOW because $\overline{WR}$ and $\overline{RD}$ inputs are HIGH. The read operation does not start until the $\overline{RD}$ input goes LOW. $\overline{PAUSE}$ will go HIGH indicating that read operation is complete and the required information is available on the DB0 through DB7 lines. This information will remain on the data lines as long as $\overline{RD}$ is LOW. The $\overline{RD}$ input can return HIGH anytime after $\overline{PAUSE}$ goes HIGH. The $\overline{CS}$ input and C/$\overline{D}$ input can change anytime after $\overline{RD}$ returns HIGH. See read timing diagram for details. If the $\overline{CS}$ is tied LOW permanently, $\overline{PAUSE}$ will remain LOW until the next Am9512 read or write access.

$\overline{RD}$ (Read, Input)

A LOW on this input is used to read information from an internal location and gate that information onto the data bus. The $\overline{CS}$ input must be LOW to accomplish the read operation. The C/$\overline{D}$ input determines what internal location is of interest. See C/$\overline{D}$, $\overline{CS}$ input descriptions and read timing diagram for details. If the END output was HIGH, performing any read operation will make the END output go LOW after the HIGH to LOW transition of the $\overline{RD}$ input (assuming $\overline{CS}$ is LOW). If the ERR output was HIGH performing a status register read operation will make the ERR output LOW. This will happen after the HIGH to LOW transition of the RD input (assuming $\overline{CS}$ is LOW).

$\overline{WR}$ (Write, Input)

A LOW on this input is used to transfer information from the data bus into an internal location. The $\overline{CS}$ must be LOW to accomplish the write operation. The C/$\overline{D}$ determines which internal location is to be written. See C/$\overline{D}$, $\overline{CS}$ input descriptions and write timing diagram for details.

If the END output was HIGH, performing any write operation will make the END output go LOW after the LOW to HIGH transition of the $\overline{WR}$ input (assuming $\overline{CS}$ is LOW).

$\overline{PAUSE}$ (Pause, Output)

This output is a handshake signal used while performing read or write transactions with the Am9512. If the $\overline{WR}$ and $\overline{RD}$ inputs are both HIGH, the $\overline{PAUSE}$ output goes LOW with the $\overline{CS}$ input in anticipation of a transaction. If $\overline{WR}$ goes LOW to initiate a write transaction with proper signals established on the DB0-DB7, C/$\overline{D}$ inputs, the $\overline{PAUSE}$ will return HIGH indicating that the write operation has been accomplished. The $\overline{WR}$ can be made HIGH after this event. On the other hand, if a read operation is desired, the $\overline{RD}$ input is made LOW after activating $\overline{CS}$ LOW and establishing proper C/$\overline{D}$ input. (The $\overline{PAUSE}$ will go LOW in response to $\overline{CS}$ going LOW.) The $\overline{PAUSE}$ will return HIGH indicating completion of read. The $\overline{RD}$ can return HIGH after this event. It should be noted that a read or write operation can be initiated without any regard to whether a command execution is in progress or not. Proper device operation is assured by obeying the $\overline{PAUSE}$ output indication as described.

FUNCTIONAL DESCRIPTION

Major functional units of the Am9512 are shown in the block diagram. The Am9512 employs a microprogram controlled stack oriented architecture with 17-bit wide data paths.

The Arithmetic Unit receives one of its operands from the Operand Stack. This stack is an eight word by 17-bit two port memory with last in – first out (LIFO) attributes. The second operand to the Arithmetic Unit is supplied by the internal 17-bit bus. In addition to supplying the second operand, this bidirectional bus also carries the results from the output of the Arithmetic Unit when required. Writing into the Operand Stack takes place from this internal 17-bit bus when required. Also connected to this bus are the Constant ROM and Working Registers. The ROM provides the required constants to perform the mathematical operations while the Working Registers provide storage for the intermediate values during command execution.

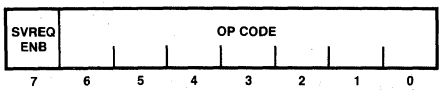
Communication between the external world and the Am9512 takes place on eight bidirectional input/output lines, DB0 through DB7 (Data Bus). These signals are gated to the internal 8-bit bus through appropriate interface and buffer circuitry. Multiplexing facilities exist for bidirectional communication between the internal eight and 17-bit buses. The Status Register and Command Register are also located on the 8-bit bus.

The Am9512 operations are controlled by the microprogram contained in the Control ROM. The Program Counter supplies the microprogram addresses and can be partially loaded from the Command Register. Associated with the Program Counter is the Subroutine Stack where return addresses are held during subroutine calls in the microprogram. The Microinstruction Register holds the current microinstruction being executed. The register facilitates pipelined microprogram execution. The Instruction Decode logic generates various internal control signals needed for the Am9512 operation.

The Interface Control logic receives several external inputs and provides handshake related outputs to facilitate interfacing the Am9512 to microprocessors.

COMMAND FORMAT

The Operation of the Am9512 is controlled from the host processor by issuing instructions called commands. The command format is shown below:



The command consists of 8 bits; the least significant 7 bits specify the operation to be performed as detailed in the accompanying table. The most significant bit is the Service Request Enable bit. This bit must be a 1 if SVREQ is to go high at end of executing a command.

The Am9512 commands fall into three categories: Single precision arithmetic, double precision arithmetic and data manipulation. There are four arithmetic operations that can be performed with single precision (32-bit), or double precision (64-bit) floating-point numbers: add, subtract, multiply and divide. These operations require two operands. The Am9512 assumes that these operands are located in the internal stack as Top of Stack

(TOS) and Next on Stack (NOS). The result will always be returned to the previous NOS which becomes the new TOS. Results from an operation are of the same precision and format as the operands. The results will be rounded to preserve the accuracy. The actual data formats and rounding procedures are described in a later section. In addition to the arithmetic operations, the Am9512 implements eight data manipulating operations. These include changing the sign of a double or single precision operand located in TOS, exchanging single precision operands located at TOS and NOS, as well as copying and popping single or double precision operands. See also the sections on status register and operand formats.

The Execution times of the Am9512 commands are all data dependent. Table 2 shows one example of each command execution time:

Table 1. Command Decoding Table.

Command Bits								Mnemonic	Description
7	6	5	4	3	2	1	0		
X	0	0	0	0	0	0	1	SADD	Add TOS to NOS Single Precision and result to NOS. Pop stack.
X	0	0	0	0	0	1	0	SSUB	Subtract TOS from NOS Single Precision and result to NOS. Pop stack.
X	0	0	0	0	0	1	1	SMUL	Multiply NOS by TOS Single Precision and result to NOS. Pop stack.
X	0	0	0	0	1	0	0	SDIV	Divide NOS by TOS Single Precision and result to NOS. Pop stack.
X	0	0	0	0	1	0	1	CHSS	Change sign of TOS Single Precision operand.
X	0	0	0	0	1	1	0	PTOS	Push Single Precision operand on TOS to NOS.
X	0	0	0	0	1	1	1	POPS	Pop Single Precision operand from TOS. NOS becomes TOS.
X	0	0	0	1	0	0	0	XCHS	Exchange TOS with NOS Single Precision.
X	0	1	0	1	1	0	1	CHSD	Change sign of TOS Double Precision operand.
X	0	1	0	1	1	1	0	PTOD	Push Double Precision operand on TOS to NOS.
X	0	1	0	1	1	1	1	POPD	Pop Double Precision operand from TOS. NOS becomes TOS.
X	0	0	0	0	0	0	0	CLR	CLR status.
X	0	1	0	1	0	0	1	DADD	Add TOS to NOS Double Precision and result to NOS. Pop stack.
X	0	1	0	1	0	1	0	DSUB	Subtract TOS from NOS Double Precision and result to NOS. Pop stack.
X	0	1	0	1	0	1	1	DMUL	Multiply NOS by TOS Double Precision and result to NOS. Pop stack.
X	0	1	0	1	1	0	0	DDIV	Divide NOS by TOS Double Precision and result to NOS. Pop Stack.

Notes: X = Don't Care Operation for bit combinations not listed above is undefined.

Table 2. Am9512 Execution Time in Cycles.

Single Precision

	Min	Typ	Max
Add	58	220	512
Subtract	56	220	512
Multiply	192	220	254
Divide	228	240	264

Double Precision

	Min	Typ	Max
Add	578	1200	3100
Subtract	578	1200	3100
Multiply	1720	1770	1860
Divide	4560	4920	5120

Note: Typical for add and subtract assumes the operands are within six decimal orders of magnitude. Max is derived from the maximum execution time of 1000 executions with random 32-bit or 64-bit patterns.

Table 3. Some Execution Examples.

Command	TOS	NOS	Result	Clock periods
SADD	3F800000	3F800000	40000000	58
SSUB	3F800000	3F800000	00000000	56
SMUL	40400000	3FC00000	40900000	198
SDIV	3F800000	40000000	3F000000	228
CHSS	3F800000	–	BF800000	10
PTOS	3F800000	–	–	16
POPS	3F800000	–	–	14
XCHS	3F800000	4000000	–	26
CHSD	3FF0000000000000	–	BFF0000000000000	24
PTOD	3FF0000000000000	–	–	40
POPD	3FF0000000000000	–	–	26
CLR	3FF0000000000000	–	–	4
DADD	3FF00000A0000000	8000000000000000	3FF00000A0000000	578
DSUB	3FF00000A0000000	8000000000000000	3FF00000A0000000	578
DMUL	BFF8000000000000	3FF8000000000000	C002000000000000	1748
DDIV	BFF8000000000000	3FF8000000000000	BFF0000000000000	4560

Note: TOS, NOS and Result are in hexadecimal; Clock period is in decimal.

COMMAND INITIATION

After properly positioning the required operands in the stack, a command may be issued. The procedure for initiating a command execution is as follows:

1. Establish appropriate command on the DB0-DB7 lines.
2. Establish HIGH on the C/$\overline{D}$ input.
3. Establish LOW on the $\overline{CS}$ input. Whenever $\overline{WR}$ and $\overline{RD}$ inputs are HIGH the $\overline{PAUSE}$ output follows the $\overline{CS}$ input. Hence $\overline{PAUSE}$ will become LOW.
4. Establish LOW on the $\overline{WR}$ input after an appropriate set up time (see timing diagrams).
5. Sometime after the HIGH to LOW level transition of $\overline{WR}$ input, the $\overline{PAUSE}$ output will become HIGH to acknowledge the write operation. The $\overline{WR}$ input can return to HIGH anytime after $\overline{PAUSE}$ goes HIGH. The DB0-DB7, C/$\overline{D}$ and $\overline{CS}$ inputs are allowed to change after the hold time requirements are satisfied (see timing diagram).

An attempt to issue a new command while the current command execution is in progress is allowed. Under these circumstances, the $\overline{PAUSE}$ output will not go HIGH until the current command execution is completed.

OPERAND ENTRY

The Am9512 commands operate on the operands located at the TOS and NOS and results are returned to the stack at NOS and then popped to TOS. The operands required for the Am9512 are one of two formats – single precision floating-point (4 bytes) or double precision floating-point (8 bytes). The result of an operation has the same format as the operands. In other words, operations using single precision quantities always result in a single precision result while operations involving double precision quantities will result in double precision result.

Operands are always entered into the stack least significant byte first and most significant byte last. The following procedure must be followed to enter operands into the stack:

1. The lower significant operand byte is established on the DB0-DB7 lines.
2. A LOW is established on the C/$\overline{D}$ input to specify that data is to be entered into the stack.
3. The $\overline{CS}$ input is made LOW. Whenever the $\overline{WR}$ and $\overline{RD}$ inputs are HIGH, the $\overline{PAUSE}$ output will follow the $\overline{CS}$ input. Thus $\overline{PAUSE}$ output will become LOW.
4. After appropriate set up time (see timing diagrams), the $\overline{WR}$ input is made LOW.
5. Sometime after this event, $\overline{PAUSE}$ will return HIGH to indicate that the write operation has been acknowledged.
6. Anytime after the $\overline{PAUSE}$ output goes HIGH the $\overline{WR}$ input can be made HIGH. The DB0-DB7, C/$\overline{D}$ and $\overline{CS}$ inputs can change after appropriate hold time requirements are satisfied (see timing diagrams).

The above procedure must be repeated until all bytes of the operand are pushed into the stack. It should be noted that for single precision operands 4 bytes should be pushed and 8 bytes must be pushed for double precision. Not pushing all the bytes of a quantity will result in byte pointer misalignment.

The Am9512 stack can accommodate 4 single precision quantities or 2 double precision quantities. Pushing more quantities than the capacity of the stack will result in loss of data which is usual with any LIFO stack.

REMOVING THE RESULTS

Result from an operation will be available at the TOS. Results can be transferred from the stack to the data bus by reading the stack.

When the stack is popped for results, the most significant byte is available first and the least significant byte last. A result is always of the same precision as the operands that produced it. Thus when the result is taken from the stack, the total number of bytes popped out should be appropriate with the precision – single precision results are 4 bytes and double precision results are 8 bytes. The following prodedure must be used for reading the result from the stack:

1. A LOW is established on the C/$\overline{D}$ input.
2. The $\overline{CS}$ input is made LOW. When $\overline{WR}$ and $\overline{RD}$ inputs are both HIGH, the $\overline{PAUSE}$ output follows the $\overline{CS}$ input, thus $\overline{PAUSE}$ will be LOW.
3. After appropriate set up time (see timing diagrams), the $\overline{RD}$ input is made LOW.
4. Sometime after this, $\overline{PAUSE}$ will return HIGH indicating that the data is available on the DB0-DB7 lines. This data will remain on the DB0-DB7 lines as long as the $\overline{RD}$ input remains LOW.
5. Anytime after $\overline{PAUSE}$ goes HIGH, the $\overline{RD}$ input can return HIGH to complete transaction.
6. The $\overline{CS}$ and C/$\overline{D}$ inputs can change after appropriate hold time requirements are satisfied (see timing diagram).
7. Repeat this procedure until all bytes appropriate for the precision of the result are popped out.

Reading of the stack does not alter its data; it only adjusts the byte pointer. If more data is popped than the capacity of the stack, the internal byte pointer will wrap around and older data will be read again, consistent with the LIFO stack.

READING STATUS REGISTER

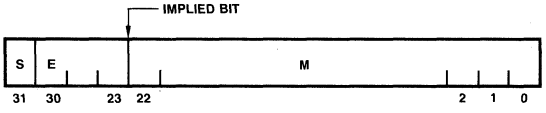
The Am9512 status register can be read without any regard to whether a command is in progress or not. The only implication that has to be considered is the effect this might have on the END and ERR outputs discussed in the signal descriptions.

The following procedure must be followed to accomplish status register reading.

1. Establish HIGH on the C/$\overline{D}$ input.
2. Establish LOW on the $\overline{CS}$ input. Whenever $\overline{WR}$ and $\overline{RD}$ inputs are HIGH, $\overline{PAUSE}$ will follow the $\overline{CS}$ input. Thus, $\overline{PAUSE}$ will go LOW.
3. After appropriate set up time (see timing diagram) $\overline{RD}$ is made LOW.
4. Sometime after the HIGH to LOW transition of $\overline{RD}$, $\overline{PAUSE}$ will become HIGH indicating that status register contents are available on the DB0-DB7 lines. These lines will contain this information as long as $\overline{RD}$ is LOW.
5. The $\overline{RD}$ input can be returned HIGH anytime after $\overline{PAUSE}$ goes HIGH.
6. The C/$\overline{D}$ input and $\overline{CS}$ input can change after satisfying appropriate hold time requirements (see timing diagram).

DATA FORMATS

The Am9512 handles floating-point quantities in two different formats – single precision and double precision. The single precision quantities are 32-bits long as shown below.



Bit 31:
S = Sign of the mantissa. 1 represents negative and 0 represents positive.

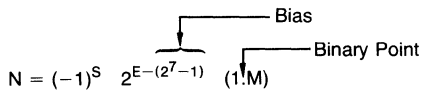
Bits 23-30

E = These 8-bits represent a biased exponent. The bias is $2^7 - 1 = 127$

Bits 0-22

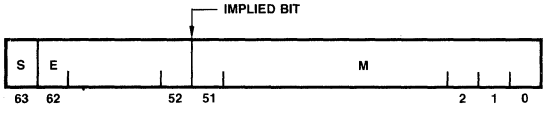
M = 23-bit mantissa. Together with the sign bit, the mantissa represents a signed fraction in sign-magitude notation. There is an implied 1 beyond the most significant bit (bit 22) of the mantissa. In other words, the mantissa is assumed to be a 24-bit normalized quantity and the most significant bit which will always be 1 due to normalization is implied. The Am9512 restores this implied bit internally before performing arithmetic; normalizes the result and strips the implied bit before returning the results to the external data bus. The binary point is between the implied bit and bit 22 of the mantissa.

The quantity N represented by the above notation is



Provided E ≠ 0 or all 1's.

A double precision quantity consists of the mantissa sign bit(s), an 11 bit biased exponent (E), and a 52-bit mantissa (M). The bias for double precision quantities is $2^{10} - 1$. The double precision format is illustrated below.



Bit 63:

S = Sign of the mantissa. 1 represents negative and 0 represents positive.

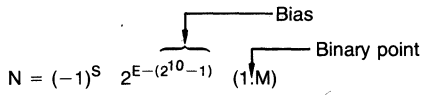
Bits 52-62

E = These 11 bits represent a biased exponent. The bias is $2^{10} - 1 = 1023$.

Bit 0-51

M = 52-bit mantissa. Together with the sign bit, the mantissa represents a signed fraction in sign-magnitude notation. There is an implied 1 beyond the most significant bit (bit 51) of the mantissa. In other words, the mantissa is assumed to a 53-bit normalized quantity and the most significant bit, which will always be a 1 due to normalization, is implied. The Am9512 restores this implied bit internally before performing arithmetic; normalizes the result and strips the implied bit before returning the result to the external data bus. The binary point is between the implied bit and bit 51 of the mantissa.

The quantity N represented by the above notation is



Provided E ≠ 0 or all 1's.

STATUS REGISTER

The Am9512 contains an 8-bit status register with the following format.

BUSY	SIGN S	ZERO Z	RESERVED	DIVIDE EXCEPTION D	EXPONENT UNDERFLOW U	EXPONENT OVERFLOW V	RESERVED
7	6	5	4	3	2	1	0

Bit 0 and bit 4 are reserved. Occurrence of exponent oerflow (V), exponent underflow (U) and divide exception (D) are indicated by bits 1, 2 and 3 respectively. An attempt to divide by zero is the only divide exception. Bits 5 and 6 represent a zero result and the sign of a result respectively. Bit 7 (Busy) of the status register indicates if the Am9512 is currently busy executing a command. All the bits are initialized to zero upon reset. Also, executing a CLR (Clear Status) command will result in all zero status register bits. A zero in Bit 7 indicates that the Am9512 is not busy and a new command may be initiated. As soon as a new command is issued, Bit 7 becomes 1 to indicate the device is busy and remains 1 until the command execution is complete, at which time it will become 0. As soon as a new command is issued, status register bits 0, 1, 2, 3, 4, 5 and 6 are cleared to zero. The status bits will be set as required during the command execution. Hence, as long as bit 7 is 1, the remainder of the status register bit indications should not be relied upon unless the ERR occurs. The following is a detailed status bit description.

Bit 0 Reserved

Bit 1 Exponent overflow (V): When 1, this bit indicates that exponent overflow has occurred. Cleared to zero otherwise.

Bit 2 Exponent Underflow (U): When 1, this bit indicates that exponent underflow has occurred. Cleared to zero otherwise.

Bit 3 Divide Exception (D): When 1, this bit indicates that an attempt to divide by zero is made. Cleared to zero otherwise.

Bit 4 Reserved

Bit 5 Zero (Z): When 1, this bit indicates that the result returned to TOS after a command is all zeros. Cleared to zero otherwise.

Bit 6 Sign (S): When 1, this bit indicates that the result returned to TOS is negative. Cleared to zero otherwise.

Bit 7 Busy: When 1, this bit indicates the Am9512 is in the process of executing a command. It will become zero after the command execution is complete.

All other status register bits are valid when the Busy bit is zero.

ALGORITHMS OF FLOATING-POINT ARITHMETIC

1. Floating Point to Decimal Conversion

   As an introduction to floating-point arithmetic, a brief description of the Decimal equivalent of the Am9512 floating-point format should help the reader to understand and verify the validity of the arithmetic operations. The Am9512 single precision format is used for the following discussions. With a minor modification of the field lengths, the discussion would also apply to the double precision format.

   There are three parts in a floating point number:

   a. The sign – the sign applies to the sign of the number. Zero means the number is positive or zero. One means the number is negative.

b. The exponent – the exponent represents the magnitude of the number. The Am9512 single precision format has an excess 127_{10} notation which means the code representation is 127_{10} higher than the actual value. The following are a few examples of actual versus coded exponent.

Actual	Coded
$+127_{10}$	$+254_{10}$
0	127_{10}
-126_{10}	$+1_{10}$

c. The mantissa – the mantissa is a 23-bit value with the binary point to the left of the most significant bit. There is a hidden 1 to the left of the binary point so the mantissa is always less than 2 and greater than or equal to 1.

To find the Decimal equivalent of the floating point number, the mantissa is multiplied by 2 to the power of the actual exponent. The number is negated if the sign bit = 1. The following are two examples of conversion:

Example 1

Floating Point No. = 0 10000011 11000000000000000000000B

Sign — Exponent — Mantissa

Coded Exponent = 10000011B
Actual Exponent = 10000011B – 01111111B = 00000100B = 4_{10}
Mantissa = 1.11000000000000000000000B
= 1 + 1/2 + 1/4 = 1.75_{10}
Decimal No. = 2^4 x 1.75 = 16 x 1.75 = 28_{10}

Example 2

Floating Point No. = 1 01111010 01100000000000000000000B

Sign — Exponent — Mantissa

Code Exponent = 01111010B
Actual Exponent = 01111010B – 01111111B = 11111011B = -5_{10}
Mantissa = 1.01100000000000000000000B
= 1 + 1/4 + 1/8 = 1.375_{10}
Decimal No. = -2^{-5} x 1.375 = $-.04296875_{10}$

2. Unpacking of the Floating-Point Numbers
The Am9512 unpacks the floating point number into three parts before any of the arithmetic operation. The number is divided into three parts as described in Section 1. The sign and exponent are copied from the original number as 1 and 8-bit numbers respectively. The mantissa is stored as a 24-bit number. The least significant 23 bits are copied from the original number and the MSB is set to 1. The binary point is assumed to the right of the MSB.

The abbreviations listed below are used in the following sections of algorithm description:

SIGN – Sign of Result
EXP – Exponent of Result
MAN – Mantissa of Result
SIGN (TOS) – Sign of Top of Stack
EXP (TOS) – Exponent of Top of Stack
MAN (TOS) – Mantissa of Top of Stack
SIGN (NOS) – Sign of Next on Stack
EXP (NOS) – Exponent of Next on Stack
MAN (NOS) – Mantissa of Next on Stack

3. Floating-Point Add/Subtract
The floating-point add and subtract essentially use the same algorithm. The only difference is that floating-point subtract changes the sign of the floating-point number at top of stack and then performs the floating-point add.

The following is a step by step description of a floating-point add algorithm (Figure 1):

a. Unpack TOS and NOS.
b. The exponent of TOS is compared to the exponent of NOS.
c. If the exponents are equal, go to step f.
d. Right shift the mantissa of the number with the smaller exponent.
e. Increment the smaller exponent and go to step b.
f. Set sign of result to sign of larger number.
g. Set exponent of result to exponent of larger number.
h. If sign of the two numbers are not equal, go to m.
i. Add Mantissas.
j. Right shift resultant mantissa by 1 and increment exponent of result by 1.
k. If MSB of exponent changes from 1 to 0 as a result of the increment, set overflow status.
l. Round if necessary and exit.
m. Subtract smaller mantissa from larger mantissa.
n. Left shift mantissa and decrement exponent of result.
o. If MSB of exponent changes from 0 to 1 as a result of the decrement, set underflow status and exit.
p. If the MSB of the resultant mantissa = 0, go to n.
q. Round if necessary and exit.

4. Floating-Point Multiply
Floating-point multiply basically involves the addition of the exponents and multiplication of the mantissas. The following is a step by step description of a floating multiplication algorithm (Figure 2):

a. Check if TOS or NOS = 0.
b. If either TOS or NOS = 0, Set result to 0 and exit.
c. Unpack TOS and NOS.
d. Convert EXP (TOS) and EXP (NOS) to unbiased form.
EXP (TOS) = EXP (TOS) -127_{10}
EXP (NOS) = EXP (NOS) -127_{10}
e. Add exponents.
EXP = EXP (TOS) + EXP (NOS)
f. If MSB of EXP (TOS) = MSB of EXP (NOS) = 0 and MSB of EXP = 1, then set overflow status and exit.
g. If MSB of EXP (TOS) = MSB of EXP (NOS) = 1 and MSB of EXP = 0, then set underflow status and exit.
h. Convert Exponent back to biased form.
EXP = EXP + 127_{10}
i. If sign of TOS = sign of NOS, set sign of result to 0, else set sign of result to 1.
j. Multiply mantissa.
k. If MSB of resultant = 1, right shift mantissa by 1 and increment exponent of resultant.
l. If MSB of exponent changes from 1 to 0 as a result of the increment, set overflow status.
m. Round if necessary and exit.

5. Floating-Point Divide
The floating-point divide basically involves the subtraction of exponents and the division of mantissas. The following is a step by step description of a division algorithm (Figure 3).

a. If TOS = 0, set divide exception error and exit.
b. If NOS = 0, set result to 0 and exit.
c. Unpack TOS and NOS.
d. Convert EXP (TOS) and EXP (NOS) to unbiased form.
EXP (TOS) = EXP (TOS) – 127_{10}
EXP (NOS) = EXP (NOS) – 127_{10}
e. Subtract exponent of TOS from exponent of NOS.
EXP = EXP (NOS) – EXP (TOS)
f. If MSB of EXP (NOS) = 0, MSB of EXP (TOS) = 1 and MSB of EXP = 1, then set overflow status and exit.
g. If MSB of EXP (NOS) = 1, MSB of EXP (TOS) = 0, and MSB of EXP = 0, then set underflow status and exit.

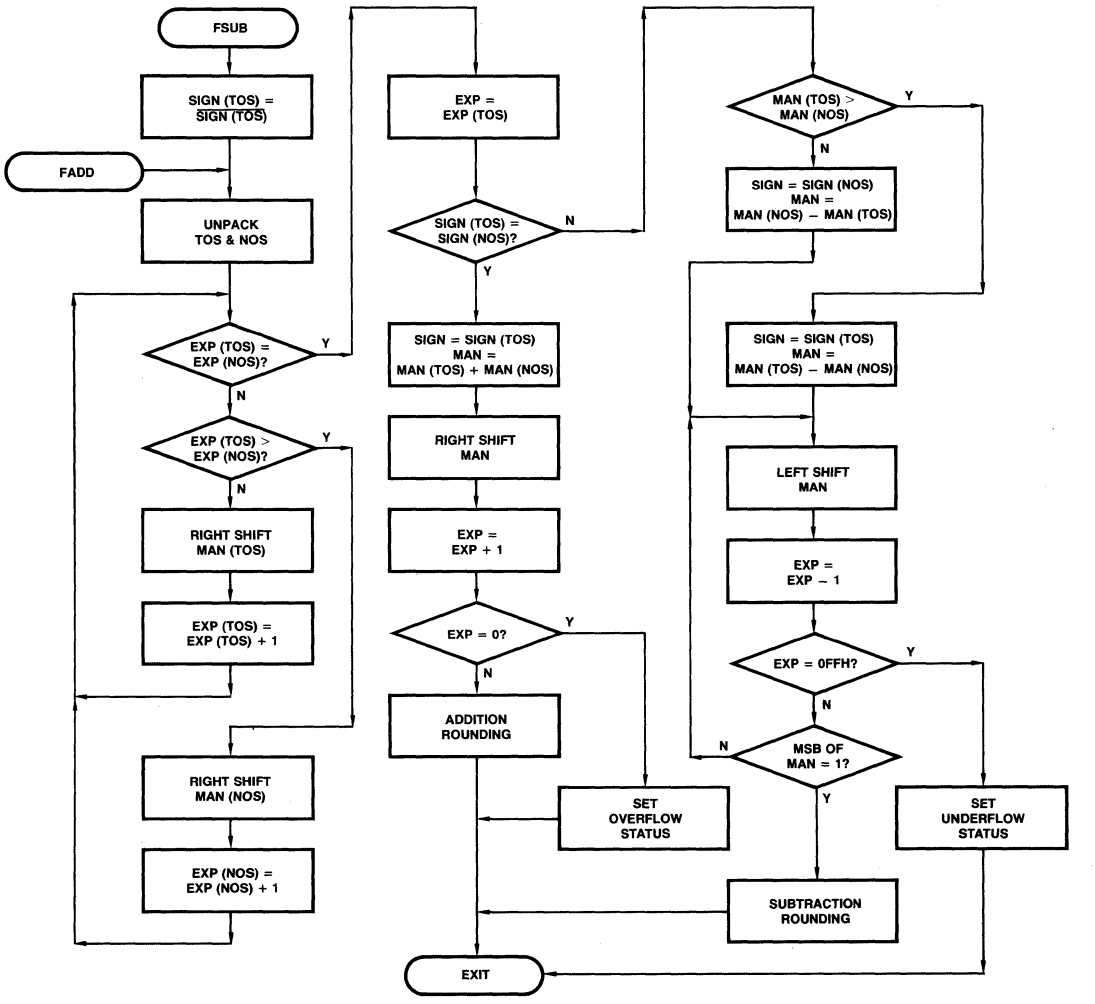


Figure 1. Conceptual Floating-Point Addition/Subtraction.

MOS-205

h. Add bias to exponent of result.
   $EXP = EXP + 127_{10}$
i. If sign of TOS = sign of NOS, set sign of result to 0, else set sign of result to 1.
j. Divide mantissa of NOS by mantissa of TOS.
k. If MSB = 0, left shift mantissa and decrement exponent of resultant, else go to n.
l. If MSB of exponent changes from 0 to 1 as a result of the decrement, set underflow status.
m. Go to k.
n. Round if necessary and exit.

The algorithms described above provide the user a means of verifying the validity of the result. They do not necessarily reflect the exact internal sequence of the Am9512.

6. Rounding

The Am9512 adopts a rounding algorithm that is consistent with the Intel® standard for floating-point arithmetic. The following description is an excerpt from the paper published in proceedings of Compsac 77, November 1977, pp. 107-112 by Dr. John F. Palmer of Intel Corporation.

The method used for doing the rounding during floating-point arithmetic is known as "Round to Even", i.e., if the resultant number is exactly halfway between two floating point numbers, the number is rounded to the nearest floating-point number whose LSB of the mantissa is 0. In order to simplify the explanation, the algorithms will be illustrated with 4-bit arithmetic. The existence of an accumulator will be assumed as shown:

OF	B1	B2	B3	B4	G	R	ST

The bit labels denote:

OF – The overflow bit
B1-B4 – The 4 mantissa bits
G – The Guard bit
R – The Rounding bit
ST – The "Sticky" bit

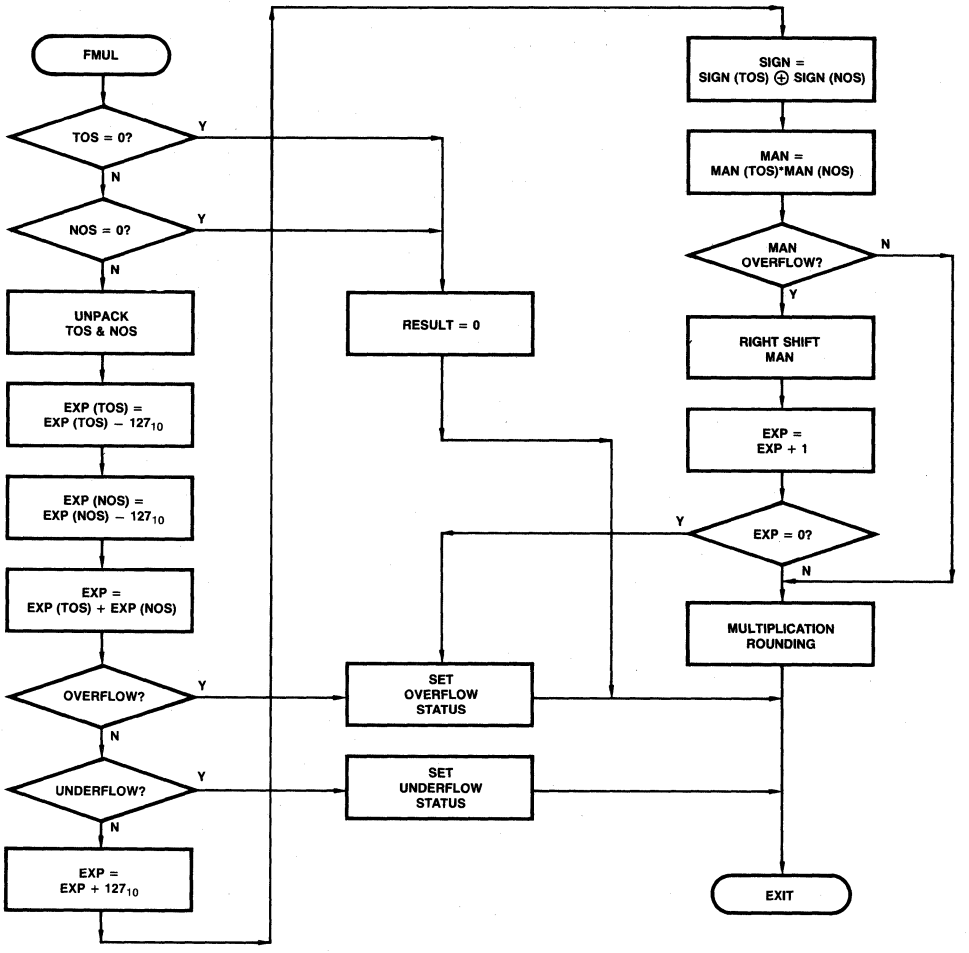


Figure 2. Conceptual Floating-Point Multiplication.

MOS-206

The Sticky bit is set to one if any ones are shifted right of the rounding bit in the process of denormalization. If the Sticky bit becomes set, it remains set throughout the operation. All shifting in the Accumulator involves the OF, G, R and ST bits. The ST bit is not affected by left shifts but, zeros are introduced into OF by right shifts.

Rounding during addition of magnitudes – add 1 to the G position, then if G=R=ST=0, set B4 to 0 ("Rounding to Even").

Rounding during subtraction of magnitudes – if more than one left shift was performed, no rounding is needed, otherwise round the same way as addition of magnitudes.

Rounding during multiplication – let the normalized double length product be:

B1	B2	B3	B4	B5	B6	B7	B8

Then G=B5, R=B6, ST=B7 V B8. The rounding is then performed as in addition of magnitudes.

Rounding during division – let the first six bits of the normalized quotient be

B1	B2	B3	B4	B5	B6

Then G=B5, R=B6, ST=0 if and only if remainder = 0. The rounding is then performed as in addition of magnitudes.

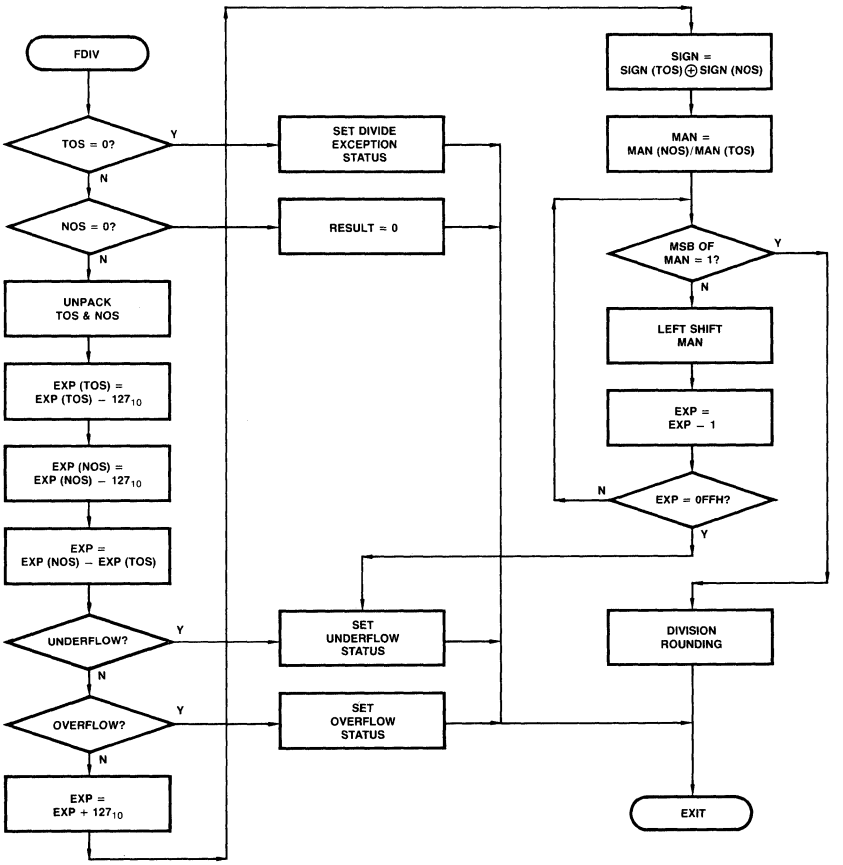


Figure 3. Conceptual Floating-Point Division.

MOS-207

CHSD

CHANGE SIGN DOUBLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	1	0	1

Hex Coding: AD IF SRE = 1
2D IF SRE = 0

Execution Time: See Table 2

Description:

The sign of the double precision TOS operand A is complemented. The double precision result R is returned to TOS. If the double precision operand A is zero, then the sign is not affected. The status bit S and Z indicate the sign of the result and if the result is zero. The status bits U, V and D are always cleared to zero.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS

BEFORE		AFTER
A	TOS	R
B	NOS	B

CHSS

CHANGE SIGN SINGLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	1	0	1

Hex Coding: 85 IF SRE = 1
05 IF SRE = 0

Execution Time: See Table 2

Description:

The sign of the single precision operand A at TOS is complemented. The single precision result R is returned to TOS. If the exponent field of A is zero, all bits of R will be zeros. The status bits S and Z indicate the sign of the result and if the result is zero. The status bits U, V and D are cleared to zero.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS

BEFORE		AFTER
A	← TOS →	R
B	← NOS →	B
C		C
D		D

CLR

CLEAR STATUS

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	0	0	0

Hex Coding: 80 IF SRE = 1
00 IF SRE = 0

Execution Time: 4 clock cycles

Description:
The status bits S, Z, D, U, V are cleared to zero. The stack is not affected. This essentially is a no operation command as far as operands are concerned.

Status Affected: S, Z, D, U, V always zero.

DADD

DOUBLE PRECISION FLOATING-POINT ADD

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	0	0	1

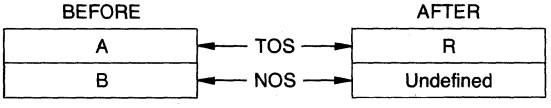
Hex Coding: A9 IF SRE = 1
29 IF SRE = 0

Execution Time: See Table 2

Description:
The double precision operand A from TOS is added to the double precision operand B from NOS. The result is rounded to obtain the final double precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



DSUB

DOUBLE PRECISION FLOATING-POINT SUBTRACT

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	0	1	0

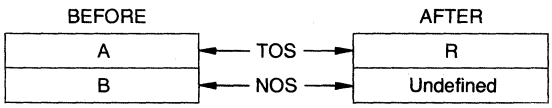
Hex Coding: AA IF SRE = 1
2A IF SRE = 0

Execution Time: See Table 2

Description:
The double precision operand A at TOS is subtracted from the double precision operand B at NOS. The result is rounded to obtain the final double precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



DMUL

DOUBLE PRECISION FLOATING-POINT MULTIPLY

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	0	1	1

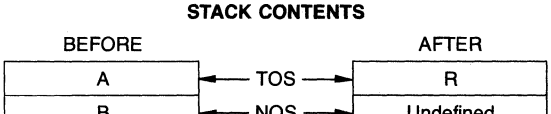
Hex Coding: AB IF SRE = 1
2B IF SRE = 0

Execution Time: See Table 2

Description:
The double precision operand A from TOS is multiplied by the double precision operand B from NOS. The result is rounded to obtain the final double precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



DDIV

DOUBLE PRECISION FLOATING-POINT DIVIDE

	7	6	5	4	3	2	1	0
Binary Code:	SRE	0	1	0	1	1	0	0

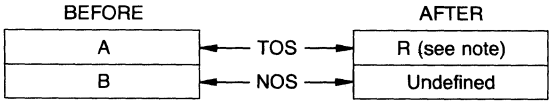
Hex Coding: AC IF SRE = 1
2C IF SRE = 0

Execution Time: See Table 2

Description:
The double precision operand B from NOS is divided by the double precision operand A from TOS. The result (quotient) is rounded to obtain the final double precision result R which is returned to TOS. The status bits, S, Z, D, U and V are affected to report sign of the result, if the result is zero, attempt to divide by zero, exponent underflow and exponent overflow respectively.

Status Affected: S, Z, D, U, V

STACK CONTENT



Note: If A is zero, then R = B (Divide exception).

SADD

SINGLE PRECISION FLOATING-POINT ADD

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	0	0	1

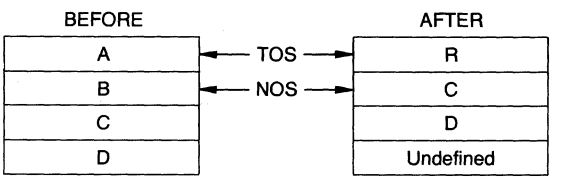
Hex Coding: 81 IF SRE = 1
01 IF SRE = 0

Execution Time: See Table 2

Description:
The single precision operand A from TOS is added to the single precision operand B from NOS. The result is rounded to obtain the final single precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report the sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENT



SSUB

SINGLE PRECISION FLOATING-POINT SUBTRACT

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	0	1	0

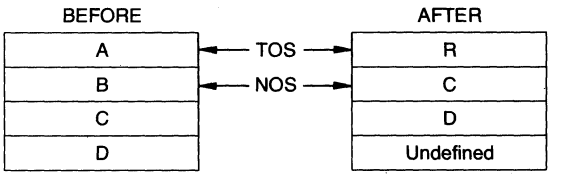
Hex Coding: 82 IF SRE = 1
02 IF SRE = 0

Execution Time: See Table 2

Description:
The single precision operand A at TOS is subtracted from the single precision operand B at NOS. The result is rounded to obtain the final single precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report the sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



SMUL

SINGLE PRECISION FLOATING-POINT MULTIPLY

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	0	1	1

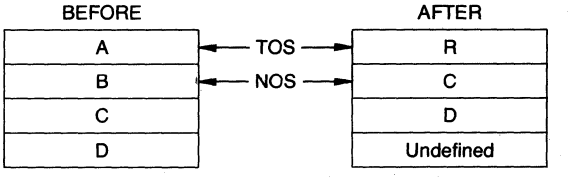
Hex Coding: 83 IF SRE = 1
03 IF SRE = 0

Execution Time: See Table 2

Description:
The single precision operand A from TOS is multiplied by the single precision operand B from NOS. The result is rounded to obtain the final single precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report the sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



SDIV

SINGLE PRECISION FLOATING-POINT DIVIDE

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	1	0	0

Hex Coding: 84 IF SRE = 1
04 IF SRE = 0

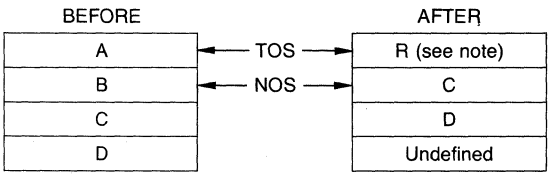
Execution Time: See Table 2

Description:

The single precision operand B from NOS is divided by the single precision operand A from TOS. The result (quotient) is rounded to obtain the final result R which is returned to TOS. The status bits S, Z, D, U and V are affected to report the sign of the result, if the result is zero, attempt to divide by zero, exponent underflow and exponent overflow respectively.

Status Affected: S, Z, D, U, V

STACK CONTENTS



Note: If exponent field of A is zero then R = B (Divide exception).

POPS

POP STACK SINGLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	1	1	1

Hex Coding: 87 IF SRE = 1
07 IF SRE = 0

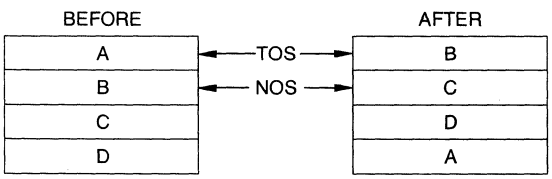
Execution Time: See Table 2

Description:

The single precision operand A is popped from the stack. The internal stack control mechanism is such that A will be written at the bottom of the stack. The status bits S and Z are affected to report the sign of the new operand at TOS and if it is zero, respectively. The status bits U, V and D will be cleared to zero. Note that only the exponent field of the new TOS is checked for zero, if it is zero status bit Z will set to 1.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS



PTOD

PUSH STACK DOUBLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	1	1	0

Hex Coding: AE IF SRE = 1
2E IF SRE = 0

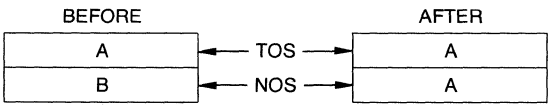
Execution Time: See Table 2

Description:

The double precision operand A from the TOS is pushed back on to the stack. This is effectively a duplication of A into two consecutive stack locations. The status S and Z are affected to report sign of the new TOS and if the new TOS is zero respectively. The status bits U, V and D will be cleared to zero.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS



PTOS

PUSH STACK SINGLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	1	1	0

Hex Coding: 86 IF SRE = 1
06 IF SRE = 0

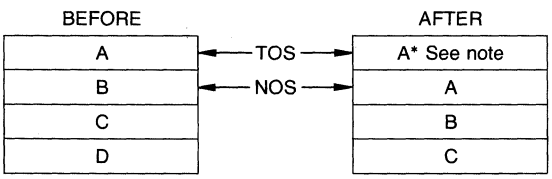
Execution Time: See Table 2

Description:

This instruction effectively pushes the single precision operand from TOS on to the stack. This amounts to duplicating the operand at two locations in the stack. However, if the operand at TOS prior to the PTOS command has only its exponent field as zero, the new content of the TOS will all be zeroes. The contents of NOS will be an exact copy of the old TOS. The status bits S and Z are affected to report the sign of the new TOS and if the content of TOS is zero, respectively. The status bits U, V and D will be cleared to zero.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS



Note: A* = A if Exponent field of A is not zero.
A* = 0 if Exponent field of A is zero.

POPD

POP STACK DOUBLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	1	1	1

Hex Coding: AF IF SRE = 1
2F IF SRE = 0

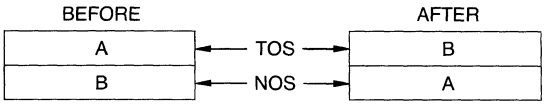
Execution Time: See Table 2

Description:

The double precision operand A is popped from the stack. The internal stack control mechanism is such that A will be written at the bottom of the stack. This operation has the same effect as exchanging TOS and NOS. The status bits S and Z are affected to report the sign of the new operand at TOS and if it is zero, respectively. The status bits U, V and D will be cleared to zero.

Status Affected: S, Z (U, V and D always zero.)

STACK CONTENTS



XCHS

EXCHANGE TOS AND NOS SINGLE-PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	1	0	0	0

Hex Coding: 88 IF SRE = 1
08 IF SRE = 0

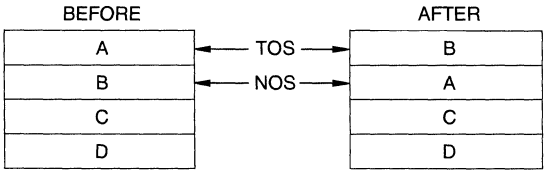
Execution Time: See Table 2

Description:

The single precision operand A at the TOS and the single precision operand B at the NOS are exchanged. After execution, B is at the TOS and A is at the NOS. All other operands are unchanged.

Status Affected: S, Z (U, V and D always zero.)

STACK CONTENTS



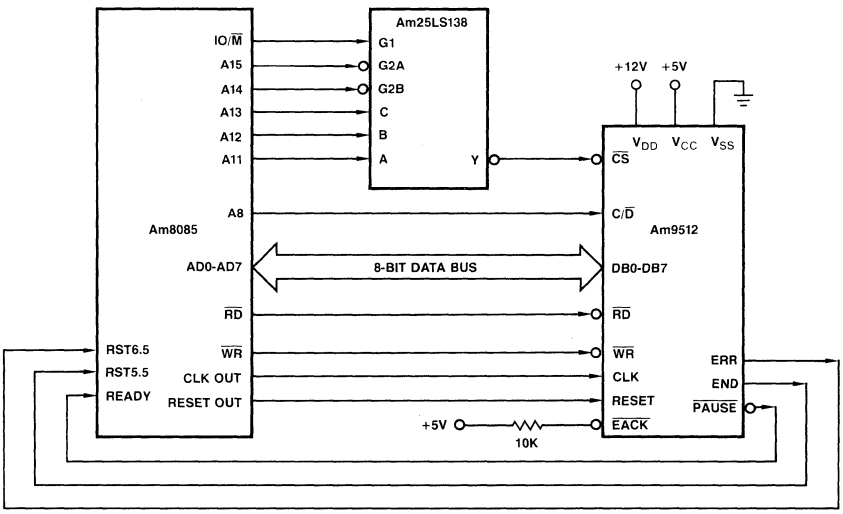


Figure 1. Am9512 to Am8085 Interface.

MOS-213

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VDD with Respect to VSS	−0.5V to +15.0V
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	2.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VSS	VCC	VDD
Am9512DC	0°C ≤ T_A ≤ +70°C	0V	+5.0V ±5%	+12V ±5%
Am9512DM	−55°C ≤ T_A ≤ +125°C	0V	+5.0V ±10%	+12V ±10%

ELECTRICAL CHARACTERISTICS Over Operating Range (Note 1)

Parameters	Description	Test Conditions	Min.	Typ.	Max.	Units
VOH	Output HIGH Voltage	IOH = −200μA	3.7			Volts
VOL	Output LOW Voltage	IOL = 3.2mA			0.4	Volts
VIH	Input HIGH Voltage		2.0		VCC	Volts
VIL	Input LOW Voltage		−0.5		0.8	Volts
IIX	Input Load Current	VSS ≤ VI ≤ VCC			±10	μA
IOZ	Data Bus Leakage	VO = 0.4V			10	μA
		VO = VCC			10	
ICC	VCC Supply Current	T_A = +25°C		50	90	mA
		T_A = 0°C			95	
		T_A = −55°C			100	
IDD	VDD Supply Current	T_A = +25°C		50	90	mA
		T_A = 0°C			95	
		T_A = −55°C			100	
CO	Output Capacitance	fc = 1.0MHz, Inputs = 0V		8	10	pF
CI	Input Capacitance			5	8	pF
CIO	I/O Capacitance			10	12	pF

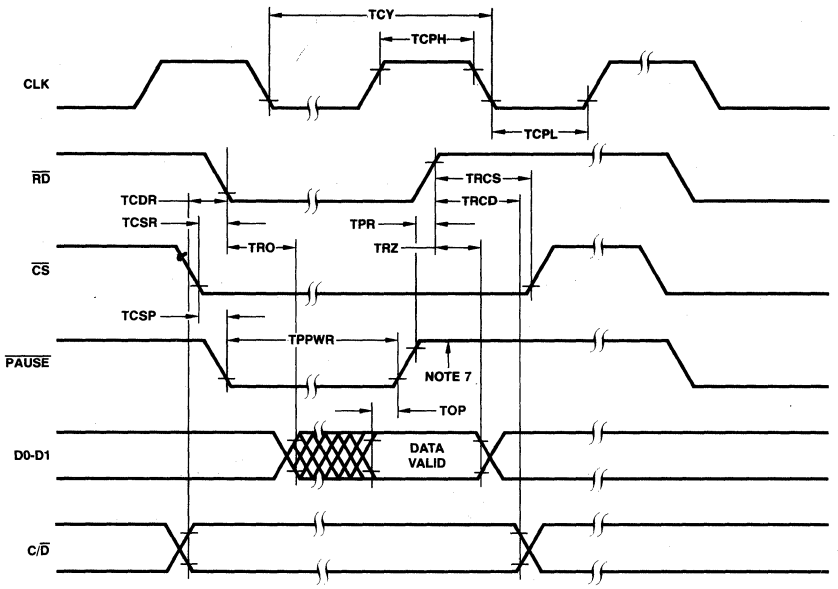
SWITCHING CHARACTERISTICS

Parameters	Description		Am9512DC Min	Am9512DC Max	Am9512-1DC Min	Am9512-1DC Max	Units
TAPW	$\overline{EACK}$ LOW Pulse Width		100		75		ns
TCDR	C/$\overline{D}$ to $\overline{RD}$ LOW Set-up Time		0		0		ns
TCDW	C/$\overline{D}$ to $\overline{WR}$ LOW Set-up Time		0		0		ns
TCPH	Clock Pulse HIGH Width		200	500	140	500	ns
TCPL	Clock Pulse LOW Width		240		160		ns
TCSP	$\overline{CS}$ LOW to $\overline{PAUSE}$ LOW Delay (Note 5)		150		100		ns
TCSR	$\overline{CS}$ to $\overline{RD}$ LOW Set-up Time		0		0		ns
TCSW	$\overline{CS}$ LOW to $\overline{WR}$ LOW Set-up Time		0		0		ns
TCY	Clock Period		480	5000	320	2000	ns
TDW	Data Valid to $\overline{WR}$ HIGH Delay			150		100	ns
TEAE	$\overline{EACK}$ LOW to END LOW Delay			200		175	ns
TEHPHR	END HIGH to $\overline{PAUSE}$ HIGH Data Read when Busy			5.5TCY+300		5.5TCY+200	ns
TEHPHW	END HIGH to $\overline{PAUSE}$ HIGH Write when Busy			200		175	ns
TEPW	END HIGH Pulse Width		400		300		ns
TEX	Execution Time		See Table 2				ns
TOP	Data Bus Output Valid to $\overline{PAUSE}$ HIGH Delay		0		0		ns
TPPWR	$\overline{PAUSE}$ LOW Pulse Width Read	Data	3.5TCY+50	5.5TCY+300	3.5TCY+50	5.5TCY+200	ns
		Status	1.5TCY+50	3.5TCY+300	1.5TCY+50	3.5TCY+200	
TPPWRB	END HIGH to $\overline{PAUSE}$ HIGH Read when Busy	Data	See Table 2				ns
		Status	1.5TCY+50	3.5TCY+300	1.5TCY+50	3.5TCY+200	
TPPWW	$\overline{PAUSE}$ LOW Pulse Width Write when Not Busy			TCSW+50		TCSW+50	ns
TPPWWB	$\overline{PAUSE}$ LOW Pulse Width Write when Busy		See Table 2				ns
TPR	$\overline{PAUSE}$ HIGH to Read HIGH Hold Time		0		0		ns
TPW	$\overline{PAUSE}$ HIGH to Write HIGH Hold Time		0		0		ns
TRCD	$\overline{RD}$ HIGH to C/$\overline{D}$ Hold Time		0		0		ns
TRCS	$\overline{RD}$ HIGH to $\overline{CS}$ HIGH Hold Time		0		0		ns
TRO	$\overline{RD}$ LOW to Data Bus On Delay		50		50		ns
TRZ	$\overline{RD}$ HIGH to Data Bus Off Delay		50	200	50	150	ns
TSAPW	SVACK LOW Pulse Width		100		75		ns
TSAR	SVACK LOW to SVREQ LOW Delay			300		200	ns
TWCD	$\overline{WR}$ HIGH to C/D Hold Time		60		30		ns
TWCS	$\overline{WR}$ HIGH to CS HIGH Hold Time		60		30		ns
TWD	$\overline{WR}$ HIGH to Data Bus Hold Time		20		20		ns

NOTES:
1. Typical values are for T_A = 25°C, nominal supply voltages and nominal processing parameters.
2. Switching parameters are listed in alphabetical order.
3. Test conditions assume transition times of 20ns or less, output loading of one TTL gate plus 100pF and timing reference levels of 0.8V and 2.0V.
4. END HIGH pulse width is specified for $\overline{EACK}$ tied to VSS. Otherwise TEAE applies.
5. $\overline{PAUSE}$ is pulled low for both command and data operations.
6. TEX is the execution time of the current command (see the Command Execution Times table).
7. $\overline{PAUSE}$ will go low at this point if $\overline{CS}$ is low and $\overline{RD}$ and $\overline{WR}$ are high.

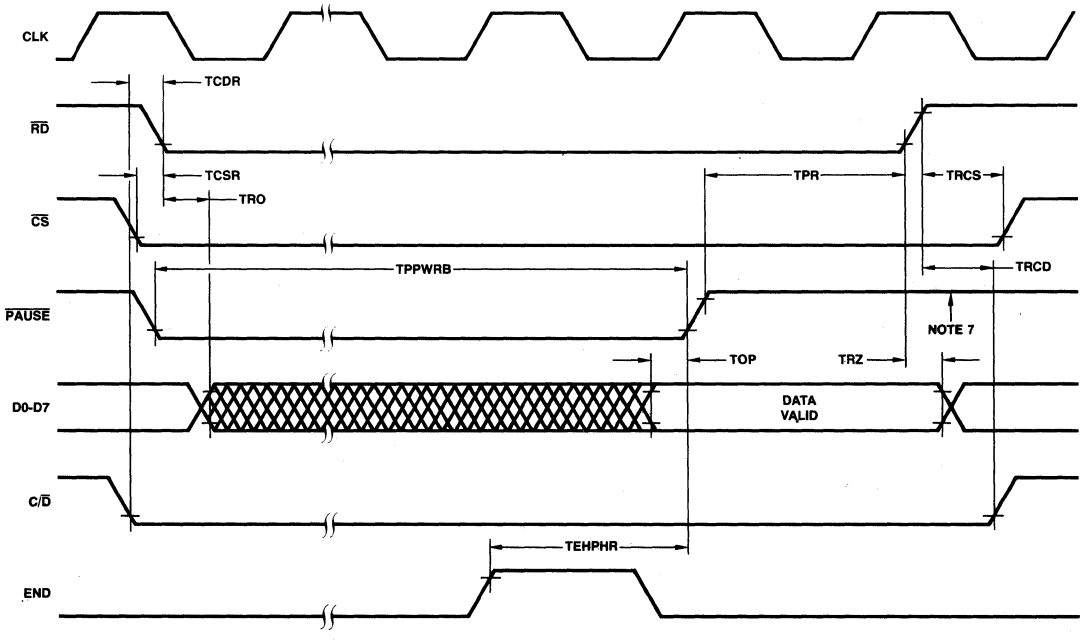
TIMING DIAGRAMS

READ OPERATION



MOS-206

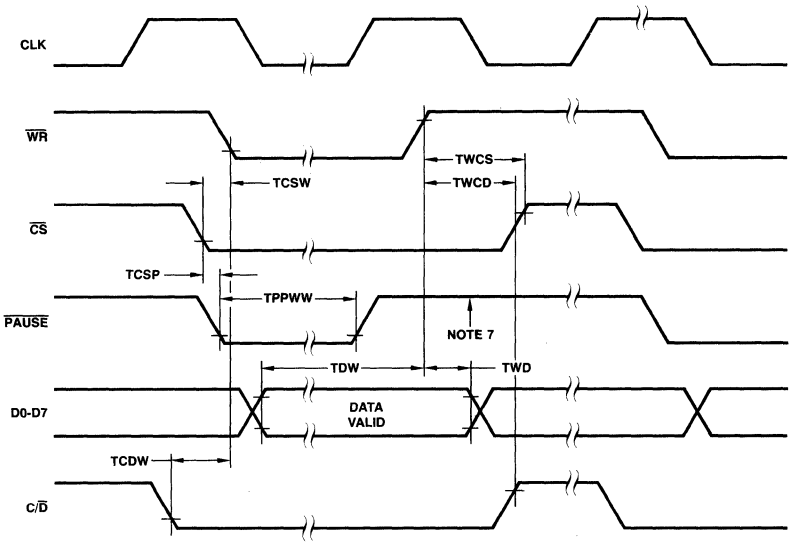
OPERAND READ WHEN Am9512 IS BUSY



MOS-209

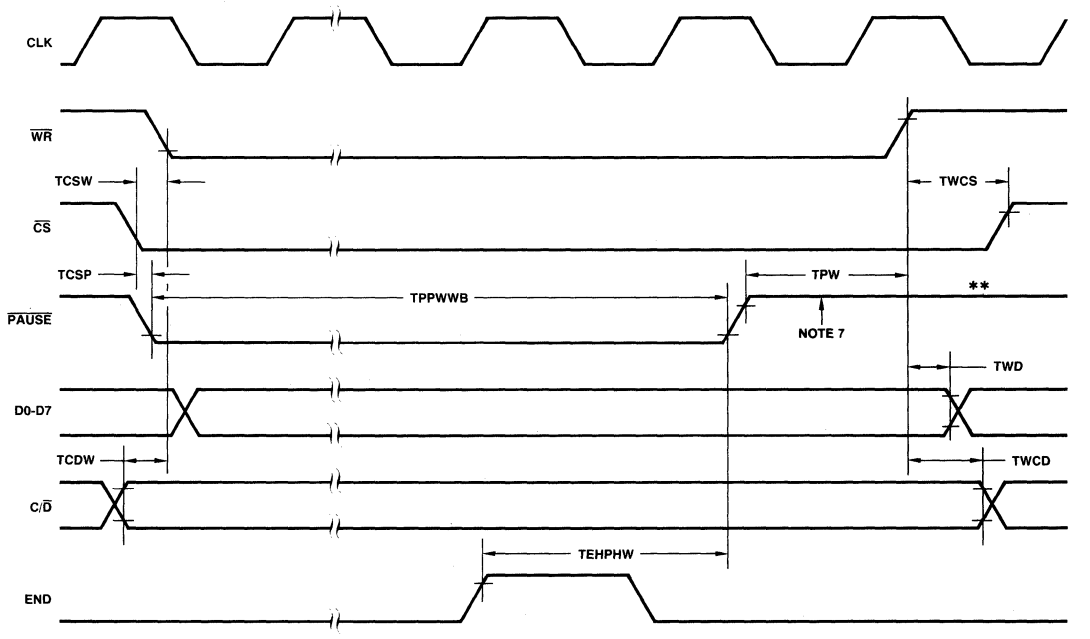
TIMING DIAGRAMS (Cont.)

OPERAND ENTRY



MOS-210

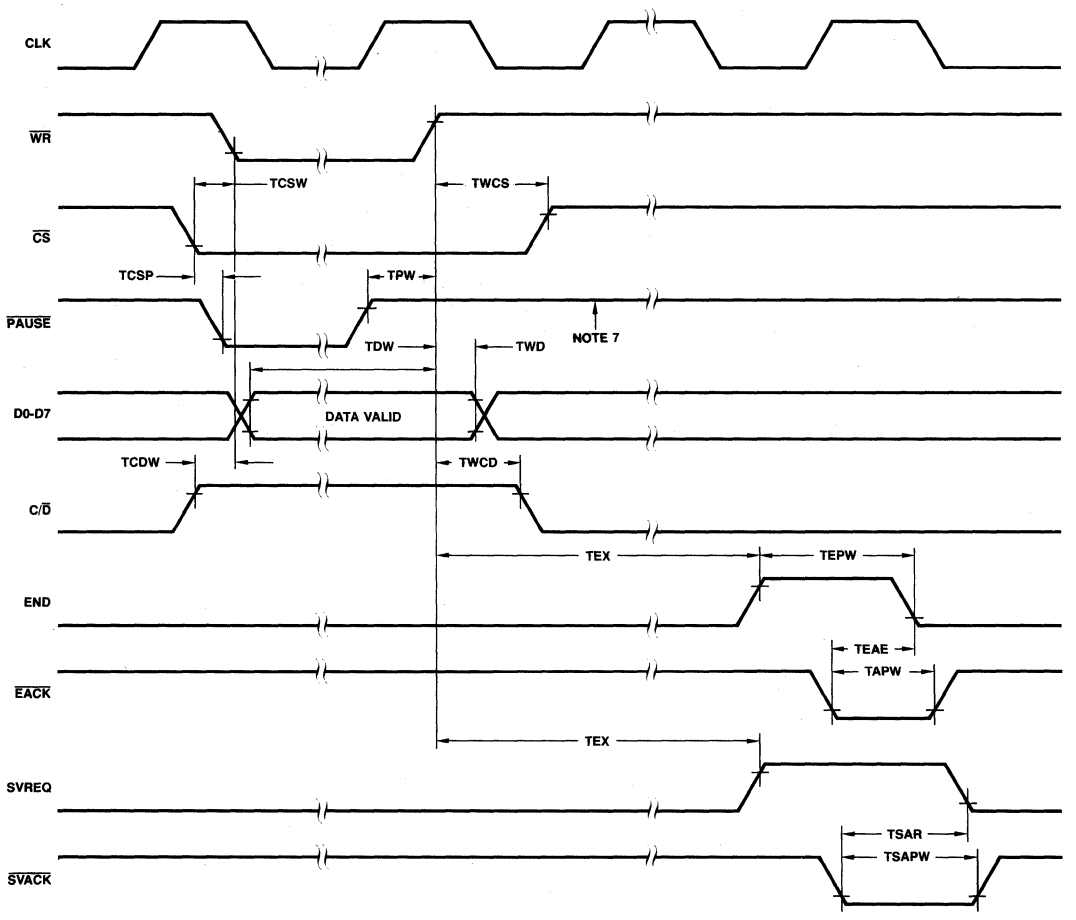
COMMAND OR DATA WRITE WHEN Am9512 IS BUSY



MOS-211

TIMING DIAGRAMS (Cont.)

COMMAND INITIATION



MOS-212

Am9513
System Timing Controller

PRELIMINARY INFORMATION

DISTINCTIVE CHARACTERISTICS

- Five independent 16-bit counters
- High speed counting rates
- Up/down and binary/BCD counting
- Internal oscillator frequency source
- Tapped frequency scaler
- Programmable frequency output
- 8-bit or 16-bit bus interface
- Time-of-day option
- Alarm comparators on counters 1 and 2
- Complex duty cycle outputs
- One-shot or continuous outputs
- Programmable count/gate source selection
- Programmable input and output polarities
- Programmable gating functions
- Retriggering capability
- +5 volt power supply
- Standard 40-pin package
- 100% MIL-STD-883 reliability assurance testing

GENERAL DESCRIPTION

The Am9513 System Timing Controller is an LSI circuit designed to service many types of counting, sequencing and timing applications. It provides the capability for programmable frequency synthesis, high resolution programmable duty cycle waveforms, retriggerable digital one-shots, time-of-day clocking, coincidence alarms, complex pulse generation, high resolution baud rate generation, frequency shift keying, stop-watching timing, event count accumulation, waveform analysis and many more. A variety of programmable operating modes and control features allow the Am9513 to be personalized for particular applications as well as dynamically reconfigured under program control.

The STC includes five general-purpose 16-bit counters. A variety of internal frequency sources and external pins may be selected as inputs for individual counters with software selectable active-high or active-low input polarity. Both hardware and software gating of each counter is available. Three-state outputs for each counter provide pulses or levels and can be active-high or active-low. The counters can be programmed to count up or down in either binary or BCD. The host processor may read an accumulated count at any time without disturbing the counting process. Any of the counters may be internally concatenated to form any effective counter length up to 80 bits.

GENERAL BLOCK DIAGRAM

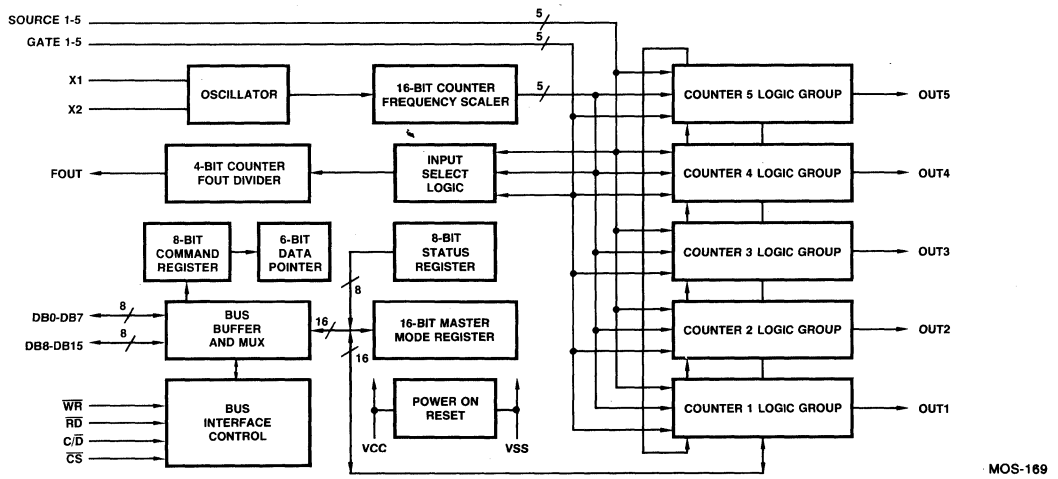


Figure 1.

ORDERING INFORMATION

Package Type	Temperature Range	Counting Frequency 7MHz	
Molded	$0°C \leq T_A \leq +70°C$	AM9513PC	
Hermetic Side-Brazed Ceramic		AM9513DC	
Hermetic Cerdip		AM9513CC	
Hermetic Side-Brazed Ceramic	$-55°C \leq T_A \leq +125°C$	AM9513DM	

FUNCTIONAL DESCRIPTION

The Am9513 block diagrams (Figures 1 and 2) indicate the interface signals and the basic flow of information. Internal control lines and the internal data bus have been omitted. The control and data registers are all connected to a common internal 16-bit bus. The external bus may be 8 or 16 bits wide; in the 8-bit mode the internal 16-bit information is multiplexed to the low order data bus pins DB0 through DB7.

An internal oscillator provides a convenient source of frequencies for use as counter inputs. Its oscillating frequency is controlled by an external reactive network such as a crystal. The oscillator output is divided by the Frequency Scaler to provide several sub-frequencies. One of the scaled frequencies (or one of ten input signals) may be selected as an input to the FOUT divider and then comes out of the chip at the FOUT interface pin.

The STC is addressed by the external system as two locations: a control port and a data port. The control port provides direct access to the Status and Command registers, as well as allowing the user to update the Data Pointer register. The data port is used to communicate with all other addressable internal locations. The Data Pointer controls the data port addressing.

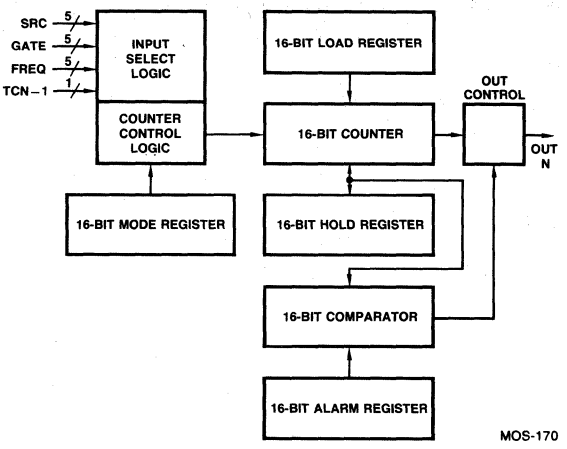
Among the registers accessible through the data port are the Master Mode register and five Counter Mode registers, one for each counter. The Master Mode register controls the programmable options that are not controlled by the Counter Mode registers.

Each of the five general purpose counters is 16 bits long and is independently controlled by its Counter Mode register. Through this register, a user can software select one of 16 sources as the counter input, a variety of gating and repetition modes, up or down counting in binary or BCD and active-high or active-low input and output polarities.

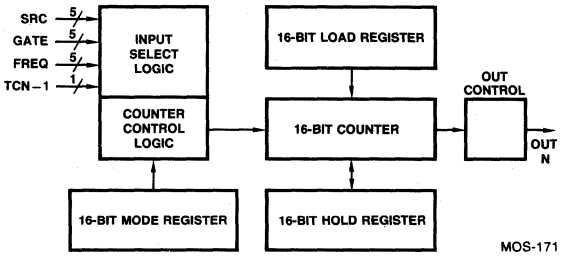
Associated with each counter is a Load register and a Hold register, both accessible through the data port. The Load register is used to automatically reload the counter to any predefined value, thus controlling its effective period. The Hold register is used to save count values without disturbing the count process, permitting the host processor to read intermediate counts. In addition, the Hold register may be used as a second Load register to generate a number of complex output waveforms.

All five counters have the same basic control logic and control registers. Counters 1 and 2 have additional alarm registers and comparators associated with them, plus the extra logic necessary for operating in a 24-hour time-of-day mode. For real-time operation the time-of-day logic will accept 50Hz, 60Hz or 100Hz input frequencies.

Each general counter has a single dedicated output pin. It may be turned off when the output is not of interest or may be configured in a variety of ways to drive interrupt controllers, Darlington buffers, bus drivers, etc. The counter inputs, on the other hand, are specifically not dedicated to any given interface line. Considerable versatility is available for configuring both the input and the gating of individual counters. This not only permits dynamic reassignment of inputs under software control, but also allows multiple counters to use a single input, and allows a single gate pin to control more than one counter.



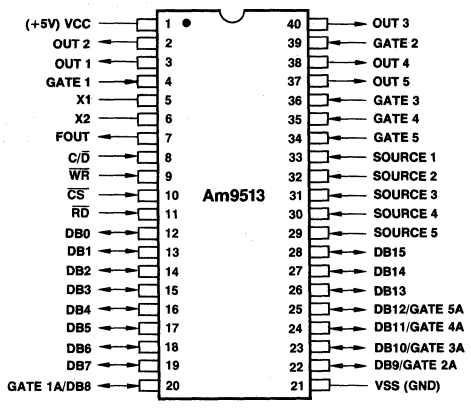
Counter Logic Groups 1 and 2.



Counter Logic Groups 3, 4, and 5.

Figure 2.

CONNECTION DIAGRAM



Top View

Pin 1 is marked for orientation. MOS-172

Figure 3.

INTERFACE SIGNAL DESCRIPTION

VCC: +5 volt power supply

VSS: Ground

X1,X2 (Crystal, Inputs)

X1 and X2 are the connections for an external crystal that determines the frequency of the internal oscillator. An RC or LC network may also be used instead of a crystal. For driving from an external frequency source, X1 should be left open and X2 should be driven with a TTL-level square wave.

FOUT (Frequency Out, Output)

The FOUT output is derived from a 4-bit counter that may be programmed to divide its input by any integer value from 1 to 16, inclusive. The input to the counter is selected from any of 15 sources, including the scaled internal frequencies. FOUT may be gated on and off under software control. Following power-up or reset, FOUT provides a frequency that is 1/16 that of the internal oscillator.

GATE1-GATE5 (Gate, Inputs)

The Gate inputs provide hardware control of the counting operations of individual counters by determining when counting may proceed. The same input may control up to three counters. Gates may also be selected as count sources for any of the counters or for the FOUT divider. The active polarity for a selected Gate input is programmable at each counter. Schmitt-trigger circuitry on the GATE inputs allows slow transition times to be used.

SRC1-SRC5 (Source, Inputs)

The Source inputs provide external signals that may be counted by any of the counters. Any Source line may be routed to any or all of the counters and the FOUT divider. The active polarity for a selected Source input is programmed at each counter. Any source waveform duty cycle will be accepted as long as the minimum pulse width is at least half the period of the maximum specified counting frequency for the part. Schmitt-trigger circuitry on the SRC inputs allows slow transition times to be used.

OUT1-OUT5 (Counter Outputs)

Each of the five counters has a dedicated output pin. Depending on the output configuration, the OUT signal may be a pulse, a square wave, or a complex duty cycle waveform. For counters 1 and 2, the OUT signal may also indicate the status of comparator circuits. Output polarities may be individually programmed.

DB0-DB7, DB8-DB15 (Data Bus, Input/Output)
GATE1A-GATE5A (Auxiliary Gates, Input)

The Data Bus lines are used to communicate with the external system. After power-up or reset, the data bus will be configured for 8-bit width. It may be reconfigured for 16-bit width by changing a control bit in the Master Mode register. Figure 4 summarizes all data bus transfers.

When operating in the 8-bit data bus environment, DB8 through DB12 may optionally be used as additional Gate inputs. See Figure 3. If unused they should be held high. When pulled low, a GateNA signal will disable the action of the gate input controlling counter N. DB13, DB14 and DB15 should be tied high for an 8-bit data bus width.

$\overline{CS}$ (Chip Select, Input)

The active-low Chip Select input enables Read and Write operations on the data bus. See Figure 4.

$\overline{RD}$ (Read, Input)

The active-low Read signal is conditioned by Chip Select and indicates that internal information is to be transferred to the data bus. WR and RD should be mutually exclusive.

$\overline{WR}$ (Write, Input)

The active-low Write signal is conditioned by Chip Select and indicates that data bus information is to be transferred to an internal location. WR and RD should be mutually exclusive.

C/$\overline{D}$ (Control/Data, Input)

The Control/Data signal selects source and destination locations for read and write operations on the data bus. Control Write operations load the Command register and the Data Pointer. Control Read operations output the Status register. Data Read and Data Write transfers communicate with all other internal registers.

Signal Configuration				Data Bus Operation
$\overline{CS}$	C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	
0	0	0	1	Transfer contents of register addressed by Data Pointer to the data bus.
0	0	1	0	Transfer contents of data bus to data register addressed by Data Pointer.
0	1	0	1	Transfer contents of Status register to data bus.
0	1	1	0	Transfer contents of data bus into Command register.
X	X	1	1	No transfer.
1	X	X	X	No transfer.
X	X	0	0	Illegal Condition.

Figure 4. Data Bus Transfers.

CONTROL PORT REGISTERS

Command Register

The 8-bit write-only Command register is loaded by writing into the control port as shown in Figure 4. With a 16-bit data bus, the low-order 8 bits are loaded into the register; the high-order byte should be FF (hex).

The Command register provides direct control over each of the five general counters and controls access through the data port by allowing the user to update the Data Pointer register. A summary of all commands appears in Figure 5. Six of the command types are used for direct software control of the counting process. Each of these six commands contains a five-bit S field. In a linear-select fashion, each bit in the S field corresponds to one of the five general counters (S1 = Counter 1, S2 = Counter 2, etc.). When an S bit is a one, the specified operation is performed on the counter so designated; when an S bit is a zero, no operation occurs for the corresponding counter.

A counter must be armed by one of the ARM commands before counting can commence. Once armed, the counting process may be further enabled or disabled using the hardware gating facilities. The ARM and DISARM commands permit software gating of the count process, in some modes.

The LOAD command causes the counter to be reloaded with the value in either the associated Load register or the associated Hold register. It will often be used as a software retrigger, or as counter initialization prior to active hardware gating.

The DISARM command disables further counting independent of any hardware gating. A disarmed counter may be reloaded using the LOAD command, may be incremented or decremented using the STEP command and may be read using the SAVE command. A count process may be resumed using an ARM command.

The SAVE command transfers the contents of a counter to its associated Hold register. The transfer takes place without interfering with any counting that may be under way. This command will overwrite any previous Hold register contents. The SAVE command is designed to allow an accumulated count to be preserved so that it can be read by the host CPU at some later time.

Two combinations of the basic commands exist to either LOAD AND ARM or to DISARM AND SAVE any combination of counters. Additional commands are provided to: step an individual counter by one count; set and clear an output toggle; issue a software reset; clear and set special bits in the Master Mode register; load the Data Pointer register.

Data Pointer Register

The 6-bit Data Pointer register is loaded by issuing the appropriate command through the control port to the Command register. As shown in Figure 6, the Data Pointer register consists of a Byte Pointer, an Element Pointer, and a Group Pointer. The content of the Data Pointer is used as an address to point to an internal register. When a register is addressed by the Data Pointer, it may be accessed through the data port.

The Byte Pointer bit in the Data Pointer register indicates which byte of a 16-bit register is to be transferred on the next access through the data port. Whenever the Data Pointer is loaded, the Byte Pointer bit is set to one, indicating a least-significant byte is expected. The Byte Pointer toggles following each 8-bit data transfer with an 8-bit data bus (MM13 = 0), or it always remains set with the 16-bit data bus option (MM13 = 1). Although the contents of the Element and Group Pointer in the Data Pointer register cannot be read by the host processor, the Byte Pointer is available as a bit in the Status register.

To permit the host processor to rapidly access the various internal registers, automatic sequencing of the Data Pointer is provided. Sequencing is enabled by clearing Master Mode bit 14 (MM14) to 0. As shown in Figure 7, several types of sequencing are available depending on the data bus width being used and the initial Data Pointer value entered by command.

When E1 = 0 or E2 = 0 and G4, G2, G1 point to a Counter Group, the Data Pointer will proceed through the Element cycle. The Element field will automatically sequence through three values: 00, 01 and 10 starting with the value entered. When the transition from 10 to 00 occurs, the Group field will also be incremented by one. Note that the Element field in this case does not sequence to a value of 11. The Group field circulates only within the five Counter Group codes.

Command Code								Command Description
C7	C6	C5	C4	C3	C2	C1	C0	
0	0	0	E2	E1	G4	G2	G1	Load Data Pointer register with contents of E and G fields. (G ≠ 000, G ≠ 110)
0	0	1	S5	S4	S3	S2	S1	Arm counting for all selected counters
0	1	0	S5	S4	S3	S2	S1	Load contents of specified source into all selected counters
0	1	1	S5	S4	S3	S2	S1	Load and Arm all selected counters
1	0	0	S5	S4	S3	S2	S1	Disarm and Save all selected counters
1	0	1	S5	S4	S3	S2	S1	Save all selected counters in hold register
1	1	0	S5	S4	S3	S2	S1	Disarm all selected counters
1	1	1	0	1	N4	N2	N1	Set output bit N (001 ≤ N ≤ 101)
1	1	1	0	0	N4	N2	N1	Clear output bit N (001 ≤ N ≤ 101)
1	1	1	1	0	N4	N2	N1	Step counter N (001 ≤ N ≤ 101)
1	1	1	0	1	0	0	0	Set MM14 (Disable Data Pointer Sequencing)
1	1	1	0	1	1	1	0	Set MM12 (Gate off FOUT)
1	1	1	0	1	1	1	1	Set MM13 (Enter 16-bit bus mode)
1	1	1	0	0	0	0	0	Clear MM14 (Enable Data Pointer Sequencing)
1	1	1	0	0	1	1	0	Clear MM12 (Gate on FOUT)
1	1	1	0	0	1	1	1	Clear MM13 (Enter 8-bit bus mode)
1	1	1	1	1	1	1	1	Master reset

Figure 5. Am9513 Command Summary.

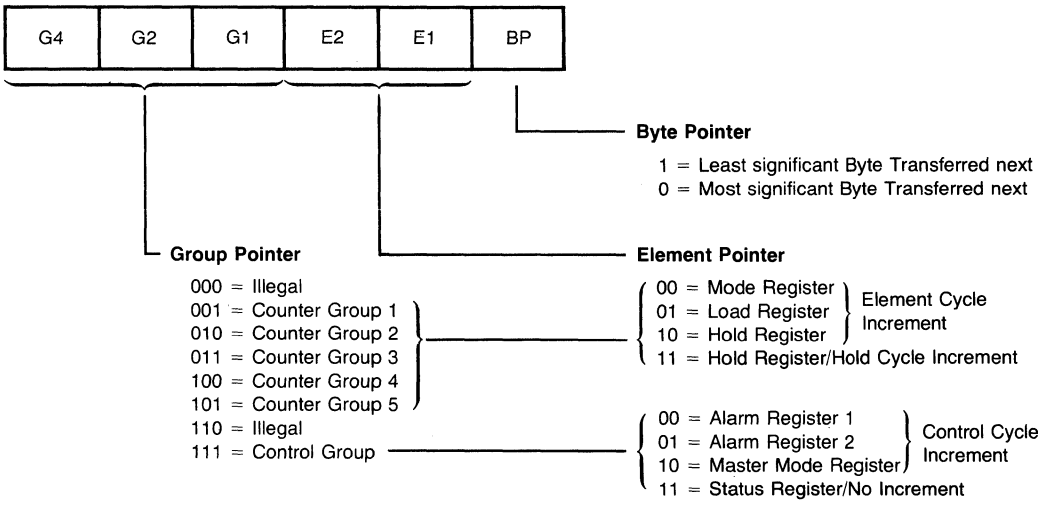


Figure 6. Data Pointer Counter.

MOS-173

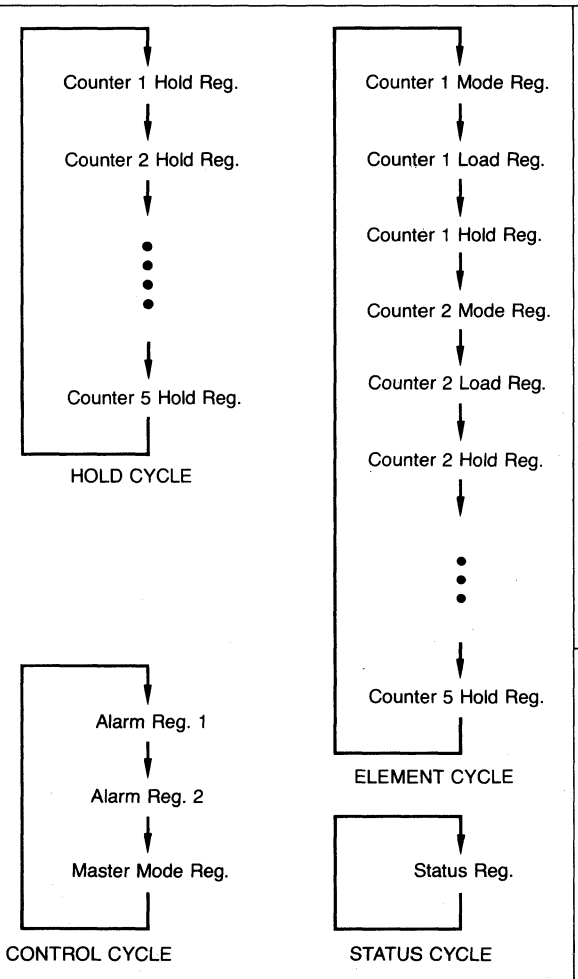


Figure 7. Data Pointer Sequencing.

MOS-174

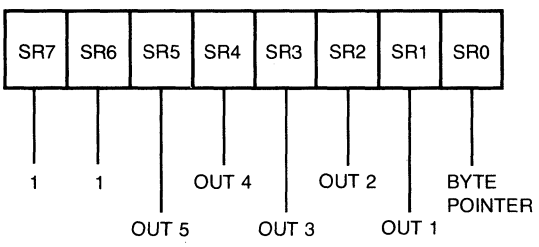
If E1 = 1 and E2 = 1, then only the Group field is sequenced. This is the Hold cycle. It allows the Hold registers to be sequentially accessed while bypassing the Mode and Load registers. The third type of sequencing is the Control cycle. If G4,G2,G1 = 111 and E2,E1 ≠ 11, the Element Pointer will be incremented through the values 00, 01 and 10, with no change to the Group Pointer.

When G4,G2,G1 = 111 and E2,E1 = 11, no incrementing takes place and only the Status register will be available through the data port. Note that the Status register can also always be read directly through the Control port.

For all of these auto-sequence modes, if an 8-bit data bus is used, the Byte pointer will toggle after every data transfer to allow the least and most significant bytes to be transferred before the Element or Group Fields are incremented.

Status Register

The 8-bit read-only Status register indicates the state of the Byte Pointer bit in the Data Pointer register and the state of the OUT signal for each of the general counters. See Figure 8. The OUT signals reported are those internal to the chip after the polarity-select logic and just before the three-state interface buffer circuitry. The Status register is normally accessed by reading the control port (see Figure 4) but may also be read via the data port as part of the Control Group.



MOS-175

Figure 8. Status Register Bit Assignments.

DATA PORT REGISTERS

Counter Logic Groups

As shown in Figure 2, each of the five Counter Logic Groups consists of a 16-bit general counter with associated control and output logic, a 16-bit Load register, a 16-bit Hold register and a 16-bit Mode register. In addition, Counter Groups 1 and 2 also include 16-bit Comparators and 16-bit Alarm registers. The comparator/alarm functions are controlled by the Master Mode register. The operation of the Counter Mode registers, then, is the same for all five counters. The host CPU has both read and write access to all registers in the Counter Logic Groups through the data port. The counter itself is never directly accessed.

The 16-bit read/write Load register is used to control the effective period of the general counter. Any 16-bit value may be written into the Load register. That value can then be transferred into the counter each time that Terminal Count (TC) occurs. "Terminal Count" is defined as that period of time when the counter contents would have been zero if an external value had not been transferred into the counter. Thus the terminal count frequency can be the input frequency divided by the value in the Load register. In all operating modes the contents of either Load or Hold will be transferred into the counter when TC occurs. In cases where values are being accumulated in the counter, the Load register action can be transparent by filling the Load register with all zeros.

The 16-bit read/write Hold register is dual-purpose. It can be used in the same way as the Load register, thus offering an alternate source for modulo definition for the counter. The Hold register may also be used to store accumulated counter values for later transfer to the host processor. This allows the count to be sampled while the counting process proceeds. Transfer of the counter contents into the Hold register is accomplished by the hardware interface in some operating modes or by the software SAVE command at any time.

The 16-bit read/write Counter Mode register controls the gating, counting, output and source select functions within each Counter Logic Group. Figure 9 shows the bit assignments for the Counter Mode registers. Generally each counter is independently configured by its Counter Mode register and does not depend on configuration information outside its Counter Logic Group.

Counter mode bits CM0 through CM2 specify the output control configuration. The OUT pin may be off and in a high impedance state, or it may be off with a low impedance to ground. The six remaining combinations are split into active-high and active-low versions of the three basic output waveforms.

One output form available is called Terminal Count (TC) and represents the period in time that the counter reaches an equivalent value of zero. Figure 10 shows a Terminal Count pulse and an example context that generated it. The TC width is determined by the period of the counting source. Regardless of any gating input, the terminal count will go active for only one clock cycle. Figure 10 assumes active-high source polarity, counter armed, counter decrementing and an external reload value of K.

The counter will always be loaded from an external location when TC occurs; the user can choose the source location and the value. If a non-zero value is picked, the counter will never really attain a zero state and TC will indicate the counter state that would have been zero had no parallel transfer occurred.

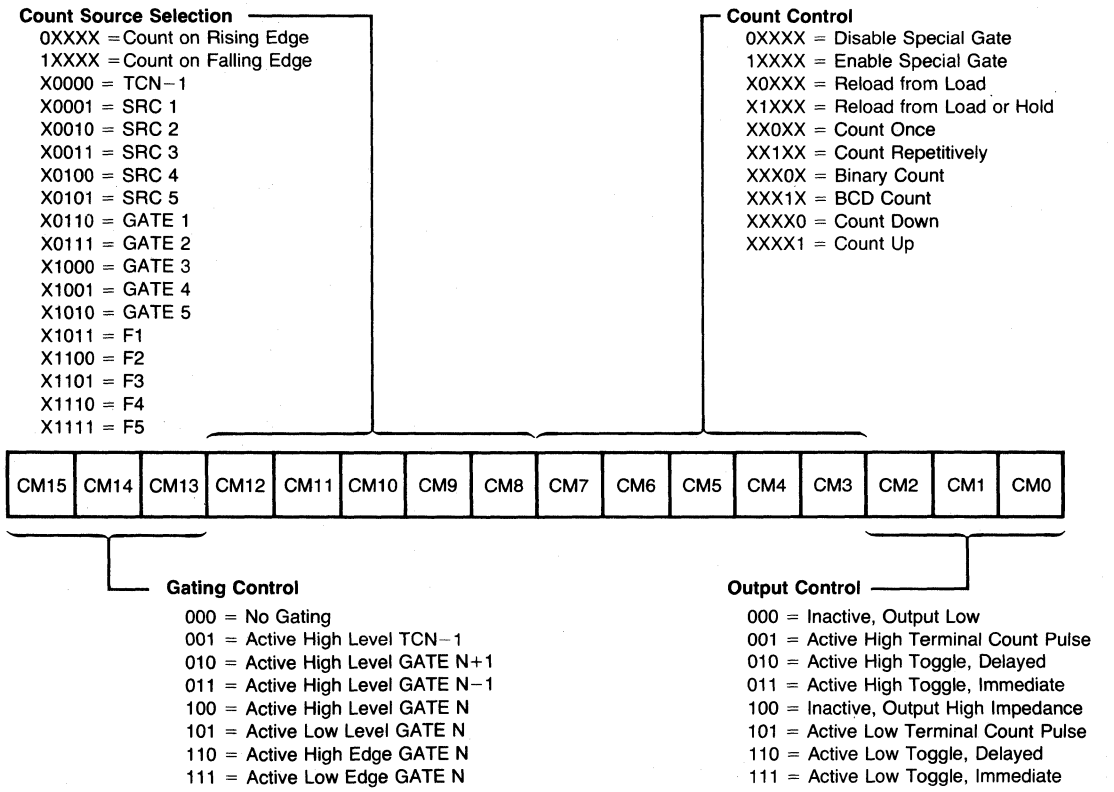


Figure 9. Counter Mode Register Bit Assignments.

MOS-176

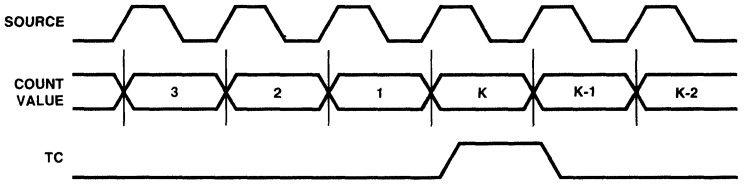


Figure 10. Terminal Count Waveform.. MOS-177

Another output form uses TC to toggle a flip-flop to generate an output level instead of a pulse. Two variations of the toggle waveforms are available, as shown in Figure 11. The one labeled "Delayed" uses only the TC pulse to change the toggle. Since TC does not occur until a full count elapses following the loading of the counter, the first transition of the toggle is delayed from the moment of arming. On the other hand, the waveform labeled "Immediate" also uses the TC pulse as the toggling source but adds a toggle transition on the first count following the arming. After the initial transition, both Delayed and Immediate waveforms are the same; for the same output polarity they will be 180° out of phase. The trailing edge of TC triggers the toggle and the toggle output is 1/2 the frequency of TC.

Counter Mode bits CM8 through CM12 specify the source used as input to the counter and the active edge that is counted. Bit CM12 controls the polarity for all the sources; logic zero counts rising edges and logic one counts falling edges. Bits CM8 through CM11 select one of sixteen counting sources to route to the counter input. Five of the available inputs are internal frequencies derived from the internal oscillator (see Figure 15 for frequency assignments). Ten of the available inputs are interface pins; five are labeled SRC and five are labeled GATE. The sixteenth available input is the TC signal from the adjacent lower-numbered counter (The Counter 5 TC wraps around to the Counter 1 input). This option allows internal concatenation that permits very long counts to be accumulated. When TCN−1 is the source, the count ripples between the connected counters.

Counter Mode bits CM3 through CM7 specify the various options available for direct control of the counting process. CM3 and CM4 operate independently of the others and control up/down and BCD/binary counting. They may be combined freely with other control bits to form many types of counting configurations. The other three bits interact in complex ways. Bit CM5 controls the repetition of the count process. When CM5 = 1, counting will proceed in the specified mode until the counter is disarmed or the mode is changed. When CM5 = 0 the count process will proceed only until one full cycle of operation occurs. This may occur after one or two TC events. The counter is then disarmed automatically. The single or double TC requirement will depend on the state of other control bits. Note that even if the counter is automatically disarmed upon a TC, it always counts the count source edge which generates the trailing TC edge.

Bit CM6 specifies the location used to reload the counter contents when TC occurs. When CM6 = 0, the contents of the Load register are transferred into the counter at every TC. When CM6 = 1, the reload location may be either the Load or Hold register. The reload location in this case may be controlled externally using a GATE pin or may alternate on each TC. Bit CM7 controls the special gating functions that allow retriggering and the selection of Load and Hold locations for counter updating. The use and definition of CM7 will depend on the status of the Gating Control field and bits CM5 and CM6. See Figure 12.

Counter Mode bits CM13 through CM15 specify the hardware gating options. When "no gating" is selected (000) the counter will proceed unconditionally as long as it is armed. For any other gating mode, the count process is conditioned by the specified gating configuration. For codes of 110 or 111 in this field, counting proceeds after the specified active Gate edge occurs. Thereafter, the Gate input is ignored and counting continues until after one or two TC pulses occur or the counter is disarmed. Other codes in the gating field select either active-high or active-low level gating from a particular GATE pin, or from the TC signal of the adjacent counter. Level gating allows the counter to count only those clock edges that occur while the gate is active. Figure 12 summarizes the various counting configurations of the Am9513.

When edge gating is specified and the CM7 bit is cleared, counting will be enabled on the first active gate edge after the ARM instruction. Counting will continue until a DISARM instruction occurs or one or two TC pulses occur. When the counting stops on TC, an active GATE edge will allow counting to resume if the specified repetition has not occurred. While the counter is counting no GATE input edge or level will influence the count sequence. This mode provides a non-retriggerable, edge-triggered, digital one-shot function. When edge gating is specified and the CM7 bit is set, counting will begin on the active gate edge after the ARM instruction. If the specified repetition has not occurred, any active edge of the specified Gate input will reload the counter and at the same time the counter's contents will be transferred to the Hold register. In this mode the counter will stop after each TC and will resume on the next active GATE edge. Thus the counter performs as a retriggerable edge-triggered one-shot. In the non-gated mode when the CM7 bit is

10

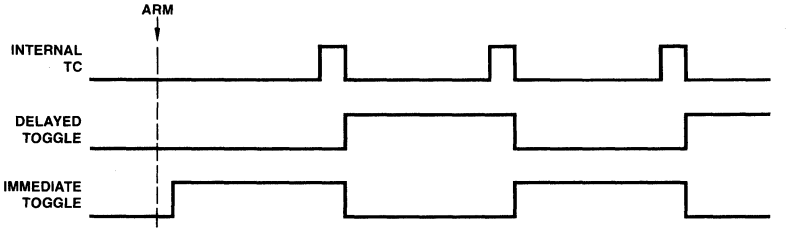


Figure 11. Output Waveforms. MOS-178

Operating Mode	A	B	C	D	E	F	G	H	I	J	K	L
Special Gate (CM7)	0	0	0	0	0	0	0	0	0	0	0	0
Reload Source (CM6)	0	0	0	0	0	0	1	1	1	1	1	1
Repetition (CM5)	0	0	0	1	1	1	0	0	0	1	1	1
Gate Control (CM15-CM13)	000	LEVEL	EDGE	000	LEVEL	EDGE	000	LEVEL	EDGE	000	LEVEL	EDGE
Count to TC once, then disarm	X	X	X									
Count to TC twice, then disarm							X	X	X			
Count to TC repeatedly				X	X	X				X	X	X
Gate input does not gate counter input	X			X			X			X		
Count only during active gate level		X			X			X			X	
Start count on active gate edge and stop count on next TC.			X			X						
Start count on active gate edge and stop count on second TC.									X			X
No hardware retriggering	X	X	X	X	X	X	X	X	X	X	X	X
Reload counter from Load Register on TC	X	X	X	X	X	X						
Reload counter on each TC, alternating reload source between Load and Hold Registers.							X	X	X	X	X	X
Transfer Load Register into counter on each TC that gate is LOW; transfer Hold Register into counter on each TC that gate is HIGH.												
On active gate edge transfer counter into Hold Register and then reload counter from Load Register.												
On active gate edge transfer counter into Hold Register and then reload counter from Load or Hold Register.												

Operating Mode	M	N	O	P	Q	R	S	T	U	V	W	X
Special Gate (CM7)	1	1	1	1	1	1	1	1	1	1	1	1
Reload Source (CM6)	0	0	0	0	0	0	1	1	1	1	1	1
Repetition (CM5)	0	0	0	1	1	1	0	0	0	1	1	1
Gate Control (CM15-CM13)	000	LEVEL	EDGE	000	LEVEL	EDGE	000	LEVEL	EDGE	000	LEVEL	EDGE
Count to TC once, then disarm		X	X									
Count to TC twice, then disarm							X	X	X			
Count to TC repeatedly					X	X				X	X	X
Gate input does not gate counter input							X			X		
Count only during active gate level		X			X			X			X	
Start count on active gate edge and stop count on next TC.			X			X						
Start count on active gate edge and stop count on second TC.									X			X
No hardware retriggering							X			X		
Reload counter from Load Register on TC		X	X		X	X						
Reload counter on each TC, alternating reload source between Load and Hold Registers.								X	X		X	X
Transfer Load Register into counter on each TC that gate is LOW; transfer Hold Register into counter on each TC that gate is HIGH.							X			X		
On active gate edge transfer counter into Hold Register and then reload counter from Load Register.		X	X		X	X						
On active gate edge transfer counter into Hold Register and then reload counter from Load or Hold Register.								X	X		X	X

Note: Operating modes M and P should not be used.

Figure 12. Am9513 Operating Modes.

set, the counter is enabled for frequency shift keying operation. In this mode, the counter's gate input will control whether the Load register or Hold register is reloaded into the counter at Terminal Count. The GATE input is synchronized with the count source to eliminate possible race conditions if it changes as the TC occurs.

When the Am9513 is set to operate with an 8-bit data bus width, pins DB8 through DB15 are not used for the data bus and are available for other functions. Pins DB13 through DB15 should be tied high. Pins DB8 through DB12 are used as auxiliary gating inputs, and are labeled GATE1A through GATE5A respectively. The auxiliary gate pin, GATENA, is logically ANDed with the gate input to Counter N, as shown in Figure 13. The output of the AND gate is then used as the gating signal for Counter N.

Master Mode Register

The 16-bit Master Mode (MM) register is used to control those internal activities that are not controlled by the individual Counter Mode registers. This includes frequency control, time-of-day operation, comparator controls, data bus width and data pointer sequencing. Figure 14 shows the bit assignments for the Master Mode register.

A 16-bit scaling counter divides the output of the on-chip oscillator into four additional sub-frequencies. This provides a total of five internal frequencies that may be routed to any of the general counters and to the FOUT divider. The scaler is tapped every 4 bits and may be programmed by Master Mode bit MM15 to divide in binary or in BCD. Figure 15 shows the resulting combinations of frequencies that are available.

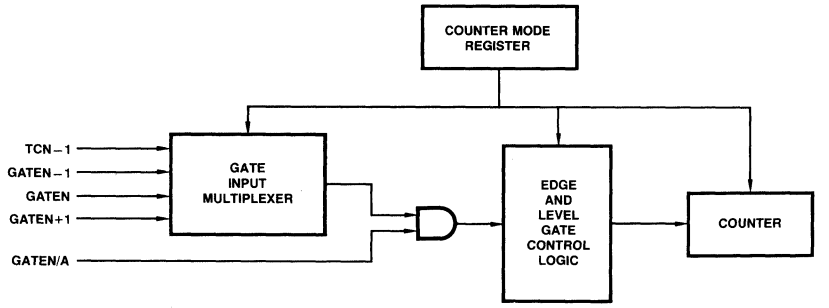


Figure 13. Gating Control.

MOS-179

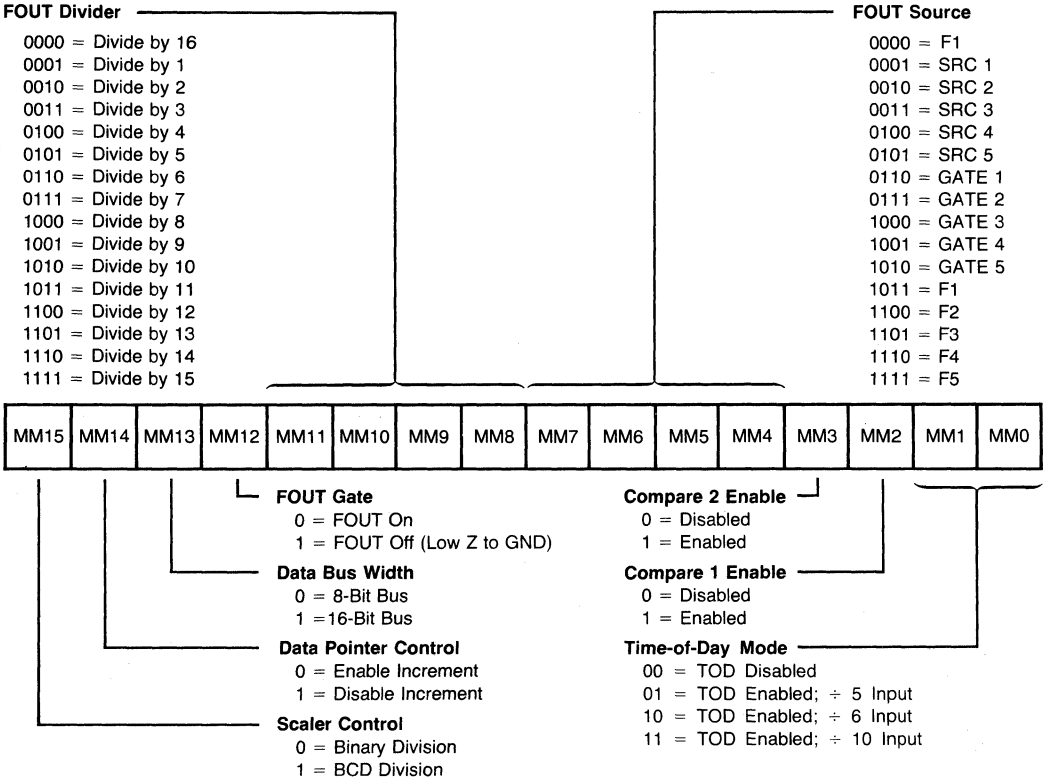
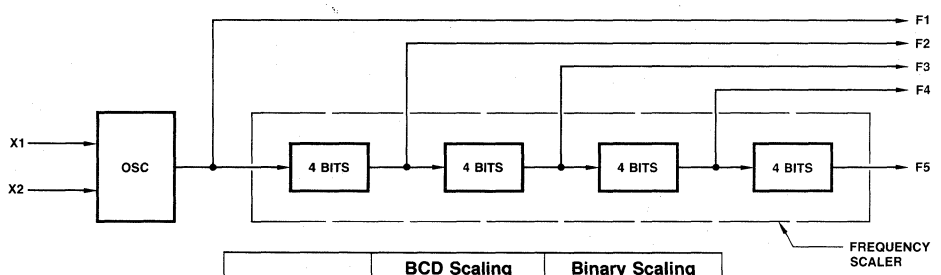


Figure 14. Master Mode Register Bit Assignments.

MOS-180

10



Frequency	BCD Scaling MM15 = 1	Binary Scaling MM15 = 0
F1	OSC	OSC
F2	F1 ÷ 10	F1 ÷ 16
F3	F1 ÷ 100	F1 ÷ 256
F4	F1 ÷ 1,000	F1 ÷ 4,096
F5	F1 ÷ 10,000	F1 ÷ 65,536

Figure 15. Internal Oscillator Frequency Scaler.

MOS-181

Bits MM0 and MM1 of the Master Mode register specify the time-of-day (TOD) options. When MM0 = 0 and MM1 = 0 the special logic used to implement TOD is disabled and counters 1 and 2 will operate in exactly the same way as counters 3, 4 and 5. When MM0 = 1 or MM1 = 1, additional counter decoding and control logic is enabled on counters 1 and 2 which causes their decades to turn over at the counts that generate appropriate 24-hour TOD accumulations.

Figure 16 shows the counter configurations for TOD operation. The least significant decade of Counter 1 is used to scale the input frequency in order to output tenth-of-second periods into the next decade. It can be setup to divide by five, divide by six, or divide by ten. Thus the input frequency for real-time clocking can be, respectively, 50Hz, 60Hz or 100Hz. The input for Counter 2 should be the TC output of Counter 1 for TOD operation. Both counters should be setup for BCD counting and no gating. The Load registers should be used to initialize the clock to the proper time.

Added functions are available in the Counter Logic Groups for counters 1 and 2 (see Figure 2). Each contains a 16-bit Alarm register and a 16-bit Comparator. Bits MM2 and MM3 control the Comparators. When a Comparator is enabled its output is substituted for the normal counter output on the associated OUT1 or OUT2 pin. The polarity definition for the Comparator output will depend on the active-high or active-low definition as programmed in the appropriate Counter Mode register. Once the compare output is true, it will remain so until the count changes and the comparison therefore goes false. The two Comparators can be used individually in most operating modes. A special case occurs when the time-of-day option is invoked and both Comparators are enabled. The operation of Comparator 2 will then be conditioned by Comparator 1 so that a full 32-bit compare must be true in order to generate a true signal on OUT2.

Master Mode bits MM4 through MM7 specify the source input for the FOUT divider. Fifteen inputs are available for selection and they include the five Source pins, the five Gate pins and the five internal frequencies derived from the oscillator. The 16th combination of the four control bits (all zeros) is used to assure that an active frequency is available at the input to the FOUT divider following reset.

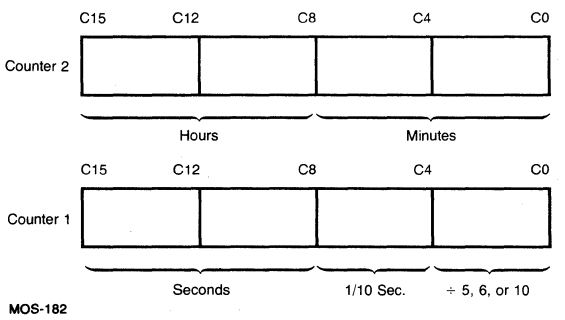
Bits MM8 through MM11 specify the dividing ratio for the FOUT Divider. The FOUT source (selected by bits MM4 through MM7) is divided by an integer value between 1 and 16, inclusive, and is then passed to the FOUT output buffer. After power-on or reset, the FOUT Divider is set to divide by sixteen.

Master Mode bit MM12 provides a software gating capability for the FOUT signal. When MM12 = 1, FOUT is off and in a low impedance state to ground. After power-up or reset, FOUT is gated on.

Bit MM13 controls the multiplexer at the data bus interface in order to configure the part for an 8-bit or 16-bit external bus. The internal bus is always 16-bits wide. When MM13 = 1, 16-bit data is transferred directly between the internal bus and all 16 of the external bus lines. In this configuration, the Byte Pointer bit in the Data Pointer register remains set at all times. When MM13 = 0, 16-bit internal data is transferred a byte at a time to and from the eight low-order external data bus lines. The Byte Pointer bit toggles with each byte transfer in this mode. When operating with an 8-bit data bus width, five of the eight high-order data bus pins (DB8 through DB12) are available for use as auxiliary gate inputs.

Bit MM14 controls the Data Pointer logic to enable or disable the automatic sequencing functions. When MM14 = 1, the contents of the Data Pointer can be changed only directly by entering a command. When MM14 = 0, several types of automatic sequencing of the Data Pointer are available. These are described in the Data Pointer register section of this document.

Bits MM12, MM13 and MM14 can be individually set and reset using commands issued to the Command register. In addition they can all be changed by writing directly to the Master Mode register.



MOS-182

Figure 16. Time-of-Day Storage Configuration.

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.5W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Temperature	VCC	VSS
Am9513DC	0°C ≤ T_A ≤ +70°C	+5V ±5%	0V
Am9513DM	−55°C ≤ T_A ≤ +125°C	+5V ±10%	0V

ELECTRICAL CHARACTERISTICS over operating range (Notes 1 and 2)

Parameter	Description	Test Conditions	Min.	Typ.	Max.	Units
VIL	Input Low Voltage (All inputs except X2)		VSS−0.5		0.8	Volts
VIH	Input High Voltage (All inputs except X2)		2.0		VCC	Volts
VITH	Input Hysteresis (SRC and GATE Inputs Only)		0.2	0.3		Volts
VOL	Output Low Voltage	IOL = 3.2mA			0.4	Volts
		IOL = 4.0mA				
		IOL = 4.5mA				
VOH	Output High Voltage	IOH = −400μA				Volts
		IOH = −2.0mA	2.4			
		IOH = −2.5mA				
IIX	Input Load Current	VSS ≤ VIN ≤ VCC			±10	μA
IOZ	Output Leakage Current	VSS ≤ VOUT ≤ VCC High Impedance State			±25	μA
ICC	VCC Supply Current	T_A = −55°C				mA
		T_A = 0°C				
		T_A = +25°C		160		
CIN	Input Capacitance	f = 1MHz, T_A = +25°C, All pins not under test at 0V.			10	pF
COUT	Output Capacitance				15	
CIO	IN/OUT Capacitance				20	

10

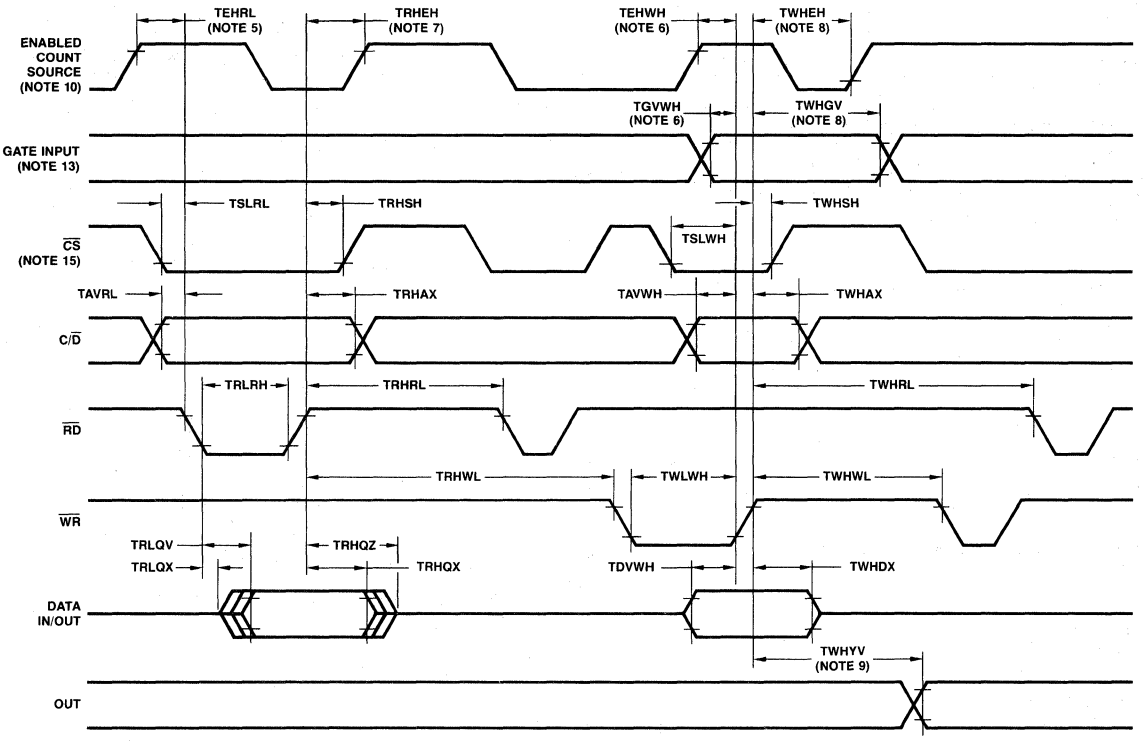


Figure 17. Bus Transfer Switching Waveforms.

MOS-183

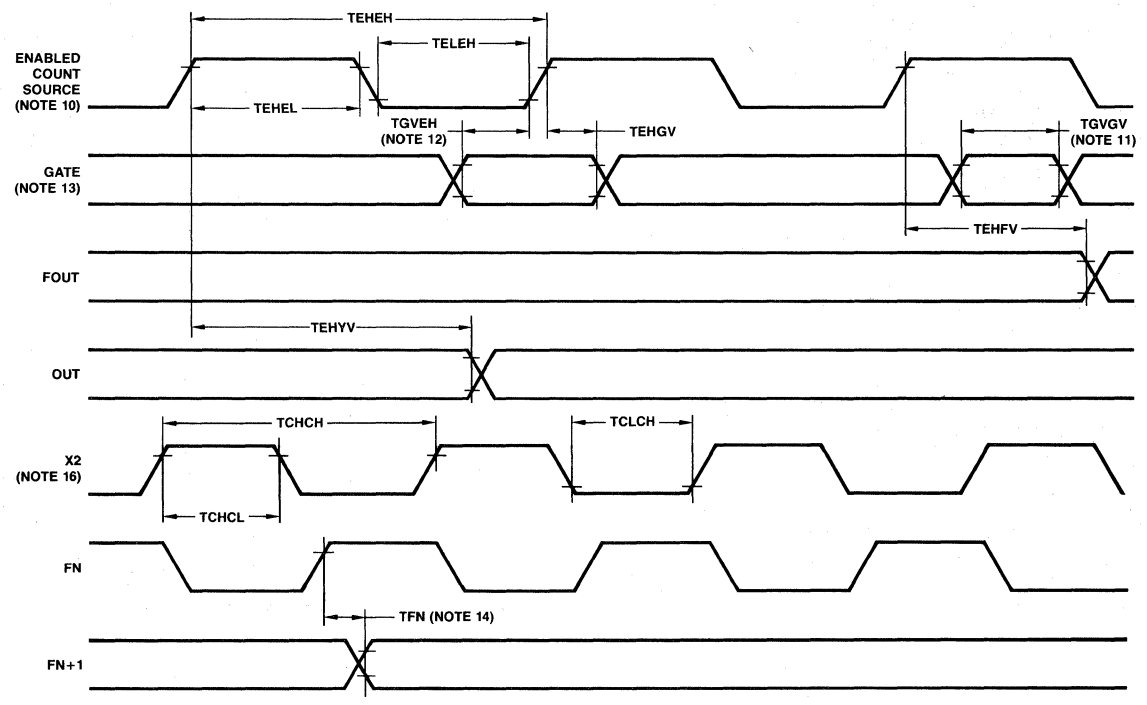


Figure 18. Counter Switching Waveforms.

MOS-184

SWITCHING CHARACTERISTICS over operating range (Notes 2, 3, 4)

Parameter	Description		Figure	Am9513 Min.	Am9513 Max.	Min.	Max.	Unit
TAVRL	C/$\overline{D}$ Valid to Read Low		17	25				ns
TAVWH	C/$\overline{D}$ Valid to Write High		17	70				ns
TCHCH	X2 High to X2 High (X2 Period)		18	145				ns
TCHCL	X2 High to X2 Low (X2 High Pulse Width)		18	70				ns
TCLCH	X2 Low to X2 High (X2 Low Pulse Width)		18	70				ns
TDVWH	Data In Valid to Write High		17	80				ns
TEHEH	Count Source High to Count Source High (Source Cycle Time) (Note 10)		18	145				ns
TEHEL TELEH	Count Source Pulse Duration (Note 10)		18	70				ns
TEHFV	Count Source High to FOUT Valid (Note 10)		18		500			ns
TEHGV	Count Source High to Gate Valid (Level Gating Hold Time) (Notes 10, 12, 13)		18	40				ns
TEHRL	Count Source High to Read Low (Set-up Time) (Notes 5, 10)		17	50				ns
TEHWH	Count Source High to Write High (Set-up Time) (Notes 6, 10)		17	0				ns
TEHYV	Count Source High to Out Valid (Note 10)	TC Output	18		140			ns
		Immediate or Delayed Toggle Output	18		150			
		Comparator Output	18		160			
TFN	FN High to FN+1 Valid (Note 14)		18		75			ns
TGVEH	Gate Valid to Count Source High (Level Gating Set-up Time) (Notes 10, 12, 13)		18	40				ns
TGVGV	Gate Valid to Gate Valid (Gate Pulse Duration) (Notes 11, 13)		18	145				ns
TGVWH	Gate Valid to Write High (Notes 6, 13)		17	0				ns
TRHAX	Read High to C/$\overline{D}$ Don't Care		17	0				ns
TRHEH	Read High to Count Source High (Notes 7, 10)		17	0				ns
TRHQX	Read High to Data Out Invalid		17	20				ns
TRHQZ	Read High to Data Out at High Impedance (Data Bus Release Time)		17		70			ns
TRHRL	Read High to Read Low (Read Recovery Time)		17		1000			ns
TRHSH	Read High to $\overline{CS}$ High (Note 15)		17	0				ns
TRHWL	Read High to Write Low (Read Recovery Time)		17		1000			ns
TRLQV	Read Low to Data Out Valid		17		160			ns
TRLQX	Read Low to Data Bus Driven (Data Bus Drive Time)		17	50				ns
TRLRH	Read Low to Read High (Read Pulse Duration) (Note 15)		17	160				ns
TSLRL	CS Low to Read Low (Note 15)		17	20				ns
TSLWH	CS Low to Write High (Note 15)		17	80				ns
TWHAX	Write High to C/$\overline{D}$ Don't Care		17	0				ns
TWHDX	Write High to Data In Don't Care		17	0				ns
TWHEH	Write High to Count Source High (Notes 8, 10)		17	400				ns
TWHGV	Write High to Gate Valid (Notes 8, 13)		17	400				ns
TWHRL	Write High to Read Low (Write Recovery Time)		17		1000			ns
TWHSH	Write High to $\overline{CS}$ High (Note 15)		17	0				ns
TWHWL	Write High to Write Low (Write Recovery Time)		17		1000			ns
TWHYV	Write High to Out Valid (Note 9)		17		500			ns
TWLWH	Write Low to Write High (Write Pulse Duration) (Note 15)		17	150				ns

PRELIMINARY

NOTES:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. Test conditions assume transition times of 10ns or less, timing reference levels of 0.8V and 2.0V and output loading of one TTL gate plus 100pF, unless otherwise noted.
3. Abbreviations used for the switching parameter symbols are given as the letter T followed by four or five characters. The first and third characters represent the signal names on which the measurements start and end. Signal abbreviations used are:

   A (Address) = C/D
   C (Clock) = X2
   D (Data In) = DB0-DB15
   E (Enabled counter source input) = SRC1-SRC5, GATE1-GATE5, F1-F5, TCN−1
   F = FOUT
   G (Counter gate input) = GATE1-GATE5, TCN−1
   Q (Data Out) = DB0-DB15
   R (Read) = RD
   S (Chip Select) = CS
   W (Write) = WR
   Y (Output) = OUT1-OUT5

   The second and fourth letters designate the reference states of the signals named in the first and third letters respectively, using the following abbreviations.

   H = High
   L = Low
   V = Valid
   X = unknown or don't care
   Z = high impedance

4. Switching parameters are listed in alphabetical order.
5. Any input transition that occurs before this minimum setup requirement will be reflected in the contents read from the status register.
6. Any input transition that occurs before this minimum setup requirement will act on the counter before the execution of the operation initiated by the write.
7. Any input transition that occurs after this minimum hold time is guaranteed to not influence the contents read from the status register on the current read operation.
8. Any input transition that occurs after this minimum hold time is guaranteed to be seen by the counter as occurring after the action initiated by the write operation.
9. This parameter applies to cases where the write operation causes a change in the output bit.
10. The enabled count source is one of F1-F5, TCN−1, SRC1-SRC5 or GATE1-GATE5, as selected in the applicable Counter Mode register. The timing diagram assumes the counter counts on rising source edges. The timing specifications are the same for falling-edge counting.
11. This parameter applies to edge gating (CM15-CM13 = 110 or 111) and gating when both CM7 = 1 and CM15-CM13 ≠ 000. This parameter represents the minimum GATE pulse width needed to ensure that the pulse initiates counting or counter reloading.
12. This parameter applies to level gating (CM15-CM13 = 001 through 101) and gating when both CM7 = 1 and CM15-CM13 = 000. This parameter represents the minimum setup or hold times to ensure that the Gate input is seen at the intended level on the active source edge.
13. This parameter assumes that the GATENA input is unused (16-bit bus mode) or is tied high. In cases where the GATENA input is used, this timing specification must be met by both the GATE and GATENA inputs.
14. Signals F1-F5 cannot be directly monitored by the user. The phase difference between these signals will manifest itself by causing counters using two different F signals to count at different times on nominally simultaneous transitions in the F signals.
15. This timing specification assumes that $\overline{CS}$ is active whenever $\overline{RD}$ or $\overline{WR}$ are active. $\overline{CS}$ may be held active indefinitely.
16. This parameter assumes X2 is driven with a TTL-level square wave.

APPLICATION INFORMATION

The X1 and X2 inputs can be driven with a RC network, an external TTL-level square wave, or a crystal. Figure 19 shows the suggested methods of connecting different frequency sources to the internal oscillator input.

The use of a crystal provides a highly accurate frequency source at moderate cost, and accordingly, will usually be the preferred method of operation. The Am9513 is designed to use a crystal in a parallel-resonant mode. The two ceramic capacitors connecting X1 and X2 to ground ensure proper loading on the crystal. The capacitor to X2 may be an adjustable type for fine-tuning the resonant frequency for critical applications.

An RC network provides a very low cost frequency source but may exhibit large frequency variations over recommended power supply and temperature ranges. Note that there is a resistor internal to the Am9513 in parallel with any external resistance.

Initialization Procedures

The reset function in the Am9513 is accomplished in two ways: automatically during power-up and by software Master Reset command. Power-up reset circuitry is internally triggered by the rising VCC voltage when a predetermined threshold is reached. An internal flip-flop is set by the rising supply voltage and controls the reset operation. The reset flip-flop remains set until cleared by the first active Chip Select input. A reset may also by initiated by the host processor by entering the Master Reset command. This software reset is active for the duration of the command write; otherwise it performs the same function as the power-up reset.

Following either type of Reset, all five counters are disabled, 0B00 is loaded into each Counter Mode register, and 0000 is loaded in the Master Mode register. This results in each counter being configured to count down in binary on the positive-going edge of the internal F1 frequency source with no repetition or gating. The Master Mode register is cleared to configure the Am9513 for an 8-bit data bus width; binary division of the internal oscillator; FOUT gated on and set to divide F1 by 16; time-of-day mode and comparators 1 and 2 disabled; and the Data Pointer increment enabled.

Reset will clear the Load and Hold registers for each counter but will not change either the counter contents or the Data Pointer register.

The following initialization procedure should be followed on Counters 1 and 2 when Time-of-Day mode is selected.

1. Set Time-of-Day enabled in the Master Mode register and load Counter Mode registers 1 and 2.
2. If Time-of-Day is to count up, load 0000 in Load registers 1 and 2 and execute command FF43 (Load) to load this value into the counters. This step conditions the count circuitry.
3. Load the desired start time into the Load registers and execute command FF43 again.
4. For counting up, load Load registers 1 and 2 with 0000.
5. Counters 1 and 2 may now be armed.

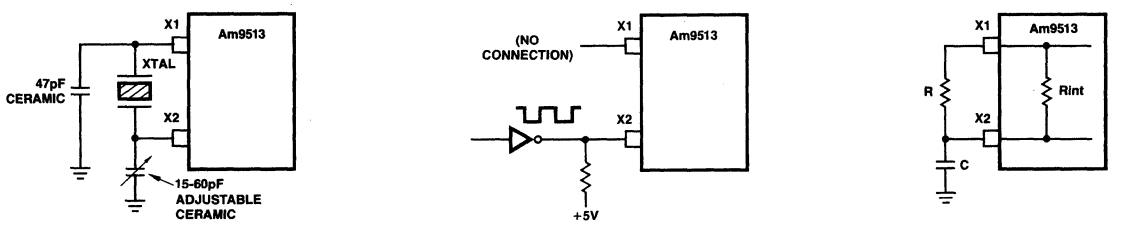
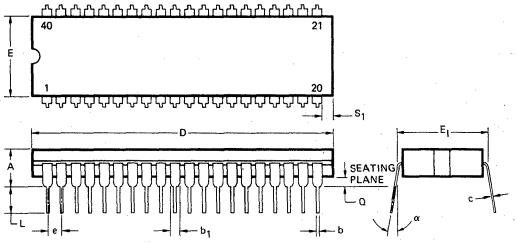


Figure 19. Driving the X1 and X2 Inputs.

PHYSICAL DIMENSIONS

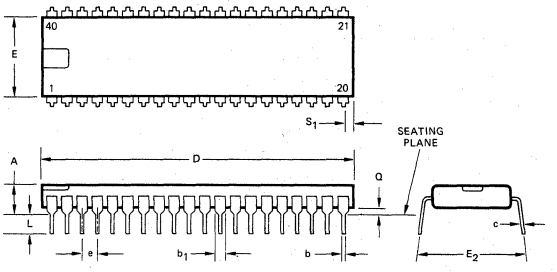
40-Pin Cerdip



Reference Symbol	Inches	
	Min.	Max.
A	.150	.225
b	.016	.020
b_1	.045	.065
c	.009	.011
D	2.020	2.100
E	.510	.550
E_1	.600	.630
e	.090	.110
L	.120	.150
Q	.015	.060
S_1*	.005	
α	3°	13°

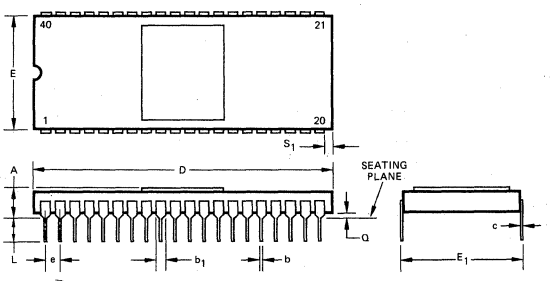
*From edge of end lead.

40-Pin Molded DIP



Reference Symbol	Inches	
	Min.	Max.
A	.150	.200
b	.015	.020
b_1	.055	.065
c	.009	.011
D	2.050	2.080
E	.530	.550
E_2	.585	.700
e	.090	.110
L	.015	.060
Q	.015	.060
S_1	.040	.070

40-Pin Side-Brazed Ceramic



Reference Symbol	Inches	
	Min.	Max.
A	.100	.200
b	.015	.022
b_1	.030	.060
c	.008	.013
D	1.960	2.040
E	.550	.610
E_1	.590	.620
e	.090	.110
L	.120	.160
Q	.020	.060
S_1	.005	

Am9517A
Multimode DMA Controller

DISTINCTIVE CHARACTERISTICS

- Four independent DMA channels, each with separate registers for Mode Control, Current Address, Base Address, Current Word Count and Base Word Count.
- Transfer modes: Block, Demand, Single Word, Cascade
- Independent autoinitialization of all channels
- Memory-to-memory transfers
- Memory block initialization
- Address increment or decrement
- Master system disable
- Enable/disable control of individual DMA requests
- Directly expandable to any number of channels
- End of Process input for terminating transfers
- Software DMA requests
- Independent polarity control for DREQ and DACK signals
- Compressed timing option speeds transfers – up to 2M words/second
- +5 volt power supply
- Advanced N-channel silicon gate MOS technology
- 40 pin Hermetic DIP package
- 100% MIL-STD-883 reliability assurance testing

GENERAL DESCRIPTION

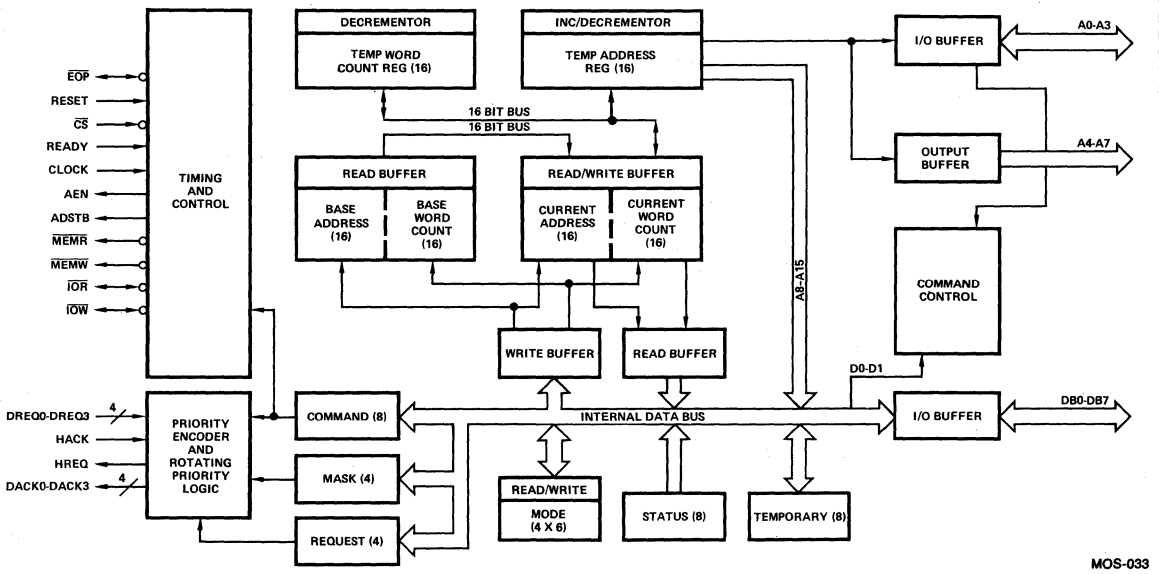
The **Am9517A** Multimode Direct Memory Access (DMA) Controller is a peripheral interface circuit for microprocessor systems. It is designed to improve system performance by allowing external devices to directly transfer information to or from the system memory. Memory-to-memory transfer capability is also provided. The Am9517A offers a wide variety of programmable control features to enhance data throughput and system optimization and to allow dynamic reconfiguration under program control.

The **Am9517A** is designed to be used in conjunction with an external 8-bit address register such as the Am74LS373. It contains four independent channels and may be expanded to any number of channels by cascading additional controller chips.

The three basic transfer modes allow programmability of the types of DMA service by the user. Each channel can be individually programmed to Autoinitialize to its original condition following an End of Process ($\overline{EOP}$).

Each channel has a full 64K address and word count capability. An external $\overline{EOP}$ signal can terminate a DMA or memory-to-memory transfer. This is useful for block search or compare operations using external comparators or for intelligent peripherals to abort erroneous services.

BLOCK DIAGRAM

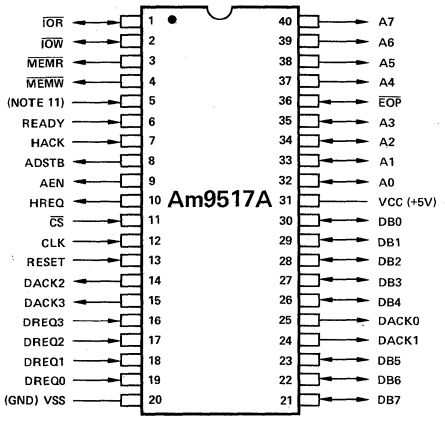


MOS-033

ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency	
		3MHz	4MHz
Hermetic DIP/ Molded DIP	$0°C \leq T_A \leq +70°C$	AM9517ADC/PC AM9517A-1DC/PC	AM9517A-4DC/PC
Hermetic DIP	$-55°C \leq T_A \leq +125°C$	AM9517ADM	

CONNECTION DIAGRAM



Top View
Pin 1 is marked for orientation.

Figure 1. MOS-034

INTERFACE SIGNAL DESCRIPTION

VCC: +5 Volt Supply
VSS: Ground

CLK (Clock, Input)

This input controls the internal operations of the Am9517A and its rate of data transfers. The input may be driven at up to 3MHz for the standard Am9517A and up to 4MHz for the Am9517A-4.

$\overline{CS}$ (Chip Select, Input)

Chip Select is an active low input used to select the Am9517A as an I/O device during an I/O Read or I/O Write by the host CPU. This allows CPU communication on the data bus. During multiple transfers to or from the Am9517A by the host CPU, $\overline{CS}$ may be held low providing $\overline{IOR}$ or $\overline{IOW}$ is toggled following each transfer.

RESET (Reset, Input)

Reset is an asynchronous active high input which clears the Command, Status, Request and Temporary registers. It also clears the first/last flip/flop and sets the Mask register. Following a Reset the device is in the Idle cycle.

READY (Ready, Input)

Ready is an input used to extend the memory read and write pulses from the Am9517A to accommodate slow memories or I/O peripheral devices.

HACK (Hold Acknowledge, Input)

The active high Hold Acknowledge from the CPU indicates that control of the system buses has been relinquished.

DREQ0-DREQ3 (DMA Request, Input)

The DMA Request lines are individual asynchronous channel request inputs used by peripheral circuits to obtain DMA service. In Fixed Priority, DREQ0 has the highest priority and DREQ3 has the lowest priority. Polarity of DREQ is programmable. Reset initializes these lines to active high.

DB0-DB7 (Data Bus, Input/Output)

The Data Bus lines are bidirectional three-state signals connected to the system data bus. The outputs are enabled during the I/O Read by the host CPU, permitting the CPU to examine the contents of an Address register, the Status register, the Temporary register or a Word Count register. The Data Bus is enabled to input data during a host CPU I/O write, allowing the CPU to program the Am9517A control registers. During DMA cycles the most significant eight bits of the address are output onto the data bus to be strobed into an external latch by ADSTB. In memory-to-memory operations data from the source memory location comes into the Am9517A's Temporary register on the read-from-memory half of the operation. On the write-to-memory half of the operation, the data bus outputs the Temporary register data into the destination memory location.

$\overline{IOR}$ (I/O Read, Input/Output)

I/O Read is a bidirectional active low three-state line. In the Idle cycle, it is an input control signal used by the CPU to read the control registers. In the Active cycle, it is an output control signal used by the Am9517A to access data from a peripheral during a DMA Write transfer.

$\overline{IOW}$ (I/O Write, Input/Output)

I/O Write is a bidirectional active low three-state line. In the Idle cycle it is an input control signal used by the CPU to load information into the Am9517A. In the Active cycle it is an output control signal used by the Am9517A to load data to the peripheral during a DMA Read transfer.

Write operations by the CPU to the Am9517A require a rising $\overline{WR}$ edge following each data byte transfer. It is not sufficient to hold the $\overline{IOW}$ pin low and toggle $\overline{CS}$.

$\overline{EOP}$ (End of Process, Input/Output)

$\overline{EOP}$ is an active low bidirectional open-drain signal providing information concerning the completion of DMA service. When a channel's Word Count goes to zero, the Am9517A pulses $\overline{EOP}$ low to provide the peripheral with a completion signal. $\overline{EOP}$ may also be pulled low by the peripheral to cause premature completion. The reception of $\overline{EOP}$, either internal or external, causes the currently active channel to terminate the service, to set its TC bit in the Status register and to reset its request bit. If Autoinitialization is selected for the channel, the current registers will be updated from the base registers. Otherwise the channel's mask bit will be set and the register contents will remain unaltered.

During memory-to-memory transfers, $\overline{EOP}$ will be output when the TC for channel 1 occurs. $\overline{EOP}$ always applies to the channel with an active DACK; external $\overline{EOPS}$ are disregarded in DACK0-DACK3 are all inactive.

Because $\overline{EOP}$ is an open-drain signal, an external pullup resistor is required. Values of 3.3K or 4.7K are recommended; the $\overline{EOP}$ pin can not sink the current passed by a 1K pullup.

A0-A3 (Address, Input/Output)

The four least significant address lines are bidirectional 3-state signals. During DMA Idle cycles they are inputs and allow the host CPU to load or read control registers. When the DMA is active, they are outputs and provide the lower 4-bits of the output address.

A4-A7 (Address, Output)

The four most significant address lines are three-state outputs and provide four bits of address. These lines are enabled only during DMA service.

HREQ (Hold Request, Output)

The Hold Request to the CPU is used by the DMA to request control of the system bus. Software requests or unmasked DREQs cause the Am9517A to issue HREQ.

DACK0-DACK3 (DMA Acknowledge, Output)

The DMA Acknowledge lines indicate that a channel is active. In many systems they will be used to select a peripheral. Only one DACK will be active at a time and none will be active unless the DMA is in control of the bus. The polarity of these lines is programmable. Reset initializes them to active-low.

AEN (Address Enable, Output)

Address Enable is an active high signal used to disable the system bus during DMA cycles to enable the output of the external latch which holds the upper byte of the address. Note that during DMA transfers HACK and AEN should be used to deselect all other I/O peripherals which may erroneously be accessed as programmed I/O during the DMA operation. The Am9517A automatically deselects itself by disabling the $\overline{CS}$ input during DMA transfers.

ADSTB (Address Strobe, Output)

The active high Address Strobe is used to strobe the upper address byte from DB0-DB7 into an external latch.

$\overline{MEMR}$ (Memory Read, Output)

The Memory Read signal is an active low three-state output used to access data from the selected memory location during a memory-to-peripheral or a memory-to-memory transfer.

Name	Size	Number
Base Address Registers	16 bits	4
Base Word Count Registers	16 bits	4
Current Address Registers	16 bits	4
Current Word Count Registers	16 bits	4
Temporary Address Register	16 bits	1
Temporary Word Count Register	16 bits	1
Status Register	8 bits	1
Command Register	8 bits	1
Temporary Register	8 bits	1
Mode Registers	6 bits	4
Mask Register	4 bits	1
Request Register	4 bits	1

Figure 2. Am9517A Internal Registers.

$\overline{MEMW}$ (Memory Write, Output)

The Memory Write signal is an active low three-state output used to write data to the selected memory location during a peripheral-to-memory or a memory-to-memory transfer.

FUNCTIONAL DESCRIPTION

The Am9517A block diagram includes the major logic blocks and all of the internal registers. The data interconnection paths are also shown. Not shown are the various control signals between the blocks. The Am9517A contains 344 bits of internal memory in the form of registers. Figure 2 lists these registers by name and shows the size of each. A detailed description of the registers and their functions can be found under Register Description.

The Am9517A contains three basic blocks of control logic. The Timing Control block generates internal timing and external control signals for the Am9517A. The Program Command Control block decodes the various commands given to the Am9517A by the microprocessor prior to servicing a DMA Request. It also decodes each channel's Mode Control word. The Priority Encoder block resolves priority contention among DMA channels requesting service simultaneously.

The Timing Control block derives internal timing from the clock input. In Am9080A systems this input will usually be the $\phi 2$ TTL clock from an Am8224. However, any appropriate system clock will suffice.

DMA Operation

The Am9517A is designed to operate in two major cycles. These are called Idle and Active cycles. Each device cycle is made up of a number of states. The Am9517A can assume seven separate states, each composed of one full clock period. State 1 (S1) is the inactive state. It is entered when the Am9517A has no valid DMA requests pending. While in S1, the DMA controller is inactive but may be in the Program Condition, being programmed by the processor. State 0 (S0) is the first state of a DMA service. The Am9517A has requested a hold but the processor has not yet returned an acknowledge. An acknowledge from the CPU will signal that transfers may begin. S1, S2, S3 and S4 are the working states of the DMA service. If more time is needed to complete a transfer than is available with normal timing, wait states (SW) can be inserted before S4 by the use of the Ready line on the Am9517A.

Memory-to-memory transfers require a read-from and a write-to-memory to complete each transfer. The states, which resemble the normal working states, use two digit numbers for identification. Eight states are required for each complete transfer. The first four states (S11, S12, S13, S14) are used for the read-from-memory half and the last four states (S21, S22, S23 and S24) for the write-to-memory half of the transfer. The Temporary Data register is used for intermediate storage of the memory byte.

IDLE Cycle

When no channel is requesting service, the Am9517A will enter the Idle cycle and perform "S1" states. In this cycle the Am9517A will sample the DREQ lines every clock cycle to determine if any channel is requesting a DMA service. The device will also sample $\overline{CS}$, looking for an attempt by the microprocessor to write or read the internal registers of the Am9517A. When $\overline{CS}$ is low and HACK is low the Am9517A enters the Program Condition. The CPU can now establish, change or inspect the internal definition of the part by reading from or writing to the internal registers. Address lines A0-A3 are inputs to the device and select which registers will be read or written. The $\overline{IOR}$ and $\overline{IOW}$ lines are used to select and time reads or writes. Due to the

10

number and size of the internal registers, an internal flip/flop is used to generate an additional bit of address. This bit is used to determine the upper or lower byte of the 16-bit Address and Word Count registers. The flip/flop is reset by Master Clear or Reset. A separate software command can also reset this flip/flop.

Special software commands can be executed by the Am9517A in the Program Condition. These commands are decoded as sets of addresses when both $\overline{CS}$ and $\overline{IOW}$ are active and do not make use of the data bus. Functions include Clear First/Last Flip/Flop and Master Clear.

ACTIVE CYCLE

When the Am9517A is in the Idle cycle and a channel requests a DMA service, the device will output a HREQ to the microprocessor and enter the Active cycle. It is in this cycle that the DMA service will take place, in one of four modes:

Single Transfer Mode: In Single Transfer mode, the Am9517A will make a one-byte transfer during each HREQ/HACK handshake. When DREQ goes active, HREQ will go active. After the CPU responds by driving HACK active, a one-byte transfer will take place. Following the transfer, HREQ will go inactive, the word count will be decremented and the address will be either incremented or decremented. When the word count goes to zero a Terminal Count (TC) will cause an Autoinitialize if the channel has been programmed to do so.

To perform a single transfer, DREQ must be held active only until the corresponding DACK goes active. If DREQ is held continuously active, HREQ will go inactive following each transfer and then will go active again and a new one-byte transfer will be made following each rising edge of HACK. In 8080A/9080A systems this will ensure one full machine cycle of execution between DMA transfers. Details of timing between the Am9517A and other bus control protocols will depend upon the characteristics of the microprocessor involved.

Block Transfer Mode: In Block Transfer mode, the Am9517A will continue making transfers until a TC (caused by the word count going to zero) or an external End of Process ($\overline{EOP}$) is encountered. DREQ need be held active only until DACK becomes active. An autoinitialize will occur at the end of the service if the channel has been programmed for it.

Demand Transfer Mode: In Demand Transfer mode the device will continue making transfers until a TC or external $\overline{EOP}$ is encountered or until DREQ goes inactive. Thus, the device requesting service may discontinue transfers by bringing DREQ inactive. Service may be resumed by asserting an active DREQ once again. During the time between services when the microprocessor is allowed to operate, the intermediate values of address and word count may be read from the Am9517A Current Address and Current Word Count registers. Autoinitialization will only occur following a TC or $\overline{EOP}$ at the end of service. Following Autoinitialization, an active-going DREQ edge is required to initiate a new DMA service.

Cascade Mode: This mode is used to cascade more than one Am9517A together for simple system expansion. The HREQ and HACK signals from the additional Am9517A are connected to the DREQ and DACK signals of a channel of the initial Am9517A. This allows the DMA requests of the additional device to propagate through the priority network circuitry of the preceding device. The priority chain is preserved and the new device must wait for its turn to acknowledge requests. Since the cascade channel in the initial device is used only for prioritizing the additional device, it does not output any address or control signals of its own. These would conflict with the outputs of the active channel in the added device. The Am9517A will respond to DREQ with DACK but all other outputs except HREQ will be disabled.

Figure 3 shows two additional devices cascaded into an initial device using two of the previous channels. This forms a two level DMA system. More Am9517As could be added at the second level by using the remaining channels of the first level. Additional devices can also be added by cascading into the channels of the second level devices forming a third level.

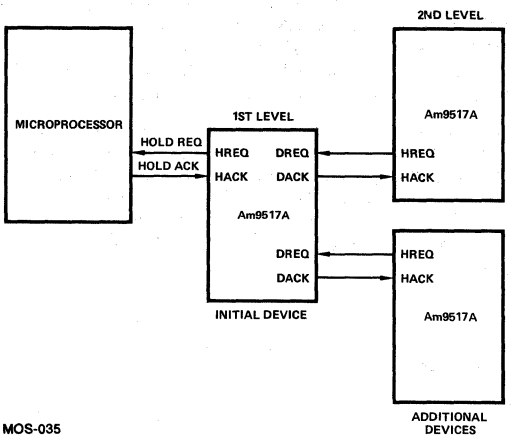


Figure 3. Cascaded Am9517As.

TRANSFER TYPES

Each of the three active transfer modes can perform three different types of transfers. These are Read, Write and Verify. Write transfers move data from an I/O device to the memory by activating $\overline{IOR}$ and $\overline{MEMW}$. Read transfers move data from memory to an I/O device by activating $\overline{MEMR}$ and $\overline{IOW}$. Verify transfers are pseudo transfers; the Am9517A operates as in Read or Write transfers generating addresses, responding to $\overline{EOP}$, etc., however, the memory and I/O control lines remain inactive.

Memory-to-Memory: The Am9517A includes a block move capability that allows blocks of data to be moved from one memory address space to another. When Bit C0 in the Command register is set to a logical 1, channels 0 and 1 will operate as memory-to-memory transfer channels. Channel 0 forms the source address and channel 1 forms the destination address. The channel 1 word count is used. A memory-to-memory transfer is initiated by setting a software DMA request for channel 0. Block Transfer Mode should be used for memory-to-memory. When channel 0 is programmed for a fixed source address, a single source word may be written into a block of memory.

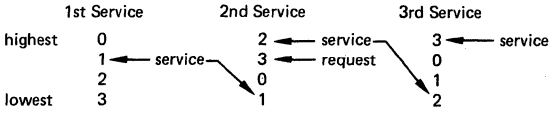
When setting up the Am9517A for memory-to-memory operation, it is suggested that both channels 0 and 1 be masked out. Further, the channel 0 word count should be initialized to the same value used in channel 1. No DACK outputs will be active during memory-to-memory transfers.

The Am9517A will respond to external $\overline{EOP}$ signals during memory-to-memory transfers. Data comparators in block search schemes may use this input to terminate the service when a match is found. The timing of memory-to-memory transfers may be found in Timing Diagram 4.

Autointialize: By programming a bit in the Mode register a channel may be set up for an Autoinitialize operation. During Autoinitialization, the original values of the Current Address and Current Word Count registers are automatically restored from the Base Address and Base Word Count registers of that channel following $\overline{EOP}$. The base registers are loaded simultaneously with the current registers by the microprocessor and remain unchanged throughout the DMA service. The mask bit is not set by $\overline{EOP}$ when the channel is in Autoinitialize. Following Autoinitialize the channel is ready to repeat its service without CPU intervention.

Priority: The Am9517A has two types of priority encoding available as software selectable options. The first is Fixed Priority which fixes the channels in priority order based upon the descending value of their number. The channel with the lowest priority is 3 followed by 2, 1 and the highest priority channel, 0.

The second scheme is Rotating Priority. The last channel to get service becomes the lowest priority channel with the others rotating accordingly. With Rotating Priority in a single chip DMA system, any device requesting service is guaranteed to be recognized after no more than three higher priority services have occurred. This prevents any one channel from monopolizing the system.



The priority encoder selects the highest priority channel requesting service on each active-going HACK edge. Once a channel is started, its operation will not be suspended if a request is received by a higher priority channel. The high priority channel will only gain control after the lower priority channel releases HREQ. When control is passed from one channel to another, the CPU will always gain bus control. This ensures generation of rising HACK edge to be used to initiate selection of the new highest-priority requesting channel.

Compressed Timing: In order to achieve even greater throughput where system characteristics permit, the Am9517A can compress the transfer time to two clock cycles. From Timing Diagram 3 it can be seen that state S3 is used to extend the access time of the read pulse. By removing state S3 the read pulse width is made equal to the write pulse width and a transfer consists only of state S2 to change the address and state S4 to perform the read/write. S1 states will still occur when A8-A15 need updating (see Address Generation). Timing for compressed transfers is found in Timing Diagram 6.

Address Generation: In order to reduce pin count, the Am9517A multiplexes the eight higher order address bits on the data lines. State S1 is used to output the higher order address bits to an external latch from which they may be placed on the address bus. The falling edge of Address Strobe (ADSTB) is used to load these bits from the data lines to the latch. Address Enable (AEN) is used to enable the bits onto the address bus through a 3-state enable. The lower order address bits are output by the Am9517A directly. Lines A0-A7 should be connected to the address bus. Timing Diagram 3 shows the time relationships between CLK, AEN, ADSTB, DB0-DB7 and A0-A7.

During Block and Demand Transfer mode services which include multiple transfers, the addresses generated will be sequential. For many transfers the data held in the external address latch will remain the same. This data need only change when a carry or borrow from A7 to A8 takes place in the normal sequence of addresses. To save time and speed transfers, the Am9517A executes S1 states only when updating of A8-A15 in the latch is necessary. This means for long services, S1 states may occur only once every 256 transfers, a savings of 255 clock cycles for each 256 transfers.

REGISTER DESCRIPTION

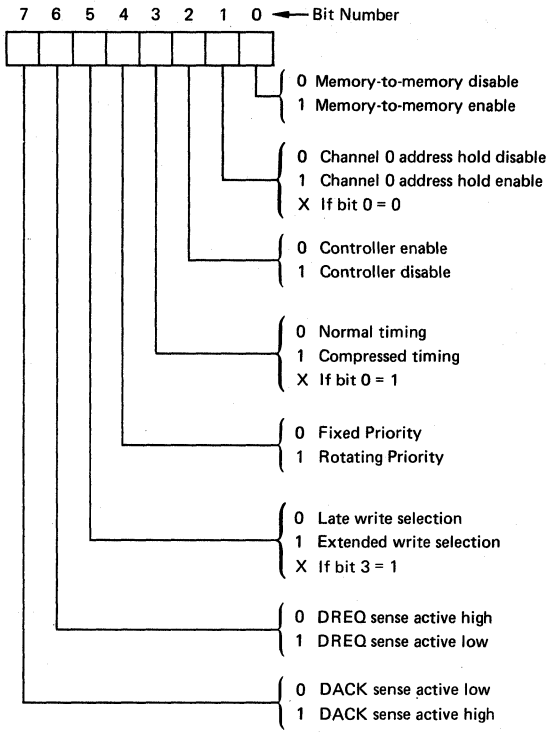
Current Address Register: Each channel has a 16-bit Current Address register. This register holds the value of the address used during DMA transfers. The address is automatically incremented or decremented after each transfer and the intermediate values of the address are stored in the Current Address register during the transfer. This register is written or read by the microprocessor in successive 8-bit bytes. It may also be reinitialized by an Autoinitialize back to its original value. Autoinitialization takes place only after an $\overline{EOP}$.

Current Word Count Register: Each channel has a 16-bit Current Word Count register. This register should be programmed with, and will return on a CPU read, a value one less than the number of words to be transferred. The word count is decremented after each transfer. The intermediate value of the word count is stored in the register during the transfer. When the value in the register goes to zero, a TC will be generated. This register is loaded or read in successive 8-bit bytes by the microprocessor in the Program Condition. Following the end of a DMA service it may also be reinitialized by an Autoinitialize back to its original value. Autoinitialize can occur only when an $\overline{EOP}$ occurs. Note that the contents of the Word Count register will be FFFF (hex) following on internally generated $\overline{EOP}$.

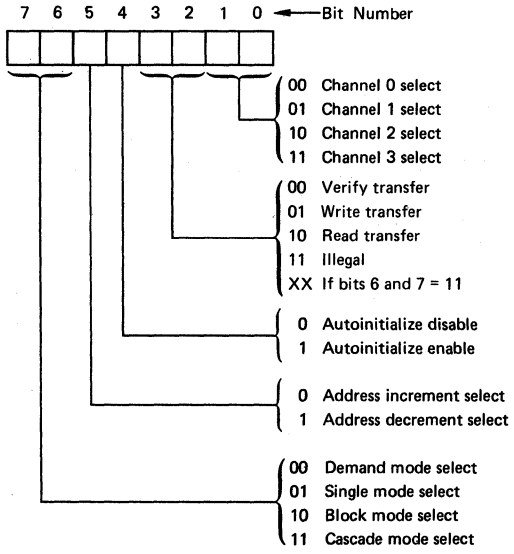
Base Address and Base Word Count Registers: Each channel has a pair of Base Address and Base Word Count registers. These 16-bit registers store the original values of their associated current registers. During Autoinitialize these values are used to restore the current registers to their original values. The base registers are written simultaneously with their corresponding current register in 8-bit bytes during DMA programming by the microprocessor. Accordingly, writing to these registers when intermediate values are in the Current registers will overwrite the intermediate values. The Base registers cannot be read by the microprocessor.

10

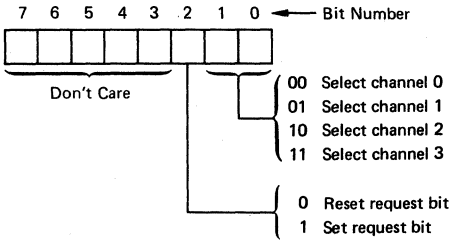
Command Register: This 8-bit register controls the operation of the Am9517A. It is programmed by the microprocessor in the Program Condition and is cleared by Reset. The following table lists the function of the command bits. See Figure 4 for address coding.



Mode Register: Each channel has a 6-bit Mode register associated with it. When the register is being written to by the microprocessor in the Program Condition, bits 0 and 1 determine which channel Mode register it to be written.

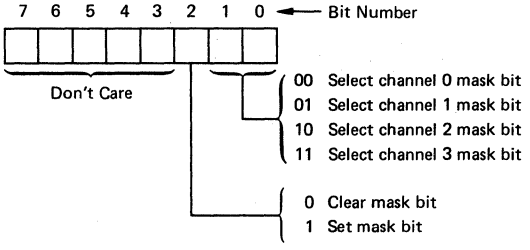


Request Register: The Am9517A can respond to requests for DMA service which are initiated by software as well as by a DREQ. Each channel has a request bit associated with it in the 4-bit Request register. These are nonmaskable and subject to prioritization by the Priority Encoder network. Each register bit is set or reset separately under software control or is cleared upon generation of a TC or external EOP. The entire register is cleared by a Reset. To set or reset a bit, the software loads the proper form of the data word. See Figure 4 for address coding.

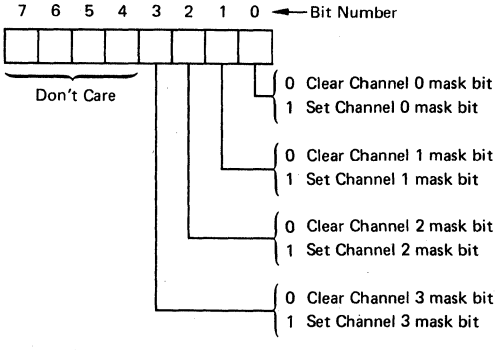


Software requests will be serviced only if the channel is in Block mode. When initiating a memory-to-memory transfer, the software request for channel 0 should be set.

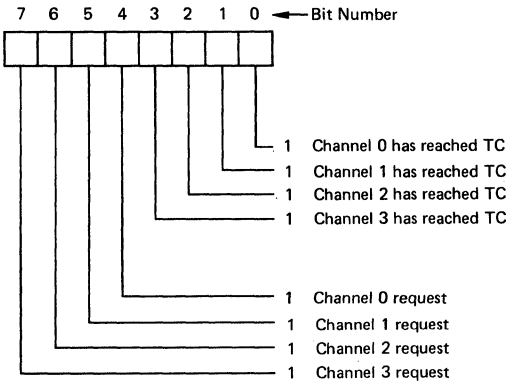
Mask Register: Each channel has associated with it a mask bit which can be set to disable the incoming DREQ. Each mask bit is set when its associated channel produces an EOP if the channel is not programmed for Autoinitialize. Each bit of the 4-bit Mask register may also be set or cleared separately under software control. The entire register is also set by a Reset. This disables all DMA requests until a clear Mask register instruction allows them to occur. The instruction to separately set or clear the mask bits is similar in form to that used with the Request register. See Figure 4 for instruction addressing.



All four bits of the Mask Register may also be written with a single command.



Status Register: The Status registers may be read out of the Am9517A by the microprocessor. It indicates which channels have reached a terminal count and which channels have pending DMA requests. Bits 0-3 are set each time a TC is reached by that channel, including after each Autoinitialization. These bits are cleared by Reset and each Status Read. Bits 4-7 are set whenever their corresponding channel is requesting service.



Temporary Register: The Temporary register is used to hold data during memory-to-memory transfers. Following the completion of the transfers, the last word moved can be read by the microprocessor in the Program Condition. The Temporary register always contains the last byte transferred in the previous memory-to-memory operation, unless cleared by a Reset.

Software Commands: There are two special software commands which can be executed in the Program Condition. They do not depend on any specific bit pattern on the data bus. The two software commands are:

Clear First/Last Flip/Flop: This command may be issued prior to writing or reading Am9517A address or word count information. This initializes the flip/flop to a known state so that subsequent accesses to register contents by the microprocessor will address lower and upper bytes in the correct sequence.

Master Clear: This software instruction has the same effect as the hardware Reset. The Command, Status, Request, Temporary and Internal First/Last Flip/Flop registers are cleared and the Mask register is set. The Am9517A will enter the Idle cycle.

Figure 4 lists the address codes for the software commands.

Interface Signals						Operation
A3	A2	A1	A0	$\overline{IOR}$	$\overline{IOW}$	
1	0	0	0	0	1	Read Status Register
1	0	0	0	1	0	Write Command Register
1	0	0	1	0	1	Illegal
1	0	0	1	1	0	Write Request Register
1	0	1	0	0	1	Illegal
1	0	1	0	1	0	Write Single Mask Register Bit
1	0	1	1	0	1	Illegal
1	0	1	1	1	0	Write Mode Register
1	1	0	0	0	1	Illegal
1	1	0	0	1	0	Clear Byte Pointer Flip/Flop
1	1	0	1	0	1	Read Temporary Register
1	1	0	1	1	0	Master Clear
1	1	1	0	0	1	Illegal
1	1	1	0	1	0	Illegal
1	1	1	1	0	1	Illegal
1	1	1	1	1	0	Write All Mask Register Bits

Figure 4. Register and Function Addressing.

Channel	Register	Operation	Signals							Internal Flip/Flop	Data Bus DB0—DB7
			$\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A3	A2	A1	A0		
0	Base & Current Address	Write	0	1	0	0	0	0	0	0	A0-A7
			0	1	0	0	0	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	0	0	0	A0-A7
			0	0	1	0	0	0	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	0	0	1	0	W0-W7
			0	1	0	0	0	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	0	1	0	W0-W7
			0	0	1	0	0	0	1	1	W8-W15
1	Base & Current Address	Write	0	1	0	0	0	1	0	0	A0-A7
			0	1	0	0	0	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	1	0	0	A0-A7
			0	0	1	0	0	1	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	0	1	1	0	W0-W7
			0	1	0	0	0	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	1	1	0	W0-W7
			0	0	1	0	0	1	1	1	W8-W15
2	Base & Current Address	Write	0	1	0	0	1	0	0	0	A0-A7
			0	1	0	0	1	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	0	0	0	A0-A7
			0	0	1	0	1	0	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	1	0	1	0	W0-W7
			0	1	0	0	1	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	0	1	0	W0-W7
			0	0	1	0	1	0	1	1	W8-W15
3	Base & Current Address	Write	0	1	0	0	1	1	0	0	A0-A7
			0	1	0	0	1	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	1	0	0	A0-A7
			0	0	1	0	1	1	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	1	1	1	0	W0-W7
			0	1	0	0	1	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	1	1	0	W0-W7
			0	0	1	0	1	1	1	1	W8-W15

Figure 5. Word Count and Address Register Command Codes.

MAXIMUM RATINGS above which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.5W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	T_A	VCC
Am9517ADC/PC	0°C to +70°C	5.0V ±5%
Am9517A-1DC/PC	0°C to +70°C	5.0V ±5%
Am9517A-4DC/PC	0°C to +70°C	5.0V ±5%
Am9517ADM	−55°C to +125°C	5.0V ±10%

ELECTRICAL CHARACTERISTICS over operating range (Note 1)

Parameter	Description	Test Conditions	Min	Typ	Max	Unit
VOH	Output HIGH Voltage	IOH = −200μA	2.4			Volts
		IOH = −100μA, (HREQ Only)	3.3			
VOL	Output LOW Voltage	IOL = 3.2mA			0.4	Volts
VIH	Input HIGH Voltage		2.0		VCC+0.5	Volts
VIL	Input LOW Voltage		−0.5		0.8	Volts
IIX	Input Load Current	VSS ≤ VI ≤ VCC	−10		+10	μA
IOZ	Output Leakage Current	VCC ≤ VO ≤ VSS+.40	−10		+10	μA
ICC	VCC Supply Current	T_A = +25°C		65	130	mA
		T_A = 0°C		75	150	
		T_A = −55°C			175	
CO	Output Capacitance	fc = 1.0MHz, Inputs = 0V		4	8	pF
CI	Input Capacitance			8	15	pF
CIO	I/O Capacitance			10	18	pF

NOTES:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. Input timing parameters assume transition times of 20ns or less. Waveform measurement points for both input and output signals are 2.0V for High and 0.8V for Low, unless otherwise noted.
3. Output loading is 1 Standard TTL gate plus 50pF capacitance unless noted otherwise.
4. The new $\overline{IOW}$ or $\overline{MEMW}$ pulse width for normal write will be TCY-100ns and for extended write will be 2TCY-100ns. The net $\overline{IOR}$ or $\overline{MEMR}$ pulse width for normal read will be 2TCY-50ns and for compressed read will be TCY-50ns.
5. TDQ is specified for two different output HIGH levels. TDQ1 is measured at 2.0V. TDQ2 is measured at 3.3V. The value for TDQ2 assumes an external 3.3kΩ pull-up resistor connected from HREQ to VCC.
6. DREQ should be held active until DACK is returned.
7. DREQ and DACK signals may be active high or active low. Timing diagrams assume the active high mode.
8. Output loading on the data bus is 1 Standard TTL gate plus 15pF for the minimum value and 1 Standard TTL gate plus 100pF for the maximum value.
9. Successive read and/or write operations by the external processor to program or examine the controller must be timed to allow at least 600ns for the Am9517A or Am9517A-1 and at least 450ns for the Am9517A-4 as recovery time between active read or write pulses.
10. Parameters are listed in alphabetical order.
11. Pin 5 is an input that should always be at a logic high level. An internal pull-up resistor will establish a logic high when the pin is left floating. Alternatively, pin 5 may be tied to VCC.
12. Signals $\overline{READ}$ and $\overline{WRITE}$ refer to $\overline{IOR}$ and $\overline{MEMW}$ respectively for peripheral-to-memory DMA operations and to $\overline{MEMR}$ and $\overline{IOW}$ respectively for memory-to-peripheral DMA operations.
13. If N wait states are added during the write-to-memory half of a memory-to-memory transfer, this parameter will increase by N (TCY).

SWITCHING CHARACTERISTICS

ACTIVE CYCLE (Notes 2, 3, 10, 11 and 12)

		Am9517A		Am9517A-1		Am9517A-4		
Parameter	Description	Min	Max	Min	Max	Min	Max	Unit
TAEL	AEN HIGH from CLK LOW (S1) Delay Time		300		300		225	ns
TAET	AEN LOW from CLK HIGH (S1) Delay Time		200		200		150	ns
TAFAB	ADR Active to Float Delay from CLK HIGH		150		150		120	ns
TAFC	$\overline{\text{READ}}$ or $\overline{\text{WRITE}}$ Float from CLK HIGH		150		150		120	ns
TAFDB	DB Active to Float Delay from CLK HIGH		250		250		190	ns
TAHR	ADR from $\overline{\text{READ}}$ HIGH Hold Time	TCY-100		TCY-100		TCY-100		ns
TAHS	DB from ADSTB LOW Hold Time	50		50		40		ns
TAHW	ADR from $\overline{\text{WRITE}}$ HIGH Hold Time	TCY-50		TCY-50		TCY-50		ns
TAK	DACK Valid from CLK LOW Delay Time		280		280		220	ns
	$\overline{\text{EOP}}$ HIGH from CLK HIGH Delay Time		250		250		190	ns
	$\overline{\text{EOP}}$ LOW to CLK HIGH Delay Time		250		250		190	ns
TASM	ADR Stable from CLK HIGH		250		250		190	ns
TASS	DB to ADSTB LOW Setup Time	100		100		100		ns
TCH	Clock High Time (Transitions ≤ 10ns)	120		120		100		ns
TCL	Clock Low Time (Transitions ≤ 10ns)	150		150		110		ns
TCY	CLK Cycle Time	320		320		250		ns
TDCL	CLK HIGH to $\overline{\text{READ}}$ or $\overline{\text{WRITE}}$ LOW Delay (Note 4)		270		270		200	ns
TDCTR	$\overline{\text{READ}}$ HIGH from CLK HIGH (S4) Delay Time (Note 4)		270		270		210	ns
TDCTW	$\overline{\text{WRITE}}$ HIGH from CLK HIGH (S4) Delay Time (Note 4)		200		200		150	ns
TDQ1	HREQ Valid from CLK HIGH Delay Time (Note 5)		160		160		120	ns
TDQ2			250		250		190	ns
TEPS	$\overline{\text{EOP}}$ LOW from CLK LOW Setup Time	60		60		45		ns
TEPW	$\overline{\text{EOP}}$ Pulse Width	300		300		225		ns
TFAAB	ADR Float to Active Delay from CLK HIGH		250		250		190	ns
TFAC	$\overline{\text{READ}}$ or $\overline{\text{WRITE}}$ Active from CLK HIGH		200		200		150	ns
TFADB	DB Float to Active Delay from CLK HIGH		300		300		225	ns
THS	HACK valid to CLK HIGH Setup Time	100		100		75		ns
TIDH	Input Data from $\overline{\text{MEMR}}$ HIGH Hold Time	0		0		0		ns
TIDS	Input Data to $\overline{\text{MEMR}}$ HIGH Setup Time	250		250		190		ns
TODH	Output Data from $\overline{\text{MEMW}}$ HIGH Hold Time	20		20		20		ns
TODV	Output Data Valid to $\overline{\text{MEMW}}$ HIGH (Note 13)	200		200		125		ns
TQS	DREQ to CLK LOW (S1, S4) Setup Time	120		120		90		ns
TRH	CLK to READY LOW Hold Time	20		20		20		ns
TRS	READY to CLK LOW Setup Time	100		100		60		ns
TSTL	ADSTB HIGH from CLK HIGH Delay Time		200		200		150	ns
TSTT	ADSTB LOW from CLK HIGH Delay Time		140		140		110	ns

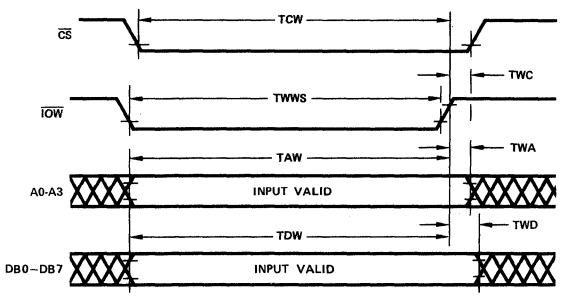
SWITCHING CHARACTERISTICS

PROGRAM CONDITION (IDLE CYCLE)

(Notes 2, 3, 10, 11 and 12)

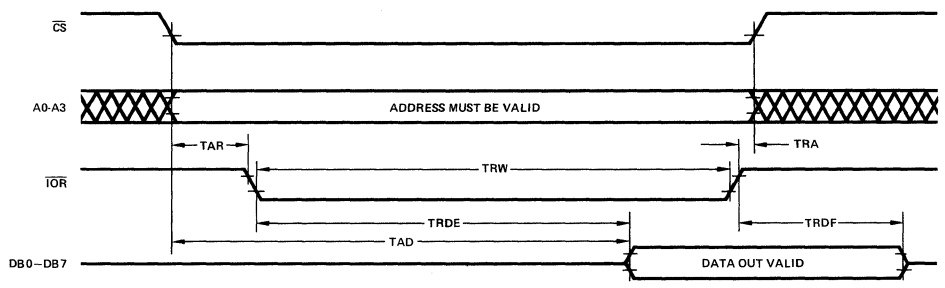
Parameter	Description	Am9517A Min.	Am9517A Max.	Am9517A-1 Min.	Am9517A-1 Max.	Am9517A-4 Min.	Am9517A-4 Max.	Unit
TAR	ADR Valid or $\overline{CS}$ LOW to $\overline{READ}$ LOW	50		50		50		ns
TAW	ADR Valid to $\overline{WRITE}$ HIGH Setup Time	200		200		150		ns
TCW	$\overline{CS}$ LOW to $\overline{WRITE}$ HIGH Setup Time	200		200		150		ns
TDW	Data Valid to $\overline{WRITE}$ HIGH Setup Time	200		200		150		ns
TRA	ADR or $\overline{CS}$ Hold from $\overline{READ}$ HIGH	0		0		0		ns
TRDE	Data Access from $\overline{READ}$ LOW (Note 8)		300		200		200	ns
TDRF	DB Float Delay from $\overline{READ}$ HIGH	20	150	20	100	20	100	ns
TRSTD	Power Supply HIGH to RESET LOW Setup Time	500		500		500		μs
TRSTS	RESET to First $\overline{IOWR}$	2		2		2		TCY
TRSTW	RESET Pulse Width	300		300		300		ns
TRW	$\overline{READ}$ Width	300		300		250		ns
TWA	ADR from $\overline{WRITE}$ HIGH Hold Time	20		20		20		ns
TWC	$\overline{CS}$ HIGH from $\overline{WRITE}$ HIGH Hold Time	20		20		20		ns
TWD	Data from $\overline{WRITE}$ HIGH Hold Time	30		30		30		ns
TWWS	Write Width	200		200		200		ns
TAD	Data Access from ADR Valid, $\overline{CS}$ LOW		350		300		300	ns

SWITCHING WAVEFORMS



Timing Diagram 1. Program Condition Write Timing (Note 9).

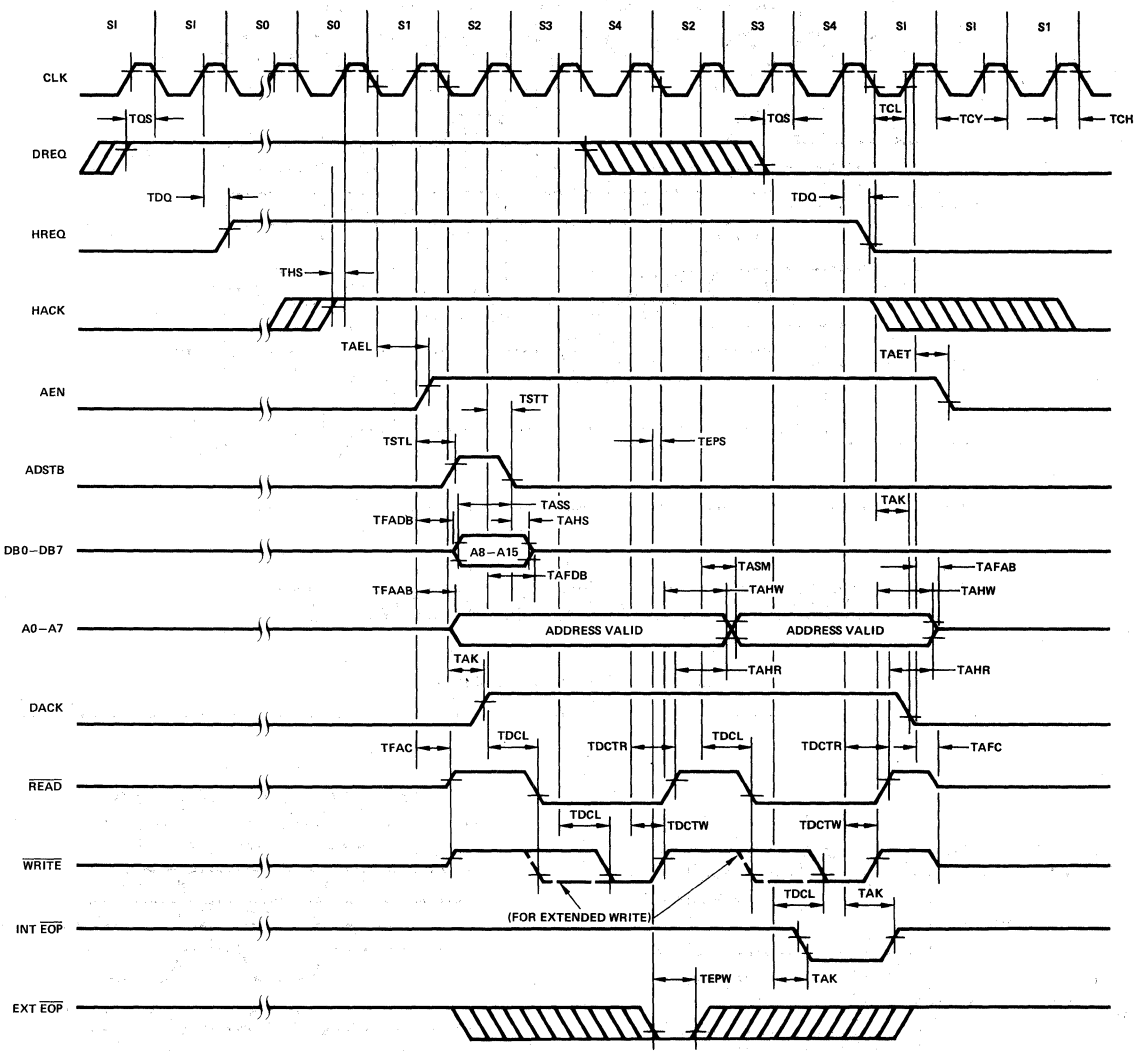
MOS-036



Timing Diagram 2. Program Condition Read Cycle (Note 9).

MOS-037

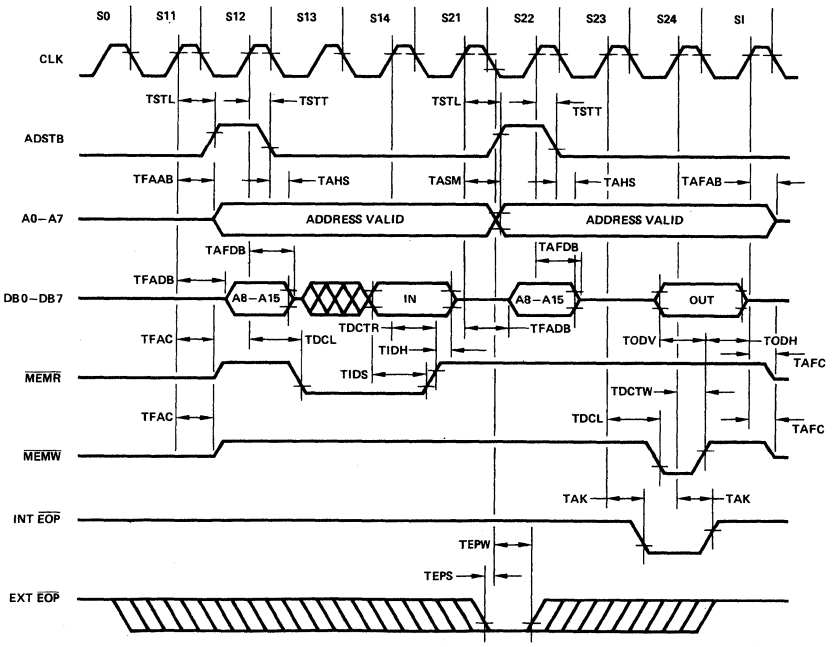
SWITCHING WAVEFORMS (Cont.)



Timing Diagram 3. Active Cycle Timing Diagram.

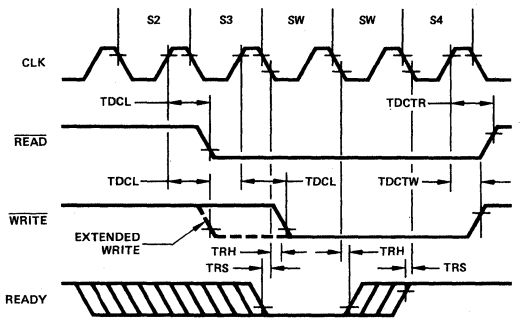
MOS-038

SWITCHING WAVEFORMS (Cont.)



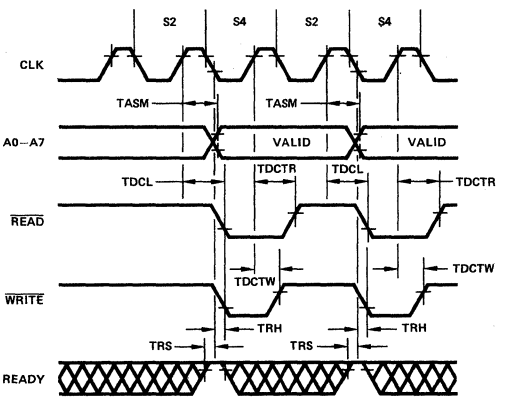
Timing Diagram 4. Memory-to-Memory.

MOS-039



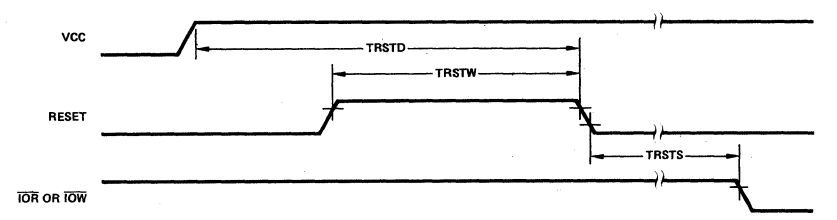
Timing Diagram 5. Ready Timing.

MOS-040



Timing Diagram 6. Compressed Timing.

MOS-041



Timing Diagram 7. Reset Timing.

MOS-042

10

APPLICATION INFORMATION

Figure 6 shows a convenient method for configuring a DMA system with the Am9517A Controller and a microprocessor system. The Multimode DMA Controller issues a Hold Request to the processor whenever there is at least one valid DMA Request from a peripheral device. When the processor replies with a Hold Acknowledge signal, the Am9517A takes control of the Address Bus, the Data Bus and the Control Bus. The address for the first transfer operation comes out in two bytes – the least significant eight bits on the eight Address outputs and the most significant eight bits on the Data Bus. The contents of the Data Bus are then latched into the Am74LS373 register to complete the full 16 bits of the Address Bus. The Am74LS373 is a high speed, low power, 8-bit, 3-state register in a 20-pin package. After the initial transfer takes place, the register is updated only after a carry or borrow is generated in the least significant address byte. Four DMA channels are provided when one Am9517A is used.

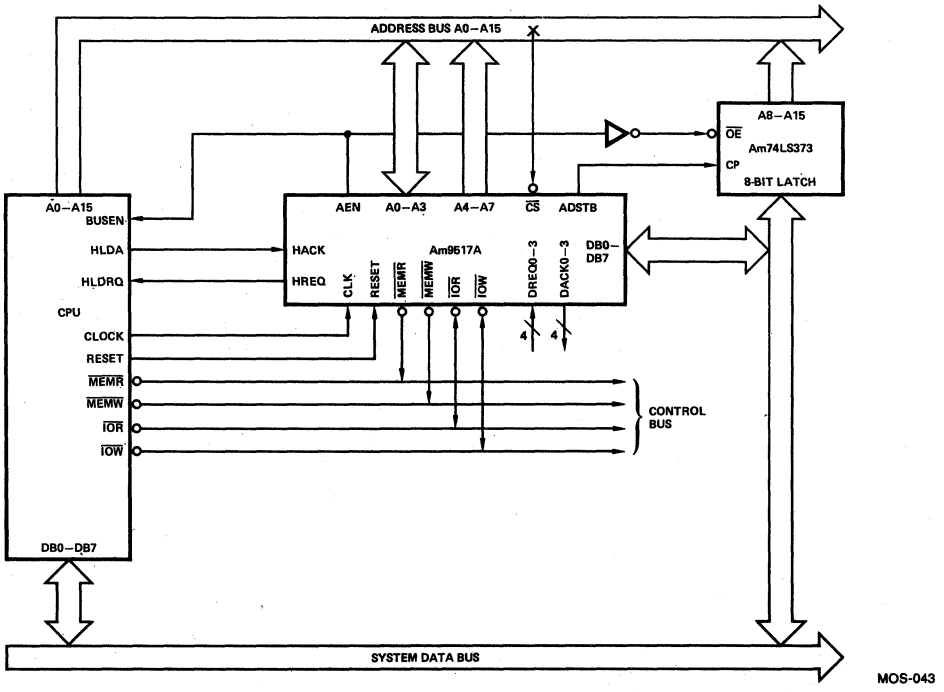
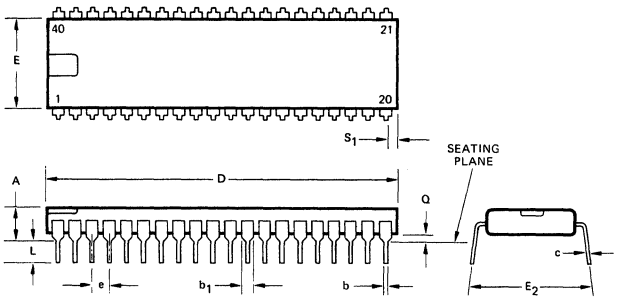


Figure 6. Basic DMA Configuration.

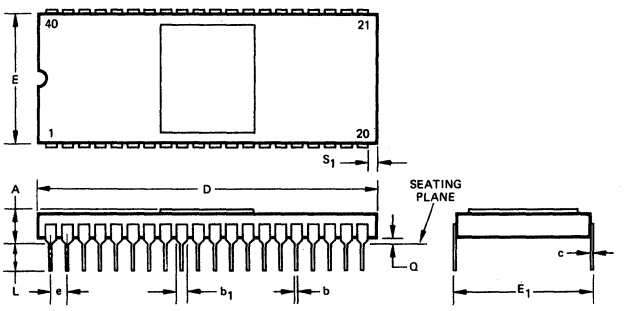
PHYSICAL DIMENSIONS
Dual-In-Line

40-Pin Plastic



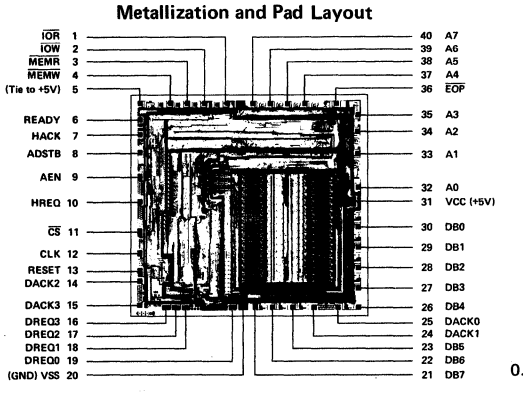
Reference Symbol	Inches	
	Min.	Max.
A	.150	.200
b	.015	.020
b_1	.055	.065
c	.009	.011
D	2.050	2.080
E	.530	.550
E_1	.585	.700
e	.090	.110
L	.125	.160
Q	.015	.060
S_1	.040	.070

40-Pin Hermetic



Reference Symbol	Inches	
	Min.	Max.
A	.100	.200
b	.015	.022
b_1	.030	.060
c	.008	.013
D	1.960	2.040
E	.550	.610
E_1	.590	.620
e	.090	.110
L	.120	.160
Q	.020	.060
S_1	.005	

Metallization and Pad Layout



DIE SIZE
0.198" X 0.210"

Am9519

Universal Interrupt Controller

DISTINCTIVE CHARACTERISTICS

- Eight individually maskable interrupt inputs
- Software interrupt request capability
- Fully programmable 1, 2, 3 or 4 byte responses
- Unlimited daisy-chain expansion capability
- Fixed or rotating priority resolution
- Common vector option
- Polled mode option
- Optional automatic clearing of acknowledged interrupts
- Bit set/reset capability for Mask register
- Master Mask bit disables all interrupts
- Pulse-catching interrupt input circuitry
- Polarity control of interrupt inputs and output
- Various timing options including 8085A compatible Am9519-1
- Single +5V supply
- 100% MIL-STD-883 reliability assurance testing

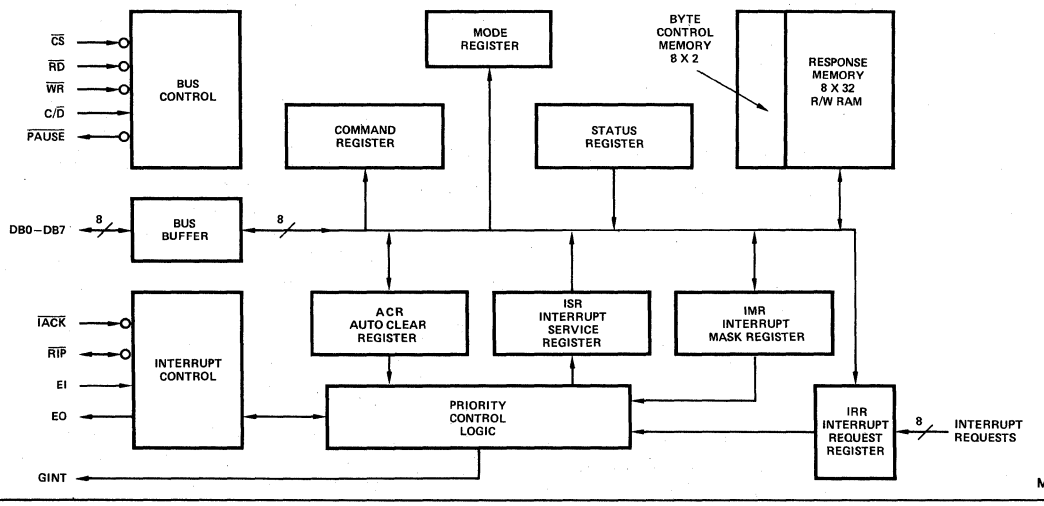
GENERAL DESCRIPTION

The Am9519 Universal Interrupt Controller is a processor support circuit that provides a powerful interrupt structure to increase the efficiency and versatility of microcomputer-based systems. A single Am9519 manages up to eight maskable interrupt request inputs, resolves priorities and supplies up to four bytes of fully programmable response for each interrupt. It uses a simple expansion structure that allows many units to be cascaded for control of large numbers of interrupts. Several programmable control features are provided to enhance system flexibility and optimization.

The Universal Interrupt Controller is designed with a general purpose interface to facilitate its use with a wide range of digital systems, including most popular 8-bit microprocessors. Since the response bytes are fully programmable, any instruction or vectoring protocol appropriate for the host processor may be used.

When the Am9519 controller receives an unmasked Interrupt Request, it issues a Group Interrupt output to the CPU. When the interrupt is acknowledged, the controller outputs the one-to-four byte response associated with the highest priority unmasked interrupt request. The ability of the CPU to set interrupt requests under software control permits hardware prioritization of software tasks and aids system diagnostic and maintenance procedures.

BLOCK DIAGRAM



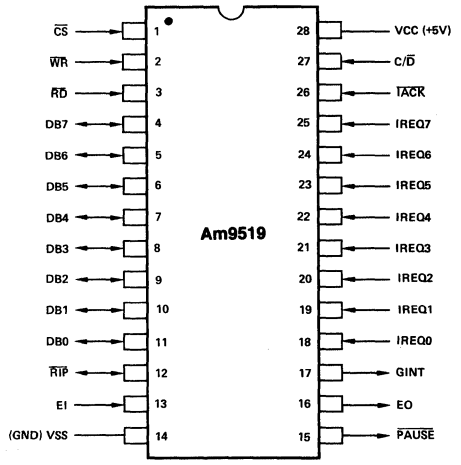
MOS-143

ORDERING INFORMATION

Package Type	Ambient Temperature	Timing Options	
		Am9519	Am9519-1
Hermetic DIP*	$0°C \leq T_A \leq +70°C$	AM9519DC/CC	AM9519-1DC/CC
	$-55°C \leq T_A \leq +125°C$	AM9519DM	
Molded DIP	$0°C \leq T_A \leq +70°C$	AM9519PC	AM9519-1PC

*DC = Side-Brazed Ceramic CC = Cerdip

CONNECTION DIAGRAM



Top View

Pin 1 is marked for orientation.

MOS-019

INTERFACE SIGNAL DESCRIPTION

VCC: +5 Volt Power Supply
VSS: Ground

DB0 – DB7 (Data Bus, Input/Output)

The eight bidirectional data bus signals are used to transfer information between the Am9519 and the system data bus. The direction of transfer is controlled by the $\overline{IACK}$, $\overline{WR}$ and $\overline{RD}$ input signals. Programming and control information are written into the device; status and response data are output by it.

$\overline{CS}$ (Chip Select, Input)

The active low Chip Select input enables read and write operations on the data bus. Interrupt acknowledge responses are not conditioned by $\overline{CS}$.

$\overline{RD}$ (Read, Input)

The active low Read signal is conditioned by $\overline{CS}$ and indicates that information is to be transferred from the Am9519 to the data bus.

$\overline{WR}$ (Write, Input)

The active low Write signal is conditioned by $\overline{CS}$ and indicates that data bus information is to be transferred from the data bus to a location within the Am9519.

C/$\overline{D}$ (Control/Data, Input)

The C/$\overline{D}$ control signal selects source and destination locations for data bus read and write operations. Data read or write transfers are made to or from preselected internal registers or memory locations. Control write operations load the command register and control read operations output the status register.

IREQ0 – IREQ7 (Interrupt Request, Input)

The Interrupt Request signals are used by external devices to indicate that service by the host CPU is desired. IREQ inputs are accepted asynchronously and they may be programmed for either a high-to-low or low-to-high edge transition. Active inputs are latched internally in the Interrupt Request Register. After the IRR bit is cleared, an IREQ transition of the programmed polarity must occur to initiate another request.

$\overline{RIP}$ (Response In Process, Input/Output)

Response In Process is a bidirectional signal used when two or more Am9519 circuits are cascaded. It permits multibyte response transfers to be completed without interference from higher priority interrupts. An Am9519 that is responding to an acknowledged interrupt will treat $\overline{RIP}$ as an output and hold it low until the acknowledge response is finished. An Am9519 without an acknowledged interrupt will treat $\overline{RIP}$ as an input and will ignore $\overline{IACK}$ pulses as long as $\overline{RIP}$ is low. The $\overline{RIP}$ output is open drain and requires an external pullup resistor to VCC.

$\overline{IACK}$ (Interrupt Acknowledge, Input)

The active low Interrupt Acknowledge line indicates that the external system is asking for interrupt response information. Depending on the programmed state of the Am9519, it will accept 1, 2, 3 or 4 $\overline{IACK}$ pulses; one response byte is transferred per pulse. The first $\overline{IACK}$ pulse causes selection of the highest priority unmasked pending interrupt request and generates a $\overline{RIP}$ output signal.

$\overline{PAUSE}$ (Pause, Output)

The active-low Pause signal is used to coordinate interrupt responses with data bus and control timing. Pause goes low when the first $\overline{IACK}$ is received and remains low until $\overline{RIP}$ goes low. The external system can use Pause to stretch the acknowledge cycle and allow the control timing to automatically adjust to the actual priority resolution delays in the interrupt system. Second, third and fourth response bytes do not cause Pause to go low. Pause is an open drain output and requires an external pullup resistor to VCC.

EO (Enable Out, Output)

The active high EO signal is used to implement daisy-chained cascading of several Am9519 circuits. EO is connected to the EI input of the next lower priority chip. On receipt of an interrupt acknowledge, each EO will go inactive until it has been determined that no valid interrupt request is pending on that chip. If an active request is present, EO remains low. EO is also held low when the master mask bit is active, thus disabling all lower priority chips.

EI (Enable In, Input)

The active high EI signal is used to implement daisy-chained cascading of several Am9519 circuits. EI is connected to EO of the next higher priority chip. It may also be used as a hardware disable input for the interrupt system. When EI is low $\overline{IACK}$ inputs are ignored. EI is internally pulled up to VCC so that no external pullup is needed when EI is not used.

GINT (Group Interrupt, Output)

The Group Interrupt output signal indicates that at least one unmasked interrupt request is pending. It may be programmed for active high or active low polarity. When active low, the output is open drain and requires an external pull up resistor to VCC.

REGISTER DESCRIPTION

Interrupt Request Register (IRR): The 8-bit IRR is used to store pending interrupt requests. A bit in the IRR is set whenever the corresponding IREQ input goes active. Bits may also be set under program control from the CPU, thus permitting software generated interrupts. IRR bits may be cleared under program control. An IRR bit is automatically cleared when its interrupt is acknowledged. All IRR bits are cleared by a reset function.

Interrupt Service Register (ISR): The 8-bit ISR contains one bit for each IREQ input. It is used to indicate that a pending interrupt has been acknowledged and to mask all lower priority interrupts. When a bit is set by the acknowledge logic in the ISR, the corresponding IRR bit is cleared. If an acknowledged interrupt is not programmed to be automatically cleared, its ISR bit must be cleared by the CPU under program control when it is desired to permit interrupts from lower priority devices. When the interrupt is programmed for automatic clearing, the ISR bit is automatically reset during the acknowledge sequence. All ISR bits are cleared by a reset function.

Interrupt Mask Register (IMR): The 8-bit IMR is used to enable or disable the individual interrupt inputs. The IMR bits correspond to the IREQ inputs and all eight may be loaded, set or cleared in parallel under program control. In addition, individual IMR bits may be set or cleared by the CPU. A reset function will set all eight mask bits, disabling all requests. A mask bit that is set does not disable the IRR, and an IREQ that arrives while a corresponding mask bit is set will cause an interrupt later when the mask bit is cleared. Only unmasked interrupt inputs can generate a Group Interrupt output.

Response Memory: An 8 x 32 read/write response memory is included in the Am9519. It is used to store up to four bytes of response information for each of the eight interrupt request inputs. All bits in the memory are programmable, allowing any desired vector, opcode, instruction or other data to be entered. The Am9519 transfers the interrupt response information for the highest priority unmasked interrupt from the memory to the data bus when the $\overline{IACK}$ input is active.

Auto Clear Register: The 8-bit Auto Clear register contains one bit for each IREQ input and specifies the operating mode for each of the ISR bits. When an auto clear bit is off, the corresponding ISR bit is set when that interrupt is acknowledged and is cleared by software command. When an auto clear bit is on, the corresponding ISR bit is cleared by the hardware before the end of the acknowledge sequence. A reset function clears all auto clear bits.

Status Register: The 8-bit Status register contains information concerning the internal state of the chip. It is especially useful when operating in the polled mode in order to identify interrupting devices. Figure 1 shows the status register bit assignments. The polarity of the GINT bit 7 is not affected by the GINT polarity control (Mode bit 3). The Status register is read by executing a read operation ($\overline{CS} = 0$, $\overline{RD} = 0$) with the control location selected ($C/\overline{D} = 1$).

Mode Register: The 8-bit Mode register controls the operating options of the Am9519. Figure 2 shows the bit assignments for the Mode register. The five low order mode bits (0 through 4) are loaded in parallel by command. Bits 5, 6 and 7 are controlled by separate commands. (See Figure 4.) The Mode register cannot be read out directly to the data bus, but Mode bits 0, 2 and 7 are available as part of the Status register.

Command Register: The 8-bit Command register stores the last command entered. Depending upon the command opcode, it may initiate internal actions or precondition the part for subsequent data bus transfers. The Command register is loaded by executing a write operation ($\overline{WR} = 0$) with the control location selected ($C/\overline{D} = 1$), as shown in Figure 3.

Byte Count Register: The length in bytes of the response associated with each interrupt is independently programmed so that different interrupts may have different length responses. The byte count for each response is stored in eight 2-bit Byte Count registers. For a given interrupt the Am9519 will expect to receive a number of $\overline{IACK}$ pulses that equals the corresponding byte count, and will hold $\overline{RIP}$ low until the count is satisfied.

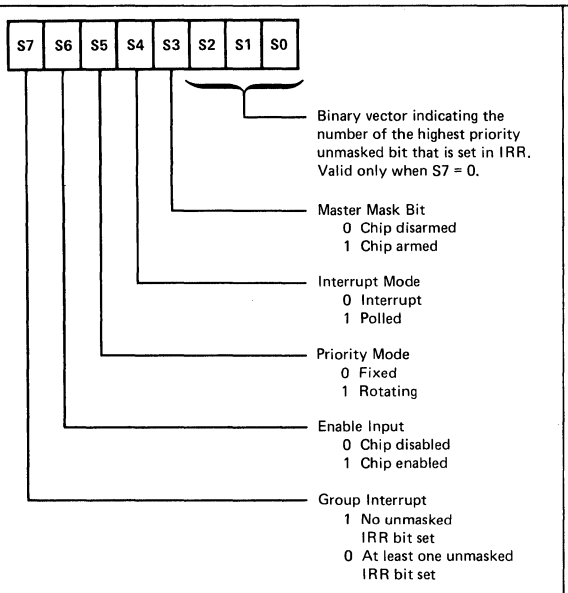


Figure 1. Status Register Bit Assignments. MOS-025

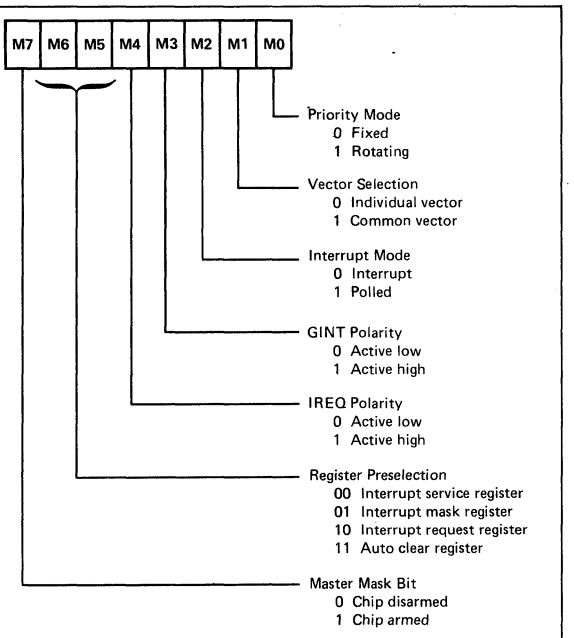


Figure 2. Mode Register Bit Assignments. MOS-026

FUNCTIONAL DESCRIPTION

Interrupts are used to improve system throughput and response time by eliminating heavy dependence on software polling procedures. Interrupts allow external devices to asynchronously modify the instruction sequence of a program being executed. In systems with multiple interrupts, vectoring can further improve performance by allowing direct identification of the interrupting device and its associated service routine. The Am9519 Universal Interrupt Controller contains, on one chip, all of the circuitry necessary to detect, prioritize and manage eight vectored interrupts. It includes many options and operating modes that permit the design of sophisticated interrupt systems.

Reset

The reset function is accomplished by software command or automatically during power-up. The reset command may be issued by the CPU at any time. Internal power up circuitry is triggered when VCC reaches a predetermined threshold, causing a brief internal reset pulse. In both cases, the resulting internal state of the machine is that all registers are cleared except the Mask register which is set. Thus no Group Interrupt will be generated and no interrupt requests will be recognized. The response memory and Byte Count registers are not affected by reset. Their contents after power-up are unpredictable and must be established by the host CPU during initialization.

Operating Sequence

A brief description of a typical sequence of events in an operating interrupt system will illustrate the general interactions among the host CPU, the interrupt controller and the interrupting peripheral.

1. The Am9519 controller is initialized by the CPU in order to customize its configuration and operation for the application at hand. Both the controller and the CPU are then enabled to accept interrupts.
2. One (or more) of the interrupt request inputs to the controller becomes active indicating that peripheral equipment is asking for service. The controller asynchronously accepts and latches the request(s).
3. If the request is masked, no further action takes place. If the request is not masked, a Group Interrupt output is generated by the controller.
4. The GINT signal is recognized by the CPU which normally will complete the execution of the current instruction, insert an interrupt acknowledge sequence into its instruction execution stream, and disable its internal interrupt structure. The controller expects to receive one or more $\overline{IACK}$ signals from the CPU during the acknowledge sequence.
5. When the controller receives the $\overline{IACK}$ signal, it brings $\overline{PAUSE}$ low and selects the highest priority unmasked pending request. When selection is complete, the $\overline{RIP}$ output is brought low and the first byte in the response memory associated with the selected request is output on the data bus. $\overline{PAUSE}$ stays low until RIP goes low. $\overline{RIP}$ stays low until the last byte of the response has been transferred.
6. During the acknowledge sequence, the IRR bit corresponding to the selected request is automatically cleared, and the corresponding ISR bit is set. When the ISR bit is set, the Group Interrupt output is disabled until a higher priority request arrives or the ISR bit is cleared. The ISR bit will be cleared by either hardware or software.
7. If a higher priority request arrives while the current request is being serviced, GINT will be output by the controller, but will be recognized and acknowledged only if the CPU has its interrupt input enabled. If acknowledged, the corresponding higher priority ISR bit will be set and the requests nested.

Information Transfers

Figure 3 shows the control signal configurations for all information transfer operations between the Am9519 and the data bus. The following conventions are assumed: $\overline{RD}$ and $\overline{WR}$ active are mutually exclusive; $\overline{RD}$, $\overline{WR}$ and C/$\overline{D}$ have no meaning unless $\overline{CS}$ is low; active $\overline{IACK}$ pulses occur only when $\overline{CS}$ is high.

For reading, the Status register is selected directly by the C/$\overline{D}$ control input. Other internal registers are read by preselecting the desired register with mode bits 5 and 6, and then executing a data read. The response memory can be read only with $\overline{IACK}$ pulses. For writing, the Command register is selected directly by the C/$\overline{D}$ control input. The Mask and Auto Clear registers are loaded following specific commands to that effect. To load each level of the response memory, the response preselect command is issued to select the desired level. An appropriate number of data write operations are then executed to load that level.

CONTROL INPUT					DATA BUS OPERATION
$\overline{CS}$	C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	$\overline{IACK}$	
0	0	0	1	1	Transfer contents of preselected data register to data bus
0	0	1	0	1	Transfer contents of data bus to preselected data register
0	1	0	1	1	Transfer contents of status register to data bus
0	1	1	0	1	Transfer contents of data bus to command register
1	X	X	X	0	Transfer contents of selected response memory location to data bus
1	X	X	X	1	No information transferred

Figure 3. Summary of Data Bus Transfers.

The Pause output may be used by the host CPU to ensure that proper timing relationships are maintained with the Am9519 when $\overline{IACK}$ is active. The $\overline{IACK}$ pulse width required depends on several variables, including: operating temperature, internal logic delays, number of interrupt controllers chained together, and the priority level of the interrupt being acknowledged. When delays in these variables combine to delay selection of a request following the falling edge of the first $\overline{IACK}$, the Pause output may be used to extend the $\overline{IACK}$ pulse, if necessary. Pause will remain low until a request has been selected, as indicated by the falling edge of $\overline{RIP}$. Typically, the internal interrupt selection process is quite fast, especially for systems with a single Am9519, and Pause will consequently remain low for only a very brief interval and will not cause extension of the $\overline{IACK}$ timing.

Operating Options

The Mode register specifies the various combinations of operating options that may be selected by the CPU. It is cleared by power-up or by a reset command.

Mode bit 0 specifies the rotating/fixed priority mode (see Figure 2). In the fixed mode, priority is assigned to the request inputs based upon their physical location at the chip interface, with IREQ0 the highest and IREQ7 the lowest. In the rotating mode, relative priority is the same as for the fixed mode and the most recently serviced request is assigned the lowest priority. In the fixed mode, a lower priority request might never receive service if enough higher priority requests are active. In the rotating mode, any request will receive service within a maximum of seven other service cycles no matter what pattern the request inputs follow.

Mode bit 1 selects the individual/common vector option. Individual vectoring provides a unique location in the response memory for each interrupt request. The common vector option always supplies the response associated with IREQ0 no matter which request is being acknowledged.

Mode bit 2 specifies interrupt or polled operation. In the polled mode the Group Interrupt output is disabled. The CPU may read the Status register to determine if a request is pending. Since $\overline{IACK}$ pulses are not normally supplied in polled mode, the IRR bit is not automatically cleared, but may be cleared by command. With no $\overline{IACK}$ input the ISR and the response memory are not used. An Am9519 in the polled mode has EI connected to EO so that in multichip interrupt systems the polled chip is functionally removed from the priority hierarchy.

Mode bit 3 specifies the sense of the GINT output. When active high polarity is selected the output is a two-state configuration. For active low polarity, the output is open drain and requires an external pull-up resistor to provide the high logic level. The open drain output allows wired-or configurations with other similar output signals.

Mode bit 4 specifies the sense of the IREQ inputs. When active low polarity is selected, the IRR responds to falling edges on the request inputs. When active high is selected, the IRR responds to rising edges.

Mode bits 5 and 6 specify the register that will be read on subsequent data read operations (C/$\overline{D}$ = 0, $\overline{RD}$ = 0). This preselection remains valid until changed by a reset or a command.

Mode bit 7 is the master mask bit that disables all request inputs. It is used to disable all interrupts without modifying the IMR so that the previous IMR contents are valid when interrupts are re-enabled. When the master mask bit is low, it causes the EO line to remain disabled (low). Thus, for multiple-chip interrupt systems, one master mask bit can disable the whole interrupt structure. Alternatively, portions of the structure may be disabled. The state of the master mask bit is available as bit S3 of the Status register.

Programming

After reset, the Am9519 must be initialized by the CPU in order to perform useful work. At a minimum, the master mask bit and at least one of the IMR bits should be enabled. If vectoring is to be used, the response memory must be loaded; if not, the mode must be changed to a non-vectored configuration. Normally, the first step will be to modify the Mode register and the Auto clear register in order to establish the configuraton desired for the application. Then the response memory and byte count will be loaded for those request levels that will be in use. Finally, the master mask bit and at least portions of the IMR will be enabled to allow interrupt processing to proceed.

Commands

The host CPU configures, changes and inspects the internal condition of the Am9519 using the set of commands shown in Figure 4. An "X" entry in the table indicates a "don't care" state. All commands are entered by directly loading the Command register as shown in Figure 3 ($C/\overline{D}$ = 1, $\overline{WR}$ = 0). Figure 5 shows the coding assignments for the Byte Count registers. A detailed description of each command is contained in the Am9519 Application Note AMPUB071.

BY1	BY0	COUNT
0	0	1
0	1	2
1	0	3
1	1	4

Figure 5. Byte Count Coding.

COMMAND CODE								COMMAND DESCRIPTION
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	Reset
0	0	0	1	0	X	X	X	Clear all IRR and all IMR bits
0	0	0	1	1	B2	B1	B0	Clear IRR and IMR bit specified by B2, B1, B0
0	0	1	0	0	X	X	X	Clear all IMR bits
0	0	1	0	1	B2	B1	B0	Clear IMR bit specified by B2, B1, B0
0	0	1	1	0	X	X	X	Set all IMR bits
0	0	1	1	1	B2	B1	B0	Set IMR bit specified by B2, B1, B0
0	1	0	0	0	X	X	X	Clear all IRR bits
0	1	0	0	1	B2	B1	B0	Clear IRR bit specified by B2, B1, B0
0	1	0	1	0	X	X	X	Set all IRR bits
0	1	0	1	1	B2	B1	B0	Set IRR bit specified by B2, B1, B0
0	1	1	0	X	X	X	X	Clear highest priority ISR bit
0	1	1	1	0	X	X	X	Clear all ISR bits
0	1	1	1	1	B2	B1	B0	Clear ISR bit specified by B2, B1, B0
1	0	0	M4	M3	M2	M1	M0	Load Mode register bits 0–4 with specified pattern
1	0	1	0	M6	M5	0	0	Load Mode register bits 5, 6 with specified pattern
1	0	1	0	M6	M5	0	1	Load Mode register bits 5, 6 and set mode bit 7
1	0	1	0	M6	M5	1	0	Load Mode register bits 5, 6 and clear mode bit 7
1	0	1	1	X	X	X	X	Preselect IMR for subsequent loading from data bus
1	1	0	0	X	X	X	X	Preselect Auto Clear register for subsequent loading from data bus
1	1	1	BY1	BY0	L2	L1	L0	Load BY1, BY0 into byte count register and preselect response memory level specified by L2, L1, L0 for subsequent loading from data bus

Figure 4. Am9519 Command Summary.

MAXIMUM RATINGS above which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.5W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VCC	VSS
Am9519DC/CC Am9519-1DC	$0°C \leq T_A \leq +70°C$	+5.0V ±5%	0V
Am9519DM	$-55°C \leq T_A \leq +125°C$	+5.0V ±10%	0V

ELECTRICAL CHARACTERISTICS Over Operating Range (Note 1)

Parameter	Description	Test Conditions		Min.	Typ.	Max.	Unit
VOH	Output High Voltage (Note 12)	IOH = −200μA		2.4			Volts
		IOH = −100μA (EO only)		2.4			
VOL	Output Low Voltage	IOL = 3.2mA				0.4	Volts
		IOL = 1.0mA (EO only)				0.4	
VIH	Input High Voltage			2.0		VCC	Volts
VIL	Input Low Voltage			−0.5		0.8	Volts
IIX	Input Load Current	VSS ≤ VIN ≤ VCC	EI Input	−60		10	μA
			Other Inputs	−10		10	
IOZ	Output Leakage Current	VSS ≤ VOUT ≤ VCC, Output off		−10		10	μA
ICC	VCC Supply Current	T_A = +25°C			80	125	mA
		T_A = 0°C			100	145	
CO	Output Capacitance	fc = 1.0MHz				15	pF
CI	Input Capacitance	T_A = 25°C				10	
CIO	I/O Capacitance	All pins at 0V				20	

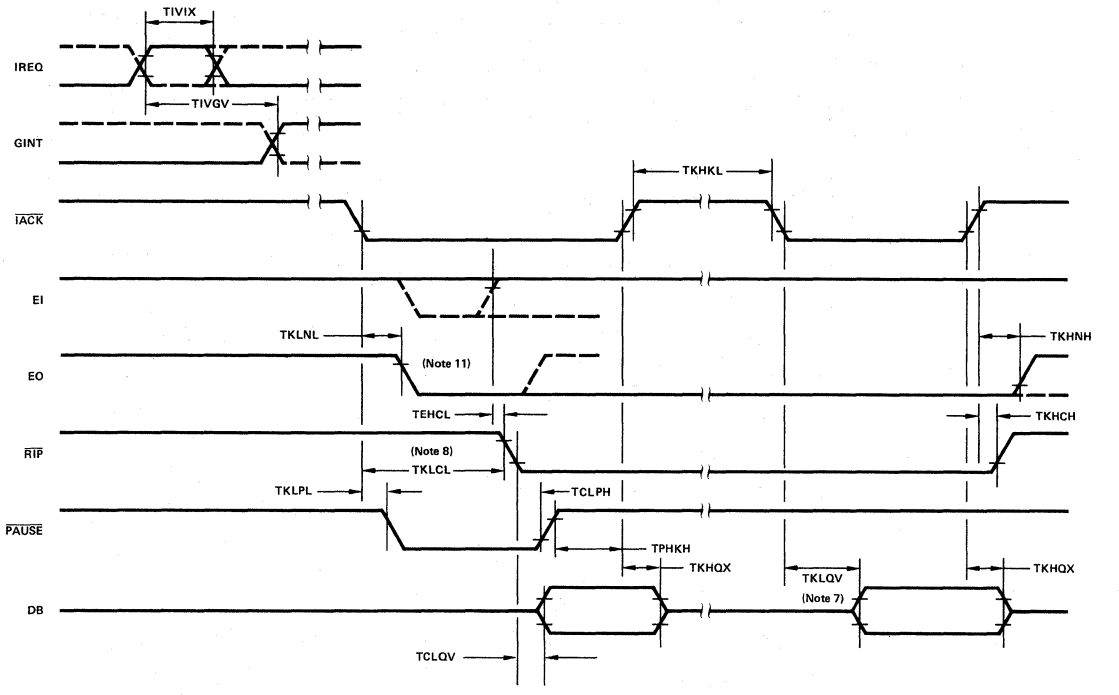
SWITCHING CHARACTERISTICS Over Operating Range (Notes 2, 3, 4, 5)

Parameters	Description	Am9519 Min.	Am9519 Max.	Am9519-1 Min.	Am9519-1 Max.	Units
TAVRL	C/$\overline{D}$ Valid and $\overline{CS}$ LOW to Read LOW	0		0		ns
TAVWL	C/$\overline{D}$ Valid and $\overline{CS}$ LOW to Write LOW	0		0		ns
TCLPH	$\overline{RIP}$ LOW to $\overline{PAUSE}$ HIGH (Note 6)	75	375	75	375	ns
TCLQV	$\overline{RIP}$ LOW to Data Out Valid (Note 7)		50		40	ns
TDVWH	Data In Valid to Write HIGH	250		200		ns
TEHCL	Enable in HIGH to $\overline{RIP}$ LOW (Notes 8, 9)	30	300	30	300	ns
TIVGV	Interrupt Request Valid to Group Interrupt Valid		800		650	ns
TIVIX	Interrupt Request Valid to Interrupt Request Don't Care (IREQ Pulse Duration)	250		250		ns
TKHCH	$\overline{IACK}$ HIGH to $\overline{RIP}$ HIGH (Note 8)		450		350	ns
TKHKL	$\overline{IACK}$ HIGH to $\overline{IACK}$ LOW ($\overline{IACK}$ Recovery)	500		500		ns
TKHNH	$\overline{IACK}$ HIGH to EO HIGH (Notes 10, 11)		975		750	ns
TKHQX	$\overline{IACK}$ HIGH to Data Out Invalid	20	200	20	100	ns
TKLCL	$\overline{IACK}$ LOW to $\overline{RIP}$ LOW (Note 8)	75	600	75	450	ns
TKLNL	$\overline{IACK}$ LOW to EO LOW (Notes 10, 11)		125		100	ns
TKLPL	$\overline{IACK}$ LOW to $\overline{PAUSE}$ LOW	25	175	25	125	ns
TKLQV	$\overline{IACK}$ LOW to Data Out Valid (Note 7)	25	300	25	200	ns
TPHKH	$\overline{PAUSE}$ HIGH to $\overline{IACK}$ HIGH	0		0		ns
TRHAX	Read HIGH to C/$\overline{D}$ and $\overline{CS}$ Don't Care	0		0		ns
TRHQX	Read HIGH to Data Out Invalid	20	200	20	100	ns
TRLQV	Read LOW to Data Out Valid		300		200	ns
TRLQX	Read LOW to Data Out Unknown	50		50		ns
TRLRH	Read LOW to Read HIGH ($\overline{RD}$ Pulse Duration)	300		250		ns
TWHAX	Write HIGH to C/$\overline{D}$ and $\overline{CS}$ Don't Care	0		0		ns
TWHDX	Write HIGH to Data In Don't Care	0		0		ns
TWHRW	Write HIGH to Read or Write LOW (Write Recovery)	600		400		ns
TWLWH	Write LOW to Write HIGH ($\overline{WR}$ Pulse Duration)	300		250		ns

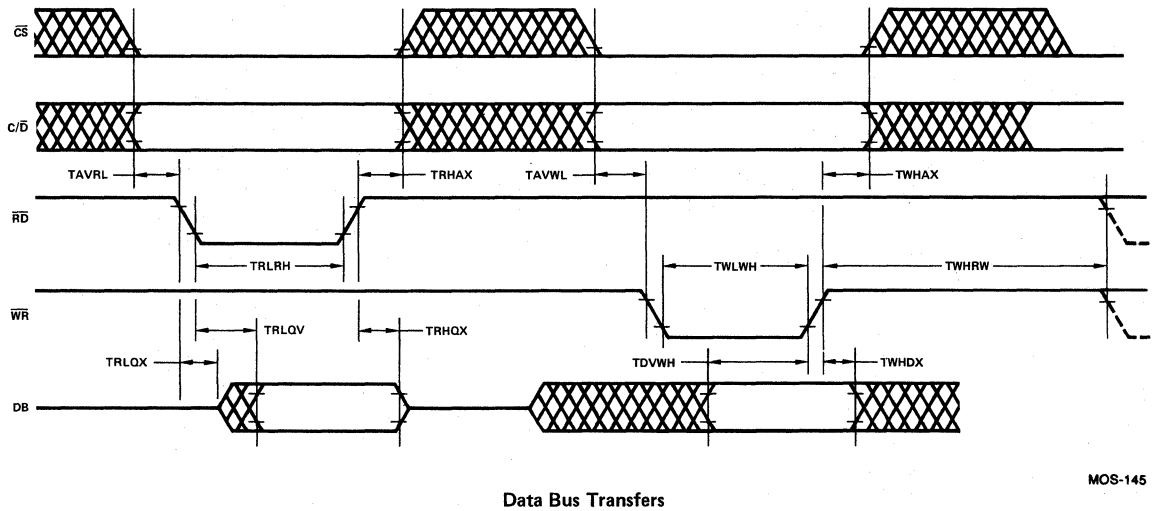
NOTES:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. Test conditions assume transition times of 20ns or less, timing reference levels of 0.8V and 2.0V and output loading of one TTL gate plus 100pF, unless otherwise noted.
3. Transition abbreviations used for the switching parameter symbols include: H = High, L = Low, V = Valid, X = unknown or don't care, Z = high impedance.
4. Signal abbreviations used for the switching parameter symbols include: R = Read, W = Write, Q = Data Out, D = Data In, A = Address ($\overline{CS}$ and C/$\overline{D}$), K = Interrupt Acknowledge, N = Enable Out, E = Enable In, P = Pause, C = $\overline{RIP}$.
5. Switching parameters are listed in alphabetical order.
6. During the first $\overline{IACK}$ pulse, $\overline{PAUSE}$ will be low long enough to allow for priority resolution and will not go high until after $\overline{RIP}$ goes low (TCLPH).
7. TKLQV applies only to second, third and fourth $\overline{IACK}$ pulses while $\overline{RIP}$ is low. During the first $\overline{IACK}$ pulse, Data Out will be valid following the falling edge of $\overline{RIP}$ (TCLQV).
8. $\overline{RIP}$ is pulled low to indicate that an interrupt request has been selected. $\overline{RIP}$ cannot be pulled low until EI is high following an internal delay. TKLCL will govern the falling edge of $\overline{RIP}$ when EI is always high or is high early in the acknowledge cycle. TEHCL will govern when EI goes high later in the cycle. The rising edge of EI will be determined by the length of the preceding priority resolution chain. $\overline{RIP}$ remains low until after the rising edge of the $\overline{IACK}$ pulse that transfers the last response byte for the selected IREQ.
9. Test conditions for the EI line assume timing reference levels of 0.8V and 2.0V with transition times of 10ns or less.
10. Test conditions for the EO line assume output loading of two LS TTL gates plus 30pF and timing reference levels of 0.8V and 2.0V. Since EO normally only drives EI of another Am9519, higher speed operation can be specified with this more realistic test condition.
11. The arrival of $\overline{IACK}$ will cause EO to go low, disabling additional circuits that may be connected to EO. If no valid interrupt is pending, EO will return high when EI is high. If a pending request is selected, EO will stay low until after the last $\overline{IACK}$ pulse for that interrupt is complete and $\overline{RIP}$ goes high.
12. VOH specifications do not apply to $\overline{RIP}$ or to $\overline{GINT}$ when active-low. These outputs are open-drain and VOH levels will be determined by external circuitry.

SWITCHING WAVEFORMS



Interrupt Operations



Data Bus Transfers

APPLICATIONS

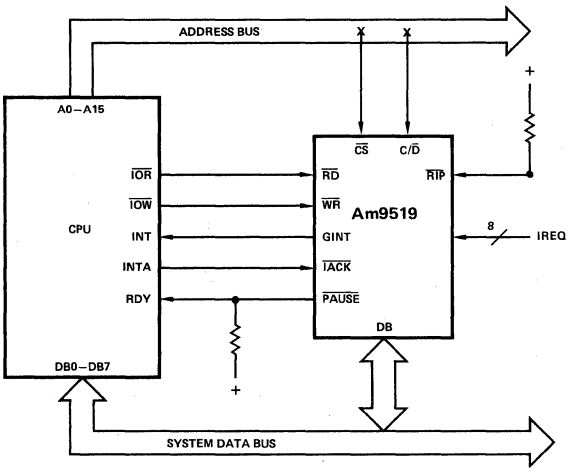


Figure 6. Base Interrupt System Configuration.

MOS-146

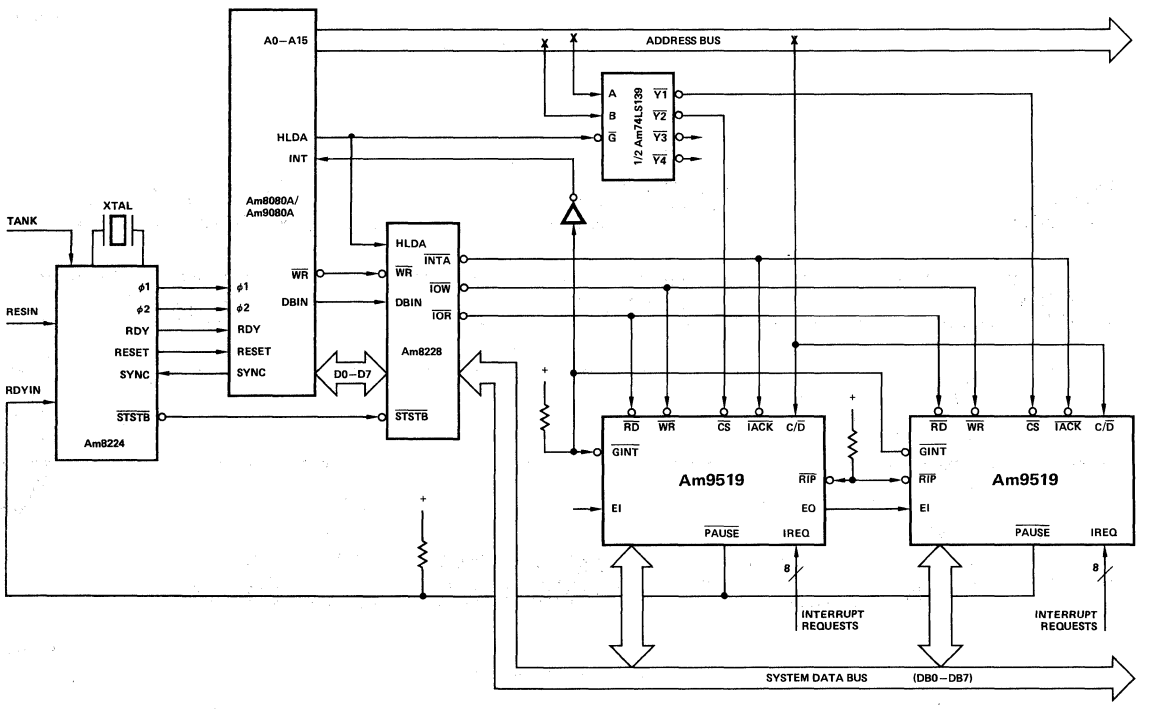


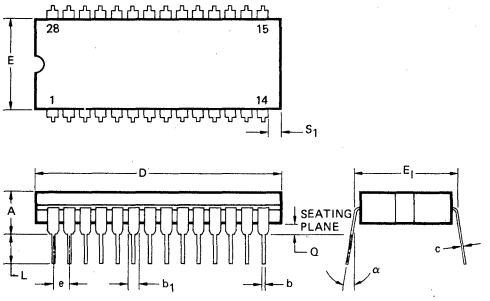
Figure 7. Expanded Interrupt System Configuration.

MOS-147

10

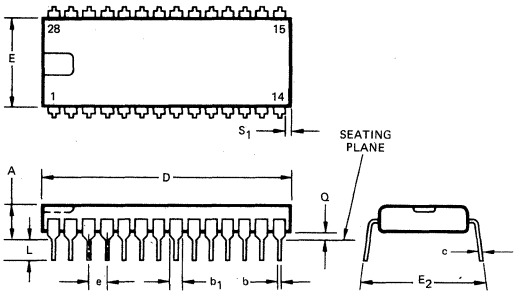
PHYSICAL DIMENSIONS

28-Pin Cerdip



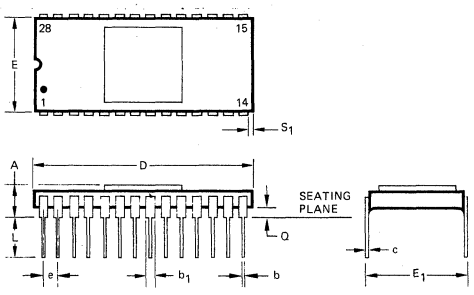
Reference Symbol	Inches	
	Min.	Max.
A	.150	.225
b	.016	.020
b_1	.045	.065
c	.009	.012
D	1.440	1.490
E	.510	.545
E_1	.600	.620
e	.090	.110
L	.125	.150
Q	.015	.060
S_1	.010	

28-Pin Molded DIP



Reference Symbol	Inches	
	Min.	Max.
A	.150	.200
b	.015	.020
b_1	.055	.065
c	.009	.011
D	1.450	1.440
E	.530	.550
E_2	.585	.700
e	.090	.110
L	.125	.160
Q	.015	.060
S_1	.040	.070

28-Pin Side-Brazed Ceramic



Reference Symbol	Inches	
	Min.	Max.
A	.100	.200
b	.015	.022
b_1	.030	.060
c	.008	.013
D	1.380	1.420
E	.560	.600
E_1	.580	.620
e	.090	.110
L	.120	.160
Q	.020	.060
S_1*	.005	
α	0	0

Applications

11

Advanced Micro Devices

The Am9517A Multimode Direct Memory Access Controller

Copyright © 1978 & 1979 by Advanced Micro Devices, Inc.

Advanced Micro Devices cannot assume responsibility for use of any circuitry described other than circuitry entirely embodied in an Advanced Micro Devices' product.

AM-PUB092

TABLE OF CONTENTS

I. INTRODUCTION
- General 11-3
- Am9517A Overview 11-4

II. INTERFACING
- Block Diagram 11-5
- Interface Considerations 11-5

III. INTERFACE SIGNALS
- Clock 11-7
- Chip Select ($\overline{CS}$) 11-7
- Reset (RST) 11-7
- Ready (RDY) 11-7
- Hold Acknowledge (HACK) 11-8
- DMA Request (DREQ0-DREQ3) 11-8
- Hold Request (HREQ) 11-8
- DMA Acknowledge (DACK) 11-8
- Address Bus (A0-A15) 11-8
- Address Strobe (ADSTB) 11-8
- Address Enable (AEN) 11-9
- Memory Read ($\overline{MEMR}$) 11-9
- Memory Write ($\overline{MEMW}$) 11-9
- IO Read ($\overline{IOR}$) 11-9
- IO Write ($\overline{IOW}$) 11-9
- End of Process ($\overline{EOP}$) 11-9
- Data Bus (DB0-DB7) 11-9
- Power (VCC, VSS) 11-9

IV. REGISTER DESCRIPTION
- Address and Word Count Registers 11-10
- Status Register 11-11
- Command Register 11-11
- Request Register 11-12
- Mask Register 11-12
- Mode Register 11-13
- Byte Pointer Flip-Flop 11-14
- Temporary Data Register 11-14

V. SYSTEM INTERFACE
- Software Configurations 11-15
- General Control Routine 11-16
- Hardware Configurations 11-19

INTRODUCTION

General

Direct Memory Access (DMA), also sometimes known as "channel I/O" or "cycle stealing", has long been a feature of mini- and mainframe computer architectures. It was developed as a means to provide data transfer between system memory and peripheral devices at speeds higher than those obtainable under control of the CPU. This is achieved by bypassing the CPU and creating a direct path between memory and the peripheral device.

DMA capability significantly enhances the throughput performance of a processor system. To understand why this is so, recall that a CPU exercises its functions by reading an instruction from memory, decoding it, performing any address calculations necessary to locate operands, and then executing the specified operation. These are necessary steps with any processor, and it may take several instruction fetch/execute sequences to transfer each byte or word. For repetitive sequential data movement operations, the CPU can be removed from the transfer path and the normal fetch/execution steps simplified.

As an example, a data acquisition subroutine flow chart is illustrated in Figure 1. The subroutine stores bytes of data from an external source into successive memory locations. The main loop consists of several instructions requiring many clock cycles for each word transferred.

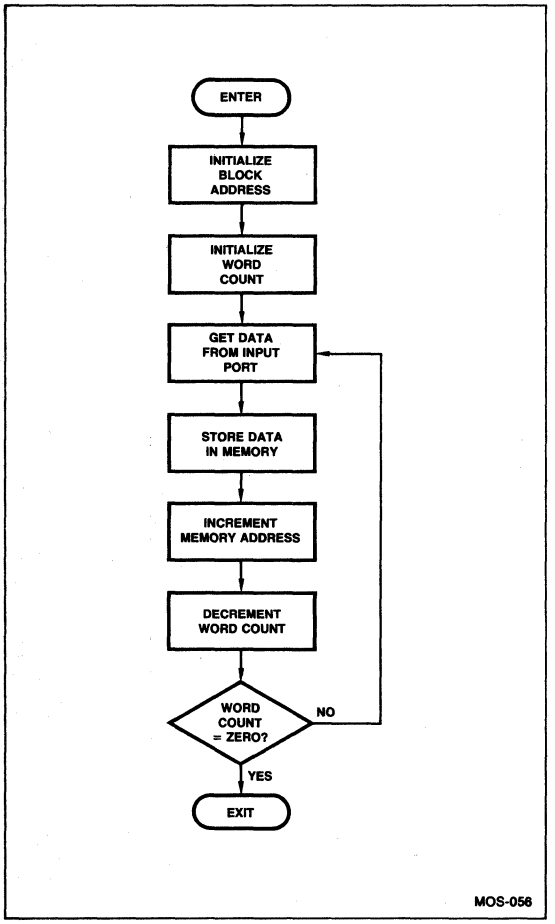


Figure 1. Block Transfer Flow Chart.

However, if what is required is to transfer a complete block of data between a source and a destination, a great deal of overhead can be eliminated by specialized hardware logic which provides access to system memory without CPU intervention. Direct Memory Access (DMA) consists of replacing the functions in the software loop with dedicated hardware. This can reduce the number of cycles required per word transferred and dramatically increase the information transfer rate.

For a more concrete comparison, the flow chart of Figure 1 has been coded in 8080A/8085 assembly language as shown in Figure 2. The transfer loop takes 10 bytes of program and executes in 46 clock cycles per byte. The same task executed in the Am9517A DMA Controller requires a little more initialization time but results in a transfer loop that takes only 3 clock cycles per byte (and an optional operating mode allows transfers only 2 clocks long for extra high speed). Thus, DMA operation can provide throughput gains of much more than an order of magnitude.

Architectural and semiconductor processing developments have occurred rapidly since the introduction of the first 8-bit general purpose microprocessor in 1973. Basic instruction execution times and minimum system component counts have both improved by more than an order of magnitude. CPU costs have fallen from several hundred dollars to well under ten dollars. The result has been an unprecedented growth in the applications for microprocessors. Furthermore, during this same interval, the 8-bit microprocessor that was originally conceived as a smart controller has evolved into a sophisticated computing element. This evolution has generated a concomitant need for efficient and high speed I/O transfers which is the forte' of direct memory access.

The ever increasing density of MOS/LSI has allowed the considerable amount of logic required within a useful direct memory access controller to be almost entirely incorporated on a single chip. Indeed, the only reason that more than one chip is required is the limitation on the number of pins available in industry-standard packages.

```
LOC   OBJ          SEQ     SOURCE STATEMENT
                     1 ;
                     2 ;    SUBROUTINE TO MOVE A BLOCK OF DATA
                     3 ;    FROM AN INPUT PORT INTO MEMORY
                     4 ;
                     5 ;
                     6 ;
                     7      EXTRN MEMA,CNT
                     8 ;
0001                 9 PORT1  EQU 01H        ;INPUT PORT NO. 1
                    10 ;
                    11      CSEG
                    12 ;
0000  210000  E     13 BM17: LXI H,MEMA     ;INIT. MEMORY ADDRESS
0003  010000  E     14       LXI B,CNT      ;INIT. WORD COUNT
0006  DB01          15 LOOP: IN PORT1       ;READ FROM INPUT PORT
0008  77            16       MOV M,A        ;STORE DATA IN MEMORY
0009  23            17       INX H          ;INCREMENT MEMORY POINTER
000A  0B            18       DCX B          ;DECREMENT WORD COUNT
000B  78            19       MOV A,B        ;TEST IF WORD COUNT = 0?
000C  B1            20       ORA C
000D  C20600  C     21       JNZ LOOP       ;LOOP UNTIL DONE
0010  C9            22       RET
                    23 ;
                    24 ;
                    25       END
```

Figure 2. Programmed Block Transfer.

Various forms of DMA control have been implemented in the past. One approach is to interleave individual memory transfers with continuing CPU memory operations. This allows "transparent" DMA, but does not take advantage of the available memory bandwidth, plus the interface and control logic becomes very CPU-specific.

The Am9517A uses another technique where it takes over complete system control and can therefore make full use of the memory speed. This Application Note describes the Am9517A developed by Advanced Micro Devices. In addition to a full description of the part and its various modes of operations, implications and configuration suggestions are made to aid the system designer.

Am9517A Overview

The functional specification of the Am9517A Multimode Direct Memory Access Controller was developed to offer a very general DMA capability with a wide range of programmable options, many system-oriented features, and a general purpose interface facilitating the use of the device with a wide range of system architectures.

Careful study of the needs of a wide variety of applications which could benefit from DMA capability indicated the need for a minimum (but expandable) set of four separate and independent channels. Each channel in the Am9517A has associated with it two 16-bit registers which contain the current address and current word count information and two more registers which contain the base address and base word count. The base registers permit any channel to be automatically re-initialized at the end of a transfer. In addition, each channel has associated with it a 6-bit mode register which determines the types of transfers and options to be executed.

Overriding control of the operation of the DMA controller is provided by a master enable/disable bit in an internal command register. In addition, a four-bit mask register is provided to allow individual channels to be enabled and disabled. An important feature of the mask register is that its contents may be set or cleared in two ways. All four channels may be simultaneously enabled and/or disabled by means of a single command from the CPU. Alternatively, individual channels may be enabled or disabled without disturbing the status of the other channels. The use of this latter technique makes it unnecessary for the system software to maintain the mask status of all channels when changing just one.

Priority resolution logic is provided in order to resolve potential conflicts among requests for DMA service. Two software-selectable priority strategies are available to the system designer. Fixed priority gives the highest priority to channel 0 and the lowest to channel 3. Rotating priority maintains the same relative order as fixed priority, but assigns the lowest priority to the channel last serviced, thus preventing any one channel from monopolizing the controller.

Each of the channels operates in one of four modes. In the Single transfer mode one word is transferred in response to each DMA request of the channel. Block transfer mode causes the DMA controller to make continuous transfers until the word count for the active channel goes to 0. Demand transfer mode makes transfers as long as the request for DMA service is active and the word count for the channel is non-zero. The fourth mode of operation, called the cascade mode, is used to provide nearly unlimited expansion of the number of DMA channels available. A channel operating in cascade mode responds to a DMA request from a cascaded DMA controller by issuing a request to the CPU for control of the bus in the usual way. Acknowledgement by the CPU of the request is passed on to the cascaded controller by acknowledging the DMA request. No other address or control signals are activated by a cascade mode channel.

Two of the channels of the Am9517A can be used to provide memory-to-memory transfer capability. This feature offers very fast transfers of data blocks within system memory and is especially valuable in editing, initialization and other data movement operations. Channel 0 provides the source address while channel 1 provides the destination address and word count. A programmable feature of the memory-to-memory transfer operation is the ability to hold the source address constant. This allows the data contained in a single memory location to be replicated at very high speed throughout a block of memory.

Other programmable features which apply to all modes include the ability to select either incrementing or decrementing of the current address during transfers, a compressed timing feature which allows transfers to be executed in just two clock cycles, and the ability to select the active sense of the DMA Request and DMA Acknowledge signals.

A software DMA request capability is included which allows DMA transfers to be initiated by the processor on any channel. This powerful feature permits the processor itself to take full advantage of the capability of the DMA controller. An external hardware input is provided that allows the system to terminate a transfer when desired.

The Am9517A makes use of Advanced Micro Devices' LINOX N-channel silicon gate MOS technology. This process utilizes low profile structures, triple ion implantation, and both depletion and enhancement transistors to achieve very dense, high speed, low power circuitry. The chip contains 6350 transistors, has a total area of 41,580 square mils and is packaged in a standard 40-pin dual in-line package.

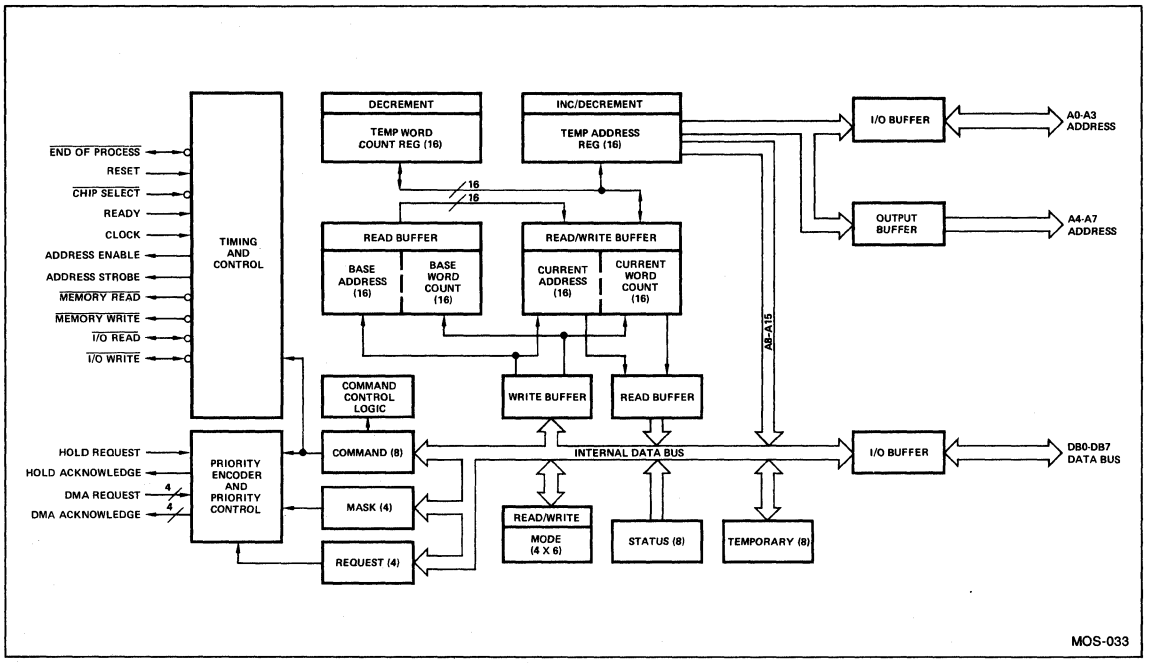


Figure 3. Am9517A Block Diagram.

INTERFACING

Block Diagram

The block diagram of the Am9517A (Figure 3) shows all of the interface signals in addition to the internal functional blocks and their data interconnections.

A peripheral device requiring service generates a DMA request to the Am9517A. If the channel receiving the request is enabled, a Hold Request to the system CPU is issued by the controller. When the CPU relinquishes control over the system busses, a Hold Acknowledge signal is output to the DMA controller to indicate that transfers may begin. On receipt of Hold Acknowledge the Am9517A issues a DMA Acknowledge to the highest priority, unmasked requesting device and begins issuing the control signals and addresses necessary to effect the desired transfers. Upon completion or termination of the transfer the Hold Request and DMA Acknowledge signals are terminated and the CPU regains control of the system busses. This procedure allows the DMA Controller to take full advantage of the available memory bandwidth and provides the greatest possible flexibility for transfer timing.

In order to establish the required operating characteristics within the DMA controller, internal registers are loaded under software control by the CPU. The numerous internal registers are addressed by means of the four least significant address lines (A0 through A3) which are thus made bidirectional. Address lines A4 through A7 are output by the controller and the eight high order address bits (A8 through A15) are demultiplexed from the data bus using the Address Strobe signal. The high order address byte is stored in an external latch and supplied to the address bus when required by the Address Enable (AEN) output signal. The I/O Read and I/O Write signals, in conjunction with Chip Select and the low order address bits, are used by the CPU to communicate with the controller.

During a DMA transfer the controller generates the combinations of read and write signals necessary to effect the transfers, using the Ready input, where necessary, to synchronize timing. End-Of-Process is a bidirectional signal which, as an output, indicates that a DMA transfer has been completed and, as an input, may be used to terminate any current transfer.

Interface Considerations

All of the input and output signals of the Am9517A are specified with worst-case levels identical to those of standard TTL circuits. Input logic levels are 2.0V high and 0.8V low; output logic levels are 2.4V high and 0.4V low. Thus, the normal worst-case noise immunity of 400mV offered by standard TTL logic is maintained. The logic level specifications take into account worst-case combinations of the three variables that effect the logic level thresholds: ambient temperature, supply voltage and processing parameters. Actual operating margins will be better than worst-case to the extent that these variables depart from worst-case conditions.

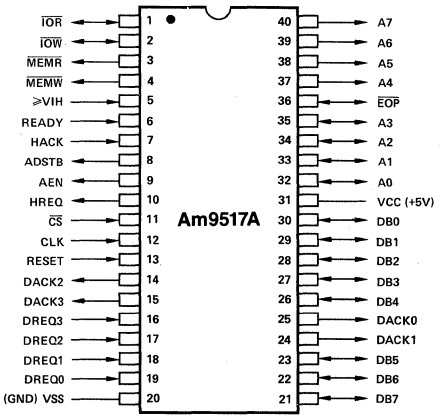
All outputs source at least 200 microamps at 2.4V. The Hold Request output is also specified for 100 microamps source current at 3.3V. All outputs sink a minimum of 3.2 milliamps at 0.4V. All the interface signals of the Am9517A are summarized by type in Figure 4a and their pin assignments are shown in Figure 4b.

The very high resistance of open gate MOS transistors exposes the device's input circuits to the risk of damaging accumulations of static charge. If charge enters the gate node of such an input faster than it can be discharged, the gate voltage can rise high enough to cause oxide breakdown, thus damaging or destroying the transistor.

All inputs to the Am9517A include protection networks designed to slow the transition times of incoming current surges and to provide low impedance discharge paths for voltages beyond normal

Signal Name	Abbreviation	Type	No. of Pins
Clock	CLK	Input	1
Chip Select	$\overline{CS}$	Input	1
Reset	RESET	Input	1
Ready	READY	Input	1
Hold Acknowledge	HACK	Input	1
DMA Request	DREQ0-DREQ3	Input	4
Hold Request	HREQ	Output	1
DMA Acknowledge	DACK0-DACK3	Output	4
Address Bus 4-7	A4-A7	Output	4
Address Strobe	ADSTB	Output	1
Address Enable	AEN	Output	1
Memory Read	$\overline{MEMR}$	Output	1
Memory Write	$\overline{MEMW}$	Output	1
I/O Read	$\overline{IOR}$	Input/Output	1
I/O Write	$\overline{IOW}$	Input/Output	1
Data Bus	DB0-DB7	Input/Output	8
Address Bus 0-3	A0-A3	Input/Output	4
End of Process	$\overline{EOP}$	Input/Output	1
+5 Volts	VCC	Power	1
Ground	VSS	Power	1

(a) Signal Summary.



(b) Connection Diagram.

MOS-034

Figure 4. Am9517A Interface Signals.

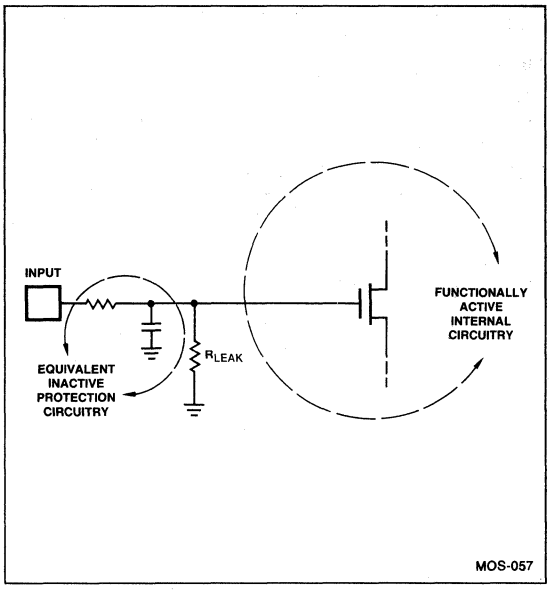


Figure 5. Input Circuitry.

operating levels. Note, however, that careless handling of MOS components can result in transfers of charge which cannot be absorbed without damage and conventional MOS handling precautions should be observed at all times.

In normal operation the input protection circuitry is inactive and may be considered as a lumped series RC network as shown in Figure 5. The active input connection during normal operation is the gate of an MOS transistor. No active sources or drains are connected to the inputs so that neither transient nor steady state currents are impressed upon the driving signals by the Am9517A The input signal is required only to charge or discharge the input capacitance and to overcome the leakage associated with the protection network and input circuit. Input capacitances are typically 6pF and leakage currents are usually less than 1μA. As is typical with MOS components, input drive specifications will usually be limited by transition time considerations rather than DC current limitations.

Good MOS design practice dictates that all inputs be terminated in order to provide discharge paths for transients. Unused inputs should be tied directly to ground or VCC as appropriate. Any input which is driven directly from a card edge connector should be terminated on the card in order to protect the input when the connection is broken. A simple pull-up resistor or on-board gate will suffice.

In general, on-chip delays will always track to a great extent and worst/best combinations will never occur together. The rising and falling edges of the read and write control pulses will track to provide minimum active widths.

INTERFACE SIGNALS

Clock

All the internal operations are clocked by this input which is used by the Am9517A to create four internal clocks. The number of operations to be performed within the controller and the high speed at which the device has been specified to operate, impose constraints on the input clock specification as illustrated in Figure 6. As long as the specified minimum high and low times are observed, the designer may use any convenient clock duty cycle. The greatest operating margins, and the best transition time tolerance, occurs when both clock low and clock high times are mutually maximized. Slowing the clock slightly, and thus extending TCY somewhat, also provides room for greater margins.

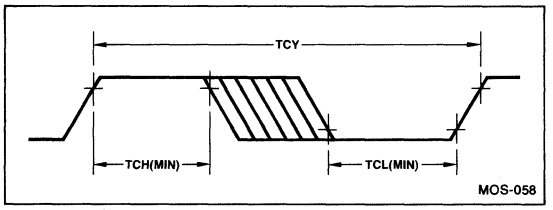


Figure 6. Clock Waveform Specifications.

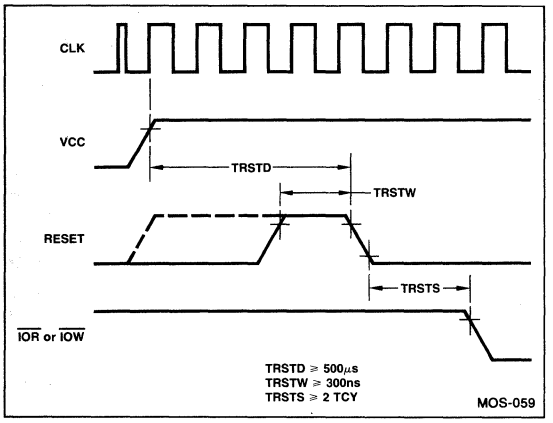


Figure 7. Reset Timing.

Chip Select ($\overline{CS}$)

Chip Select is an active-low input that enables data transfers between the DMA controller and the data bus. It is usually derived from an address output by the CPU to specify the device to be operated upon. The Chip Select input is recognized by the controller only when no active DMA or memory-to-memory transfers are taking place. It is ignored whenever the HACK input is true.

Reset

Reset is an active-high asynchronous input which terminates any operation that may be in progress within the controller. This input also clears the internal control registers, with the exception of the mask register which is set to all ones thus disabling all channels. After a reset the controller is ready to accept initialization commands from the CPU. Reset must be applied for a minimum of 300ns in order to insure that the controller is reset. The end of Reset should occur at least 500μs after both VCC and CLK are stable. At least two clock cycles must elapse following the end of Reset before the first I/O write is applied to the controller. Figure 7 shows the reset timing.

Ready

Ready is an active-high input which can be used to extend the read and write pulses generated by the DMA controller. If the Ready input is low throughout the Ready setup (TRS) and hold (TRH) times as shown in the left portion of Figure 8, the read and write pulses will be extended by one full clock cycle. The right portion of Figure 8 shows extension by two clocks. Ready is tested on the falling edge of each succeeding clock cycle and the read and write pulses are extended in multiple increments of TCY until Ready becomes true prior to the setup time. When utilizing the compressed timing feature (discussed later) to achieve a transfer rate of one word every two clock cycles, care must be taken to insure that Ready is true throughout the setup and hold time prior to the read and write pulses being true. Ready transitions should not occur during the defined setup-hold window in any operating mode.

In addition to extending the width of the Read and Write pulses, the Ready input going low causes the state of the address, data and control lines to be held constant. Note that, just as in the case of the CPU, the Ready input to the DMA controller remaining low for an extended period will hang up the system.

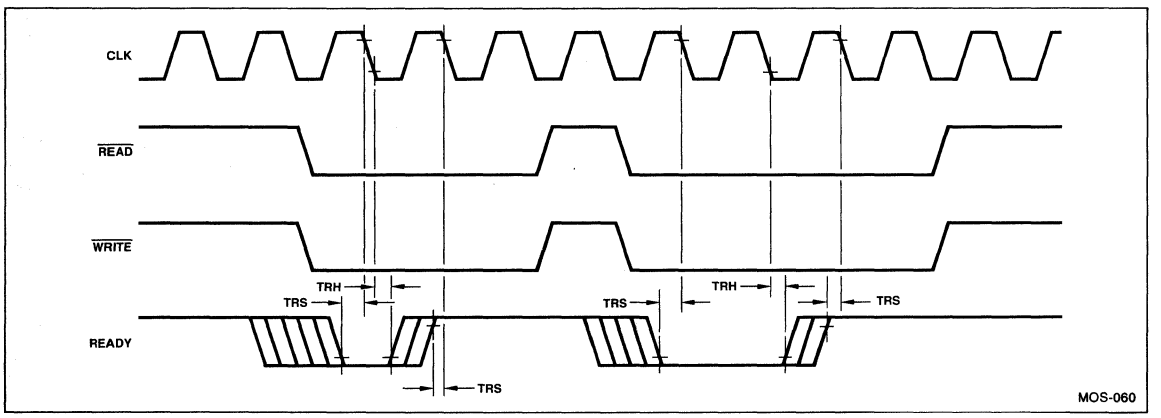


Figure 8. Ready Timing.

Hold Acknowledge (HACK)

The active-high Hold Acknowledge signals the controller that the system busses have been released by the CPU and placed in the high impedance state. This input initiates DMA transfers and should be asserted only in response to the issuance of a request for service by the Am9517A (HREQ). The DMA operations begin with the falling edge of the clock cycle following the satisfaction of the Hold Acknowledge setup time requirement (THS). The Hold Acknowledge input must remain true until the Am9517A relinquishes control of the system busses. The controller requires that one full clock cycle transpire between issuance of a HREQ and receipt of the Hold Acknowledge.

DMA Request (DREQ0-DREQ3)

The four asynchronous DMA Request inputs are used by peripheral devices to request DMA service. The polarity of the DREQ inputs is program selectable to be either active-high or active-low. A reset places them in the active-high condition.

An active DMA request is recognized by the Am9517A at the clock falling edge next following the satisfaction of the DREQ setup time (TQS). Any or all of the DMA request inputs may be active simultaneously and contention is resolved by priority logic contained within the DMA controller. Each Hold Acknowledge received by the controller selects the highest priority unmasked active DMA Request for the next transfer sequence. This means that a DMA transfer must be completed or otherwise terminated in order for a higher priority channel to be serviced.

Once asserted, an active DREQ input should be maintained at least until the corresponding DACK goes active. The implications of various methods of managing the DREQ signal timing after DACK is returned will depend on the operating mode being used for the associated channel.

Hold Request (HREQ)

The active-high Hold Request output indicates that the Am9517A requires service and is generated whenever an unmasked active DMA Request input is received. Timing of the Hold Request relative to the DREQ input is shown in Figure 9. In order to accommodate processors with higher level input specifications, the Hold Request output is designed to source at least 100 microamps at a VOH of 3.3 volts. The TDQ parameter shown in Figure 9 is specified at both 2.4 volts and 3.3 volts in the Am9517A data sheet. The higher level can be attained without assistance, but the timing specified requires an external pullup resistor.

DMA Acknowledge (DACK)

The four DMA Acknowledge outputs are each associated with one of the DMA Request inputs. Their active levels are selectable under program control. Unlike the DREQ inputs, after a reset the DMA Acknowledge outputs will be in the active-low condition. When a HACK is received from the CPU in response to the HREQ from the DMA controller, the highest priority, active, unmasked DMA Request will be granted service and the DMA Acknowledge output associated with that channel will become active, as shown in Figure 9. The DMA Acknowledge output from the Am9517A remains true until the completion of the requested DMA service and becomes inactive after the Hold Request output becomes inactive as shown in Figure 9. There will never be more than one DACK active at a time. No DACK is issued for memory-to-memory operations.

Address Bus (A0-A15)

In order to accommodate all of the functions provided by the Am9517A within the constraints of a 40-pin package, the 16 bits of address information provided by the device are output on two paths. The least significant eight are output on the eight address lines A0-A7. The four least significant address lines are bidirectional. As inputs they address the internal registers of the Am9517A when programming the device. Address lines A4 through A7 are tri-state outputs which are enabled only during DMA operations. The most significant eight bits of the address are output on the data bus during certain portions of the DMA operation. They can be demultiplexed from the data bus and stored in an external latch. The timing for the address bits and the two associated control lines is shown in Figure 10.

Address Strobe (ADSTB)

The active-high Address Strobe output from the Am9517A is the control signal used to latch the high order address bits (A8-A15) from the data bus into an external register or latch. Note that, as shown in Figure 10, the falling edge of Address Strobe should be used to clock the address bits into the latch; the data bus may not be valid at the rising edge. An important feature of the Am9517A is that the high order address and its associated strobe are issued only when required, namely during the first active cycle of a transfer and thereafter only when a carry or borrow is generated by the least significant address byte. This eliminates a clock cycle from the vast majority of transfers.

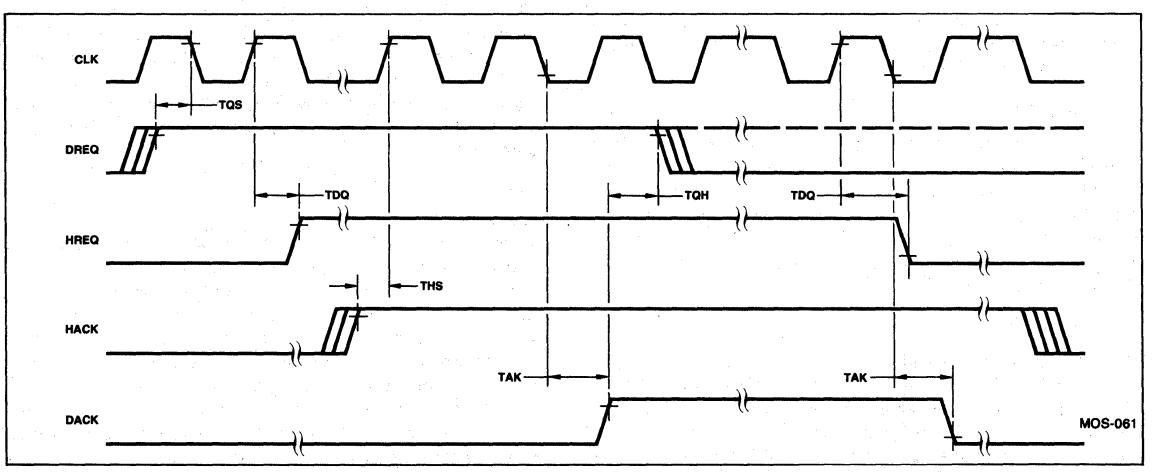


Figure 9. Peripheral and CPU Handshaking Interfaces.

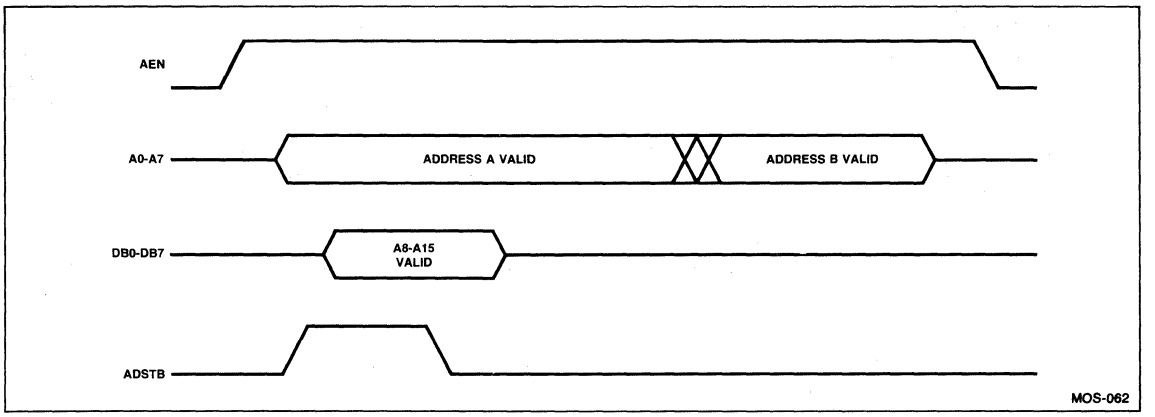


Figure 10. Address Control Relationships.

Address Enable (AEN)

Address Enable is an active-high control signal output by the Am9517A during the first clock cycle of a DMA operation. It remains valid throughout the transfer, as shown in Figure 10. The principal function of this signal is to enable the outputs of the tri-state latch which holds the eight high order address bits. Since the Address Enable output is asserted early in the DMA transfer and remains true until completion, it may also be used to put other system signals into their tri-state condition and to disable system activities such as IO chip select decoding. If AEN is true and all DACK remain false, a memory-to-memory operation is taking place.

Memory Read ($\overline{\text{MEMR}}$)

The transfer of data under the control of the Am9517A requires, in addition to the generation of address information, the provision of control signals to read data from a source location and write it into a destination. These control signals are derived from internally generated read and write signals based upon the type of transfer being executed. The active-low, three-state Memory Read output is used to initiate the reading of data from system memory.

Memory Write ($\overline{\text{MEMW}}$)

The active-low three-state Memory Write output is one of a group of four control signals utilized to initiate the reading and writing of data under the control of Am9517A. Its function is to control the writing of data into the system memory.

IO Read ($\overline{\text{IOR}}$)

IO Read is a bidirectional active-low signal. As an output it is used to control the reading of data from an external peripheral port. As an input, IO Read is recognized only if the DMA controller has been selected by the Chip Select input and no DMA operations are underway. If these conditions are met the IO Read input will cause an 8-bit byte to be read from the register addressed by A0-A3 onto the data bus.

IO Write ($\overline{\text{IOW}}$)

IO Write is a bidirectional active-low signal. As an output it is used to control the writing of data into peripheral ports. Like IO Read, this signal will not be recognized as an input unless CS is valid and no DMA operations are in progress. When recognized as an input this signal causes the information on the data bus to be loaded into the Am9517A register addressed by A0-A3.

$\overline{\text{IOR}}$, $\overline{\text{IOW}}$, $\overline{\text{MEMR}}$, $\overline{\text{MEMW}}$ operate in pairs to control DMA information transfers. For peripheral-to-memory, $\overline{\text{IOR}}$ and $\overline{\text{MEMW}}$ are both active at the same time. For memory-to-peripheral, $\overline{\text{MEMR}}$ and $\overline{\text{IOW}}$ are both active in the same cycle. For memory-to-memory only $\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$ are used and only one at a time.

End of Process ($\overline{\text{EOP}}$)

End of Process is a bidirectional active-low signal. As an output it is active for a single clock period when the Word Count of an active DMA channel goes to zero. Asserting End of Process as an input causes the termination of DMA operations.

$\overline{\text{EOP}}$ may be used in many ways. As an interrupt, it can inform the CPU that a DMA transfer has occurred. It also can help coordinate peripheral device activity. When channel-specific signals are desired, $\overline{\text{EOP}}$ can be simply gated with individual DACK lines to generate EOP0-EOP3.

Data Bus (DB0-DB7)

The eight bidirectional three-state data bus signals transfer information between the Am9517A and the system data bus. During DMA operations the data bus signals are activated as outputs to supply the high order address byte. Note that the DMA data being transferred does not enter the Am9517A except in the case of memory-to-memory operation. During memory-to-memory operations the data being transferred is stored in a temporary register within the Am9517A between the read and write operations. During memory-to-memory operations the controller cycle which outputs the high order address byte and the address strobe is always present. During programmed IO operations on the Am9517A the data bus is utilized to transfer bytes between the CPU and the DMA controller.

Power (VCC, VSS)

The Am9517A makes use of a single +5 volt power supply and ground. One pin (pin 5) of the device which is not used by the interface must be at a logic high level. An internal pullup resistor is connected to pin 5 to provide the high level when the input is floated. Pin 5 may also be connected directly to VCC. The Am9517A requires a maximum of 150mA at 5V ±5% over the standard operating temperature range of 0°C to 70°C and 175mA at 5V ±10% over the full military temperature range of −55°C to +125°C. These maximum supply currents are worst-case values and apply at the lowest specified temperatures. Supply current always decreases with increasing temperatures; worst-case current at 25°C is 130mA.

REGISTER DESCRIPTION

Figure 11 summarizes the registers contained within the Am9517A. The various Address and Word Count registers control locations and numbers of transfers for active operations. The Command, Mode, Mask and Request registers manage the operating options and control features available to the system. Figure 12 shows the addressing used to access the Address and Word Count locations. Notice that A3 is always logic zero for that range of addresses. When A3 is a logic one, other locations are addressed as shown in Figure 13. All operations marked illegal should not be attempted as results will be unknown.

Name	Size	Number
Base Address Registers	16 bits	4
Base Word Count Registers	16 bits	4
Current Address Registers	16 bits	4
Current Word Count Registers	16 bits	4
Temporary Address Register	16 bits	1
Temporary Word Count Register	16 bits	1
Status Register	8 bits	1
Command Register	8 bits	1
Temporary Register	8 bits	1
Mode Registers	6 bits	4
Mask Register	4 bits	1
Request Register	4 bits	1

Figure 11. Am9517A Internal Registers.

Address and Word Count Registers

Each of the four channels provided by the Am9517A has associated with it four 16-bit programmable registers. Two of these, the current address and base address registers, are used to provide address information for data transfer. The current word count and base word count registers determine the number of words to be transferred by a DMA operation.

The base registers are loaded in parallel with the current registers. Two single-byte IO write operations to the same address are used to fill the 16-bit registers. An internal byte pointer flip-flop, which is cleared by a reset, master clear, or by IO command, changes state each time one of the 16-bit registers is accessed. It is used to steer the incoming 8-bit data to the least and most significant halves of the registers.

Two points must be emphasized with regard to the loading of the address and word count registers. First, since the current and base registers are loaded in parallel, no attempt should be made to change the contents of the base register while a DMA service is in progress on that channel. If, for example, transfer concatenation is desired, two channels should be utilized, with the incoming DMA requests being switched between them.

Second, the byte pointer flip-flop toggles automatically upon register access. This requires that care must be taken when accessing these registers other than in the initialization mode. For example, a subroutine which is called as a result of an interrupt and which accesses the address or word count registers should include the IO instruction which clears the byte pointer flip-flop.

Channel	Register	Operation	Interface Signals						Byte Pointer Flip/Flop	Data Bus DB0-DB7
			$\overline{IOR}$	$\overline{IOW}$	A3	A2	A1	A0		
0	Base & Current Address	Write	1	0	0	0	0	0	0	A0-A7
			1	0	0	0	0	0	1	A8-A15
	Current Address	Read	0	1	0	0	0	0	0	A0-A7
			0	1	0	0	0	0	1	A8-A15
	Base & Current Word Count	Write	1	0	0	0	0	1	0	W0-W7
			1	0	0	0	0	1	1	W8-W15
	Current Word Count	Read	0	1	0	0	0	1	0	W0-W7
			0	1	0	0	0	1	1	W8-W15
1	Base & Current Address	Write	1	0	0	0	1	0	0	A0-A7
			1	0	0	0	1	0	1	A8-A15
	Current Address	Read	0	1	0	0	1	0	0	A0-A7
			0	1	0	0	1	0	1	A8-A15
	Base & Current Word Count	Write	1	0	0	0	1	1	0	W0-W7
			1	0	0	0	1	1	1	W8-W15
	Current Word Count	Read	0	1	0	0	1	1	0	W0-W7
			0	1	0	0	1	1	1	W8-W15
2	Base & Current Address	Write	1	0	0	1	0	0	0	A0-A7
			1	0	0	1	0	0	1	A8-A15
	Current Address	Read	0	1	0	1	0	0	0	A0-A7
			0	1	0	1	0	0	1	A8-A15
	Base & Current Word Count	Write	1	0	0	1	0	1	0	W0-W7
			1	0	0	1	0	1	1	W8-W15
	Current Word Count	Read	0	1	0	1	0	1	0	W0-W7
			0	1	0	1	0	1	1	W8-W15
3	Base & Current Address	Write	1	0	0	1	1	0	0	A0-A7
			1	0	0	1	1	0	1	A8-A15
	Current Address	Read	0	1	0	1	1	0	0	A0-A7
			0	1	0	1	1	0	1	A8-A15
	Base & Current Word Count	Write	1	0	0	1	1	1	0	W0-W7
			1	0	0	1	1	1	1	W8-W15
	Current Word Count	Read	0	1	0	1	1	1	0	W0-W7
			0	1	0	1	1	1	1	W8-W15

Figure 12. Address for Word Count and Address Registers.

Interface Signals						Operation
A3	A2	A1	A0	$\overline{IOR}$	$\overline{IOW}$	
1	0	0	0	0	1	Read Status Register
1	0	0	0	1	0	Write Command Register
1	0	0	1	0	1	Illegal
1	0	0	1	1	0	Write Request Register
1	0	1	0	0	1	Illegal
1	0	1	0	1	0	Write Single Mask Register Bit
1	0	1	1	0	1	Illegal
1	0	1	1	1	0	Write Mode Register
1	1	0	0	0	1	Illegal
1	1	0	0	1	0	Clear Byte Pointer Flip/Flop
1	1	0	1	0	1	Read Temporary Register
1	1	0	1	1	0	Master Clear
1	1	1	0	0	1	Illegal
1	1	1	0	1	0	Illegal
1	1	1	1	0	1	Illegal
1	1	1	1	1	0	Write All Mask Register Bits

Figure 13. Register and Function Addressing.

Similarly, if only a single byte of these registers is to be accessed care must be taken to properly maintain the status of the byte pointer.

The Current Address register contains the memory address that is provided by the DMA controller during a transfer on that channel. The Current Address is automatically incremented or decremented, depending upon the programmable option selected, after each word is transferred and so, when read, will indicate the address of the next word to be transferred.

If the autoinitialization feature has been enabled for the channel, the end of a DMA operation, as indicated by either internal or external $\overline{EOP}$, will cause the contents of the Base Address register to be transferred to the Current Address register. Similarly, the Current Word Count register, which is decremented after each word is transferred, may be reinitialized with the contents of the Base Word Count register.

Status Register

This 8-bit read only register, which is accessed by an $\overline{IOR}$ at the address shown in Figure 13, provides the status of the four DREQ inputs and indicates whether a DMA operation has been completed. See Figure 14.

The four least significant bits of the status byte are individually set by an internal or external $\overline{EOP}$ signal. These bits are cleared by a reset, a master clear command, or by reading the status register. Since the status is not retained by the DMA controller after a read, system software will usually save all four of the least significant bits in order to insure that all completed DMA operations, and not merely the one of interest, may be detected.

The four high-order bits of the status register indicate the state of the four DREQ inputs. An active level at DREQ, independent of the programmed DREQ polarity, sets the status bit corresponding to the channel requesting service. Reading the status register has no effect upon these bits.

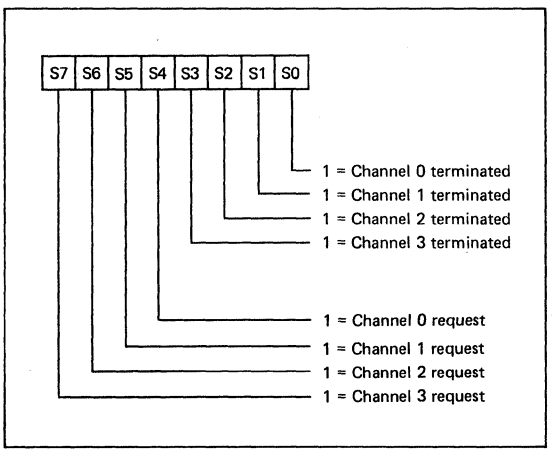


Figure 14. Status Register Bit Assignments.

Command Register

The 8-bit write-only command register is accessed at the location shown in Figure 13. It is cleared by a reset or master clear. Figure 15 indicates the functions carried out by each bit of the command register.

The least significant command bit (C0) controls the memory-to-memory feature. When performing memory-to-memory transfers the Channel 0 Address registers provide the source address, the Channel 1 Address registers provide the destination address and the Channel 1 Word Count registers determine the number of words to be transferred. Memory-to-memory operations are initiated by setting the software DREQ for channel 0.

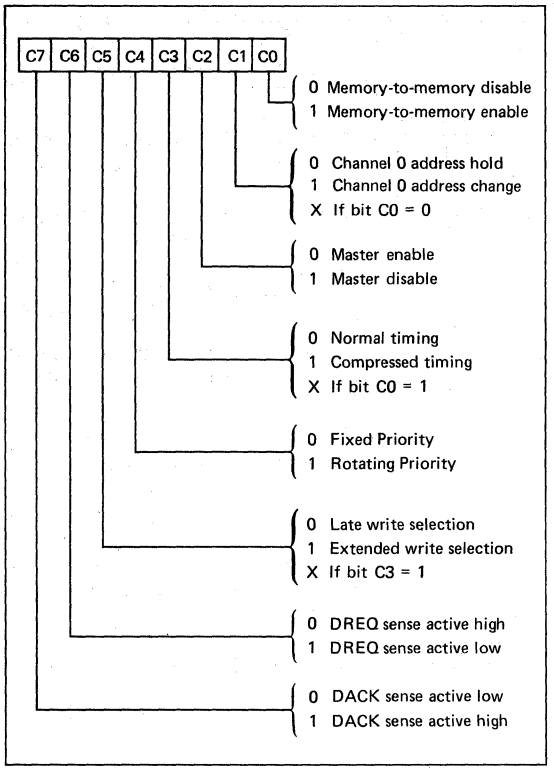


Figure 15. Command Register Bit Assignments.

Transfers proceed at a rate of 8 clock cycles per word. The transfers occur in block mode, that is to say, words will be transferred continuously until the Channel 1 word count reaches 0. It is recommended that Channels 0 and 1 be masked and that the Channel 0 Word Count be set to the same value as that in Channel 1 in preparation for memory-to-memory.

Command bit C1 is effective only if memory-to-memory transfers have been enabled. Under these conditions, setting this bit causes the normal incrementing or decrementing of the Channel 0 Current Address to be inhibited. The result is that the contents of the location defined by the Channel 0 Address register will be written throughout the block of memory defined by the Channel 1 Address and Word Count registers. This feature is useful in applications which require inserting spaces, building histograms and other multiple uses of a single character within data buffers for printers or displays. In some data acquisition applications it is useful to initialize a memory block with an offset value. In displays it will often be convenient to be able to clear screen very rapidly.

Command bit C2 controls the master enable/disable function. When this bit is set the HREQ output of the Am9517A is inhibited, thus preventing any DMA operations from occurring. Note that the entire command register is cleared by a reset or master clear operation, including C2. Although all the hardware DREQ inputs to the Am9517A are disabled by reset, software DMA requests are not masked and should be handled with care.

The compressed timing feature of the Am9517A is selected by means of Command bit C3. Normal timing uses three clock cycles per transfer except when the high-order address bits are output and four clocks are used. Compressed timing removes one clock cycle from each type of transfer thus shortening the transfers to two and three clocks. The effect of this is to reduce the read pulse width by TCY and to cause the read and write pulses to coincide. Where the requirements of system memory and the peripheral controller permit, a substantial increase in throughput can be obtained. This feature is not available during memory-to-memory transfers and the compressed timing bit is ignored if bit zero of the command word is set.

Bit C4 of the command word determines the type of priority arbitration to be utilized in resolving contending active DREQ inputs. If C4 is cleared, the four channels will be prioritized in fixed order, with channel 0 having the highest priority and channel 3 the lowest. If C4 is set, rotating priority is selected. The relative sequence of channel priorities will remain the same; however, upon completion of a DMA operation the most recently serviced channel will become the lowest priority. Thus if channel 2 where the last channel serviced, the order of priority would be channels 3, 0, 1 and 2. Rotating priority prevents a single channel from blocking service to other channels.

Priority arbitration is carried out as the first action of a DMA service upon receipt of the HACK input to the DMA controller. Thus, the highest priority unmasked DREQ input or software DMA request present at that time will be selected for service. Once a channel has been selected for service, control remains with that channel until the service is terminated. HREQ is relinquished following each service so reprioritization will occur.

When the cascade mode of operation is in use, service requests from cascaded DMA controllers are prioritized just like any other DREQ input before being passed on.

Bit C5 of the command register controls the Write pulse width and placement relative to the Read pulse. The optional extended write, which may be required for some memory systems or for some special IO requirements, is selected by this bit. The effect is similar to that of the compressed timing feature in that the write pulse becomes active simultaneously with the read pulse, although both are now two clock cycles wide. The extended write control bit is ignored with the compressed timing option is selected.

The last two bits (C6, C7) of the command register control the active levels of the DREQ inputs and DACK outputs. Note that the polarity of the two bits is such that DREQ will be active-high following reset and DACK will be active-low following reset.

Request Register

The 4-bit write-only Request register is accessed at the location shown in Figure 13. Each bit in the request register may be individually set or cleared based upon the state of the three least significant data bus bits. See Figure 16. The two least significant data bus bits select a channel while the third bit determines whether the selected bit is to be set or cleared. All four of the Request register bits are cleared by a reset or master clear. The request bit for an active channel is cleared by an $\overline{EOP}$.

Software DMA requests are nonmaskable but are disabled by the master disable bit in the command word and are subject to priority arbitration. Due to the nonmaskable nature of these requests, they should be issued only at the end of an initialization or other command sequence when all system setup is complete.

Mask Register

The four-bit write-only Mask register provides the capability to disable any or all of the hardware DREQ inputs to the Am9517A. In order to provide as much flexibility as possible in controlling the mask bits, two addresses are assigned, as shown in Figure 13.

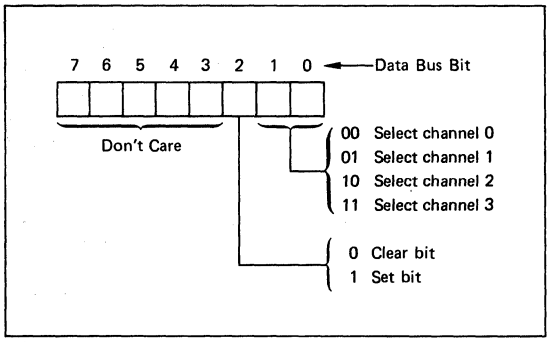


Figure 16. Single Bit Control for Request and Mask Registers.

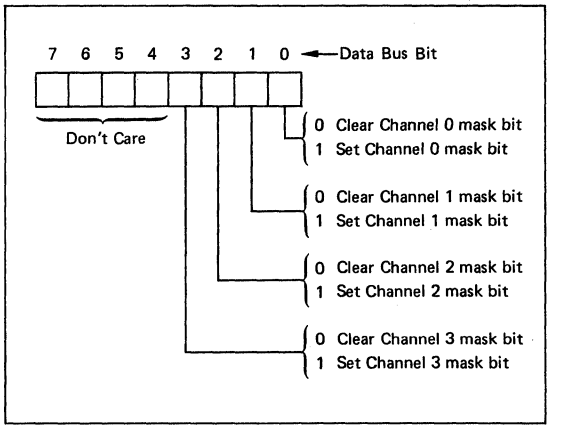


Figure 17. Parallel Mask Loading.

$\overline{IOW}$ commands enabled by $\overline{CS}$ and directed to address 1010 set or clear individual mask bits using the format of Figure 16. Alternatively, all four bits of the mask register may be written with a single command to address 1111. This command utilizes the four least significant data bus inputs to establish the status of the four DMA channels as shown in Figure 17.

In addition to program control of the mask register, the entire register is set by a reset or master clear thus disabling all external DMA requests. Furthermore, an $\overline{EOP}$ on an active channel not programmed to autoinitialize will set the corresponding bit in the mask register. This is done to prevent another DMA transfer from occurring before new address and word data have been set up. Software DMA requests are not maskable.

Mode Register

Each of the four DMA channels has an independent 6-bit write-only Mode register associated with it. This register is accessed at the location shown in Figure 13. When accessing the Mode registers the two least significant data bus inputs select the mode register to which the remaining 6 bits are to be transferred, as shown in Figure 18. These six bits determine the type of transfer, its mode, whether the transfers are to be in ascending or descending order, and whether autoinitialization is to be utilized.

Three types of transfer are selectable by the two least significant bits of the mode register (M2, M3). The verify transfer is a dummy operation during which all program selected functions of the Am9517A are executed except that the read and write control outputs are disabled. This permits the device and its addresses to be exercised without actually transferring data and can be used to verify proper operation of the controller.

The read transfer moves data from system memory to an IO device by activating the $\overline{MEMR}$ and $\overline{IOW}$ control outputs of the Am9517A. Write transfers utilize the $\overline{IOR}$ and $\overline{MEMW}$ control lines to move data from an IO device to system memory. The fourth possible combination of M2, M3 is an undefined state and should not be used. If the channel has been programmed for cascade operation the type of transfer is determined by the attached controller and bits M2, M3 are ignored.

Mode bit M4 specifies the autoinitialization option for the channel. When M4 = 0, the Current Address and Word Count registers are not affected by $\overline{EOP}$ and the associated mask bit is set, disabling the channel. When M4 = 1, occurrence of $\overline{EOP}$ causes the contents of the Base Address and Word Count registers to be transferred into the respective Current Address and Word Count registers. The mask bit is not set.

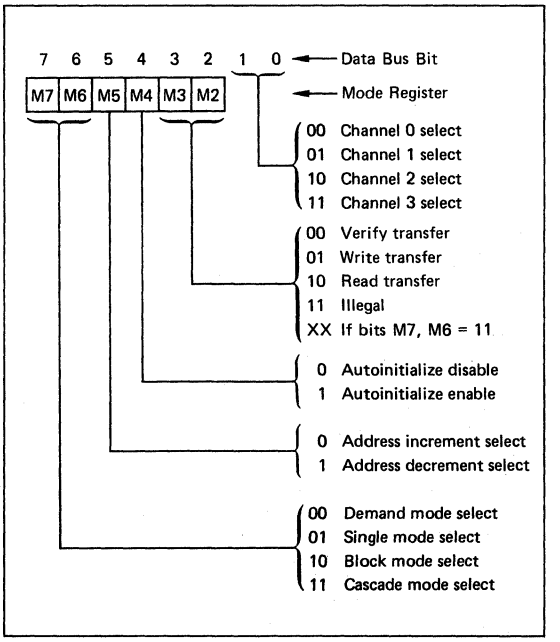


Figure 18. Mode Register Addressing and Assignments.

Autoinitialization allows repetitive DMA operations to proceed without software intervention between blocks. Since each channel has independent base registers, each can be independently autoinitialized without disturbing other channels.

Mode bit M5 specifies the address increment/decrement option for the channel. When M5 = 0 the address will increment following each transfer. When M5 = 1 the address will decrement following each transfer. This feature allows significant versatility in data movement. Blocks that may arrive from peripherals in reverse can be written in descending order in memory so that they end up being forward in memory. With the source incrementing and the destination decrementing, a memory-to-memory move can invert a list.

Memory-to-memory control can override the M5 bit for channel 0, forcing the address to neither increment nor decrement. See the Command Register description for details.

Mode bits M6 and M7 specify one of four available operating modes for the channel. These types of transfer management provide versatility for the interface between the peripheral device and the DREQ input to the Am9517A. One mode is also provided to greatly simplify expansion of the DMA system. All DMA transfers are initiated by an active-going DREQ signal. (Memory-to-memory transfers are initiated by a software request.) All DMA transfers are terminated by an internal (word count = 0) or external $\overline{EOP}$, and by a reset or master clear. The following mode descriptions indicate methods for managing transfers between initiation and final termination.

When M7,M6 = 00 Demand mode is selected. Once the DREQ has been accepted, continuous transfers will occur until the DREQ goes inactive (or until $\overline{EOP}$). When DREQ returns active, transfers will resume where they left off. This allows the requesting device to control the lengths of sub-block-sized bursts of transfers.

For example, if DREQ is cleared at the time that DACK is received, only one word will be transferred. Alternatively, if DREQ is cleared at the time that the internal $\overline{EOP}$ goes active, a complete block (as defined by the word count) will be transferred. Between those extremes Demand mode allows interaction of system activities with DMA transfers. A memory refresh cycle can be executed in the midst of a transfer; availability of external data can control the transfer duty cycle.

When M7,M6 = 01 the Single mode is selected. It operates in two ways. This mode always returns system control to the CPU following each word transferred. If DREQ is then inactive, transfers do not continue. If DREQ remains active transfers will continue (as long as word count is greater than zero) but will always be interleaved with a full HREQ/HACK handshake with the CPU. In the case of the 8080A, this means that machine cycles will alternate with transfers of single words. Notice that the recurring arrival of active HACK means that priority will be re-resolved after each Single transfer. If a higher priority request is pending, it will be serviced. Using rotating priority arbitration with several channels set up in Single mode would then interleave a transfer on each channel with CPU machine cycles.

When M7,M6 = 10, the Block mode is selected. Once the DREQ has been acknowledged (DACK), continuous transfers will take place until $\overline{EOP}$ occurs, independent of the state of the DREQ input. Thus a DREQ pulse wide enough to encompass DACK will cause the movement of an entire block of data. Memory-to-memory operations use Block mode exclusively.

When M7,M6 = 11, the channel may be used to cascade an additional Am9517A circuit in order to expand the channel capacity of the DMA system. Any channel or combination of channels at any level may be used for expansion. Two cascaded Am9517A chips provide a net of seven available channels.

A channel used for cascading simply provides access to the internal priority resolution circuitry by bypassing the other functions of the channel. Address, data and control outputs are disabled when a cascade channel is active, allowing the next active chip to be wire-ORed to the system signals.

Byte Pointer Flip/Flop

An internal flip/flop, toggled by each access to 16-bit registers, is used to select the most significant or least significant register byte. An $\overline{IOW}$ operation to address 1100 clears the pointer, causing it to select the least significant byte of the next 16-bit register accessed. The flip/flop is also cleared by a reset or by the master clear command.

Temporary Data Register

The 8-bit Temporary Data register is used during memory-to-memory transfers to provide temporary storage of the data being transferred. Each byte transferred remains in the temporary register until overwritten by the next and may be read out by the CPU. The Temporary Data register is not used during DMA transfers. It is cleared by a reset or a master clear.

Programming

The Am9517A will accept programming from the host processor any time that HACK is inactive; this is true even if HREQ is active. The responsibility of the host is to assure that programming and HACK are mutually exclusive. Note that a logical conflict can occur if the host masks out a DREQ that has just initiated a HREQ: when HACK arrives no valid request may be pending. To prevent this situation it is suggested that the controller be disabled before the channel mask is set, and then that the controller be re-enabled.

After power-up it is suggested that all internal locations, especially the Mode registers, be loaded with some valid value. This should be done even if some channels are unused.

SYSTEM INTERFACE

Software Configurations

Direct Memory Access is, by definition, transparent to the CPU and does not involve direct software. The programmer (and system designer) will, however, be concerned with initialization of the device, subsequent changes to its programmed configuration, and the potential effect on system activities of a device which is capable of suspending CPU operations for indeterminate times and is, when active, beyond control of the CPU.

This section presents a few routines as examples of ways to drive the Am9517A. They are shown in 8080A/8085 coding. The comments in the routines plus the surrounding text should provide sufficient guidance to allow translation into other machine languages without difficulty. Similarly, these approaches should be modified and extended to reflect the differing requirements of specific applications.

Figure 19 is an "EQU" table that defines the mnemonic labels used in the routines. The values are taken from Figures 12 and 13 and are written in binary to shown that correspondence. Some programmers may want to translate those entries to octal or hex values. Note that this table assumes that hardware IO port Chip Select decoding for the Am9517A selects zero for system address lines A4 through A7. For other arrangements the four high order bits in each EQU table entry should be changed to the value decoded.

Figure 20 shows a simple, straight-forward routine to configure the Am9517A as part of a system initialization procedure, perhaps following power-on for example. It shows the setup for a single channel – in this case channel 2 – but could easily be expanded to include more channels.

Interrupts are disabled to prevent other routines from possibly disturbing the byte pointer flip/flop. The command byte (line 50) specifies these options:

DACK active low
DREQ active low
Extended Write
Fixed Priority
Normal Timing
Controller Enabled
Address not held
Memory-to-memory off.

This routine does not use a software DMA request. If it did, the command byte might want to disable the controller until the initialization is complete in order to prevent unwanted transfers. Master Clear blocks hardware-originated DMA transfers by setting all mask bits, but does not prevent software-originated requests.

```
LOC     OBJ     SEQ          SOURCE STATEMENT
                  1 ;
                  2 ;
                  3 ;PORT ASSIGNMENTS FOR AM9517A A0-A3 INPUTS.
                  4 ;
                  5 ;ADDRESS AND WORD COUNT REGISTERS:
                  6 ;
0000              7 ADR0     EQU 00000000B     ;ADDRESS REGISTERS
0002              8 ADR1     EQU 00000010B
0004              9 ADR2     EQU 00000100B
0006             10 ADR3     EQU 00000110B
0001             11 WCT0     EQU 00000001B     ;WORD COUNT REGISTERS
0003             12 WCT1     EQU 00000011B
0005             13 WCT2     EQU 00000101B
0007             14 WCT3     EQU 00000111B
                 15 ;
                 16 ;CONTROL REGISTERS:
                 17 ;
0008             18 STAT     EQU 00001000B     ;STATUS REGISTER
0008             19 CMND     EQU 00001000B     ;COMMAND REGISTER
0009             20 RQST     EQU 00001001B     ;REQUEST REGISTER
000B             21 MODE     EQU 00001011B     ;MODE REGISTER
000D             22 TEMP     EQU 00001101B     ;TEMPORARY DATA REGISTER
000F             23 MSKR     EQU 00001111B     ;FULL MASK REGISTER
000A             24 MSKB     EQU 00001010B     ;SINGLE MASK BIT
                 25 ;
                 26 ;DIRECT COMMANDS:
                 27 ;
000C             28 CLBP     EQU 00001100B     ;CLEAR BYTE POINTER
000D             29 MCLR     EQU 00001101B     ;MASTER CLEAR
                 30 ;
                 31 ;
                 32 ;
```

Figure 19. EQU Table.

```
LOC   OBJ      SEQ        SOURCE STATEMENT
                 42 ;
                 43 ;
                 44 ;SETUP ROUTINE FOR DMA CHANNEL #2
                 45 ;
3000             46         ORG 3000H
                 47 ;
3000 F3          48 STUP:   DI              ;DISABLE INTERRUPTS.
3001 D30D        49         OUT  MCLR       ;MASTER CLEAR.
3003 3E62        50         MVI  A,01100000B ;SELECT COMMAND OPTIONS,
3005 D308        51         OUT  CMND       ;OUTPUT TO COMMAND PORT.
3007 3E9A        52         MVI  A,10011010B ;SELECT MODE OPTIONS,
3009 D30B        53         OUT  MODE       ;OUTPUT TO MODE PORT.
300B 3E00        54         MVI  A,0        ;FORM LOWER BYTE,
300D D304        55         OUT  ADR2       ;OUTPUT TO ADDRESS REG.
300F 3E0F        56         MVI  A,0FH      ;FORM UPPER BYTE,
3011 D304        57         OUT  ADR2       ;OUTPUT TO ADDRESS REG.
3013 3E07        58         MVI  A,7FH      ;FORM LOWER BYTE,
3015 D305        59         OUT  WCT2       ;OUTPUT TO WORD COUNT.
3017 3E00        60         MVI  A,0        ;FORM UPPER BYTE,
3019 D305        61         OUT  WCT2       ;OUTPUT TO WORD COUNT.
301B 3E02        62         MVI  A,00000010B ;CLEAR CHANNEL #2
301D D30A        63         OUT  MSKB       ;MASK BIT.
301F FB          64         EI              ;ENABLE INTERRUPTS.
3020 C9          65         RET             ;RETURN.
                 66 ;
                 67 ;
                 68 ;
```

Figure 20. Simple SETUP Routine.

The mode byte (line 52) specifies these options:

Block transfer mode
Address increment
Autoinitialize
Read transfer.

With the autoinitialization option in effect, this channel will not have to be setup again until the channel function changes. The Read transfer moves data from the system memory out to the peripheral on Channel 2.

The memory addess of the first byte to be transferred is 0F00 hex. The number of bytes to be transferred is 007F hex. In both cases, the upper and lower bytes are output to the same port and are steered internally by the byte pointer flip-flop.

Once the channel is ready to go, its mask can be cleared (line 63) so that it will recognize a DREQ input. Interrupts are then enabled and control returned to the main program. If this routine, or a variation of it, is used as part of a more general system initialization program segment, it may be appropriate to not clear the mask (and possibly to not enable interrupts) until the complete system setup is finished. If this routine is embedded in the midst of application programming, it may be appropriate to omit the Master Clear function and instead to simply set the mask bit for the channel to be changed. In that case it would also be appropriate to clear the byte pointer before the address is output.

General Control Routine

The approach outlined in Figure 20 can be expanded, of course, to include as many channels as desired. In applications where there are variations in the setup data for a given channel, however, this scheme can be awkward. Figure 21 shows a more general design that provides more versatility for many applications.

The SDMA routine picks up the parameters it needs from the calling program. These "in-line" parameters are located immediately following the instruction that calls SDMA. The format is shown at the start of the routine. SDMA first pulls in the mode data which contains the channel number information as well. This is used to index into a branch table that then transfers control to the code segment appropriate for the channel. This portion of the routine then moves the Address and Word Count values from their in-line position into the DMA registers. SDMA assumes that the command configuration has already been established and will not be changed.

Notice that SDMA masks out the channel selected and then clears the mask after the changes are made. Some applications may want to wait until some other point in the program before clearing the mask.

Once SDMA is in place, other configurations and control sequences are easier to implement. For example, the STUP routine in Figure 20 can then be accomplished by inserting a CALL SDMA procedure at line 52 and eliminating lines 52 through 64.

```
LOC     OBJ       SEQ          SOURCE STATEMENT
                   75 ;
                   76 ;
                   77 ;GENERAL PURPOSE ROUTINE TO SETUP DMA CHANNELS
                   78 ;FOR THE AM9517A. SETUP PARAMETERS ARE PASSED
                   79 ;IN-LINE FROM THE CALLING PROGRAM.
                   80 ;
                   81 ;
                   82 ;
                   83 ;CALL FORMAT:
                   84 ;        CALL SDMA
                   85 ;        (MODE BYTE)
                   86 ;        (LOW ADDRESS BYTE)
                   87 ;        (HIGH ADDRESS BYTE)
                   88 ;        (LOW WORD COUNT BYTE)
                   89 ;        (HIGH WORD COUNT BYTE)
                   90 ;
                   91 ;
                   92 ;
3000               93          ORG 3000H
                   94 ;
                   95 ;
                   96 ;
3000 E3            97 SDMA:  XTHL              ;GET ADDRESS OF MODE BYTE.
3001 7E            98        MOV A,M           ;GET MODE BYTE.
3002 23            99        INX  H            ;POINT TO NEXT PARAMETER
3003 E3           100        XTHL              ;AND REPLACE ADDRESS.
3004 D30B         101        OUT  MODE         ;MODE BYTE TO MODE PORT.
3006 E603         102        ANI  03H          ;ISOLATE CHANNEL #
3008 F604         103        ORI  04H          ;FORM MASK BIT FORMAT AND
300A D30A         104        OUT  MSKB         ;MASK OUT THE BIT.
300C E603         105        ANI  03H          ;RE-ISOLATE CHANNEL # AND
300E 07           106        RLC               ;MULTIPLY BY 2.
300F 217C30       107        LXI  H,BTAB       ;GET BRANCH TABLE ADDRESS
3012 85           108        ADD  L            ;AND INDEX INTO TABLE BY
3013 6F           109        MOV L,A           ;TWICE THE CHANNEL #.
3014 D21830       110        JNC  BBB          ;COMPLETE 16 BIT ADDRESS
3017 24           111        INR  H            ;IF NECESSARY
3018 5E           112 BBB:   MOV E,M           ;USING INDEXED TABLE
3019 23           113        INX  H            ;POINTER, ASSEMBLE
301A 56           114        MOV D,M           ;BRANCH ADDRESS AND
301B EB           115        XCHG              ;MOVE IT INTO H,L.
301C D30C         116        OUT  CLBP         ;CLEAR BYTE POINTER.
301E F3           117        DI                ;DISABLE INTERRUPTS
301F E9           118        PCHL              ;BRANCH TO CODE SEGMENT
                  119                          ;FOR SELECTED CHANNEL.
                  120 ;
                  121 ;
                  127 ;
                  128 ;
3020 E1           129 CH0:  POP  H            ;ADDRESS AND WORD COUNT
3021 7E           130       MOV A,M           ;PARAMETERS ARE
3022 D300         131       OUT  ADR0         ;FETCHED FROM CALLING
3024 23           132       INX  H            ;PROGRAM AND OUTPUT
3025 7E           133       MOV A,M           ;TO CHANNEL 0.
3026 D300         134       OUT  ADR0
3028 23           135       INX  H
3029 7E           136       MOV A,M
302A D301         137       OUT  WCT0
302C 23           138       INX  H
```

Figure 21. General Purpose SETUP Routine.

```
LOC    OBJ      SEQ        SOURCE STATEMENT
302D 7E         139        MOV A,M
302E D301       140        OUT WCT0
3030 23         141        INX H
3031 FB         142        EI
3032 3E00       143        MVI A,0
3034 D30A       144        OUT MSKB          ;CLEAR #0 MASK.
3036 E9         145        PCHL              ;BRANCH TO CALLER.
                146 ;
                147 ;
3037 E1         148 CH1:   POP H             ;ADDRESS AND WORD COUNT
3038 7E         149        MOV A,M           ;PARAMETERS ARE
3039 D302       150        OUT ADR1          ;FETCHED FROM CALLING
303B 23         151        INX H             ;PROGRAM AND OUTPUT
303C 7E         152        MOV A,M           ;TO CHANNEL 1.
303D D302       153        OUT ADR1
303F 23         154        INX H
3040 7E         155        MOV A,M
3041 D303       156        OUT WCT1
3043 23         157        INX H
3044 7E         158        MOV A,M
3045 D303       159        OUT WCT1
3047 23         160        INX H
3048 FB         161        EI
3049 3E01       162        MVI A,01H
304B D30A       163        OUT MSKB          ;CLEAR #1 MASK.
304D E9         164        PCHL              ;BRANCH TO CALLER.
                165 ;
                166 ;
                167 ;
                171 ;
                172 ;
                173 ;
304E E1         174 CH2:   POP H             ;ADDRESS AND WORD COUNT
304E 7E         175        MOV A,M           ;PARAMETERS ARE
3050 D304       176        OUT ADR2          ;FETCHED FROM CALLING
3052 23         177        INX H             ;PROGRAM AND OUTPUT
3053 7E         178        MOV A,M           ;TO CHANNEL 2.
3054 D304       179        OUT ADR2
3056 23         180        INX H
3057 7E         181        MOV A,M
3058 D305       182        OUT WCT2
305A 23         183        INX H
305B 7E         184        MOV A,M
305C D305       185        OUT WCT2
305E 23         186        INX H
305F FB         187        EI
3060 3E02       188        MVI A,02H
3062 D30A       189        OUT MSKB          ;CLEAR #2 MASK.
3064 E9         190        PCHL              ;BRANCH TO CALLER.
                191 ;
                192 ;
                193 ;
3065 E1         194 CH3:   POP H             ;ADDRESS AND WORD COUNT
3066 7E         195        MOV A,M           ;PARAMETERS ARE
3067 D306       196        OUT ADR3          ;FETCHED FROM CALLING
3069 23         197        INX H             ;PROGRAM AND OUTPUT
306A 7E         198        MOV A,M           ;TO CHANNEL 3.
306B D306       199        OUT ADR3
306D 23         200        INX H
```

Figure 21. General Purpose SETUP Routine. (Cont.)

```
LOC    OBJ      SEQ        SOURCE STATEMENT
306E 7E         201        MOV A,M
306F D307       202        OUT WCT3
3071 23         203        INX H
3072 7E         204        MOV A,M
3073 D307       205        OUT WCT3
3075 23         206        INX H
3076 FB         207        EI
3077 3E03       208        MVI A,03H
3079 D30A       209        OUT MSKB          ;CLEAR #3 MASK.
307B E9         210        PCHL              ;BRANCH TO CALLER.
                211 ;
                212 ;
                213 ;
307C 2030       214 BTAB:  DW  CH0           ;BRANCH TABLE
307E 3730       215        DW  CH1
3080 4E30       216        DW  CH2
3082 6530       217        DW  CH3
                218 ;
                219 ;
```

Figure 21. General Purpose SETUP Routine. (Cont.)

Hardware Configuration

Figure 22 shows the interconnections for interfacing a single Am9517A to a microprocessor system and an 'LS373. Other 8-bit latches may also be used and Figure 23 shows several other possibilities for both single and double controller configurations. The 'LS373 and 'LS374 parts offer small 20-pin packages and low power. Figure 24 is a block diagram of the general Am9517A expansion scheme.

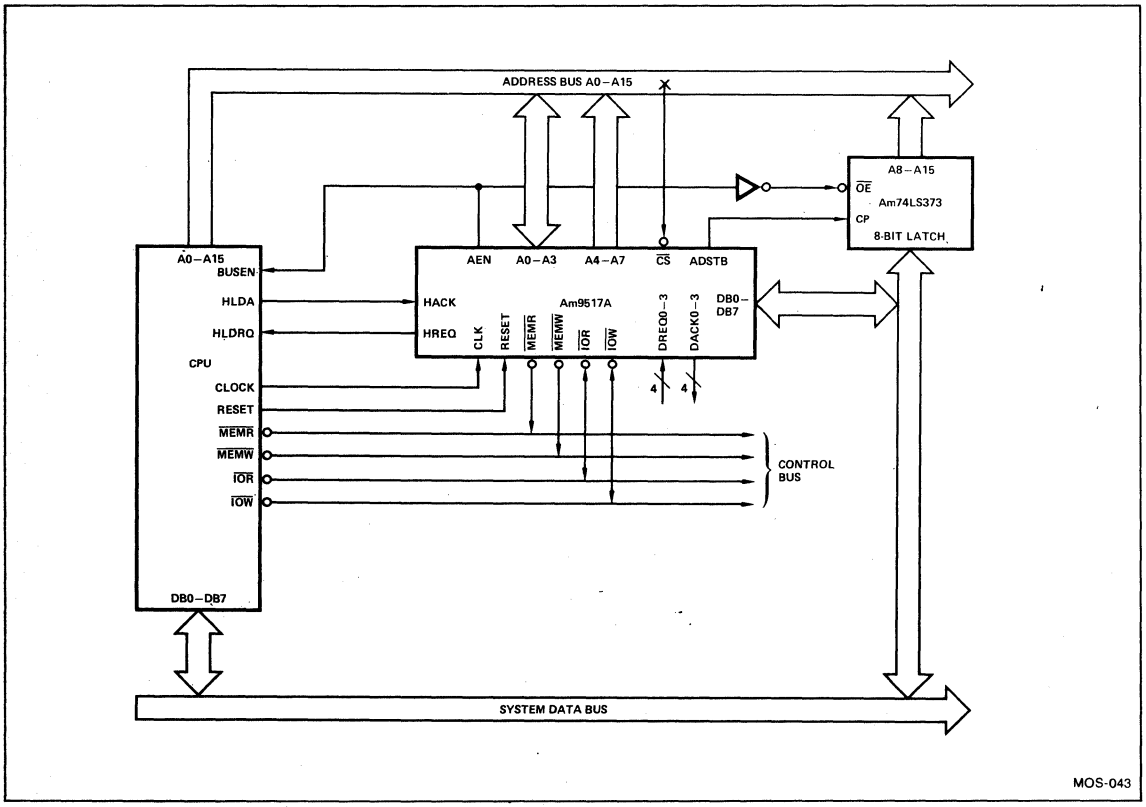


Figure 22. Basic DMA Configuration.

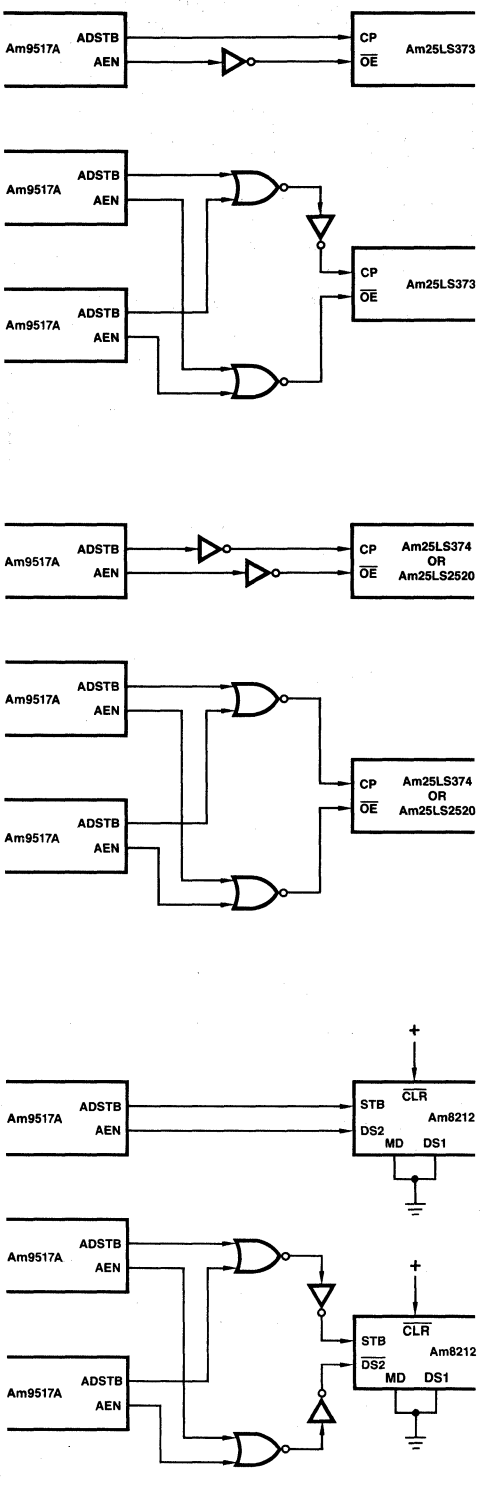

Am9517A
ADSTB
AEN
CP
OE
Am25LS373
Am25LS374
OR
Am25LS2520
STB
CLR
DS2
MD
DS1
Am8212
+
MOS-064

Figure 23.

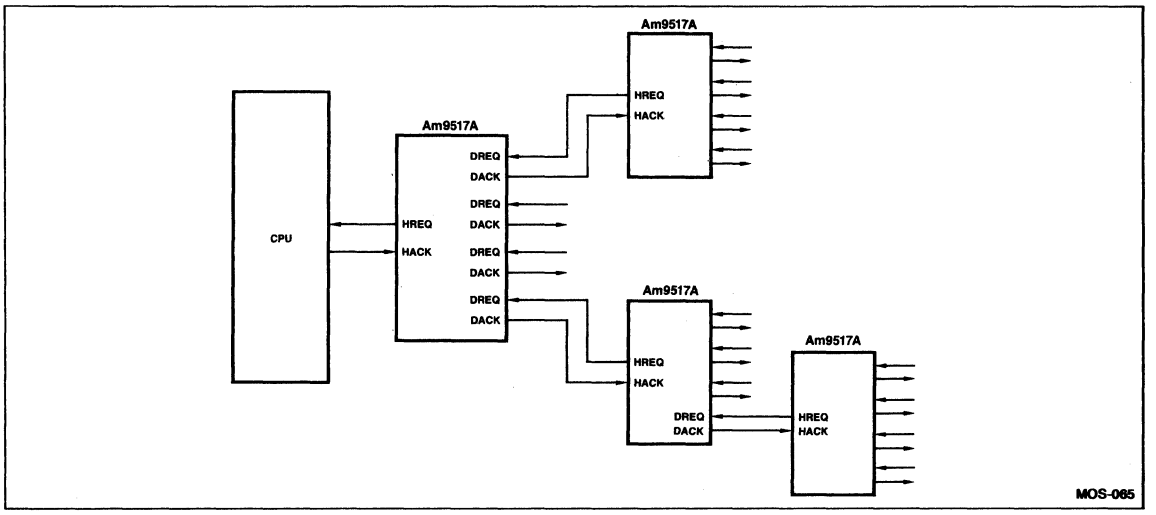

Am9517A
HREQ
HACK
Am9517A
DREQ
DACK
CPU
HREQ
HACK
DREQ
DACK
DREQ
DACK
DREQ
DACK
Am9517A
HREQ
HACK
DREQ
DACK
Am9517A
HREQ
HACK
MOS-065

Figure 24.

Advanced Micro Devices

Designing Interrupt Systems With the Am9519 Universal Interrupt Controller

By Joseph H. Kroeger

Copyright © 1978 by Advanced Micro Devices, Inc.

Advanced Micro Devices cannot assume responsibility for use of any circuitry described other than circuitry entirely embodied in an Advanced Micro Devices' product.

AM-PUB071

TABLE OF CONTENTS

I INTRODUCTION
- General 11-24
- Features 11-24

II HARDWARE INTERFACE
- Block Diagram 11-25
- Interface Signal Description 11-25
- Interface Considerations 11-26
- IREQ Timing 11-27
- Power Supply 11-28

III OPERATING DESCRIPTION
- Reset 11-29
- Register Description 11-29
- Information Transfers 11-31
- Operating Options 11-31
- Operating Sequence 11-33

IV COMMAND DESCRIPTIONS 11-34

V SYSTEM INTERFACE
- Expansion 11-37
- Initialization and Support 11-38

INTRODUCTION

General

Processors exist as tools for the implementation of information system transfer functions. All useful processor systems include at least one peripheral device in order to communicate with the user of the system. The processor not only manipulates information once it is in the system, but also handles the transfer of information to and from the user via the peripherals. Often several devices are integral parts of the overall system. All peripherals must be serviced in one way or another by the system processor. The basic parameters that influence the design of peripheral servicing algorithms are the frequency of service required, the service latency allowed and the service duty cycle of the devices.

There are two general methods used to initiate and coordinate this activity: Program controlled service and Interrupt driven service. In program controlled transfers, the processor schedules all peripheral events; an Interrupt driven system, on the other hand, allows modification of the system activities by external devices.

With no interrupt capability, processors must depend on software polling techniques to service peripheral devices. As the number of such devices grows and/or as the complexity of service increases, the polling program becomes very time consuming and the overhead devoted to polling becomes a significant fraction of the available processing resource. When this limits system performance, the use of interrupts can often provide substantial improvement.

Interrupts are used to enhance processor system throughput and response time by minimizing or eliminating the need for software polling procedures. Interrupts are hardware mechanisms that allow devices external to the processor to asynchronously modify the instruction sequence of the processor program being executed. An elementary single interrupt could be used simply to alert the processor to the fact that some kind of service is desired and thus to initiate a polling routine. More complex systems may have multiple interrupts and vectoring protocols which can be used to further improve performance and eliminate all polling requirements. Vectoring allows direct identification of the interrupting device and its associated service routine.

Figure 1 illustrates the essential functioning of a typical interrupt procedure. As the main program is executing instructions, an external interrupt arrives, in this example during instruction M+2.The processor completes M+2 and then, instead of executing M+3, it performs some kind of interrupt acknowledge procedure, often involving execution of an additional interrupt instruction. The result will usually be that the address of instruction M+3 is saved for future reference, and the location of instruction N is determined. The processor then proceeds to execute the interrupt service routine starting with instruction N. The service routine may save, and later restore, the processor status as well as perform tasks requested by the interrupting device. The last instruction in the routine (N+K) directs the processor to resume the main program at instruction M+3.

Notice that the presence of the hardware interrupt has caused a modification of the sequence of instruction execution; an additional block of instructions has been inserted in the main program. Interrupts provide the system designer with a significant capability that can help optimize his cost/performance tradeoffs.

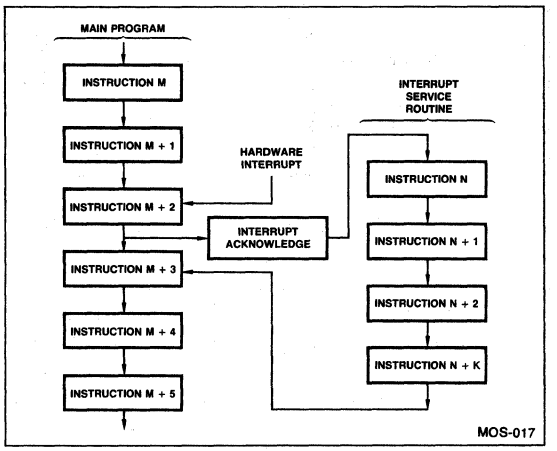


Figure 1. Basic Interrupt Procedure.

Features

The Am9519 Universal Interrupt Controller is a processor support device designed to enhance the interrupt handling capability of a wide variety of processors. A single Am9519 manages the masking, priority resolution and vectoring of up to eight interrupts. It may be easily expanded by the addition of other Am9519 chips to handle a nearly unlimited set of interrupt inputs. It offers many programmable operating options to improve both the efficiency and versatility of its host system operations. The Am9519 is well adapted to a wide range of uses including small, simple, as well as large, sophisticated, interrupt systems.

The Am9519 provides any mix of one, two, three and four byte responses to the host processor during the interrupt acknowledge process. The response bytes are all fully programmable so that any appropriate addressing, vectoring, instruction or other message protocol may be used. Contention among multiple interrupts is managed internally using either fixed or rotating priority resolution circuitry. The direct vectoring capability of the Am9519 may be bypassed using the polled mode option.

An internal mask register permits individual interrupts to be disabled. It may be loaded in parallel by the host processor with any bit pattern, or mask bits may be individually controlled. The interrupt inputs use "pulse-catching" circuitry so that an external register is not needed to capture interrupt pulses. Narrow noise pulses, however, are ignored. The interrupt polarity may be selected as either active-high or active-low.

Another important feature of the Am9519 is its ability to generate software interrupts. The host processor can set interrupt requests under program control, thus permitting hardware to resolve the priority of software tasks. This is often a powerful system asset, especially for sophisticated operating software, as well as an aid for system testing, diagnostic, debugging and maintenance procedures.

The Am9519 is implemented with AMD's LINOX n-channel silicon gate MOS technology. This process features low profile structures, triple ion-implantation, both depletion and enhancement transistors, and small, low capacitance, low power, high speed circuitry. The chip contains 4,400 transistors within a total chip area of 28,765 square mils. It is packaged in a standard 28-pin dual in-line package.

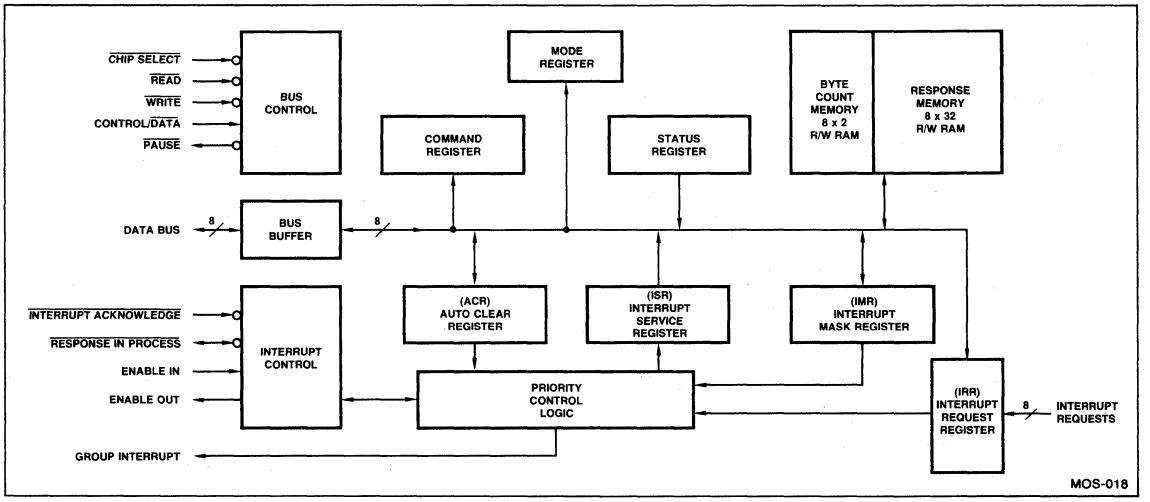


Figure 2. Am9519 Block Diagram.

HARDWARE INTERFACE

Block Diagram

The block diagram of the Am9519 shown in Figure 2 indicates the interface signals and the basic internal information flow. Interrupt Request inputs are captured and latched in the Interrupt Request register. Any requests not masked by the Interrupt Mask register will cause a Group Interrupt output to the host processor if the unit is enabled. When the processor is ready to handle the interrupt it issues an Interrupt Acknowledge pulse which causes (a) the priority of pending interrupts to be resolved and (b) a byte from the response memory associated with the highest priority interrupt to be transferred to the data bus. The transfer of additional response bytes is controlled by additional Interrupt Acknowledge signals. Other interrupt management functions are controlled by the Auto Clear register, the Interrupt Service register and the Mode register. Control of the Am9519 is exercised by the host processor using the Command register. The Status register reports on the internal condition of the part.

The Am9519 is addressed by the host processor as two distinct ports: a control port and a data port. The control port provides direct access to the Status register and the Command register. The data port is used to communicate with all other internal locations.

Interface Signal Description

Figure 3 summarizes the interface signals. Figure 4 shows the interface signal pin assignments.

Data Bus (DB)

The eight three-state bidirectional data bus lines are used to transfer information between the Am9519 and the system data bus. The direction of information flow is controlled by the $\overline{CS}$, $\overline{RD}$, $\overline{WR}$ and $\overline{IACK}$ input signals. Data and command information are written into the device; status, data and response information are output by it.

Description	Abbreviation	Type	Pins
+5 Volts	VCC	Power	1
Ground	VSS	Power	1
Data Bus	DB	I/O	8
Response In Process	RIP	I/O	1
Interrupt Request	IREQ	Input	8
Chip Select	CS	Input	1
Read	RD	Input	1
Write	WR	Input	1
Control/Data	C/D	Input	1
Interrupt Acknowledge	IACK	Input	1
Enable In	EI	Input	1
Enable Out	EO	Output	1
Group Interrupt	GINT	Output	1
Pause	PAUSE	Output	1

Figure 3. Am9519 Interface Signal Summary.

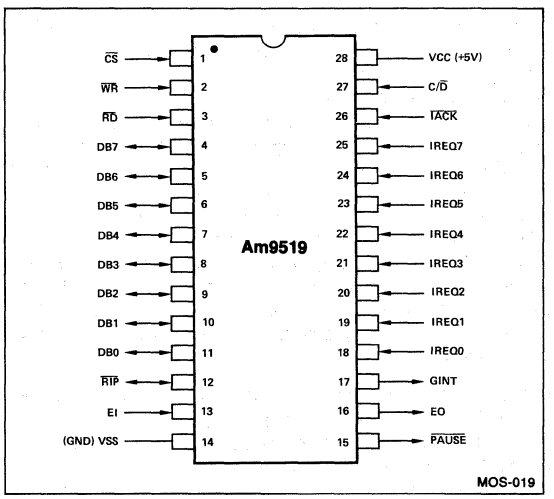


Figure 4. Connection Diagram.

Chip Select ($\overline{CS}$)

The Chip Select input is an active low signal used to condition the chip for read and write operations on the data bus; Read/Write transfers will not take place unless the $\overline{CS}$ input is low. Chip Select does not condition Interrupt Acknowledge operations. Chip Select is usually derived by decoding an address output by the host processor; the negative-true polarity matches outputs from typical decoder circuits.

Read ($\overline{RD}$)

The Read input is an active low signal conditioned by Chip Select that indicates information is to be transferred from the Am9519 to the data bus. Read is usually a timed pulse issued by the host processor.

Write ($\overline{WR}$)

The Write input is an active low signal conditioned by Chip Select that indicates information is to be transferred from the data bus to the Am9519. Write is usually a timed pulse issued by the host processor.

Control/Data (C/$\overline{D}$)

The Control/Data input acts as the port address line and is used to select source and destination locations for read and write transfers. Data transfers (C/$\overline{D}$=0) are made to or from preselected internal memory or register locations. Control transfers (C/$\overline{D}$=1) write into the command register or read from the status register.

Interrupt Request (IREQ)

The eight Interrupt Request inputs are used by external devices to indicate that service is desired. The Interrupt Request Register associated with the inputs uses asynchronous pulse-catching circuitry to latch any active requests that occur. The input polarity may be programmed to capture either positive-going or negative-going transitions. Reset selects the active low option.

Response In Process ($\overline{RIP}$)

The Response In Process signal is a bidirectional line designed to be used when two or more Am9519 circuits are connected together. $\overline{RIP}$ is used to prevent new higher priority interrupts from interferring with an Interrupt Acknowledge process that is underway. An Am9519 that is responding to a selected interrupt will treat $\overline{RIP}$ as an output and will hold the signal low until the acknowledge response is complete. An Am9519 without a selected interrupt will treat $\overline{RIP}$ as an input and will ignore $\overline{IACK}$ pulses as long as $\overline{RIP}$ is low. The $\overline{RIP}$ lines from multiple Am9519 circuits may be wired directly together. $\overline{RIP}$ is an open drain signal, and requires an external pullup resistor to VCC in order to establish the logic high level.

Group Interrupt (GINT)

When active, the Group Interrupt output indicates that at least one bit is set in the Interrupt Request Register (IRR) which is not masked by the Interrupt Mask Register or the Interrupt Service Register. GINT is used to notify the host processor that service is desired. It may be programmed for either active high or active low polarity in order to simplify the interface with the host circuitry. Reset selects active low. When active high is selected the output is a standard two-state buffer configuration. When active low is selected the output is open drain and requires an external pullup resistor to VCC in order to establish the logic high level. The open drain configuration is useful for wired-or connections in systems with more than one Am9519.

Interrupt Acknowledge ($\overline{IACK}$)

The Interrupt Acknowledge input is an active low signal generated by the host processor and used to request interrupt response information. One response byte will be transferred by the Am9519 for each $\overline{IACK}$ pulse received and up to four bytes may be transferred during each interrupt acknowledge sequence. The first $\overline{IACK}$ pulse following a GINT output also initiates the internal selection of the highest priority unmasked interrupt.

Many processors provide interrupt acknowledge signals directly, including the 8085, the 8080A and the 2650. For others, such as the Z80 and the 6800, it can be generated quite easily with simple gating.

Pause

The Pause output is an active low signal used during $\overline{IACK}$ cycles to indicate that the Am9519 has not completed the data bus transfer operation presently underway. The $\overline{IACK}$ pulse should be extended by the host processor at least until the $\overline{PAUSE}$ output goes high. The width of active $\overline{PAUSE}$ pulses is a function of several variables; it will be quite short in some systems and longer in others. $\overline{PAUSE}$ is an open drain output and requires an external pullup resistor to establish the high logic level. $\overline{PAUSE}$ signals should be wired together in multiple chip interrupt systems.

Enable In (EI)

The Enable In input is an active high signal used to implement a "daisy-chain" expansion capability with other Am9519 chips. EI may also be used as a hardware disable/enable input for the interrupt system. When EI is low, $\overline{IACK}$ inputs to the chip are ignored. Internally, a relatively high impedance resistor is connected between EI and VCC so that an unused EI requires no external pullup resistor.

Enable Out (EO)

The Enable Out output is an active high signal used to implement a "daisy-chain" expansion capability with other Am9519 chips. When the $\overline{IACK}$ input goes low, EO goes low until EI goes high and the chip determines that no unmasked request is pending. EO is a two-state output with relatively modest drive capability.

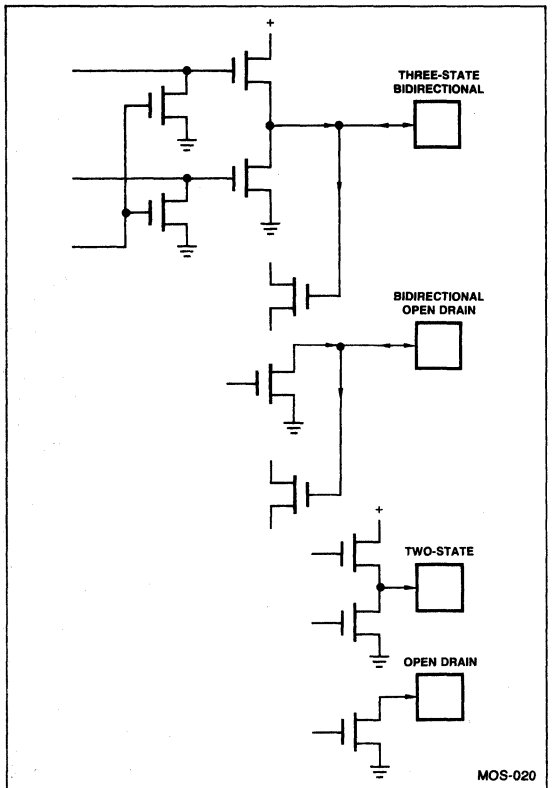
Interface Considerations

All of the input and output signals for the Am9519 are specified with logic levels identical to those of standard TTL circuits. The worst-case input logic levels are 2.0V high and 0.8V low. Except for the open drain signals, the worst-case output logic levels are 2.4V high and 0.4V low. Thus, for TTL interfacing, the normal worst-case noise immunity of at least 400mV is maintained. The logic level specifications take into account all combinations of the three variables that affect the logic level threshold: ambient temperature, supply voltage and processing parameters. A change in any of these toward nominal values will improve the actual operating margins.

The $\overline{PAUSE}$ and $\overline{RIP}$ outputs are open drain with no active pullup transistors; their output high levels are established by the external circuitry. The GINT output, when programmed for active low polarity ($\overline{GINT}$), is also an open drain output that does not control its output high level.

All of the output buffers except EO and the open drain outputs can source at least 200μA worst-case and can sink at least 3.2mA worst-case while maintaining TTL output logic levels. EO normally only drives EI of another Am9519 chip and is specified with less drive capability in order to improve the

priority resolution speed in multi-chip interrupt systems. The open drain outputs all sink at least 3.2mA as the other outputs do. Current sourcing for the open drain outputs is determined by the external circuitry. Figure 5 summarizes the types of outputs on the Am9519.



Signal	Output Description
Data Bus (DB0-DB7)	Bidirectional, Three-State
Response In Process (RIP)	Bidirectional, Open-drain
Pause	Open-drain
Group Interrupt (GINT)	{ Two-State when active high Open-Drain when active low
Enable Out (EO)	Two-state, low drive

Figure 5. Am9519 Output Buffer Summary and Circuitry.

Unprotected open gate inputs of high quality MOS transistors exhibit very high resistances on the order of 10^{14} ohms. It is easy in many circumstances for charge to enter the gate node of such an input faster than it can be discharged and consequently for the gate voltage to rise high enough to break down the oxides and destroy the transistor. All inputs to the Am9519 include protection networks to help prevent damaging accumulations of static charge. The protection circuitry is designed to slow the transitions of incoming current surges and to provide low impedance discharge paths for voltages beyond the normal operating levels. Please note, however, that input energy levels can nonetheless be too high to be successfully absorbed. Conventional design, storage, and handling precautions should be observed so that the protection networks themselves are not overstressed.

Within the limits of normal operation, the input protection circuitry is inactive and may be modeled as a lumped series RC as shown in Figure 6. The functionally active input connection during normal operation is the gate of an MOS transistor. Except for EI, no active sources or drains are connected to the inputs so that neither transient nor steady-state currents are impressed on the driving signals by the Am9519 other than the charging or discharging of the input capacitance and the accumulated leakage associated with the protection network and the input circuit. Lumped input capacitances are usually around 6pF and leakage currents are usually less than 1μA.

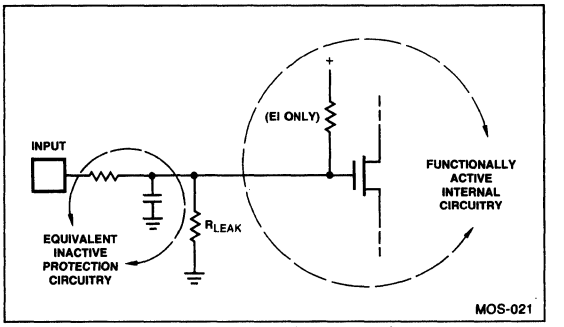


Figure 6. Input Circuitry.

Fanout from the driving circuitry into the Am9519 inputs will generally be limited by transition time considerations rather than DC current limitations when the loading is dominated by MOS circuits like the Am9519. In an operating environment, all inputs should be terminated so they do not float and accumulate stray static charges. Unused inputs should be tied directly to Ground or to VCC, as appropriate. An input in use will have some type of logic output driving it and termination during operation will not be a problem. Where inputs are driven from logic external to the card containing this chip, however, on-board termination should be provided to protect the chip when the board is unplugged and the input would otherwise float. A pull-up resistor or a simple inverter or gate will suffice.

IREQ Timing

The circuitry at the IREQ inputs is quite straightforward and is illustrated in Figure 7. Inverters 1 and 2 buffer the input and shift the logic voltages to the somewhat wider swing used internally. The exclusive-or gate is used to select the sense of the active transition edge that will set the IRR. Mode register bit M4 is used directly for control of the exclusive-or gate. The selected interface edge will always produce a negative going transition at output 3. Inverters 4, 5, 6, 7 and 8 form a delay chain. Nor gate 9 has three inputs and the IRR bit will be set when all three inputs to 9 are low. As shown in the timing diagram of Figure 8, the input to gate 9 from inverter 8 is normally low when there is no active IREQ signal at the interface. When a transition occurs, the output of gate 3 will go low and only the signal from inverter 5 prevents the immediate setting of the IRR bit. As shown in the left portion of the timing diagram, if the output from 3 has returned high before the output from 5 goes low, the IREQ transition will be ignored and the IRR bit will not be set. On the other hand, the right side of the timing diagram shows that if the active IREQ input is present long enough, then the output from both 3 and 5 will become low at the same time, and output 9 will go high. Output 8 is used to turn off Nor gate 9 after the IRR bit is set.

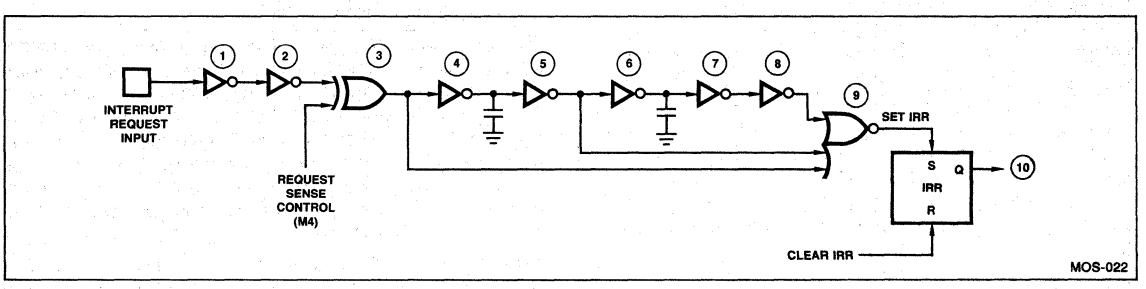


Figure 7. Interrupt Request Logic.

In summary, the input circuitry for the IREQ signals provides these characteristics:

1. Polarity for IREQ inputs is controlled;
2. Narrow IREQ pulses are ignored;
3. Wide IREQ pulses are captured;
4. Transitions to active levels are captured just once;
5. New transitions are required to generate new interrupts.

The IRR thus acts in a "pulse-catching" mode with respect to the IREQ inputs. Figure 9 shows the types of IREQ waveforms that will be recognized and latched by the IRR. Note that a transition to a level may be used although only a pulse is required; it is not necessary to maintain an IREQ input active level. Further, a continuously active level on IREQ will not cause a new interrupt each time IRR is cleared. There must be a new active transition on IREQ after IRR is cleared in order to generate a new interrupt. An active level must go inactive for a specific interval before its new active edge will be recognized.

To minimize noise sensitivity, all active IREQ pulses narrower than a specific value will be ignored by the IRR. To maintain the pulse-catching characteristics, all active IREQ pulses wider than the specified data sheet minimum will be captured by the IRR. The results for intermediate pulse widths will depend on characteristics of the particular part being used and its operating conditions, especially temperature.

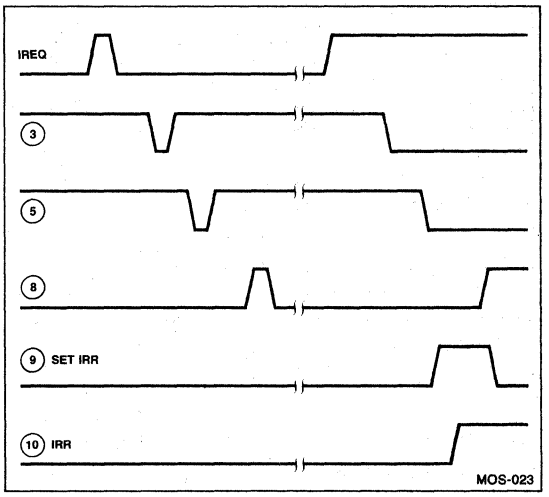


Figure 8. IREQ Internal Timing.

Power Supply

The Am9519 requires only a single +5V power supply. The commercial temperature range parts have a voltage tolerance of ±5%; the military temperature range tolerance is ±10%. Maximum supply currents are specified in the data sheet at the high end of the voltage tolerance and the low end of the temperature range. In addition, the current specifications take into account the worst-case distribution of processing parameters that may be encountered during the manufacturing life of the product. Typical supply current values, on the other hand, are specified for a nominal supply of +5.0 volts, nominal ambient temperature of 25°C, and nominal processing parameters. Supply current always decreases with increasing ambient temperature; thermal run-away is not a problem.

Although supply current will vary from part to part, a given unit at a given operating temperature will exhibit a nearly constant power drain. There is no functional operating region that will cause more than a few percent change in the supply current. Decoupling of VCC, then, is straightforward and will generally be used simply to isolate the Am9519 from external VCC noise.

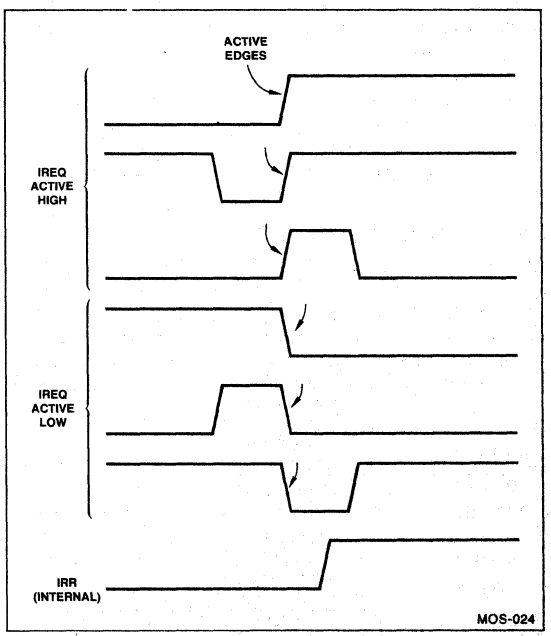


Figure 9. IREQ Waveforms.

OPERATING DESCRIPTION

Reset

The Am9519 does not include an external hardware reset input. The reset function is accomplished either by software command or automatically during power-up. The reset may be initiated by the host processor at any time simply by writing all zeros into the command port. Power-up reset circuitry is internally triggered by the rising VCC voltage when a predetermined threshold is reached, generating a brief internal reset pulse.

The response memory and byte count registers are not affected by resets. Their content after power-up are unpredictable and if they are to be used, they must first be initialized by the host processor. A software reset does not disturb previous response memory and byte count contents.

The Interrupt Mask register is set to all ones by a reset, thus disabling recognition of interrupts by the chip. The Status register continues to reflect the internal condition of the chip and is not otherwise directly affected by a reset. All other registers are cleared to all zeros by a reset. The polarities of the Mode register control bits are assigned to provide a reasonable operating option environment when cleared by a reset.

Register Description

The Am9519 uses several control and operation registers plus the response memory to perform and manage its many functions. Figure 10 lists these elements and summarizes their size and number.

Description	Abbreviation	Bit Size	Quantity
Interrupt Request Register	IRR	8	1
Interrupt Service Register	ISR	8	1
Interrupt Mask Register	IMR	8	1
Auto Clear Register	ACR	8	1
Status Register	–	8	1
Mode Register	–	8	1
Command Register	–	8	1
Byte Count	–	2	8
Response Memory	–	32	8

Figure 10. Am9519 Register and Memory Summary.

Interrupt Request Register (IRR)

The IRR is eight bits long and is used to recognize and store active transitions on the eight Interrupt Request input lines. A bit in the IRR is set whenever the corresponding IREQ input makes an inactive-to-active transition and meets the minimum active pulse width requirements. IRR bits may also be set by the host processor under program control using two types of commands. This capability allows software initiated interrupts, and is a significant tool for system testing and for sophisticated software designs.

All IRR bits are cleared by a reset. Individual IRR bits are cleared automatically when their interrupts are acknowledged by the host processor. Four types of commands, in addition to reset, allow the host program to clear IRR bits.

The IRR may be read onto the data bus by preselecting it in Mode register bits M5 and M6, followed by a read operation at the data port.

Interrupt Service Register (ISR)

The ISR is eight bits long and is used to store the acknowledge status of individual interrupts. When an $\overline{IACK}$ pulse arrives, the Am9519 selects the highest priority request that is pending, then clears the associated IRR bit and sets the associated ISR bit. When the ISR bit is programmed for automatic clearing, it is reset by the internal hardware before the end of the acknowledge sequence. When the ISR bit is not programmed for automatic clearing, it must be reset by command from the host processor.

Internally, the Am9519 uses the ISR to erect a "masking fence". When an ISR bit is set and fixed priority mode is selected, only requests of higher priority will cause a new GINT output. Thus, requests from lower priority interrupts (and from new requests associated with the set ISR bit) will be fenced out and ignored until the ISR bit is cleared. In the rotating priority mode, all requests are fenced by an ISR bit that is set, and no new GINT outputs will be generated until the ISR is cleared. When auto clear is specified, no fence is erected since the ISR bit is cleared.

If an unmasked interrupt arrives from a device of higher priority than the current ISR, GINT will go true and the host processor will be interrupted if its interrupt input is enabled. When the new interrupt is acknowledged, the associated higher priority ISR bit is set and the fence moves up to the new level. When the new ISR bit is cleared, the fence will then fall back to the previous ISR level.

The ISR may be read onto the data bus by preselecting it in Mode register bits M5 and M6, followed by a read operation at the data port.

Interrupt Mask Register (IMR)

The IMR is eight bits long and is used to enable/disable the processing of individual interrupts. Only unmasked IRR bits can cause a Group Interrupt to be generated. The IMR does not otherwise affect the operation of the IRR. An IRR bit that is set while masked will cause a GINT when its IMR bit is cleared.

All eight IMR bits may be set, cleared, read or loaded in parallel by the host processor. In addition, individual IMR bits may be set or cleared by command. This allows a control routine to directly enable and disable an individual interrupt without disturbing the other mask bits and without knowledge of their state or the system context.

The IMR polarity is active high for masking; a zero enables the interrupt and a one disables it. The power-on reset and the software reset cause all IMR bits to be set, thus disabling all requests.

Auto Clear Register (ACR)

The ACR is eight bits long and specifies the automatic clearing option for each of the ISR bits. When an auto clear bit is set, the corresponding ISR bit that has been set in an $\overline{IACK}$ cycle is cleared by the internal hardware before the end of the $\overline{IACK}$ sequence. When an auto clear bit is not set, the corresponding ISR bit that has been set in an $\overline{IACK}$ cycle is cleared by command from the host processor.

The auto clear option, when selected, provides two concomitant functional effects. First, it eliminates the need for the associated interrupt service routine to issue a command to clear the ISR bit. Secondly, it eliminates the masking fence that would otherwise have been erected, allowing lower priority interrupts to cause a new GINT output.

The ACR is loaded in parallel from the data bus by issuing the ACR load preselect command followed by a write into the data port. The ACR may be read onto the data bus by preselecting it in Mode register bits M5 and M6, followed by a read operation at the data port.

Status Register

The Status Register is eight bits long and contains information describing the internal state of the Am9519 chip. The Status register is read directly by executing a read operation at the control port. Figure 11 shows the Status bit assignments.

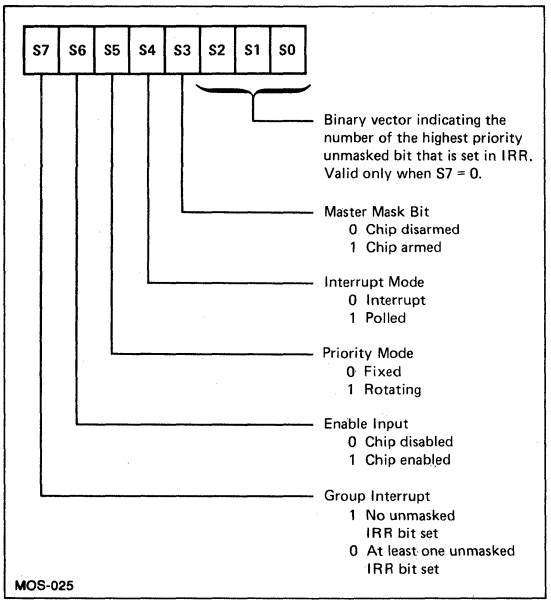


Figure 11. Status Register Bit Assignments.

The high order status bit, S7, reflects the information state of the Group Interrupt signal. Note that the polarity definition of S7 is independent of the defined polarity of GINT (Mode bit M3). Bit S7 remains valid when GINT is disabled by the polled mode option, thus permitting the host processor to check for "interrupts" by reading the Status register.

Status bit S6 reflects the state of the Enable In input signal and is used to indicate, in a multiple chip interrupt structure, which chips in the chain are disabled. When S6 is high, the chip can generate a GINT output and operation of its EO signal proceeds. When S6 is low, no GINT will be generated and EO will be forced low.

Status bit S5 reflects the state of the Priority Mode option, as specified by bit M0 of the Mode register. When S5 is high, rotating priority has been selected. When S5 is low, fixed priority has been selected.

Status bit S4 reflects the state of the Interrupt Mode option, as specified by bit M2 of the Mode register. When S4 is high, the polled mode has been selected and GINT disabled. When S4 is low, the interrupt mode has been selected.

Status bit S3 reflects the state of the Master Mask bit as specified by bit M7 of the Mode register. When S3 is low, the chip has been disarmed and IRR bits that are set will not generate GINT outputs. When S3 is high, the chip has been armed and interrupts can occur.

Status bits S2, S1 and S0 form a three bit field indicating the encoded binary number of the highest priority unmasked bit that is set in the IRR. This field should be considered invalid except when bit S7 of the Status register is low, indicating that at least one unmasked interrupt request is present. The binary coding of the field corresponds to the zero through seven numbering of the IREQ inputs. When more than one unmasked IRR bit is set, the S2, S1, S0 field will indicate the one unfenced request that is the highest priority as determined by the priority mode being used. Thus, the number of the dominant interrupt after all masking, fencing and priority resolution, is encoded into the Status register. This field is quite useful in the polled mode since it can act as a psuedo-vector for the host processor software.

Command Register

The Command Register is eight bits long and is used to store the most recently entered command. It is loaded directly from the data bus by executing a write operation at the control port. Depending on the specific command opcode that is entered, an immediate internal activity may be initiated or the part may be preconditioned for subsequent data bus transfers. The "Command Description" section of this note explains each command operation. The commands are summarized in Figure 17.

Mode Register

The Mode register is eight bits long and controls the operating modes and options of the Am9519. Figure 12 shows the bit assignments for the Mode register. No single command or interface operation will load all bits of the Mode register in parallel. The five low order bits (M0 through M4) are loaded in parallel directly from the command register. Mode bits M5, M6, and M7 are controlled by separate commands. The Mode register cannot be read out on the data bus. The data in Mode bits M0, M2, and M7 are available as part of the Status register. The Mode register is cleared by a software reset or a power-up reset. The "Operating Options" section of this note describes the detailed functions associated with each Mode bit.

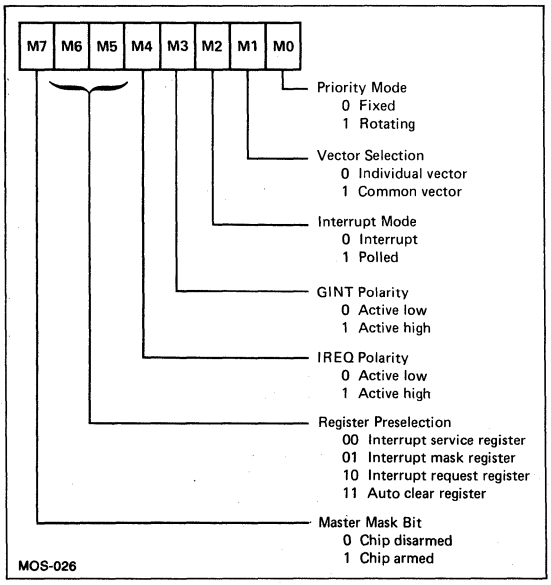


Figure 12. Mode Register Bit Assignments.

Information Transfers

Figure 13 summarizes the control signal configurations for all information transfers on the Am9519 data bus. The interface control logic assumes the following conventions:

1. $\overline{RD}$ and $\overline{WR}$ are never active at the same time.
2. $\overline{RD}$, $\overline{WR}$ and C/$\overline{D}$ are ignored unless $\overline{CS}$ is low.

Control Input					Data Bus Operation
$\overline{CS}$	C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	$\overline{IACK}$	
0	0	0	1	1	Transfer contents of data register specified by Mode bits M5, M6 to data bus.
0	0	1	0	1	Transfer contents of data bus to data register specified by Command register.
0	1	0	1	1	Transfer contents of Status register to data bus.
0	1	1	0	1	Transfer contents of data bus to Command register.
1	X	X	X	0	Transfer contents of selected response memory location to data bus.
1	X	X	X	1	No information transferred; data bus outputs off.

Figure 13. Summary of Data Bus Transfers.

When $\overline{IACK}$ is low, internal logic disables the $\overline{CS}$ input. This prevents signals on the address bus from inadvertently selecting the chip.

The host processor may read the Status register directly by simply performing a read operation with the control port selected. When a read is executed at the data port, the information transferred will be the contents of the ISR, IMR, IRR or ACR, depending on the state of Mode register bits M5 and M6.

The host processor may write directly into the command register by simply performing a write operation with the control port selected. When a write is executed into the data port, the contents of the data bus will be transferred to the ACR, IMR or response memory, depending on which command preceded the data write. Note that Mode bits M5 and M6 do not preselect the location for data write operations; only a command can do so.

When the response memory preselect command is issued, it should be followed by an appropriate number of data write operations to load 1, 2, 3, or 4 bytes of response information. If more than four bytes are written, the response memory addressing will "wrap around" and overwrite the information already entered. Response bytes are output by the Am9519 during $\overline{IACK}$ operations in the same order they were entered. Entry of response information into each new level must be preceded by a new response memory preselect command.

Interrupt Acknowledge operations are initiated by the host processor and occur following recognition of a GINT signal from the Am9519. When an $\overline{IACK}$ signal arrives, the interrupt system selects the highest priority unmasked pending interrupt request and then outputs a response byte associated with the selected interrupt. The selection process and the access of the response byte will take a variable amount of time that depends on several parameters, including:

1. the operating temperature,
2. the actual internal logic delays,
3. the number of Am9519 chips cascaded together,
4. the priority level of the interrupt being acknowledged,
5. the Mode register operating options,
6. the byte position within the response sequence.

The worst-case $\overline{IACK}$ pulse widths must be long enough to accomodate the accumulated delays that can occur in large interrupt systems operating in worst-case situations. Yet small systems operating under typical conditions will require only relatively narrow $\overline{IACK}$ pulses. The $\overline{PAUSE}$ output from the Am9519 is designed to provide interactive feedback to the host processor so that the $\overline{IACK}$ pulse width may be adaptively adjusted to meet the requirements of the actual interrupt being processed. $\overline{PAUSE}$ will go low fairly quickly following the falling edge of $\overline{IACK}$, and will return high when $\overline{IACK}$ is no longer required.

During the first $\overline{IACK}$ of a complete acknowledge sequence, the $\overline{PAUSE}$ output remains low until the highest priority interrupt has been selected and the $\overline{RIP}$ output goes low. On subsequent $\overline{IACK}$ pulses for additional responses bytes associated with the same interrupt ($\overline{RIP}$ still low), $\overline{PAUSE}$ will remain high. The Am9519 expects the first $\overline{IACK}$ input to remain low at least until the $\overline{PAUSE}$ output goes high. Subsequent $\overline{IACK}$ inputs should meet the specified input pulse width requirements as called out in the data sheet.

It will normally be convenient for the $\overline{PAUSE}$ signal to provide a "not ready" indication to the host processor which would then stall the Interrupt Acknowledge operation until $\overline{PAUSE}$ goes high. In 8080A/9080A microprocessor systems, $\overline{PAUSE}$ can be used directly in the CPU Ready logic and many other processor systems have similar coordination schemes.

Operating Options

The Mode register bits are used to establish the operating modes and conditions for the many functional features of the Am9519. The Mode register allows the host processor to personalize the interrupt system for the application at hand.

Priority Selection

Bit M0 in the Mode register specifies the priority operating mode for the Am9519. When M0=0, fixed priority is selected and the eight IREQ inputs are assigned a priority based on their physical location at the chip interface. IREQ0 has the highest priority and IREQ7 has the lowest. See Figure 14.

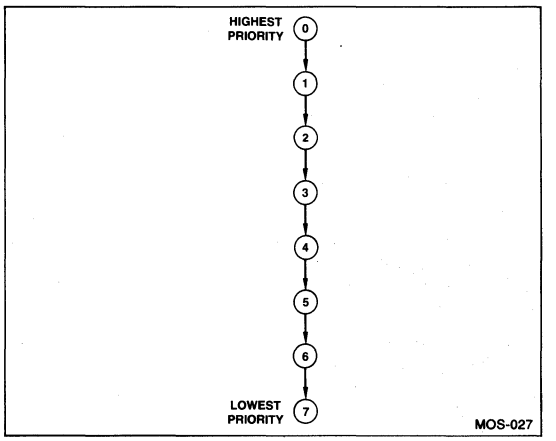


Figure 14. Fixed Priority Mode.

Priority is not resolved until the host processor initiates the interrupt acknowledge sequence. Thus, for example, an IREQ5 input may cause a GINT output to the host, but if an input on IREQ2 arrives before the falling edge of $\overline{IACK}$, then it is IREQ2 that will be selected and serviced. Notice that inherent in the fixed priority structure is the possibility that IREQ5 might never be selected and serviced as long as there are higher priority interrupts pending. IREQ2 could end up being serviced many times before IREQ5 is acknowledged. In many systems this is an appropriate method for handling the interrupting devices. Where circumstances permit, the masking capability of the Am9519 can be used by the host processor to modify the effective priority structure, perhaps by masking out recently serviced high priority devices, thus allowing lower priority inputs to be recognized.

Alternatively, where the eight interrupts have similar priority and service bandwidth requirements, the rotating priority mode may be selected (Mode register bit M0=1). As shown in Figure 15 the relative priorities remain the same as in the fixed mode; that is, IREQ2 is higher than IREQ3 which is higher than IREQ4, etc. However, in rotating priority mode, the lowest priority position in the circular chain is assigned by the hardware to the most recently serviced interrupt.

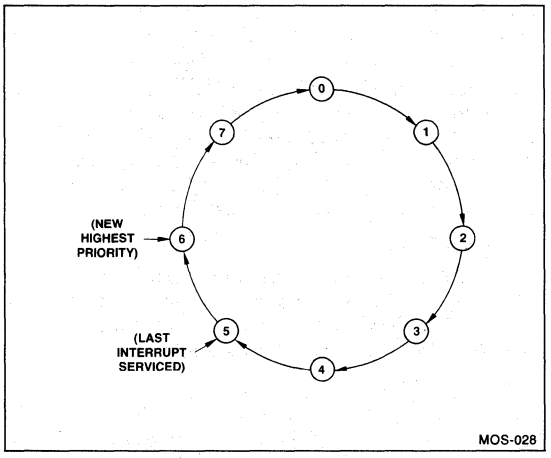


Figure 15. Rotating Priority Mode.

The example illustrated in Figure 15 assumes that IREQ5 has just finished being serviced and has therefore been assigned the lowest priority. Thus, IREQ6 occupies the new highest priority position, IREQ7 next-to-highest, etc. If two new interrupts then arrive at level 1 and level 4, IREQ1 will be selected and serviced, and will become the lowest priority. IREQ4 will then be acknowledged unless an active input on IREQ2 or IREQ3 has arrived in the meantime.

This rotating priority scheme prevents any request from dominating the system. It assures that an input will not have to wait for more than seven other service cycles before being acknowledged. Rotation occurs when the ISR bit of the presently selected interrupt is cleared.

In the rotating priority mode, inputs other than the one currently being serviced are fenced out and will not cause interrupts until the ISR bit is cleared. Thus, only one bit at a time will be set in the ISR. Care should be used when selecting the rotating mode to keep from doing so at a time when more than one ISR is set.

Vectoring

Bit M1 of the Mode register specifies the vectoring option. When M1=0 the individual vector mode is selected and each interrupt is associated with its own unique four-byte location in the response memory. When M1=1, on the other hand, the common vector mode is selected and all response information is supplied from the location associated with IREQ0, no matter which request is being acknowledged. This operating option will be useful in situations where several similar devices share a common service routine and direct individual device identification is not important. This may be true simply because of the nature of the peripheral/system interaction, or it may be a transient system condition that only uses the common vector option temporarily, perhaps to save the overhead involved in filling the response memory twice.

Polled Mode

Bit 2 of the Mode register allows the system to disable the GINT output. When M2=0 the interrupt mode is selected with the GINT output enabled. This might be considered the "normal" interrupt mode and makes full use of the interrupt control and management capabilities of the Am9519. When M2=1 the polled mode is selected which prevents the GINT output from going true by forcing it to its inactive state. In this condition, since no interrupts are supplied to the host processor, there will usually not be any $\overline{IACK}$ pulses returned to the Am9519. Consequently, ISR bits are not set, fences are not erected and IRR bits will not be automatically cleared. In the polled mode the host processor may read the Status register to determine if a request is pending and which request has the highest priority. IRR bits may be cleared by the host software. When the polled option is selected, the EI input is connected directly to the EO output thus functionally removing the polled chip from the external priority hierarchy.

Effectively, the polled mode of operation bypasses the hardware interrupt, inter-chip priority resolution, vectoring and fencing functions of the Am9519. What remains is the request latching, masking and intra-chip priority resolution.

GINT Polarity

Bit 3 of the Mode register specifies the sense of the GINT output. When M3=0, Group Interrupt is selected as active low ($\overline{GINT}$) and becomes an open drain output. This allows simple wired-or connections to other similar Am9519 outputs as well as to other sources of interrupts, and matches the polarity required by many processors. When M3=1, Group Interrupt is selected as active high (GINT) and becomes a two-state push-pull output, simplifying the interface to processors with active high interrupt inputs.

IREQ Polarity

Bit 4 of the Mode register specifies the sense of the IREQ inputs. When M4=0 the Interrupt Request signals are selected as active low ($\overline{IREQ}$) and a negative-going transition is required to set the IRR. When M4=1 the Interrupt Request signals are selected as active high (IREQ) and a positive-going transition is required to set the IRR. This sense option helps simplify the interface to interrupting devices.

Register Preselection

Bits 5 and 6 of the Mode register specify the internal data register that will be output by the Am9519 on any read operation at the data port ($\overline{CS}$=0, $\overline{RD}$=0, C/$\overline{D}$=0). These bits do not affect destinations for write operations. The four

registers available for reading are the IRR, ISR, IMR and ACR. Preselect coding for each register is shown in Figure 12. The preselection remains in effect for all data read transfers until the contents of M5 and M6 are changed.

The ability to examine these important operating registers, combined with the information available in the Status register, provides significant insight into the internal conditions of the Am9519. This allows the host processor not only enhanced dynamic operating flexibility, but also access to important diagnostic/testing/debugging information.

Master Mask

Bit 7 of the Mode register specifies the armed status of the Am9519 by way of the Master Mask control bit. When M7=0 the chip is disarmed just as if all eight bits in the IMR had been set. That is, IREQ inputs will be accepted and latched but will not cause GINT outputs to the host. In addition, the EO output is brought low, disabling any lower priority chips that may be attached. When M7=1, the chip is armed and any active unmasked interrupt inputs will be able to cause GINT outputs to the host processor.

The Master Mask capability permits the host system to disarm a chip and prevent processing of the interrupts without disturbing the contents of the IMR. Thus when the chip is re-armed, the old IMR conditions remain in effect and need not be reloaded. Note that a single command to the Master Mask bit of the highest priority interrupt chip is able to shut down the complete interrupt system, no matter how large.

Mode Reset

When a power-up or software reset occurs, the Mode register is cleared to all zeros. This means that after reset the following Mode register operating options will be in effect:

Fixed priority
Individual vectoring
Interrupt (non-polled) operation
GINT active low sense
IREQ active low sense
ISR preselected for reading
Chip disarmed by Master Mask

Operating Sequence

The management of interrupts by the Am9519 is illustrated below with a description of a fairly typical sequence of events. The Am9519 has already been initialized and enabled and is ready to run. The host processor has enabled its internal interrupt structure.

1. One (or more) of the IREQ inputs becomes active indicating that service is desired.
2. The requests are captured and latched in the IRR asynchronously. The latching action of the IRR cannot be disabled and active requests will always be stored unless a previous request at the same IRR bit has not been cleared.
3. If the active IRR bit is masked by the corresponding bit in the IMR, no further action takes place. When the IRR bit is not masked, an active Group Interrupt output will be generated if the Am9519 is not in its polled mode.
4. The GINT output from the Am9519 is used by the host processor as an interrupt input. When GINT is recognized by the host, it normally will complete the execution of its current instruction and will then execute some form of interrupt acknowledge sequence instead of the next program instruction. As part of the acknowledge cycle, the processor usually automatically disables its interrupt input. The Am9519 expects to receive one or more $\overline{IACK}$ signals from the processor during the acknowledge sequence.
5. When $\overline{IACK}$ is received, the Am9519 brings its $\overline{PAUSE}$ output low and begins selection of the highest priority unmasked active IRR bit. All interrupts that have become active before the falling edge of $\overline{IACK}$ are considered. When selection is complete, the $\overline{RIP}$ output is pulled low by the Am9519 and the contents of the first byte in the response memory associated with the selected request is accessed. $\overline{PAUSE}$ stays low until $\overline{RIP}$ goes low. $\overline{RIP}$ stays low until the last byte of the response has been transferred.
6. After $\overline{PAUSE}$ goes high, the host processor accepts the response byte on the data bus and brings the $\overline{IACK}$ line high. If another byte of response is required, another $\overline{IACK}$ pulse is output and is used by the Am9519 to access the next byte.
7. In parallel with the transfer of the first response byte, the Am9519 automatically clears the selected IRR bit and automatically sets the selected ISR bit. If the auto clear function is not in force for the selected interrupt, the ISR bit will cause a masking fence to be erected and GINT will be disabled until a higher priority interrupt arrives or until the ISR bit is cleared. The interrupt service routine will usually clear the ISR bit, often near the end of the routine.
8. If a higher priority request arrives while the current request is being serviced, and if the fixed priority mode is in effect, then GINT will be output again by the Am9519. The GINT signal will be recognized by the host processor only if the host has enabled its interrupt input. If this new request is acknowledged, the Am9519 will clear the corresponding IRR bit and set the corresponding ISR bit.
9. When the host processor has completed all interrupt service activity to satisfy the interrupting devices, it will normally clear the remaining ISR bit, if any, enable its internal interrupt system, if it has not already done so, and then return to the main program.

COMMAND DESCRIPTIONS

The Am9519 command set allows the host processor to customize and alter the interrupt operating modes and features for particular applications, to initialize and update the response locations, and to manipulate the internal controlling bit sets during interrupt servicing. Commands are entered from the data bus directly into the Command register by writing into the Am9519 control port ($\overline{CS}$=0, $\overline{WR}$=0, C/$\overline{D}$=1). All the available commands are described below and are summarized in Figure 17. In the binary coding of the commands, "X" indicates a do-not-care bit position.

RESET

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	0	0	0	0	0	0

Description: The Reset command allows the host processor to establish a known internal condition. The response memory and byte count registers are not affected by the software reset. The IMR is set to all ones. The ISR, IRR, ACR and Mode registers are cleared to all zeros.

CLEAR IRR AND IMR

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	0	1	0	X	X	X

Description: All bits in the IMR and all bits in the IRR are cleared at the same time. Thus all interrupts are enabled and the previous history of all IREQ transitions is forgotten. If GINT was active when the command was entered, it will go inactive.

CLEAR SINGLE IMR AND IRR BIT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	0	1	1	B2	B1	B0

Description: The same single bit position is cleared in both the IMR and the IRR. Other bits are not changed. If the specified IRR bit was generating an active interrupt output, GINT may go inactive upon entry of the command. The bit position cleared is specified by the B2, B1, B0 field as shown in Figure 16.

CLEAR IMR

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	0	0	X	X	X

Description: All bits in the IMR are cleared to zeros. All IRR bits will therefore be unmasked and any IRR bits that had been set will be able to cause an active GINT output after the command is entered.

CLEAR SINGLE IMR BIT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	0	1	B2	B1	B0

Description: A single bit in the IMR is cleared. Other bits are not changed. If the corresponding bit in the IRR was set, it will be unmasked and will be able to cause an active GINT after entry of the command. The IMR bit cleared is specified by the B2, B1, B0 field as shown in Figure 16.

SET IMR

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	1	0	X	X	X

Description: All bits in the IMR are set to ones. All IRR bits will therefore be masked and unable to generate an active GINT. If GINT had been active, it will go inactive after the command is entered.

SET SINGLE IMR BIT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	1	1	B2	B1	B0

Description: A single bit in the IMR is set. Other bits are not changed. If the corresponding bit in the IRR was active and generating a GINT output, GINT will become inactive after the command is entered. The IMR bit set is specified by the B2, B1, B0 field as shown in Figure 16.

CLEAR IRR

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	0	0	X	X	X

Description: All bits in the IRR are cleared to zeros. GINT will become inactive. New transitions on the IREQ inputs will be necessary to cause an interrupt.

CLEAR SINGLE IRR BIT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	0	1	B2	B1	B0

Description: A single bit in the IRR is cleared to zero. It will not cause an active GINT until it is set. The IRR bit cleared is specified by the B2, B1, B0 field as shown in Figure 16.

SET IRR

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	1	0	X	X	X

Description: All bits in the IRR are set to ones. Any that are unmasked will be able to cause an active GINT output. This command allows the host CPU to initiate eight interrupts in parallel.

SET SINGLE IRR BIT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	1	1	B2	B1	B0

Description: A single bit in the IRR is set to a one. If it is unmasked it will be able to generate an active GINT. This command allows the host processor to simulate with software the arrival of a hardware interrupt request. It also gives the software access to the hardware priority resolution, masking and control features of the Am9519. The bit set is specified by the B2, B1, B0 field as shown in Figure 16.

CLEAR HIGHEST PRIORITY ISR BIT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	1	0	X	X	X	X

Description: A single bit in the ISR is cleared to zero. If only one bit was set, that is the one cleared. If more than one bit was set, this command clears the one with the highest priority. This command is useful in software contexts where the service routine does not know which device is being serviced. It should be used with caution since the highest priority ISR bit may not really be the bit intended. When using the auto clear option on some interrupts and/or when a subroutine nesting hierarchy is not priority driven, the highest priority ISR bit may not correspond to the one being serviced.

CLEAR ISR

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	1	1	0	X	X	X

Description: All bits in the ISR are cleared to zeros. Mask fencing is eliminated.

CLEAR SINGLE ISR BIT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	1	1	1	B2	B1	B0

Description: A single bit in the ISR is cleared to zero. If the bit was already cleared, no effective operation takes place. The bit cleared is specified by the B2, B1, B0 field as shown in Figure 16. This will be the most useful command for service routines to use in managing the ISR without the help of the auto-clear option.

LOAD MODE BITS M0 THROUGH M4

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	0	0	M4	M3	M2	M1	M0

Description: The five low order bits of the Command register are transferred into the five low order bits of the Mode register. This command controls all of the Mode options except the master mask and the register preselection.

CONTROL MODE BITS M5, M6, M7

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	0	1	0	M6	M5	N1	N0

Description: The M6, M5 field in the command is loaded into the M6, M5 locations in the Mode register. This field controls the register preselection bits in the Mode register. The N1, N0 field in the command controls Mode bit M7 (Master Mask) and is decoded as follows:

N1	N0	
0	0	No change to M7
0	1	Set M7
1	0	Clear M7
1	1	(Illegal)

Thus, this command may be considered as three distinct commands, depending on the coding of N1 and N0:

1. Load M5, M6 only
2. Load M5, M6 and set M7
3. Load M5, M6 and clear M7

The Command Summary in Figure 17 lists all three versions.

PRESELECT IMR FOR WRITING

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	0	1	1	X	X	X	X

Description: The IMR is targeted to be loaded from the data bus when the next write operation occurs at the data port. All subsequent data write operations will also load the IMR until a different command is entered. Read operations may be successfully inserted between the entry of this command and the subsequent writing of data into the IMR. The Mode register is not affected by this command.

PRESELECT ACR FOR WRITING

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	0	0	X	X	X	X

Description: The ACR is targeted to be loaded from the data bus when the next write operation occurs at the data port. All subsequent data write operations will also load the ACR until a different command is entered. Read operations may be successfully inserted between the entry of this command and the subsequent writing of data into the ACR. The Mode register is not affected by this command.

PRESELECT RESPONSE MEMORY FOR WRITING

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	BY1	BY0	L2	L1	L0

Description: One level in the response memory is targeted for loading from the data bus by subsequent data write operations. The byte count register for that level is loaded from the BY1, BY0 field in the command. The L2, L1, L0 field specifies which of the eight response levels is being selected. This command should be followed by one to four data write operations to load response bytes. Field coding:

BY1	BY0	Count
0	0	1
0	1	2
1	0	3
1	1	4

L2	L1	L0	Level
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

The byte count value does not control the number of bytes entered into the response memory. It does control the number of bytes read from the memory by $\overline{IACK}$ pulses. Response bytes are output by the Am9519 in the same order they were entered.

B2	B1	B0	Bit Specified
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Figure 16. Coding of B2, B1, B0 Field of Commands.

COMMAND CODE								COMMAND DESCRIPTION
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	Reset
0	0	0	1	0	X	X	X	Clear all IRR and all IMR bits
0	0	0	1	1	B2	B1	B0	Clear IRR and IMR bit specified by B2, B1, B0
0	0	1	0	0	X	X	X	Clear all IMR bits
0	0	1	0	1	B2	B1	B0	Clear IMR bit specified by B2, B1, B0
0	0	1	1	0	X	X	X	Set all IMR bits
0	0	1	1	1	B2	B1	B0	Set IMR bit specified by B2, B1, B0
0	1	0	0	0	X	X	X	Clear all IRR bits
0	1	0	0	1	B2	B1	B0	Clear IRR bit specified by B2, B1, B0
0	1	0	1	0	X	X	X	Set all IRR bits
0	1	0	1	1	B2	B1	B0	Set IRR bit specified by B2, B1, B0
0	1	1	0	X	X	X	X	Clear highest priority ISR bit
0	1	1	1	0	X	X	X	Clear all ISR bits
0	1	1	1	1	B2	B1	B0	Clear ISR bit specified by B2, B1, B0
1	0	0	M4	M3	M2	M1	M0	Load Mode register bits 0–4 with specified pattern
1	0	1	0	M6	M5	0	0	Load Mode register bits 5, 6 with specified pattern
1	0	1	0	M6	M5	0	1	Load Mode register bits 5, 6 and set Mode bit 7
1	0	1	0	M6	M5	1	0	Load Mode register bits 5, 6 and clear Mode bit 7
1	0	1	1	X	X	X	X	Preselect IMR for subsequent loading from data bus
1	1	0	0	X	X	X	X	Preselect ACR for subsequent loading from data bus
1	1	1	BY1	BY0	L2	L1	L0	Load BY1, BY0 into byte count register and preselect response memory level specified by L2, L1, L0 for subsequent loading from data bus

Figure 17. Am9519 Command Summary.

SYSTEM INTERFACE

Expansion

Several Am9519 chips may be cascaded to expand the number of interrupts than can be handled by the system. A two-chip configuration is shown connected to an 8080A/9080A microprocessor in Figure 18. In general, expansion past a single Am9519 will require simply an added Chip Select signal for each extra chip, and perhaps an inverter for the GINT signal if the processor interrupt input is active-high. The $\overline{GINT}$, $\overline{PAUSE}$, and $\overline{RIP}$ signals are all designed to be wire-OR'ed in expanded systems.

Priority management in expanded systems is controlled by the Enable In, Enable Out and Response In Process signals. Figure 19 shows the basic interconnections for an example interrupt system that can accept up to 40 interrupts, using five Am9519 chips. Notice that $\overline{IACK}$ is wired in parallel to all five circuits, and that the $\overline{GINT}$, $\overline{RIP}$, and $\overline{PAUSE}$ lines are respectively tied together. The three pullup resistors are used to establish the high logic levels for the open-drain outputs. Enable In of the first chip (A) is allowed to float, or may be tied high. Each Enable Out signal is connected to the next lower level Enable In input. Each chip accepts eight IREQ inputs; for purposes of this example it is assumed that an active interrupt arrives at chip D in the chain.

Figure 20 shows the timing relationships for the configuration of Figure 19. When the IREQ arrives, a $\overline{GINT}$ output is generated by chip D and is used to interrupt the host processor. When the host returns an $\overline{IACK}$ pulse, all the EO lines are brought low in parallel. $\overline{PAUSE}$ also goes low, and is used to extend the $\overline{IACK}$ pulse.

After the fall of $\overline{IACK}$, all chips wait until a brief internal delay elapses and then examine EI. If EI is low, internal activity is suspended until EI goes high. If EI is high, then the internal circuitry is checked to see if an unmasked request is pending. If so, $\overline{RIP}$ is brought low, $\overline{PAUSE}$ is brought high, EO is kept low, and the first response byte is output on the data bus. In this example, there is no request in chip A and therefore the EO(A) line is brought high. This then allows chip B to see if it has an unmasked request waiting for service. If not, EO(B) goes high also and, with no interrupts at C, EO(C) goes high, driving EI(D) high. Since chip D finds a waiting request, it does not bring EO(D) high but it does bring $\overline{RIP}$ low. When $\overline{RIP}$ goes low it allows all the $\overline{PAUSE}$ outputs to switch high which permits the termination of the $\overline{IACK}$ pulse.

It can be seen, then, that the $\overline{PAUSE}$ output will automatically adjust the position of its rising edge to accommodate the exact functional and operational conditions that occur for each particular $\overline{IACK}$ cycle. For larger systems, like that in Figure 19, operating at high temperatures with slow versions of the Am9519 and servicing low priority interrupts, the processor delay caused by $\overline{PAUSE}$ may be quite long and a few processor wait cycles may be required to extend the $\overline{IACK}$ pulse. On the other hand, when a system like Figure 18 is running at typical room temperatures with typical parts and the interrupt is a high priority one, the $\overline{PAUSE}$ output width will be quite narrow and no wait cycles will be necessary.

The $\overline{RIP}$ output serves two basic functions within the interrupt system. First, its falling edge informs the other connected chips that an interrupt request has been selected and $\overline{PAUSE}$ may, therefore, be released. Secondly, as long as $\overline{RIP}$ is low, only the single chip that is pulling $\overline{RIP}$ down is allowed to respond to $\overline{IACK}$ inputs. $\overline{RIP}$ stays low until all response bytes for the selected interrupt have been transferred.

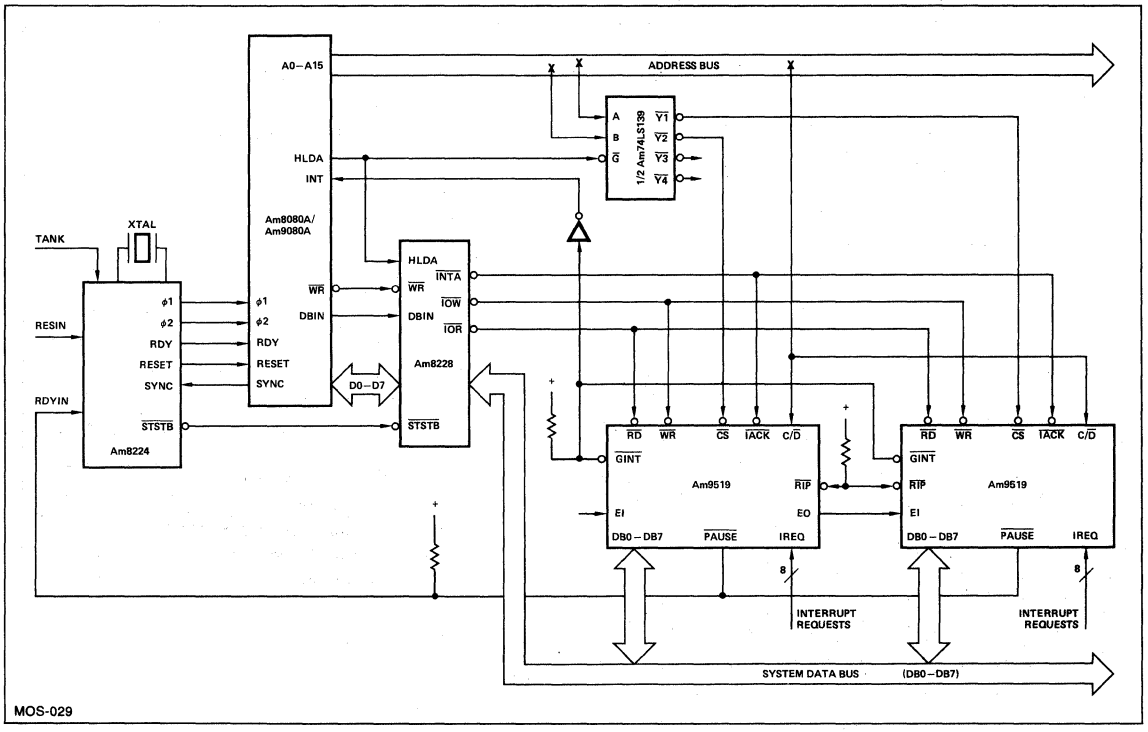


Figure 18. 16 Interrupt Configuration with 8080A/9080A.

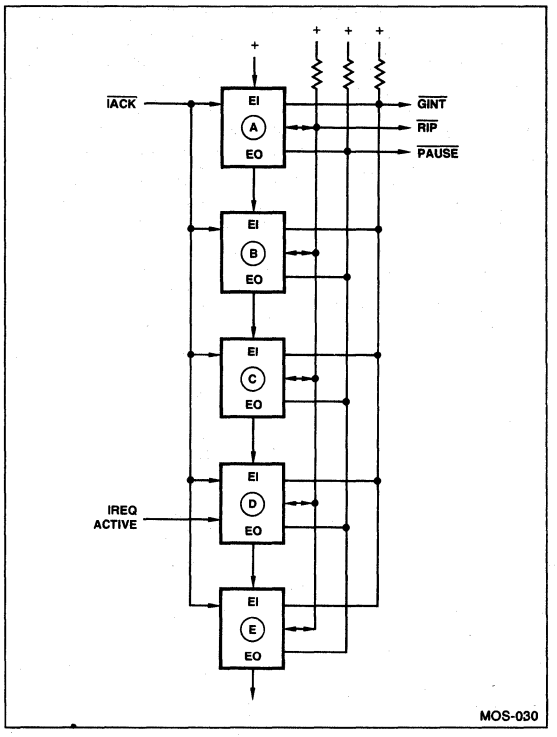


Figure 19. Five-Chip Expansion Example.

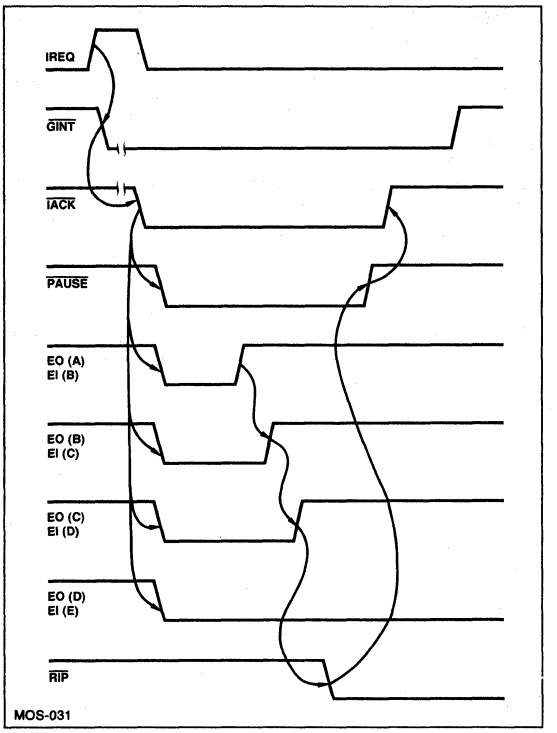


Figure 20. Inter-Chip Priority Resolution.

Assume that a new interrupt arrives at chip B in Figure 19 during the time that the first byte of a multibyte response for the interrupt at chip D is being transferred. Without the $\overline{RIP}$ signal there would be confusion when the second $\overline{IACK}$ pulse arrives. Both chips might try to send out response bytes since the interrupt at chip B is a higher priority, yet chip D is in the midst of a response sequence. With $\overline{RIP}$ present, however, no problem arises. Chip D pulls $\overline{RIP}$ low when it is selected and keeps $\overline{RIP}$ low until its response is complete. Chip B treats $\overline{RIP}$ as an input and will not respond to $\overline{IACK}$ until $\overline{RIP}$ goes high.

Initialization and Support

Before the Am9519 can perform useful work, it must be initialized to customize it for a particular application and to load it with appropriate data values. During active operation, control options may be changed and response data may be modified. Because of the many ways it might be used, the Am9519 can be programmed using many different approaches. The following sequence description shows only one of several possible methods for constructing a basic service routine:

1. Disable processor interrupts.
2. Execute software reset at Am9519.
3. Transfer commands and response data from a control table into the Am9519.
4. Transfer operating options into the Mode register.
5. Transfer the operating Mask conditions into the IMR.
6. Clear the IRR.
7. Clear Master Mask.
8. Enable processor interrupts.
9. Return.

Figure 21 shows an example listing for such a routine using 8080A/9080A coding. Several assumptions are made about the hardware and software environment in which the routine will function:

1. When the routine is entered, register pair H, L contains the address of the first location of a control table, and register B contains a count that indicates how many entries the table contains.
2. One Am9519 is in the system. Its data port is decoded by the hardware as hex I/O address C2. Its control port is hex I/O address C3.
3. Only the first five interrupts will be in use by the main program. The others will be used later to support other processes.
4. Main program options: Fixed priority, Individual vectoring, Interrupt mode, GINT active high, IREQ active low, IRR selected for reading, Auto clear not used.

The control table is an important part of the routine and contains command information as well as the response data itself. The table consists of up to eight entries, each up to five bytes long, with all entries the same length. The first byte of each entry contains the response memory preselect command code with fields for the BY1, BY0 byte count and the L2, L1, L0 level pointer. The next one to four bytes of each entry contain the data loaded into the response memory. In this example the table has entries of four bytes each and is illustrated in Figure 22.

This type of table organization may contain extra bytes, but it compensates for this by allowing a brief, simple program to handle it. The table is fairly general and allows any length response to be programmed independently for each interrupt. It

```
LOC  OBJ          SEQ          SOURCE STATEMENT

                    0 ;
                    1 ;       * * * * * * * * * * * * * * * * * * * *
                    2 ;       *                                       *
                    3 ;       *          EXAMPLE PROGRAM              *
                    4 ;       *  FOR CONTROL AND INITIALIZATION       *
                    5 ;       *            OF THE AM9519              *
                    6 ;       *  UNIVERSAL INTERRUPT CONTROLLER       *
                    7 ;       *                                       *
                    8 ;       * * * * * * * * * * * * * * * * * * * *
                    9 ;
                   10 ;
                   11 ;
3000               12           ORG 3000H
                   13                                ;
00C3               14 CPORT     EQU 0C3H             ; CONTROL PORT ADDRESS.
00C2               15 DPORT     EQU 0C2H             ; DATA PORT ADDRESS.
00B0               16 LMASK     EQU 10110000B        ; LOAD MASK COMMAND.
00E0               17 MASK1     EQU 11100000B        ; MASK VALUE TO ENABLE THE FIRST
                   18                                ;   FIVE INTERRUPTS.
0088               19 MODE1     EQU 10001000B        ; MODE COMMAND FOR  NORMAL
                   20                                ;   OPERATION. (M0 - M4 ONLY)
00A9               21 MODE2     EQU 10101001B        ; MODE COMMAND TO CLEAR MASTER
                   22                                ;   MASK AND PRESELECT IRR.
0040               23 CLIRR     EQU 01000000B        ; CLEAR IRR COMMAND.
                   24                                ;
                   25                                ;
3000 F3            26 ENTRY:    DI                   ; DISABLE CPU INTERRUPTS
3001 3E00          27           MVI A,00000000B      ; GET SOFTWARE RESET COMMAND AND
3003 D3C3          28           OUT CPORT            ; SEND TO CONTROL PORT.
                   29                                ;
3005 7E            30 AAA:      MOV A,M              ; GET CONTROL BYTE FROM TABLE &
3006 D3C3          31           OUT CPORT            ; SEND TO CONTROL PORT.
3008 23            32           INX H                ; INCREMENT TABLE POINTER.
3009 0E03          33           MVI C,3              ; INITIALIZE VECTOR BYTE COUNT.
300B 7E            34 BBB:      MOV A,M              ; GET VECTOR BYTE AND
300C D3C2          35           OUT DPORT            ; SEND TO DATA PORT.
300E 23            36           INX H                ; POINT TO NEXT TABLE BYTE.
300F 0D            37           DCR C                ; DECREMENT BYTE COUNT.
3010 C20B30        38           JNZ BBB              ; ENTRY DONE?  NO:BACK TO BBB.
3013 05            39           DCR B                ; YES:DECREMENT ENTRY COUNTER.
3014 C20530        40           JNZ AAA              ; TABLE DONE? NO:BACK TO AAA.
                   41                                ;
3017 3E88          42           MVI A,MODE1          ; YES: PROCEED WITH MODE BYTE.
3019 D3C3          43           OUT CPORT            ; SEND MODE TO CONTROL PORT.
301B 3EB0          44           MVI A,LMASK          ; GET MASK LOAD COMMAND AND
301D D3C3          45           OUT CPORT            ; SEND TO CONTROL PORT.
301F 3EE0          46           MVI A,MASK1          ; GET OPERATING MASK AND
3021 D3C2          47           OUT DPORT            ; SEND TO DATA PORT.
3023 3E40          48           MVI A,CLIRR          ; GET CLEAR IRR COMMAND AND
3025 D3C3          49           OUT CPORT            ; SEND TO CONTROL PORT.
3027 3EA9          50           MVI A,MODE2          ; GET CLEAR MASTER MASK COMMAND
3029 D3C3          51           OUT CPORT            ; AND SEND TO CONTROL PORT.
302B FB            52           EI                   ; ENABLE CPU INTERRUPTS.
302C C9            53           RET                  ; RETURN TO CALLING PROGRAM.
                   54                                ;
                   55           END
```

Figure 21. Example Routine.

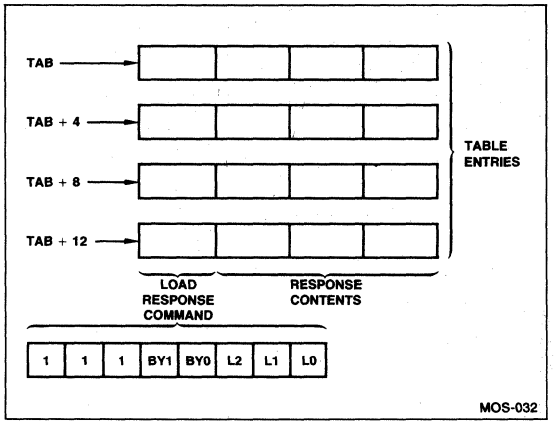


Figure 22. Example Control Table.

also allows any number of response locations to be updated in any order. The program driving the table simply assumes that every response level receives the same number of bytes as the level with the longest response.

Other table organizations are also possible. A more general table could contain the IMR value to be used, the ACR value, the table byte length, the operating mode values, etc. As more of the variable control information is added to the table, the software routine becomes more general and can be used not only for initialization, but for operational changes as well. Then there might be several tables in memory with an address supplied to the routine that points to the controlling table to be used. Note that the calling program can use just portions of an existing table if desired, simply by controlling the contents of the machine registers when the routine is entered.

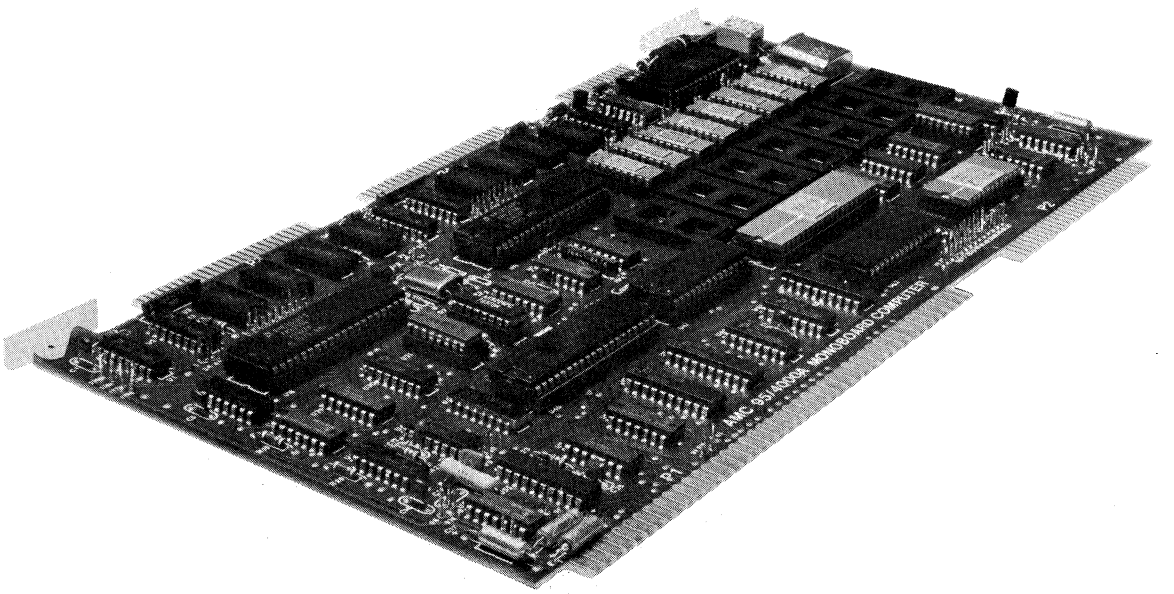
Another approach is to omit the byte count/level command code from the table and compute its value in the driving routine. This may be especially appropriate when all the response entries are the same length and contiguous levels are being filled. The BY1, BY0 field need not change then, and a simple increment instruction will generate the proper command coding by changing the L2, L1, L0 field. To minimize the table length, which might become an important consideration for larger systems with many more interrupts, it is also possible to use the byte count to control the number of bytes transferred into each memory level.

The Am9519 offers new levels of versatility and sophistication for interrupt systems. It represents interesting opportunities for both hardware and software engineers to enhance new designs and to take advantage of the features now available.



Advanced Micro Computers
Boards and Development Systems

Am95/4005
Multimaster MonoBoard™ Computer
(Including the Am95/4000)



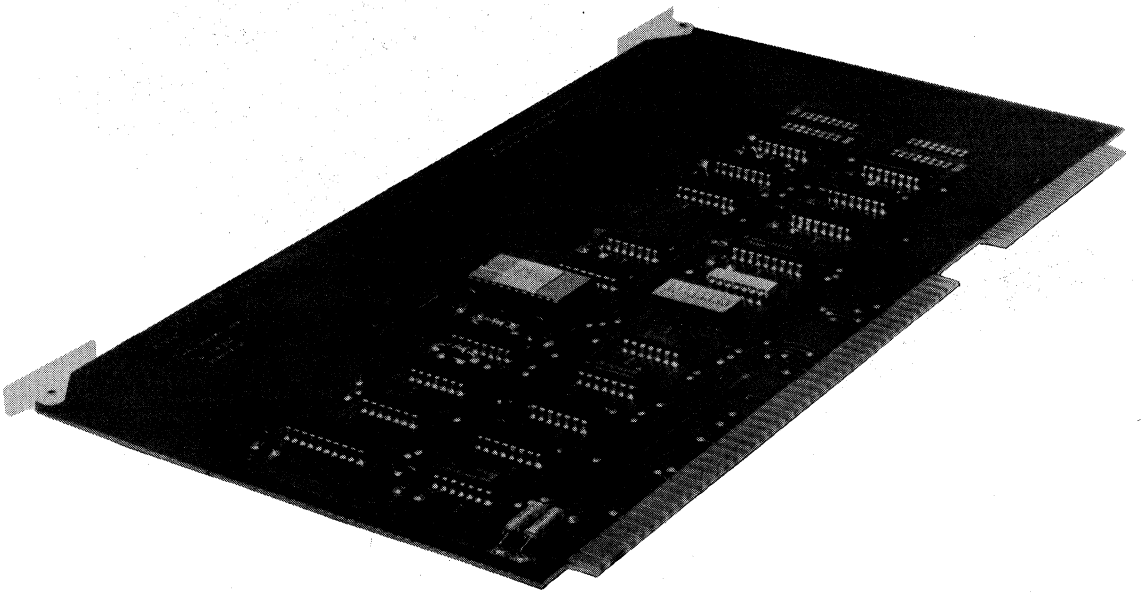
Features

- ☐ Am9080A CPU* 2MHz (2μsec Instruction Cycle) Standard; 3MHz (1.3μsec) Optional
- ☐ Am9511A or Am9512 Arithmetic Processing Unit Allows High Speed Arithmetic Computations Concurrent with CPU Operation. Ideal for Industrial Monitoring and Process Control, Navigation, Medical, Instrumentatic and Other Computation Intensive Applications.
- ☐ 4K Bytes of High-Speed Static RAM (Am9114)
- ☐ Sockets for 4/8/16K Bytes of ROM/E-PROM (2708/2716/2732)
- ☐ 4 Channels of DMA (Am9517A) with Block Memory-to-Memory and Memory-to-I/O Transfer Capability
- ☐ 8 High-Speed Interrupt Channels with Vectored Priority (Am9519)
- ☐ 1.2kHz Crystal Controlled Real-Time Clock
- ☐ 48 Programmable I/O Lines (two Am9555s) with Sockets for Line Drivers and Terminators
- ☐ Serial Interface for RS232C or Current Loop Interface (Am9551), with Switch Selectable Baud Rate (50-9600 Baud)
- ☐ Memory Shadow: Bootstrap Program in On-Board or Off-Board ROM can be Selected by Power-On or Initialization and then Program Disabled
- ☐ Memory can be Reconfigured by Changing the PROM Decoder
- ☐ Multimaster Bus Control Logic allows up to three CPUs to Share the System Bus.
- ☐ Compatible with Multibus† Bus Standard, SBC-80 Card Format
- ☐ Complemented by a Family of Boards (Memory, Floppy Disk Controller, ROM/E-PROM and I/O Expansion) and a Rugged 6-Slot Card Cage

†Registered trademark of Intel Corporation.

*The Am9080A is a high-speed, pin compatible version of the 8080A.

Am95/6012
Arithmetic Floating-Point Processor Unit Board
(IEEE Data Format)

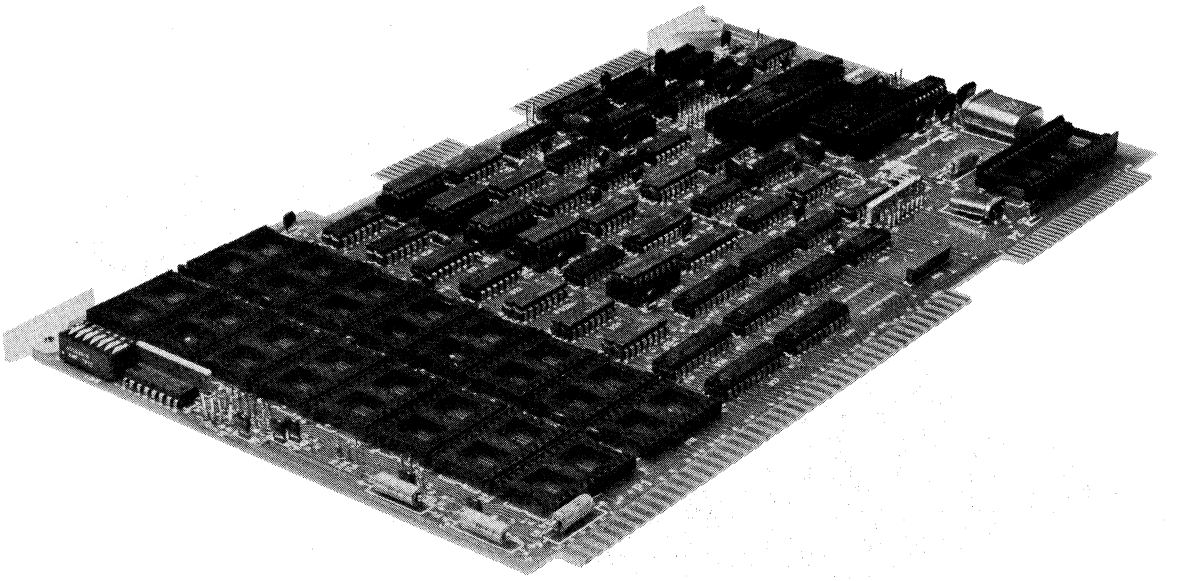


Features

- ☐ Floating-Point Processor
- ☐ 32-Bit and 64-Bit Floating-Point Arithmetic
- ☐ Conforms to IEEE Data Format
- ☐ Single Board Design
- ☐ Uses Advanced LSI Am9512 Arithmetic Processing Unit
- ☐ Low Power Consumption
- ☐ Plug and Bus Compatible with:
  - Intel Intellec® Microcomputer Development Systems
  - Intel's iSBC® Family of Single Board Computers
- ☐ Switch Selectable I/O Addresses
- ☐ Requires Only Four Input/Output Addresses
- ☐ Independent On-Board Counter Operates Under CPU Control to Monitor APU Execution Times

®Registered trademark of Intel Corporation.

Am95/5032
PROM/ROM I/O Board

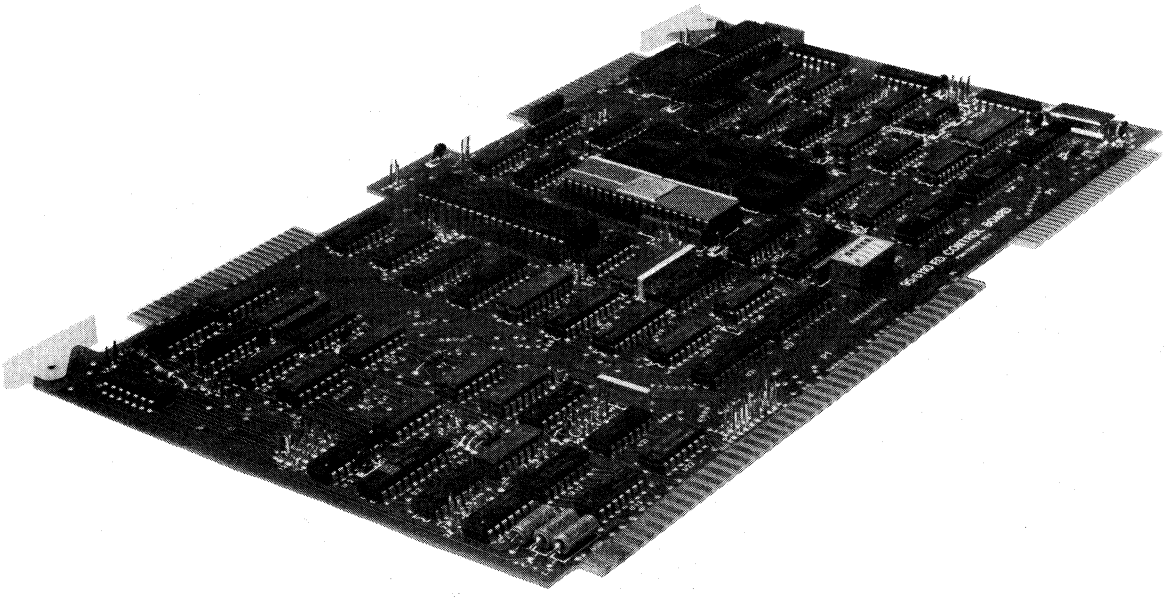


Features

- ☐ Provides Sockets for Up to 64K Bytes of PROM/ROM in a Multibus† Compatible Board
- ☐ 8-Bit and 16-Bit CPU Compatibility
- ☐ Permits RAM/PROM to Co-Exist in the Same Memory Space
- ☐ Programmed Control to Enable/Disable All Combinations of PROM/ROM Sockets. Permits Multiple Independent Program Modules to Occupy the Same Memory Area
- ☐ Jumper-Selectable Boot Option; Swappable for RAM Under Program Control
- ☐ Supports All Type of Multibus† Data Transfers Under Automatic Firmware Control
- ☐ Allows Jumper-Selectable Combinations of 2708, 2716 and 2732 E-PROMS and Similar ROMs
- ☐ Total Versatility for Memory Mapping PROM/ROM Area via Bipolar PROMs; Assures the Integrity of the Memory Map and Eliminates the Potential Problems from Multiple Mechanical Components
- ☐ Optional I/O Section Includes:
  - RS232 Serial I/O Port Under USART Control
  - Three 8-Bit (or 24-Line) Parallel Ports
  - Timing Controller (Am9513) with Five 16-Bit High Speed Counters
  - All Functions Programmable via I/O Ports
  - Nine Interrupt Sources Jumper-Selectable to Bus Interrupt Lines.

†Multibus is a registered trademark of Intel Corporation.

Am95/6110
Floppy Disk Controller

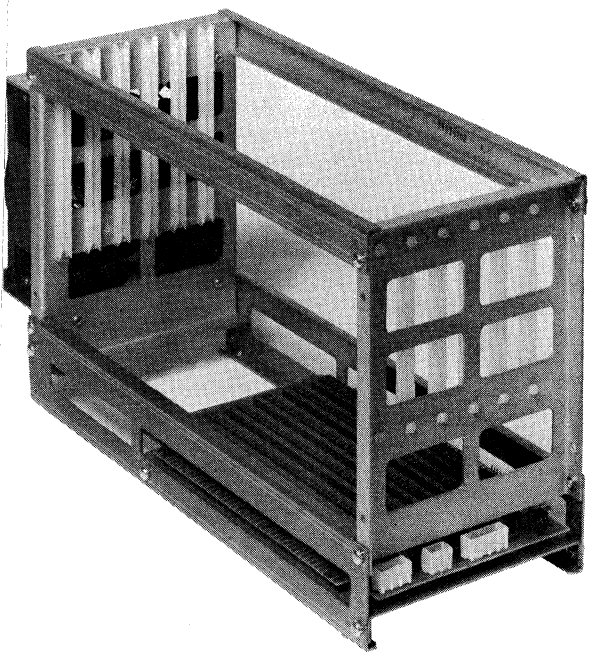


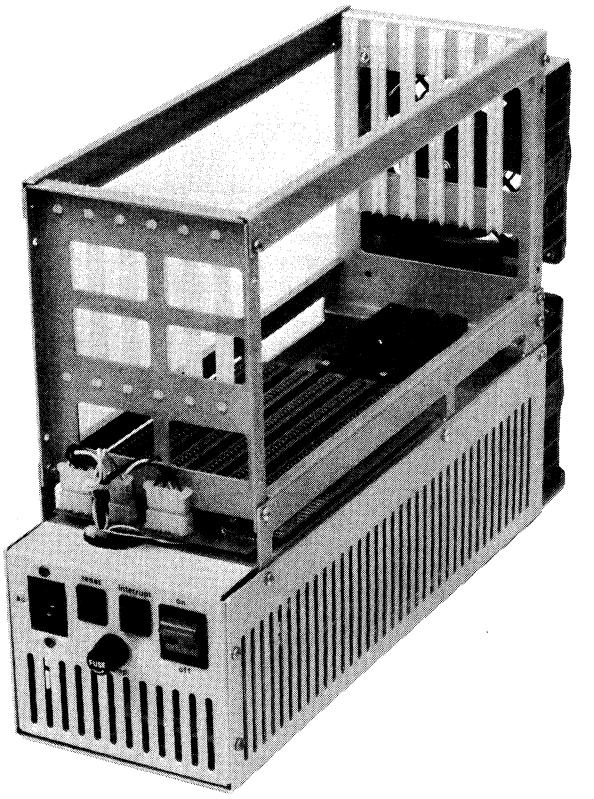
Features

- ☐ CONTROLS FOUR DRIVES, 8-Inch Flexible Disk Drives, Single or Double-Sided, IBM 3740 Soft Sector Media Format, Single Density. Compatible with Shugart, Memorex, CDC, Siemens Drives.
- ☐ DISTRIBUTED I/O PROCESSOR ARCHITECTURE. Performs all Disk I/O without Host CPU Intervention by means of its own Dedicated Processor, Control Firmware and RAM Buffer.
- ☐ CONTROL FIRMWARE. Provides Program Code for READ, WRITE, EXECUTE and INITIALIZE plus several Other Commands for Error Checks and Status Words.
- ☐ HIGH THROUGHPUT by means of On-Board DMA Controller, Programmable for Either Block or Byte Mode Transfers. 20-Bit Addressing Allows Transfers Up to 1M Byte.
- ☐ CONFIDENCE CHECK. Automatically Provides a Diagnostic Check on Start-Up of RAM, ROM, FDC and DMA . . . and Provides Status Word Response to CPU.
- ☐ AUTOMATIC SYSTEM BOOT. Provides a Selectable System Boot Capability with the Booting Routine Residence on Disk and Automatically Read into Memory on Start-up, Initialization or Reset.
- ☐ VERSATILE INTERFACE. SBC/Multibus† Compatible, Single or Multimaster Environment; Operates as an Intelligent Slave; Interfaces to Both 8-Bit and 16-Bit CPU.
- ☐ MULTIBUS SBC-80 COMPATIBLE.

†Multibus is a trademark of Intel Corporation.

Am95/6440 and Am95/6448
Card Cages





Features: Am95/6440

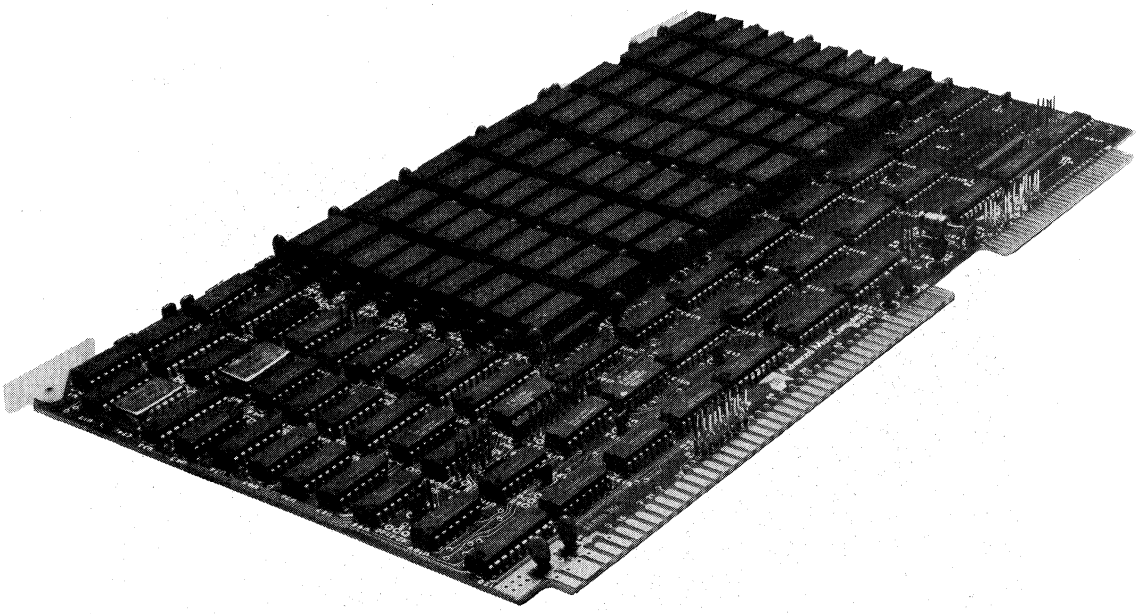
- ☐ Six-slot PC board capacity
- ☐ Multibus™/SBC-80 compatible
- ☐ Rugged metal construction
- ☐ Heavy-duty power connectors
- ☐ Jumper options for Multi-Master bus capability
- ☐ Optional cooling fan
- ☐ Mounting holes for auxiliary connectors

Multibus is a trademark of Intel Corporation; it defines a versatile bus structure that is popular in the industrial control and data processing industry. It is currently being considered for adoption as an IEEE standard.

Features: Am95/6448

- ☐ Built-in high-efficiency power supply
- ☐ Compact integral design, ideal for industrial chassis and bench top systems
- ☐ Convenient input panel containing:
  - Power switch w/indicator
  - Reset switch
  - Interrupt switch
  - AC receptacle with EMI filter
  - Fuse holder
- ☐ Cooling fans included
- ☐ Provides all Multibus-required voltages
- ☐ Includes all features of standard Am95/6440 card cage

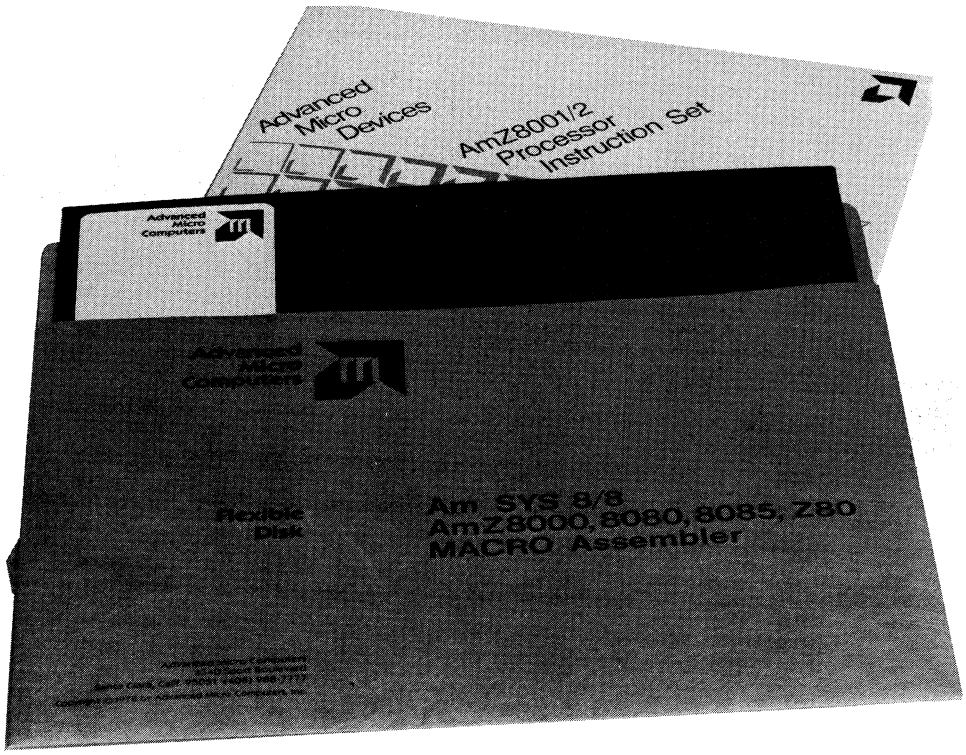
Am96/1000
Memory System



Features

- ☐ One of the AMC Supercomponent™ Family of LSI Boards
- ☐ 32K, 64K, 96K and 128K Byte Storage Options Available
- ☐ High-Speed Operation Supports Up to 4.0MHz Operation with High-Performance AmZ8000 CPUs
- ☐ 8-Bit/16-Bit Data Bus Compatibility for Most 8- or 16-Bit Microcomputers
- ☐ On-Board Transparent Refresh
- ☐ Jumper Option for Advanced Acknowledge (AACK) Signal to Improve Response Time and Throughput
- ☐ Optional Parity Option in Both Byte and Word Mode with Interrupt Capability
- ☐ Dual Bus Accesses Provide a Global RAM Link Between P1 Multimaster Connector and P2 Auxiliary Connector
- ☐ Address Space in 4K Byte Boundaries
- ☐ Multibus and SBC-80 Compatible

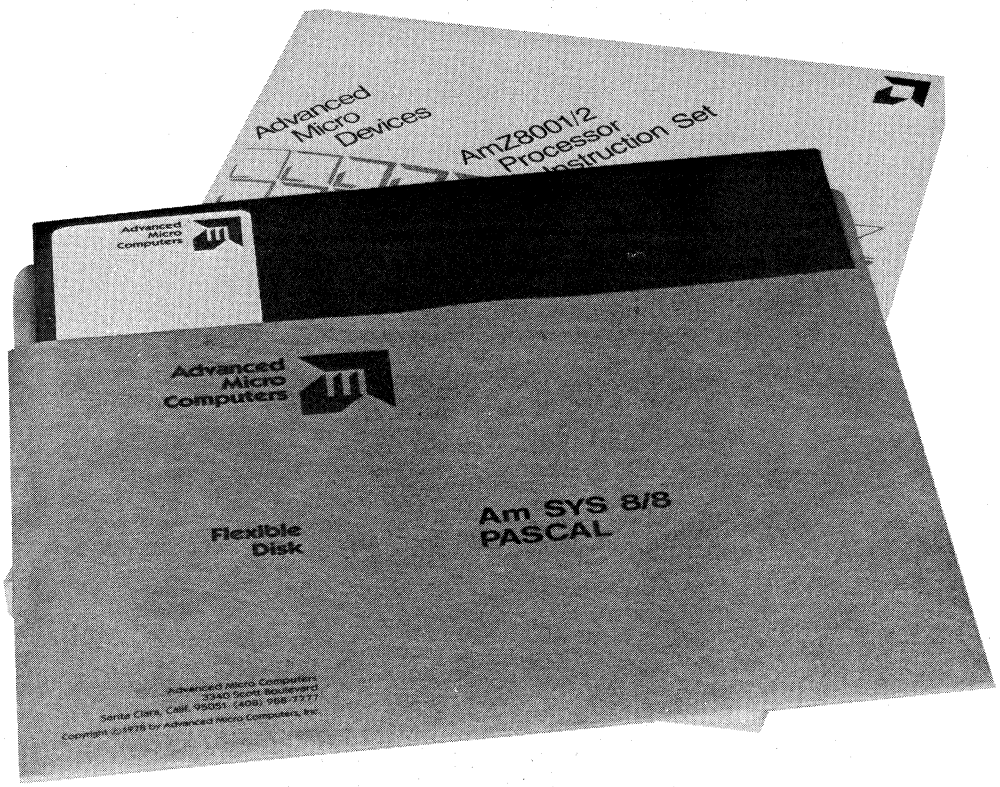
AMC's MACRO8000
AmZ8000 Macro Assembler



Features

- ☐ Functions in the AmSYS 8/8 or AmSYS 29 microcomputer development system environments
- ☐ Generates absolute or relocatable AmZ8000 object code
- ☐ Supports the full AmZ8000 instruction set
- ☐ Symbolic operands that are constants or variables
- ☐ Syntactic macros
- ☐ Program segmentation for address space control
- ☐ Arithmetic expressions, string expressions, and comparisons
- ☐ Superior assembly speed and a variety of assembly options

AMC's PASCAL
Compiler



Features

- ☐ The Compiler Runs in the AmSYS 8/8 or the AmSYS 29 Microcomputer Development Systems Environments
- ☐ Object Program may be one of the Following:
  - P-code (Executed by an Interpreter)
  - Am9080 Machine Code
  - AmZ8000 Machine Code
- ☐ Block Structured High Level Language that Supports Structured Design and Programming
- ☐ Many Extensions such as Separate Compilation, Strings, etc.

AmSYS™8/8 Microcomputer Development System

Especially Designed to Support
the AmZ8000 Microprocessor
and the 8080A, 8085A, Z80 & 8048

For Total Capability, AMC gives you the AmSYS 8/8 Microcomputer Development System

SYSTEM DESCRIPTION

The AmSYS 8/8 is designed to support a variety of microprocessors that include the AmZ8000, Z80, Am9080, Am8085, and Am8048.

This system is especially designed to support the AmZ8000 CPU in both hardware and software development. An AmZ8000 Macro Assembler, linker and line oriented editor give the system powerful software development tools. The use of the RTE 8/8050 In-Circuit Emulator for AmZ8000 hardware/software debug and availability of high level languages allow the user great flexibility in development of AmZ8000 products.

The basic system is contained in a single enclosure that fits on a desk top. It has the option of being rack mountable in a 19" relay rack by the addition of slide mounts. It has 64K bytes of main memory, two single density floppy disk drives, providing 512K bytes of storage, an RS232 serial port and two parallel I/O channels as standard features in the system. This system contains extra card slots for use for prototyping hardware.

The AmSYS 8/8 has a multi-master bus structure that allows multiple 8 and 16 bit CPUs to be used at the same time. This bus structure allows the system to be easily upgraded to a full 16 bits by replacing the CPU board on the bus and adding the appropriate software.

The AmSYS 8/8 has very powerful software included as part of the standard system to be delivered. This software includes a Disk Operating system, an excellent text editor, debugger, library manager and linking loader. The powerful macroassemblers included support the AmZ8000 and the 8080, 8085, & Z80. Optionally, an 8048 macroassembler can be added.

The AmSYS 8/8 features the use of Pascal along with extended Basic, Fortran IV and Cobol for high level language support. A Pascal program can be compiled to generate either Am9080 or AmZ8000 code.

In-Circuit Emulation capability is available for both 8 & 16 bit processors. This emulation support is optional for the AmZ8000, Z80, 8080, 8085, & 8048 microprocessors.

The AmSYS 8/8 also has a number of peripherals that are optional to the system. These include double density Floppy Disk Drives, 60, 120 CPS and 300 LPM Line Printers, CRT's, and soon a cartridge disk drive.

SYSTEM FEATURES

Complete Turnkey Development System with Powerful Hardware including:

- ☐ Microprocessor CPU
- ☐ 64 kilobytes of read/write main memory
- ☐ Serial interface RS232 compatible
- ☐ Two parallel Channels
- ☐ Power up Monitor
- ☐ Dual Floppy Disk Drives with 512K bytes of storage capacity
- ☐ Extra card slots for prototyping
- ☐ Multi-master bus

Powerful Software Including:

- ☐ AMDOS Disk Operating System
- ☐ Sophisticated Editor
- ☐ Debugger
- ☐ Macroassemblers for AmZ8000, Z80, 8080, 8085
- ☐ Translator to Z8000
- ☐ Linkers

Powerful Options Including:

- ☐ High level languages
  - • Pascal
  - • Extended Basic
  - • Fortran
  - • Cobol
- ☐ Line Printers 60 & 120 CPS & 300 LPM
- ☐ CRT w/extra keypad
- ☐ PROM Programmer
- ☐ Double Density Floppy Disk Drives
- ☐ Expansion Chassis containing two additional Disk Drives
- ☐ Additional Serial I/O lines
- ☐ In-Circuit Emulation for
  - • AmZ8000
  - • 8080, 8085
  - • 8048, 8021, 8049, & 8035

Additional Features that will be available in the near future include:

- ☐ Expansion to 1 megabyte of main memory
- ☐ Upgrade to AmZ8000 CPU
- ☐ Cartridge Disk
- ☐ Additional High level languages
- ☐ Expansion chassis for more Main Memory

System Hardware Features

The AmSYS 8/8 features the use of a multiple master bus structure. This allows the system to use both 8 and 16 bit microprocessors. The SYS 8/8 can be upgraded to include a 16 bit CPU board and appropriate software so that both microprocessors may operate simultaneously using the multi-master bus structure. This allows the system user maximum flexibility in the development of hardware & software.

MAIN CPU MODULE

The Central Processor (CPU) module uses an 8-bit Am9080 microprocessor. This module includes a ROM based auto bootstrap that first performs a diagnostic confidence check of the system and next loads the operating system from the system disk drives. The CPU module has multi-master capability allowing operation in a master/slave environment with other 8-bit CPU modules or the 16-bit CPU module. The CPU module also includes 4 high speed DMA channels and 8 vectored interrupt channels. An optional Am9511A/Am9512 Arithmetic Processing Unit can be added to provide high speed fixed and floating point computations and floating point transcendental functions to the system. A 1.2KHz crystal controlled Real Time Clock is also part of this module.

MEMORY MODULE

This memory module is organized as a 64 Kilobyte storage unit. This module is organized to provide either 8 or 16-bit compatibility. It also contains internal memory refresh capability and is multimaster bus compatible.

FLOPPY DISK MODULE

This module features a controller with its own internal microprocessor and controls up to 4 single or double sided floppy disks. This controller has a 20-bit address and DMA capability allowing it to address up to 1 megabyte of main memory.

This module is IBM 3740 soft sector compatible. AmSYS 8/8 provides a capacity of 512K bytes of disk storage in the basic system. This storage capacity is contained on two floppy disk drives each with 256K bytes of storage. On board buffer memory allows high speed data transfer to the CPU module. Additional storage is available by the use of the optional double density disk controller or by the addition of a double disk drive expansion chassis.

CHASSIS

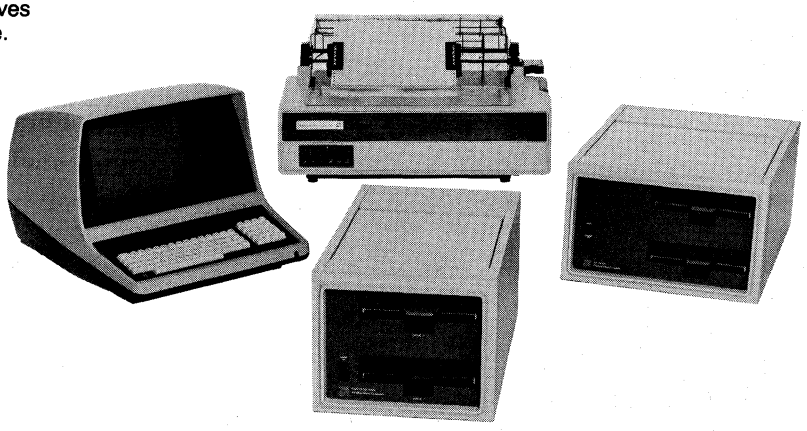
The AmSYS 8/8 is normally supplied as a desk top chassis. As an option, this chassis can be rack mounted with the addition of a slide mount kit. It contains seven card slots with three available for use with prototyping or additional I/O P.C. cards. This chassis is totally self contained with cooling and internal power supplies.

The front of the chassis contains an illuminated off/on switch and also a system reset button. The rear of this unit contains all I/O and power connections including fuses.

The system includes as standard, one serial port and two parallel channels. The serial port is RS232 compatible and allows the user to connect a variety of terminals to the system. The parallel channels can be used to connect high and low speed printers or other peripheral devices. Additional I/O ports can be added to the System by the use of plug in cards.

The parallel channels each consist of 24 parallel I/O lines. These lines can be configured under software control as sets of input, output, or bidirectional I/O lines. Sockets are provided within the system to add drivers or terminators to each line.

The Am96/4016 Evaluation Board can be plugged into one of the empty card cage slots to provide a 16 bit execution module within AmSYS 8/8. The board is powered by the system power supplies and uses the system floppy disks via the parallel up-load/down-load link to the CPU module. A second card cage slot can be used to upgrade the Evaluation Board RAM to 64k.



Development Software

The AmSYS 8/8 combines a sophisticated set of hardware along with an even more powerful set of software to make the complex process of product development easier to handle for the user. This software includes an Editor, File Manager, Macroassemblers, disk operating system, and a number of high level languages.

The software that comes standard with the AmSYS 8/8 is the AMDOS® Operating System, AmZ8000, 8080, 8085, and Z80 Macro Assemblers, a linking loader, a powerful Editor, debugger and Translator. Optional software includes an 8048 Macroassembler, Pascal, Fortran, Basic, Cobol, and upload/download packages for execution of AmZ8000 code. The following describes the major software modules provided with or as options to the system.

OPERATING SYSTEM

AMDOS 8 is the disk operating system for the AmSYS 8/8 and provides rapid access to programs through a comprehensive file management structure. A file subsystem supports a named file structure allowing the dynamic allocation of file space as well as sequential and random file access. System calls are provided permitting files to be opened, closed, renamed, read, written onto disk, or searched for by name. The AMDOS 8 file system allows a large number of distinct programs to be stored in both source and machine executable forms. AMDOS 8 also provides the ability to access disk storage, Terminal, Printer, and support for PROM programming.

LIBR is an object file library manager that permits the creation and listing of user libraries for the Am8080, Am8085, and Z80 microcomputers. Individual object-module files may be selectively processed to extract modules from existing libraries or relocatable object files to build libraries for later selective searching by the LINK linkage editor.

EDIT is a line-oriented context editor providing the user the ability to create and modify ASCII source text for the AmSYS 8/8 compilers and assemblers. A powerful set of user commands is available to simplify the task of generating line oriented or character oriented files.

In the line oriented mode, numbers will automatically be added to new lines as they are appended to the file. Also, as lines are inserted or deleted within the text, these numbers are updated. The users position within the file can then be controlled by referring to the desired line number. The ALTER mode allows the user to insert, delete, and replace individual characters, strings in a line, or a range of lines in the source file. Searching for occurrences of characters or strings is also included along with substitution of characters or strings into the source text.

DEBUG is designed to provide dynamic interactive testing and debugging of 8-bit programs generated in the AmSYS 8/8 system. This program consists of two parts: The debug nucleus and the assembler/disassembler module. These along with powerful user commands make debug a very valuable tool for debugging software programs.

MACRO ASSEMBLERS

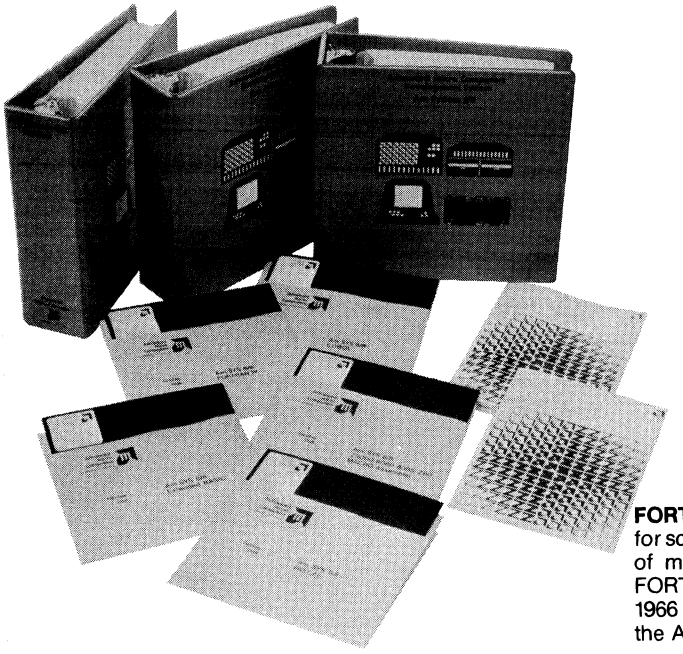
Two macroassemblers come as part of the standard AmSYS 8/8. These are MACRO 8 supporting the 8080, 8085, and Z80 microprocessors and MACRO 8000 for support of the AmZ8000 microprocessor. An optional macroassembler supports the 8048 family.

MACRO 8 is a comprehensive macroassembler providing the ability to assemble relocatable 8080/ 8085/ Z80 programs. These can be combined with object files produced by high-level language compilers (using LINK) to form composite executable programs. In addition to comprehensive conditional assembly directives, hierarchical expression evaluation, definition of alternate entry-points and external symbols, MACRO 8 provides a companion cross-reference facility (CREF8) which lists the assembler output with line numbers.An alphabetic variable-name directory is also provided and includes line number references for each occurrence of the variable name.

MACRO 8000 is a powerful macroassembler providing the ability to assemble relocatable AmZ8000 programs. This assembler includes sophisticated features like segmentation, block-structured program organization, algorithmic assignment statements, IF-THEN-ELSE conditional assembly, and execution provisions as well as recursive macro calls.

LINK 8 & LINK 8000 are linkage editors used to combine the relocatable module ready for loading as a program. Link 8 supports the editing of 8 bit segments. The linkers process a series of interactive subcommands that specify files to be linked, files to be searched for satisfaction of unresolved external references, and specific directives which allow a memory map to be printed or direct execution to be initiated.

TRANZ 8000 is an AmZ8000 translator program accepting standard 8080, 8085, and Z80 source code as input and provides standard AmZ8000 source code as output, aiding the user in conversion of existing 8-bit programs to the AmZ8000 environment.



HIGH LEVEL LANGUAGES

PASCAL is a new, up-to-date, language developed for users who are seeking new features to solve today's problems. PASCAL incorporates new features like the concept of variable data types: bits, bytes, words, records, sets, scalars . . . and others that are appropriate to the solution of complex problems. The block structured nature of the language permits the user to create software in a structured environment resulting in a lower development costs, more concise documentation and lower maintenance costs.

The AMC PASCAL is upwards compatible with Jensen and Wirth PASCAL. Extensions provided include:

Interactive files	Comparison of arrays
Untyped files	Comparison of records
Random access of files	Text intrinsics
Strings intrinsics	

The Pascal user can compile a Pascal program and then execute the program on AmSYS 8/8 with the Pascal interpreter and run-time library. The output of this compiler is a special code called P-code and is compatible with the well-known UCSD P-code. The Pascal interpreter executes the P-code instructions when running the Pascal program. The run-time library routines are used as needed in Pascal program execution.

The Pascal user can compile a Pascal program and then generate either 8080, or AmZ8000 code. For 8080 code generation, an 8080 macro library, the MACRO 8 8-bit assembler, and the linker LINK 8 are automatically used. For AmZ8000 code generation, the AmZ8000 macro library, MACRO8000 16-bit assembler, and the linker LINK 8000 are used automatically.

FORTRAN has long been accepted as the standard for scientific programming and is the "native" language of many professional programmers. AMC supports FORTRAN with a compiler conforming to the ANSI 1966 specification for the 8080 microprocessor in the AmSYS 8/8 system.

Full conformance to the ANSI standard insures that accumulated libraries of FORTRAN programs will be immediately usable in the AmSYS 8/8 environment. AMC FORTRAN opens the door to the richest traditions of scientific programming in a small, inexpensive environment.

BASIC is one of the most comprehensive 8-bit BASIC language software programs available today. It contains many unique features not found in other implementations, like:

a. Direct access to the CPU I/O Ports
b. Full Print Using Capability
c. Trace facilities
d. Four variable types — Integer, String, Single (7 digits) and Double (16 digits) precision floating point.

These are only a few of the features available in AMC's BASIC. It also has one of the largest sets of statements making it a very powerful language available to the AmSYS 8/8 user.

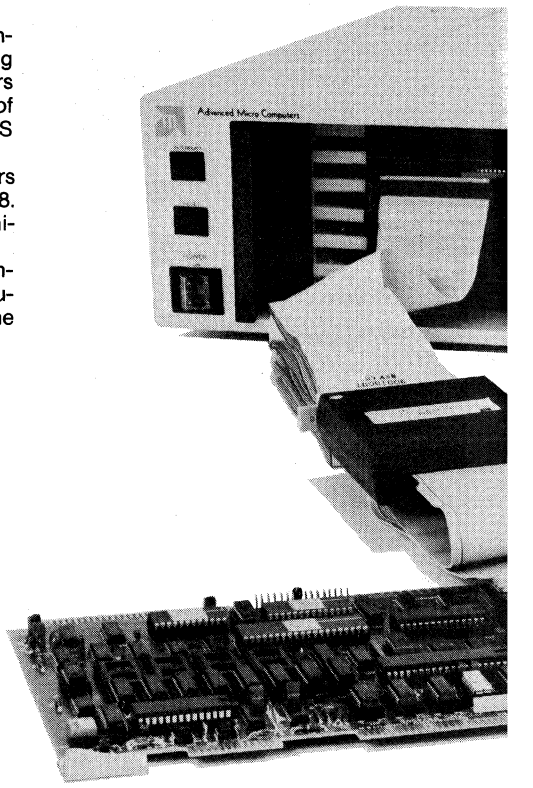
COBOL has been developed in strict accordance to ANSI '74 standards for support of the 8080 within the AmSYS System. At the root of the language is a full ANSI '74 Level 1 COBOL. Beyond that, many Level 2 features have been incorporated to make AMC's COBOL more powerful in every instance where the fundamental speed and size of the package is extremely important. In addition to this, special display-oriented features (ACCEPT, DISPLAY, etc.) have been added to the language in anticipation of a strong emphasis on interactive data entry applications in the microcomputer environment. A powerful interactive debug structure has also been added to greatly decrease program development time.

In-Circuit Emulation

The AmSYS 8/8 Development System provides optional In-Circuit Emulation capability to support a wide range of microprocessors including the AmZ8000, 8080, 8085, Z80, and 8048. The AmSYS series of emulators contain many unique features to assist the designer in the debugging of hardware and software. There are two types of emulators in the AmSYS 8/8 Development System Series. These are:

(1) AmSYS 8/8800 In-Circuit Emulator series for 8-bit microprocessors including individual emulators for the 8080, 8085, Z80, and 8048.
(2) AmSYS 8/8050 In-Circuit Emulator for the AmZ8001 and 8002 microprocessors.

These in-circuit emulators come as complete subsystems and are connected to the AmSYS 8/8 by a high speed serial I/O interface. Each emulator provides its own internal memory storage to give the user real time emulation.



AmSYS 8/8050 16 BIT IN-CIRCUIT EMULATION SUBSYSTEM

The AmSYS RTE 8/8050 Emulator has been designed especially to support the AmZ8000 microprocessor. This subsystem provides real time emulation for both the AmZ8001 segmented and AmZ8002 non-segmented versions.

The AmSYS RTE 8/8050 Emulator has a number of key features including TRACE, 8 trigger points, memory mapping of internal high speed static RAM, and medium speed dynamic RAM providing real time emulation in the user's target system. The AmSYS RTE 8/8050 Emulator has two modes of operation:

1) Interrogation — this mode allows the user to access AmZ8000 resources, registers, I/O ports, and target RAM.
2) Emulation — this mode allows the designer the choice of an internal or user designated clock for software execution or emulation in his target system

RTE 8/8050 provides two types of memory for the user. One type is high speed static memory providing the user with real time emulation at up to 4 MHz for the AmZ8000 with no wait states.It is available in 4K or 8K bytes. The second type is medium speed dynamic RAM available in 64K or 128K byte increments with one wait state at 4 MHz and is expandable up to 192K bytes.

RTE 8/8050 allows for memory mapping on 1K byte boundaries within the AmZ8000 8M byte address space .

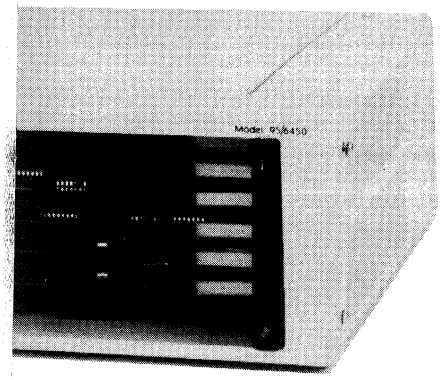
There are up to 8 trigger points available to the user that can be used as breakpoints, enabling patching trace qualifiers or allowing selective trace after the breakpoint is encountered. Two compound breakpoints are also permitted. These allow breakpoints on a value within or outside a specified range, i.e., address, data, address and data, I/O address, I/O data, I/O, address and data.

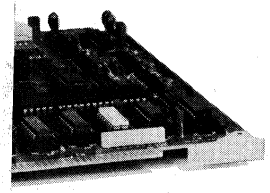
Trace is an integral part of the system. This feature provides for two basic modes of operation. These are:

Micro Trace — this saves Address/Data bus and status together with 8 or 16 user designated probes every machine cycle.

Macro Trace — this saves Address bus and status during T1 displaying address bus in HEX format and status in binary and symbolically decoded form. Also saved is Data Bus during T3 in disassembled form with operand values in HEX format.

The Micro or Macro trace can be enabled only for selected cycles by use of a trigger point match. A user option is available to tag each traced event with a 16 bit number for counting the number of machine states since the counter was enabled. A unique feature of the RTE 8/8050 is the PATCH allowing patching of ROM/RAM code while running in user memory. Up to 4 patches are allowed with up to 16 locations each providing a substantial improvement over conventional patching techniques that require subroutines.



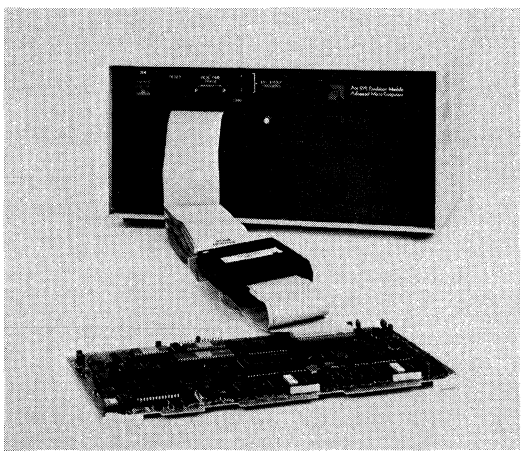


AmSYS 8/8800 8-BIT IN-CIRCUIT EMULATION SUBSYSTEM

The RTE 8/8800 in-circuit emulator is designed to allow replacement of the target microprocesser during the debugging and prototyping phase. It provides the user with real time emulation of each designated microprocessor together with sophisticated debug tools for hardware/software integration resulting in a reduction of overall development time.

AmSYS RTE 8/8800 provides versatile emulation capabilities for the 8080, 8085, Z80A, or 8048 depending upon the personality module being used. It also has the capability to examine and alter registers, memory, and I/O ports. The emulator utilizes the users system clock, thus eliminating potential timing problems caused by separate clocks.

There are up to 8K bytes of high speed Static RAM Emulator memory for mapping on 1K byte boundaries in the target system, thereby utilizing known memory into the target system. This unit has a real time trace for storing the last 128 bus operations as well as the 8 external probes. During selected emulator operations, 16 address lines, 8 data lines, and the clock signal are stored during emulator operation. Disassemblers for each supported microprocessor are provided together with host software for the AmSYS 8/8 Development System.



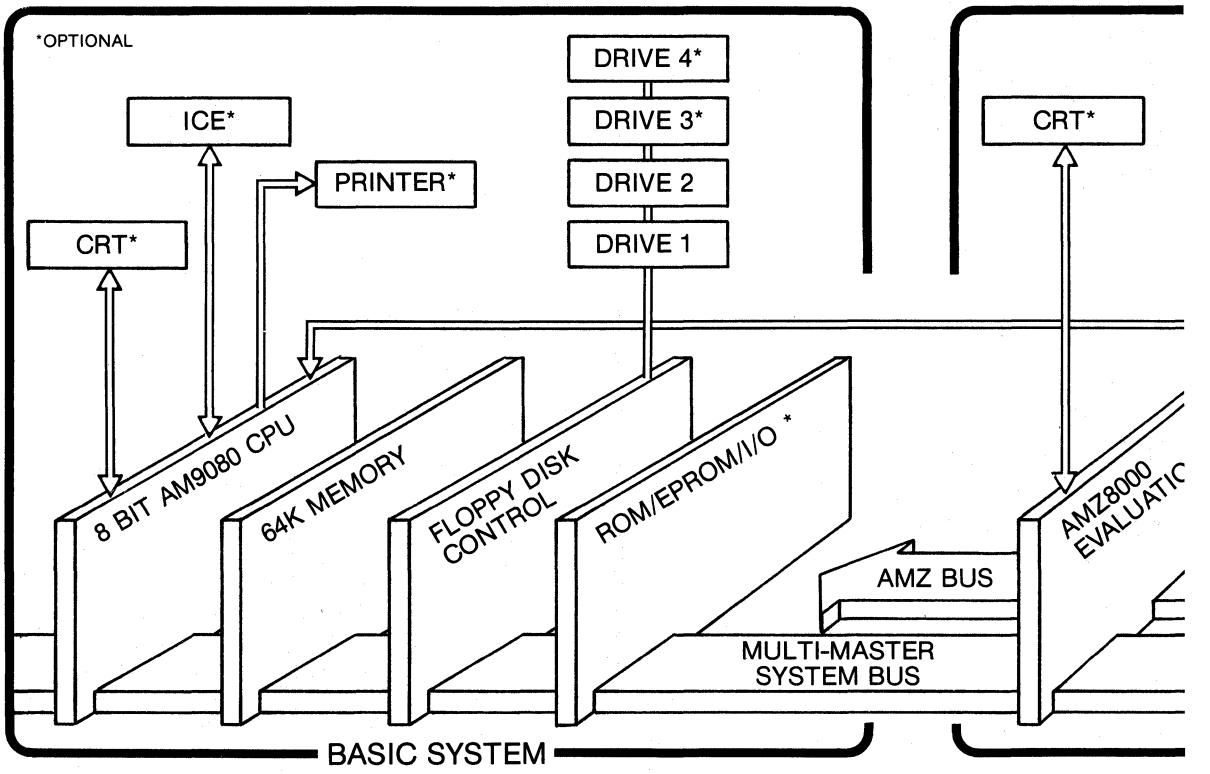
The AmSYS RTE 8/8800 in-circuit emulator consists of a basic sub-system containing a trace module, emulator module, serial communications module and 8K bytes of high speed static RAM. The connection to the AmSYS 8/8 Development System is via a serial I/O port. Each microprocessor personality module is supplied with its own emulator pod attached to a cable terminating in a 40-pin connector to provide the interface to the target system. Host software operating on AmSYS 8/8 provides interrogation mapping and a command structure for the RTE 8/8800 in-circuit emulators.

The trace RAM buffer is 256 words deep and 48 bits wide and is expandable to 1024 words. This unit has the ability to time stamp each entry in the trace buffer.

The RTE 8/8050 Emulation Subsystem is provided as a stand-alone unit with integral power supplies,emulator module, trace module and serial I/O port. Programs prepared on the AmSYS 8/8 development system are down-line loaded to the emulator subsystem under the control of the emulator control processor, which services and issues all emulator commands. RTE 8/8050 Emulator commands and results can be displayed via the AmSYS 8/8 Development System CRT or via a separate CRT console plugged into the RTE 8/8050 Subsystem. (This configuration leaves the AmSYS 8/8 Development System free to perform software development, i.e., assembly, editing, etc.) The RTE 8/8050 Emulator has been designed to give the designer maximum support in system debugging.

The AmSYS 8/8 is Easy to Adapt

The AmSYS 8/8 is designed to allow easy addition or reconfiguration into a more powerful system. These optional additions and configurations allow execution and debug of AmZ8000 programs, the addition of Double Density Floppy Disks, Hard memory expansion, and reconfiguration as an AmZ8000 based development system.



AmZ8000 EXECUTION AND DEBUG (8/8610 and 8620)

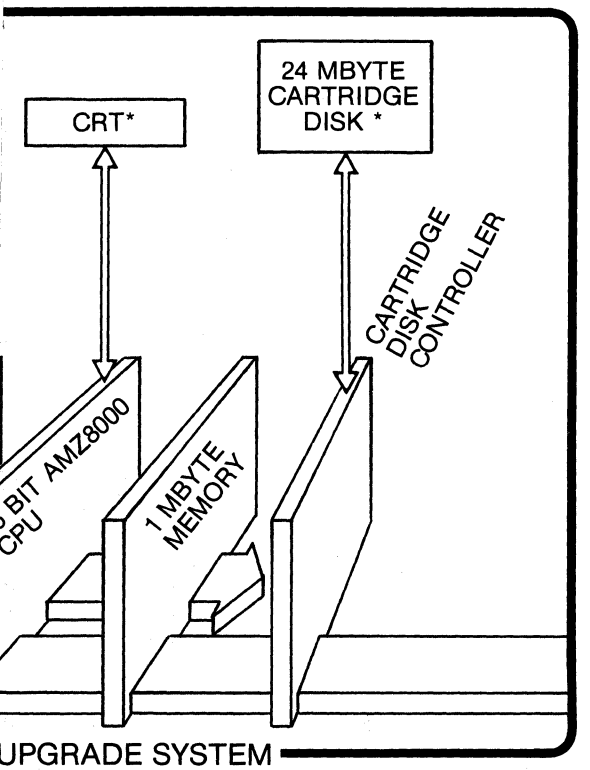
The 8/8610 up-load/down-load package provides an ideal breadboarding, software execution, and debug tool when used with the AmSYS 8/8 development system. It features the AMC 96/4016 AmZ8000 Evaluation Board. AmZ8000 programs generated on AmSYS 8/8 can be down-loaded into or up-loaded from RAM on the Evaluation board where the resident monitor allows execution and debugging to take place under the control of the AmSYS 8/8 console. This Evaluation board provides the user with 8K byte RAM. The 8/8610 can be upgraded to the 8/8620 package with the addition of the 64k RAM board and the AmZ bus motherboard. This expands the RAM storage of the Evaluation Board to 64k. The connection between the RAM board and the Evaluation Board is via the AmZ address and data bus. These options are conveniently packaged as options to AmSYS 8/8.

ADDITIONAL FLOPPY DISK DRIVES

The Floppy Disk controller contained within the AmSYS S 8/8 has the capability of addressing up to four (4) Floppy Disk drives. The addition of the optional 8/8510 Floppy Disk chassis gives the user two (2) additional Floppy Disk drives.

DUAL DENSITY FLOPPY DISK EXPANSION

AmSYS 8/8 can be configured to accept Dual Density Floppy Disk capability. This will give the user Floppy Disk storage capacity of up to 1024K bytes within the AmSYS 8/8. This expansion can be accomplished as a field upgrade.



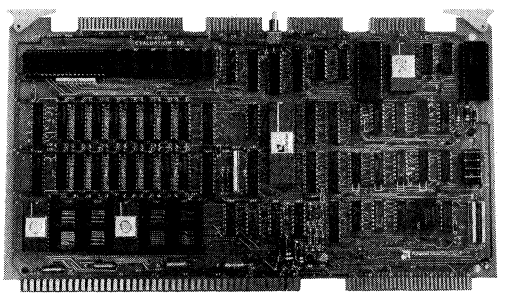
HARD DISK

AmSYS 8/8 is designed for the addition of a Hard Disk with bulk storage of up to 24 megabytes. This Hard Disk option will be available in the near future to be added to the present 8 bit operating system and also as part of an upgrade to the AmZ8000 based development system.

This Hard Disk option will give the user access to larger amounts of bulk storage and also increases his system throughput.

AmZ8000 EVALUATION BOARD

For evaluation of the AmZ8000, the 96/4016 provides a standalone monoboard computer. The 4016 has 8k bytes of dynamic RAM, 2 serial RS-232 ports, 24 parallel I/O lines, and 3-16 bit counters. A 4k byte monitor program provides debug capability and an interface to the optional 8k line by-line assembler. The monitor also provides the drivers necessary to up-load or down-load files through the parallel or serial interface. The 96/4016 is SBC form factor compatible and can plug into AmSYS 8/8 to get its power from the P1 connector. The memory and I/0 on the board can be expanded through the AmZ bus on the P2 connector.



AmZ8000 UPGRADE

The AmSYS 8/8 multimaster bus provides an 8/16 bit data bus with 20 address lines. The peripheral controllers and memory boards all have 8/16 bit compatibility and allow the reconfiguration of the AmSYS 8/8 into a powerful AmZ8000 based development system. The Am9080 board is replaced by the AmZ8000 CPU board, allowing up to 1M byte addressing capability. The addition of AMC's multitasking foreground background system, together with additional 64K or 128K byte RAM memory modules and AmZ8000 software development programs provide the user with a powerful 16-bit system with greatly increased utility and performance.

SPECIFICATIONS

CPU

Am9080 Upgradable to AmZ8000 (see sections entitled "Options")

Memory

64K bytes standard

Disk Storage

512K bytes (2 Single Density Floppy disks, 256K bytes each) (Dual Density is optional) (see section entitled "Optional Upgrades")

I/O Channels

1 serial port RS232 compatible
2 parallel I/O channels consisting of three 8-bit ports each

Interrupt

8 fully programmable vectored channels

AC Power Requirement

60 Hz, 115 VAC std.
100, 120, 220, 240 VAC optional
50/60 Hz Optional

Environmental Requirements

Operating Temperature:	10°C to +40°C
Humidity:	10% - 90% relative (non-condensing)

Physical Characteristics

Dimensions:	24" deep, 17" wide, 11" high
Weight:	60 lbs.
User Panel:	Contains: a) on/off switch, power indicator b) system reset
Chassis:	Desk top stds., rack mountable (optional)

Baud Rates

Selectable 50-9600 Baud

Cooling

Internal fans

Standard System Includes the Following:

Am9080 CPU
64K Bytes RAM
512K Bytes Disk storage (2 single sided floppy disks)
1 RS232 serial port
6 8-bit parallel ports
1 Chassis consisting of 7 Card slots
AmZ8000 Macroassembler
AmZ8000 Translator
8080, 8085, and Z80 Macroassemblers
Linker
Editor
Debugger
AMDOS 8 Disk Operating System
1 set System Documentation
2 Diskettes
Multi-master Bus

Optional Items

Universal Prototyping Board
High-Speed Extender Board
Additional Serial and Parallel I/O Ports
Expansion Chassis with 2 floppy disk drives
CRT Terminal
Line Printers 60 CPS, 120 CPS, and 300 LPM
FORTRAN (8 bit)
Extended BASIC (8 bit)
PASCAL (8 bit and 16 bit)
COBOL (8 bit)
9511/9512 Arithmetic Processing Unit
8048, 8049, 8021 Macroassembler
8048, 8049, 8021 In-Circuit Emulation/TRACE
8080 In-Circuit Emulation/TRACE
8085 In-Circuit Emulation/TRACE
Z80 In-Circuit Emulation/TRACE

Options and Upgrades

(not available with initial shipments—contact your AMD salesman for delivery information)

Z8000 In-Circuit Emulation/TRACE
Double Density Floppy Disk Drives
Expansion to 1 megabyte of Main Memory
AmZ8000 Microcomputer Board Upgrades
Cartridge Disk with Capabilities of 12 or 24 Megabytes
High Level 16-bit languages
Additional Expansion chassis

AmSYS 8/8 Ordering Information

PART NO.	DESCRIPTION
8/8010	Standard AmSYS 8/8 Development System 117V, 60 Hz
8/8012	Standard AmSYS 8/8 Development System 117V, 60 Hz Double Density Floppy Disks
8/8020	Standard AmSYS 8/8 Development System 220V, 50 Hz
8/8022	Standard AmSYS 8/8 Development System 220V, 50 Hz Double Density Floppy Disks
8/8030	Standard AmSYS 8/8 Development System 100V, 50 Hz
8/8032	Standard AmSYS 8/8 Development System 100V, 50 Hz Double Density Disks
8/3310	Serial I/O Board (4 Ports)
8/5032	ROM/EPROM Board I/O Board
8/6410	Univ. Prototyping Board
8/6420	High Speed Extender Board
8/8210	CRT with extra key pad
8/8310	Line Printer 120 CPS
8/8340	Line Printer 300 LPM
8/8410	Fortran 8
8/8420	Basic 8
8/8430	Cobol 8
8/8440	Pascal compiler (with code generators)
8/8610	Up/Down Load Execution Package w/8K bytes
8/8620	Up/Down Load Execution Package w/64K bytes
8/8800	In-Circuit Emulator Subsystem
8/8880	8080 In-Circuit Emulator POD
8/8885	8085 In-Circuit Emulator POD
8/8888	Z80 In-Circuit Emulator POD
8/8848	8048 In-Circuit Emulator POD
8/8050	Z8000 In-Circuit Emulator Subsystem
8/8250	Diskette Package of 10
8/8510	Optional Floppy Disk Chassis (adds add'l two Floppy Disk drives to system)

NOW! Evaluate the features of the AmZ8000 . . . develop software . . . execute programs with THE AMC 96/4016 EVALUATION BOARD.



The new generation of microprocessors is here. Now you can evaluate the AmZ8000 with the system-oriented **AMC 96/4016 Evaluation Board** that makes it easy to utilize the latest microcomputer technology. The **AMC 96/4016 Evaluation Board** puts a versatile and intelligent tool in the hands of engineers, designers and programmers allowing them to explore the exceptional capabilities of the AmZ8000. The **AMC 96/4016** integrates powerful hardware and extensive software resources on an assembled and tested printed-circuit board that allows the evaluation of the AmZ8000 by the addition of a power supply and I/O device. Power can be provided by plugging the board into an SBC 80 type card cage, AMC's development system, or with a lab supply. Two ports are provided to interface to a CRT terminal or to the 96/4016-KBD keyboard/display board.

Some of the features of the **AMC 96/4016 Evaluation Board** are:

- ☐ Fully assembled and tested computer board
- ☐ AmZ8002 Microprocessor – 4MHz operation
- ☐ 8K bytes (4K words) of RAM memory
- ☐ Sockets for up to 12K bytes of PROM/EPROM
- ☐ PROM-based monitor with debugging capability
- ☐ Two serial ports with programmable baud rates
- ☐ 24-line parallel port (three byte-wide ports)
- ☐ Three interval timers
- ☐ Optional PROM-based ASCII Assembler
- ☐ Interfaces for direct I/O to a CRT terminal or the 96/4016-KBD keyboard/display board
- ☐ SBC 80 physical size
- ☐ Can be used as an execution vehicle with the **AmSYS 8/8 Development System**
- ☐ Provides up-load/down-load capability with the **AmSYS 8/8 Development System**

The AmZ8000 architecture – in terms of CPU resources, instruction set, system interface and software-oriented features – represents a major advance in microprocessor sophistication and system-level performance. It is efficient enough to service simple tasks effectively, yet can easily handle complex, high-performance applications as well.

The AmZ8000 architecture and partitioning is well suited for today's technology and for a very wide range of today's applications. It is also a significant departure from the constraints of past architectures, and establishes a clean attractive and nearly open-ended base for evolution and development.

The **AMC 96/4016 Board** makes use of these features and assists software and system engineers in evaluating these features of the AmZ8000 for existing and future needs. Moreover, with the cleaner architecture of this new-generation microprocessor, the Evaluation Board can help hardware and software professionals develop better interface circuits and programs.

ABUNDANT CPU RESOURCES

The AmZ8000 offers sixteen 16-bit general-purpose registers in addition to special system registers. All 16 registers may be used as accumulators and all but one can serve as index registers. The first eight of these 16-bit registers may be used as sixteen 8-bit byte registers if needed. The AmZ8000 also supports seven main data types; bits, BCD digits, bytes, words (16 bits), long words (32 bits), byte strings and word strings. Additionally, many other data elements such as memory addresses, I/O addresses, segment table entries and program status words are also provided.

The **AMC 96/4016** gives the user access to the AmZ8000 and the ability to reduce programming overhead and shorten product and project development time. Hands-on experience with the Evaluation Board can help demonstrate the effectiveness of the AmZ8000 to provide fewer program modifications and less debug time.

Compared to other microprocessors or even 16-bit minicomputers, the number and power of individual instructions have greatly increased. Over 110 distinct instruction types are available with the AmZ8000, compared to approximately 60 for the PDP 11/45. With few exceptions, byte, word and long-word data elements can be processed by all the instructions. Each instruction – again with few exceptions – can use any of the five main addressing modes.

System designers, and especially programmers, will find the **AMC 96/4016 Board** useful in evaluating the AmZ8000's instruction set and its ability to generate higher code densities that can result in significant memory savings and shorter execution times.

HIGHER THROUGHPUT COMPILERS

Many applications normally involve high-level languages, operating systems and data-base management. The AmZ8000, with proven N-channel MOS technology and a 4MHz clock, allows the use of lower-cost dynamic RAMs. The AmZ8000 overlaps instruction execution with next instruction fetch to avoid the problems associated with deep unconditional prefetching.

The AmZ8000 can achieve this high degreee of performance because its regular architecture does not have critical bottlenecks and because the sophisticated instruction set substantially reduces the number of instructions. Some examples of this sophistication are:

a. 32-bit operations (including multiply and divide in single instructions)
b. String manipulation, including compare & translate
c. Block I/O instructions
d. Direct addressing of the entire memory
e. Two operating modes (systems/normal or supervisor/user)
f. Powerful interrupt handling

The **AMC 96/4016 Evaluation Board** makes effective use of these features so designers and managers can interpret these appealing features in specific terms and assess the capability for improved systems that can be designed quicker, easier and with more efficient results.

The AmZ8000 is designed to span a wide variety of applications. It's features allow it to be used effectively in complex high-throughput systems, yet it remains efficient for simpler systems as well.

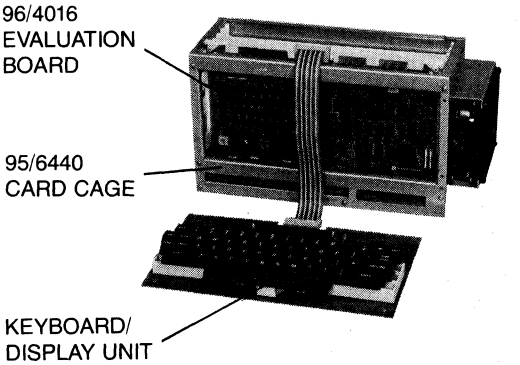
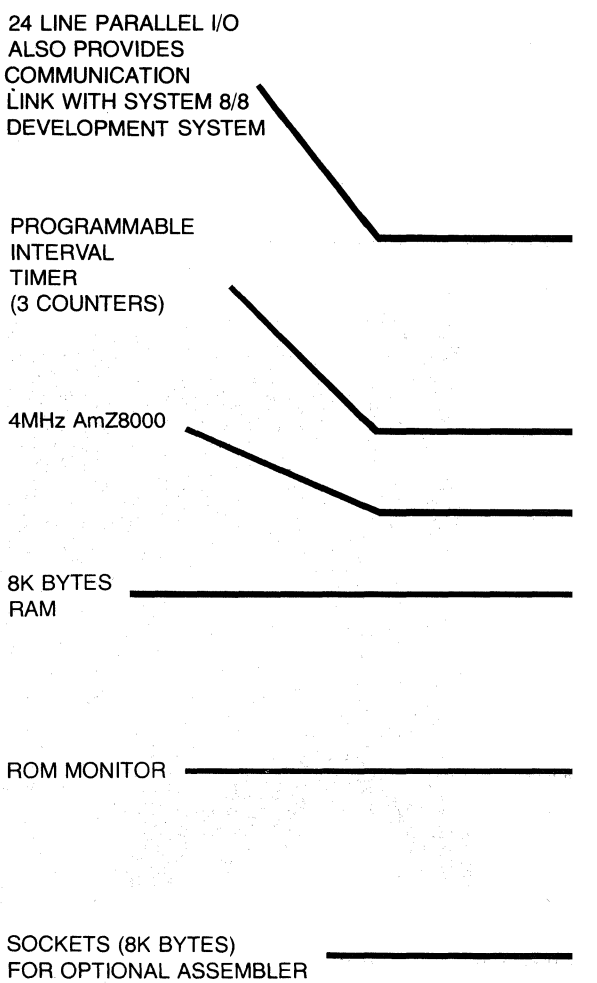
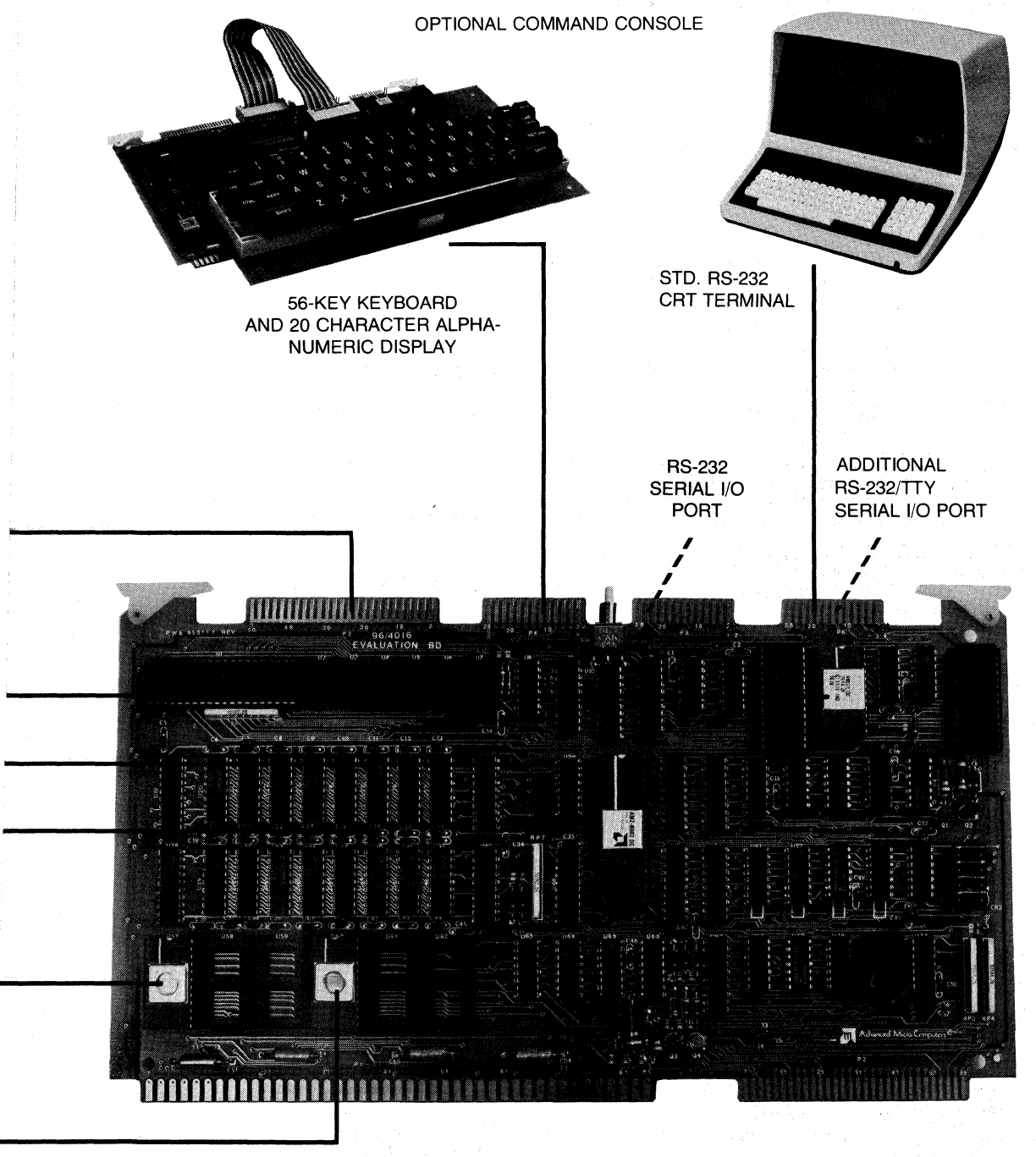


Photo showing use of the AMC 95/6440 Card Cage to house and power the 96/4016 Evaluation Board. Optional Keyboard/Display Unit shown above (or standard CRT Terminal) attaches via cable to edge-card connector. This arrangement provides both convenience and expansion capabilities for specific project and product development.

THE AmZ8000 EVALUATION BOARD NOW FOR BEST RESULTS

SPECIFICATIONS: AMC 96/4016 EVALUATION BOARD	
CPU	AmZ8000, non-segmented
Time Base	Crystal controlled; 3.9936MHz
Serial I/O	Two RS-232C serial ports with software programmable baud rates (50-9600). One port jumper selectable for 20mA operation.
Parallel I/O	24 parallel I/O lines. Also provides interconnection to AmSYS 8/8 Development System.
RAM Memory	8K bytes of on-board dynamic memory; CPU refreshed (transparent).
PROM Space	12K bytes of PROM/ROM Space provided; six sockets; ROM monitor occupies two sockets.
Counter/Timer	Three programmable 16-bit interval counters; two counters used for baud rate control; third counter available for user programs.
Power	+12VDC; −12VDC; +5VDC
Dimensions	12.0" (305 mm) x 6.75" (172 mm); SBC 80 form factor with six edge-card connectors
Memory Mapping	ROM monitor: 0-0FFF(H) PROM space: 0-2FFF(H) RAM space: 4000-5FFF(H)
Environmental Conditions	0 to 55°C ambient in free-air space with relative humidity to 90% without condensation.
Edge-of-Card Connectors	P1: 86-Pin, for power and ground P2: 60-Pin; CPU bus P3: 50-Pin parallel I/O for up/down link as execution vehicle P4: 26-Pin; interface for optional keyboard and display P5: 26-Pin; RS-232 and counter/timer interface P6: 26-Pin; RS-232 or 20mA current loop for CRT terminal or TTY as command console.
Monitor	4K PROM monitor included
Assembler	Optional ASCII, one-pass, line assembler in EPROM
Up/Down-Load Capability	Can be plugged directly into AmSYS 8/8 Development System to provide up-load and down-load capability.
Execution Vehicle	Can be used with other computer systems to execute AmZ8000 code.
Optional Keyboard/ Display	56-key keyboard; 20-character alpha-numeric LED display; same physical form as 96/4016 Board with attaching standoff connectors and interconnecting ribbon cable.




OPTIONAL COMMAND CONSOLE
56-KEY KEYBOARD
AND 20 CHARACTER ALPHA-
NUMERIC DISPLAY
STD. RS-232
CRT TERMINAL
RS-232
SERIAL I/O
PORT
ADDITIONAL
RS-232/TTY
SERIAL I/O PORT
96/4016
EVALUATION BD

12

With Powerful Software Support . . .

AMC 96/4016 ROM MONITOR

The Monitor Program provides the capability to examine and change data in RAM and in any and all of the AmZ8000 registers. It features a hardware breakpoint command along with a single-step routine to provide precise monitoring and debugging capability through step-by-step execution of AmZ8000 code. Additional commands allow users to load registers from memory and execute routines. The combination of 'Breakpoint/Load/Single Step' (with display) provides a versatile and efficient debugging capability. This monitor also supports the Evaluation Board when it is used with Advanced Micro Computer's **AmSYS 8/8 Development System** as an up/down-load device through its SAVE (to disk) and LOAD (disk to RAM) commands. The monitor resides in 4K bytes of EPROM and uses two of the six ROM/EPROM sockets available on the Evaluation Board.

AMC 96/4016 ASSEMBLER

The **AMC 96/4016 Evaluation Board** is supported by a one-pass line-by-line ASCII code Assembler (AMC 96/4016-ASM). This PROM-resident assembler provides the capability to enter symbolic programs into RAM and translates mnemonic op codes, symbolic labels and symbolic or absolute operands to machine code. The assembler reads user-supplied symbolic assembly statements from the command console and assembles the statements as received. Diagnostic messages are displayed when incorrect statements are entered as well as at the end-of-assembly for unsatisfied references with associated locations. Forward references are also permitted. The assembler features:

- ☐ A set of ten special characters for syntax and definition
- ☐ Seven pseudo op codes
- ☐ Six register formats to handle various word lengths
- ☐ Six direct-addressing formats to reference counter locations, labels and strings (ASCII, decimal, hex)
- ☐ Eight diagnostic messages for errors involving syntax, system operation, duplication, overflow and undefined references.

. . . and many ways to use it

AS A STANDALONE MICROCOMPUTER

The **96/4016 Evaluation Board**, along with available options, has the capability of standalone operation. On-board resources, including RAM, I/O interfaces, monitor and interval timer, allow it to serve as a self-contained single-board computer. A terminal or optional keyboard/display can be attached as a command console. Communications to peripherals and other equipment can be achieved through the I/O ports.

EXECUTION DEVICE WITH CROSS-ASSEMBLERS

Many users will want to evaluate the AmZ8000 execution speed and throughput performance with respect to existing programs and system designs. The **AMC 96/4016** is designed to serve as a vehicle to execute AmZ8000 machine codes that may have originated from development systems and cross assemblers. I/O ports on the Evaluation Board provide the necessary parallel and serial interfaces. AmZ8000 CPU signals are brought out to an edge connector which allows the needed connection for other circuitry.

UP-LOAD/DOWN-LOAD CAPABILITY

The **AMC 96/4016 Evaluation Board** features both up-load and down-load capability when used as an execution vehicle with other computer systems, such as the AmSYS 8/8. Programs generated in a development system can be down-loaded to the RAM on the Evaluation Board. Similarly, programs in the board's RAM memory can be up-loaded to the system for further development and for disk storage.

The Evaluation Board is hardware and software compatible with Advanced Micro Computer's **AmSYS 8/8 Development System**. It can plug directly into the AmSYS 8/8 as an execution vehicle to run developed AmZ8000 code. The on-board ROM monitor provides SAVE and LOAD commands that control bidirectional data flow between the RAM on the Evaluation Board and disk files on the Development System.

The **AMC AmSYS 8/8 Development System** contains a comprehensive set of hardware and software resources to fully utilize AmZ8000 capabilities. The system contains dual floppy-disk drives, 64K bytes of RAM, serial and parallel ports, hardware computation and an SBC 80 Multi-Master bus. Existing programming support includes an operating system with linking loader, editor and debugger and, of particular interest, an AmZ8000 macroassembler, 8080 macroassembler and AmZ8000 translator. High-level languages, including PASCAL, are available. See the AmSYS 8/8 brochure for more details or contact one of the sales offices listed on the rear of this brochure.

Advanced Micro Devices International Sales Offices

BELGIUM
Advanced Micro Devices
Overseas Corporation
Avenue de Tervueren, 412, bte 9
B-1150 Bruxelles
Tel: (02) 771 9993
TELEX: 61028

FRANCE
Advanced Micro Devices, S.A.
European Marketing Centre
27, Blvd. Général-Vautrin
F-06400 Cannes
Tel: (093) 43.60.75
TELEX: 470966

Advanced Micro Devices, S.A.
Silic 314, Immeuble Helsinki
74, Rue d'Arcueil
F-94588 Rungis Cedex
Tel: (01) 686.91.86
TELEX: 202053

GERMANY
Advanced Micro Devices
Mikro-Elektronik GmbH
Rosenheimer Str. 139
D-8000 Muenchen 80
Tel: (089) 40 19 76
TELEX: 0-523883

Advanced Micro Devices
Mikro-Elektronik GmbH
Harthaeuser Hauptstrasse 4
D-7024 Filderstadt 3
Tel: (07158) 62 63 0
TELEX: 0-721211

ITALY
Advanced Micro Devices S.R.L.
Centro Direzionale
Palazzo Vasari, 3° Piano
I-20090 MI2 – Segrate
Tel: (02) 215 4913-4-5
Telefax: (02) 656878
TLX: via Milano P.O. 311250PPMI

JAPAN
Advanced Micro Devices, K.K.
Dai 3 Hoya Building
8-17, Kamitakaido 1-Chome
Suginami-ku, Tokyo 168
Tel: (03) 329-2751
TELEX: 2324064

UNITED KINGDOM
Advanced Micro Devices (U.K.) Ltd.
A.M.D. House,
Goldsworth Road,
Woking,
Surrey GU21 1JT
Tel: Woking (24862) 22121
TELEX: 859103

International Sales Representatives and Distributors

ARGENTINA
Thiko, S.A. Electronica
Peru 653
1068 – Buenos Aires
Argentina
Tel: (1) 30-4132
(1) 33-4870
TELEX: 17825

Koti International Corporation
1660 Rollins Road
Burlingame, California 94010
Tel: (415) 692-3200
TELEX: 331491

AUSTRALIA
A.J. Distributors Pty. Ltd.
P. O. Box 71
Prospect, S.A. 5082
Tel: (8) 269-1244
TELEX: 82635

R and D Electronics
257 Burwood Highway
P.O. Box 206
Burwood 3125
Victoria
Tel: (03) 288-8232/62
TELEX: AA33288

R and D Electronics
P.O. Box 57
Crows Nest N.S.W. 2065
Sydney
Tel: (61) 439-5488
TELEX: (790) 25468

AUSTRIA
Kontron Ges.m.b.H.
Industriestr. D 13
A-2345 Brunn am Gebirge
Tel: (02236) 8 66 31
TLX: 79337

BELGIUM
MCA Tronix
62, Route du Condroz
B-4200 Ougree
Tel: (041) 362780/362795
TELEX: 42052

DENMARK
Advanced Electronic of Denmark ApS
55, Mariendalsvej
DK-2000 Copenhagen F
Tel: (01) 19 44 33
TELEX: 224 31

FINLAND
Komdel OY
Box 32
SF-02210 Espoo 21
Tel: (90) 88 50 11
TELEX: 12 1926

FRANCE
A2M
6, Avenue du Général De Gaulle
Hall A
F-78150 Le Chesnay
Tel: (01) 954.91.13
TELEX: 698 376

LED
18, Rue Henri Pensier
F-69352 Lyon Cedex 2
Tel: (078) 58.85.85
TELEX: 300 270

RTF
73, Avenue Charles De Gaulle
F-92200 Neuilly-sur-Seine
Tel: (01) 747.11.01
TELEX: 611985

GERMANY
Cosmos Elektronik GmbH
Hegelstrasse 16
D-8000 Muenchen 83
Tel: (089) 60 20 88
TELEX: 0-522545

EBV-Elektronik Vertriebs GmbH
Oberweg 6
D-8025 Unterhaching b. Muenchen
Tel: (089) 61 10 51
TLX: 0-524535

EBV-Elektronik Vertriebs GmbH
Oststr. 129
D-4000 Duesseldorf
Tel: (0211) 84 84 6
TELEX: 0-8587267

EBV-Elektronik Vertriebs GmbH
Kiebitzrain 18
D-3006 Burgwedel 1
Tel: (05139) 50 38
TLX: 0-923694

EBV-Elektronik Vertriebs GmbH
Myliusstr. 54
D-6000 Frankfurt 1
Tel: (0611) 72 04 16
TELEX: 0-413590

EBV-Elektronik Vertriebs GmbH
Alexanderstr. 42
D-7000 Stuttgart 1
Tel: (0711) 24 74 81
TELEX: 0-722271

Elbatex GmbH
Caecilienstr. 24
D-7100 Heilbronn
Tel: (07131) 89 00 1
TLX: 0-728362

Nordelektronik Vertriebs GmbH
Bahnhofstr. 14
D-2301 Kiel-Raisdorf
Tel: (04307) 54 83

Nordelektronik Vertriebs GmbH
Harksheiderweg 238-240
D-2085 Quickborn
Tel: (04106) 40 31
TELEX: 0-214299

MEV-Mikro Elektronik Vertriebs GmbH
Muenchner Strasse 16A
D-8021 Deining b. Muenchen
Tel: (08170) 71 61
TELEX: 0-527828

HOLLAND
Arcobel BV
Van Almondestraat 6
P.O. Box 344
Oss
Tel: (04120) 24200/27574
TELEX: 50835

HONG KONG
Flint Electronics Co.
Flat A, 10/F
62D Robinson Road
Tel: (5) 255732
TELEX: 63075

INDIA
SRI RAM Associates
778 Blue Sage Drive
Sunnyvale, CA 94086
Tel: (408) 737-2692
TELEX: 348369

Hindustan Semiconductor
24 Sayed Abdulla
Brelvi Road
Bombay
Tel: 818105/812772
TELEX: 011-2726

SRI RAM Associates
245 Jayanagara III Block
Bangalore 560011

ISRAEL
Talviton Electronics
P.O. Box 21104
9, Biltmor Street
Tel-Aviv
Tel: (3) 444572
TELEX: VITKO 33400

ITALY
Indelco, S.R.L. – Roma
Via C. Colombo, 134
I-00147 Roma
Tel: (06) 5140722
TELEX: 611517

Indelco, S.R.L. – Milano
Via S. Simpliciano, 2
I-20121 Milano
Tel: (02) 862963

JAPAN
Advanced Technology Corporation
of Japan
Tashi Bldg., 3rd Floor
No. 8, Minami Motomachi
Shinjuku-ku, Tokyo 160
Tel: (03) 265-9416

Dainichi Electronics
Kohraku Building
1-8, 1-Chome, Kohraku
Bunkyo-ku, Tokyo
Tel: (03) 813-6876

Dainichi Electronics
Kintetsu-Takama Building
38-3 Takama-cho
Narashi 630

ISI Ltd.
8-3, 4-Chome, Lidabashi
Chiyoda-ku, Tokyo 102
Tel: (03) 264-3301

Kanematsu Denshi K.K.
Takanawa Bldg., 2nd Floor
19-26, 3-Chome, Takanawa
Minatoku, Tokyo 108

Microtek, Inc.
Naito Building
7-2-8 Nishishinjuku
Shinjuku-ku, Tokyo 160
Tel: (03) 363-2317
TELEX: J28497

KOREA
Duksung Trading Co.
Room 508 – Sindo Bldg. 65-4
2KA Chungmurq
Seoul
Tel: 778-3198
TELEX: K23459

Leewood International, Inc.
C.P.O. Box 4046
Seoul
Tel: 28-5927
778-5831
TELEX: K25582

NEW ZEALAND
G.T.S. Engineering, Ltd.
P.O. Box 42-053, Orakei
Auckland
Tel: (9) 543 214
(9) 546 745
TELEX: NZ2591

NORWAY
A/S Kjell Bakke
Nygaten 48
P.O. Box 143
N-2011 Stroemmen
Tel: (02) 71 18 72
(02) 71 43 30
TELEX: 19407

SOUTH AFRICA
South Continental Devices (Pty.) Ltd.
P.O. Box 56420
Pinegowrie 2123
Tel: (011) 789-2400
TELEX: 4-24849

SOUTH AMERICA
Intectra
2349 Charleston Road
Mountain View, CA 94043
Tel: (415) 967-8818/25
TELEX: 345 545

SPAIN
Regula S.A.
Avda. de Ramón y Cajal, 5
Madrid-16
Tel: (1) 459 33 00/04/08
TELEX: 42 207

SWEDEN
Svensk Teleindustri AB
Box 5024
S-162 05 Vällingby
Tel: (08) 89 04 35
TELEX: 13033

SWITZERLAND
Kurt Hirt AG
Thurgauerstr. 74
CH-8050 Zuerich
Tel: (00411) 51 21 21
TELEX: 53461

TAIWAN
Multitech America Co.
19046 Bonnet Way
Saratoga, CA 95070
Tel: (408) 867-6201

Multitech International Corp.
2nd Floor
977 Min Shen E. Road
Taipei, 105, R.O.C.
Tel: 768-1232
CABLE: MULTIIC

Everdata International Corp.
11/F 219, Chung Hsiao E. Rd.
Sec. 4, Taipei
Tel: 752-9911
TELEX: 21528

UNITED KINGDOM
Candy Electronic Components
Eden House
32 Well Road
Maidstone, Kent ME14 1XL
Tel: (0622) 54051
TELEX: 965633

Hawke-Cramer Components Ltd.
Hawke House
Green Street
Sunbury-on-Thames
Middlesex TW16 6RA
Tel: (01) 979-7799
TELEX: 923592

Dage (Great Britain), Ltd.
Haywood House
64 High Street
Pinner, Middlesex, HA5 5QA
Tel: (01) 868 0028
TELEX: 24506

ITT Electronic Services
Edinburgh Way
Harlow, Essex CM20 2DF
Tel: Harlow (0279) 26777
TELEX: 81146

Memec, Ltd.
Thame Park Industrial Estate
Thame
Oxon OX9 3RS
Tel: Thame (084 421) 3146
TELEX: 837508

Quarndon Electronics
(Semiconductors) Ltd.
Slack Lane
Derby DE3 3ED
Tel: Derby (0332) 32651
TELEX: 37163



ADVANCED
MICRO
DEVICES, INC.
901 Thompson Place
P.O. Box 453
Sunnyvale,
California 94086
(408) 732-2400
TWX: 910-339-9280
TELEX: 34-6306

