

Am29117

A High-Performance 16-Bit Bipolar Microprocessor

PRELIMINARY

DISTINCTIVE CHARACTERISTICS

- Optimized for High-Performance Controllers**
 Architecture and instruction set optimized for high-performance, intelligent controllers
- Flow-Through Architecture**
 Separate input and output ports avoid bus turnaround for higher throughput
- 32 Working Registers**
 Contains 32 working registers with latched outputs
- Fast**
 Supports 100 ns microcycle time/10 MHz data rate for all instructions
- 16-Bit Barrel Shifter**
 Contains a 16-bit Barrel Shifter which can shift or rotate a word up to 15 positions in a single instruction cycle
- 68-Pin Pin Grid Array Package**

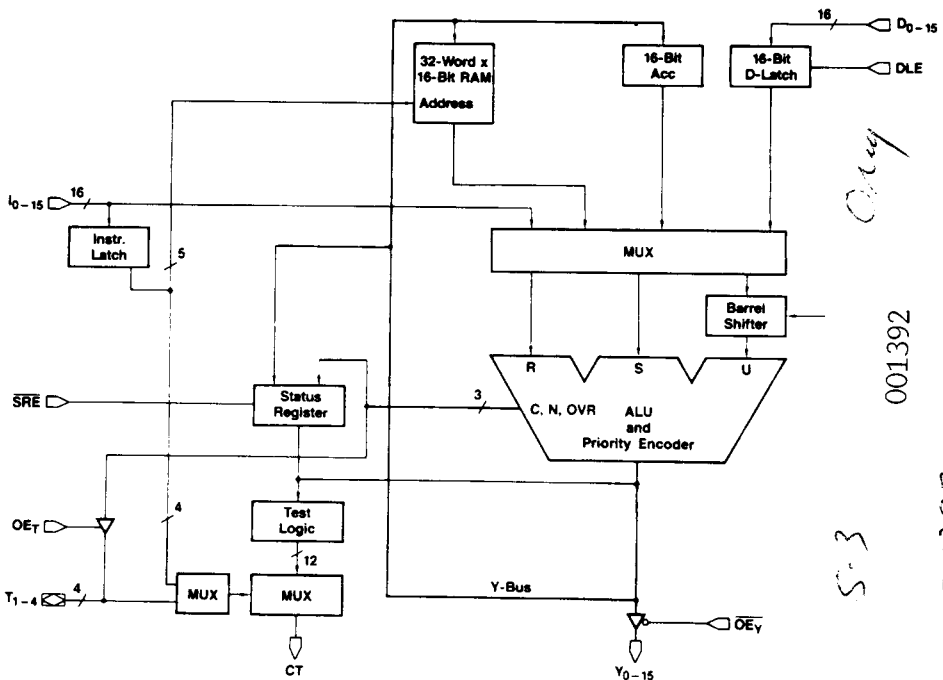
GENERAL DESCRIPTION

The Am29117 is a microprogrammable 16-bit bipolar microprocessor whose architecture and instruction set are identical to the Am29116's except for the I/O bus structure. Since the device has separate input and output ports, designers can avoid quick bus turnaround requirements.

The architecture and instruction set are not only optimized for high-performance peripheral controllers, but also suitable for microprogrammed processor applications when combined with the Am29517A 16 x 16 Multiplier.

The instruction set contains unique functions besides ordinary logic and arithmetic functions: bit manipulation instructions (set, reset and test), rotate merge/compare instructions, prioritize instruction and CRC instruction.

BLOCK DIAGRAM



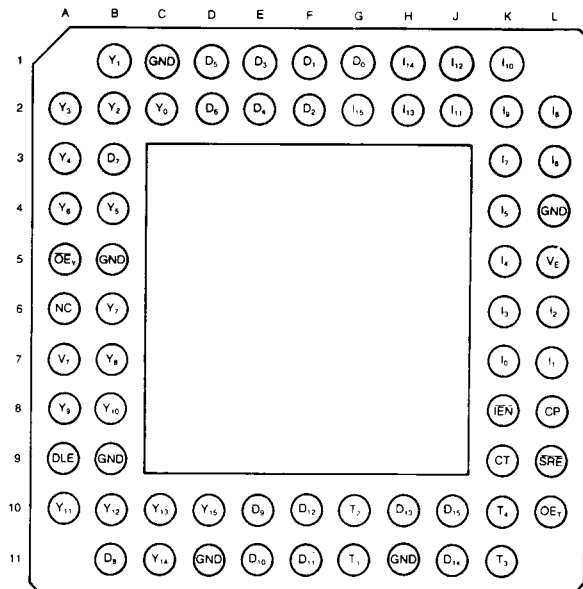
BD001972

Am29117

Advanced Micro Devices

November 1985

Bottom View



Note: Notch indicates orientation.

*Available in 68-pin ceramic LCC

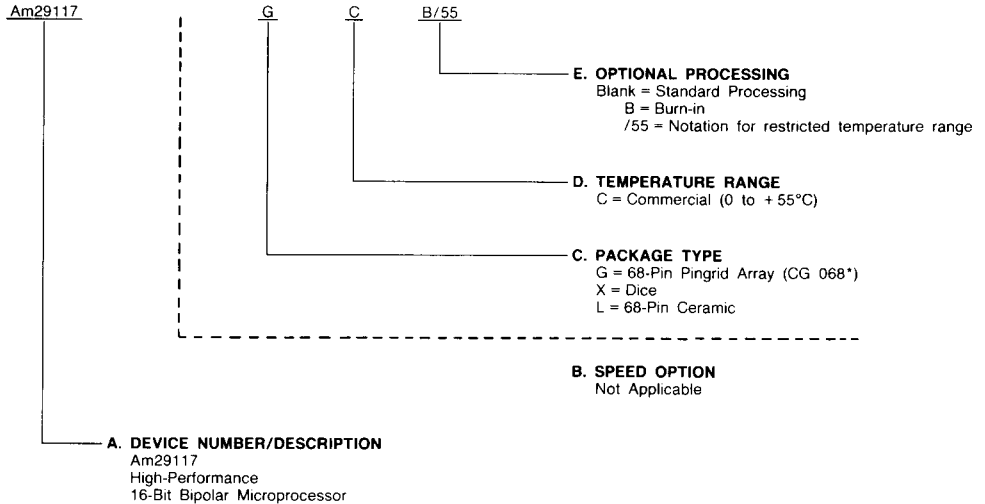
* These GND pads are internally connected inside the package, therefore they do not have external pin numbers.

ORDERING INFORMATION

Standard Products

AMD products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of:

- A. Device Number**
- B. Speed Option** (if applicable)
- C. Package Type**
- D. Temperature Range**
- E. Optional Processing**



*Preliminary. Subject to Change.

| Valid Combinations | |
|--------------------|----------------------------------|
| Am29117 | GC/55, LC/55 GCB/55, XC/55 |

Valid Combinations

Consult the local AMD sales office to confirm availability of specific valid combinations, to check on newly released valid combinations, and to obtain additional data on AMD's standard military grade products.

PIN DESCRIPTION

D₀ – D₁₅ Bidirectional Data (Input)

Data Input Lines, D₀ – D₁₅, are used as external data inputs which allow data to be directly loaded into the 16-bit data latch.

Y₀ – Y₁₅ General Output (Output)

Data Output lines. When \overline{OE}_Y is HIGH, the 16-bit Y outputs are disabled (high impedance); having \overline{OE}_Y LOW allows the ALU data to be output on Y₀ – Y₁₅.

DLE Data Latch Enable (Input)

When \overline{OE}_Y is HIGH, the 16-bit data latch is transparent and is latched when DLE is LOW.

\overline{OE}_Y Output Enable (Input)

When \overline{OE}_Y is HIGH, the 16-bit Y outputs are disabled (high impedance); when \overline{OE}_Y is LOW, the 16-bit Y outputs are enabled (HIGH or LOW).

I₀ – I₁₅ Instruction Inputs (Input)

Sixteen Instruction inputs, used to select the operation to be performed in the Am29117. Also used as data inputs while performing immediate instructions.

\overline{IEN} Instruction Enable (Input)

When \overline{IEN} is LOW, data can be written into RAM when the clock is LOW. The Accumulator can accept data during the LOW-to-HIGH transition of the clock. Having \overline{IEN} LOW, the Status Register can be updated when \overline{SRE} is LOW. With \overline{IEN} is HIGH, the conditional test output, CT, is disabled as a function of the instruction inputs.

\overline{SRE} Status Register Enable (Input)

When \overline{SRE} and \overline{IEN} are both LOW, the Status Register is

updated at the end of all instructions with the exception of NO-OP, Save Status and Test Status. Having either \overline{SRE} or \overline{IEN} HIGH will inhibit the Status Register from changing.

CP Clock Pulse (Input)

The clock input to the Am29117. The RAM latch is transparent when the clock is HIGH. When the clock goes LOW, the RAM output is latched. Data is written into the RAM during the LOW period of the clock, provided \overline{IEN} is LOW, and if the instruction being executed designates the RAM as the destination of operation. The Accumulator and Status Register will accept data on the LOW-to-HIGH transition of the clock if \overline{IEN} is also LOW. The instruction latch becomes transparent when it exits an immediate instruction mode during a LOW-to-HIGH transition of the clock.

T₁ – T₄ Test I/O Pins (Input/Output)

Under the control of OE_T , the four lower status bits, Z, C, N and OVR become outputs on T₁ – T₄, respectively, when OE_T goes HIGH. When OE_T is LOW, T₁ – T₄ are used as inputs to generate the CT output.

OE_T Output Enable (Output)

When OE_T is LOW, 4-bit T outputs are disabled (high impedance); when OE_T is HIGH, the 4-bit T outputs are enabled (HIGH or LOW).

CT Conditional Test (Output)

The condition code multiplexer selects one of the twelve condition code signals and places it on the CT output. A HIGH on the CT output indicates a passed condition and a LOW indicates a failed condition.

FUNCTIONAL DESCRIPTION

Architecture of the Am29117

The Am29117 is a high-performance, microprogrammable 16-bit bipolar microprocessor.

As shown in the Detailed Block Diagram (Figure 1), the device consists of the following elements interconnected with 16-bit data paths.

- 32-Word by 16-Bit RAM
- Accumulator
- Data Latch
- Barrel Shifter
- Arithmetic Logic Unit (ALU)
- Priority Encoder
- Status Register
- Condition-Code Generator/Multiplexer
- Three-State Output Buffers
- Instruction Latch and Decoder

32-Word by 16-Bit RAM

The 32-Word by 16-Bit RAM is a single-port RAM with a 16-bit latch at its output. The latches are transparent when the clock input (CP) is HIGH and latched when the clock input is LOW. Data is written into the RAM while the clock is LOW if the \overline{IEN} input is also LOW and if the instruction being executed defines the RAM as the destination of the operation. For byte instructions, only the lower eight RAM bits are written into; for word instructions, all 16 bits are written into. With the use of an external multiplexer on five of the instruction inputs, it is possible to select separate read and write addresses for the same instruction. This two-address operation is not allowed for immediate instructions.

Accumulator

The 16-bit Accumulator is an edge-triggered register. The Accumulator accepts data on the LOW-to-HIGH transition of the clock input if the \overline{IEN} input is LOW and if the instruction being executed defines the Accumulator as the destination of the operation. For byte instructions, only the lower eight bits of the Accumulator are written into; for word instructions, all 16 bits are written into.

Data Latch

The 16-bit Data Latch holds the data input to the Am29117 on the D bus. The latch is transparent when the DLE input is HIGH and latched when the DLE input is LOW.

Barrel Shifter

A 16-bit Barrel Shifter is used as one of the ALU inputs. This permits rotating data from either the RAM, the Accumulator or the Data Latch up to 15 positions. In the word mode, the Barrel Shifter rotates a 16-bit word; in the byte mode, it rotates only the lower eight bits.

Arithmetic Logic Unit

The Am29117 contains a 16-bit ALU with full carry lookahead across all 16 bits in the arithmetic mode. The ALU is capable of operating on either one, two or three operands, depending upon the instruction being executed. It has the ability to execute all conventional one and two operand operations, such as pass, complement, two's complement, add, subtract, AND, NAND, OR, NOR, EXOR, and EX-NOR. In addition, the ALU can also execute three-operand instructions such as rotate and merge and rotate and compare with mask. All ALU operations can be performed on either a word or byte basis, byte operations being performed on the lower eight bits only.

The ALU produces three status outputs, C (carry), N (negative) and OVR (overflow). The appropriate flags are generated at

the byte or word level, depending upon whether the device is executing in the byte or word mode. The Z (zero) flag, although not generated by the ALU, detects zero at both the byte and word level.

The carry input to the ALU is generated by the Carry Multiplexer which can select an input of zero, one, or the stored carry bit from the Status Register, QC. Using QC as the carry input allows execution of multiprecision addition and subtractions.

Priority Encoder

The Priority Encoder produces a binary-weighted code to indicate the locations of the highest order ONE at its input. The input to the Priority Encoder is generated by the ALU which performs an AND operation on the operand to be prioritized and a mask. The mask determines which bit locations to eliminate from prioritization. In the word mode, if no bit is HIGH, the output is a binary zero. If bit 15 is HIGH, the output is a binary one. Bit 14 produces a binary two, etc. Finally, if only bit 0 is HIGH, a binary 16 is produced.

In the byte mode, bits 8 thru 15 do not participate. If none of bits 7 thru 0 are HIGH, the output is a binary zero. If bit 7 is HIGH a binary one is produced. Bit 6 produces a binary two, etc. Finally, if only bit 0 is HIGH, a binary 8 is produced.

Status Register

The Status Register holds the 8-bit status word. With the Status-Register Enable, (\overline{SRE}) input LOW and the \overline{IEN} input LOW, the Status Register is updated at the end of all instructions except NO-OP, Save-Status and Test-Status instructions. \overline{SRE} going HIGH or \overline{IEN} going HIGH inhibits the Status Register from changing.

The lower four bits of the Status Register contain the ALU status bits of Zero (Z), Carry, (C) Negative (N), and Overflow (OVR). The upper four bits contain a Link bit and three user-definable status bits (Flag 1, Flag 2, Flag 3).

With \overline{SRE} LOW and \overline{IEN} LOW, the lower four status bits are updated after each instruction except those mentioned above, NO-OP, Save Status, Status Test and the Status Set/Reset instruction for the upper four bits. Under the same conditions, the upper four status bits are changed only during their respective Status Set/Reset instructions and during Status Load instructions in the word mode. The Link-Status bit is also updated after each shift instruction.

The Status Register can be loaded from the internal Y-bus, and can also be selected as a source for the internal Y-bus. When the Status Register is loaded in the word mode, all 8-bits are updated; in the byte mode, only the lower 4 bits (Z, C, N, OVR) are updated.

When the Status Register is selected as a source in the word mode, all eight bits are loaded into the lower byte of the destination; the upper byte of the destination is loaded with all zeros. In the byte mode, the Status Register again loads into the lower byte of the destination, but the upper byte remains unchanged. This Store and Load combination allows saving the restoring the Status Register for interrupt and subroutine processing. The four lower status bits (Z, C, N, OVR) can be read directly via the bidirectional T bus. These four bits are available as outputs on the T₁₋₄ outputs whenever OE_T is HIGH.

Condition-Code Generator/Multiplexer

The Condition-Code Generator/Multiplexer contains the logic necessary to develop the 12 condition-code test signals. The multiplexer portion can select one of these test signals and place it on the CT output for use by the microprogram sequence. The multiplexer may be addressed in two different

ways. One way is through the Test Instruction. This instruction specifies the test condition to be placed in the CT output, but does not allow an ALU operation at the same time. The second method uses the bidirectional T bus as an input. This requires extra bits in the microword, but provides the ability to simultaneously test and execute. The test instruction lines, I₀₋₄, have priority over T₁₋₄, for testing status.

Three-State Output Buffers

There are two sets of Three-State Output Buffers in the Am29117. One set controls the 16-bit Y bus. These outputs are enabled by placing a LOW on the \overline{OE}_Y input. A HIGH puts the Y outputs in the high-impedance state, allowing data to be input to the Data latch from an external source.

The second set of Three-State Output Buffers controls the bidirectional 4-bit T bus and is enabled by placing a HIGH on the OE_T input. This allows storing the four internal ALU status bits (Z, C, N, OVR) externally. A LOW OE_T input forces the T outputs into the high-impedance state. External devices can

then drive the T bus to select a test condition for the CT output.

Instruction Latch and Decoder

The 16-bit Instruction Latch is normally transparent to allow decoding of the Instruction Inputs by the Instruction Decoder into the internal control signals for the Am29117. All instructions except Immediate Instructions are executed in a single clock cycle.

Immediate instructions require two clock cycles for execution. During the first clock cycle, the Instruction Decoder recognizes that an Immediate Instruction is being specified and captures the data on the Instruction Inputs in the Instruction Latch. During the second clock cycle, the data on the Instruction Inputs is used as one of the operands for the function specified during the first clock cycle. At the end of the second clock cycle, the Instruction Latch is returned to its transparent state.

Instruction Set

The instruction set of the Am29117 is very powerful. In addition to the single and two operand logical and arithmetic instructions, the Am29117 instruction set contains functions particularly useful in controller applications: bit set, bit reset, bit test, rotate and merge, rotate and compare, and cyclic-redundancy-check (CRC) generation. Complex instructions.

Three data types are supported by the Am29116.

- Bit
- Byte
- Word (16-bit)

In the byte mode data is written into the lower half of the word and the upper half is unchanged. The special case is when the status register is specified as the destination. In the byte mode the LSH (OVR, N, C, Z) of the status register is updated and in the word mode all eight bits of the status register are updated. The status register does not change for save status and test status instructions. In the test status instructions the CT output has the result and the Y-bus is undefined.

The Am29117 Instruction Set can be divided into eleven types of instructions. These are:

- | | |
|--------------------|---------------------------|
| ● Single Operand | ● Rotate and Compare |
| ● Two Operand | ● Prioritize |
| ● Single Bit Shift | ● Cyclic-Redundancy-Check |
| ● Rotate and Merge | ● Status |
| ● Bit Oriented | ● No-Op |
| ● Rotate by n Bits | |

Each instruction type is arbitrarily divided into quadrants. Two of the sixteen instruction lines decode to four quadrants labelled from 0 to 3. The quadrants were defined mainly for convenience in classification of the instruction set and addressing modes and can be used together with the OP CODES to distinguish the instructions.

The following pages describe each of the instruction types in detail. Table 1 illustrates Operand Source-Destination Combinations for each instruction type.

TABLE 1. OPERAND SOURCE-DESTINATION COMBINATIONS

| Instruction Type | Operand Combinations (Note 1) | | |
|--------------------|-------------------------------|----------|--|
| Single Operand | Source (R/S) | | Destination |
| | RAM (Note 2) | | RAM |
| | ACC | | ACC |
| | D | | Y Bus |
| | D(0E) | | Status |
| Two Operand | Source (R) | | Destination |
| | RAM | | ACC |
| | D | | Y Bus |
| | D | | Status |
| | D | | ACC and Status |
| Single Bit Shift | Source (U) | | Destination |
| | RAM | | RAM |
| | ACC | | ACC |
| | D | | Y Bus |
| | D | | ACC |
| Rotate n Bits | Source (U) | | Destination |
| | RAM | | RAM |
| | ACC | | ACC |
| | D | | Y Bus |
| Bit Oriented | Source (R/S) | | Destination |
| | RAM | | RAM |
| | ACC | | ACC |
| | D | | Y Bus |
| Rotate and Merge | Rotated Source (U) | Mask (S) | Non-Rotated Source/ Destination (R) |
| | D | I | ACC |
| | D | RAM | ACC |
| | D | I | RAM |
| | D | ACC | RAM |
| Rotate and Compare | Rotated Source (U) | Mask (S) | Non-Rotated Source/ Destination (R) |
| | D | I | RAM |
| | D | I | RAM |
| | D | ACC | RAM |
| | RAM | I | ACC |

| Instruction Type | Operand Combinations (Note 1) | | |
|-------------------------|-------------------------------|-------------|----------------|
| Prioritize (Note 3) | Source (R) | Mask (S) | Destination |
| | RAM | RAM | RAM |
| | ACC | ACC | ACC |
| | D | I | Y Bus |
| | D | 0 | |
| Cyclic Redundancy Check | Data In | Destination | Polynomial |
| | QLINK | RAM | ACC |
| No Operation | - | | |
| Set Reset Status | Bits Affected | | |
| | OVR, N, C, Z | | |
| | LINK | | |
| | Flag1 | | |
| | Flag2 | | |
| Store Status | Source | | Destination |
| | Status | | RAM |
| | | | ACC |
| | | | Y Bus |
| | | | |
| Status Load | Source (R) | Source (S) | Destination |
| | D | ACC | Status |
| | ACC | I | Status and ACC |
| | D | I | |
| | | | |
| Test Status | Test Condition (CT) | | |
| | (N \oplus OVR) + Z | | |
| | N \oplus OVR | | |
| | Z | | |
| | OVR | | |
| | Low | | |
| | C | | |
| | Z + \bar{C} | | |
| | N | | |
| | LINK | | |
| | Flag 1 | | |
| | Flag 2 | | |
| | Flag 3 | | |

- Notes: 1. When there is no dividing line between the R&S OPERAND or SOURCE and DESTINATION, the two must be used as a given pair. But where there exists such a separation, any combination of them is possible.
2. In the SINGLE OPERAND INSTRUCTION, RAM cannot be used when both ACC and STATUS are designated as a DESTINATION.
3. In the PRIORITIZE INSTRUCTION, OPERAND and MASK must be different sources.

Single Operand Instructions

The Single Operand Instructions contain four indicators: byte or word mode, opcode, source and destination. They are further subdivided into two types. The first type uses RAM as a source or destination or both, and the second type does not use RAM as a source or destination. Both types have different instruction formats as shown below. Under the control of instruction inputs, the desired function is performed on the source and the result is either stored in the specified destination or placed on the Y-bus or both. For a special case where

8-bit to 16-bit conversion is needed, the Am29117 is capable of extending sign bit (D(SE)) or binary zero (D(0E)) over 16-bits in the word mode. The least significant four bits of the Status Register (OVR, N, C, Z) are affected by the function performed in this category. The most significant bits of status register (FLAG1, FLAG2, FLAG3, LINK) are not affected. The only limitation in this type is that the RAM cannot be used as a source when both ACC and the Status Register are specified as a destination.

SINGLE OPERAND FIELD DEFINITIONS

| | | | | | | | | | |
|-----|-----|------|--------|----------|-------------|---|---|---|---|
| | 15 | 14 | 13 | 12 | 9 | 8 | 5 | 4 | 0 |
| SOR | B/W | Quad | Opcode | SRC-Dest | RAM Address | | | | |

| | | | | | |
|------|-----|------|--------|-----|------|
| SONR | B/W | Quad | Opcode | SRC | Dest |
|------|-----|------|--------|-----|------|

SINGLE OPERAND INSTRUCTION

| | | | 15 | 14 | 13 | 12 | 9 | | | | 8 | 5 | | | | 4 | 0 |
|--------------------------|------------------|-------------------|--------|------|------|------------|-------|------------------------------------|-------|--------|-------|--------------------------|-------------|--|--|---|---|
| Instruction ¹ | B/W ² | Quad ³ | Opcode | | | | | R/S ⁴ Dest ⁴ | | | | | RAM Address | | | | |
| SOR | 0 = B 1 = W | 10 | 1100 | MOVE | SRC | ..Dest | 0000 | SORA | RAM | ACC | 00000 | R00 | RAM Reg 00 | | | | |
| | | | 1101 | COMP | SRC | ..Dest | 0010 | SORY | RAM | Y Bus | ... | .. | | | | | |
| | | | 1110 | INC | SRC | + 1 ..Dest | 0011 | SORS | RAM | Status | 11111 | R31 | RAM Reg 31 | | | | |
| | | | 1111 | NEG | SRC | + 1 ..Dest | 0100 | SOAR | ACC | RAM | | | | | | | |
| | | | | | | | 0110 | SODR | D | RAM | | | | | | | |
| | | | | | | | 0111 | SOIR | I | RAM | | | | | | | |
| | | | | | | | 1000 | SOZR | 0 | RAM | | | | | | | |
| | | | | | | | 1001 | SOZER | D(0E) | RAM | | | | | | | |
| | | | | | | | 1010 | SOSER | D(SE) | RAM | | | | | | | |
| | | | | | | | 1011 | SORR | RAM | RAM | | | | | | | |
| Instruction | B/W | Quad | Opcode | | | | | R/S ⁴ | | | | | Destination | | | | |
| SONR | 0 = B 1 = W | 11 | 1100 | MOVE | SRC | ..Dest | 0100 | SOA | ACC | 00000 | NRY | Y Bus | | | | | |
| | | | 1101 | COMP | SRC | ..Dest | 0110 | SOD | D | 00001 | NRA | ACC | | | | | |
| | | | 1110 | INC | SRC | + 1 ..Dest | 0111 | SOI | I | 00100 | NRS | Status ⁵ | | | | | |
| | | | 1111 | NEG | SRC | + 1 ..Dest | 1000 | SOZ | 0 | 00101 | NRAS | ACC, Status ⁵ | | | | | |
| | | | | | | | 1001 | SOZE | D(0E) | | | | | | | | |
| | | | | | 1010 | SOSE | D(SE) | | | | | | | | | | |

- Notes: 1. The instruction mnemonic designates different instruction formats used in the Am29117. They are useful in assembly microcode with the System 29 AMDASM™ meta assembler.
 2. B = Byte Mode, W = Word Mode.
 3. See Instruction Set description.
 4. R = Source; S = Source; Dest = Destination.
 5. When status is destination,
 Status i .. Yi i = 0 to 3 (Byte mode)
 i = 0 to 7 (Word mode)

Y BUS AND STATUS – SINGLE OPERAND INSTRUCTIONS

| Instruction | Opcode | Description | B/W | Y – Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-------------|--------|-----------------|-------|--------------|-------|-------|-------|------|-----|---|---|---|
| SOR | MOVE | SRC ..Dest | 0 = B | Y .. SRC | NC | NC | NC | NC | 0 | U | 0 | U |
| SONR | COMP | SRC ..Dest | 1 = W | Y .. SRC | NC | NC | NC | NC | 0 | U | 0 | U |
| | INC | SRC + 1 .. Dest | | Y .. SRC + 1 | NC | NC | NC | NC | U | U | U | U |
| | NEG | SRC + 1 ..Dest | | Y .. SRC + 1 | NC | NC | NC | NC | U | U | U | U |

SRC = Source
 U = Update
 NC = No Change

0 = Reset
 1 = Set
 i = 0 to 15 when not specified

AMDASM is a registered trademark of Advanced Micro Devices, Inc.

Two Operand Instructions

The Two Operand Instructions contain five indicators: byte or word mode, opcode, R source, S source, and destination. They are further subdivided into two types. The first type uses RAM as the source and/or destination and the second type does not use RAM as source or destination. The first type has two formats; the only difference is in the quadrant. Under the control of instruction inputs, the desired function is performed on the specified sources and the result is stored in the

specified destination or placed on the Y-bus or both. The least significant four bits of the status register (OVR, N, C, Z) are affected by the arithmetic functions performed and only the N and Z bits are affected by the logical functions performed. The OVR and C bits of the status register are forced to ZERO for logical functions. Add with carry and Subtract with carry instructions are useful for Multiprecision Add or Subtract.

TWO OPERAND FIELD DEFINITIONS

| | | | | | | | | | |
|------|-----|------|------------------|--------|-------------|---|---|---|---|
| | 15 | 14 | 13 | 12 | 9 | 8 | 5 | 4 | 0 |
| TOR1 | B/W | Quad | SRC-SRC -Dest | Opcode | RAM Address | | | | |
| TOR2 | B/W | Quad | SRC-SRC -Dest | Opcode | RAM Address | | | | |
| TONR | B/W | Quad | SRC-SRC | Opcode | Dest | | | | |

TWO OPERAND INSTRUCTIONS

| Instruction | B/W | Quad | R ¹ S ¹ Dest ¹ | | | Opcode | | | RAM Address | | | | |
|-------------|----------------|------|---|-------|-----|------------|--------------------|------------|--------------------|-----------|-------|-----|------------|
| TOR1 | 0 = B 1 = W | 00 | 0000 | TORAA | RAM | ACC | ACC | 0000 | SUBR | S minus R | 00000 | R00 | RAM Reg 00 |
| | | | 0010 | TORIA | RAM | I | ACC | 0001 | SUBRC ² | S minus R | ... | ... | ... |
| | | | 0011 | TODRA | D | RAM | ACC | with carry | | | 11111 | R31 | RAM Reg 31 |
| | | | 1000 | TORAY | RAM | ACC | Y Bus | 0010 | SUBS | R minus S | | | |
| | | | 1010 | TORIY | RAM | I | Y Bus | 0011 | SUBSC ² | R minus S | | | |
| | | | 1011 | TODRY | D | RAM | Y Bus | with carry | | | | | |
| | | | 1100 | TORAR | RAM | ACC | RAM | 0100 | ADD | R plus S | | | |
| | | | 1110 | TORIR | RAM | I | RAM | 0101 | ADDC | R plus S | | | |
| | | | 1111 | TODRR | D | RAM | RAM | with carry | | | | | |
| | | | | | | 0110 | AND | R • S | | | | | |
| | | | | | | 0111 | NAND | R • S | | | | | |
| | | | | | | 1000 | EXOR | R ⊕ S | | | | | |
| | | | | | | 1001 | NOR | R + S | | | | | |
| | | | | | | 1010 | OR | R + S | | | | | |
| | | | | | | 1011 | EXNOR | R ⊕ S | | | | | |
| Instruction | B/W | Quad | R ¹ S ¹ Dest ¹ | | | Opcode | | | RAM Address | | | | |
| TOR2 | 0 = B 1 = W | 10 | 0001 | TODAR | D | ACC | ACC | 0000 | SUBR | S minus R | 00000 | R00 | RAM Reg 00 |
| | | | 0010 | TOAIR | ACC | I | RAM | 0001 | SUBRC ² | S minus R | ... | ... | ... |
| | | | 0101 | TODIR | D | I | RAM | with carry | | | 11111 | R31 | RAM Reg 31 |
| | | | | | | 0010 | SUBS | R minus S | | | | | |
| | | | | | | 0011 | SUBSC ² | R minus S | | | | | |
| | | | | | | with carry | | | | | | | |
| | | | | | | 0100 | ADD | R plus S | | | | | |
| | | | | | | 0101 | ADDC | R plus S | | | | | |
| | | | | | | with carry | | | | | | | |
| | | | | | | 0110 | AND | R • S | | | | | |
| | | | | | | 0111 | NAND | R • S | | | | | |
| | | | | | | 1000 | EXOR | R ⊕ S | | | | | |
| | | | | | | 1001 | NOR | R + S | | | | | |
| | | | | | | 1010 | OR | R + S | | | | | |
| | | | | | | 1011 | EXNOR | R ⊕ S | | | | | |

- Notes: 1. R = Source
S = Source
Dest = Destination
2. During subtraction the carry is interpreted as borrow.

TWO OPERAND INSTRUCTIONS

| Instruction | B/W | Quad | R ¹ S ¹ | | | Opcode | | | Destination | | | |
|-------------|----------------|------|-------------------------------|------|-----|--------|------|--------------------|----------------------|-------|------|--------------------------|
| TONR | 0 = B 1 = W | 11 | 0001 | TODA | D | ACC | 0000 | SUBR | S minus R | 00000 | NRY | Y Bus |
| | | | 0010 | TOAI | ACC | I | 0001 | SUBRC ³ | S minus R with carry | 00001 | NRA | ACC |
| | | | 0101 | TODI | D | I | 0010 | SUBS | R minus S | 00100 | NRS | Status ² |
| | | | | | | | 0011 | SUBSC ³ | R minus S with carry | 00101 | NRAS | ACC, Status ² |
| | | | | | | | 0100 | ADD | R plus S | | | |
| | | | | | | | 0101 | ADDC | R plus S with carry | | | |
| | | | | | | | 0110 | AND | $R \cdot S$ | | | |
| | | | | | | | 0111 | NAND | $R \cdot S$ | | | |
| | | | | | | | 1000 | EXOR | $R \oplus S$ | | | |
| | | | | | | | 1001 | NOR | $R + S$ | | | |
| | | | | | | | 1010 | OR | $R + S$ | | | |
| | | | | | | | 1011 | EXNOR | $R \oplus S$ | | | |

- Notes: 1. R = Source
S = Source
2. When status is destination,
Status i. Y_i i = 0 to 3 (Byte mode)
i = 0 to 7 (Word mode)
3. During subtraction the carry is interpreted as borrow.

Y BUS AND STATUS CONTENTS - TWO OPERAND INSTRUCTIONS

| Instruction | Opcode | Description | B/W | Y - Bus | Flag3 | Flag2 | Flag 1 | LINK | OVR | N | C | Z |
|----------------------|--------|--------------------------|-------|--------------------------------|-------|-------|--------|------|-----|---|---|---|
| TOR1 TOR2 TONR | SUBR | S minus R | 0 = B | $Y_- S + \bar{R} + 1$ | NC | NC | NC | NC | U | U | U | U |
| | SUBRC | S minus R with carry | 1 = w | $Y_- S + \bar{R} + QC$ | NC | NC | NC | NC | U | U | U | U |
| | SUBS | R minus S | | $Y_- R + \bar{S} + 1$ | NC | NC | NC | NC | U | U | U | U |
| | SUBSC | R minus S with carry | | $Y_- R + \bar{S} + QC$ | NC | NC | NC | NC | U | U | U | U |
| | ADD | R plus S | | $Y_- R + S$ | NC | NC | NC | NC | U | U | U | U |
| | ADDC | R plus S with carry | | $Y_- R + S + QC$ | NC | NC | NC | NC | U | U | U | U |
| | AND | R · S | | $Y_- R_i \text{ AND } S_i$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | NAND | $\bar{R} \cdot \bar{S}$ | | $Y_i - R_i \text{ NAND } S_i$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | EXOR | R ⊕ S | | $Y_i - R_i \text{ EXOR } S_i$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | NOR | R + S | | $Y_i - R_i \text{ NOR } S_i$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | OR | R + S | | $Y_i - R_i \text{ OR } S_i$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | EXNOR | $\bar{R} \oplus \bar{S}$ | | $Y_i - R_i \text{ EXNOR } S_i$ | NC | NC | NC | NC | 0 | 0 | 0 | U |

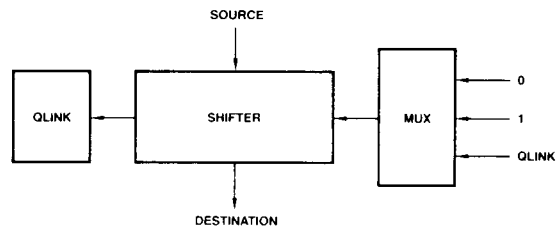
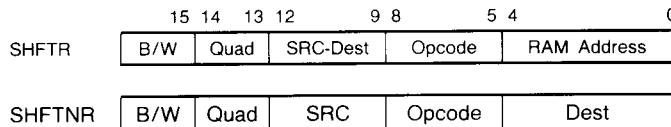
- U = Update
NC = No Change
0 = Reset
1 = Set
i = 0 to 15 when not specified

Single Bit Shift Instructions

The Single Bit Shift Instructions contain four indicators: byte or word mode, direction and shift linkage, source and destination. They are further subdivided into two types. The first type uses RAM as the source and/or destination and the second type does not use RAM as source or destination. Under the control of the instruction inputs, the desired shift function is performed on the specified source and the result is stored in the specified destination or placed on the Y-bus or both. The direction and shift linkage indicator defines the direction of the shift (up or down) as well as what will be shifted into the vacant bit. On a shift-up instruction, the LSB may be loaded with ZERO, ONE,

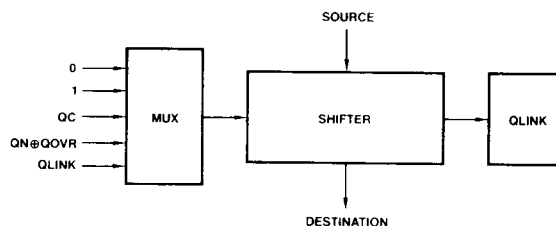
or the Link-Status bit (QLINK). The MSB is loaded into the Link-Status bit as shown in Figure 2. On a shift-down instruction, the MSB may be loaded with ZERO, ONE, the contents of the Status Carry flip-flop, (QC), the Exclusive-OR of the Negative-Status bit and the Overflow-Status bit ($QN \oplus QOVR$) or the Link-Status bit. The LSB is loaded into the Link-Status bit as shown in Figure 3. The N and Z bits of the Status register are affected but the OVR and C bits are forced to ZERO. The Shift-Down with $QN \oplus QOVR$ is useful for Two's Complement multiplication.

SINGLE BIT SHIFT FIELD DEFINITIONS:



PF000360

Figure 2. Shift Up Function



PF000350

Figure 3. Shift Down Function

SINGLE BIT SHIFT INSTRUCTIONS

SINGLE BIT SHIFT

| Instruction | B/W | Quad | U ¹ | Dest ¹ | Opcode | RAM Address |
|-------------|-------|------|----------------|-------------------|--------------------------|----------------------|
| SHFTR | 0 = B | 10 | 0110 | SHRR RAM | 0000 SHUPZ Up 0 | 00000 R00 RAM Reg 00 |
| | 1 = W | | 0111 | SHDR D RAM | 0001 SHUP1 Up 1 | ... |
| | | | | | 0010 SHUPL Up QLINK | 11111 R31 RAM Reg 31 |
| | | | | | 0100 SHDNZ Down 0 | |
| | | | | | 0101 SHDN1 Down 1 | |
| | | | | | 0110 SHDNL Down QLINK | |
| | | | | | 0111 SHDNC Down QC | |
| | | | | | 1000 SHDNOV Down QN⊕QOVR | |
| Instruction | B/W | Quad | U ¹ | | Opcode | Destination |
| SHFTNR | 0 = B | 11 | 0110 | SHA ACC | 0000 SHUPZ Up 0 | 00000 NRY Y Bus |
| | 1 = W | | 0111 | SHD D | 0001 SHUP1 Up 1 | 00001 NRA ACC |
| | | | | | 0010 SHUPL Up QLINK | |
| | | | | | 0100 SHDNZ Down 0 | |
| | | | | | 0101 SHDN1 Down 1 | |
| | | | | | 0110 SHDNL Down QLINK | |
| | | | | | 0111 SHDNC Down QC | |
| | | | | | 1000 SHDNOV Down QN⊕QOVR | |

Note 1. U = Source
Dest = Destination

Y BUS AND STATUS - SINGLE BIT SHIFT INSTRUCTIONS

| Instruction | Opcode | Description | B/W | Y - Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-------------|--------|--------------|-------|--|-------|-------|-------|---------------------|-----|-------------------|---|---|
| SHR SHNR | SHUPZ | Up 0 | 1 = W | Y _i - SRC _{i-1} , i = 1 to 15; Y ₀ - Shift Input | NC | NC | NC | SRC ₁₅ * | 0 | SRC ₁₄ | 0 | U |
| | SHUP1 | Up 1 | 0 = B | Y _i - SRC _{i-1} , i = 1 to 7; Y ₀ - Shift Input; Y ₈ - SRC ₇ , Y _i - SRC _{i-8} for i = 9 to 15 | NC | NC | NC | SRC ₇ * | 0 | SRC ₆ | 0 | U |
| | SHUPL | Up QLINK | | | | | | | | | | |
| | SHDNZ | Down 0 | 1 = W | Y _i - SRC _{i+1} , i = 0 to 14; Y ₁₅ - Shift Input | NC | NC | NC | SRC ₀ * | 0 | Shift Input | 0 | U |
| | SHDN1 | Down 1 | 0 = B | Y _i - SRC _{i+1} , i = 0 to 6; Y _i - SRC _{i-7} , i = 8 to 14; Y _{7,15} - Shift Input | NC | NC | NC | SRC ₀ * | 0 | Shift Input | 0 | U |
| | SHDNL | Down QLINK | | | | | | | | | | |
| | SHDNC | Down QC | | | | | | | | | | |
| | SHCNOV | Down QN⊕QOVR | | | | | | | | | | |

SRC = Source
U = Update
NC = No Change
0 = Reset
1 = Set
i = 0 to 15 when not specified

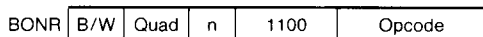
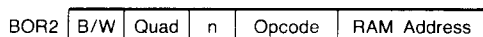
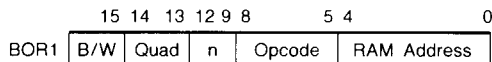
*Shifted Output is loaded into the QLINK.

Bit Oriented Instructions

The Bit Oriented Instructions contain four indicators: byte or word mode, operation, source/destination, and the bit position of the bit to be operated on (Bit 0 is the least significant bit). They are further subdivided into two types. The first type uses the RAM as both source and destination and has two kinds of formats which differ only by quadrant. The second type does not use the RAM as a source or a destination. Under the control of the instruction inputs, the desired function is performed on the specified source and the result is stored in the specified destination or placed on the Y-bus or both. The operations which can be performed are: Set Bit n which forces the n^{th} bit to a ONE leaving other bits unchanged; Reset Bit n

which forces the n^{th} bit to ZERO leaving the other bits unchanged; Test Bit n , which sets the ZERO Status Bit depending on the state of bit n leaving all the bits unchanged; Load 2^n , which loads ONE in Bit position n and ZERO in all other bit positions; Load $\overline{2^n}$ which loads ZERO in bit position n and ONE in all other bit positions; increment by 2^n , which adds 2^n to the operand; and decrement by 2^n which subtracts 2^n from the operand. For all the Load, Set, Reset and Test instructions, the N and Z bits are affected and OVR and C bit of the Status register are forced to ZERO. For all arithmetic instructions the LSH (OVR, C, N, Z bits) of the Status register is affected.

BIT ORIENTED FIELD DEFINITIONS



BIT ORIENTED INSTRUCTIONS

| Instruction | B/W | Quad | n | Opcode | | | RAM Address | | |
|-------------|----------------|------|---------|--------|--------|------------------------|--------------------------|-----|------------|
| BOR1 | 0 = B 1 = W | 11 | 0 to 15 | 1101 | SETNR | Set RAM, bit n | 00000 | R00 | RAM Reg 00 |
| | | | | 1110 | RSTNR | Reset RAM, bit n | ... | ... | ... |
| | | | | 1111 | TSTNR | Test RAM, bit n | 11111 | R31 | RAM Reg 31 |
| Instruction | B/W | Quad | n | Opcode | | | RAM Address | | |
| BOR2 | 0 = B 1 = W | 10 | 0 to 15 | 1100 | LD2NR | 2^n - RAM | 00000 | R00 | RAM Reg 00 |
| | | | | 1101 | LDC2NR | $\overline{2^n}$ - RAM | ... | ... | ... |
| | | | | 1110 | A2NR | RAM plus 2^n - RAM | ... | ... | ... |
| | | | | 1111 | S2NR | RAM minus 2^n - RAM | 11111 | R31 | RAM Reg 31 |
| Instruction | B/W | Quad | n | Opcode | | | | | |
| BONR | 0 = B 1 = W | 11 | 0 to 15 | 1100 | 00000 | TSTNA | Test ACC, bit n | | |
| | | | | | 00001 | RSTNA | Reset ACC, bit n | | |
| | | | | | 00010 | SETNA | Set ACC, bit n | | |
| | | | | | 00100 | A2NA | ACC plus 2^n - ACC | | |
| | | | | | 00101 | S2NA | ACC minus 2^n - ACC | | |
| | | | | | 00110 | LD2NA | 2^n - ACC | | |
| | | | | | 00111 | LDC2NA | $\overline{2^n}$ - ACC | | |
| | | | | | 10000 | TSTND | Test D, bit n | | |
| | | | | | 10001 | RSTND | Reset D, bit n | | |
| | | | | | 10010 | SETND | Set D, bit n | | |
| | | | | | 10100 | A2NDY | D plus 2^n - Y BUS | | |
| | | | | | 10101 | S2NDY | D minus 2^n - Y BUS | | |
| | | | | | 10110 | LS2NY | 2^n - Y BUS | | |
| | | | | | 10111 | LDC2NY | $\overline{2^n}$ - Y BUS | | |

BIT ORIENTED INSTRUCTIONS
Y BUS AND STATUS – BIT ORIENTED INSTRUCTIONS

| Instruction | Opcode | Description | B/W | Y – Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-------------|--------|----------------------------|----------------|--|-------|-------|-------|------|-----|---|---|---|
| BOR1 | SETNR | Set RAM Bit n | 0 = B 1 = W | $Y_i \leftarrow RAM_i$ for $i \neq n$; $Y_n \leftarrow 1$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | RSTNR | Reset RAM, Bit n | | $Y_i \leftarrow RAM_i$ for $i \neq n$; $Y_n \leftarrow 0$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | TSTNR | Test Ram, Bit n | | $Y_i \leftarrow 0$ for $i \neq n$; $Y_n \leftarrow SRC_n$ | NC | NC | NC | NC | 0 | U | 0 | U |
| BOR2 | LD2NR | $2^n \leftarrow RAM$ | | $Y_i \leftarrow 0$ for $i \neq n$; $Y_n \leftarrow 1$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | LDC2NR | $2^n \leftarrow RAM$ | | $Y_i \leftarrow 1$ for $i \neq n$; $Y_n \leftarrow 0$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | A2NR | $RAM + 2^n \leftarrow RAM$ | | $Y_i \leftarrow RAM + 2^n$ | NC | NC | NC | NC | U | U | U | U |
| | S2NR | $RAM - 2^n \leftarrow RAM$ | | $Y_i \leftarrow RAM - 2^n$ | NC | NC | NC | NC | U | U | U | U |
| BONR | TSTNA | Test ACC, Bit n | | $Y_i \leftarrow 0$ for $i \neq n$; $Y_n \leftarrow ACC_n$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | RSTNA | Reset ACC, Bit n | | $Y_i \leftarrow ACC_i$ for $i \neq n$; $Y_n \leftarrow 0$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | SETNA | Set ACC, Bit n | | $Y_i \leftarrow ACC_i$ for $i \neq n$; $Y_n \leftarrow 1$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | A2NA | $ACC + 2^n \leftarrow ACC$ | | $Y_i \leftarrow ACC + 2^n$ | NC | NC | NC | NC | U | U | U | U |
| | S2NA | $ACC - 2^n \leftarrow ACC$ | | $Y_i \leftarrow ACC - 2^n$ | NC | NC | NC | NC | U | U | U | U |
| | LD2NA | $2^n \leftarrow ACC$ | | $Y_i \leftarrow 0$ for $i \neq n$; $Y_n \leftarrow 1$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | LDC2NA | $2^n \leftarrow ACC$ | | $Y_i \leftarrow 1$ for $i \neq n$; $Y_n \leftarrow 0$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | TSTND | Test D, Bit n | | $Y_i \leftarrow 0$ for $i \neq n$; $Y_n \leftarrow D_n$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | RSTND | Reset D, Bit n* | | $Y_i \leftarrow D_i$ for $i \neq n$; $Y_n \leftarrow 0$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | SETND | Set D, Bit n* | | $Y_i \leftarrow D_i$ for $i \neq n$; $Y_n \leftarrow 1$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | A2NDY | $D + 2^n \leftarrow Y$ Bus | | $Y \leftarrow D + 2^n$ | NC | NC | NC | NC | U | U | U | U |
| | S2NDY | $D - 2^n \leftarrow Y$ Bus | | $Y \leftarrow D - 2^n$ | NC | NC | NC | NC | U | U | U | U |
| | LD2NY | $2^n \leftarrow Y$ Bus | | $Y_i \leftarrow 0$ for $i \neq n$; $Y_n \leftarrow 1$ | NC | NC | NC | NC | 0 | U | 0 | 0 |
| | LDC2NY | $2^n \leftarrow Y$ Bus | | $Y_i \leftarrow 1$ for $i \neq n$; $Y_n \leftarrow 0$ | NC | NC | NC | NC | 0 | U | 0 | 0 |

SRC = Source

U = Update

NC = No Change

0 = Reset

1 = Set

i = 0 to 15 when not specified

*Destination is not D Latch but Y Bus.

Rotate by n Bits Instructions

The Rotate by n Bits Instructions contain four indicators: byte or word mode, source, destination and the number of places the source is to be rotated. They are further subdivided into two types. The first type uses RAM as a source and/or a destination and the second type does not use RAM as a source or destination. The first type has two different formats and the only difference is in the quadrant. The second type has only one format as shown in the table. Under the control of instruction inputs, the n indicator specifies the number of bit positions the source is to be rotated up (0 to 15), and the result

is either stored in the specified destination or placed on the Y-bus or both. An example of this instruction is given in Figure 4. In the Word mode, all 16 bits are rotated up while in the Byte mode, only lower 8 bits (0-7) are rotated up. In the Word mode, a rotate up by n bits is equivalent to a rotate down by (16-n) bits. Similarly, in the Byte mode a rotate up by n bits is equivalent to a rotate down by (8-n) bits. The N and Z bits of the Status Register are affected and OVR and C bits are forced to ZERO.

EXAMPLE: n = 4, Word Mode

| | | | | |
|-------------|------|------|------|------|
| Source | 0001 | 0011 | 0111 | 1111 |
| Destination | 0011 | 0111 | 1111 | 0001 |

EXAMPLE: n = 4, Byte Mode

| | | | | |
|-------------|------|------|------|------|
| Source | 0001 | 0011 | 0111 | 1111 |
| Destination | 0001 | 0011 | 1111 | 0111 |

ROTATE BY n BITS FIELD DEFINITIONS

| | | | | | | | | | | | |
|-------|-----|------|----|----------|----------|---------|--|---|---|--|---|
| | 15 | 14 | 13 | 12 | 9 | 8 | | 5 | 4 | | 0 |
| ROTR1 | B/W | Quad | n | SRC-Dest | RAM | Address | | | | | |
| ROTR2 | B/W | Quad | n | SRC-Dest | RAM | Address | | | | | |
| ROTNR | B/W | Quad | n | 1100 | SRC-Dest | | | | | | |

Figure 4. Rotate by n Example

ROTATE BY n BITS INSTRUCTIONS

| Instruction | B/W | Quad | n | U ¹ | Dest ¹ | RAM Address |
|-------------|----------------|------|---------|-------------------------------------|----------------------------|--|
| ROTR1 | 0 = B 1 = W | 00 | 0 to 15 | 1100 RTRA 1110 RTRY 1111 RTRR | RAM ACC Y Bus RAM | 00000 R00 RAM Reg 00 11111 R31 RAM Reg 31 |
| Instruction | B/W | Quad | n | U ¹ | Dest ¹ | RAM Address |
| ROTR2 | 0 = B 1 = W | 01 | 0 to 15 | 0000 RTAR 0001 RTDR | ACC D RAM RAM | 00000 R00 RAM Reg 00 11111 R31 RAM Reg 31 |
| Instruction | B/W | Quad | n | U ¹ | Dest ¹ | RAM Address |
| ROTNR | 0 = B 1 = W | 11 | 0 to 15 | 1100 | | 11000 RTDY D Y Bus 11001 RTDA D ACC 11100 RTAY ACC Y Bus 11101 RTAA ACC ACC |

Note: 1. U = Source
Dest = Destination

Y BUS AND STATUS - ROTATE BY n BITS INSTRUCTIONS

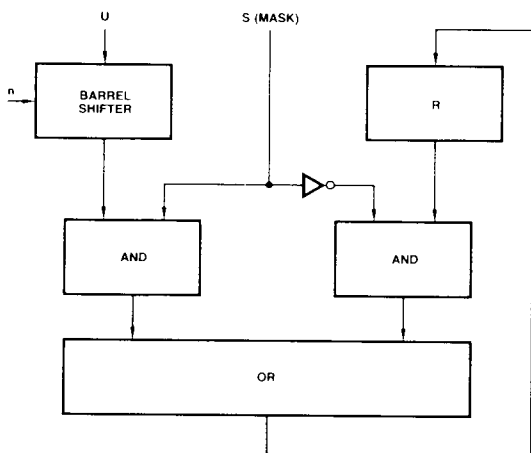
| Instruction | Op-code | B/W | Y - Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-------------|---------|-------|--|-------|-------|-------|------|-----|--------------|---|---|
| ROTR1 | | 1 = W | $Y_i \leftarrow SRC_{(i-n) \bmod 16}$ | NC | NC | NC | NC | 0 | SRC_{15-n} | 0 | U |
| ROTR2 | | 0 = B | $Y_i \leftarrow SRC_i + 8 = SRC_{(i-n) \bmod 8}$ for i = 0 to 7 | NC | NC | NC | NC | 0 | SRC_{8-n} | 0 | U |
| ROTNR | | | | | | | | | | | |

SRC = Source
U = No Change
0 = Reset
1 = Set
i = 0 to 15 when not specified

Rotate and Merge Instruction

The Rotate and Merge Instructions contain five indicators: byte or word mode, rotated source, non-rotated source/destination, mask and the number of bit positions a source is to be rotated. The function performed by the Rotate and Merge instruction is illustrated in Figure 5. The rotated source, U, is rotated up by the Barrel Shifter n places. The mask input then selects, on a bit by bit basis, the rotated U input or R

input. A ZERO in bit i of the mask will select the i^{th} bit of the R input as the i^{th} output bit, while ONE in bit i will select the i^{th} rotated U input as the output bit. The output word is stored in the non-rotated operand location. The N and Z bits are affected. The OVR and C bits of the Status register are forced to ZERO. An example of this instruction is given in Figure 6.



PF000630

Figure 5. Rotate and Merge Function

ROTATE AND MERGE FIELD DEFINITIONS:

| | 15 | 14 | 13 | 12 | 9 | 8 | 5 | 4 | 0 |
|------|-----|------|----|----------------------------------|---|---|-------------|---|---|
| ROTM | B/W | Quad | n | ROT SRC- Non ROT SRC- Mask | | | RAM Address | | |

EXAMPLE: $n = 4$, Word Mode

| | | | | |
|-------------|------|------|------|------|
| U | 0011 | 0001 | 0101 | 0110 |
| Rotated U | 0001 | 0101 | 0110 | 0011 |
| R | 1010 | 1010 | 1010 | 1010 |
| Mask (S) | 0000 | 1111 | 0000 | 1111 |
| Destination | 1010 | 0101 | 1010 | 0011 |

Figure 6. Rotate and Merge Example

ROTATE AND MERGE INSTRUCTION

| Instruction | B/W | Quad | n | U ¹ R/Dest ¹ S ¹ | | | RAM Address | | |
|-------------|----------------|------|---------|---|------|-----|-------------|-----|--|
| ROTM | 0 = B 1 = W | 01 | 0 to 15 | 0111 | MDAI | D | ACC | I | 00000 R00 RAM Reg 00 11111 R31 RAM Reg 31 |
| | | | | 1000 | MDAR | D | ACC | RAM | |
| | | | | 1001 | MDRI | D | RAM | I | |
| | | | | 1010 | MDRA | D | RAM | ACC | |
| | | | | 1100 | MARI | ACC | RAM | I | |
| | | | | 1110 | MRAI | RAM | ACC | I | |

U = Rotated Source
R/Dest = Non-Rotated Source and Destination
S = Mask

Y BUS AND STATUS - ROTATE AND MERGE INSTRUCTIONS

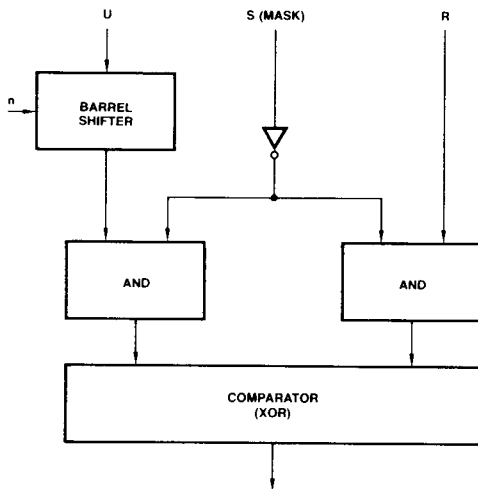
| Instruction | Opcode | B/W | Y - Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-------------|--------|-----|--|-------|-------|-------|------|-----|---|---|---|
| ROTM | | 1=W | $Y_i - (\text{Non Rot Op})_i \cdot (\text{mask})_i + (\text{Rot Op})_{(i-n) \bmod 16} \cdot (\text{mask})_i$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | | 0=B | $Y_i - (\text{Non Rot Op})_i \cdot (\text{mask})_i + (\text{Rot Op})_{(i-n) \bmod 8} \cdot (\text{mask})_i$ | NC | NC | NC | NC | 0 | U | 0 | U |

U = Update
NC = No Change
0 = Reset
1 = Set
 $i = 0$ to 15 when not specified

Rotate and Compare Instructions

The Rotate and Compare Instructions contain five indicators: byte or word mode, rotated source, non-rotated source, mask, and the number of bit positions the rotated source is to be rotated up. Under the control of instruction inputs, the function performed by the Rotate and Compare instruction is illustrated in Figure 7. The rotated operand is rotated by the Barrel Shifter n places. The mask is inverted and ANDed on a bit-by-bit basis

with the output of the Barrel Shifter and R input. Thus, a ONE in the mask input eliminates that bit from the comparison. A ZERO allows the comparison. If the comparison passes, the Zero flag is set. If it fails, the Zero flag is reset. The N and Z bit are affected. The OVR and C bits of the Status register are forced to ZERO. An example of this instruction is given in Figure 8.



PF000650

Figure 7. Rotate and Compare Function

ROTATE AND COMPARE FIELD DEFINITIONS

| | 15 | 14 | 13 | 12 | 9 | 8 | 5 | 4 | 0 |
|------|-----|------|----|----------------------------------|-------------|---|---|---|---|
| ROTC | B/W | Quad | n | Rot Src- Non Rot Src- Mask | RAM Address | | | | |

EXAMPLE: $n = 4$, Word Mode

| | | | | |
|--------------|------|------|------|------|
| U | 0011 | 0001 | 0101 | 0110 |
| U Rotated | 0001 | 0101 | 0110 | 0011 |
| R | 0001 | 0101 | 1111 | 0000 |
| Mask (S) | 0000 | 0000 | 1111 | 1111 |
| Z (status) = | 1 | | | |

Figure 8. Rotate and Compare Examples

ROTATE AND COMPARE INSTRUCTIONS

| Instruction | B/W | Quad | n | U ¹ R ¹ S ¹ | | | | | RAM Address | | |
|-------------|----------------|------|---------|--|------|-----|-----|-----|-------------|-----|------------|
| ROTC | 0 = B 1 = W | 01 | 0 to 15 | 0010 | CDAL | D | ACC | I | 00000 | R00 | RAM Reg 00 |
| | | | | 0011 | CDRI | D | RAM | I | ... | ... | ... |
| | | | | 0100 | CDRA | D | RAM | ACC | 11111 | R31 | RAM Reg 31 |
| | | | | 0101 | CRAI | RAM | ACC | I | | | |

U = Rotated Source
R = Non-Rotated Source
S = Mask

Y BUS AND STATUS - ROTATE AND COMPARE INSTRUCTIONS

| Instruction | Opcode | B/W | Y - Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-------------|--------|-------|--|-------|-------|-------|------|-----|---|---|---|
| ROTC | | 1 = W | $Y_i \leftarrow (\text{Non Rot Op})_i \cdot (\overline{\text{mask}})_{i-16}$ $(\text{Rot Op})_{(i-n) \bmod 16} \cdot (\text{mask})_i$ | NC | NC | NC | NC | 0 | U | 0 | U |
| | | 0 = B | $Y_i \leftarrow (\text{Non Rot Op})_i \cdot (\overline{\text{mask}})_{i-8}$ $(\text{Rot Op})_{(i-n) \bmod 8} \cdot (\text{mask})_i$ | NC | NC | NC | NC | 0 | U | 0 | U |

U = Update
NC = No Change
0 = Reset
1 = Set
 $i = 0$ to 15 when not specified

Prioritize Instruction

The Prioritize Instructions contain four indicators: byte or word mode, operand source (R), mask source (S) and destination. They are further subdivided into two types. The function performed by the Prioritize instruction is shown in Figure 9. The R operand is ANDed with the complement of the Mask operand. A ZERO in the Mask operand allows the corresponding bit in the R operand to participate in the priority encoding function. A ONE in the Mask operand forces the corresponding bit in the R operand to a ZERO, eliminating it from participation in the priority encoding function.

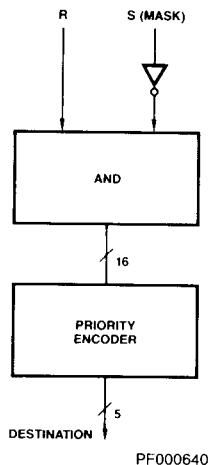


Figure 9. Prioritize Function

The priority encoder accepts a 16-bit input and produces a 5-bit binary-weighted code indicating the bit position of the highest priority active bit. If none of the inputs are active, the output is ZERO. In the Word mode, if input bit 15 is active, the output is 1, etc. Figure 10 lists the output as a function of the highest-priority active-bit position in both the Word and Byte mode. The N and Z bits are affected and the OVR and C bits of the status register are forced to ZERO. The only limitation in this instruction is that the operand and the mask must be different sources.

PRIORITIZE FIELD DEFINITIONS

| 15 14 13 12 | | | 9 8 | | 5 4 | | 0 |
|-------------|------|-------------|-----|-------------|-----|-------------------------|---|
| B/W | Quad | Destination | | Source (R) | | RAM Address/Mask (S) | |
| B/W | Quad | Mask (S) | | Destination | | RAM Address/Source (R) | |
| B/W | Quad | Mask (S) | | Source (R) | | RAM Address/Destination | |
| B/W | Quad | Mask (S) | | Source (R) | | Destination | |

WORD MODE

BYTE MODE*

| Highest Priority Active Bit | Encoder Output | Highest Priority Active Bit | Encoder Output |
|-----------------------------|----------------|-----------------------------|----------------|
| None | 0 | None | 0 |
| 15 | 1 | 7 | 1 |
| 14 | 2 | 6 | 2 |
| . | . | . | . |
| . | . | . | . |
| 1 | 15 | 1 | 7 |
| 0 | 16 | 0 | 8 |

*Bits 8 through 15 do not participate

Figure 10.

PRIORITIZE INSTRUCTION

| Instruction | B/W | Quad | Destination | | | Source (R) | | | RAM Address/Mask (S) | | |
|-------------|----------------|------|-------------|------|-------|-------------|-------|-------|------------------------|-----|------------|
| PRT1 | 0 = B 1 = W | 10 | 1000 | PRIA | ACC | 0111 | RPT1A | ACC | 00000 | R00 | RAM Reg 00 |
| | | | 1010 | PR1Y | Y Bus | .. | .. | .. | .. | .. | .. |
| | | | 1011 | PR1R | RAM | 1001 | PR1D | D | 11111 | R31 | RAM Reg 31 |
| Instruction | B/W | Quad | Mask (S) | | | Destination | | | RAM Address/Source (R) | | |
| PRT2 | 0 = B 1 = W | 10 | 1000 | PRA | Acc | 0000 | PR2A | ACC | 00000 | R00 | RAM Reg 00 |
| | | | 1010 | PRZ | 0 | .. | .. | .. | .. | .. | .. |
| | | | 1011 | PRI | I | 0010 | PR2Y | Y Bus | 11111 | R31 | RAM Reg 31 |
| Instruction | B/W | Quad | Mask (S) | | | Source (R) | | | RAM Address/Dest | | |
| PRT3 | 0 = B 1 = W | 10 | 1000 | PRA | ACC | 0011 | PR3R | RAM | 00000 | R00 | RAM Reg 00 |
| | | | 1010 | PRZ | 0 | 0100 | PR3A | ACC | .. | .. | .. |
| | | | 1011 | PRI | I | 0110 | PR3D | D | 11111 | R31 | RAM Reg 31 |
| Instruction | B/W | Quad | Mask (S) | | | Source (R) | | | Destination | | |
| PRTNR | 0 = B 1 = W | 11 | 1000 | PRA | ACC | 0100 | PRTA | ACC | 00000 | NRY | Y Bus |
| | | | 1010 | PRZ | 0 | .. | .. | .. | 00001 | NRA | ACC |
| | | | 1011 | PRI | I | 0110 | PRTD | D | .. | .. | .. |

| Y BUS AND STATUS - PRIORITIZE INSTRUCTIONS | | | | | | | | | | | |
|--|--------|-----------------------------|---|---|-------|-------|------|-----|---|---|---|
| Instruction | Opcode | B/W | Y – Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
| PRT1 PRT2 | | 1 = W | Y _i - CODE (SCR _n - mask _n), Y _m - 0; i = 0 to 4 and n = 0 to 15 m = 5 to 15 | NC | NC | NC | NC | 0 | U | 0 | U |
| PRT3 PRTNR | | 0 = B | Y _i - CODE (SCR _n - mask _n), Y _m - 0; i = 0 to 3 and n = 0 to 7 m = 4 to 15 | NC | NC | NC | NC | 0 | U | 0 | U |
| SRC = Source U = Update | | NC = No Change 0 = Reset | | 1 = Set i = 0 to 15 when not specified | | | | | | | |

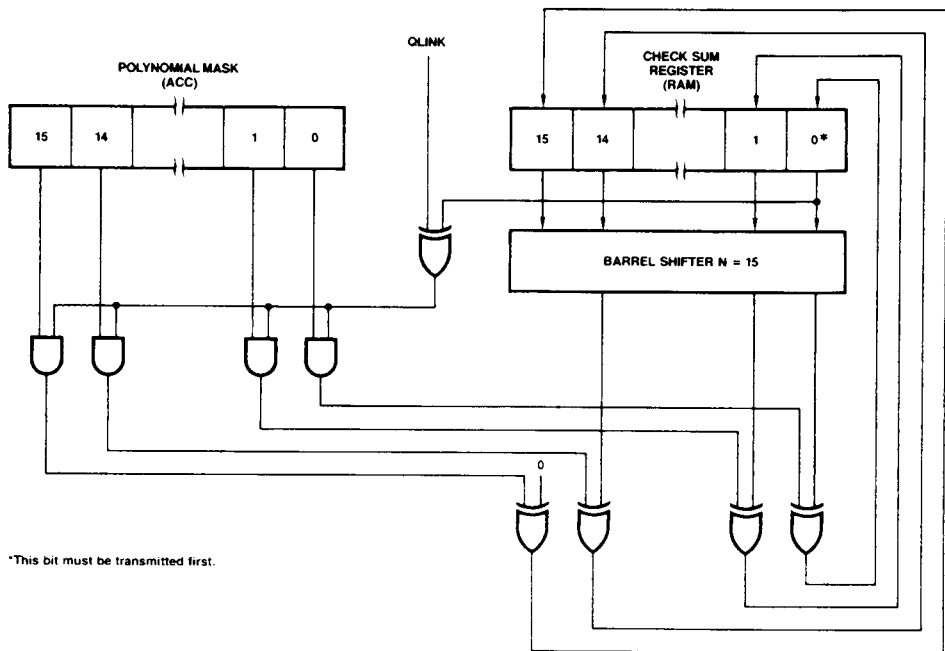
The Cyclic-Redundancy-Check (CRC) Instructions contain one indicator: address of a RAM register to use as the check sum register. The CRC instruction provides a method for generation of the check bits in a CRC calculation. Two CRC instructions are provided – CRC Forward and CRC Reverse. The reason for providing two instructions is that CRC standards do not specify which data bit is to be transmitted first, the LSB or the MSB, but they do specify which check bit must be transmitted first. Figure 11 illustrates the method used to generate these check bits for the CRC Forward function and

Figure 12 illustrates method used for the CRC Reverse function. The ACC serves as a polynomial mask to define the generating polynomial while the RAM register holds the partial result and eventually the calculated check sum. The LINK-bit is used as the serial input. The serial input combines with the MSB of the check-sum register, according to the polynomial defined by the polynomial mask register. When the last input bit has been processed, the check-sum register contains the CRC check bits. The LINK, N and Z bits are affected and the QVR and C bits of the Status register are forced to ZERO.

| | | | | | | | | | |
|------|----|------|------|------|-------------|---|---|---|---|
| | 15 | 14 | 13 | 12 | 9 | 8 | 5 | 4 | 0 |
| CRCF | 1 | Quad | 0110 | 0011 | RAM Address | | | | |
| CRCR | 1 | Quad | 0110 | 1001 | RAM Address | | | | |



Figure 11. CRC Forward Function



PF000320

| Instruction | B/W | Quad | | | RAM Address | | |
|-------------|-----|------|------|------|-------------|-----------|--------------------|
| CRCF | 1 | 10 | 0110 | 0011 | 00000 .. | R00 .. | RAM Reg 00 |
| | | | | | 11111 | R31 | RAM Reg 31 |
| Instruction | B/W | Quad | | | RAM Address | | |
| CRCR | 1 | 10 | 0110 | 1001 | 00000 .. | R00 .. | RAM Reg 00 |
| | | | | | 11111 | R31 | RAM Reg 31 |

| Instruction | Opcode | B/W | Y – Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-------------|--------|-------|---|-------|-------|-------|---------------------|-----|---|---|---|
| CRCF | | 1 = W | $Y_i \leftarrow [(QLINK \oplus RAM_{15}) \cdot ACC_i]$ $\oplus RAM_{i-1} \text{ for } i = 15 \text{ to } 1$ $Y_0 \leftarrow [(QLINK \oplus RAM_{15}) \cdot ACC_0] \oplus 0$ | NC | NC | NC | RAM ₁₅ * | 0 | U | 0 | U |
| CRCR | | 1 = W | $Y_i \leftarrow [(QLINK \oplus RAM_0) \cdot ACC_i]$ $\oplus RAM_{i+1} \text{ for } i = 14 \text{ to } 0$ $Y_{15} \leftarrow [(QLINK \oplus RAM_0) \cdot ACC_{15}] \oplus 0$ | NC | NC | NC | RAM ₀ * | 0 | U | 0 | U |

*QLINK is loaded with the shifted out bit from the checksum register.

Status Instructions

Status Instructions

The Set Status Instruction contains a single indicator. This indicator specifies which bit or group of bits, contained in the status register (Figure 13), are to be set (forced to a ONE).

| | | | | | | | |
|-------|-------|-------|------|-----|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |

MPR-775

Figure 13. Status Byte

The Reset Status Instruction contains a single indicator. This indicator specifies which bit or group of bits, contained in the status register, are to be reset (forced to ZERO).

The Store Status Instruction contains two indicators; byte/word and a second indicator that specifies the destination of the status register. The Store Status Instruction allows the status of the processor to be saved and restored later, which is an especially useful function for interrupt handling.

The status register is always stored in the lower byte of the RAM or the ACC register. Depending upon byte or word mode the upper byte is unchanged or loaded with all ZEROs respectively.

The Load Status instructions are included in the single operand and two operand instruction types.

The Test Status Instructions contain a single indicator which specifies which one of the 12 possible test conditions are to be placed on the Conditional-Test output. Besides the eight bits in the Status register (QZ, QC, QN, QOVR, QLINK, QFlag1, QFlag 2, and QFlag3), four logical functions ($QN \oplus QOVR$, $(QN \oplus QOVR) + QZ$, $QZ + \overline{QC}$ and LOW may also be selected. These functions are useful in testing results of Two's Complement and unsigned number arithmetic operations. The status register may also be tested via the bidirectional T bus. The code to test the status register via T bus is similar to the code used by instruction lines I_1 to I_4 as shown below. Instruction lines $I_0 - 4$ have priority over T bus for testing the

status register on CT output. See the discussion on the status register for a full description.

| T ₄ I ₄ | T ₃ I ₃ | T ₂ I ₂ | T ₁ I ₁ | CT |
|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------|
| 0 | 0 | 0 | 0 | $(N \oplus OVR) + Z$ |
| 0 | 0 | 0 | 1 | $N \oplus OVR$ |
| 0 | 0 | 1 | 0 | Z |
| 0 | 0 | 1 | 1 | OVR |
| 0 | 1 | 0 | 0 | LOW |
| 0 | 1 | 0 | 1 | C |
| 0 | 1 | 1 | 0 | $Z + \overline{C}$ |
| 0 | 1 | 1 | 1 | N |
| 1 | 0 | 0 | 0 | LINK |
| 1 | 0 | 0 | 1 | Flag1 |
| 1 | 0 | 1 | 0 | Flag2 |
| 1 | 0 | 1 | 1 | Flag3 |

STATUS FIELD DEFINITIONS

| | 15 | 14 | 13 | 12 | 9 | 8 | 5 | 4 | 0 |
|--------|-----|------|------|------|------------------|---|---|---|---|
| SETST | 0 | Quad | 1011 | 1010 | Opcode | | | | |
| RSTST | 0 | Quad | 1010 | 1010 | Opcode | | | | |
| SVSTR | B/W | Quad | 0111 | 1010 | RAM Address/Dest | | | | |
| SVSTNR | B/W | Quad | 0111 | 1010 | Destination | | | | |

STATUS INSTRUCTIONS

| Instruction | B/W | Quad | | | Opcode | | |
|-------------|----------------|------|------|------|---|----------------------------------|---|
| SETST | 0 | 11 | 1011 | 1010 | 00011 00101 00110 01001 01010 | SONCZ SL SF1 SF2 SF3 | Set OVR, N, C, Z Set LINK Set Flag1 Set Flag2 Set Flag3 |
| Instruction | B/W | Quad | | | Opcode | | |
| RSTST | 0 | 11 | 1011 | 1010 | 00011 00101 00110 01001 01010 | RONCZ RL RF1 RF2 RF3 | Reset OVR, N, C, Z Reset LINK Set Flag1 Set Flag2 Set Flag3 |
| Instruction | B/W | Quad | | | RAM Address/Dest | | |
| SVSTR | 0 = B 1 = W | 10 | 0111 | 1010 | 00000 ... 11111 | R00 ... R31 | RAM Reg 00 ... RAM Reg 31 |
| Destination | | | | | | | |
| SVSTNR | 0 = B 1 = W | 11 | 0111 | 1010 | 00000 00001 | NRY NRA | Y Bus ACC |

STATUS INSTRUCTIONS

| Instruction | B/W | Quad | | | Opcode (CT) | | |
|-------------|-----|------|------|------|--|---|---|
| Test | 0 | 11 | 1001 | 1010 | 00000 00010 00100 00110 01000 01010 01100 01110 10000 10010 10100 10110 | TNOZ TNO TZ TOVR TLOW TC TZC TN TL TF1 TF2 TF3 | Test (N \oplus OVR) + Z Test N \oplus OVR Test Z Test OVR Test LOW Test C Test Z + \bar{C} Test N Test LINK Test Flag1 Test Flag2 Test Flag3 |

Note:

Y BUS AND STATUS - STATUS INSTRUCTIONS

| Instruction | Opcode | Description | B/W | Y - Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-----------------|--------|---------------------------|----------------|---|-------|-------|-------|------|-----|----|----|----|
| SETST | SONCZ | Set OVR, N, C, Z | 0 = B | $Y_i = 1$ for $i = 0$ to 15 | NC | NC | NC | NC | 1 | 1 | 1 | 1 |
| | SL | Set LINK | | | NC | NC | NC | 1 | NC | NC | NC | NC |
| | SF1 | Set Flag1 | | | NC | NC | 1 | NC | NC | NC | NC | NC |
| | SF2 | Set Flag2 | | | NC | 1 | NC | NC | NC | NC | NC | NC |
| | SF3 | Set Flag3 | | | 1 | NC | NC | NC | NC | NC | NC | NC |
| RSTST | RONCZ | Reset OVR, N, C, Z | 0 = B | $Y_i = 0$ for $i = 0$ to 15 | NC | NC | NC | NC | 0 | 0 | 0 | 0 |
| | RL | Reset LINK | | | NC | NC | NC | 0 | NC | NC | NC | NC |
| | RF1 | Reset Flag1 | | | NC | NC | 0 | NC | NC | NC | NC | NC |
| | RF2 | Reset Flag2 | | | NC | 0 | NC | NC | NC | NC | NC | NC |
| | RF3 | Reset Flag3 | | | 0 | NC | NC | NC | NC | NC | NC | NC |
| SVSTR SVSTNR | | Save Status* | 0 = B 1 = W | Y_i - Status for $i = 0$ to 7; $Y_i = 0$ for $i = 8$ to 15 | NC | NC | NC | NC | NC | NC | NC | NC |
| Test | TNOZ | Test (N \oplus OVR) + Z | 0 = B | ** | NC | NC | NC | NC | NC | NC | NC | NC |
| | TNO | Test N \oplus OVR | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TZ | Test Z | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TOVR | Test OVR | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TLOW | Test LOW | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TC | Test C | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TZC | Test Z + \bar{C} | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TN | Test N | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TL | Test LINK | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TF1 | Test Flag1 | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TF2 | Test Flag2 | | | NC | NC | NC | NC | NC | NC | NC | NC |
| | TF3 | Test Flag3 | | | NC | NC | NC | NC | NC | NC | NC | NC |

U = Update
NC = No Change

0 = Reset

1 = Set

i = 0 to 15 when not specified

*In byte mode only the lower byte from the Y bus is loaded into the RAM or ACC and in word mode all 16-bits from the Y bus are loaded into the RAM or ACC.

**Y-Bus is Undefined.

NO-OP Instruction

The NO-OP Instruction has a fixed 16-bit code. This instruction does not change any internal registers in the Am291176. It preserves the status register, RAM register and the ACC register.

NO OPERATION FIELD DEFINITION

| | | | | | | | | | | | | |
|------|----|----|----|----|------|---|---|------|---|---|-------|---|
| | 15 | 14 | 13 | 12 | | 9 | 8 | | 5 | 4 | | 0 |
| NOOP | 0 | | 11 | | 1000 | | | 1010 | | | 00000 | |

NO-OP INSTRUCTION

| Instruction | B/W | Quad | | | |
|-------------|-----|------|------|------|-------|
| NOOP | 0 | 11 | 1000 | 1010 | 00000 |

Y BUS AND STATUS - NO-OP INSTRUCTION

| Instruction | Opcode | B/W | Y - Bus | Flag3 | Flag2 | Flag1 | LINK | OVR | N | C | Z |
|-------------|--------|-------|---------|-------|-------|-------|------|-----|----|----|----|
| NOOP | | 0 = B | * | NC | NC | NC | NC | NC | NC | NC | NC |

SRC = Source

U = Update

NC = No Change

0 = Reset

1 = Set

i = 0 to 15 when not specified

*Y-Bus is undefined.

SUMMARY OF MNEMONICS

INSTRUCTION TYPE

| | |
|--------|---------------------------------|
| SOR | Single Operand RAM |
| SONR | Single Operand Non-RAM |
| TOR1 | Two Operand RAM (Quad 0) |
| TOR2 | Two Operand RAM (Quad 2) |
| TONR | Two Operand Non-RAM |
| SHFTR | Single Bit Shift RAM |
| SHFTNR | Single Bit Shift Non-RAM |
| ROTR1 | Rotate n Bits RAM (Quad 0) |
| ROTR2 | Rotate n Bits RAM (Quad 1) |
| ROTNR | Rotate n Bits Non-RAM |
| BOR1 | Bit Oriented RAM (Quad 3) |
| BOR2 | Bit Oriented RAM (Quad 2) |
| BONR | Bit Oriented Non-RAM |
| ROTM | Rotate and Merge |
| ROTC | Rotate and Compare |
| PRT1 | Prioritize RAM; Type 1 |
| PRT2 | Prioritize RAM; Type 2 |
| PRT3 | Prioritize RAM; Type 3 |
| PRTNR | Prioritize Non-RAM |
| CRCF | Cyclic Redundancy Check Forward |
| CRCR | Cyclic Redundancy Check Reverse |
| NOOP | No Operation |
| SETST | Set Status |
| RSTST | Reset Status |
| SVSTR | Save Status RAM |
| SVSTNR | Save Status Non-RAM |
| TEST | Test Status |

SOURCE AND DESTINATION

Single Operand

| | |
|-------|------------------------------|
| SORA | Single Operand RAM to ACC |
| SORY | Single Operand RAM to Y Bus |
| SORS | Single Operand RAM to Status |
| SOAR | Single Operand ACC to RAM |
| SODR | Single Operand D to RAM |
| SOIR | Single Operand I to RAM |
| SOZR | Single Operand 0 to RAM |
| SOZER | Single Operand D(0E) to RAM |
| SOSER | Single Operand D(SE) to RAM |
| SORR | Single Operand RAM to RAM |
| SOA | Single Operand ACC |
| SOD | Single Operand D |
| SOI | Single Operand I |
| SOZ | Single Operand 0 |
| SOZE | Single Operand D(0E) |
| SOSE | Single Operand D(SE) |
| NRY | Non-RAM Y Bus |
| NRA | Non-RAM ACC |
| NRS | Non-RAM Status |
| NRAS | Non-RAM ACC, Status |

Two Operand

| | |
|-------|-------------------------------|
| TORAA | Two Operand RAM, ACC to ACC |
| TORIA | Two Operand RAM, I to ACC |
| TODRA | Two Operand D, RAM to ACC |
| TORAY | Two Operand RAM, ACC to Y Bus |
| TORIY | Two Operand RAM, I to Y Bus |
| TODRY | Two Operand D, RAM to Y Bus |
| TORAR | Two Operand RAM, ACC to RAM |
| TORIR | Two Operand RAM, I to RAM |
| TODRR | Two Operand D, RAM to RAM |
| TODAR | Two Operand D, ACC to RAM |
| TOAIR | Two Operand ACC, I to RAM |
| TODIR | Two Operand D, I to RAM |
| TODA | Two Operand D, ACC |
| TOAI | Two Operand ACC, I |
| TODI | Two Operand D, I |

Single Bit Shift

| | |
|------|-------------------------|
| SHRR | Shift RAM, Store in RAM |
| SHDR | Shift D, Store in RAM |
| SHA | Shift ACC |
| SHD | Shift D |

Rotate n Bits

| | |
|------|----------------------------|
| RTRA | Rotate RAM, Store in ACC |
| RTRY | Rotate RAM, Place on Y Bus |
| RTRR | Rotate RAM, Store in RAM |
| RTAR | Rotate ACC, Store in RAM |
| RTDR | Rotate D, Store in RAM |
| RTDY | Rotate D, Place on Y Bus |
| RTDA | Rotate D, Store in ACC |
| RTAY | Rotate ACC, Place on Y Bus |
| RTAA | Rotate ACC, Store in ACC |

Rotate and Merge

| | |
|------|---|
| MDAI | Merge Disjoint Bits of D and ACC Using I as Mask and Store in ACC |
| MDAR | Merge Disjoint Bits of D and ACC Using RAM as Mask and Store in ACC |
| MDRI | Merge Disjoint Bits of D and RAM Using I as Mask and Store in RAM |
| MDRA | Merge Disjoint Bits of D and RAM Using ACC as Mask and Store in RAM |
| MARI | Merge Disjoint Bits of ACC and RAM Using I as Mask and Store in RAM |
| MRAI | Merge Disjoint Bits of RAM and ACC Using I as Mask and Store in ACC |

Rotate and Compare

| | |
|------|--|
| CDAI | Compare Unmasked Bits of D and ACC Using I as Mask |
|------|--|

| | |
|------|--|
| CDRI | Compare Unmasked Bits of D and RAM Using I as Mask |
| CDRA | Compare Unmasked Bits of D and RAM Using ACC as Mask |
| CRAI | Compare Unmasked Bits of RAM and ACC Using I as Mask |

Prioritize

| | |
|-------|--|
| PR1A | ACC as Destination for Prioritize Type 1 |
| PR1Y | Y Bus as Destination for Prioritize Type 1 |
| PR1R | RAM as Destination for Prioritize Type 1 |
| PRT1A | ACC as Source for Prioritize Type 1 |
| PR1D | D as Source for Prioritize Type 1 |
| PR2A | ACC as Destination for Prioritize Type 2 |
| PR2Y | Y Bus as Destination for Prioritize Type 2 |
| PR3R | RAM as Source for Prioritize Type 3 |
| PR3A | ACC as Source for Prioritize Type 3 |
| PR3D | D as Source for Prioritize Type 3 |
| PRTA | ACC as source for Prioritize Type Non-RAM |
| PRTD | D as Source for Prioritize Type Non-RAM |
| PRA | ACC as Mask for Prioritize Type 2, 3, and Non-RAM |
| PRZ | Mask Equal to Zero for Prioritize Type 2, 3, and Non-RAM |
| PRI | I as Mask for Prioritize Type 2, 3, and Non-RAM |

OPCODE

Addition

| | |
|-------|--------------------------------|
| ADD | Add without Carry |
| ADDC | Add with Carry |
| A2NA | Add 2^n to ACC |
| A2NR | Add 2^n to RAM |
| A2NDY | Add 2^n to D, Place on Y Bus |

Subtraction

| | |
|-------|---------------------------------------|
| SUBR | Subtract R from S without Carry |
| SUBRC | Subtract R from S with Carry |
| SUBS | Subtract S from R without Carry |
| SUBSC | Subtract S from R with Carry |
| S2NR | Subtract 2^n from RAM |
| S2NA | Subtract 2^n from ACC |
| S2NDY | Subtract 2^n from D, Place on Y Bus |

Logical Operations

| | |
|-------|---------------|
| AND | Boolean AND |
| NAND | Boolean NAND |
| EXOR | Boolean EXOR |
| NOR | Boolean NOR |
| OR | Boolean OR |
| EXNOR | Boolean EXNOR |

SHIFTS

| | |
|-------|---------------------------------------|
| SHUPZ | Shift Up Towards MSB with 0 Insert |
| SHUP1 | Shift Up Towards MSB with 1 Insert |
| SHUPL | Shift Up Towards MSB with LINK Insert |

| | |
|--------|---|
| SHDNZ | Shift Down Towards LSB with 0 Insert |
| SHDN1 | Shift Down Towards LSB with 1 Insert |
| SHDNL | Shift Down Towards LSB with LINK Insert |
| SHDNC | Shift Down Towards LSB with Carry Insert |
| SHDNOV | Shift Down Towards LSB with Sign EXOR Overflow Insert |

Loads

| | |
|--------|----------------------|
| LD2NR | Load 2^n into RAM |
| LDC2NR | Load 2^n into RAM |
| LD2NA | Load 2^n into ACC |
| LDC2NA | Load 2^n into ACC |
| LD2NY | Place 2^n on Y Bus |
| LDC2NY | Place 2^n on Y Bus |

Bit Oriented

| | |
|-------|--|
| SETNR | Set RAM, Bit n |
| SETNA | Set ACC, Bit n |
| SETND | Set D, Bit n |
| SONCZ | Set OVR, N, C, Z, in Status Register |
| SL | Set LINK Bit in Status Register |
| SF1 | Set Flag1 Bit in Status Register |
| SF2 | Set Flag2 Bit in Status Register |
| SF3 | Set Flag3 Bit in Status Register |
| RSTNR | Reset RAM, Bit n |
| RSTNA | Reset ACC, Bit n |
| RSTND | Reset D, Bit n |
| RONCZ | Reset OVR, N, C, Z, in Status Register |
| RL | Reset LINK Bit in Status Register |
| RF1 | Reset Flag1 Bit in Status Register |
| RF2 | Reset Flag2 Bit in Status Register |
| RF3 | Reset Flag3 Bit in Status Register |
| TSTNR | Test RAM, Bit n |
| TSTNA | Test ACC, Bit n |
| TSTND | Test D, Bit n |

Arithmetic Operations

| | |
|------|-----------------------------|
| MOVE | Move and Update Status |
| COMP | Complement (1's Complement) |
| INC | Increment |
| NEG | Two's Complement |

Conditional Test

| | |
|------|---------------------------|
| TNOZ | Test $(N \oplus OVR) + Z$ |
| TNO | Test $N \oplus OVR$ |
| TZ | Test Zero Bit |
| TOVR | Test Overflow Bit |
| TLOW | Test for LOW |
| TC | Test Carry Bit |
| TZC | Test $Z + \bar{C}$ |
| TN | Test Negative Bit |
| TL | Test LINK Bit |
| TF1 | Test Flag1 Bit |
| TF2 | Test Flag2 Bit |
| TF3 | Test Flag3 Bit |

ABSOLUTE MAXIMUM RATINGS

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

Stresses above those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

OPERATING RANGES

Commercial (C) Devices

Temperature (T_A) 0 to +55°C
Supply Voltage +4.75 V to +5.25 V
Air Velocity 300 linear feet per minute

Operating ranges define those limits over which the functionality of the device is guaranteed.

DC CHARACTERISTICS over operating range unless otherwise specified

| Parameters | Description | Test Conditions (Note 1) | | Min. | Max. | Units |
|------------|--|--|--|---|--|---------------|
| V_{OH} | Output HIGH Voltage | $V_{CC} = \text{Min.}$ $V_{IN} = V_{IH} \text{ or } V_{IL}$ | Y_{0-15} T_{1-4} CT | $I_{OH} = -1.6 \text{ mA}$ | 2.4 | Volts |
| V_{OL} | Output LOW Voltage | $V_{CC} = \text{Min.}$ $V_{IN} = V_{IH} \text{ or } V_{IL}$ | Y_{0-15} T_{1-4} CT | $I_{OL} = 16 \text{ mA}$ | 0.5 | Volts |
| V_{IH} | Guaranteed Input Logical HIGH Voltage (Note 5) | | All Inputs | | 2.0 | Volts |
| V_{IL} | Guaranteed Input Logical LOW Voltage (Note 5) | | All Inputs | | 0.8 | Volts |
| V_I | Input Clamp Voltage | $V_{CC} = \text{Min.}$ | All Inputs | $I_{IN} = -18 \text{ mA}$ | -1.5 | Volts |
| I_{IL} | Input LOW Current | $V_{CC} = \text{Max.}$ $V_{IN} = 0.5 \text{ Volts}$ (Note 3) | I_{EN} SRE DLE I_{0-4} I_{5-15} OET OEY CP T_{1-4} D_{0-15} | | -0.50 -0.50 -1.00 -1.00 -0.50 -0.50 -0.50 -1.50 -0.55 -0.50 | mA |
| I_{IH} | Input HIGH Current | $V_{CC} = \text{Max.}$ $V_{IN} = 2.4 \text{ Volts}$ (Note 3) | I_{EN} SRE DLE I_{0-4} I_{5-15} OET OEY CP T_{1-4} D_{0-15} | | 50 50 100 100 50 50 50 150 100 50 | μA |
| I_I | Input HIGH Current | $V_{CC} = \text{Max.}$ $V_{IN} = 5.5 \text{ Volts}$ | All Inputs | | 1.0 | mA |
| I_{OZH} | Off State (HIGH Impedance) Output Current | $V_{CC} = \text{Max.}$ $V_O = 2.4 \text{ Volts}$ (Note 3) | T_{1-4} (Note 3) Y_{0-15} | | 100 50 | μA |
| I_{OZL} | Off State (HIGH Impedance) Output Current | $V_{CC} = \text{Max.}$ $V_O = 0.5 \text{ Volts}$ (Note 4) | T_{1-4} (Note 3) Y_{0-15} | | -550 -50 | μA |
| I_{OS} | Output Short Circuit Current | $V_{CC} = \text{Max.} + 0.5 \text{ Volts}$ $V_O = 0.5 \text{ Volts}$ (Note 2) | | -30 | -85 | mA |
| I_{CC} | Power Supply Current (Note 4) | $V_{CC} = \text{Max.}$ | COM'L | $T_{AL} = 0 \text{ to } 55^\circ\text{C}$ (Note 6) $T_A = 55^\circ\text{C}$ | 735 535 | mA |

- Notes: 1. For conditions shown as Min. or Max., use the appropriate value specified under Operating Ranges for the applicable device type.
2. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
3. Y_{0-15} , T_{1-4} are three-state outputs internally connected to TTL inputs. Input characteristics are measured under conditions such that the outputs are in the OFF state.
4. Worst case I_{CC} is at minimum temperature.
5. These input levels provide zero noise immunity and should be tested only in a static, noise-free environment.
6. Cold start.

SWITCHING CHARACTERISTICS over operating range unless otherwise specified(T_A = 0 to +55°C, V_{CC} = 4.75 to 5.25 V, C_L = 50 pF) (Note 7)**A. Combinational Delays (nsec)**

| | | Outputs | | |
|-------|---------------------------|-------------------|------------------|----|
| | | Y ₀₋₁₅ | T ₁₋₄ | CT |
| | I ₀₋₄ (ADDR) | 79 | 84 | — |
| | I ₀₋₁₅ (DATA) | 79 | 84 | — |
| | I ₀₋₁₅ (INSTR) | 79 | 84 | 48 |
| Input | DLE | 58** | 60 | — |
| | T ₁₋₄ | — | — | 39 |
| | CP | 56 | 62 | 36 |
| | D ₀₋₁₅ | 62 | 64 | — |
| | IEN | — | — | 43 |




B. Enable/Disable Times (nsec)(C_L = 5pF for disable only)

| From Input | To Output | Enable | | Disable | |
|-----------------|-------------------|------------------|------------------|------------------|------------------|
| | | t _{pZH} | t _{pZL} | t _{pHZ} | t _{pLZ} |
| OE _Y | Y ₀₋₁₅ | 20 | 20 | 20 | 20 |
| OE _T | T ₁₋₄ | 25 | 25 | 25 | 25 |

C. Clock and Pulse Requirements (nsec)

| Input | Min Low Time | Min High Time |
|-------|--------------|---------------|
| CP | 20 | 30 |
| DLE | — | 15 |
| IEN | 22 | — |

D. Setup and Hold Times (nsec)

| Input | With Respect to | High-to-Low Transition | | Low-to-High Transition | | Comment |
|---|---|-------------------------|-------------------------|------------------------|---------------------------|------------------------------|
| | | Set-up | Hold | Set-up | Hold | |
| I ₀₋₄ (RAM ADDR) | CP | (t _{s1}) 24 | (t _{h1}) 0 | — | — | Single ADDR (Source) |
| I ₀₋₄ (RAM ADDR) | CP and IEN both LOW | (t _{s2}) 10 | Do Not Change | | (t _{h7}) 2 | Two ADDR (Destination) |
| I ₀₋₁₅ (DATA) | CP | — | — | (t _{s8}) 65 | (t _{h8}) 2 | |
| I ₀₋₁₅ (INSTR) | CP | (t _{s3}) 38† | (t _{h3}) + 17 | (t _{s9}) 65 | (t _{h9}) 2 | |
| I ₀₋₁₅ (INSTR) | IEN  | (t _{s16}) 6 | (t _{h16}) 18 | — | — | Two ADDR Immediate |
| IEN  | CP  | — | — | — | (t _{h15}) 8 | Two ADDR Immediate |
| IEN HIGH | CP | (t _{s4}) 10 | — | — | (t _{h10}) 1 | Disable |
| IEN LOW | CP | — (t _{s5}) 20 | — (t _{h5})† 2 | (t _{s11}) 22 | — (t _{h11})†† 1 | Enable Immediate first cycle |
| SRE | CP | — | — | (t _{s12}) 17 | (t _{h12}) 0 | |
| D | CP | — | — | (t _{s13}) 44 | (t _{h13}) 1 | |
| D | DLE | (t _{s6}) 10 | (t _{h6}) 6 | — | — | |
| DLE | CP | — | — | (t _{s14}) 42 | (t _{h14}) 0 | |

†Timing for immediate instruction for first cycle.

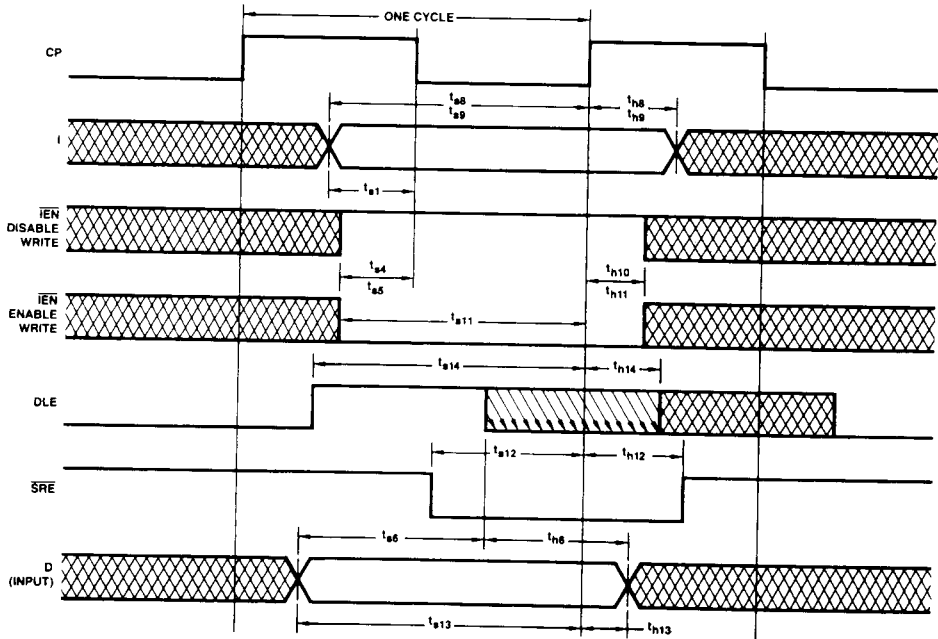
†† Status register and accumulator destination only.

Notes on Testing

Incoming test procedures on this device should be careful planned, taking into account the high complexity and power levels of the part. The following notes may be useful:

1. Insure the part is adequately decoupled at the test head. Large changes in V_{CC} current as the device switches may cause erroneous function failure due to V_{CC} changes.
2. Do not leave inputs floating during any tests, as they may start to oscillate at high frequency.
3. Do not attempt to perform threshold tests at high speed. Following an input transition, ground current may change by as much as 400 mA in 5–8 ns. Inductance in the ground cable may allow may allow the ground pin at the device to rise by 100s of millivolts momentarily.
4. Use extreme care in defining input levels for AC tests. Many inputs may be changed at once, so there will be significant noise at the device pins and they may not actually reach V_{IL} or V_{IH} until the noise has settled. AMD recommends using V_{IL} ≤ 0 V and V_{IH} ≥ 3.0 V for AC tests.
5. To simplify failure analysis, programs should be designed to perform DC, Function, and AC tests as three distinct groups of tests.
6. To assist in testing, AMD offers complete documentation on our test procedures and, in most cases, can provide Fairchild Sentry programs, under license.
7. At air velocity of 300 linear feet per minute.

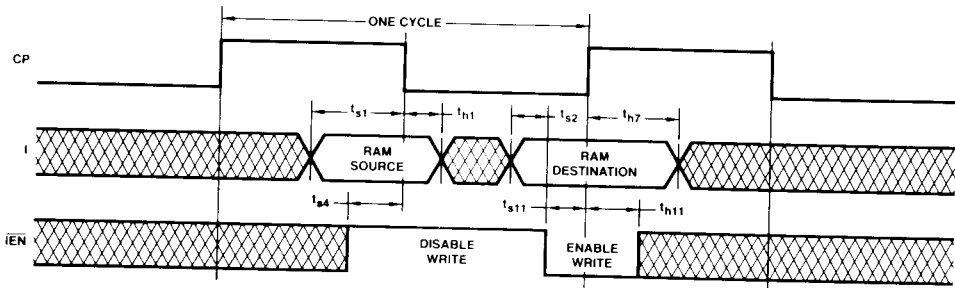
SWITCHING WAVEFORMS



WF002561

Single Address Access Timing

If t_{h6} is satisfied, t_{h13} need not be satisfied.

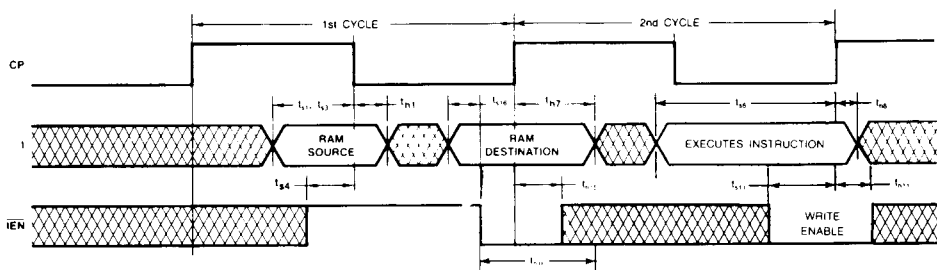


WF002540

Double Address Access Timing

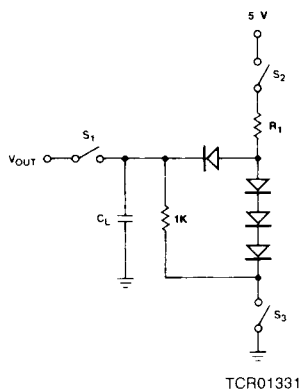
The diagram illustrates the timing relationships for the 6800 microprocessor across two clock cycles. The signals shown are CP (Clock Pulse), IEN (Instruction Enable), and I (Instruction/Data bus). The I bus is divided into two sections: the first section is labeled "CAPTURES INSTRUCTION" and the second is labeled "EXECUTES INSTRUCTION". The I bus is also labeled with $I_0-15 = \text{INSTRUCTIONS}$ and $I_0-15 = \text{DATA}$. The timing parameters shown are t_{s5} , t_{h5} , t_{s3} , t_{h3} , t_{s4} , t_{h10} , t_{h11} , t_{s8} , and t_{h8} . The diagram shows that the IEN signal is active during the first cycle and the second cycle. The I bus is active during the first cycle and the second cycle. The I bus is active during the first cycle and the second cycle.

One-Address Immediate Instruction Cycle Timing



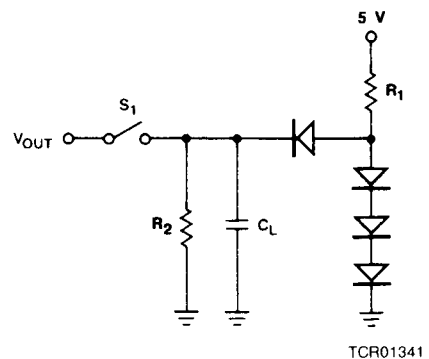
WF002542

SWITCHING TEST CIRCUITS



A. Three-State Outputs

$$R_1 = 300 \, \Omega$$



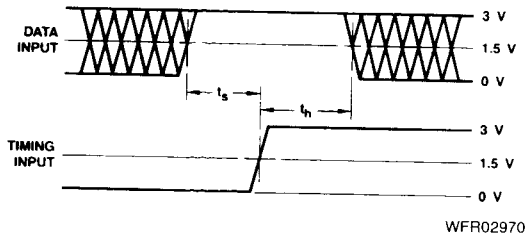
B. Normal Outputs

$$R_1 = 300 \, \Omega$$

$R_2 = 3.0 \text{ k}\Omega$

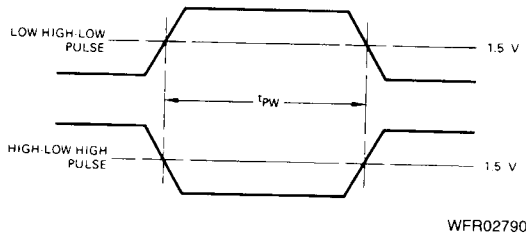
- Notes:
1. $C_L = 50$ pF includes scope probe, wiring and stray capacitances without device in test fixture.
 2. S_1 , S_2 , S_3 are closed during function tests and all AC tests except output enable tests.
 3. S_1 and S_3 are closed while S_2 is open for t_{pZH} test.
 S_1 and S_2 are closed while S_3 is open for t_{pZL} test.
 4. $C_L = 5.0$ pF for output disable tests.

SWITCHING TEST WAVEFORMS



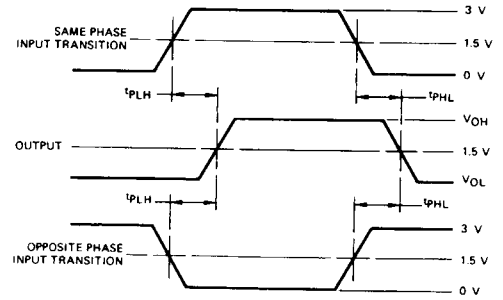
Set-up, Hold, and Release Times

- Notes: 1. Diagram shown for HIGH data only. Output transition may be opposite sense.
2. Cross hatched area is don't care condition.

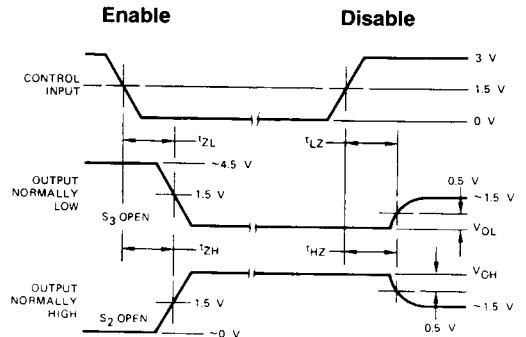


Pulse Width

- Note: 1. Pulse generator for all pulses
Rate ≤ 1.0 MHz; $Z_O = 50 \Omega$;
 $t_r \leq 2.5$ ns; $t_f \leq 2.5$ ns.



Propagation Delay

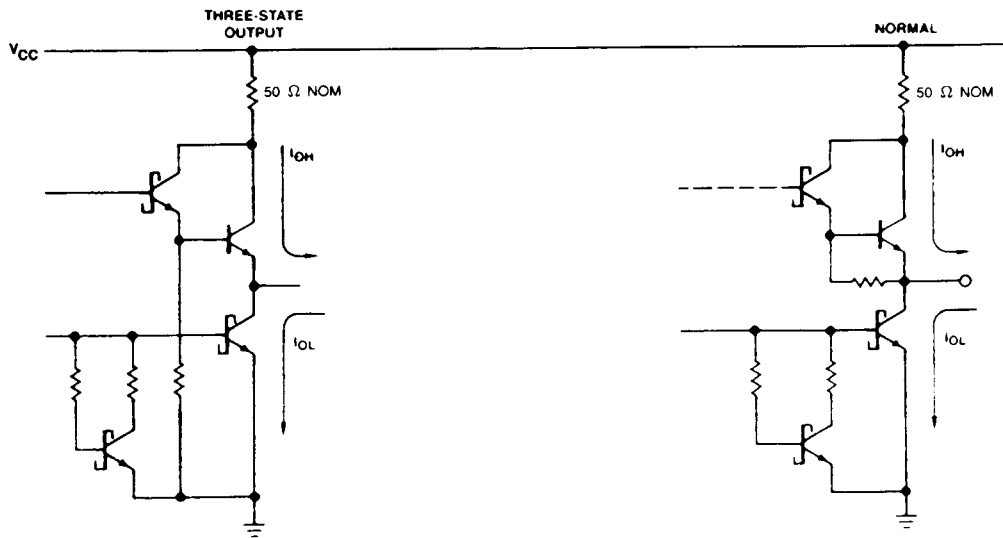


Enable and Disable Times

- Notes: 1. Diagram shown for Input Control Enable-LOW and Input Control Disable-HIGH
2. S_1 , S_2 and S_3 of Load Circuit are closed except where shown.

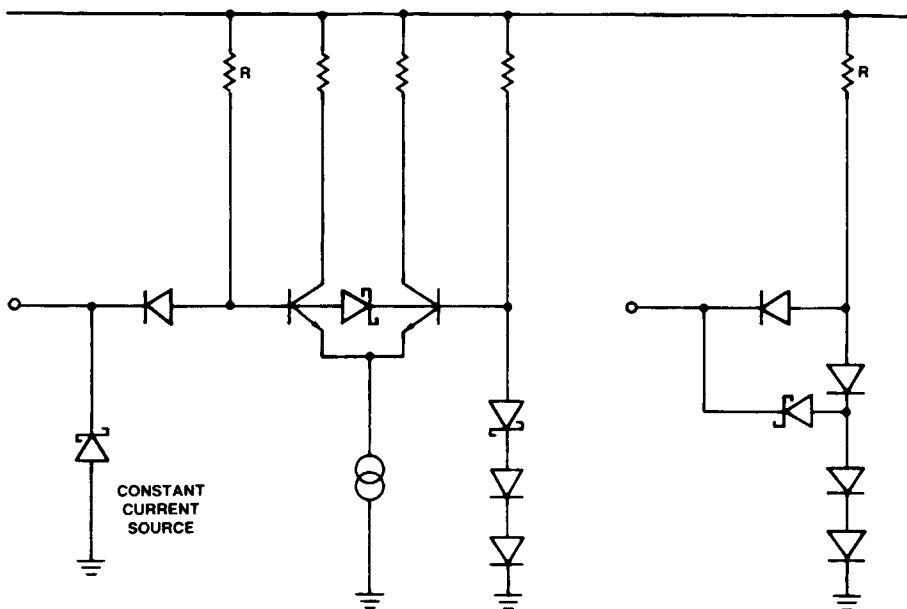
INPUT/OUTPUT CURRENT INTERFACE

TTL



ICR00523

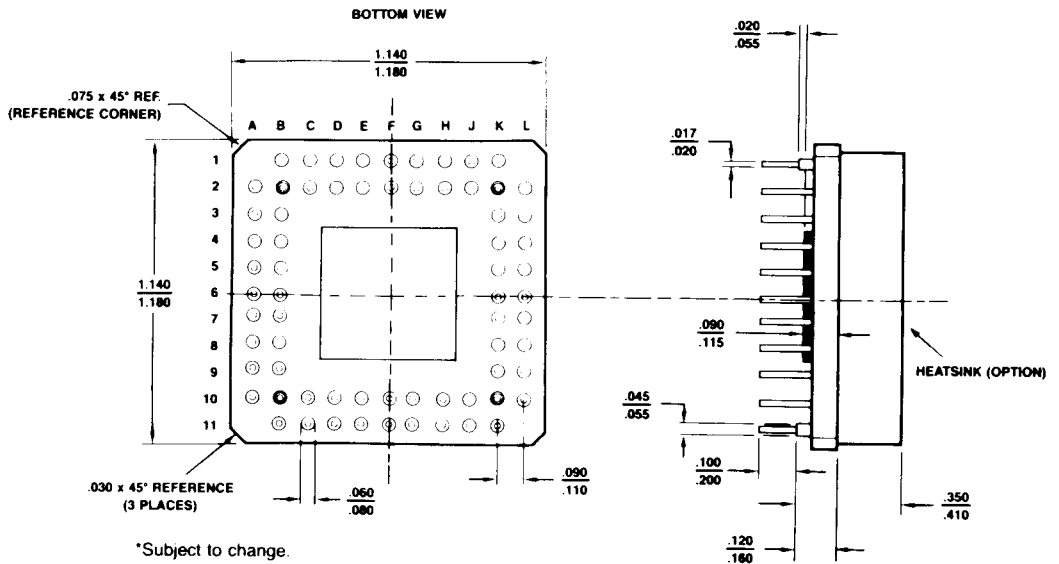
$C_O \cong 5.0$ pF, all outputs



ICR00531

PHYSICAL DIMENSIONS

CG068



07547A

The International Standard of
Quality guarantees a 0.05% AQL on all
electrical parameters, AC and DC,
over the entire operating range.

INT·STD·500

ADVANCED MICRO DEVICES DOMESTIC SALES OFFICES

| | | | |
|-------------------------|----------------|---------------------|----------------|
| ALABAMA | (205) 882-9122 | MASSACHUSETTS | (617) 273-3970 |
| ARIZONA, | | MINNESOTA | (612) 938-0001 |
| Tempe | (602) 242-4400 | NEW JERSEY | (201) 299-0002 |
| Tucson | (602) 792-1200 | NEW YORK, | |
| CALIFORNIA, | | Liverpool | (315) 457-5400 |
| El Segundo | (213) 640-3210 | Poughkeepsie | (914) 471-8180 |
| Newport Beach | (714) 752-6262 | Woodbury | (516) 364-8020 |
| San Diego | (619) 560-7030 | NORTH CAROLINA, | |
| Sunnyvale | (408) 720-8811 | Charlotte | (704) 525-1875 |
| Woodland Hills | (818) 992-4155 | Raleigh | (919) 847-8471 |
| COLORADO | (303) 691-5100 | OREGON | (503) 245-0080 |
| CONNECTICUT, | | OHIO, | |
| Southbury | (203) 264-7800 | Columbus | (614) 891-6455 |
| FLORIDA, | | PENNSYLVANIA, | |
| Altamonte Springs | (305) 339-5022 | Allentown | (215) 398-8006 |
| Clearwater | (813) 530-9971 | Willow Grove | (215) 657-3101 |
| Ft. Lauderdale | (305) 484-8600 | TEXAS, | |
| Melbourne | (305) 254-2915 | Austin | (512) 346-7830 |
| GEORGIA | (404) 449-7920 | Dallas | (214) 934-9099 |
| ILLINOIS | (312) 773-4422 | Houston | (713) 785-9001 |
| INDIANA | (317) 244-7207 | WASHINGTON | (206) 455-3600 |
| KANSAS | (913) 451-3115 | WISCONSIN | (414) 782-7748 |
| MARYLAND | (301) 796-9310 | | |

INTERNATIONAL SALES OFFICES

| | | | |
|---------------------|----------------------|-------------------------|---------------------|
| BELGIUM, | | HONG KONG, | |
| Bruxelles | TEL: (02) 771 99 93 | Kowloon | TEL: 3-695377 |
| | FAX: 762-3716 | | FAX: 1234276 |
| | TLX: 61028 | | TLX: 50426 |
| CANADA, Ontario, | | ITALY, Milano | TEL: (02) 3390541 |
| Kanata | TEL: (613) 592-0060 | | FAX: 3498000 |
| Willowdale | TEL: (416) 224-5193 | | TLX: 315286 |
| | FAX: (416) 224-0056 | JAPAN, Tokyo | TEL: (03) 345-8241 |
| FRANCE, | | | FAX: 3425196 |
| Paris | TEL: (01) 687.36.66 | | TLX: J24064 AMDTKOJ |
| | FAX: 6862185 | LATIN AMERICA, | |
| | TLX: 20253 | Ft. Lauderdale, | TEL: (305) 484-8600 |
| GERMANY, | | | FAX: (305) 485-9736 |
| Hannover area | TEL: (05143) 50 55 | SWEDEN, Stockholm | TEL: (08) 733 03 50 |
| | FAX: 5553 | | FAX: 7332285 |
| | TLX: 925287 | | TLX: 11602 |
| München | TEL: (089) 41 14-0 | UNITED KINGDOM, | |
| | FAX: 406490 | Manchester area | TEL: (0925) 828008 |
| | TLX: 523883 | | FAX: 827693 |
| Stuttgart | TEL: (0711) 62 33 77 | | TLX: 628524 |
| | FAX: 625187 | London area | TEL: (04862) 22121 |
| | TLX: 721882 | | FAX: 22179 |
| | | | TLX: 859103 |

NORTH AMERICAN REPRESENTATIVES

| | | | |
|-------------------------------|---------------------|--------------------------------|----------------|
| CALIFORNIA | | NEW JERSEY | |
| I ² INC | OEM (408) 988-3400 | TAI CORPORATION | (609) 933-2600 |
| | DIST (408) 498-6868 | NEW MEXICO | |
| CONNECTICUT | | THORSON DESERT STATES | (505) 293-8555 |
| SCIENTIFIC COMPONENTS | (203) 272-2963 | NEW YORK | |
| IDAHO | | NYCOM, INC | (315) 437-8343 |
| INTERMOUNTAIN TECH MKGT | (208) 322-5022 | OHIO | |
| INDIANA | | Dayton | |
| SAI MARKETING CORP | (317) 241-9276 | DOLFUSS ROOT & CO | (513) 433-6776 |
| IOWA | | Strongsville | |
| LORENZ SALES | (319) 377-4666 | DOLFUSS ROOT & CO | (216) 238-0300 |
| MICHIGAN | | PENNSYLVANIA | |
| SAI MARKETING CORP | (313) 227-1786 | DOLFUSS ROOT & CO | (412) 221-4420 |
| NEBRASKA | | UTAH | |
| LORENZ SALES | (402) 475-4660 | R ² MARKETING | (801) 595-0631 |

Advanced Micro Devices reserves the right to make changes in its product without notice in order to improve design or performance characteristics. The performance characteristics listed in this document are guaranteed by specific tests, correlated testing, guard banding, design and other practices common to the industry. For specific testing details, contact your local AMD sales representative. The company assumes no responsibility for the use of any circuits described herein.



ADVANCED MICRO DEVICES 901 Thompson Pl., P.O. Box 3453, Sunnyvale, CA 94088, USA
TEL: (408) 732-2400 • TWX: 910-339-9280 • TELEX: 34-6306 • TOLL FREE: (800) 538-8450

© 1985 Advanced Micro Devices, Inc.
Printed in U.S.A. AMT-B-15M-10/85-0