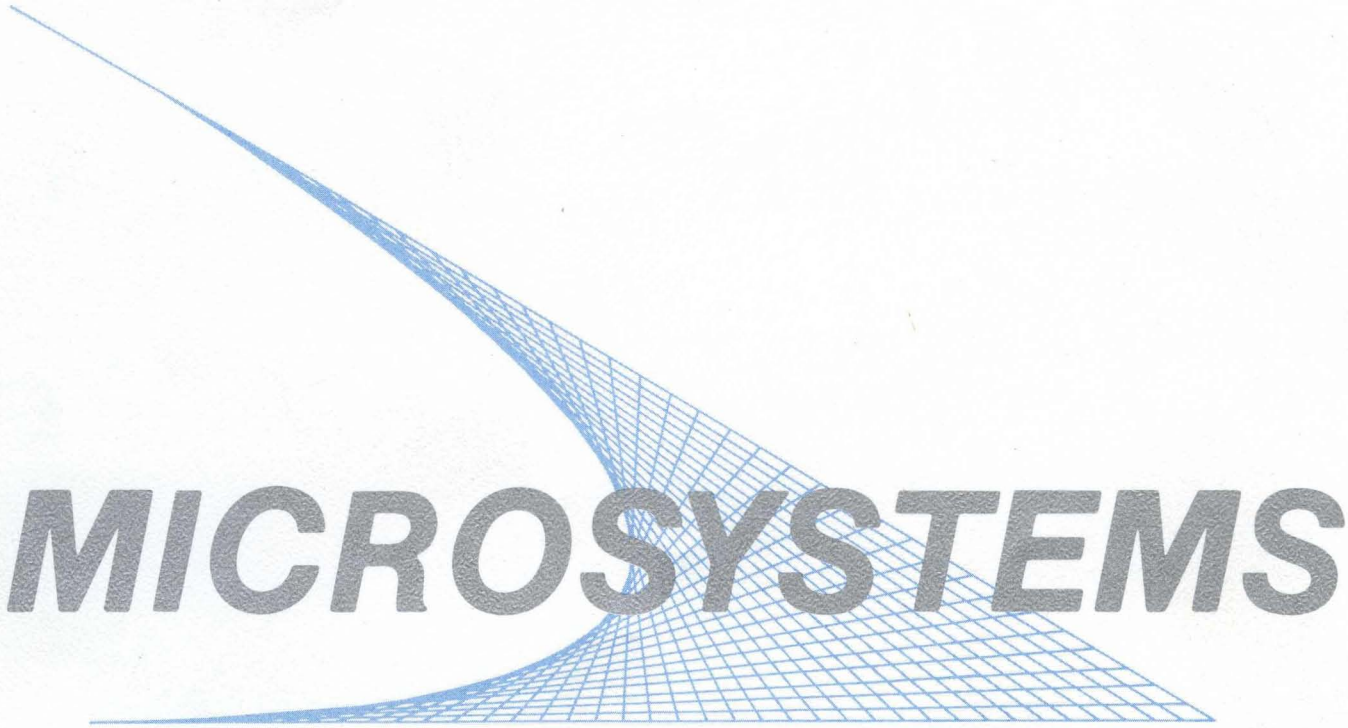
MOTOROLA

MVME120/D2

MVME120, MVME121, MVME122, MVME123
VMEbus Microprocessor Module
User's Manual



QUALITY • PEOPLE • PERFORMANCE

MOTOROLA

MVME120/D2

JUNE 1985

MVME120, MVME121, MVME122, MVME123

VMEbus MICROPROCESSOR MODULE

USER'S MANUAL

WARNING

THIS EQUIPMENT GENERATES, USES, AND CAN RADIATE RADIO FREQUENCY ENERGY AND IF NOT INSTALLED AND USED IN ACCORDANCE WITH THE INSTRUCTIONS MANUAL, MAY CAUSE INTERFERENCE TO RADIO COMMUNICATIONS. IT HAS BEEN TESTED AND FOUND TO COMPLY WITH THE LIMITS FOR A CLASS A COMPUTING DEVICE PURSUANT TO SUBPART J OF PART 15 OF FCC RULES, WHICH ARE DESIGNED TO PROVIDE REASONABLE PROTECTION AGAINST SUCH INTERFERENCE WHEN OPERATED IN A COMMERCIAL ENVIRONMENT. OPERATION OF THIS EQUIPMENT IN A RESIDENTIAL AREA IS LIKELY TO CAUSE INTERFERENCE IN WHICH CASE THE USER, AT HIS OWN EXPENSE, WILL BE REQUIRED TO TAKE WHATEVER MEASURES NECESSARY TO CORRECT THE INTERFERENCE.

Second Edition

First Edition September 1984

MICROSYSTEMS

SAFETY SUMMARY
SAFETY DEPENDS ON YOU

The following general safety precautions must be observed during all phases of operation, service, and repair of this equipment. Failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the equipment. Motorola Inc. assumes no liability for the customer's failure to comply with these requirements. The safety precautions listed below represent warnings of certain dangers of which we are aware. You, as the user of the product, should follow these warnings and all other safety precautions necessary for the safe operation of the equipment in your operating environment.

GROUND THE INSTRUMENT.

To minimize shock hazard, the equipment chassis and enclosure must be connected to an electrical ground. The equipment is supplied with a three-conductor ac power cable. The power cable must either be plugged into an approved three-contact electrical outlet or used with a three-contact to two-contact adapter, with the grounding wire (green) firmly connected to an electrical ground (safety ground) at the power outlet. The power jack and mating plug of the power cable meet International Electrotechnical Commission (IEC) safety standards.

DO NOT OPERATE IN AN EXPLOSIVE ATMOSPHERE.

Do not operate the equipment in the presence of flammable gases or fumes. Operation of any electrical equipment in such an environment constitutes a definite safety hazard.

KEEP AWAY FROM LIVE CIRCUITS.

Operating personnel must not remove equipment covers. Only Factory Authorized Service Personnel or other qualified maintenance personnel may remove equipment covers for internal subassembly or component replacement or any internal adjustment. Do not replace components with power cable connected. Under certain conditions, dangerous voltages may exist even with the power cable removed. To avoid injuries, always disconnect power and discharge circuits before touching them.

DO NOT SERVICE OR ADJUST ALONE.

Do not attempt internal service or adjustment unless another person, capable of rendering first aid and resuscitation, is present.

USE CAUTION WHEN EXPOSING OR HANDLING THE CRT.

Breakage of the Cathode-Ray Tube (CRT) causes a high-velocity scattering of glass fragments (implosion). To prevent CRT implosion, avoid rough handling or jarring of the equipment. Handling of the CRT should be done only by qualified maintenance personnel using approved safety mask and gloves.

DO NOT SUBSTITUTE PARTS OR MODIFY EQUIPMENT.

Because of the danger of introducing additional hazards, do not install substitute parts or perform any unauthorized modification of the equipment. Contact Motorola Microsystems Warranty and Repair for service and repair to ensure that safety features are maintained.

DANGEROUS PROCEDURE WARNINGS.

Warnings, such as the example below, precede potentially dangerous procedures throughout this manual. Instructions contained in the warnings must be followed. You should also employ all other safety precautions which you deem necessary for the operation of the equipment in your operating environment.

WARNING

Dangerous voltages, capable of causing death, are present in this equipment. Use extreme caution when handling, testing, and adjusting.

PREFACE

Unless otherwise specified, all address references are in hexadecimal throughout this manual.

An asterisk (*) following the signal name for signals which are level significant denotes that the signal is true or valid when the signal is low.

An asterisk (*) following the signal name for signals which are edge significant denotes that the actions initiated by that signal occur on a high to low transition.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont'd)

TABLE OF CONTENTS (cont'd)

Page

LIST OF ILLUSTRATIONS

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK.

CHAPTER 1

GENERAL INFORMATION

1.1 INTRODUCTION

This manual provides general information, preparation for use and installation instructions, operating instructions, functional description, and support information for the MVME120, MVME121, MVME122, and MVME123 VMEbus Microprocessor Modules (hereinafter referred to as MPU module). A typical MPU module is shown in Figure 1-1.

1.2 FEATURES

The features of the MPU module include:

- MC68010 MPU
- MC68451 MMU (optional)
- 4Kb logical instruction cache (optional)
- Two, 28-pin ROM/EPROM sockets
- Onboard dual-port RAM with byte parity
- Interrupt handler
- Status and control registers
- Programmable timer
- RS-232C serial debug port
- Bus requester
- A24, D16 VMEbus interface
- Local RESET switch
- Local software ABORT switch
- Mode switch
- FAIL indicator
- HALT indicator
- RUN indicator

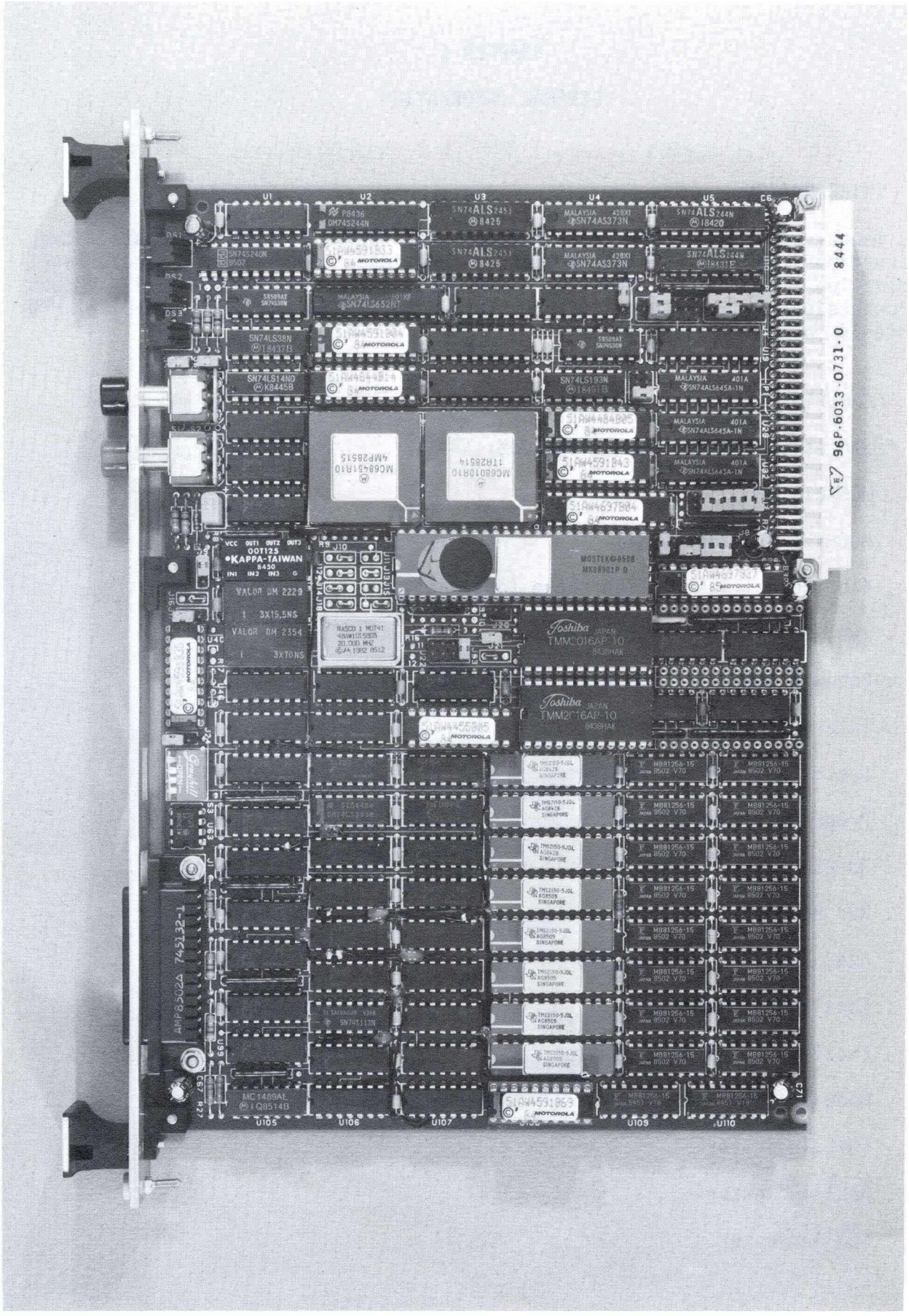


FIGURE 1-1. Typical MPU Module

1.3 SPECIFICATIONS

The MPU module specifications are identified in Table 1-1.

TABLE 1-1. MPU Module Specifications

CHARACTERISTICS	SPECIFICATIONS
Microprocessor	MC68010
Clock signal	12.5 MHz (or 10.0 MHz)
Memory size	
RAM	128Kb/512Kb
EPROM/ROM	Two sockets for user supplied 4K x 8, 8K x 8, 16K x 8, 32K x 8, or 64K x 8 devices
Interrupt handler	All onboard plus seven VMEbus interrupts
User input/output signals	RS-232C serial debug port (terminal only)
Temperature	
Operating	0 degrees C to 50 degrees C (forced air is required)
Storage	-40 degrees to 85 degrees C
Relative humidity	0% to 90% (noncondensing)
Physical characteristics	
Width x height	7.40 in. (188 mm) x 10.20 in. (261 mm)
Thickness	0.83 in. (21 mm)
Power requirements	+5 Vdc typical current maximum current 120 4.2 A 4.65 A 121 4.0 A 4.45 A 122 3.2 A 3.55 A 123 4.3 A 4.65 A -12 Vdc @ 12 mA (typical), 14 mA (maximum) +12 Vdc @ 17 mA (typical), 20 mA (maximum)

1

1.4 GENERAL DESCRIPTION

The MPU module is a high performance, VMEbus, microprocessor module that offers a solution for high speed data processing, management, and control applications. The module contains an MC68010 MPU, MC68451 MMU, logical instruction cache, a serial debug terminal port, programmable tick timer, private ROM, and a large dynamic dual port RAM that may be loaded externally via the VMEbus. The MPU module, when used with a system controller module (such as the MVME050 system controller or the MVME110 CPU module), satisfies system requirements ranging from simple, high speed single processor applications to complex multiprocessor system architectures.

1.5 MVME120 FAMILY CONFIGURATIONS

Four configurations are available for the MVME120 Family of Microprocessor Modules. The configurations are listed below:

MVME120	10 MHz MC68010, 128K dynamic RAM, cache, MMU
MVME121	10 MHz MC68010, 512K dynamic RAM, cache, MMU
MVME122	12.5 MHz MC68010, 128K dynamic RAM, no cache, no MMU
MVME123	12.5 MHz MC68010, 512K dynamic RAM, cache, no MMU

NOTE

The MC68451 MMU does not function at 12.5 MHz.

1.6 RELATED DOCUMENTATION

The following publications may provide additional helpful information. If not shipped with this product, they may be obtained from Motorola Literature Distribution Center, 616 West 24th Street, Tempe, Arizona 85282; telephone (602) 994-6561.

DOCUMENT TITLE	MOTOROLA PUBLICATION NUMBER
Data Book	MC68901
MVMEbus Specification Manual	MVMEBS

CHAPTER 2

HARDWARE PREPARATION AND INSTALLATION INSTRUCTIONS

2.1 INTRODUCTION

This chapter provides unpacking instructions, hardware preparation, and installation instructions for the MPU module.

2.2 UNPACKING INSTRUCTIONS

NOTE

If the shipping carton is damaged upon receipt, request carrier's agent be present during unpacking and inspection of equipment.

Unpack equipment from shipping carton. Refer to packing list and verify that all items are present. Save packing material for storing and reshipping of equipment.

2.3 HARDWARE PREPARATION

For reliable operation, forced air cooling should be used. The cooling should be enough to maintain the temperature at the back of the MC68010 case below 80 degrees centigrade.

To select the desired configuration and ensure proper operation of the MPU module, certain modifications may be made before installation. These modifications are made through jumper or wire-wrap arrangements on the headers. Figure 2-1 illustrates the location of the headers and connectors on the MPU module. The MPU module has been factory tested and is shipped with factory-installed jumper configurations that are also shown in Figure 2-1. The MPU module is operational with the factory-installed jumpers. It is necessary to make changes in the jumper arrangements for the following conditions:

a. ACFAIL*/SYSFAIL* select (J2)

b. VMEbus request level select (J3,J4)

c. ABORT switch disable select (J5)

d. RESET switch disable select (J6)

2

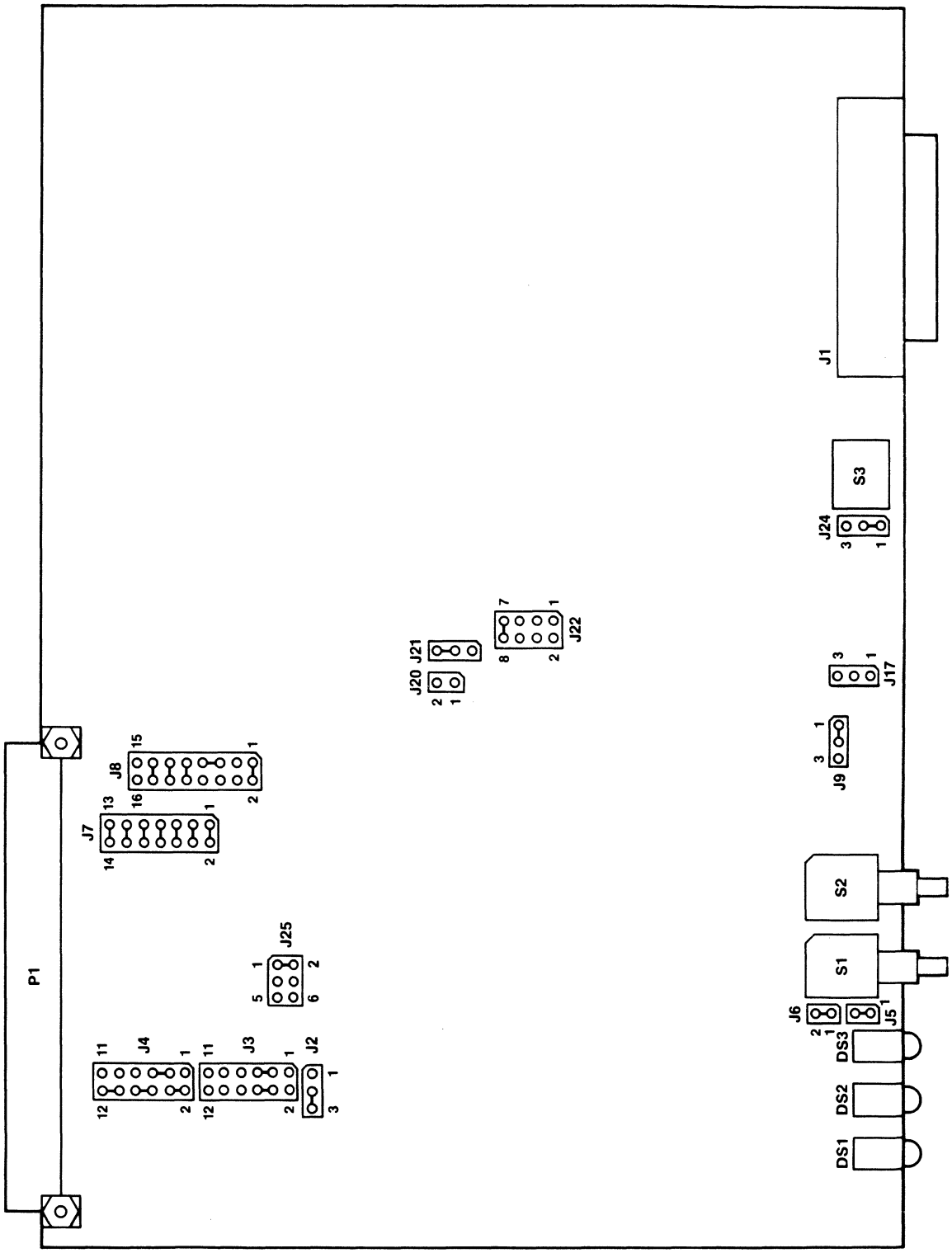


FIGURE 2-1. MPU Module Option Locations

e. VMEbus interrupt connection select (J7)

f. ROM/EPROM device configuration select (J8)

g. Cache configuration select (J9, J17)

h. SW1 source select (J20)

i. SW2 source select (J21)

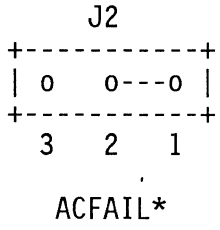
j. ROM access time select (J22)

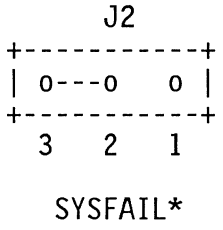
k. Reset vector fetch mode select (J24)

l. Local time-out select (J25)

2.3.1 ACFAIL*/SYSFAIL* Select Header (J2)

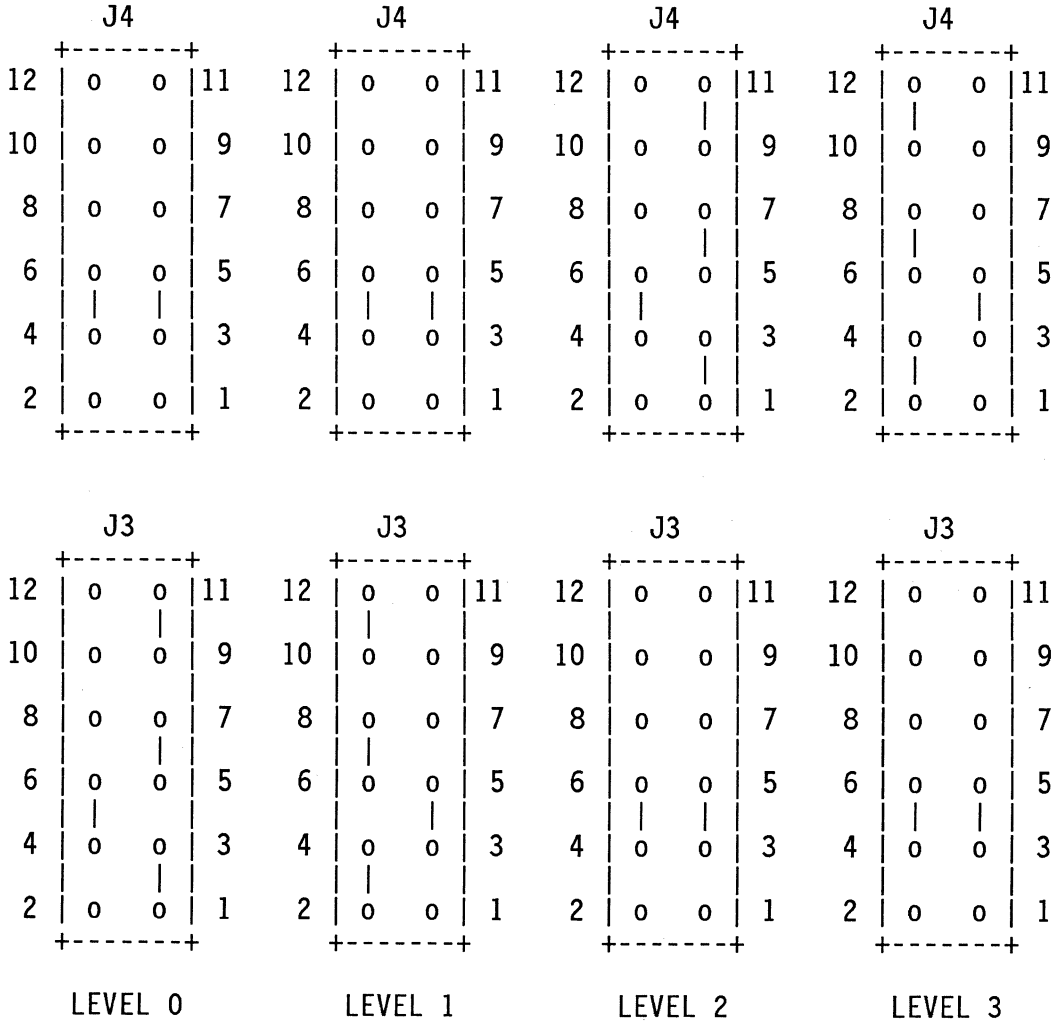
Bit 7 of the Module Status Register (MSR) can be configured to monitor either the Alternating Current Failure (ACFAIL*) signal on the VMEbus or to monitor the System Failure (SYSFAIL*) signal on the VMEbus. ACFAIL* is monitored when the jumper is positioned between pins 1 and 2 on header J2. SYSFAIL* is monitored when the jumper is positioned between pins 2 and 3.





2.3.2 VMEbus Request Level Select Headers (J3, J4)

The MPU module can be configured to request VMEbus mastership on any one of four levels. The desired level is selected by jumper configuration on headers J3 and J4 as shown below:



2.3.3 ABORT Switch Disable Select Header (J5)

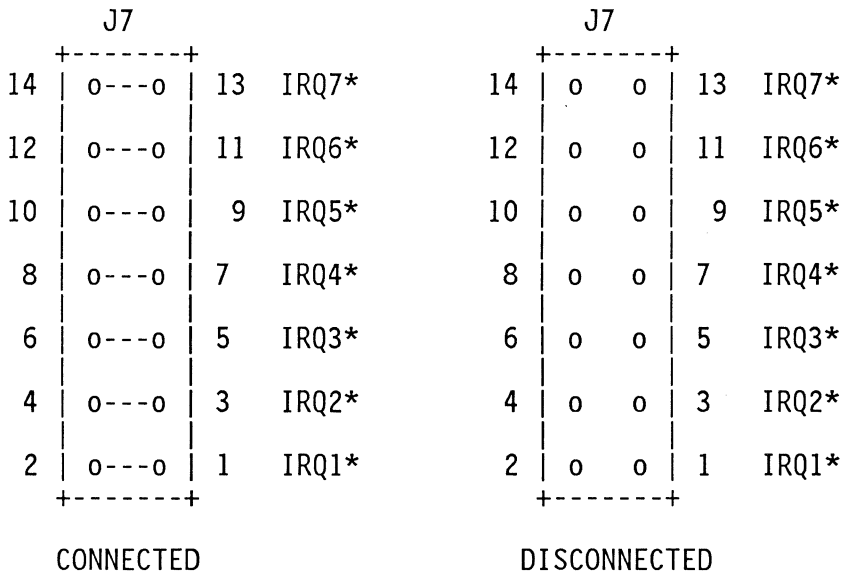
The front panel software ABORT switch on the MPU module can be disabled. The switch is disabled when the jumper is removed from header J5. The MPU module is shipped with the switch enabled as shown below.

2.3.4 RESET Switch Disable Select Header (J6)

The front panel RESET switch on the MPU module can be disabled. The switch is disabled when the jumper is removed from header J6. As shown below, the MPU module is shipped with the switch enabled.

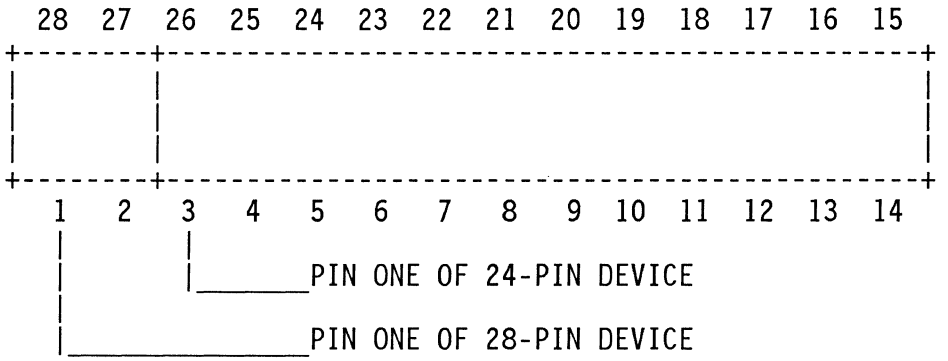
2.3.5 VMEbus Interrupt Connection Select Header (J7)

Each VMEbus Interrupt Request (IRQ*) signal can be connected as an interrupt request to the MPU. Jumpers on header J7 determine which, if any, of the VMEbus IRQ* signal lines are connected. J7 connects the IRQ* lines so that they drive the corresponding interrupt request levels to the MPU (i.e., IRQ1* requests level 1 to the MPU, IRQ2* requests level 2 to the MPU, etc.).



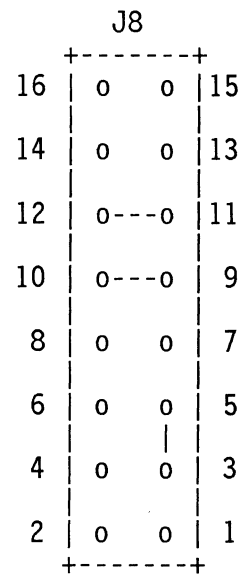
2.3.6 ROM/EPROM Device Configuration Select Header (J8)

There are several types of ROM/EPROM devices that can be used in sockets XU44 and XU52. Header J8 must be configured to match the devices used. The figures below show configurations of J8 for the most commonly used devices. When inserting devices into XU44 and XU52, the device containing even data must be inserted into XU44 and the device containing odd data must be inserted into XU52. Even though XU44 and XU52 are 28-pin sockets, 24-pin devices may be used as shown below:

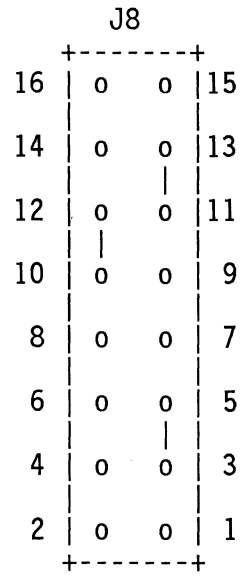


2

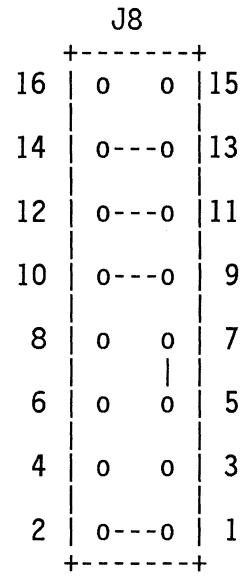
Header configuration for 4K x 8 EPROM memory devices (AM2732, INT2732) is shown below:



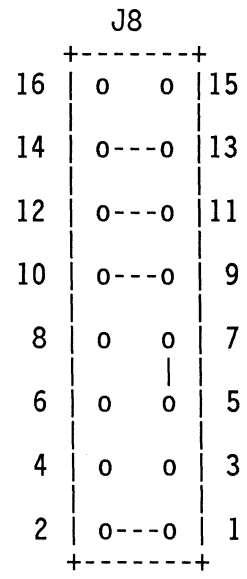
Header configuration for 8K x 8 EPROM memory devices (MCM68764, MCM68766) is shown below:



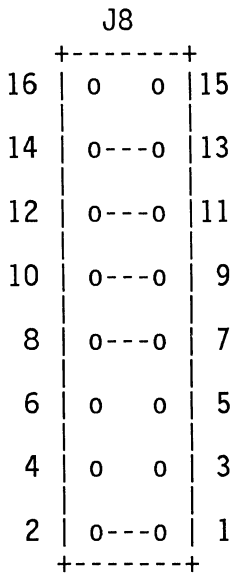
Header configuration for 8K x 8 EPROM memory devices (AM2764, INT2764) is shown below:



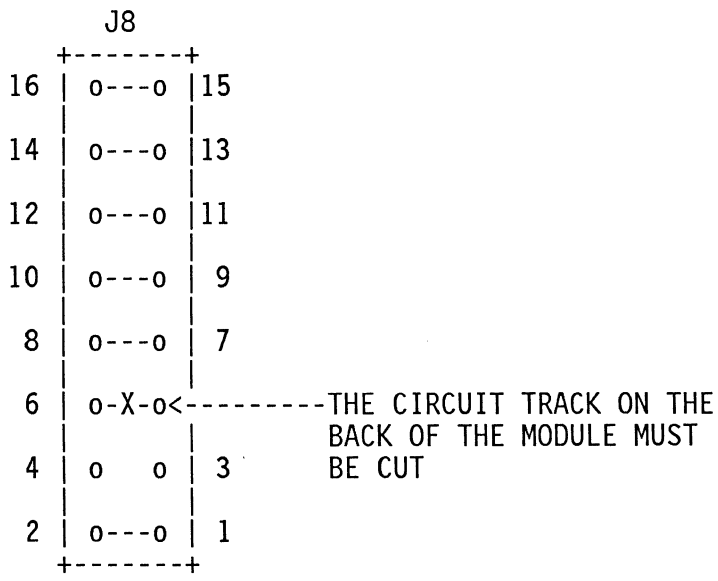
Header configuration for 16K x 8 EPROM memory devices (AM27128, INT27128) is shown below:



Header configuration for 32K x 8 EPROM memory devices (AM27256) is shown below:



Header configuration for 64K x 8 EPROM memory devices (AM27512) is shown below:



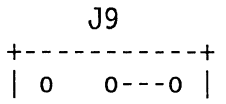
2.3.7 Cache Configuration Select Headers (J9, J17)

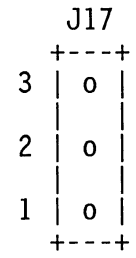
The cache can be configured for one of four modes. The cache has a total of 4Kb. The modes are user only, supervisor only, mixed user and supervisor, and 2Kb supervisor and 2Kb user. Jumper position on headers J9 and J17 determine what mode is selected, as shown below:

ALL 4KB OF CACHE ARE USER ONLY

ALL 4KB OF CACHE ARE SUPERVISOR ONLY

ALL 4KB OF CACHE ARE MIXED USER AND SUPERVISOR

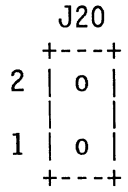


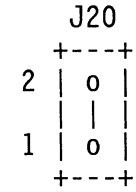


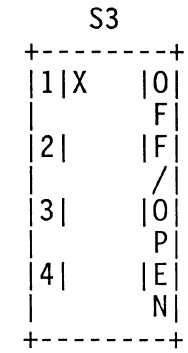
2KB OF CACHE ARE USER ONLY
2KB OF CACHE ARE SUPERVISOR ONLY

2.3.8 MSR Bit 1 Source Select Header (J20)

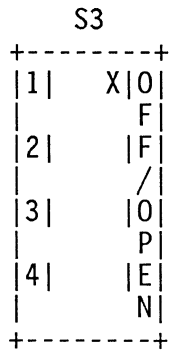
Bit 1 of the MSR is always connected to section 1 of switch S3. It is high when section 1 is open and low when closed. The signal line Cache Error (CACHERR*) can also be connected to bit 1 of the MSR and section 1. When CACHERR* is connected through header J20 and section 1 is closed, then cache is disabled. When section 1 is open, the signal line CACHERR* can be monitored but this is not a useful configuration. Header configurations and switch settings are shown below:



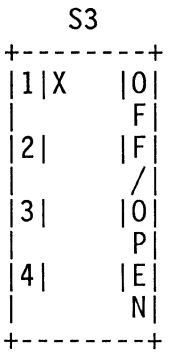




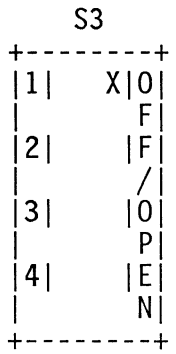
MSR BIT 1=0



MSR BIT 1=1



CACHE DISABLED (MSR BIT 1 IS ALWAYS LOW)

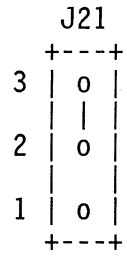


THIS CONFIGURATION NOT RECOMMENDED

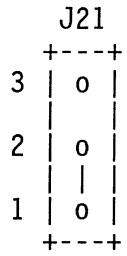
2.3.9 MSR Bit 0 Source Select Header (J21)

2

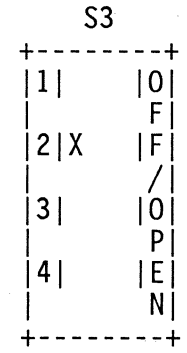
Bit 0 of the MSR can be configured to monitor section 2 of switch S3 or to monitor the signal line Cache Hit (CACHEHIT*). If it is connected to section 2, it is high when section 2 is open and low when closed. If bit 0 is connected to the signal CACHEHIT*, it matches the level of CACHEHIT*. Header and switch configurations are shown below:



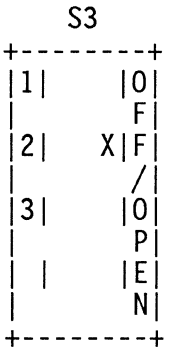
SECTION 2 OF S3 CONNECTED TO BIT 0 OF THE MSR



[CACHEHIT*] CONNECTED TO BIT 0 OF THE MSR



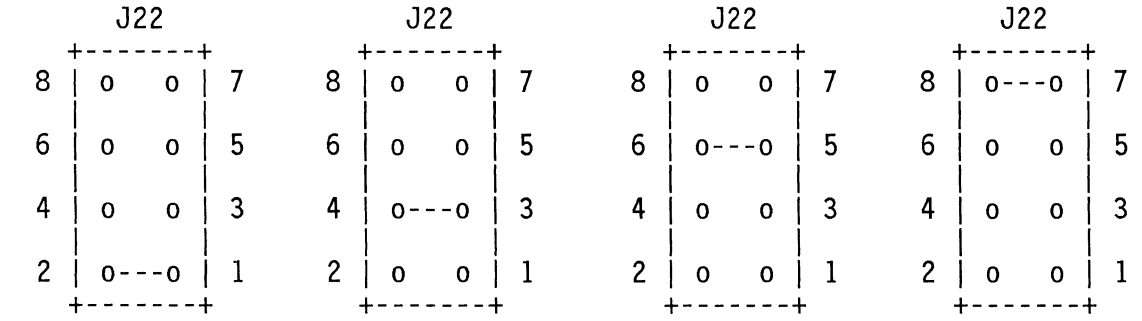
MSR BIT 0=0



MSR BIT 0=1

2.3.10 ROM Access Time Select Header (J22)

Header J22 must be configured to match the access time of the EPROM/ROM devices that are installed in sockets XU44 and XU52 as shown in the figures and tables below. Access time is the longer of the time from valid address to valid data or from chip enable to valid data at the pins of the ROM devices. Note that other header configurations affect the configuration of J22 for any given ROM access time.

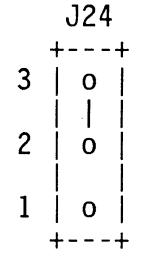


ROM ACCESS TIME LESS THAN OR EQUAL TO	ROM ACCESS TIME LESS THAN OR EQUAL TO	ROM ACCESS TIME LESS THAN OR EQUAL TO	ROM ACCESS TIME LESS THAN OR EQUAL TO
MVME 120, 121, 123			
N/A	250 ns	350 ns	450 ns
MVME 122			
200 ns	400 ns	450 ns	500 ns

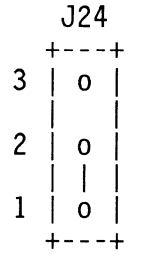
2.3.11 Reset Vector Fetch Mode Select Header (J24)

The MPU module can fetch the reset vector from onboard ROM or from the VMEbus. When the MPU module fetches the reset vectors from the VMEbus, address modifier code 1E or 16 can be used. The mode used is controlled by the position of the jumper on the header and by section 3 and 4 of switch S3 as shown below:

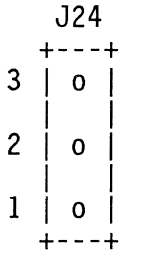
2



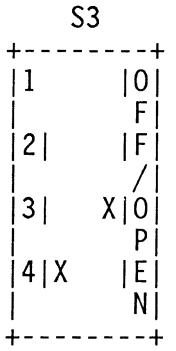
FETCH RESET
VECTORS FROM
VMEbus
(SWITCH HAS NO EFFECT)



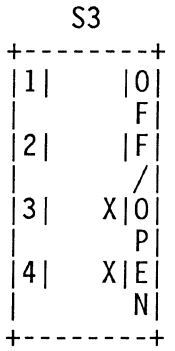
SECTION 4 OF S3
CONTROLS MODE
(SEE BELOW)



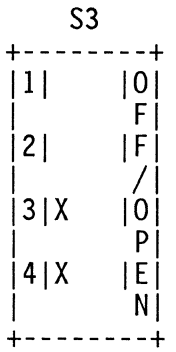
NEVER FETCH
RESET VECTORS
FROM VMEbus
(SWITCH HAS NO EFFECT)



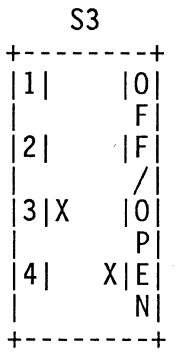
FETCH RESET
VECTORS FROM
VMEbus
ADDRESS MODIFIER=1E



FETCH RESET
VECTORS FROM
ONBOARD



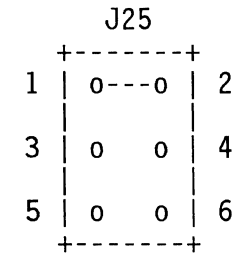
FETCH RESET
VECTORS FROM
VMEbus
ADDRESS MODIFIER=16



FETCH RESET
VECTORS FROM
ONBOARD

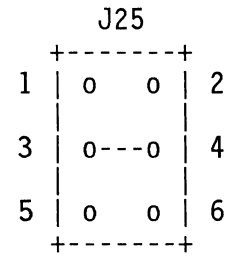
2.3.12 Local Time-out Select Header (J25)

The local bus time-out of the MPU module can be configured for one of three times using jumper positioning on header J25. The user must ensure that the system controller module has the global bus time-out set correctly.



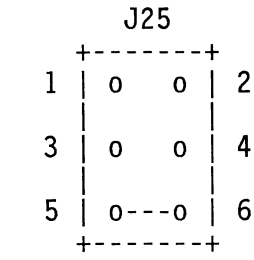
LOCAL TIME-OUT = 103-137 MICROSECONDS

THE VMEbus GLOBAL BUS TIME-OUT MUST BE LESS THAN OR EQUAL TO 110 MICROSECONDS



LOCAL TIME-OUT =43-77 MICROSECONDS

THE VMEbus GLOBAL BUS TIME-OUT MUST BE LESS THAN OR EQUAL TO 170 MICROSECONDS



LOCAL TIME-OUT =13-47 MICROSECONDS

THE VMEbus GLOBAL BUS TIME-OUT MUST BE LESS THAN OR EQUAL TO 200 MICROSECONDS

2.3.13 RAM Dual Port Base Address Select PAL (U28)

The onboard RAM of the MPU Module is accessible by the onboard MPU and by other VMEbus masters. The lower address at which the onboard RAM appears for the VMEbus master is called the RAM dual port base address. The RAM dual port base address is controlled by the program in the PAL in socket U28. The PAL can be programmed to place the RAM dual port base address on any 128Kb boundary for 128K of onboard RAM or on any 512Kb boundary for 512K of onboard RAM. The PAL in socket U28 is a PAL16L2 with a propagation delay time that is less than or equal to 35 ns.

2.3.13.1 Factory Configured RAM Dual Port Base Address. The factory configuration of U28 places the RAM dual port base address at 000000. The figures below show the source code for the factory configuration of U28 for 128Kb of onboard RAM and for 512Kb of onboard RAM.

2

```
PAL16L2                                                    2-17-84
U28-SHEET 8      MVME120,MVME122                           CKSM= 0744
PALDP20                                                    REV A

A23 A22 A21 A20 A19 A18 A17 /LWORD /IACK GND
AM1 AM3 AM2 AM4 15 /DPMATCH AM0 AM5 VMEAV VCC

DPMATCH = /A23*/A22*/A21*/A20*/A19*/A18*/A17     ;$000000-$01FFFF
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*AM2*AM1*AM0                ;STANDARD PRIV PROG

        + /A23*/A22*/A21*/A20*/A19*/A18*/A17      ;$000000-$01FFFF
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*AM2*/AM1*AM0               ;STANDARD PRIV DATA

        + /A23*/A22*/A21*/A20*/A19*/A18*/A17      ;$000000-$01FFFF
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*/AM2*AM1*/AM0              ;STANDARD NON-PRIV PROG

        + /A23*/A22*/A21*/A20*/A19*/A18*/A17      ;$000000-$01FFFF
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*/AM2*/AM1*AM0              ;STANDARD NON-PRIV DATA

DESCRIPTION: DUAL PORT MAP DECODER(128KBYTE ONBOARD RAM)
             SET FOR VMEBUS ADDRESSES 0-1FFFF


PAL16L2                                                    7-23-84
U28-SHEET 8      MVME121,MVME123                           CKSM= 07CC
PALDP21                                                    REV A

A23 A22 A21 A20 A19 A18 A17 /LWORD /IACK GND
AM1 AM3 AM2 AM4 15 /DPMATCH AM0 AM5 VMEAV VCC

DPMATCH = /A23*/A22*/A21*/A20*/A19      ;$000000-$07FFFF
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*AM2*AM1*/AM0     ;STANDARD PRIV PROG

        + /A23*/A22*/A21*/A20*/A19      ;$000000-$07FFFF
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*AM2*/AM1*AM0     ;STANDARD PRIV DATA

        + /A23*/A22*/A21*/A20*/A19      ;$000000-$07FFFF
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*/AM2*AM1*/AM0    ;STANDARD NON-PRIV PROG

        + /A23*/A22*/A21*/A20*/A19      ;$000000-$07FFFF
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*/AM2*/AM1*AM0    ;STANDARD NON-PRIV DATA

DESCRIPTION: DUAL PORT MAP DECODER(512KBYTE ONBOARD DYNAMIC RAM)
             SET FOR VMEBUS ADDRESSES 0-7FFFF
```

2.3.13.2 Changing U28. The device used in U28 is a PAL16L2 with a maximum propagation delay time of 35 ns. PAL16L2's are not reprogrammable so the new firmware in U28 requires a new PAL16L2. When changing the source code for U28 only the lines that affect the ADDRESS should be changed. The other lines must remain intact for proper operation of the module. Below are sample ADDRESS source lines and the resultant base addresses.

```
/A23*/A22*/A21*/A20*/A19*/A18* A17 ;ADDRESS=$20000-$3FFFF
/A23*/A22*/A21*/A20*/A19* A18*/A17 ;ADDRESS=$40000-$5FFFF
/A23*/A22*/A21*/A20*/A19* A18* A17 ;ADDRESS=$60000-$7FFFF
```

SAMPLE ADDRESS LINES FOR 128KBYTES OF ONBOARD RAM

```
/A23*/A22*/A21*/A20* A19       ;ADDRESS=$80000-$FFFFF
/A23*/A22*/A21* A20*/A19       ;ADDRESS=$100000-$17FFFF
/A23*/A22*/A21* A20* A19       ;ADDRESS=$180000-$1FFFFF
```

SAMPLE ADDRESS LINES FOR 1/2MBYTE OF ONBOARD RAM

2.4 INSTALLATION

The following paragraphs discuss installation of the MPU module into a VME chassis and the connection of an RS-232C cable. Before inserting the module into the VMEbus chassis ensure that the desired EPROM/ROM devices are installed and configured and that all other headers, switches, and PAL's are configured for desired operation.

2.4.1 Module Installation

Now that the module is ready for installation, proceed as follows:

a. Turn all equipment power OFF.

CAUTION

INSERTING/REMOVING MODULES WHILE POWER IS APPLIED COULD RESULT IN DAMAGE TO MODULE COMPONENTS.

b. The module may be installed into any double wide unused card slot which has a VMEbus backplane on the connector P1 side.

2

c. Ensure that the BG* and IACK* daisy-chains are complete on the P1 backplane. Each daisy-chain line must be connected from IN to OUT at each connector on the P1 backplane. This must either be a direct (shorted) connection or an indirect (gated) connection. Modules that use a daisy-chain line automatically provide an indirect connection from that line's IN pin to the OUT pin on the connector into which the module is inserted. If the module does not use a daisy-chain line, then it should automatically provide a direct connection from that line's IN pin to the OUT pin on the connector into which the module is inserted. However, some modules do not provide direct connections. With these modules the daisy-chain signal line in question must be shorted at that connector using the means provided on the P1 backplane (generally shorting jumpers on a header). In addition, any slots that do not have a module inserted into the P1 connector, require that all the daisy-chain signal lines be shorted using the means provided on the backplane. Daisy-chain signal lines that are indirectly connected by a module should not be shorted on the backplane at that slot. The MPU module connects all the daisy-chain signal lines indirectly (except for the IACKIN*-IACKOUT* signal lines which it connects directly), so that none of the daisy-chain signal lines should be shorted on the backplane at the slot where it resides.

d. Carefully slide module into card slot. Be sure module is seated properly into connectors on the P1 backplane. Fasten module in chassis with front panel screws provided.

e. Make sure that all devices on the VMEbus respond to unique address locations. More than one VMEbus device may not respond to the same address and address modifier combination. For example, if two MPU modules reside on the same P1 backplane, the RAM dualport base address must be programmed differently on one module then it is on the other.

f. Ensure that only one module responds to any one IRQ signal line on the VMEbus.

g. Ensure that a system controller module is present at the highest priority slot of the P1 backplane (the MPU module is not a system controller).

h. Make sure that forced air cooling is in place for the modules in the system.

i. Turn equipment power ON.

2.4.2 Terminal Connection

The RS-232C serial port on the front panel is configured for terminal operation only. A 25-pin RS-232C cable may be connected to the front panel connector with the other end connected to a compatible terminal. Refer to Table 5-2 for detailed information on signals supported.

CHAPTER 3

OPERATING INSTRUCTIONS

3.1 INTRODUCTION

This chapter provides the necessary information to use the MPU module in a system configuration.

3.2 CONTROLS AND INDICATORS

The MPU module has a RESET switch, an ABORT switch, a mode switch, a FAIL indicator, a HALT indicator, and a RUN indicator. All switches and indicators are located on the front panel of the module.

3.2.1 RESET Switch

The reset performed by the RESET switch is a board-level reset only. A system level reset is performed by the system controller module that is needed to operate with the MPU module. The MPU is not immediately reset when the RESET switch is depressed, but instead, waits for the current access of the MPU to be completed before entering the reset state. This sequence prevents short cycling onboard and unintentional VMEbus accesses. The module is immediately reset by SYSRESET*.

3.2.2 ABORT Switch

A software ABORT switch is located on the front panel. Programs may be aborted with this switch. Depressing this switch causes a level 7 interrupt to be input to the MPU. The ABORT switch does not abort the MPU during the execution of the STOP instruction.

3.2.3 Mode Switch

The mode switch (section 4 of S3) provides the ability through hardware for the MPU to fetch the reset stack pointer and Program Counter (PC) from either the onboard ROM/EPROM sockets, or from a VMEbus resource that can respond to Address Modifier (AM) code 1E or 16. The MVME050 controller module can perform this task. Section 3 of switch S3 selects either AM code 16 or 1E for VMEbus accesses of reset vector and stack pointer. The mode switch may be disabled through header J24. Header J24 can be configured to force either the normal or alternate mode of operation. The result of the header configuration and mode switch position is also readable through the MSR.

3.2.4 FAIL Indicator

The red LED FAIL indicator is lit whenever the MPU enters a halted state (MPU driving the HALT line low -- usually the result of a double bus fault). The indicator is lit when the MPU is being reset. The indicator is lit when the module fail bit of the MCR is set (whenever the FAIL bit is set, the module also drives the SYSFAIL* line low on the VMEbus). The dual port lock condition lights the indicator momentarily. Refer to Table 3-1 for details.

3.2.5 HALT Indicator

The red LED HALT indicator is lit whenever the MPU enters a halted state (usually the result of a double bus fault). The indicator is lit when the MPU is being reset. The dual port lock condition lights the indicator momentarily. Refer to Table 3-1 for details.

3.2.6 RUN Indicator

The green LED RUN indicator is lit whenever the Address Strobe (AS*) pin of the MPU is true and off when it is false. Refer to Table 3-1 for details.

TABLE 3-1. Interpretation of Front Panel Indicators

FAIL	HALT	RUN	DESCRIPTION
OFF	OFF	OFF	No power applied to the module, or processor running but is not the local bus master.
OFF	OFF	ON	Normal operation.
ON	ON	OFF	Module is being reset or MPU has halted (usually the result of a double bus fault).
ON	OFF	ON	MPU is executing code but BRDFAIL (bit 0) in the MCR is set to 1. Bit may have been set to 1 by software, such as self-test routine, or the bit was not cleared to 0 by software since the last time the module was reset.
ON	OFF	OFF	Bit 0 in the MCR is set to 1 either by software or reset and MPU is not local bus master.
ON	ON	ON	MPU is running and dual port locks are occurring. Frequency of dual port lock determines intensity of halt and fail.

NOTE: All other combinations are illegal. If they occur, there is most likely some type of hardware malfunction.

3.3 MPU MODULE MEMORY MAP

The MPU module memory map as viewed from the onboard MPU is shown below:

ADDRESS RANGE	D15 D08 D07 D00
000000 000007	ONBOARD ROM (IF ONBOARD MODE)/VMEbus (IF OFFBOARD MODE) FOR FIRST 4 MEMORY CYCLES AFTER RESET. ONBOARD RAM THEREAFTER.
000008 01FFFF (07FFFF)	ONBOARD RAM - 128Kb 512Kb (OPTIONAL)
020000 (080000) EFFFFF	VMEbus
F00000 F0FFFF	ONBOARD ROM
F10000 F1FFFF	RESERVED FOR ONBOARD ROM EXPANSION
F20000	RESERVED (1) | MFP GPIP
F20002	RESERVED (1) | MFP AER
F20004	RESERVED (1) | MFP DDR
F20006	RESERVED (1) | MFP IERA
F20008	RESERVED (1) | MFP IERB
F2000A	RESERVED (1) | MFP IPRA
F2000C	RESERVED (1) | MFP IPRB
F2000E	RESERVED (1) | MFP ISRA
F20010	RESERVED (1) | MFP ISRB
F20012	RESERVED (1) | MFP IMRA
F20014	RESERVED (1) | MFP IMRB
F20016	RESERVED (1) | MFP VR
F20018	RESERVED (1) | MFP TACR
F2001A	RESERVED (1) | MFP TBCR
F2001C	RESERVED (1) | MFP TCDCR

3

ADDRESS RANGE	D15 D08	D07 D00
F2001E	RESERVED (1)	MFP TADR
F20020	RESERVED (1)	MFP TBDR
F20022	RESERVED (1)	MFP TCDR
F20024	RESERVED (1)	MFP TDDR
F20026	RESERVED (1)	MFP SCR
F20028	RESERVED (1)	MFP UCR
F2002A	RESERVED (1)	MFP RSR
F2002C	RESERVED (1)	MFP TSR
F2002E	RESERVED (1)	MFP UDR
F20030 F3FFFF	THE ABOVE MFP REGISTERS OCCUR REPEATEDLY IN THIS SPACE	
F40000	VME120 CONTROL REGISTER	RESERVED
F40002 F5FFFF	VME120 CONTROL REGISTER REPEATS IN THIS SPACE	RESERVED
F60000	MMU AST 0 (2)	RESERVED
F60002	MMU AST 1 (2)	RESERVED
F60004	MMU AST 2 (2)	RESERVED
F60006	MMU AST 3 (2)	RESERVED
F60008	MMU AST 4 (2)	RESERVED
F6000A	MMU AST 5 (2)	RESERVED
F6000C	MMU AST 6 (2)	RESERVED
F6000E	MMU AST 7 (2)	RESERVED
F60010	MMU AST 8 (2)	RESERVED
F60012	MMU AST 9 (2)	RESERVED
F60014	MMU AST 10 (2)	RESERVED
F60016	MMU AST 11 (2)	RESERVED

ADDRESS RANGE	D15 D08	D07 D00
F60018	MMU AST 12 (2)	RESERVED
F6001A	MMU AST 13 (2)	RESERVED
F6001C	MMU AST 14 (2)	RESERVED
F6001E	MMU AST 15 (2)	RESERVED
F60020	MMU AC0 (2)	MMU AC1 (2)
F60022	MMU AC2 (2)	MMU AC3 (2)
F60024	MMU AC4 (2)	MMU AC5 (2)
F60026	MMU AC6 (2)	MMU AC7 (2)
F60028	MMU AC8 (2)	MMU DP (2)
F6002A	RESERVED	MMU IVR (2)
F6002C	RESERVED	MMU GSR (2)
F6002E	RESERVED	MMU LSR (2)
F60030	RESERVED	MMU SSR (2)
F60032	RESERVED	RESERVED
F60034	RESERVED	RESERVED
F60036	RESERVED	RESERVED
F60038	RESERVED	MMU IDP (2)
F6003A	RESERVED	MMU RDP (2)
F6003C	RESERVED	MMU DIRECT TRANSLATION
F6003E	RESERVED	LOAD DESCRIPTOR
F60040 F7FFFF	THE ABOVE MMU REGISTERS OCCUR REPEATEDLY IN THIS SPACE	
F80000	RESERVED	
F80002	BANK 2 CACHE CLEAR (READ OR WRITE CYCLE)	

ADDRESS RANGE	D15 D08 D07 D00
F80004	BANK 1 CACHE CLEAR (READ OR WRITE CYCLE)
F80006	BANKS 1 AND 2 CACHE CLEAR (READ OR WRITE CYCLE)
F80008 F9FFFF	THE ABOVE CACHE CLEAR AREAS APPEAR REPEATEDLY IN THIS SPACE
FA0000 FEFFFF	VMEbus
FF0000 FFFFFF	VMEbus (GLOBAL SHORT I/O PAGE)

NOTES: Bank 1 is the user half and bank 2 is the supervisory half of the cache when the cache is configured as 1/2 user and 1/2 supervisory. If the whole cache is supervisory or user or mixed then cache should be completely cleared using F80006.

(1) Accesses to only the even bytes of these locations causes the MPU module to hang up until reset occurs.

(2) Only if the MMU is present (accesses to these locations when no MMU is present causes the MPU module to hang up until reset occurs).

3.3.1 MPU Module Memory Map as Viewed from the VMEbus

The onboard RAM appears to the VMEbus at the base address programmed into U28. The factory configuration of U28 causes the base address of onboard RAM as viewed from the VMEbus to be 000000. If U28 is changed (refer to paragraph 2.3.13.2 for instructions to change U28), then the user must be aware that RAM now appears at a different address for the VMEbus. For example, if the MPU executes a routine that tells a Direct Memory Access (DMA) module on the VMEbus to fill the onboard RAM with a program, the MPU must give the correct address (which is different from the address at which it executes the program) to the DMA module.

CHAPTER 4

FUNCTIONAL DESCRIPTION

4.1 INTRODUCTION

This chapter provides the overall block diagram level description for the MPU module. The general description provides a overview of the module, followed by a detailed description of each section of the module. Figure 4-1 shows the block diagram of the MPU module.

4.2 GENERAL DESCRIPTION

During normal operation, the MPU fetches and executes instructions that were already loaded in the logical instruction cache memory. If there is a cache miss (instruction not in cache, data r/w, or CPU cycle), the hardware activates the MMU causing it to generate physical addresses, which are decoded to access either onboard memory and I/O, or VMEbus memory and I/O. If the MPU was fetching a cachable instruction when the cache miss occurred, then the cache is updated with the new instruction. The entire cache is flushed by accessing a specific location in the memory map, or by resetting the module.

To minimize access time from the onboard memory when a cache miss occurs, the onboard dynamic memory begins its access cycle using the lower nine address lines for its row address. It then waits for the physical addresses from the MMU to become valid, and for a select from the address decoder to complete the column addressing. The lowest nine address lines are always required to be nontranslatable with this mode of operation, and restricts memory segmentation to a minimum size of 1024 bytes.

Programs are aborted by depressing the ABORT switch on the front panel of the module. Depressing this switch interrupts the MPU at level 7. The module is reset whenever the entire system is reset (SYSTEM RESET line on the bus goes low), or locally by depressing the RESET switch on the front panel of the module. The RESET switch on the MPU module does not generate a global reset because the MPU module is always used with a system controller module that generates the global reset.

The two ROM/EPROM sockets are normally used to hold the user-supplied self-test/boot load program. Immediately after a module reset, the MPU fetches its stack pointer and PC from the socket pair. However, if the front panel mode select switch on the MPU module is placed in its alternate position before the module reset, the MPU fetches its stack pointer and PC from the VMEbus at locations 0-7 using AM code 1E or 16.

A serial RS-232C debug terminal port connector is located on the front panel of the MPU module. This connector is configured for terminal use only. The baud rates are software programmable for 300 to 9600 baud. The serial port is normally used for debugging.

The onboard RAM is accessible from the VMEbus by multi-porting the onboard local bus. The intention of the VMEbus port to the local RAM is to allow programs to be loaded into the onboard RAM from DMA type disk controller or Local Area Network (LAN) module. The address is set by the Programmable Array Logic (PAL), that must be replaced with a user programmed unit, if another address is required. For multiprocessor systems, only one MPU can use the original address decoder. All others must have new decoder.

4.3 BLOCK DIAGRAM DESCRIPTION

The block diagram is shown in Figure 4-1. The MPU module operates through the following functional blocks.

- MPU
- MMU
- Cache
- RAM
- Multi-port controller
- Bus requester
- Interrupt handler
- VMEbus interface
- I/O and control
- ROM

4.3.1 MPU

The MC68010 MPU in a pin grid array package is the modules engine. The MPU and its associated circuitry are designed to run at 12.5 MHz. The clock is a 25 MHz oscillator divided by a flip-flop. Although the circuitry is designed to run at 10 MHz or 12.5 MHz depending on the version of the module.

4.3.2 MMU

The MMU is an MC68451 in a pin grid array package. The MMU is able to translate A8-A23 of the processor. However, to improve access time from the onboard memory, the MMU translates A10-A23 only. This restriction saves one wait cycle when accessing onboard memory, but limits the smallest segment size to 1024 bytes.

On versions of the module where the MMU is not present, the MMU is replaced with a printed circuit jumper block. When accesses are made to the MMU address space with no MMU on module (MVME122/123), the MPU hangs up waiting until reset occurs. Do not attempt to access the MMU address space if no MMU is present.

4.3.3 Cache

The cache memory is single set-associative with a block size of 1 word. It has a total capacity of 2048 sixteen bit words, and is for instruction storage only. As shown in the block diagram, the cache is located between the MPU and MMU. This placement of cache provides no wait state operation by the MPU on cache hit cycles. The following is an in depth discussion of the cache on the MPU module.

4.3.3.1 Cache Organization. The cache has four jumper controlled configurations:

a. 2K words of user program space cache.

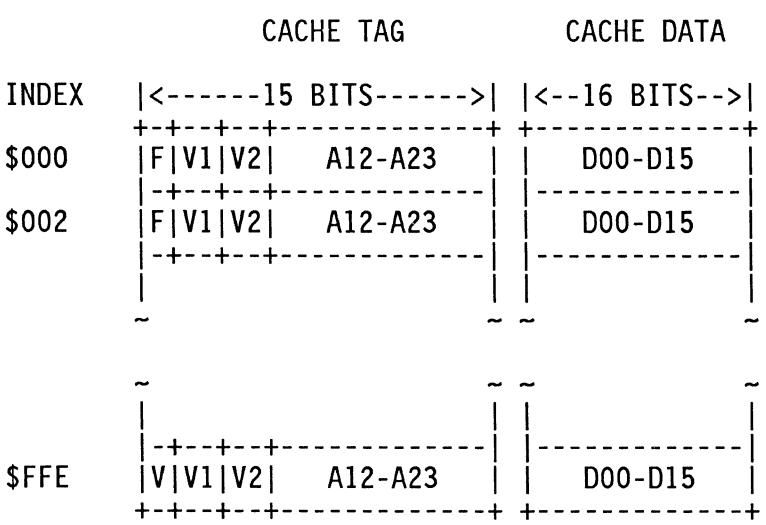
b. 2K words of supervisory program space cache.

c. 2K words of user program space/supervisory program space (mixed) cache.

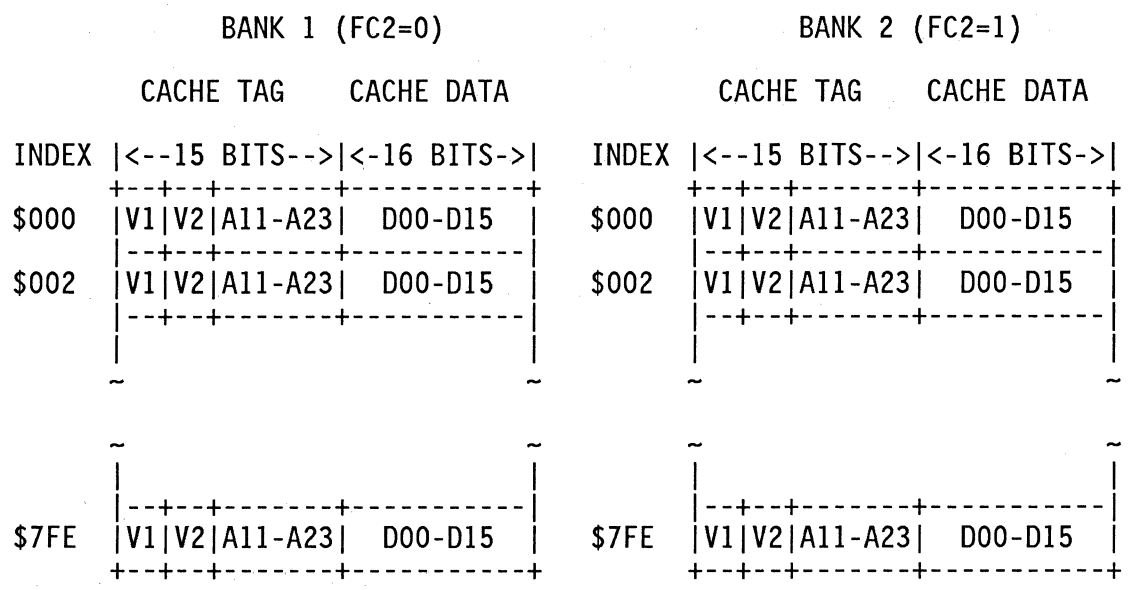
d. 1K word of user space cache and 1K word of supervisory space cache.

Configurations a,b,c each have one bank of 2K entries. Configuration d has two banks of 1K entries each. Each entry within cache is addressed by the MPU lower address lines (A1-A10 for configuration d or A1-A11 for configurations a,b,c). These lower address lines are called the index. For every possible index value there is a unique cache entry. Each cache entry contains two information fields - cache data (16 bits), and cache tag (15 bits). Refer to the figure below:

CACHE ORGANIZATION FOR CONFIGURATIONS A,B,C



CACHE ORGANIZATION FOR CONFIGURATION D



Cache data is the data that has been cached from main memory. Cache tag consists of the following:

Tag bits A11 (or A12) through A23 constitute the upper logical address of the data in main memory from where the cache data was copied.

Tag bits V1 and V2 indicate the validity of cache data. When either of them is 0, the cache data for that entry is not valid. When they are both 1, it is valid.

Tag bit F matches the value of FC2 when the cache data was originally copied from main memory (FC2=1 for supervisory address space and 0 for user address space).

This organization of cache is called single set-associative. It causes each entry in cache to align itself with specific main memory locations. For example, for configurations a,b,c, cache entry 1 (index = 000) is only capable of caching memory locations 0, 1000, 2000, 3000, etc. Cache entry 4 (index = 006) is only capable of of caching memory locations 0006, 1006, 2006, 3006. This means that at any given time, cache entry 4 can have cached memory location 0006 or 1006 or 2006 or 3006, etc., but never 0000-0004 or 0008-1004 or 1008-2004, etc. Adjacent cache entries always have adjacent indexes but do not necessarily come from the same block of memory. For example, cache entry 8 (index=00E) may contain data from memory location 200E while cache entry 7 (index=00C) contains data from memory location 400C and cache entry 6 (index=00A) contains data from memory location 1000A. See figure below:

SAMPLE CACHE STATE USING CONFIGURATION C

	CACHE TAG				CACHE DATA	
INDEX	F	V1	V2	A23-A12	D00-D15	CACHED MEMORY ADDRESS
$000	1	1	1	$001	$4E71	$1000
$002	1	1	1	$008	$60F0	$8002
$004	X	0	0	XXXX	XXXXX	XXXXX
$006	0	1	1	$008	$4E71	$8006
$008	0	1	1	$008	$60FC	$8008
$00A	0	1	1	$010	$4E71	$1000A
$00C	0	1	1	$004	$60FE	$400C
$00E	0	1	1	$002	$60FA	$200E
~	~	~	~	~	~	~
$FFE	1	1	1	$001	$4AFB	$1FFE

4

4.3.3.2 Cache Operation. The cache performs its operations only during memory access cycles by the MPU. It operates on one entry per memory access cycle (except for cache flush operations). The cache does not perform any function during RAM refresh cycles, or VMEbus to RAM access cycles.

At the beginning of each processor memory access cycle (before any devices on the module can be selected) the MPU lower address lines index a specific entry in the cache. The valid bits for that entry are checked and the tag from that entry is compared to the MPU upper address lines and to FC2. Depending on the results of the validity check and the comparison, the rest of the memory access cycle is classified as one of four types:

Cache hit cycle
Cache validate cycle
Cache invalidate cycle
Cache ignore cycle

Cache Hit Cycle

When the cache has determined that the current cycle is a cache hit cycle, the following occurs:

a. All selects to non-cache devices are disabled.

b. The cache presents the cache data word to the MPU.

c. The cache drives [DTACK*] to the MPU in time for the cycle to complete in 0 wait states.

Cache Validate Cycle

When the cache has determined that the current cycle is a cache validate cycle, the following occurs:

a. The selects to the rest of the module are allowed to happen.

b. The selected device presents its data to the MPU.

c. The cache latches the data that is being driven onto D00-D15 by the selected device, into the cache data word.

d. The cache latches the value that is being driven onto A11 (or A12) and FC2 by the MPU, into the cache tag.

e. The cache sets V1 and V2 in the cache tag.

f. The MPU finishes the cycle with the number of wait cycles indicated by the selected device.

Cache Invalidate Cycle

When the cache has determined that the current cycle is a cache invalidate cycle, the following occurs:

a. The selects to the rest of the module are allowed to happen.

b. The selected device may or may not present data to the MPU.

c. The cache clears V1 or V2 or both.

d. The MPU finishes the cycle with the number of wait cycles indicated by the selected device.

Cache Ignore Cycle

When the cache has determined that the current cycle is a cache ignore cycle, the following occurs:

a. The selects to the rest of the module are allowed to happen.

b. The selected device may or may not present data to the MPU or the MPU drives the data bus.

c. The cache does not latch anything.

d. The MPU finishes the cycle with the number of wait cycles indicated by the selected device.

Memory Access Cycle Type Determination

Several factors affect which of the four types of memory access cycle occur:

a. Read to user data space always causes a cache ignore cycle.

b. Read to supervisory data space always causes a cache ignore cycle.

c. Write to user data space causes a cache invalidate cycle if the cache entry addressed by the index has all of the following:

    1. Both V1 and V2 are true.

    2. The cache tag address matches the current upper logical address from the MPU.

    3. The F bit in the tag is 0.

   Otherwise, write to user data space causes a cache ignore cycle.

d. Write to supervisory data space causes a cache invalidate cycle if the cache entry addressed by the index has all of the following:

    1. Both V1 and V2 are true.

    2. The cache tag address matches the current upper logical address from the MPU.

    3. The F bit in the tag is 1.

   Otherwise, write to supervisory data space causes a cache ignore cycle.

Software Considerations

The operation of the cache is transparent to software in most situations. However, there are times when software must be written with the cache in mind in order to avoid stale cache problems. Below is a list of stale cache sources and possible solutions:

Problem 1 - Code from main memory is cached in user space, the same main memory is written with new code in supervisory space and then the MPU attempts to execute the new code in user space or vice-versa.

Solution - Flush cache before executing from the new address space.

Problem 2 - Code from main memory is cached, a device other than the onboard MPU DMA's into that memory space, and then the MPU attempts to execute the new code.

Solution - Flush cache before executing the new code.

Problem 3 - Code from main memory is cached, the descriptors in the MMU are changed so that the logical address from where the code was cached translates to a different physical address. The MPU attempts to execute the new code at the same logical address.

Solution - Flush cache when changing descriptors in the MMU.

Problem 4 - The software uses PC relative reads of data (that appears as program space on the function code pins of the MPU). It writes to that data in another address space, and then it uses PC relative reads of those same logical addresses expecting to have updated information.

Solution - Avoid the use of PC relative instructions for data that is to be altered or flush cache after altering the data.

If cache is operating in configuration d, then only one of its banks may need to be flushed in some of the above situations. Flushing the cache degrades the cache hit rate and should only be used where necessary.

4.3.4 RAM

The RAM array provides eighteen 64K x 1 or 256K x 1, 150 nanosecond access time dynamic RAMs with either 128 row 2 millisecond or 256 row 4 millisecond refreshing requirements. Odd byte parity is always written to the RAM's. Parity detection is software enabled through the module control register if the hardware parity enable jumper is installed. Parity check is always performed on both the lower byte and upper byte of a word of onboard dynamic RAM whether either byte or both bytes are read.

4.3.4.1 Refresh. The RAM is refreshed every 30 microseconds with two RAS-only refresh cycles. During refresh cycles, all onboard processing stops.

4.3.4.2 Local Accesses. The dynamic RAM is accessed locally starting at physical address 0. The row address is strobed into the RAMs from the lower address lines when the address strobe from the MPU goes low. If the translated addresses from the MMU result in selecting the RAM, then the RAM is enabled for reading or writing data. If the RAM is not selected, the RAMs do not receive a column address strobe. This results in a RAS-only refresh cycle that completes when the MPU completes its current access.

4.3.4.3 Accesses From The VMEbus. The RAM is accessed from the VMEbus starting at a base address specified by the user configurable PAL. The RAM is contiguous starting at that base address.

4.3.5 Multi-Port Controller

The Multi-Port Controller (MPC) resolves accessing conflicts between the MPU, VMEbus, and the onboard RAM refresh controller. It also generates the local bus time-out when the onboard MPU cannot attain VMEbus mastership within the prescribed amount of time.

When the MPC receives a refresh request from the RAM refresh timer, or a VME access request to the onboard RAM, it issues a request to the onboard MPU for mastership of the onboard local bus. When the MPU gives up local bus mastership, the MPC gives bus mastership to either the RAM refresh control logic or the VMEbus. In the event of simultaneous requests, refresh has the highest priority.

The MPC also handles MPU/VMEbus lock conditions. The lock occurs when the onboard MPU attempts to access the VMEbus at the same time a VMEbus master attempts an access to the onboard RAM. In this case, the MPC issues a rerun handshake to the MPU. The MPU gives up mastership of the local bus at this time. If the instruction the MPU was executing is a Test and Set (TAS), it does not rerun the cycle, but executes a bus error exception instead. It is up to the software to recover from this instruction. The software should merely RTE from such a condition.

If the MPU attempts to execute a cycle on the VMEbus and the cycle is not completed within 220 microseconds, the RAM is not properly refreshed. The time required to complete a VMEbus cycle consists of the time required to obtain VMEbus mastership plus the time required for a VMEbus slave to respond once VMEbus mastership has been obtained and the MPU module strobes have been driven true on the VMEbus. Therefore, the total time from the requesting of mastership to the activation of DTACK* or BERR* by the VMEbus slave must not exceed 220 microseconds. The MPC issues a bus error to the MPU if the bus

mastership is not obtained within the hardware selected local time-out period (Tlto). The VMEbus system controller must activate BERR* on the VMEbus if no VMEbus slave responds to the MPU module within 220 microseconds-Tlto. When there is a local bus timeout or a dual port lock on TAS the software should merely RTE.

NOTE

If a local bus time-out occurs at the same time that a dual port access occurs, the MPU response may be undefined. The system should be organized such that dual port accesses do not happen at the end of long periods of withholding the VMEbus from the module. (Long period = the selected local bus time-out on the module.)

4.3.6 Bus Requester

The bus requester requests VMEbus mastership on any one of four request levels (user configurable through jumpers), and fully supports the bus grant daisy-chain. The bus requester operates in the Release on Request (ROR) mode and requests bus mastership only if the MPU attempts a VMEbus transfer and the module does not already have VMEbus mastership.

4.3.7 Interrupt Handler

The interrupt handler gives the onboard MPU the ability to sense and respond to all onboard interrupts, all seven VMEbus interrupts, VMEbus ACFAIL*/SYSFAIL*, and the ABORT switch. All the onboard interrupts and ACFAIL*/SYSFAIL*, interrupt the MPU indirectly via the MC68901. The VMEbus interrupts and the ABORT switch interrupt the MPU directly. All interrupt requests to the MPU are disabled by the IE* bit in the MCR.

4.3.7.1 VMEbus Interrupts. The VMEbus interrupts are independently routed through onboard headers to provide a means whereby the onboard MPU can selectively ignore individual VMEbus interrupts. All seven VMEbus interrupts are routed to the MPU at their original request level. For example, IRQ1* requests level one interrupt to the MPU. The vector that is passed to the MPU during acknowledgement of VMEbus interrupts is the vector passed by the interrupting VMEbus slave.

4.3.7.2 ABORT Switch Interrupts. The ABORT switch interrupts the MPU on level seven. There is no vector passed during the acknowledgement of an abort. Instead, VPA is driven true, causing the MPU to use the exception table address for autovector level 7 ($7C).

NOTE

The ABORT switch does not interrupt the MPU during the execution of a STOP instruction.

4.3.7.3 MC68901 Interrupts. All the onboard interrupts plus ACFAIL*/SYSFAIL* are routed indirectly to the MPU via the GPIO pins of the MC68901. Onboard serial I/O and timer interrupts are internal interrupts from the MC68901 and do not come in through the GPIO pins. All the interrupts that come through the MC68901 are funneled into one interrupt request level (level 6). The vector that is passed for each of these interrupts is partially programmable and partially fixed in the MC68901 (refer to the MC68901 Data Sheet for details).

4.3.7.4 Interrupt Source and Vectors. The following table summarizes all the interrupt sources on the module:

INTERRUPT SOURCE	PATH	VECTOR PASSED	EXCEPTION ADDRESS	LEVEL
VMEbus IRQ1*	DIRECT	FROM INTERRUPTING VMEbus SLAVE	4 X VECTOR	1
VMEbus IRQ2*	DIRECT	SAME AS ABOVE	4 X VECTOR	2
VMEbus IRQ3*	DIRECT	SAME AS ABOVE	4 X VECTOR	3
VMEbus IRQ4*	DIRECT	SAME AS ABOVE	4 X VECTOR	4
VMEbus IRQ5*	DIRECT	SAME AS ABOVE	4 X VECTOR	5
VMEbus IRQ6*	DIRECT	SAME AS ABOVE	4 X VECTOR	6
SECTION 2 (J21 2-3) (NOT DEBOUNCED) / CACHEHIT* (J21 1-2)	MC68901 GPIO0	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
SECTION 1 (NOT DEBOUNCED)	MC68901 GPIO1	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
VMEbus BERR	MC68901 GPIO2	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
PARITY ERROR	MC68901 GPIO3	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
MC68901 TIMER D	MC68901	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
MC68901 TIMER C	MC68901	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
MODE (NOT DEBOUNCED)	MC68901 GPIO4	REFER TO MC68901 DATA SHEET	4 X VECTOR	6

MMU IRQ* (MUST BE ENABLED IN MMU)	MC68901 GPIO5	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
MC68901 TIMER B	MC68901	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
TRANSMIT ERROR (SERIAL)	MC68901	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
TRANSMIT BUFFER EMPTY (SERIAL)	MC68901	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
RECEIVE ERROR (SERIAL)	MC68901	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
RECEIVE BUFFER FULL (SERIAL)	MC68901	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
MC68901 TIMER A	MC68901	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
DTR*	MC68901 GPIO6	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
ACFAIL* (J2 1-2) / SYSFAIL* (J2 2-3)	MC68901 GPIO7	REFER TO MC68901 DATA SHEET	4 X VECTOR	6
VMEbus IRQ7*	DIRECT	FROM INTERRUPTING VMEbus SLAVE	4 X VECTOR	7
ABORT*	DIRECT	NONE	$7C	7

4.3.7.5 Enabling Interrupts. VMEbus interrupts are enabled/disabled using jumpers, ABORT is enabled/disabled using a jumper, and all interrupts that come through the MC68901 are enabled in the MC68901. In addition, most of the interrupt sources must be enabled at the source such as VMEbus interrupts and MMUIRQ*.

All interrupts are disabled when the IE* bit in the MCR is 1.

4.3.8 VMEbus Interface

The VMEbus interface supports both master and slave modes. In either mode, the address and data paths are A24:D16. In the master mode, the MPU module is capable of initiating the following types of cycles:

AM CODE	FUNCTION
XX	Interrupt Acknowledge
3E	Standard supervisory program access
3D	Standard supervisory data access
3A	Standard non-privileged program access
39	Standard non-privileged data access
1E,16	These codes are in the user-defined category of the VMEbus specification and are used for fetching an alternate stack pointer and program counter after a module reset occurs with the mode switch in its alternate position. A second switch determines what AM code is generated.
2D	Short supervisory I/O access
29	Short non-privileged I/O access

In the slave mode, the interface is capable of responding to the following types of cycles:

AM CODE	FUNCTION
3E	Standard supervisory program access
3D	Standard supervisory data access
3A	Standard non-privileged program access
39	Standard non-privileged data access

4.3.9 I/O and Control

The serial port, timers, and module status are implemented using an MC68901 multifunction peripheral device and some additional control logic. Details of the MC68901 can be found in the MC68901 data sheet. The device is capable of generating 16 internally prioritized interrupts on one interrupt level, and passing any one of the 16 unique vectors when the interrupt level is acknowledged. The interrupt level is fixed in hardware to level 6.

When the MPU accesses even bytes in the MC68901 address space, the MPU sets no response as either DTACK* or BERR*. Consequently, the MPU hangs up, waiting until reset occurs. Do not access even bytes in the MC68901 address space.

4.3.9.1 Module Status. The status of the MPU module is available on eight signal lines that are connected to the General Purpose I/O (GPIO) pins of the MC68901. In the general case the GPIO lines are programmable as inputs or outputs. However, on the MPU module, they must always be programmed as inputs to avoid buffer fight. Some of the signal lines that are connected to the GPIO pins are steady state type status lines, while others are pulsed status lines. Table 4-1 lists the signal names.

TABLE 4-1. Module Status

GPIO BIT NO.	SIGNAL NAME	DESCRIPTION	COMMENTS
0		Bit 0 is high when section 2 of S3 is open and low when it is closed.	Section 2 is connected only when J21 2-3 are connected.
0	CACHEHIT*	CACHEHIT* pulses low momentarily during a cache hit cycle. It is high at all other times.	CACHEHIT* is connected only when J21 1-2 are connected.
1		Bit 1 is high when section 1 of S3 is open and low when it is closed.	None
2	VMEBERR	VMEBERR pulses high momentarily when the VMEbus BERR* signal line goes true. It is low at all other times.	This only occurs when the MPU module is accessing the VMEbus.
3	PARERR	PARERR pulses high momentarily during an onboard RAM read cycle that results in a parity error. It is low at all other times.	None
4	Mode	Mode is low when the MPU module is configured to fetch its reset vectors from the VMEbus, and high when it is configured to fetch them from onboard ROM.	Mode is optionally controlled by section 4 of S3. (open=high and closed=low)
5	MMUIRQ*	MMUIRQ* is directly connected to the MC68451 IRQ pin. It is low when the MC68451 generates an interrupt request.	None
6	DTR*	DTR* is low when the terminal is ready. DTR* is high when it is not.	DTR* is the inverse of DTR signal line at the RS-232C connector.

TABLE 4-1. Module Status (cont'd)

GPIO BIT NO.	SIGNAL NAME	DESCRIPTION	COMMENTS
7	ACFAIL:	ACFAIL* is the ACFAIL* signal from the VMEbus.	This is only when J2 1-2 are connected.
7	SYSFAIL:	SYSFAIL* is the SYSFAIL* signal from the VMEbus.	This is only when J2 2-3 are connected.

NOTE: Only VMEBERR, PARERR, and MMUBERR status bits are available for software monitoring. But, there are two other sources of bus errors (TAS on dual port lock and local bus time-out). The software should treat the presence of BERR exception and absence of VMEBERR, PARERR, and MMUBERR as a "soft bus error" and should use the normal hardware recovery RTE that reruns the bus cycle.

4

4.3.9.2 Module Control Register. The MCR provides software control for various functions. The control register bits (outputs) are accessed at location F40000 and are defined as shown below:

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
WWP*	ALTCLR*	FREEZE	CACHEN	PAREN*	IE*	CTS*	BRDFAIL

BIT 7 - WWP*, when 0, causes faulty parity to be stored when any good onboard RAM location is written to by the MPU. When the same location is read, a parity error occurs. WWP*, when 1, allows correct parity to be written by the MPU. Correct parity is defined as being odd.

BIT 6 - ALTCLR*, when 1, allows bus error to start the alternate interrupt mode. When ALTCLR* is 0, it stops the alternate interrupt mode (if in effect), and disables bus errors from starting the alternate interrupt mode.

BIT 5 - When FREEZE is 1, the cache cannot be updated (except writes to cached locations invalidate then if in same FC2 space). When FREEZE is 0, cache can be updated. Note that freezing the cache does not necessarily prevent cache hits from occurring.

BIT 4 - When CACHEN is 0, the cache cannot be updated and no cache hits can occur. When CACHEN is 1, cache is enabled with restrictions imposed by the FREEZE bit.

BIT 3 - When PAREN* is 1, bus error exceptions can not occur because of parity errors in the onboard RAM. When it is 0, they can, if parity is enabled in hardware.

BIT 2 - When IE* is 1, no interrupt requests reach the processor. When IE* is 0, interrupt requests reach the processor.

BIT 1 - When CTS* is 0, the CTS line to the terminal is true. When CTS* is 1, the CTS line is to the terminal is false.

BIT 0 - When BRDFAIL is 1, the SYSFAIL* line on the VMEbus is driven low by the MPU module, and the FAIL indicator is lit. When BRDFAIL is 0, the SYSFAIL* is not driven low by the MPU module.

NOTE

The contents of the control register are readable at the same address.

Immediately after reset, all control register bits are set to logic 1 by hardware.

4.3.9.3 Timers. Four timers are provided onboard through the MC68901, and each timer is capable of generating an interrupt. For detailed information refer to the MC68901 Data Sheet. The timers are assigned as follows:

Timer C - Baud rate generator for the serial port.
Timer A - Software tick timer.
Timer B - Tick timer overflow/watchdog time-out.
Timer D - Delay mode only - unassigned by hardware.

NOTE

The watchdog time-out resets the MPU module when timer B output is set.

The XTAL input to the MC68901 is MPU clock divided by four (3.125 MHz for 12.5 MHz module and 2.5 MHz for 10 MHz module). The baud rates supported by the baud rate timer are programmable via timer C as shown below:

BAUD RATE	THEOR. FREQ 16 X CLOCK	12.5 MHz MODULE XTAL = 3.125 MHz PRE SCALE	TIMER C COUNT	ACT FREQ	ERROR	10 MHz MODULE XTAL = 2.5 MHz PRE SCALE	TIMER C COUNT	ACT FREQ	ERROR
9600	153600	10	1	156250	1.7	4	2	156250	1.7
4800	76800	4	5	78125	1.7	4	4	78125	1.7
2400	38400	4	10	39063	1.7	4	8	39063	1.7
1200	19200	4	20	19531	1.7	4	16	19531	1.7
600	9600	4	41	9527	.8	4	33	9470	1.4
300	4800	4	81	4823	.5	4	65	4807	.2
110	1760	16	55	1775	.9	10	71	1760.6	.03

4

4.3.10 ROM

Two 28-pin ROM sockets for user-supplied ROM/EPROM are provided. The sockets support 4K x 8 and 8K x 8 24-pin devices, and 8K x 8, 16K x 8, 32K x 8, and 64K x 8 28-pin parts. Configurations for the devices may be chosen by jumper positions on headers.

The ROM/EPROM is normally accessed as shown in the memory map in Chapter 3. However, when the module is reset, if the mode switch is in the normal position (disabled) prior to the reset, the MPU fetches its reset stack pointer and program counter from the first eight bytes of onboard ROM. Writes to ROM do not cause a bus error exception to occur.

4.3.11 Source of Bus Error Exceptions

There are several sources of bus errors. Some of the sources are fatal errors while others are not. The MSR can be read to determine what source a bus error came from. All the potentially fatal error sources are status bits in the MSR. The other sources are not. When a bus error has occurred, if there are no bus error status bits set in the MSR, then the error should be treated as benign, and the bus error service routine should execute an RTE to cause a processor rerun.

4.3.11.1 VMEbus BERR. VMEbus BERR is a status bit in the MSR. This bus error exception occurs when the VMEbus BERR* signal line is driven true during a VMEbus access by the MPU module. This can only happen after the MPU module has attained VMEbus mastership.

4.3.11.2 Onboard Parity Error. Onboard parity error is a status bit in the MSR. This bus error exception occurs when a location with bad parity is encountered in the onboard RAM while parity is enabled in the MCR. Bad parity in RAM can be caused by uninitialized RAM after power-up, a write to RAM with the WWP bit true in the MCR, or by a data failure in the RAM.

4.3.11.3 Dual Port Lock on TAS Error. Dual port lock on TAS is not a status bit. This bus error exception occurs when a retry happens during the execution of a TAS instruction. This is not a fatal error and should rerun with a processor rerun.

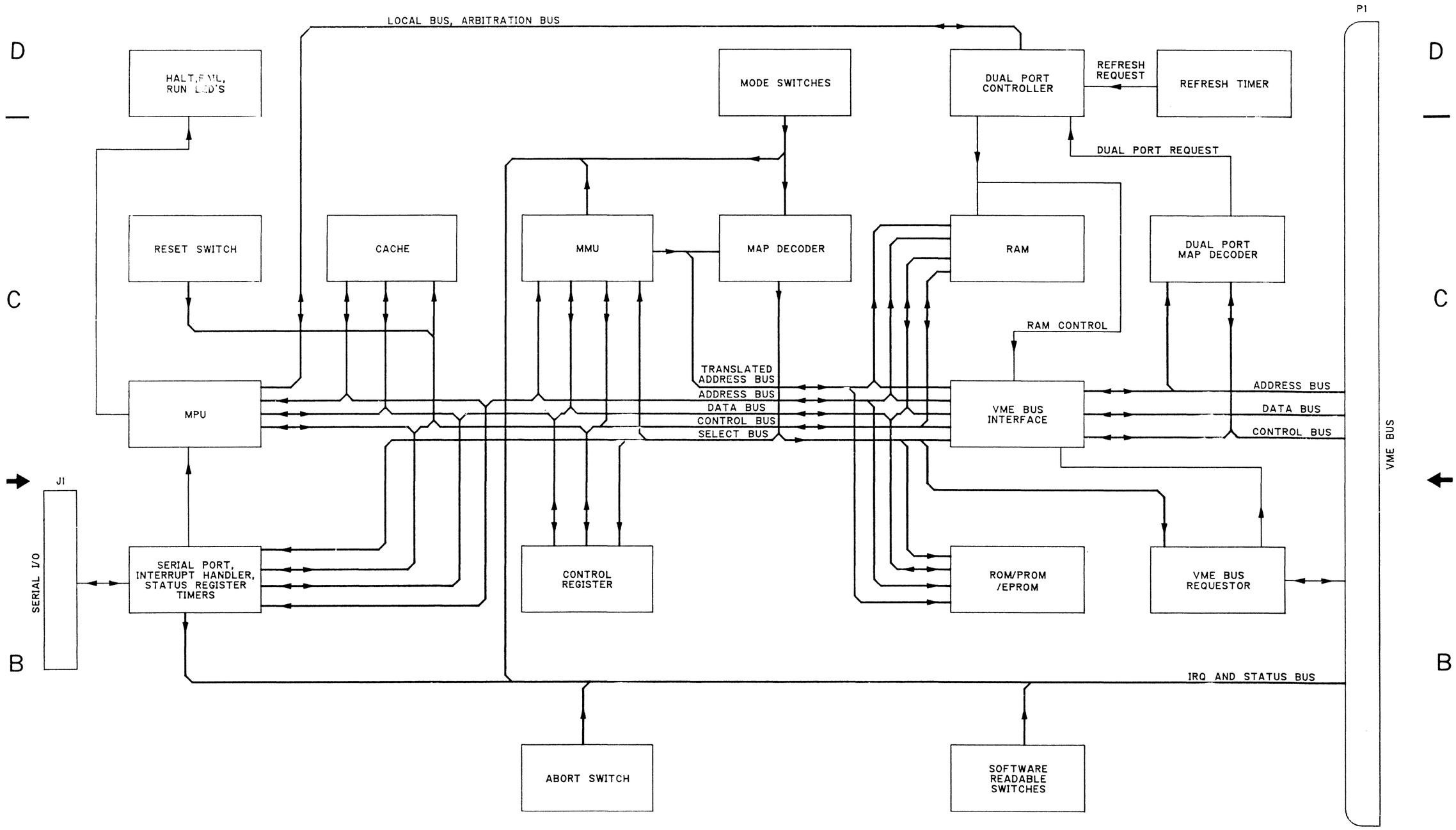
4.3.11.4 MMU Fault. MMU fault is not a status bit in the MSR, but is available as status in the MMU. This bus error exception occurs whenever the MMU drives its FAULT* pin low.

4.3.11.5 Local Bus Time-out. Local bus time-out is not a status bit. This bus error exception occurs when VMEbus mastership is not attained in the alloted time. Local bus time-out should be rerun with a processor rerun.

NOTE

If a local bus time-out occurs at the same time that a dual port access occurs, the MPU response may be undefined. The system should be organized such that dual port accesses do not happen at the end of long periods of withholding the VMEbus from the module. (Long period = the selected local bus timeout on the module.)

THIS PAGE INTENTIONALLY LEFT BLANK.



BLOCK DIAGRAM
63DW3293B REV B SH 2 OF 15

FIGURE 4-1. MPU Module Block Diagram

CHAPTER 5

SUPPORT INFORMATION

5.1 INTRODUCTION

This chapter provides the interconnection signals, parts list with parts location illustration, and schematic diagram for the MPU module.

5.2 INTERCONNECT SIGNALS

The MPU module interconnects with the VMEbus through connector P1.

5.2.1 Connector P1 Interconnect Signals

Connector P1 is a standard DIN 41612 triple row, 96 pin male connector. All Motorola VMEbus specifications are met by the MPU module. Table 5-1 lists each pin connection, signal mnemonic, and signal characteristic for the connector.

TABLE 5-1. Connector P1 Interconnect Signals

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
A1-A8	D00-D07	DATA bus (bits 0-7) - eight of 16 three-state bidirectional data lines that provide the data path between VMEbus master and slave.
A9	GND	GROUND
A10		Not used.
A11	GND	GROUND
A12	DS1*	DATA STROBE 1 - signal driven by VMEbus master that indicates a data transfer on data bus lines D08-D15.
A13	DS0*	DATA STROBE 0 - signal driven by VMEbus master that indicates a data transfer on data bus lines D00-D07.
A14	WRITE*	WRITE - signal driven by VMEbus master that specifies the direction of data transfers.
A15	GND	GROUND

TABLE 5-1. Connector P1 Interconnect Signals (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
A16	DTACK*	DATA TRANSFER ACKNOWLEDGE - signal driven by VMEbus slave that indicates that valid data is available on the data bus during a read cycle, or that data has been accepted from the data bus during a write cycle.
A17	GND	GROUND
A18	AS*	ADDRESS STROBE - the falling edge of this signal,driven by VMEbus master, indicates a valid address is present on the address bus.
A19	GND	GROUND
A20	IACK*	INTERRUPT ACKNOWLEDGE - signal driven by the VMEbus master that indicates a VME interrupt acknowledge cycle. A VME master has been interrupted on one of seven levels and is now acknowledging the specific interrupt with a service routine.
A21	IACKIN*	INTERRUPT ACKNOWLEDGE IN - IACKIN* and IACKOUT* form a daisy-chained acknowledge. The IACKIN* signal is connected directly to IACKOUT*.
A22	IACKOUT*	INTERRUPT ACKNOWLEDGE OUT - IACKIN* and IACKOUT* form a daisy-chained acknowledge. The IACKOUT* signal is connected directly to IACKIN*.
A23	AM4	ADDRESS MODIFIER (bit 4) - one of the three-state lines driven by VMEbus master that provide additional information about the address bus, such as size, cycle type, and/or data transfer bus master identification.
A24	A07	ADDRESS bus (bit 7) - one of 23 three-state lines driven by VMEbus master that specify an address in the memory map.
A25	A06	ADDRESS bus (bit 6) - same as A07 on pin A24.
	A26	A05 ADDRESS bus (bit 5) - same as A07 on pin A24.
	A27	A04 ADDRESS bus (bit 4) - same as A07 on pin A24.
	A28	A03 ADDRESS bus (bit 3) - same as A07 on pin A24.
	A29	A02 ADDRESS bus (bit 2) - same as A03 on pin A28.

TABLE 5-1. Connector P1 Interconnect Signals (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
A30	A01	ADDRESS bus (bit 1) - same as A03 on pin A28.
A31	-12 VDC	-12 Vdc Power - used by the logic circuits on the MPU module.
A32	+5 VDC	+5 Vdc Power - used by the logic circuits on the MPU module.
B1	BBSY*	BUS BUSY - this signal is driven low when the MPU module is the bus master.
	B2	Not used.
B3	ACFAIL*	AC FAILURE - Input signal that indicates a power failure has occurred.
B4	BG0IN*	BUS GRANT (bit 0) IN - bus-grant-in and bus-grant-out a daisy-chained bus grant. A grant received at the jumpered level indicates the MPU module may become the bus master. The remaining three bus-grant-in lines are connected directly to their respective bus-grant-out lines.
B5	BG0OUT*	BUS GRANT (bit 0) OUT - bus-grant-in and bus-grant-out form a daisy-chained bus grant. When a bus-grant-in is received at the jumpered level and the MPU is not awaiting bus mastership, the bus-grant-out signal is true on the respective level.
B6	BG1IN*	BUS GRANT (bit 1) IN - same as BG0IN* on pin B4.
B7	BG1OUT*	BUS GRANT (bit 1) OUT - same as BG0OUT* on pin B5.
B8	BG2IN*	BUS GRANT (bit 2) IN - same as BG0IN* on pin B4.
B9	BG2OUT*	BUS GRANT (bit 2) OUT - same as BG0OUT* on pin B5.
B10	BG3IN*	BUS GRANT (bit 3) IN - same as BG0IN* on pin B4.
B11	BG3OUT*	BUS GRANT (bit 3) OUT - same as BG0OUT* on pin B5.

TABLE 5-1. Connector P1 Interconnect Signals (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
B12-B15	BR0*-BR3*	BUS REQUEST (0-3) - the bus request at the jumpered level is true when the MPU requires bus mastership. When one or more bus request lines is true in the ROR mode, bus mastership is released.
B16-B19	AM0-AM3	ADDRESS MODIFIER (bits 0-3) - same as AM4 on pin A23.
B20	GND	GROUND
B21,B22		Not used.
B23	GND	GROUND
B24-B30	IRQ7*-IRQ1*	INTERRUPT REQUEST (7-1) - seven prioritized interrupt request inputs. Jumper enabled, level seven is the highest priority.
B31		Not used.
B32	+5 VDC	+5 Vdc Power - same as +5 VDC on pin A32.
C1-C8	D08-D015	DATA bus (bits 8-15) - eight of 16 three-state bidirectional data lines that provide the data path between VMEbus master and slave.
C9	GND	GROUND
C10	SYSFAIL*	SYSTEM FAIL - signal driven by the MPU module when fail bit is true in MCR. Also can be monitored in MSR.
C11	BERR*	BUS ERROR - an active low signal driven by VMEbus slave that indicates an error has occurred during a data transfer cycle.
C12	SYSRESET*	SYSTEM RESET - the system controller provides this input signal that causes a module level reset on the MPU module.

TABLE 5-1. Connector P1 Interconnect Signals (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
C13	LWORD*	LONGWORD - not driven. This terminated signal remains at a high level when the MPU module is bus master. The dual port RAM does not respond to VMEbus accesses when LWORD* is true.
C14	AM5	ADDRESS MODIFIER (bit 5) - same as AM4 on pin A23.
C15	A23	ADDRESS bus (bit 23) - same as A07 on pin A24.
C16	A22	ADDRESS bus (bit 22) - same as A07 on pin A24.
C17	A21	ADDRESS bus (bit 21) - same as A07 on pin A24.
C18	A20	ADDRESS bus (bit 20) - same as A07 on pin A24.
C19	A19	ADDRESS bus (bit 19) - same as A07 on pin A24.
C20	A18	ADDRESS bus (bit 18) - same as A07 on pin A24.
C21	A17	ADDRESS bus (bit 17) - same as A07 on pin A24.
C22	A16	ADDRESS bus (bit 16) - same as A07 on pin A24.
C23	A15	ADDRESS bus (bit 15) - same as A07 on pin A24.
C24	A14	ADDRESS bus (bit 14) - same as A07 on pin A24.
C25	A13	ADDRESS bus (bit 13) - same as A07 on pin A24.
C26	A12	ADDRESS bus (bit 12) - same as A07 on pin A24.
C27	A11	ADDRESS bus (bit 11) - same as A07 on pin A24.
C28	A10	ADDRESS bus (bit 10) - same as A07 on pin A24.
C29	A09	ADDRESS bus (bit 9) - same as A07 on pin A24.
C30	A08	ADDRESS bus (bit 8) - same as A07 on pin A24.
C31	+12 VDC	+12 Vdc Power - used by the logic circuits on the MPU module.

5.2.2 Terminal Port Connector J1 Interconnect Signals

A standard RS-232C cable mates with connector J1. Refer to Appendix A for detailed information on RS-232C interfacing. Table 5-2 lists each pin connection, signal mnemonic, and signal characteristic for the connector.

The VME120 has only a partial RS-232C implementation. No facility is provided to connect it to a modem. As wired, it provides permanently true Data Set Ready (DSR), Data Carrier Detect (DCD) signals, and a programmable CTS signal. This allows the system to tell the terminal when it is ready to transmit data. The VME120 also can detect a Data Terminal Ready (DTR) signal, that can be controlled by software to clear buffers or send a message, if desired. CTS is controlled by bit 1 of the module control register at F40000, and DTR is sensed by bit 6 of the IPRA in the MK68901 peripheral chip. Refer to Mostek data sheet for details.

TABLE 5-2. Connector J1 Interconnect Signals

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
1		Not used.
2	TxD	TRANSMIT DATA - data to be transmitted is furnished on this line to the MVME120.
3	RxD	RECEIVE DATA - data from the receive line is presented to the terminal.
4		Not used.
5	CTS	CLEAR TO SEND - CTS is a function supplied to the terminal by the MVME120, that indicates that it is permissible to begin transmission of a message.
6	DSR	DATA SET READY - DSR is a function supplied by the MVME120 to the terminal to indicate that the MVME120 is ready to transmit data.
7	SIG-GND	SIGNAL GROUND - common return for all signals.
8	DCD	DATA CARRIER DETECT - furnished by the MVME120 to the terminal to enable its receive circuits.
9-19		Not used.
20	DTR	DATA TERMINAL READY - a signal from the terminal to the MVME120 indicating that the terminal is ready to send or receive data.
21-25		Not used.

5.3 PARTS LIST

Table 5-3 lists the components of the MPU module. The parts locations are illustrated in Figure 5-1. Parts listed are for all versions unless otherwise noted. These parts reflect the latest issue of hardware at the time of printing.

TABLE 5-3. MPU Module Parts List

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
	84-W8293B01	Printed wiring board
CR1	48NW9616A03	Diode, 1N4148/1N914
C1,C6,C26, C28,C67,C71	23NW9618A71	Capacitor, electrolytic, 47 uF @ 10 Vdc
C2,C4,C5, C7-C17,C19-C22,C24,C25, C27,C29-C49, C51-C53,C55-C66,C68	21NW9632A03	Capacitor, fixed, ceramic, 0.1 uF @ 50 Vdc
C23	23NW9704A99	Capacitor, tantalum, 33 uF @ 15 Vdc
DL1	01NW9804C33	Delay module, triple, 40 ns
DL2	01NW9804C12	Delay module, triple, 20 ns
DL3	01NW9804c34	Delay module, triple, 70 ns
DS1,DS2	48NW9612A49	Indicator, LED, red
DS3	48NW9612A59	Indicator, LED, green
J1	28NW9802F88	Connector, 25-pin
J2,J9,J17, J21,J24	28NW9802D04	Header, single row post, 3-pin
J3,J4	28NW9802C63	Header, double row post, 12-pin
J5,J6,J20	28NW9802D01	Header, double row post, 2-pin
J7	28NW9802C36	Header, double row post, 14-pin

TABLE 5-3. MPU Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
J8	28NW9802B34	Header, double row post, 16-pin
J22	28NW9802C43	Header, double row post, 8-pin
J25	28NW9802B21	Header, double row post, 6-pin
P1	28NW9802E51	Connector, 96-pin
R1	51NW9626A37	Resistor network, 9/10k ohm
R2	06SW-124A18	Resistor, fixed, film, 51 ohm, 5%, 1/4 W
R3,R4	06SW-124A23	Resistor, fixed, film, 82 ohm, 5%, 1/4 W
R5	06SW-124A25	Resistor, fixed, film, 100 ohm, 5%, 1/4 W
R6	06SW-124A97	Resistor, fixed, film, 100k ohm, 5%, 1/4 W
R7,R21,R23, R25	51NW9626A49	Resistor network, 7/10k ohm
R8,R29	51NW9626A22	Resistor network, 5/10k ohm
R9	51NW9626A41	Resistor network, 9/4.7k ohm
R10	51NW9626B55	Resistor network, 9/4.7k ohm
R11	51NW9626B52	Resistor network, 5/10k ohm
R12	51NW9626A63	Resistor network, 5/2.2k ohm
R13	51NW9626B56	Resistor network, 9/10k ohm
R14	51NW9626B51	Resistor network, 5/1k ohm
R15	51NW9626B49	Resistor network, 9/15k ohm
R16	06SW-124A17	Resistor, fixed, film, 47 ohm, 5%, 1/4 W
R17	29NW9805B44	Jumper, 2-pin male
R18	51NW9626B50	Resistor network, 3/47 ohm
R19,R20	51NW9626B48	Resistor network, 5/47 ohm
R22	06SW-124A73	Resistor, fixed, film, 10k ohm, 5%, 1/4 W
R24,R26	06SW-124A41	Resistor, fixed, film, 470 ohm, 5%, 1/4 W

TABLE 5-3. MPU Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
R27,R28	06SW-124A83	Resistor, fixed, film, 27k ohm, 5%, 1/4 W
R30	06SW-124A42	Resistor, fixed, film, 510 ohm, 5%, 1/4 W
S1,S2	40NW9801A54	Switch, push, momentary SPDT
S3	40NW9801B29	Switch, piano, 4 position, SPST
U1,U45,U46	51NW9615K66	I.C. 74F32PC
U2,U5,U10, U19	51NW9615G07	I.C. SN74S244N
U3,U8	51NW9615H11	I.C. SN74LS645N
U4,U9	51NW9615G47	I.C. MC6882L
U6	51NW9615F79	I.C. SN74S240N
U7	(NOTE)	I.C. programmed
U11,U18	51NW9615F85	I.C. SN74S38N
U12	51NW9615N76	I.C. SN74LS652NT
U13,U14	51NW9615K98	I.C. 74F280PC
U15	51NW9615G38	I.C. SN74LS38N
U16	(NOTE)	I.C. programmed
U17,U22	51NW9615K18	I.C. 74F373PC
U20	51NW9615E93	I.C. SN74LS14N
U21	(NOTE)	I.C. programmed
U23	51NW9615C30	I.C. SN74LS193N
U24,U29,U32	51NW9615R26	I.C. SN74ALS645-1N
U25,U66,U101	51NW9615K73	I.C. 74F00PC
U26	51NW9615N04	I.C. MC68451R10 (used on 120, 121)
U26	01-W3288B01	MMU bypass (used on 122, 123)
U27	51NW9615N03	I.C. MC68010R10 (used on 120, 121)

TABLE 5-3. MPU Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
U27	51NW9615R11	I.C. MC68010R12 (used on 122, 123)
U28	(NOTE)	I.C. programmed
U30,U33,U36, U43,U64,U72, U76,U77,U81, U82,U87,U93, U99,U100, U106	51NW9615J39	I.C. 74F74PC
U31	(NOTE)	I.C. programmed
U34	(NOTE)	I.C. programmed
U35	51NW9615N40	I.C. MC68901
U37	51NW9615F30	I.C. DM74S05N
U38	51NW9615D93	I.C. SN74S30N
U39	(NOTE)	I.C. programmed
U40	(NOTE)	I.C. programmed
U41,U49	51NW9615H79	I.C. TMM2016P-1 (used on 120, 121, 123)
U42,U83,U95, U107	51NW9615K70	I.C. 74F08PC
U47	51NW9615L74	I.C. 74F163PC
U48,U59	51NW9615N32	I.C. 74F164PC
U50,U51,U53	51NW9615N45	I.C. 74F257PC
U54	51NW9615K65	I.C. 74F64PC
U55,U70	51NW9615K71	I.C. 74F04PC
U56	(NOTE)	I.C. programmed
U57	51NW9615K69	I.C. 74F10PC
U58,U88	51NW9615K72	I.C. 74F92PC

TABLE 5-3. MPU Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
U60,U67,U73, U78,U84,U90, U96,U102	51NW9615N43	I.C. TMS2150-5JL (used on 120, 121, 123)
U61,U62,U68, U69,U74,U75, U79,U80,U85, U86,U91,U92, U97,U98,U103, U104,U109, U110	51NW9615M09	I.C. MCM6665AP15 (used on 120, 122)
U61,U62,U68, U69,U74,U75, U79,U80,U85, U86,U91,U92, U97,U98,U103, U104,U109, U110	51NW9615P38	I.C. HM50256-15 (used on 121, 123)
U63	51NW9615N47	I.C. MC3488AP1
U65	51NW9615F38	I.C. SN74LS393N
U71	51NW9615K67	I.C. 74F20PC
U89	51NW9615K68	I.C. 74F11PC
U94	51NW9615D26	I.C. SN74S113N
U105	51NW9615B30	I.C. MC1489AP
U108	(NOTE)	I.C. programmed
Y1	48AW1068B03	Crystal oscillator, 20.0 MHz (used on 120, 121)
Y1	48AW1015B11	Crystal oscillator, 25.0 MHz (used on 122, 123)
	09NW9811A78	Socket, DIL, 20-pin (use at U7,U16,U21,U28,U31,U40,U56,U108)
	09NW9811A71	Socket, PGA, 68-pin (use at U26,U27)

TABLE 5-3. MPU Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
	09NW9811B01	Socket, DIL, 24-pin (use at U34,U39)
	09-W4659B24	Socket, 24-pin (use at U35)
	09-W4659B12	Socket, 12-pin (use at U41,U49)
	09-W4659B14	Socket, 14-pin (use at UX44,UX52)
	09NW9811A46	Socket, crystal, 4-lead (use at Y1)
	55NW9501A12	Handle
	64-W4736B01	Panel, front
	29NW9805B17	Jumper, shorting, insulated (use at J2-J9, J17,J20,J24,J25)

NOTE: When ordering, use number labeled on part.

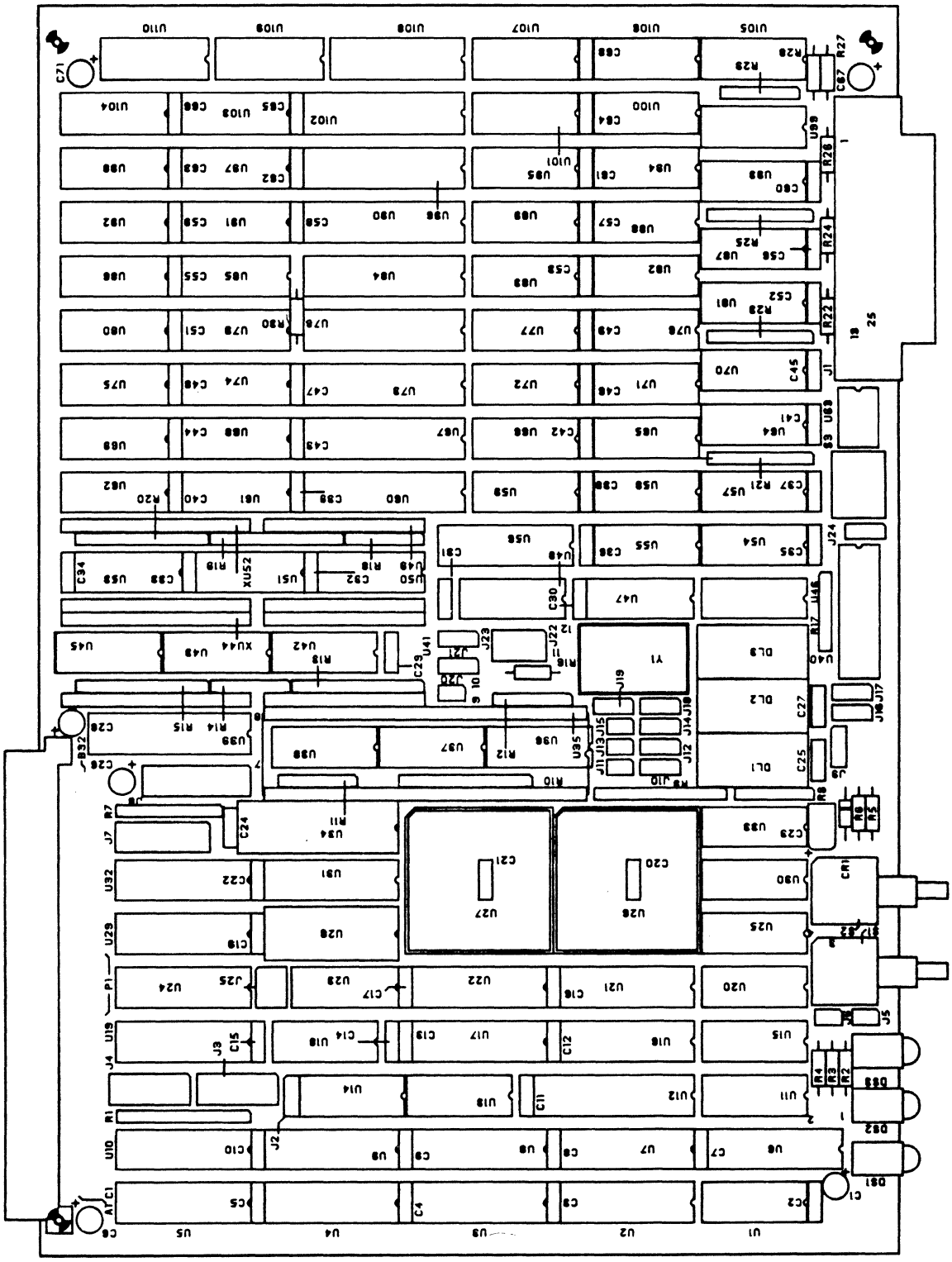


FIGURE 5-1. MPU Module Parts Location

5.4 SCHEMATIC DIAGRAMS

FIGURE 5-2 illustrates the schematic diagram for the MPU module.

NOTES:

1. FOR REFERENCE DRAWINGS REFER TO BILL(S) OF MATERIAL 01-W3293B01
2. UNLESS OTHERWISE SPECIFIED:
   ALL RESISTORS ARE IN OHMS, ±5PCT, 1/4 WATT.
   ALL CAPACITORS ARE IN UF.
   ALL VOLTAGES ARE DC.
3. INTERRUPTED LINES CODED WITH THE SAME LETTER OR LETTER COMBINATIONS ARE ELECTRICALLY CONNECTED.
4. DEVICE TYPE NUMBER IS FOR REFERENCE ONLY. THE NUMBER VARIES WITH THE MANUFACTURER.
5. SPECIAL SYMBOL USAGE:
   ‹ DENOTES – ACTIVE LOW SIGNAL.
   [] DENOTES – ON BOARD SIGNAL.
6. INTERPRET DIAGRAM IN ACCORDANCE WITH AMERICAN NATIONAL STANDARDS INSTITUTE SPECIFICATIONS, CURRENT REVISION.
7. PART TYPES ARE ABBREVIATED IN THE FIELD OF THE DRAWING, FOR FULL PART TYPE, REFER TO TABLE 1.
8. CODE FOR SHEET TO SHEET REFERENCES IS AS FOLLOWS:

   SHEET — 6 A6 — ZONE

12	
XU52	XU1-43,45-51
Y1	
U110	U44,52
S3	
R30	
P1	
J25	
DL3	
CR1	
C71	C18,50 54,69,70
HIGHEST NUMBER USED	NOT USED
REFERENCE	DESIGNATIONS

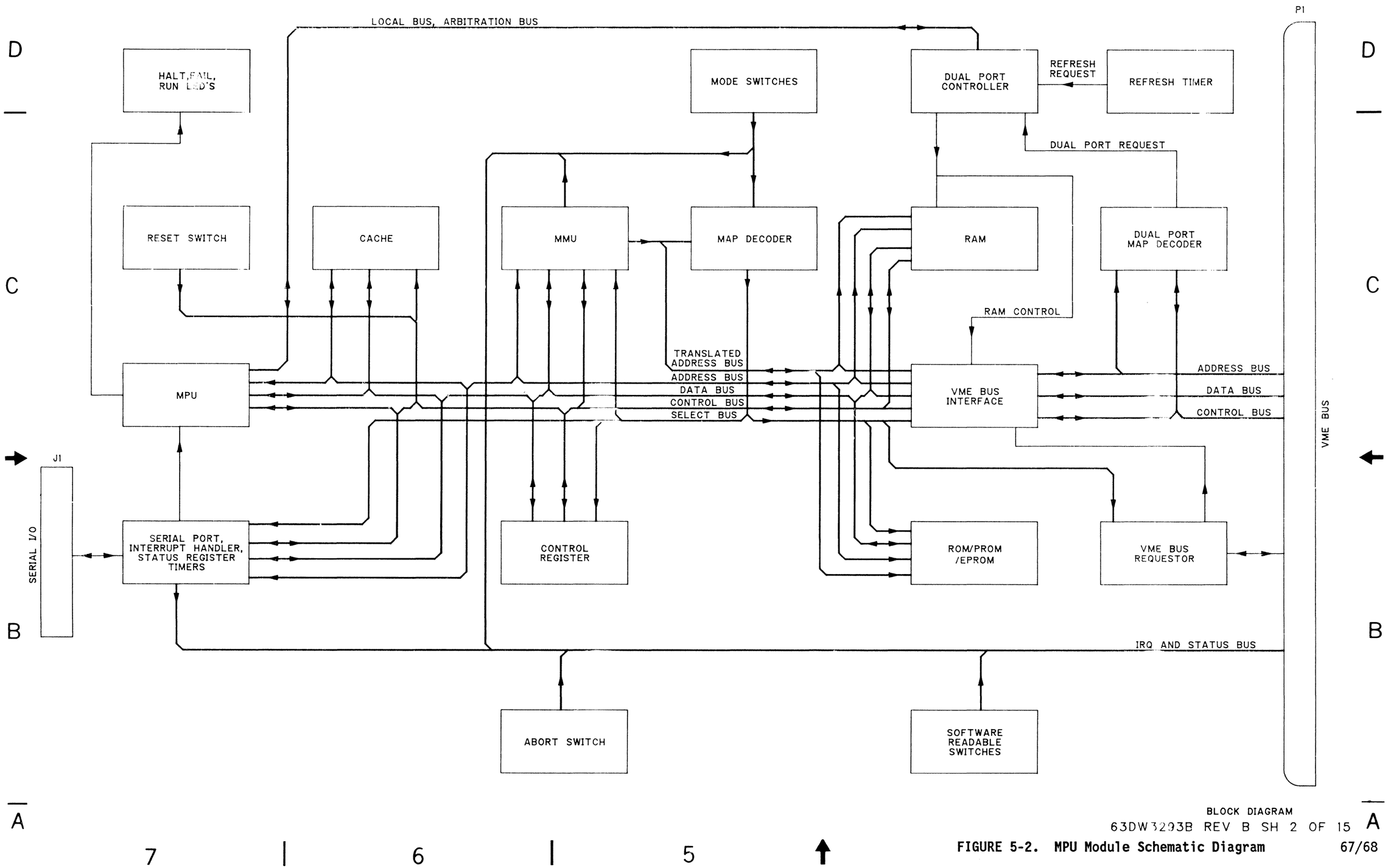
7 TABLE 1

REF DES	TYPE 4	GND	+5V	SH
DL1	40NS	7	14	6,12,13
DL2	20NS	7	14	6,8,11
DL3	70NS	7	14	12,13
U1	74F32	7	14	3,4,7,8
U2	74S244	10	20	3,4,12
U3	74LS645	10	20	7
U4	MC6882B	10	20	12
U5	74S244	10	20	12
U6	74S240	10	20	3,8,12,13
U7	PAL16L8	10	20	12
U8	74LS645	10	20	7
U9	MC6882B	10	20	12
U10	74S244	10	20	12
U11	74S38	7	14	4,7
U12	74LS652	12	24	7
U13	74F280	7	14	11
U14	74F280	7	14	11
U15	74LS38	7	14	3,4,11
U16	PAL16L8A	10	20	8
U17	74F373	10	20	7
U18	74S38	7	14	9,11,13
U19	74S244	10	20	11
U20	74LS14	7	14	4,6,9,11
U21	82S153A	10	20	7
U22	74F373	10	20	7
U23	74LS193	8	16	8
U24	74LS645-1	10	20	12
U25	74F00	7	14	4,6,11
U26	MC68451	A10	H8	7
U26	MC68451	J2	C3	7
U26	MC68451	H3	H9	7
U26	MC68451	C8		7
U27	MC68010	E2	E9	4
U27	MC68010	D9	D2	4
U28	PAL16L2	10	20	8
U29	74LS645-1	10	20	12
U30	74F74	7	14	4
U31	PAL16L8B	10	20	9
U32	74LS645-1	10	20	12
U33	74F74	7	14	12,15
U34	PAL20L8A	12	24	7
U35	MK68901	36	11	15
U36	74F74	7	14	9
U37	74S05	7	14	4,8,11,12
U38	74S30	7	14	11
U39	PAL20L9A	12	24	15
U40	PAL16L8B	10	20	6
U41	MCM65116	12	24	6
U42	74F08	7	14	4,7,11,12
U43	74F74	7	14	4,9
U45	74F32	7	14	3,9,12
U46	74F32	7	14	4,11
U47	74F163	8	16	4
U48	74F164	7	14	12
U49	MCM65116	12	24	6
U50	74F257	8	16	10
U51	74F257	8	16	10
U53	74F257	8	16	10

7 TABLE 1 CON'T

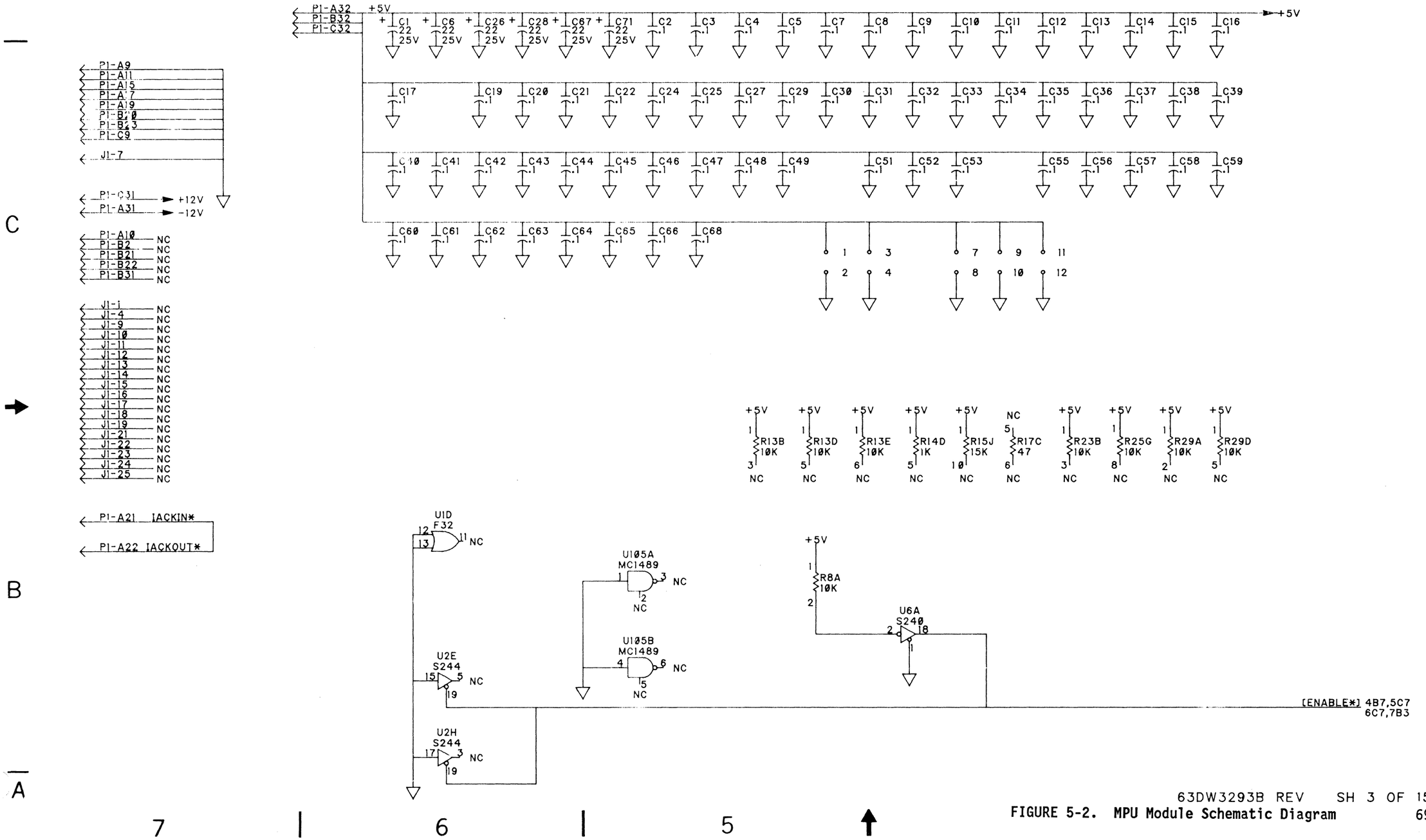
REF DES	TYPE 4	GND	+5V	+12V	-12V	SH
U54	74F64	7	14			6
U55	74F04	7	14			4,7,9,12,13
U56	PAL16R8	10	20			10
U57	74F10	7	14			8,9,12
U58	74F02	7	14			7,8,11
U59	74F164	7	14			6
U60	TMS2150	12	24			5
U61	MCM3256	16	8			10
U62	MCM6256	16	8			10
U63	MC3488	4		8	5	15
U64	74F74	7	14			11
U65	74LS393	7	14			8
U66	74F00	7	14			9,12,13,15
U67	TMS2150	12	24			5
U68	MCM6256	16	8			10
U69	MCM6256	16	8			10
U70	74F04	7	14			8,9,13
U71	74F20	7	14			8,9
U72	74F74	7	14			6,9
U73	TMS2150	12	24			5
U74	MCM6256	16	8			10
U75	MCM6256	16	8			10
U76	74F74	7	14			13
U77	74F74	7	14			13
U78	TMS2150	12	24			5
U79	MCM6256	16	8			10
U80	MCM6256	16	8			10
U81	74F74	7	14			9
U82	74F74	7	14			13
U83	74F08	7	14			9,13
U84	TMS2150	12	24			5
U85	MCM6256	16	8			10
U86	MCM6256	16	8			10
U87	74F74	7	14			9
U88	74F02	7	14			8,13
U89	74F11	7	14			8,9,13
U90	TMS2150	12	24			5
U91	MCM6256	16	8			10
U92	MCM6256	16	8			10
U93	74F74	7	14			8,9
U94	74S113	7	14			8
U95	74F08	7	14			8,13
U96	TMS2150	12	24			5
U97	MCM6256	16	8			10
U98	MCM6256	16	8			10
U99	74F74	7	14			8,9
U100	74F74	7	14			8
U101	74F00	7	14			3,8,9
U102	TMS2150	12	24			5
U103	MCM6256	16	8			10
U104	MCM6256	16	8			10
U105	MC1489	7	14			15
U106	74F74	7	14			8
U107	74F08	7	14			8,9
U108	PAL16L8A	10	20			6
U109	MCM6256	16	8			10
U110	MCM6256	16	8			10
XU44	SOCKET					14
XU52	SOCKET					14
Y1	K1100	7	14			4

FIGURE 5-2. MPU Module Schematic Diagram



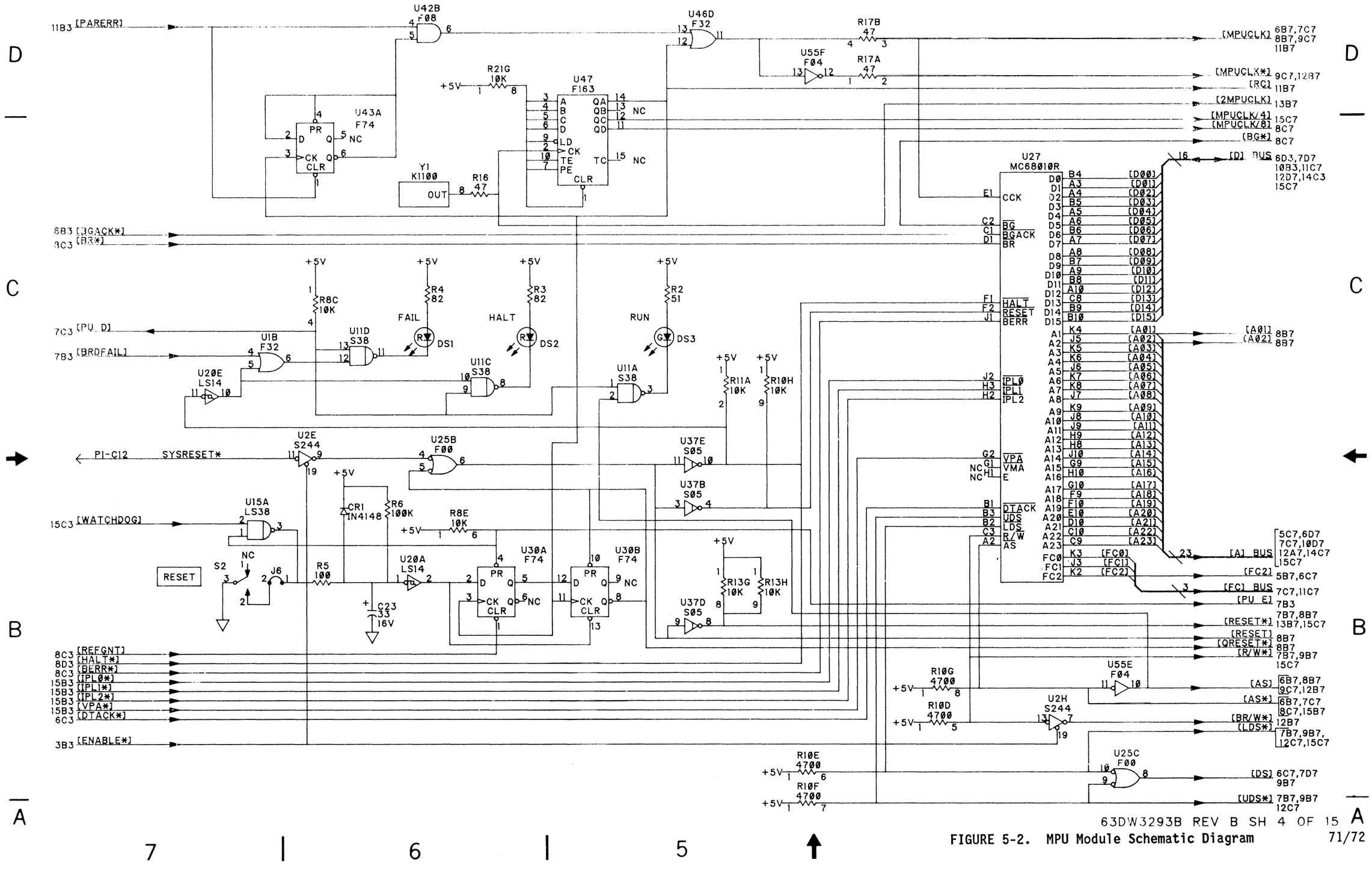
BLOCK DIAGRAM
63DW3293B REV B SH 2 OF 15

FIGURE 5-2. MPU Module Schematic Diagram



63DW3293B REV SH 3 OF 15

FIGURE 5-2. MPU Module Schematic Diagram



63DW3293B REV B SH 4 OF 15

FIGURE 5-2. MPU Module Schematic Diagram

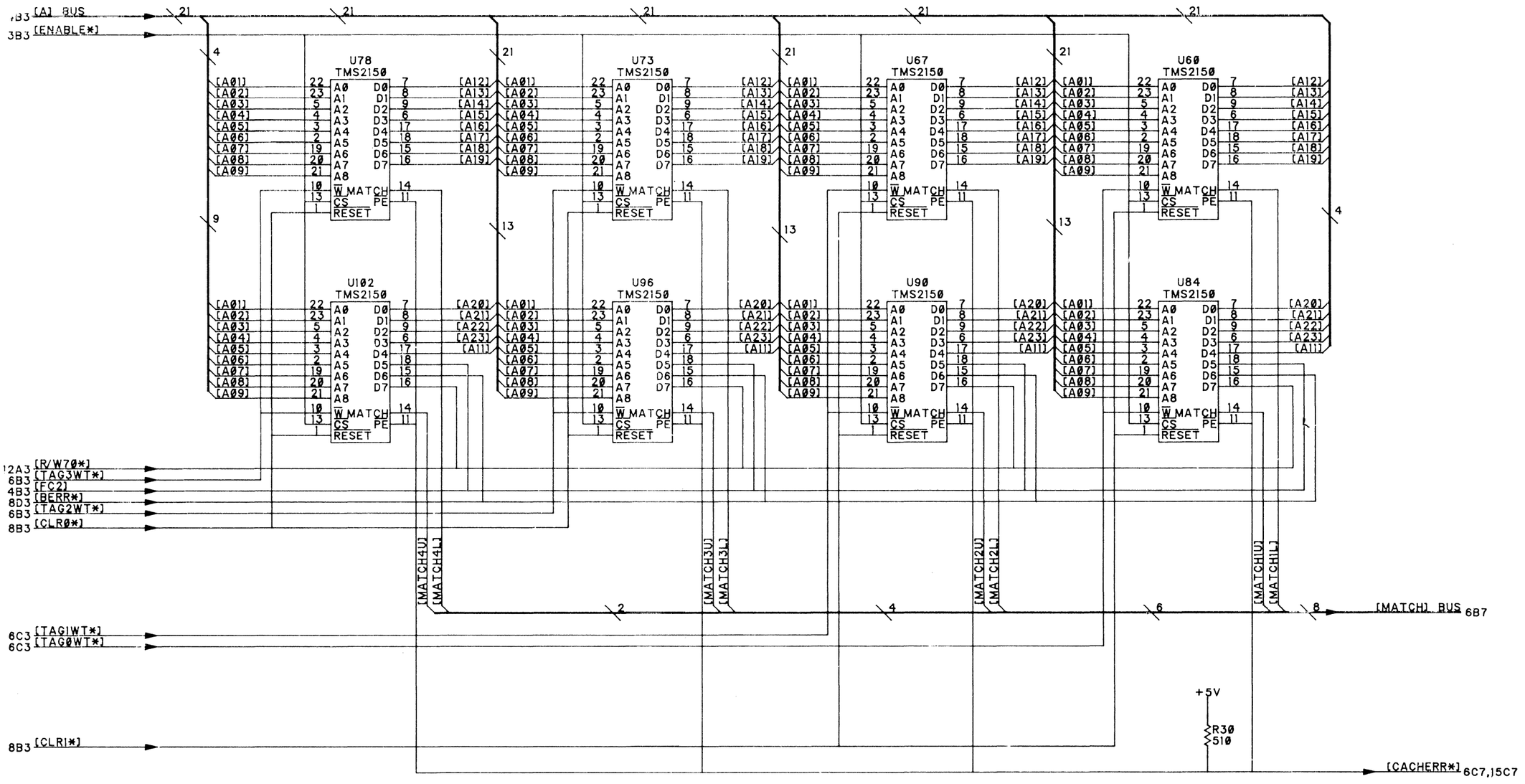


FIGURE 5-2. MPU Module Schematic Diagram

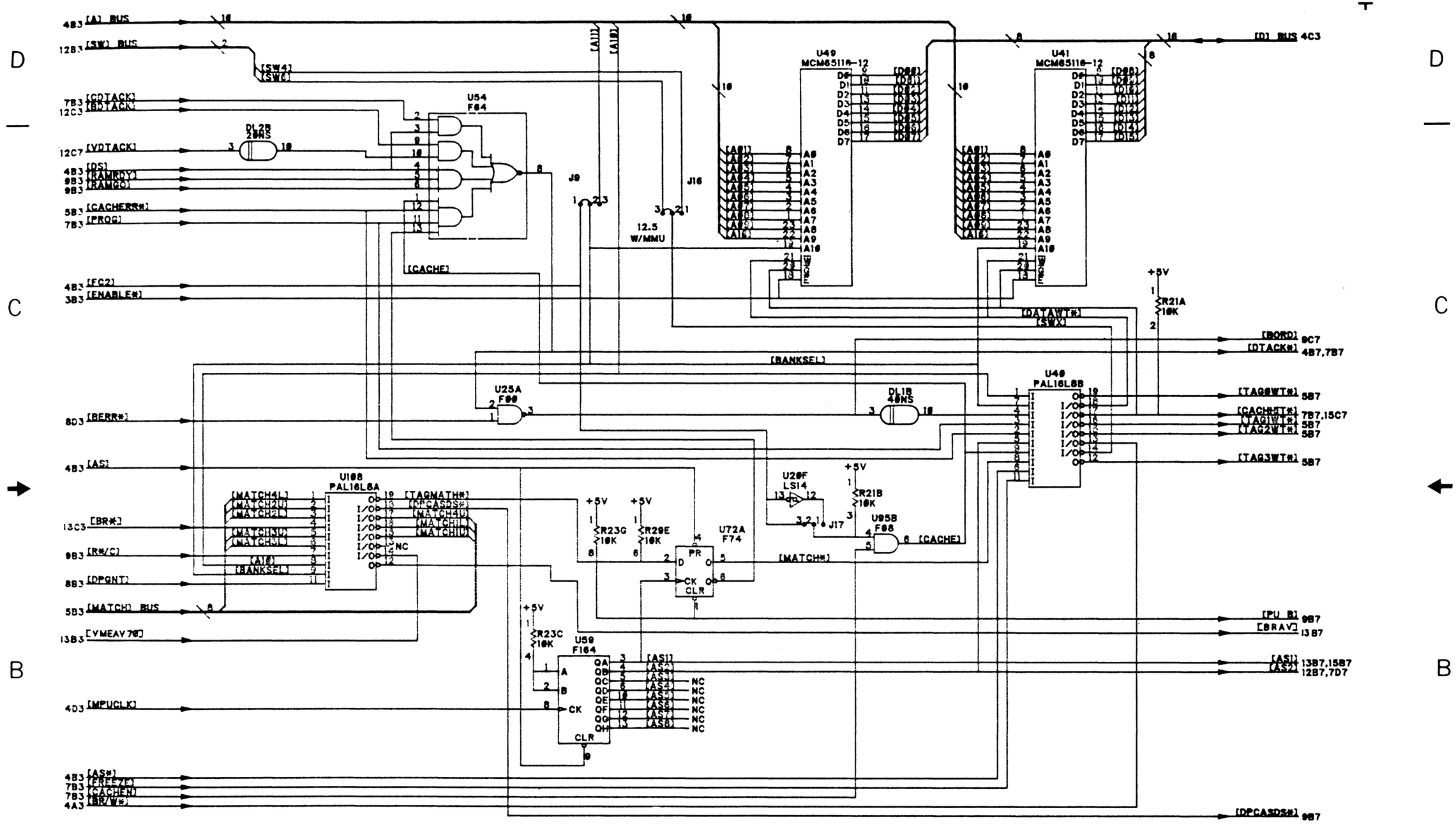
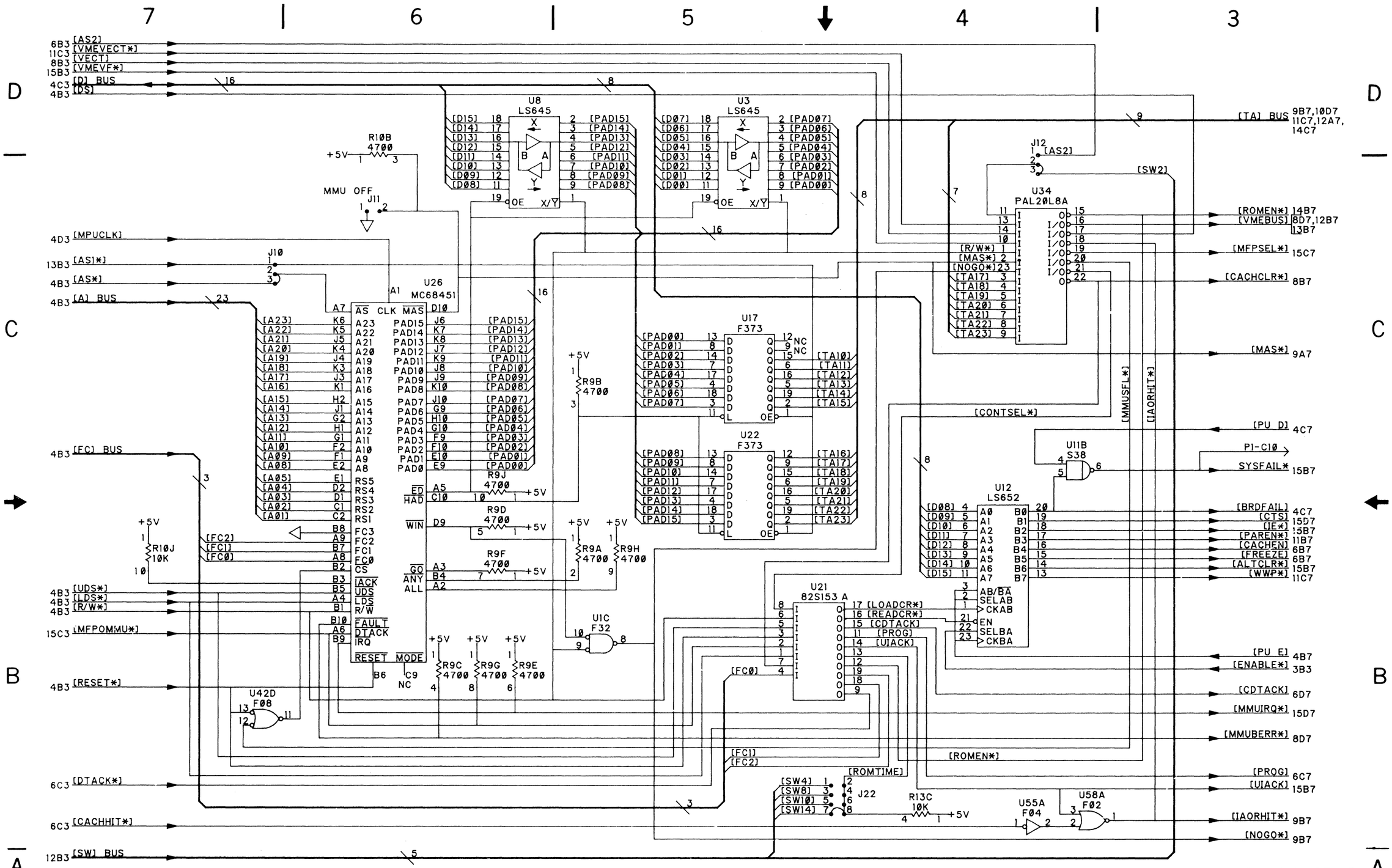


FIGURE 5-2. MPU Module Schematic Diagram



63DW3293B REV B SH 7 OF 15

FIGURE 5-2. MPU Module Schematic Diagram 77/78

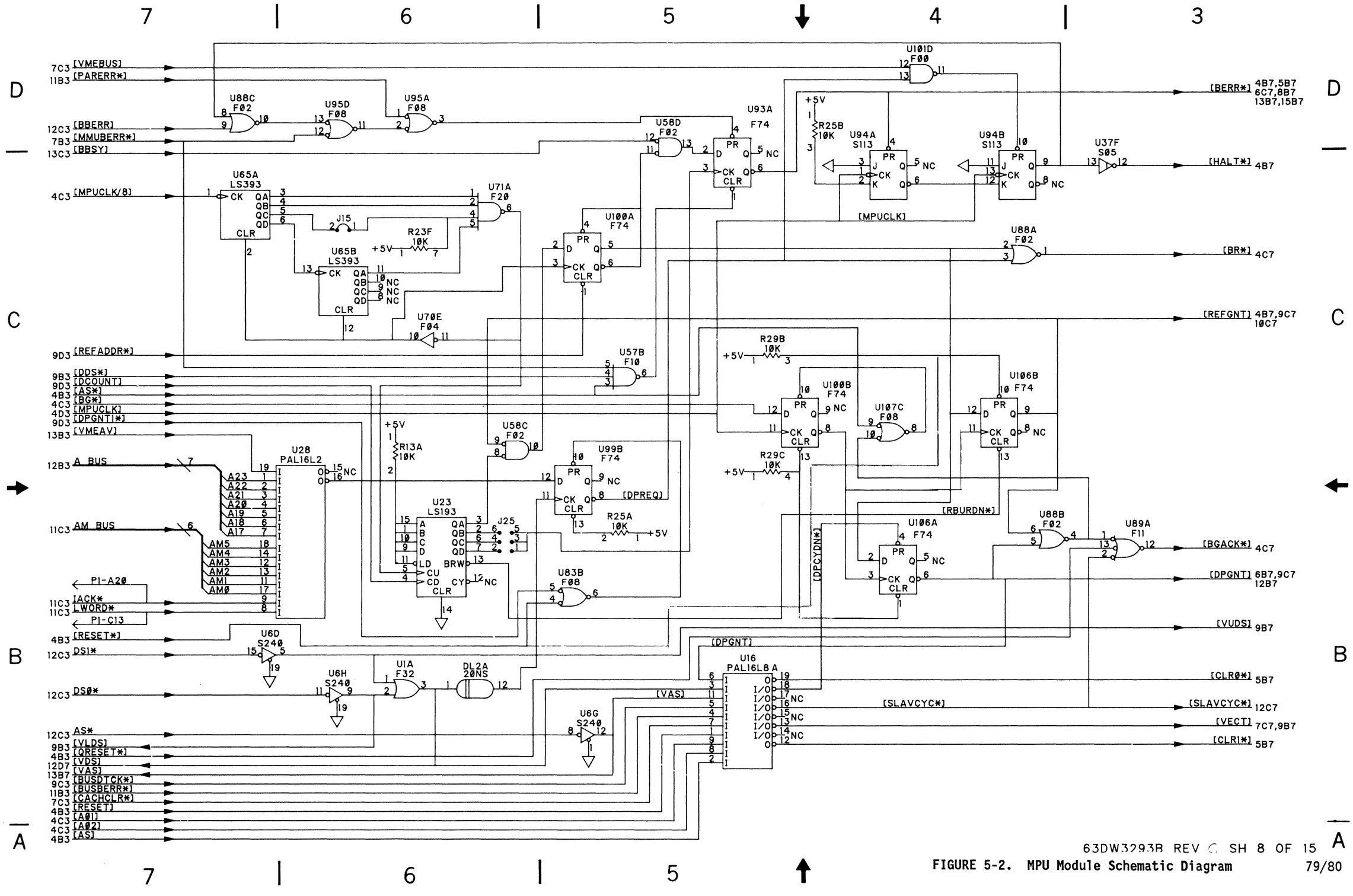
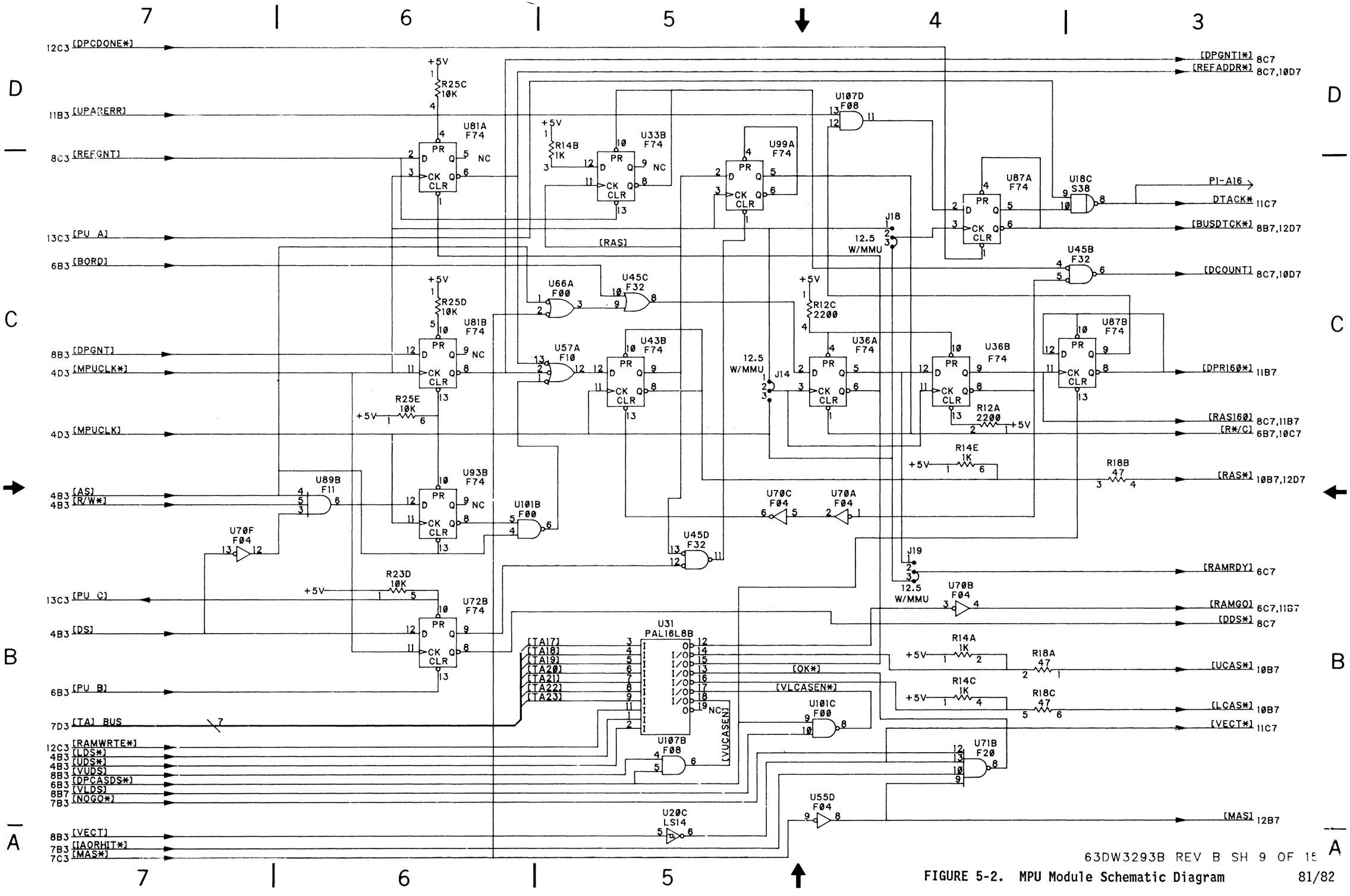
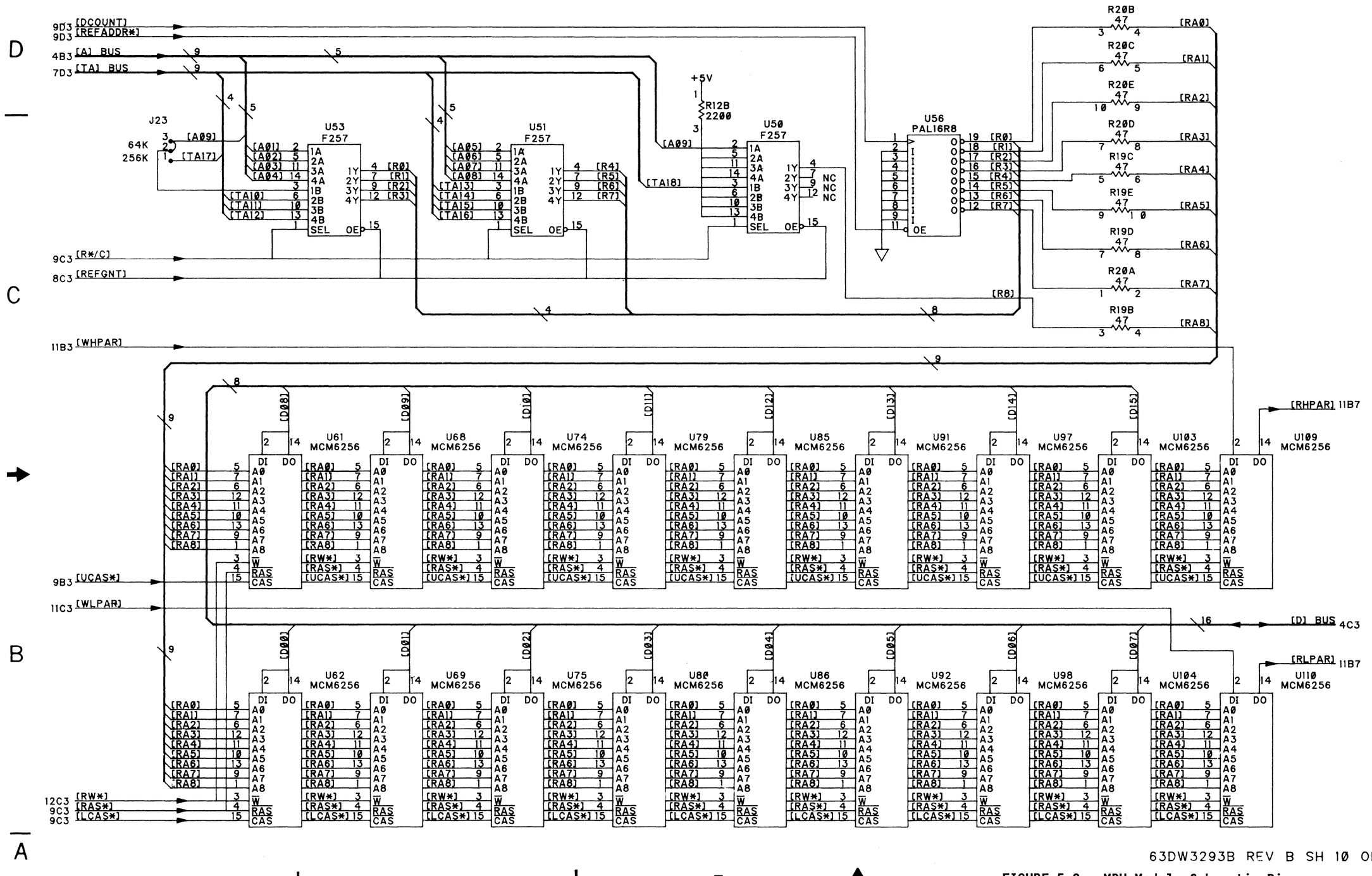


FIGURE 5-2. MPU Module Schematic Diagram 79/80



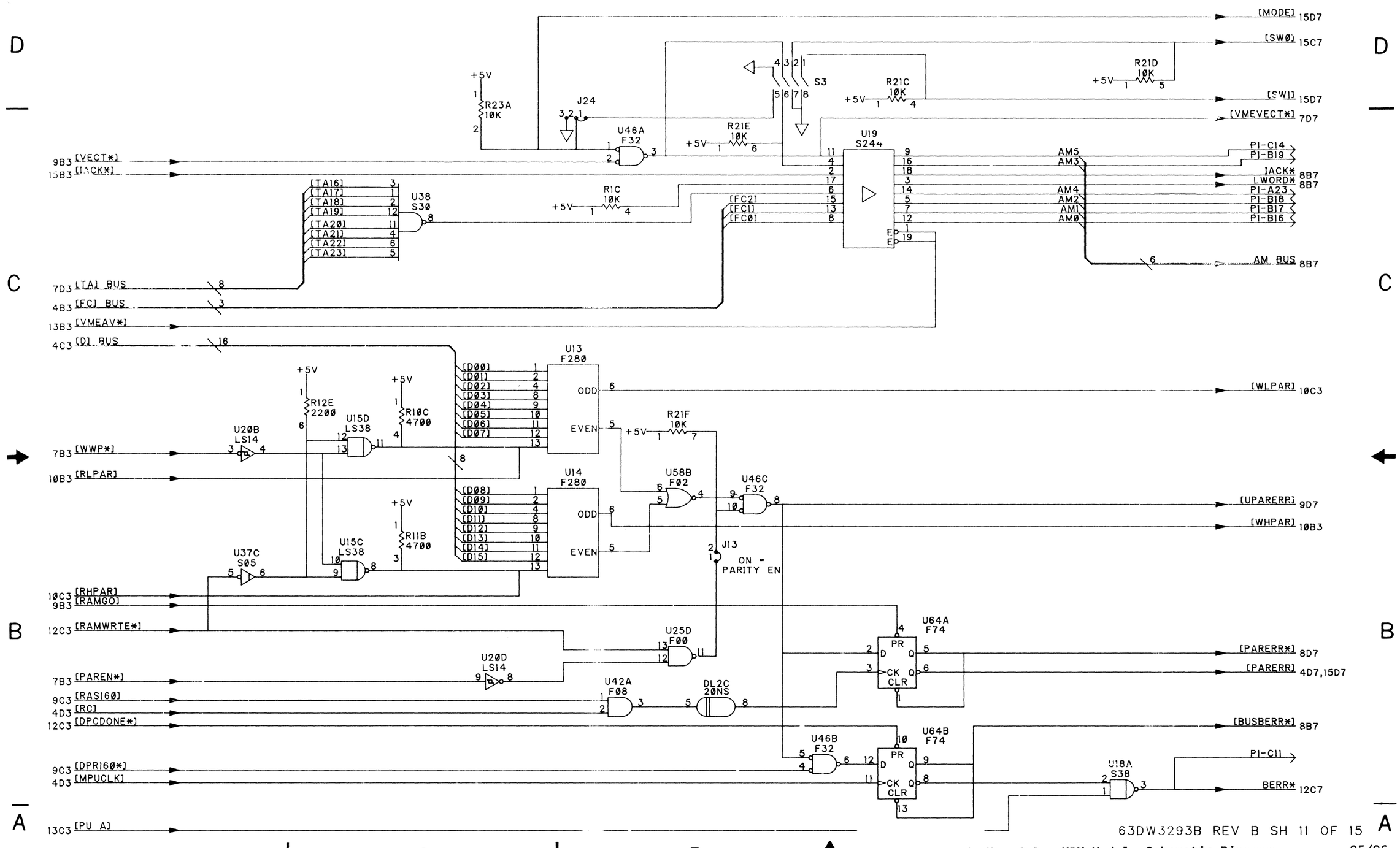
63DW3293B REV B SH 9 OF 15

FIGURE 5-2. MPU Module Schematic Diagram 81/82



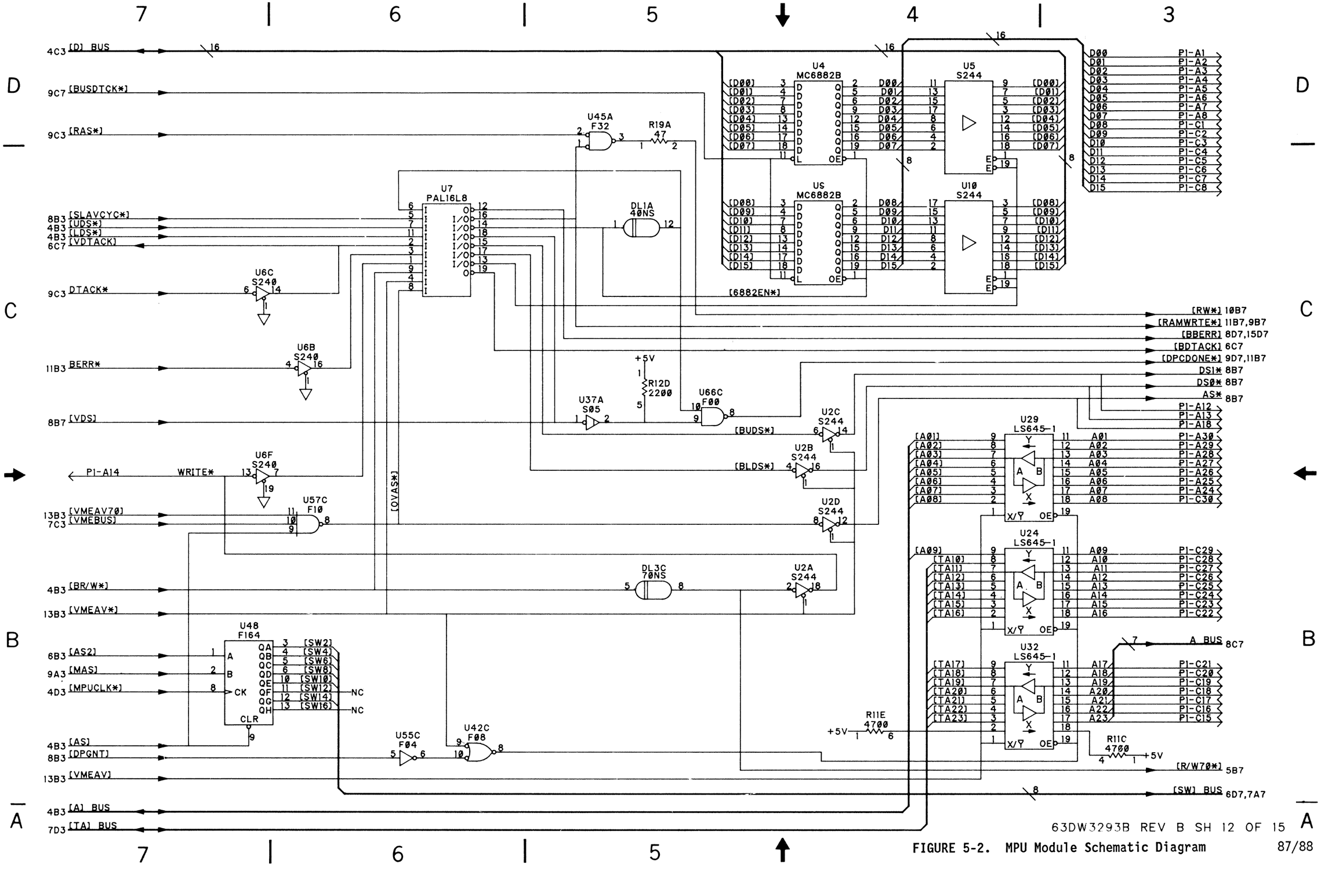
63DW3293B REV B SH 10 OF 15

FIGURE 5-2. MPU Module Schematic Diagram



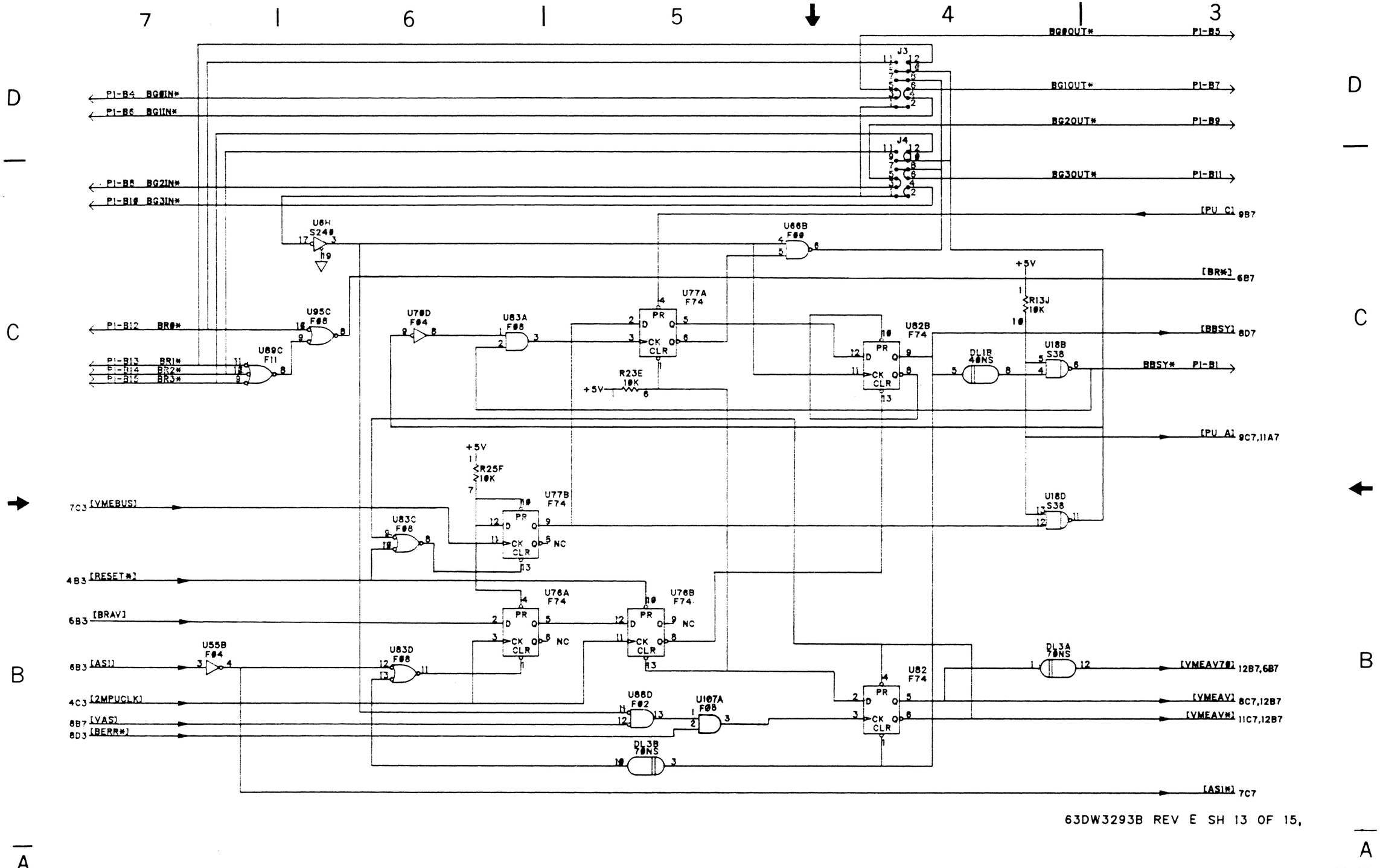
63DW3293B REV B SH 11 OF 15

FIGURE 5-2. MPU Module Schematic Diagram



63DW3293B REV B SH 12 OF 15

FIGURE 5-2. MPU Module Schematic Diagram



63DW3293B REV E SH 13 OF 15,

FIGURE 5-2. MPU Module Schematic Diagram

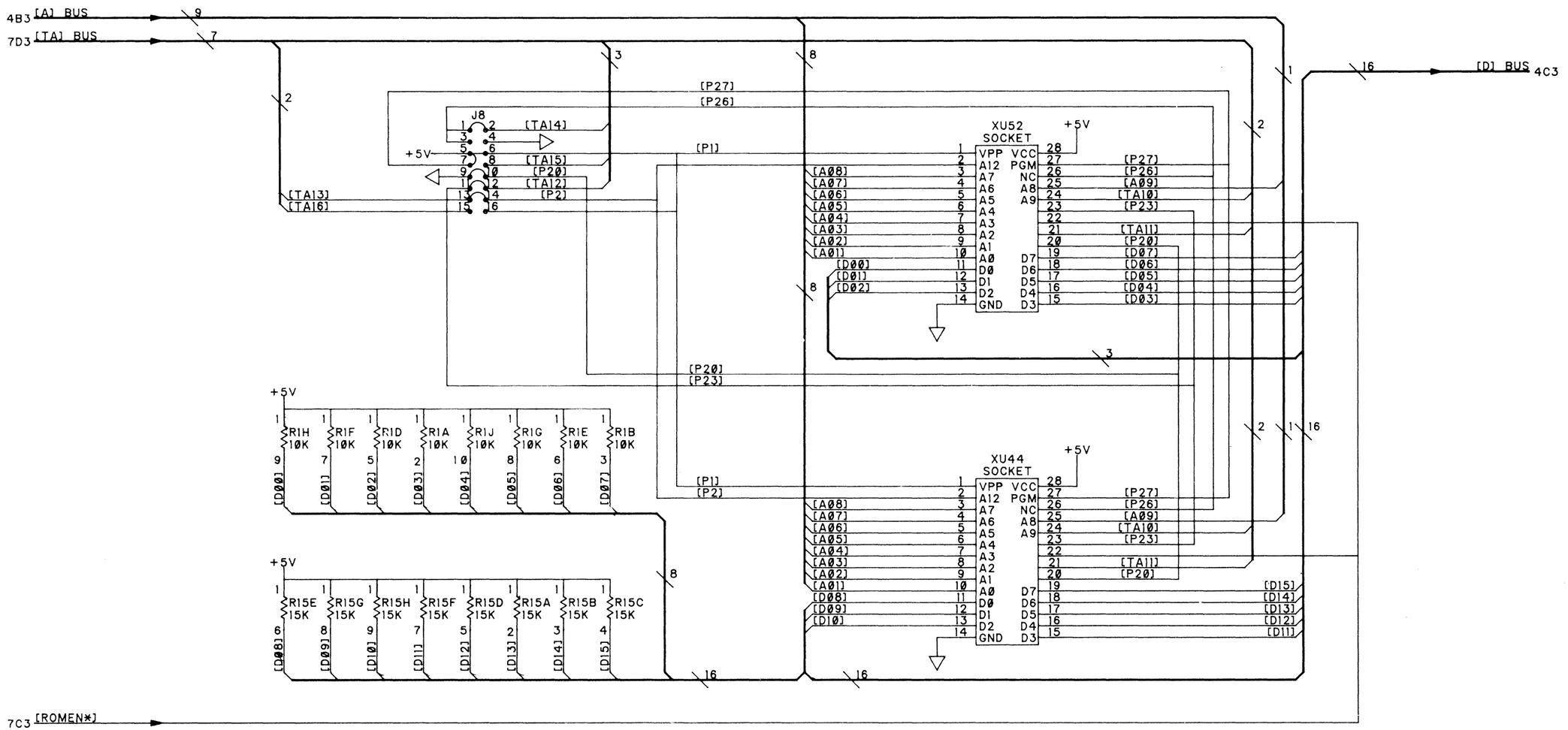
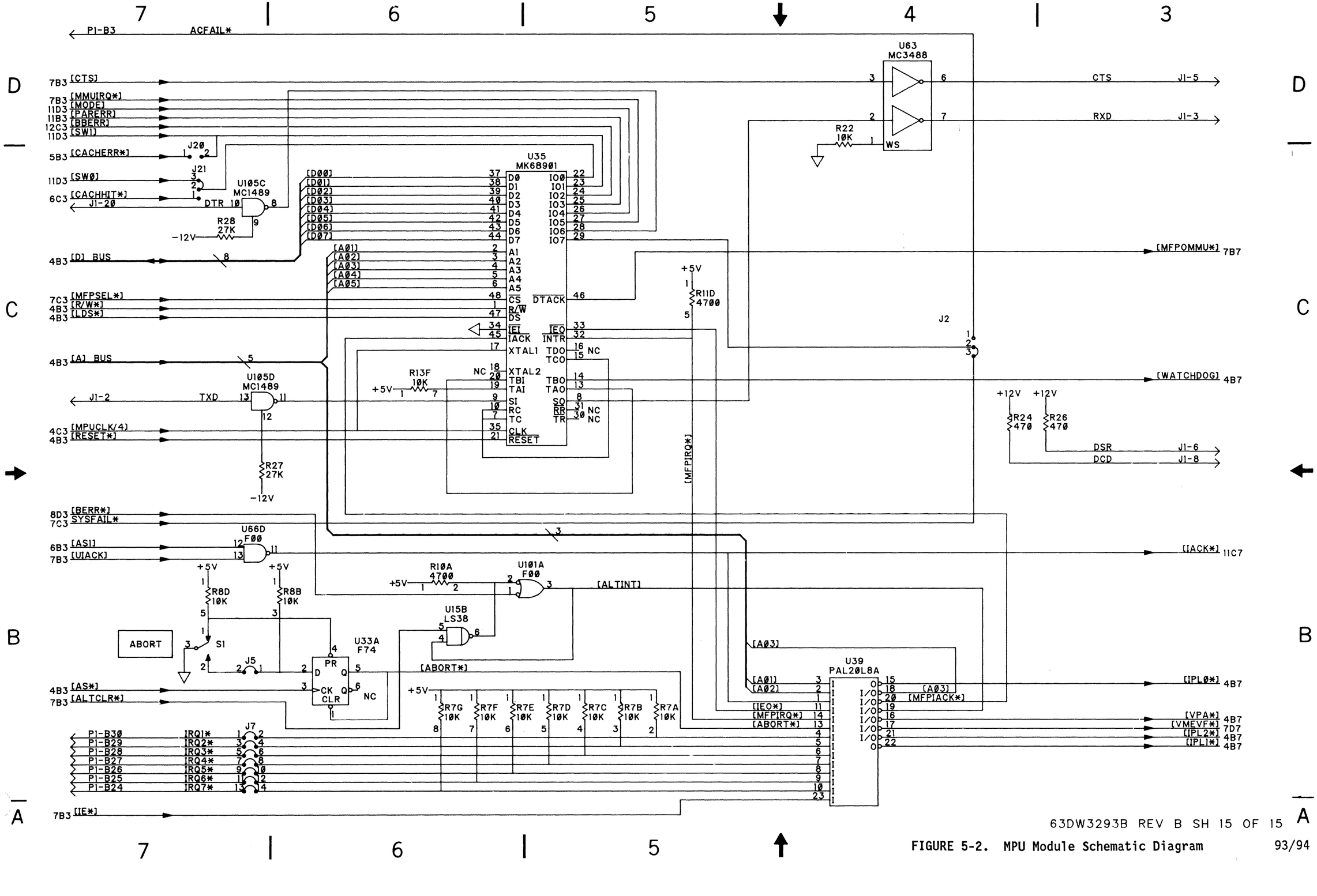


FIGURE 5-2. MPU Module Schematic Diagram



63DW3293B REV B SH 15 OF 15

FIGURE 5-2. MPU Module Schematic Diagram 93/94

A

APPENDIX A

RS-232C INTERCONNECTIONS

The RS-232C standard is the most widely used interface between terminals and computers or modems, and yet it is not fully understood. This is because all the lines are not clearly defined, and many users do not see the need to conform for their applications. A system should easily connect to any other. Many times designers think only of their own equipment, but the state-of-the-art is computer-to-computer or computer-to-modem operation.

The RS-232C standard was originally developed by the Bell System to connect terminals via modems. Therefore, several handshaking lines were included. In many applications these are not needed, but since they permit diagnosis of problems, they are included in many applications.

Table 1 lists the standard RS-232C interconnections. In order to interpret this information correctly it is necessary to know that RS-232C is intended to connect a terminal to a modem. When computers are connected to computers without modems, one must be configured as a terminal and the other as a modem. Because computers are normally configured to work with terminals, they are said to be configured as a modem. Also, the signal levels must be between +3 and +15 volts for a high level, and between -3 and -15 volts for a low level. Any attempt to connect units in parallel may result in out of range voltages and is not allowed by the RS-232C specification.

TABLE 1. RS-232C Interconnections

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
1		Not used.
2	TxD	TRANSMIT DATA - Data to be transmitted is furnished on this line to the modem from the terminal.
3	RxD	RECEIVE DATA - Data that is demodulated from the receive line is presented to the terminal by the modem.
4	RTS	REQUEST TO SEND - RTS is supplied by the terminal to the modem when required to transmit a message. With RTS off, the modem carrier remains off. When RTS is turned on, the modem immediately turns on the carrier.

A

TABLE 1. RS-232C Interconnections (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
5	CTS	CLEAR TO SEND - CTS is a function supplied to the terminal by the modem that indicates that it is permissible to begin transmission of a message. When using a modem, CTS follows the off-to-on transition of RTS after a time delay.
6	DSR	DATA SET READY - DSR is a function supplied by the modem to the terminal to indicate that the modem is ready to transmit data.
7	SIG-GND	SIGNAL GROUND - Common return line for all signals at the modem interface.
8	DCD	DATA CARRIER DETECT - Sent by the modem to the terminal to indicate that a valid carrier is being received.
9-14		Not used.
15	TxC	TRANSMIT CLOCK - This line clocks output data to the modem from the terminal.
16		Not used.
17	RxC	RECEIVE CLOCK - This line clocks input data from a terminal to a modem.
18,19		Not used.
20	DTR	DATA TERMINAL READY - A signal from the terminal to the modem indicating that the terminal is ready to send or receive data.
21		Not used.
22	RI	RING INDICATOR - RI is sent by the modem to the terminal. This line indicates to the terminal that an incoming call is present. The terminal causes the modem to answer the phone by carrying DTR true while RI.
23		Not used.
24	TxC	TRANSMIT CLOCK - Same as TxC on pin 15.

A

TABLE 1. RS-232C Interconnections (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
25	BSY	BUSY - A positive EIA signal applied to this pin causes the modem to go off-hook and make the associated phone busy.

NOTES: 1. High level = +3 to +15 volts. Low level = -3 to -15 volts.

2. RS-232C is intended to connect a terminal to a modem. When computers are connected to computers without modems, one of the computers must be configured as a modem and the other as a terminal.

There are several levels of conformance that are appropriate for typical RS-232C interconnections. The bare minimum requirement is the two data lines and a ground. The full version of RS-232C requires 12 lines and accommodates automatic dialing, automatic answering, and synchronous transmission. A middle-of-the-road approach is illustrated in Figure 1.

One set of handshaking signals frequently implemented are RTS and CTS. CTS is used in many systems to inhibit transmission until the signal is high. In the modem application, RTS is turned around and returned as CTS after 150 microseconds. RTS is programmable in some systems to work with the older type 202 modem (half duplex). CTS is used in some systems to provide flow control to avoid buffer overflow. This is not possible if modems are used. It is usually necessary to make CTS high by connecting it to RTS or to some source of +12 volts such as the resistors shown in Figure 1. It is also frequently jumpered to an MC1488 gate that has its inputs grounded (the gate is provided for this purpose). Another signal used in many systems is DCD. The original purpose of this signal was to tell the system that the carrier tone from the distant modem was being received. This signal is frequently used by the software to display a message like CARRIER NOT PRESENT to help the user to diagnose failure to communicate. Obviously, if the system is designed properly to use this signal, and it is not connected to a modem, the signal must be provided by a pullup resistor or gate as described before (see Figure 1). Many modems expect a DTR high signal and issue a DSR. These signals are used by software to help prompt the operator as to possible causes of trouble. The DTR signal is used sometimes to disconnect the phone circuit in preparation for another automatic call. It is necessary to provide these signals to talk to all possible modems (see Figure 1). As shown, Figure 1 is a good minimum configuration that almost always works. If the CTS and DCD signals are not received from the modem, the jumpers can be moved to provide the needed signal artificially. Figure 2 shows a way that an RS-232C connector can be wired to enable a computer to connect to a basic terminal with only three wires. This is because most terminals have a DTR signal that

is ON, and that can be used to pullup the CTS, DCD, and DSR signals. Two of these connectors wired back-to-back can be used. It must be realized that all the handshaking has been bypassed and possible diagnostic messages do not occur. Also the Tx and Rx lines may have to be crossed since Tx from a terminal is outgoing but the Tx line on a modem is an incoming signal.

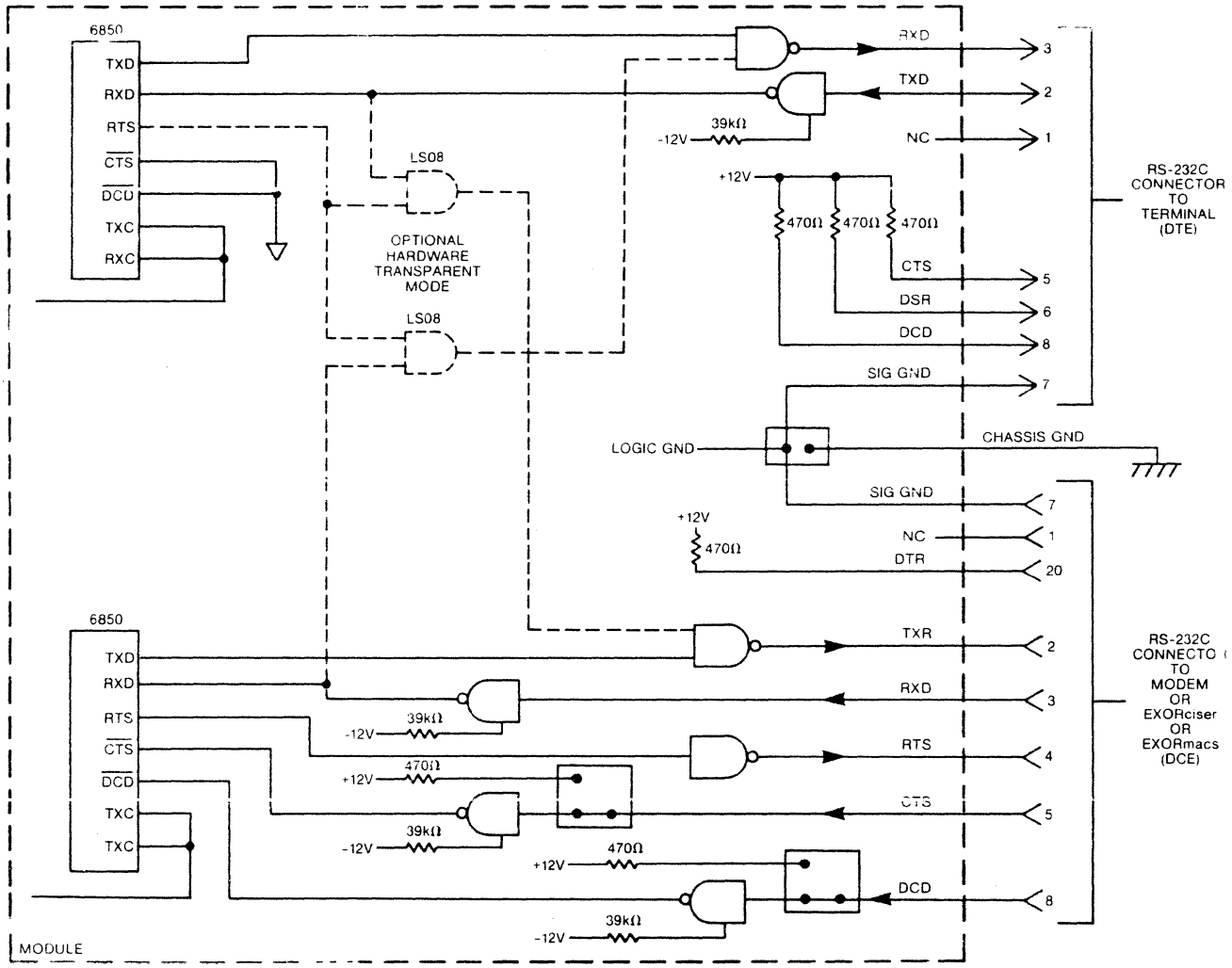


FIGURE 1. Middle-of-the-Road RS-232C Configuration

Another subject that needs to be considered is the use of ground pins. There are two pins labeled GND. Pin 7 is the SIGNAL GROUND and must be connected to the distant device to complete the circuit. Pin 1 is the CHASSIS GROUND, but it must be used with care. The chassis is connected to the power ground through the green wire in the power cord and must be connected to the chassis to be in compliance with the electrical code. The problem is that when units are connected to different electrical outlets, there may be several volts

difference in ground potential. If pin 1 of the devices are interconnected with a cable, several amperes of current could result. This not only may be dangerous for the small wires in a typical cable, but could result in electrical noise that could cause errors. That is the reason that Figure 1 shows no connection for pin 1. Normally, pin 7 should only be connected to the CHASSIS GROUND at one point, and if several terminals are used with one computer, the logical place for that point is at the computer. The terminals should not have a connection between the logic ground return and the chassis.

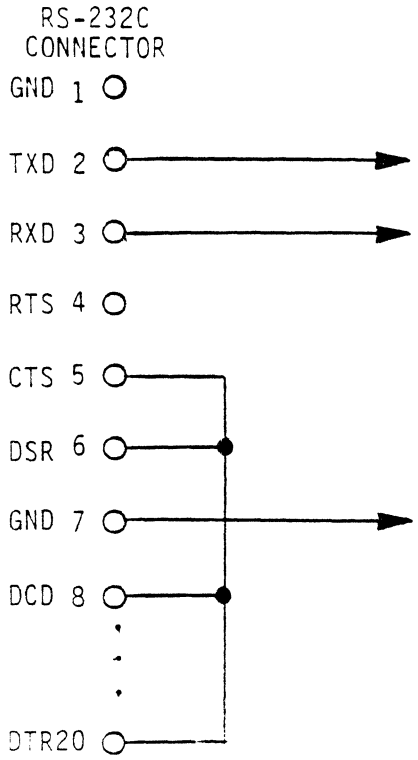


FIGURE 2. Minimum RS-232C Connection

THIS PAGE INTENTIONALLY LEFT BLANK.

APPENDIX B

PROGRAMMABLE ARRAY LOGIC

B

Programmable Array Logic (PAL) source code for the various devices on the MVME120 is listed in the following pages. In the upper left corner of each page is the PAL number (PAL16L8B), the device reference designation (U108 E), and the schematic sheet number (SHEET 6). The pages are arranged in sheet number order.

```
PAL16L8A                                                        5-31-84
U40-SHEET 6          MVME120/MVME121                            CKSM= 4566
PALCAC20                                                        REV A

A10  /CACHERR  PROG  BEDT40 AS2 /AS       BANKSEL /MATCH   CACHE   GND
FREEZE /TAG3WT /WRITE SWX /TAG2WT /TAG1WT /CACHHIT /DATAWT /TAG0WT VCC

IF (VCC) CACHHIT = AS*PROG*MATCH*/CACHERR*CACHE

IF (VCC) DATAWT  = AS*/BEDT40*PROG*/FREEZE*/MATCH *AS2*CACHE
                 + AS*/BEDT40*PROG*/FREEZE*CACHERR*AS2*CACHE
                 + AS*/BEDT40*MATCH*AS2*CACHE*WRITE
                 +    /BEDT40*DATAWT
                 + /SWX*AS*DATAWT

IF (VCC) TAG0WT  = DATAWT*AS*/A10*/BANKSEL*/SWX
                 + DATAWT*AS*/A10*/BANKSEL*/BEDT40

IF (VCC) TAG1WT  = DATAWT*AS*A10*/BANKSEL*/SWX
                 + DATAWT*AS*A10*/BANKSEL*/BEDT40

IF (VCC) TAG2WT  = DATAWT*AS*/A10*BANKSEL*/SWX
                 + DATAWT*AS*/A10*BANKSEL*/BEDT40

IF (VCC) TAG3WT  = DATAWT*AS*A10*BANKSEL*/SWX
                 + DATAWT*AS*A10*BANKSEL*/BEDT40

DESCRIPTION:  CACHE CONTROLLER PAL
```

B

```
PAL16L8B                                                        5-31-84
U40-SHEET 6          MVME123                                    CKSM= 4566
PALCAC23                                                        REV A

A10  /CACHERR  PROG  BEDT40 AS2 /AS      BANKSEL /MATCH   CACHE   GND
FREEZE /TAG3WT /WRITE SWX /TAG2WT /TAG1WT /CACHHIT /DATAWT /TAG0WT VCC

IF (VCC) CACHHIT = AS*PROG*MATCH*/CACHERR*CACHE

IF (VCC) DATAWT  = AS*/BEDT40*PROG*/FREEZE*/MATCH *AS2*CACHE
                 + AS*/BEDT40*PROG*/FREEZE*CACHERR*AS2*CACHE
                 + AS*/BEDT40*MATCH*AS2*CACHE*WRITE
                 +    /BEDT40*DATAWT
                 + /SWX*AS*DATAWT

IF (VCC) TAG0WT  = DATAWT*AS*/A10*/BANKSEL*/SWX
                 + DATAWT*AS*/A10*/BANKSEL*/BEDT40

IF (VCC) TAG1WT  = DATAWT*AS*A10*/BANKSEL*/SWX
                 + DATAWT*AS*A10*/BANKSEL*/BEDT40

IF (VCC) TAG2WT  = DATAWT*AS*/A10*BANKSEL*/SWX
                 + DATAWT*AS*/A10*BANKSEL*/BEDT40

IF (VCC) TAG3WT  = DATAWT*AS*A10*BANKSEL*/SWX
                 + DATAWT*AS*A10*BANKSEL*/BEDT40

DESCRIPTION:  CACHE CONTROLLER PAL
```

```
PAL16L8A                                                        11-6-84
U108-SHEET 6         MVME120,MVME121                            CKSM= 3823
PALMAT                                                          REV B

MATCH4L MATCH2U MATCH2L /BR MATCH3U MATCH3L COLUMN A10 BANKSEL GND
DPGNT BRAV /SETUP VMEAV70 MATCH1U MATCH1L MATCH4U /DPCASDIS /TAGMATCH VCC

IF (VCC) TAGMATCH = /A10*/BANKSEL*MATCH1U*MATCH1L
                  +  A10*/BANKSEL*MATCH2U*MATCH2L
                  + /A10* BANKSEL*MATCH3U*MATCH3L
                  +  A10* BANKSEL*MATCH4U*MATCH4L

IF (VCC) SETUP = DPGNT*/COLUMN
                 + SETUP*DPGNT

IF (VCC) DPCASDIS = /SETUP*DPCASDIS
                  + /COLUMN*DPCASDIS
                  + /DPGNT

IF (VCC) /BRAV = /BR + /VMEAV70

DESCRIPTION: CACHE MATCH QUALIFIER(BOARD HAS CACHE)
```

B

```
PAL16L8A                                                    11-16-34
U108-SHEET 6          MVME122                               CKSM= 2907
PALMAT22                                                    REV A

MATCH4L MATCH2U MATCH2L /BR MATCH3U MATCH3L COLUMN A10 BANKSEL GND
DPGNT BRAV /SETUP VMEAV70 MATCH1U MATCH1L MATCH4U /DPCASDIS /TAGMATCH VCC

IF (VCC) TAGMATCH = /VCC

IF (VCC) SETUP = DPGNT*/COLUMN
               + SETUP*DPGNT

IF (VCC) DPCASDIS = /SETUP*DPCASDIS
                  + /COLUMN*DPCASDIS
                  + /DPGNT

IF (VCC) /BRAV = /BR + /VMEAV70

DESCRIPTION: DUAL PORT CAS SETUP AND NO CACHE INDICATOR(CACHE NOT PRESENT)
```

```
PAL16L8A                                                    11-6-84
U108-SHEET 6          MVME123                               CKSM= 3823
PALMAT23                                                    REV A

MATCH4L MATCH2U MATCH2L /BR MATCH3U MATCH3L COLUMN A10 BANKSEL GND
DPGNT BRAV /SETUP VMEAV70 MATCH1U MATCH1L MATCH4U /DPCASDIS /TAGMATCH VCC

IF (VCC) TAGMATCH = /A10*/BANKSEL*MATCH1U*MATCH1L
                  +  A10*/BANKSEL*MATCH2U*MATCH2L
                  + /A10* BANKSEL*MATCH3U*MATCH3L
                  +  A10* BANKSEL*MATCH4U*MATCH4L

IF (VCC) SETUP = DPGNT*/COLUMN
               + SETUP*DPGNT

IF (VCC) DPCASDIS = /SETUP*DPCASDIS
                  + /COLUMN*DPCASDIS
                  + /DPGNT

IF (VCC) /BRAV = /BR + /VMEAV70

DESCRIPTION: CACHE MATCH QUALIFIER(BOARD HAS CACHE)
```

B

```
82S153                                                          3-06-84
U21-SHEET 7              MVME120/MVME121                        CKSM= CA17
PAL15320                                                        REV A

/LDS    /MMUORMFP  /RESET FC0 /UDS READ /CACHCLR /CONTSEL /DTACK GND
  1L        2L        3L    4   5L   6       7L        8L       9L   10
PROG /ROMEN ROMTIME UIACK CDTACK /READCR /LOADCR FC1 FC2 VCC
 11    12L    13      14     15     16L      17L   18  19  20

IF (VCC) LOADCR = /ROMTIME*CONTSEL*/READ*UDS
          17L        13       8L      6     5L      OR
        1834-1     0+23     0+15    0+11   0+9    0+38
         [-]       [23]     [15]    [11]   [9]    [33]

                +     RESET
                        3L         OR
                      46+5       46+38
                      [51]       [84]

IF (VCC) READCR = READ*CONTSEL*UDS
           16L      6      8L      5L      OR
         1835-1   92+10  92+15   92+9    92+39
          [-]     [102]  [107]   [101]   [131]

IF (VCC) CDTACK = ROMEN*ROMTIME
           15       12L    13       OR
         1836-0   138+21 138+22 138+40
        [1836]    [159]  [160]  [178]

                + MMUORMFP
                    2L       OR
                  184+3    184+40
                  [187]    [224]

                + CONTSEL*ROMTIME
                    8L       13       OR
                  276+15   276+22   276+40
                  [291]    [298]    [316]

                + CACHCLR
                    7L       OR
                  322+13   322+40
                  [335]    [362]

                + DTACK
                    9L       OR
                  368+17   368+40
                  [385]    [408]

IF (VCC) PROG   = /FC0     *FC1      *UDS      *LDS    *READ
           11       4        18        5L       1L       6          OR
         1840-0   414+7    414+32    414+9    414+1    414+10     414+44
         [1840]   [421]    [446]     [423]    [415]    [424]      [458]

IF (VCC) UIACK  = FC0*    FC1*    FC2
           14       4        18      19       OR
         1837-0   460+6   460+32  460+34   460+41
         [1837]   [466]   [492]   [494]    [501]

INPUTS ONLY = 9            12          13            18          19
          1472-("+35) 1544-("+35) 1580-("+35) 1760-("+35) 1796-("+35)
        [1472-1507]   [1544-1579] [1580-1615] [1760-1795] [1796-1831]

NOTE: NUMBERS IN BRACKETS INDICATE FUSES TO BE LEFT INTACT.
```

B

```
82S153A                                                          3-06-84
U21-SHEET 7            MVME122,MVME123                           CKSM= CA17
PAL15322                                                         REV A

/LDS    /MMUORMFP  /RESET FC0 /UDS READ /CACHCLR /CONTSEL /DTACK GND
  1L        2L        3L    4   5L   6       7L        8L       9L   10
PROG /ROMEN ROMTIME UIACK CDTACK /READCR /LOADCR FC1 FC2 VCC
 11    12L    13      14     15     16L      17L   18  19  20

IF (VCC) LOADCR = /ROMTIME*CONTSEL*/READ*UDS
          17L        13       8L      6     5L      OR
         1834-1    0+23     0+15    0+11    0+9    0+38
          [-]       [23]     [15]    [11]    [9]    [38]

                +      RESET
                         3L         OR
                        46+5       46+38
                        [51]       [84]

IF (VCC) READCR = READ*CONTSEL*UDS
           16L      6       8L      5L      OR
         1835-1   92+10  92+15   92+9    92+39
          [-]     [102]  [107]   [101]   [131]

IF (VCC) CDTACK = ROMEN*ROMTIME
           15       12L    13       OR
         1836-0   138+21 138+22 138+40
        [1836]    [159]  [160]  [178]

                + MMUORMFP
                    2L       OR
                   184+3    184+40
                   [187]    [224]

                + CONTSEL*ROMTIME
                    8L       13       OR
                   276+15   276+22   276+40
                   [291]    [298]    [316]

                + CACHCLR
                    7L       OR
                   322+13   322+40
                   [335]    [362]

                + DTACK
                    9L       OR
                   368+17   368+40
                   [385]    [408]

IF (VCC) PROG   = /FC0     *FC1      *UDS      *LDS    *READ
           11       4        18        5L        1L       6          OR
         1840-0   414+7    414+32    414+9    414+1    414+10     414+44
         [1840]   [421]    [446]     [423]    [415]    [424]      [458]

IF (VCC) UIACK  = FC0*    FC1*    FC2
           14       4       18      19       OR
         1837-0   460+6   460+32  460+34   460+41
         [1837]   [466]   [492]   [494]    [501]

INPUTS ONLY = 9            12          13            18          19
          1472-("+35) 1544-("+35) 1580-("+35) 1760-("+35) 1796-("+35)
         [1472-1507]  [1544-1579] [1580-1615] [1760-1795] [1796-1831]

NOTE: NUMBERS IN BRACKETS INDICATE FUSES TO BE LEFT INTACT.
```

B

```
PAL20L8A                                                    7-19-84
U34-SHEET 7      MVME120,MVME122                            CKSM= 5657
PALMAP20                                                    REV A

READ /MAS TA17 TA18 TA19 TA20 TA21 TA22 TA23 /VMEVF SX GND
/VMEVECT VECT /ROMEN /NOVMEBUS DS /IAORHIT /MFPSEL /MMUSEL
/CONTSEL /CACHCLR /NOGO VCC

IF (VCC) NOVMEBUS = /TA23*/TA22*/TA21*/TA20*/TA19*/TA18*/TA17
                    */VMEVECT*/VMEVF
                    + TA23*TA22*TA21*TA20*/TA19*/VMEVF
                    + IAORHIT*/VMEVF
                    + TA23*TA22*TA21*TA20*TA19*/TA18*/TA17*/VMEVF
                    + NOGO
                    + /SX
                    + /DS*NOVMEBUS

IF (VCC) MFPSEL   = MAS*TA23*TA22*TA21*TA20*/TA19*/TA18*TA17
                    */IAORHIT*SX*/NOGO*DS

IF (VCC) CONTSEL  = MAS*TA23*TA22*TA21*TA20*/TA19*TA18*/TA17
                    */IAORHIT*SX*/NOGO*DS

IF (VCC) MMUSEL   = MAS*TA23*TA22*TA21*TA20*/TA19*TA18*TA17
                    */IAORHIT*SX*/NOGO*DS

IF (VCC) ROMEN    = MAS*TA23*TA22*TA21*TA20*/TA19*/TA18*/TA17
                    */IAORHIT*SX*/NOGO
                    + MAS*/IAORHIT*SX*VECT*/VMEVECT

IF (VCC) CACHCLR  = MAS*TA23*TA22*TA21*TA20*TA19*/TA18*/TA17
                    */IAORHIT*SX*/NOGO*DS

DESCRIPTION:  BOARD MAP DECODER(128KBYTE ONBOARD DYNAMIC RAM)
```

```
PAL20L8A                                                    7-23-84
U34-SHEET 7          MVME121,MVME123                        CKSM= 5679
PALMAP21                                                    REV A

READ /MAS TA17 TA18 TA19 TA20 TA21 TA22 TA23 /VMEVF SX GND
/VMEVECT VECT /ROMEN /NOVMEBUS DS /IAORHIT /MFPSEL /MMUSEL
/CONTSEL /CACHCLR /NOGO VCC

IF (VCC) NOVMEBUS = /TA23*/TA22*/TA21*/TA20*/TA19
                  */VMEVECT*/VMEVF
                  + TA23*TA22*TA21*TA20*/TA19*/VMEVF
                  + IAORHIT*/VMEVF
                  + TA23*TA22*TA21*TA20*TA19*/TA18*/TA17*/VMEVF
                  + NOGO
                  + /SX
                  + /DS*NOVMEBUS

IF (VCC) MFPSEL   = MAS*TA23*TA22*TA21*TA20*/TA19*/TA18*TA17
                  */IAORHIT*SX*/NOGO*DS

IF (VCC) CONTSEL  = MAS*TA23*TA22*TA21*TA20*/TA19*TA18*/TA17
                  */IAORHIT*SX*/NOGO*DS

IF (VCC) MMUSEL   = MAS*TA23*TA22*TA21*TA20*/TA19*TA18*TA17
                  */IAORHIT*SX*/NOGO*DS

IF (VCC) ROMEN    = MAS*TA23*TA22*TA21*TA20*/TA19*/TA18*/TA17
                  */IAORHIT*SX*/NOGO
                  + MAS*/IAORHIT*SX*VECT*/VMEVECT

IF (VCC) CACHCLR  = MAS*TA23*TA22*TA21*TA20*TA19*/TA18*/TA17
                  */IAORHIT*SX*/NOGO*DS

DESCRIPTION:  BOARD MAP DECODER(512KBYTE ONBOARD DYNAMIC RAM)
```

B

```
PAL16L8A                                                        3-06-84
U16-SHEET 8          MVME120,MVME121,MVME122,MVME123            CKSM= 5780
PALMIS20                                                        REV A

RESET AS VDS /BUSBERR /BUSDTCK DPGNT /CACHCLR A02 A01 GND
VAS /CLR1 /NOVECT /14 /15 /SLAVECYC 17 /DPCYDN /CLR0 VCC

IF (VCC) 14 = /VAS*SLAVECYC
            + 14*SLAVECYC

IF (VCC) SLAVECYC = VDS*DPGNT
                  + VDS*SLAVECYC
                  + SLAVECYC*/14

IF (VCC) DPCYDN = RESET
                + BUSDTCK
                + BUSBERR

IF (VCC) CLR0 = CACHCLR*A01
              + RESET

IF (VCC) CLR1 = CACHCLR*A02
              + RESET

IF (VCC) 15 = A01*A02*AS
            + /RESET*15

IF (VCC) NOVECT = 15*/AS*/RESET
              + /RESET*NOVECT

DESCRIPTION:  MISCELLANEOUS FUNCTIONS
```

B

```
PAL16L2                                                    2-17-84
U28-SHEET 8       MVME120,MVME122                          CKSM= 0744
PALDP20                                                    REV A

A23 A22 A21 A20 A19 A18 A17 /LWORD /IACK GND
AM1 AM3 AM2 AM4 15 /DPMATCH AM0 AM5 VMEAV VCC

DPMATCH = /A23*/A22*/A21*/A20*/A19*/A18*/A17      ;SUPER PROG
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*AM2*AM1*/AM0

        + /A23*/A22*/A21*/A20*/A19*/A18*/A17      ;SUPER DATA
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*AM2*/AM1*AM0

        + /A23*/A22*/A21*/A20*/A19*/A18*/A17      ;USER PROG
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*/AM2*AM1*/AM0

        + /A23*/A22*/A21*/A20*/A19*/A18*/A17      ;USER DATA
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*/AM2*/AM1*AM0

DESCRIPTION: DUAL PORT MAP DECODER(128KBYTE ONBOARD RAM)
             SET FOR VMEBUS ADDRESSES 0-1FFFF
```

```
PAL16L2                                                    7-23-84
U28-SHEET 8       MVME121,VME123                           CKSM= 07CC
PALDP21                                                    REV A

A23 A22 A21 A20 A19 A18 A17 /LWORD /IACK GND
AM1 AM3 AM2 AM4 15 /DPMATCH AM0 AM5 VMEAV VCC

DPMATCH = /A23*/A22*/A21*/A20*/A19      ;SUPER PROG
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*AM2*AM1*/AM0

        + /A23*/A22*/A21*/A20*/A19      ;SUPER DATA
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*AM2*/AM1*AM0

        + /A23*/A22*/A21*/A20*/A19      ;USER PROG
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*/AM2*AM1*/AM0

        + /A23*/A22*/A21*/A20*/A19      ;USER DATA
          */IACK*/LWORD*/VMEAV
          *AM5*AM4*AM3*/AM2*/AM1*AM0

DESCRIPTION: DUAL PORT MAP DECODER(512KBYTE ONBOARD DYNAMIC RAM)
             SET FOR VMEBUS ADDRESSES 0-7FFFF
```

B

```
PAL16L8A                                                    5-31-84
U31-SHEET 9        MVME120,MVME122                          CKSM= 4242
PALCAS20                                                    REV A

/LDS /UDS TA17 TA18 TA19 TA20 TA21 TA22 TA23 GND
/RAMWRT /RAMGO /OK /UCAS /CASTIME /LCAS /VLCASEN VUCASEN 19 VCC

IF (VCC) LCAS = /TA23*/TA22*/TA21*/TA20*/TA19*/TA18*/TA17
                  *OK*LDS*CASTIME
                + /TA23*/TA22*/TA21*/TA20*/TA19*/TA18*/TA17
                  *OK*UDS*CASTIME*/RAMWRT
                + LCAS*CASTIME
                + VLCASEN*CASTIME
                + VLCASEN*LCAS
                + VUCASEN*CASTIME*/RAMWRT
                + VUCASEN*LCAS*/RAMWRT

IF (VCC) UCAS = /TA23*/TA22*/TA21*/TA20*/TA19*/TA18*/TA17
                  *OK*UDS*CASTIME
                + /TA23*/TA22*/TA21*/TA20*/TA19*/TA18*/TA17
                  *OK*LDS*CASTIME*/RAMWRT
                + UCAS*CASTIME
                + VUCASEN*CASTIME
                + VUCASEN*UCAS
                + VLCASEN*CASTIME*/RAMWRT
                + VLCASEN*UCAS*/RAMWRT

IF (VCC) RAMGO = /TA23*/TA22*/TA21*/TA20*/TA19*/TA18*/TA17
                   *UDS*OK*CASTIME
                 + /TA23*/TA22*/TA21*/TA20*/TA19*/TA18*/TA17
                   *LDS*OK*CASTIME

DESCRIPTION:  RAM CAS'S(128KBYTE ONBOARD DYNAMIC RAM)
              USE THIS PAL WHEN AN MMU IS PRESENT OR WHEN J13 1-2 ARE
              CONNECTED
```

```
PAL16L8A                                                    7-23-84
U31- SHEET 9      MVME121,MVME123                           CKSM= 430E
PALCAS21                                                    REV A

/LDS /UDS TA17 TA18 TA19 TA20 TA21 TA22 TA23 GND
/RAMWRT /RAMGD /CK /UCAS /CASTIME /LCAS /VLCASEN VUCASEN 19 VCC

IF (VCC) LCAS = /TA23*/TA22*/TA21*/TA20*/TA19
                 *CK*LDS*CASTIME
               + /TA23*/TA22*/TA21*/TA20*/TA19
                 *CK*UDS*CASTIME*/RAMWRT
               + LCAS*CASTIME
               + VLCASEN*CASTIME
               + VLCASEN*LCAS
               + VUCASEN*CASTIME*/RAMWRT
               + VUCASEN*LCAS*/RAMWRT

IF (VCC) UCAS = /TA23*/TA22*/TA21*/TA20*/TA19
                 *CK*UDS*CASTIME
               + /TA23*/TA22*/TA21*/TA20*/TA19
                 *CK*LDS*CASTIME*/RAMWRT
               + LCAS*CASTIME
               + VUCASEN*CASTIME
               + VUCASEN*UCAS
               + VLCASEN*CASTIME*/RAMWRT
               + VLCASEN*UCAS*/RAMWRT

IF (VCC) RAMGD = /TA23*/TA22*/TA21*/TA20*/TA19
                  *UDS*CK*CASTIME
                + /TA23*/TA22*/TA21*/TA20*/TA19
                  *LDS*CK*CASTIME

DESCRIPTION:  RAM CAS'S(512KBYTE ONBOARD DYNAMIC RAM)
```

B

```
PAL16R8                                                          7-20-84
U56-SHEET 10        MVME120,MVME121,MVME122,MVME123              CKSM= 7C40
PALCNT20                                                         REV A

CLK 2 3 4 5 6 7 8 9 GND
/OE /A7 /A6 /A5 /A4 /A3 /A2 /A1 /A0 VCC

A0 := /A0

A1 := A0*/A1 + /A0*A1

A2 := A0*A1*/A2 + /A0*A2 + /A1*A2

A3 := A0*A1*A2*/A3 + /A0*A3 + /A1*A3 + /A2*A3

A4 := A0*A1*A2*A3*/A4 + /A0*A4 + /A1*A4 + /A2*A4 + /A3*A4

A5 := A0*A1*A2*A3*A4*/A5 + /A0*A5 + /A1*A5 + /A2*A5 + /A3*A5
      + /A4*A5

A6 := A0*A1*A2*A3*A4*A5*/A6 + /A0*A6 + /A1*A6 + /A2*A6
      + /A3*A6 + /A4*A6 + /A5*A6

A7 := A0*A1*A2*A3*A4*A5*A6*/A7 + /A0*A7 + /A1*A7 + /A2*A7
      + /A3*A7 + /A4*A7 + /A5*A7 + /A6*A7

DESCRIPTION:  RAM REFRESH COUNTER
```

```
PAL16L8A                                                        3-06-84
U7-SHEET 12        MVME120,MVME121,MVME122,MVME123              CKSM= 6BB0
PALVME20                                                        REV A

VWRITE VDTACK VBERR /VMEAV /SLAVECYC /EN40 /UDS /DVAS READ GND
/LDS /NOBBERR /S244EN /EN /BUDS /RAMWRT /BLDS VDS /NOBDTACK VCC

IF (VCC) EN = /VBERR*/VDTACK*VMEAV*UDS*/READ
            + /VBERR*/VDTACK*VMEAV*LDS*/READ
            + SLAVECYC*/VWRITE*VDS
            + EN*UDS
            + EN*LDS

IF (VCC) S244EN = UDS*READ*DVAS
                + VWRITE*SLAVECYC*VDS*/VDTACK
                + LDS*READ*DVAS
                + S244EN*BUDS
                + S244EN*BLDS

IF (VCC) BUDS = DVAS*UDS*/VBERR*/VDTACK*VMEAV*READ
              + DVAS*UDS*/VBERR*/VDTACK*VMEAV*/READ*EN40
              + BUDS*UDS

IF (VCC) BLDS = DVAS*LDS*/VBERR*/VDTACK*VMEAV*READ
              + DVAS*LDS*/VBERR*/VDTACK*VMEAV*/READ*EN40
              + BLDS*LDS

IF (VCC) NOBBERR = /VBERR
                 + /BUDS*/BLDS

IF (VCC) NOBDTACK = /VDTACK
                  + /BUDS*/BLDS

IF (VCC) RAMWRT = /READ
                + VWRITE*SLAVECYC

DESCRIPTION:  VMEBUS STROBES ETC.
```

B

```
PAL20L8A                                                          4-15-85
U39-SHEET 15        MVME120, MVME121, MVME122, MVME123            CKSM=98D5
PALINT20-51AW4697B27                                              REV B

/IACK A02 A01 /IRQ1 /IRQ2 /IRQ3 /IRQ4 /IRQ5 /IRQ6 /IRQ7 /IEO GND /ABORT
/MFPIRQ /IPL0 /VPA /VMEVF A03 ALTINT /MFPIACK /IPL2 /IPL1 /IE VCC

IF (VCC) IPL0 = IE*IRQ1*/IRQ2*/IRQ4*/IRQ6*/MFPIRQ
       + IE*IRQ3*/IRQ4*/IRQ6*/MFPIRQ
       + IE*IRQ5*/IRQ6*/MFPIRQ
       + IE*IRQ7
       + IE*ABORT

IF (VCC) IPL1 = IE*IRQ2*/IRQ4*/IRQ5
       + IE*IRQ3*/IRQ4*/IRQ5
       + IE*IRQ6
       + IE*MFPIRQ
       + IE*IRQ7
       + IE*ABORT

IF (VCC) IPL2 = IE*IRQ4
       + IE*IRQ5
       + IE*IRQ6
       + IE*MFPIRQ
       + IE*IRQ7
       + IE*ABORT

IF (VCC) MFPIACK = A03*A02*/A01*IACK*/ALTINT
                 + MFPIACK*IACK

IF (VCC) VPA = /A01*/A02*/A03*IACK
       + A01*A02*A03*IACK*ABORT
       + ALTINT*IACK*/VMEVF

IF (VCC) VMEVF = A03*A02*A01*/ABORT*IACK*/ALTINT
       + A03*A02*/A01*IACK*IEO*/ALTINT
       + A03*/A02*IACK*/ALTINT
       + /A03*A02*IACK*/ALTINT
       + /A03*/A02*A01*IACK*/ALTINT
       + VMEVF*IACK

DESCRIPTION:   INTERRUPT HANDLER
```

INDEX

e. Read to user program data space causes one of two cycle types to occur:

   1. Cache hit cycle if in the indexed cache entry; V1=1, V2=1, F=0, and the cache tag address matches the current logical address from the MPU.

   2. Cache validate cycle if in the indexed cache entry; V1=0, V2=0, F=1, or the cache tag address does not match the current logical address from the MPU.

f. Read to supervisory program space causes one of two cycle types to occur:

   1. Cache hit cycle if in the indexed cache entry; V1=1, V2=1, F=1, and the cache tag address matches the current logical address from the MPU.

   2. Cache validate cycle if in the indexed cache entry; V1=0, V2=0, F=0, or the cache tag address does not match the current logical address from the MPU.

g. Write to user program space should never be used by the software if cache is enabled.

h. Write to supervisory program space should never be used by the software if cache is enabled.

i. Any memory access cycle that would normally be a cache validate cycle, becomes a cache invalidate cycle if an address error or a bus error occurs.

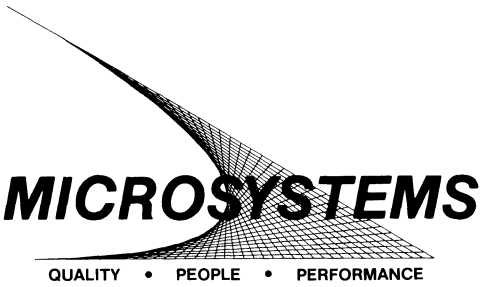
j. The cache can be enabled/disabled by software using the CACHEN bit in the MCR. When the cache is disabled, all cycles are cache ignore cycles.

k. The cache can be frozen/unfrozen by software using the FREEZE bit in the MCR. The only effect of freezing the cache is that all cache validate cycles become cache ignore cycles. Cache invalidate cycles are not affected.

Cache Flushing

The cache can be flushed by the software accessing a particular location in the memory map. The access can be either a read cycle or a write cycle. When using configurations a,b,c, the cache is flushed by accessing location F80006. When using configuration d, all the cache is flushed by accessing location F80006 or the user program space is flushed by accessing location F80004, or the supervisory program space is flushed by accessing location F80002. Locations F80002 or F80004 should not be accessed when using configurations a,b,c. When cache is flushed, its V1 and V2 bits are cleared to 0 making all cache entries become invalid. Reset also causes cache to be cleared. The cache clear function happens whether or not the cache is enabled or frozen in the MCR.

SUGGESTION/PROBLEM REPORT



Motorola welcomes your comments on its products and publications. Please use this form.

To: Motorola Inc.
Microsystems
2900 S. Diablo Way
Tempe, Arizona 85282
Attention: Publications Manager
Maildrop DW164

Product: _____ Manual: _____

COMMENTS: _____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Please Print

Name _____ Title _____

Company _____ Division _____

Street _____ Mail Drop _____ Phone _____

City _____ State _____ Zip _____

For Additional Motorola Publications
Literature Distribution Center
616 West 24th Street
Tempe, AZ 85282
(602) 994-6561

Four Phase/Motorola Customer Support, Tempe Operations
(800) 528-1908
(602) 438-3100

MOTOROLA

MOTOROLA ***Semiconductor Products Inc.***

P.O. BOX 20912 • PHOENIX, ARIZONA 85036 • A SUBSIDIARY OF MOTOROLA INC.

19108 PRINTED IN USA (7/85) MESSENGER 1500