MVME133/D1

MVME133 VMEmodule 32-Bit Monoboard Microcomputer User's Manual

MOTOROLA INC.

SYSTEMS

MOTOROLA

MVME133/D1

SEPTEMBER 1986

MVME133 VMEmodule

32-BIT MONOBOARD MICROCOMPUTER

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.

First Edition

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.

This manual, MVME133/D1, dated September 1986, covers models MVME133 (part number 01-W3434B01) and MVME133-1 (part number 01-W3434B02). Newer versions of these modules have part numbers 01-W3465B01 and 01-W3465B02. Differences between models are explained in Customer Letter MVME133/L1. Please contact your Motorola Sales Office representative in order to obtain a copy of this letter.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont'd)

TABLE OF CONTENTS (cont'd)

LIST OF ILLUSTRATIONS

TABLE OF CONTENTS (cont'd)

Page

LIST OF TABLES

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 Motorola MVME133 and MVME133-1 VMEmodule 32-Bit Monoboard Microcomputers (referred to as the MVME133 throughout this manual). The MVME133 is shown in Figure 1-1.

1.2 MODEL DESIGNATIONS

The MVME133 is available in two variations, which are listed in Table 1-1.

TABLE 1-1. MVME133 Model Designations

PRODUCT NUMBER	DESCRIPTION
MVME133	12.5 MHz MC68020 with 12.5 MHz MC68881
MVME133-1	16.67 MHz MC68020 with 16.67 MHz MC68881

1.3 FEATURES

The MVME133 is an intelligent single-board processor module containing both the MC68020 microprocessor and the MC68881 coprocessor. The main features of the MVME133 are as follows:

- Double-high/single-wide VMEboard
- Address 24/Data 32 (A24/D32) VMEbus master (A24/D16 compatible)
- MC68020 Microprocessor with 32-bit address and data, 12.5 MHz (MVME133) or 16.67 MHz (MVME133-1)
- MC68881 Floating Point Coprocessor, 12.5 MHz (MVME133) or 16.67 MHz (MVME133-1)
- 1Mb of shared local Dynamic RAM, 32-bits wide, accessible from VMEbus
- Four 28-pin JEDEC sockets for ROMs/PROMs/EPROMs/EEPROMs (in two banks, each 16-bits wide) (total 256Kb maximum)
- Three 8-bit programmable timers for tick and watchdog functions

03-86-3833

NOTE: Debug monitor EPROMs U31 and U46 are not included with the MVME133, but are sold separately as MVME133Bug.

FIGURE 1-1. MVME133 VMEmodule 32-Bit Monoboard Microcomputer

- Time-of-day clock/calendar (real-time clock) (MM58274A)
- Front panel asynchronous DB25 serial debug RS-232C port (on MC68901 MFP)
- Dual multiprotocol (synchronous/asynchronous) serial ports (Z8530)

  RS-232C (port B)
  RS-485/RS-422 (port A)

- VMEbus system controller functions with level-three arbiter
- Single level bus requester (level jumper selectable)
- VMEbus interrupter
- VMEbus interrupt handler for all seven levels
- Front panel RUN, HALT, and FAIL status LEDs
- Front panel RESET and ABORT switches
- Remote reset through edge connector P2
- Five-position software-readable header; part of Module Status Register (MSR)

1.4 SPECIFICATIONS

General specifications for the MVME133 are provided in Table 1-2. Paragraphs 1.4.1 and 1.4.2 detail cooling requirements and FCC compliance, respectively.

1.4.1 Cooling Requirements

The Motorola MVME133 VMEmodule is specified, designed, and tested to operate reliably with an incoming air temperature range from 0 degrees C to 55 degrees C (32 degrees to 131 degrees F) with forced air cooling at a velocity typically achievable by using a 71 CFM axial fan. Temperature qualification is performed in a standard Motorola VMEsystem 1000 chassis. Twenty-five watt load boards are inserted in two card slots, one on each side, adjacent to the board under test, to simulate a high power density system configuration. An assembly of three axial fans, rated at 71 CFM per fan, is placed directly under the VME card cage. The incoming air temperature is measured between the fan assembly and the card cage, where the incoming airstream first encounters the module under test. Test software is executed as the module is subjected to ambient temperature variations. Case temperatures of critical, high power density integrated circuits are monitored to ensure component vendors specifications are not exceeded.

TABLE 1-2. MVME133 Module Specifications

CHARACTERISTICS	SPECIFICATIONS
Power requirements (with full set of ROMs/PROMs/ EPROMs/EEPROMs	+5 Vdc @ 4.0 A (typ), 5.0 A (max)(MVME133) +5 Vdc @ 4.5 A (typ), 6.0 A (max)(MVME133-1) +12 Vdc @ 100 mA (typical), 250 mA (maximum) -12 Vdc @ 100 mA (typical), 250 mA (maximum)
Microprocessor	MC68020
Coprocessor	MC68881
Clock signal to MPU	12.5 MHz (MVME133); 16.67 MHz (MVME133-1)
Addressing	
Total range (on and offboard)	16Mb (A24)
ROM/PROM/EPROM/EEPROM	256Kb maximum: four sockets (two banks of two each, 16 bits wide) for 2K x 8, 8K x 8, 16K x 8, 32K x 8, or 64K x 8 devices
Dynamic RAM	1Mb (32 bits wide)
Serial I/O ports	Port B multiprotocol RS-232C through P2
	Port A multiprotocol RS-485/422 through P2
	Asynchronous RS-232C debug serial port DCE (to terminal only) through front panel J14
Time-of-Day Clock/Calendar	0.1 second resolution
Timers (on MC68901 MFP)	4 total (3 available to user)
Debug port (not available)	8 bit
Watchdog	8 bit
Tick	8 bit
Spare	8 bit
Bus configuration	Data Transfer Bus (DTB) master or slave, with 24-bit address (A24) and 32-bit data (D32)
Interrupt handler	Any or all onboard plus up to seven VMEbus interrupts

TABLE 1-2. MVME133 Module Specifications (cont'd)

CHARACTERISTICS	SPECIFICATIONS
Interrupter	Level-three with status ID of $FF
Bus arbitration	When MVME133 is system controller, it arbitrates bus requests/grants on level 3 only
Reset	By SYSRESET*, power-up, RESET switch, watchdog timer time-out, remote reset, or MC68020 RESET instruction.
Temperature	
Operating (Refer to paragraph 1.4.1.)	0 degrees to 55 degrees at point of entry of forced air (approximately 150 LFM)
Storage	-40 degrees to 85 degrees C
Relative humidity	5% to 90% (non-condensing)
Physical characteristics (including front panel)	
Height	10.31 inches (261.8 mm)
Width	7.40 inches (188.0 mm)
Thickness	0.83 inches (21.0 mm)
Connectors	
VMEbus	DIN No. 41612C96 (P1, P2)
RS-232C	DB-25 female (J14)

While the exact amount of airflow required for cooling depends on the ambient air temperature and the type, number, and location of boards and other heat sources, adequate cooling can usually be achieved with 10 CFM and 150 LFM flowing over the module. Less airflow is required to cool the module in environments having lower maximum ambients. Under more favorable thermal conditions, it may be possible to operate the module reliably at higher than 55 degrees C with increased airflow. It is important to note that there are several factors, in addition to the rated CFM of the air mover, which determine the actual volume and speed of air flowing over a module.

1

1.4.2 FCC Compliance

This VMEmodule (MVME133) was tested in an FCC-compliant chassis, and meets the requirements for Class A equipment. FCC compliance was achieved under the following conditions:

a. Shielded cables on all external I/O ports.

b. Cable shields connected to earth ground via metal shell connectors bonded to a conductive module front panel.

c. Conductive chassis rails connected to earth ground. This provides the path for connecting shields to earth ground.

d. Front panel screws properly tightened.

For minimum RF emissions, it is essential that the conditions above be implemented; failure to do so could compromise the FCC compliance of the equipment containing the module.

1.5 GENERAL DESCRIPTION

The MVME133 32-Bit Monoboard Microcomputer is a double-high VMEmodule. The module has large onboard DRAM (1Mb), ROM/PROM/EPROM/EEPROM capability (256Kb), serial ports including debug port, floating point coprocessor, tick timer, watchdog timer, time-of-day clock/calendar, and VMEbus interface with system controller functions. The MVME133 is a single-board MPU module intended to be used in a single processor system, but not stand-alone.

1.6 EQUIPMENT REQUIRED

The following equipment is required to make a complete system using the MVME133:

Terminal(s)
Disk drives and controllers
Chassis and power supply
Debug monitor MVME133Bug
Operating system

Note that the MVME133 contains no parallel ports. To use a parallel device, such as a printer, with the MVME133, it is necessary to add a module such as the MVME050 System Controller Module to your system.

1.7 RELATED DOCUMENTATION

The following publications are applicable to the MVME133 and may provide additional helpful information. If not shipped with this product, they may be purchased from Motorola's Literature Distribution Center, 616 West 24th Street, Tempe, Arizona 85282; telephone (602) 994-6561. Non-Motorola documents may be obtained from the sources listed.

DOCUMENT TITLE	MOTOROLA PUBLICATION NUMBER
The VMEbus Specification	HB212/D
MVME133 Debug Monitor User's Manual	MVME133BUG
VERSAdos to VME Hardware and Software Configuration User's Manual	MVMEVDOS
MC68020 32-Bit Microprocessor User's Manual	MC68020UM
MC68881 Floating-Point Coprocessor User's Manual	MC68881UM
MC68901 Multi-Function Peripheral Data Sheet	ADI-984

MM58274 Real-Time Clock; data sheet and application note 365; National Semiconductor Corporation, 2900 Semiconductor Drive, Santa Clara, CA 95051

Z8530 Serial Communications Controller; data sheet; Zilog, Inc., Corporate Communications, Building A, 1315 Dell Ave, Campbell, California 95008

CHAPTER 2

HARDWARE PREPARATION AND INSTALLATION INSTRUCTIONS

2.1 INTRODUCTION

This chapter provides unpacking instructions, hardware preparation, and installation instructions for the MVME133.

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.

CAUTION

AVOID TOUCHING AREAS OF INTEGRATED CIRCUITRY; STATIC DISCHARGE CAN DAMAGE CIRCUITS.

2.3 HARDWARE PREPARATION

To select the desired configuration and ensure proper operation of the MVME133, 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 MVME133. The MVME133 has been factory tested and is shipped with factory-installed jumper configurations that are also shown on Figure 2-1. The MVME133 will operate with its optional add-on Debug Monitor, MVME133Bug, with the factory-installed jumper configurations. Headers J1 through J13, and J15 through J16 are factory-configured as follows:

2

. System controller enable	J1, 1-2	MVME133 module is system controller
. Onboard RAM base address select	J2, 1-2, 3-4	RAM base address is $000000 as seen from VMEbus
. VMEbus requester level select	J3, 5-6 and J4, 1-2, 5-6, 7-8, 9-11, 10-12	Level three requested
. RMW cycle type select	J5, 1-2	MVME133 requests VMEbus mastership on all RMW cycles
. ROM/PROM/EPROM/EEPROM size	J6, 1-3, 4-6 and J7, 1-3, 4-6	Banks 1 & 2 each set for two 64K x 8 ROMs/PROMs/ EPROMs
. Global time-out	J8, 1-2	Enabled if MVME133 is the system controller
. RESET switch	J9, 1-2	Enabled
. ABORT switch	J10, 1-2	Enabled
. VMEbus interrupter	J11, 1-2	Enabled. DO NOT CHANGE THIS FACTORY CONFIGURATION.
. VMEbus interrupts	J12, 1-2, 3-4, 5-6, 7-8, 9-10, 11-12, 13-14	IRQ1* through IRQ7* all enabled
. Serial port B configuration	J13, 1-2, 4-5, 7-8, 10-11, 13-14, 16-17, 19-20, 22-23, 25-26, 28-29, 31-32	Port B as DCE (to terminal)

NOTE: J14 is the front-panel RS-232C connector.

. Software-readable header	J15, 1-2, 3-4, 5-6, 7-8, 9-10	Module Status Register (MSR) bits 0 through 4 all = 0
. Serial ports RTXCx source select	J16, 1-3, 2-4	RTXCA and RTXCB inputs from onboard 1.23 MHz signal

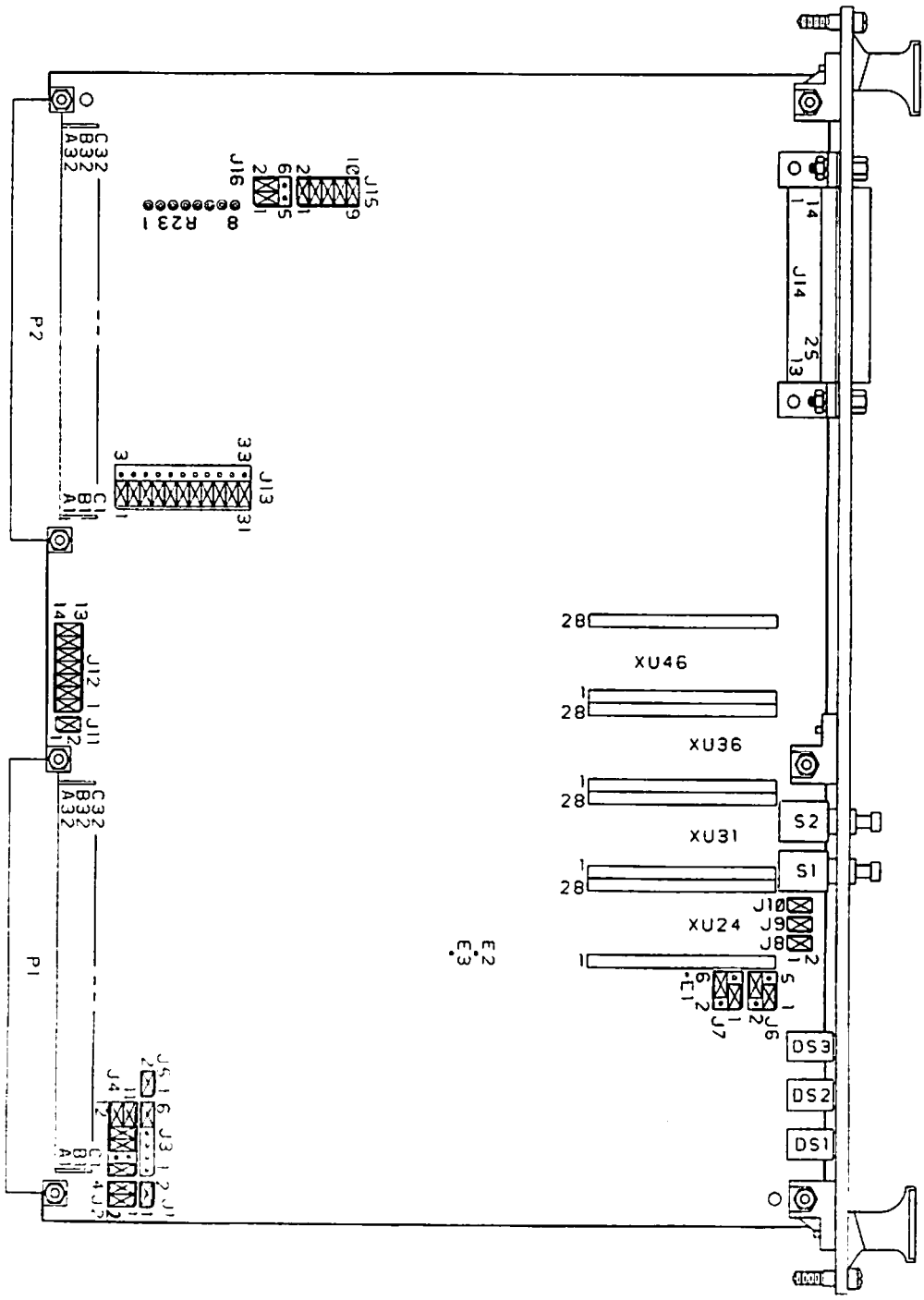
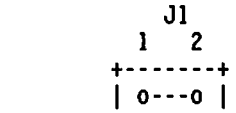


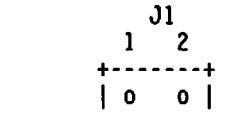
FIGURE 2-1. MVME133 Headers and Connectors

2

2.3.1 System Controller Enable Header (J1)



SYSTEM CONTROLLER
(FACTORY CONFIGURATION)

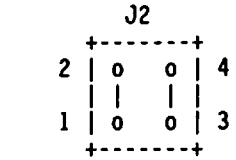


NOT SYSTEM CONTROLLER

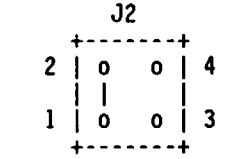
2.3.2 Onboard RAM Base Address Select Header (J2)

NOTE

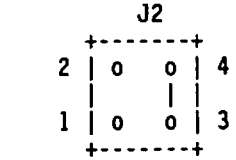
J2 controls the base address of the RAM, only as seen by the VMEbus. But the RAM is shared. To the onboard logic (for example, a monitor), the RAM address is fixed at $00000-$FFFFF. Refer to the memory map information in Chapter 3 and the shared RAM information in Chapter 4 for more details.



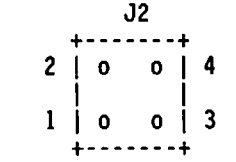
BASE ADDRESS = $000000
(FACTORY CONFIGURATION)



BASE ADDRESS = $100000

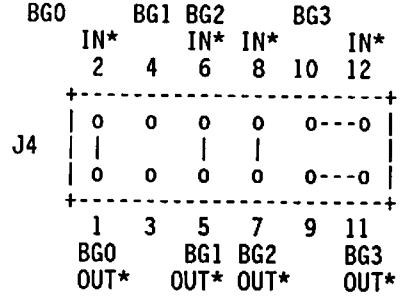


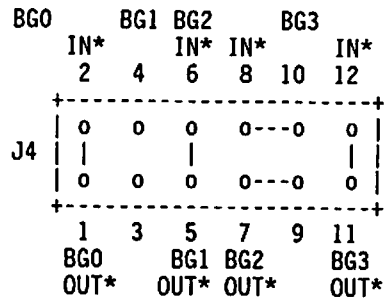
BASE ADDRESS = $200000

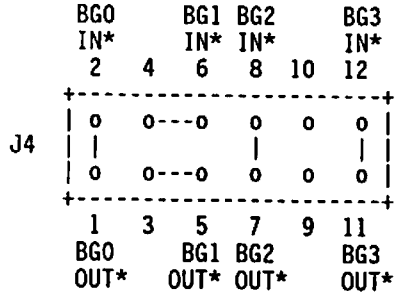


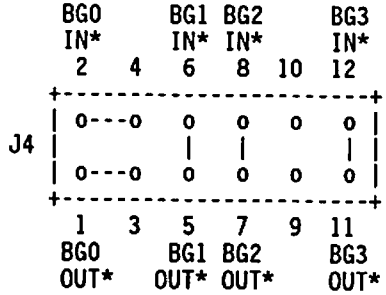
BASE ADDRESS = $300000

2.3.3 VMEbus Requester Level Select Headers (J3, J4)



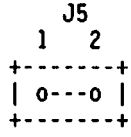




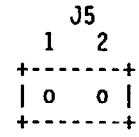


2

2.3.4 RMW Cycle Type Select Header (J5)

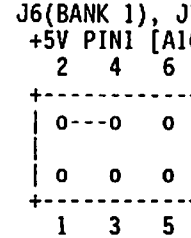


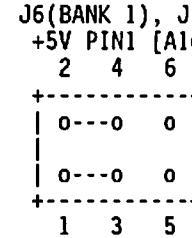
MVME133 REQUESTS VMEbus MASTERSHIP ON ALL RMW CYCLES (FACTORY CONFIGURATION).

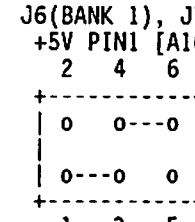


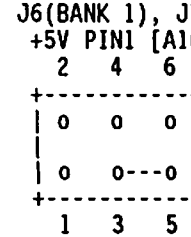
MVME133 DOES NOT REQUEST VMEbus MASTERSHIP ON RMW CYCLES TO ONBOARD RAM. SOFTWARE MUST NEVER GENERATE RMW CYCLES TO THE VMEbus, AND VMEbus ACCESSES TO ONBOARD RAM MUST NEVER INCLUDE MULTIPLE ADDRESS RMW CYCLES.

2.3.5 ROM/PROM/EPROM/EEPROM Size Headers (J6, J7)



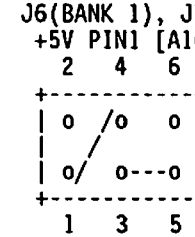




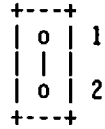


NOTES

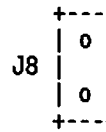
1. Customer must provide all the ROM, PROM, EPROM, or EEPROM chips to install in the four sockets.
2. Bank 1 = XU31 and XU46, bank 2 = XU24 and XU36. Each bank may be jumpered differently.
3. Refer to paragraph 4.3.12.1.



2.3.6 Global Time-out Header (J8)

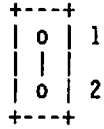


GLOBAL TIME-OUT ENABLED (IF SYSTEM CONTROLLER). REFER TO PARAGRAPH 2.3.1. (FACTORY CONFIGURATION)

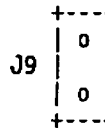


GLOBAL TIME-OUT DISABLED

2.3.7 RESET Switch Header (J9)

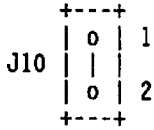


RESET SWITCH ENABLED (FACTORY CONFIGURATION)

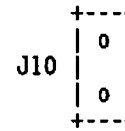


RESET SWITCH DISABLED

2.3.8 ABORT Switch Header (J10)

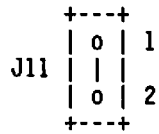


ABORT SWITCH ENABLED (FACTORY CONFIGURATION)



ABORT SWITCH DISABLED

2.3.9 VMEbus Interrupter Header (J11)



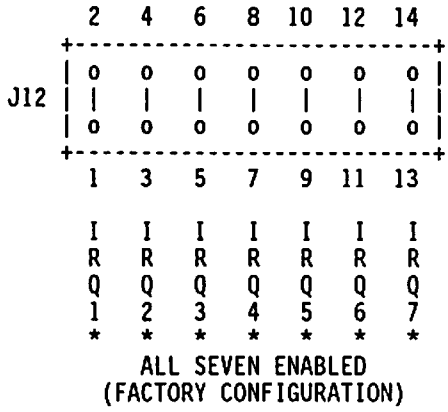
ENABLED (FACTORY CONFIGURATION)

CAUTION

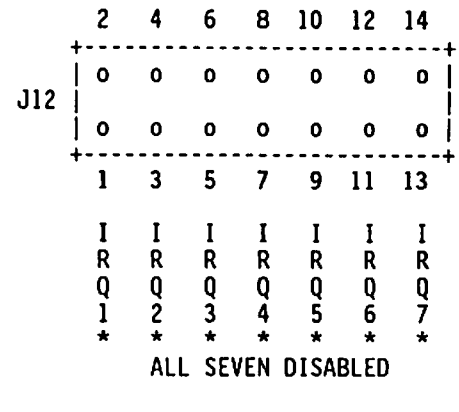
DO NOT CHANGE FROM THIS FACTORY CONFIGURATION OR THE MVME133 WILL NOT OPERATE PROPERLY.

2

2.3.10 VMEbus Interrupts Header (J12)

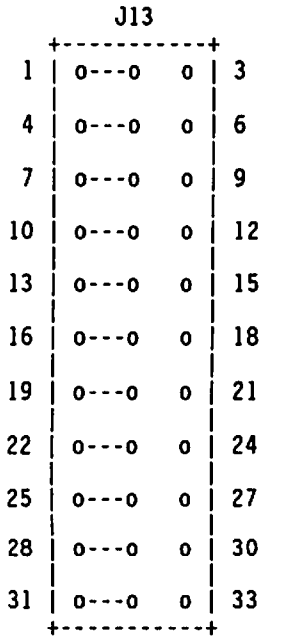


ALL SEVEN ENABLED
(FACTORY CONFIGURATION)

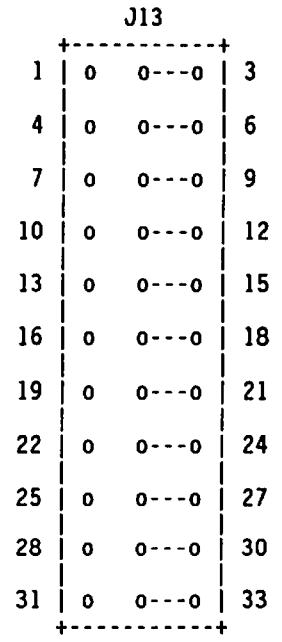


ALL SEVEN DISABLED

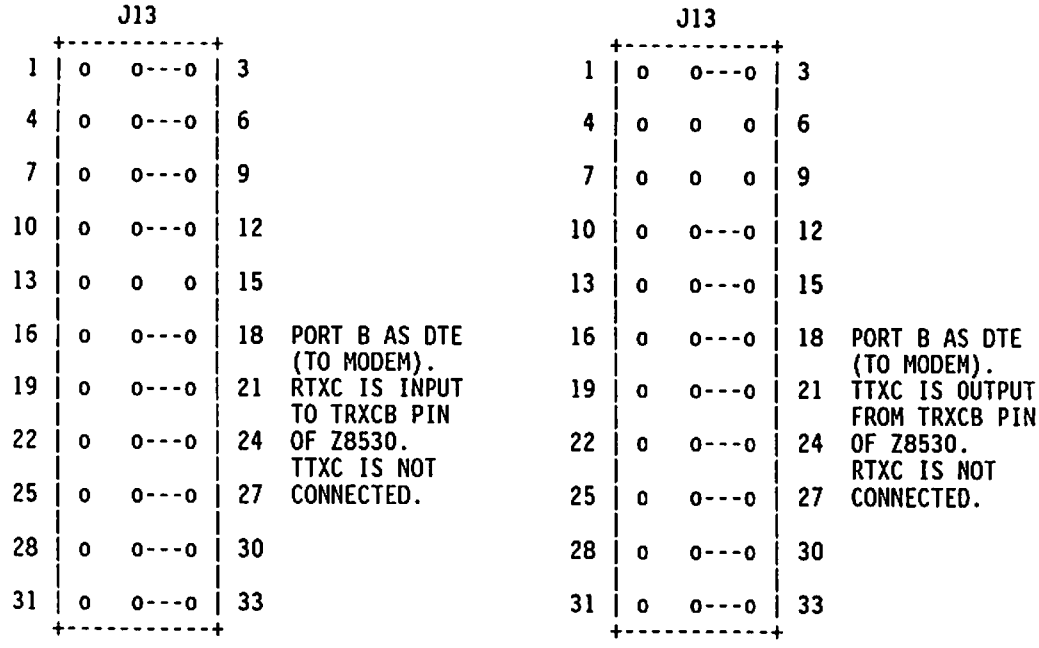
2.3.11 Serial Port B Configuration Header (J13)



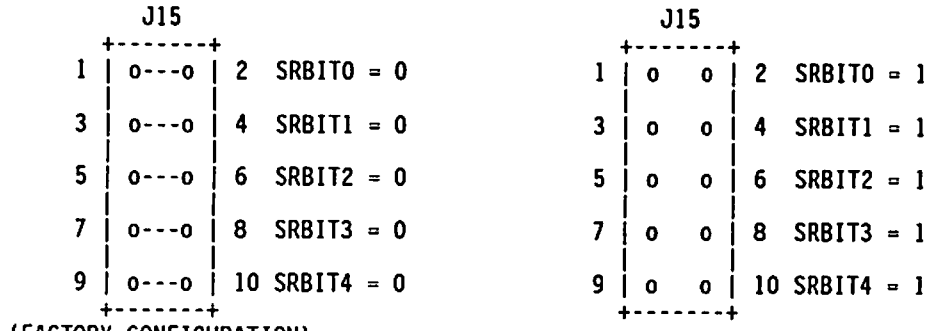
PORT B AS DCE (TO TERMINAL). (FACTORY CONFIGURATION)



PORT B AS DTE (TO MODEM). RTXC IS INPUT TO TRXCB PIN OF Z8530 AND IS ECHOED ON TTXC.



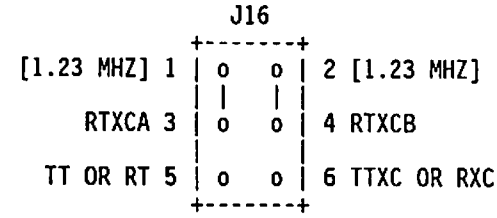
2.3.12 Software-Readable Header for Module Status Register (MSR) (J15)



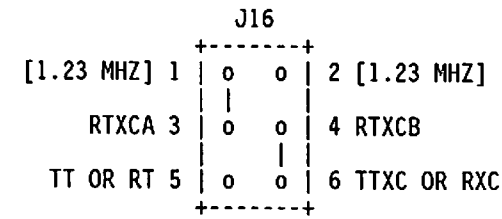
(FACTORY CONFIGURATION)

2

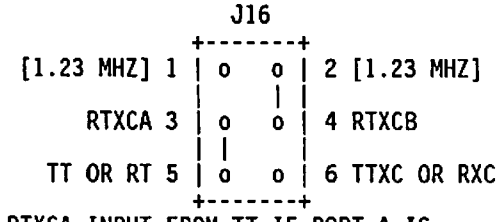
2.3.13 Serial Ports RTXCx Source Select Header (J16)



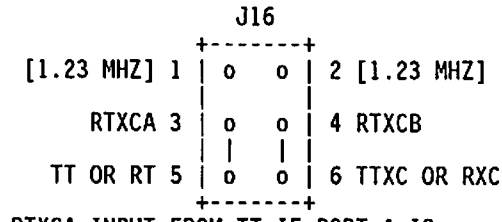
RTXCA INPUT FROM ONBOARD 1.23 MHz.
RTXCB INPUT FROM ONBOARD 1.23 MHz.
(FACTORY CONFIGURATION)



RTXCA INPUT FROM ONBOARD 1.23 MHz.
RTXCB INPUT FROM TTXC IF PORT B IS CONFIGURED TO GO TO A TERMINAL, OR FROM RXC IF PORT B IS CONFIGURED TO GO TO A MODEM.



RTXCA INPUT FROM TT IF PORT A IS CONFIGURED AS A SLAVE, OR FROM RT IF PORT A IS CONFIGURED AS A MASTER.
RTXCB INPUT FROM ONBOARD 1.23 MHz.



RTXCA INPUT FROM TT IF PORT A IS CONFIGURED AS A SLAVE, OR FROM RT IF PORT A IS CONFIGURED AS A MASTER.
RTXCB INPUT FROM TTXC IF PORT B IS CONFIGURED TO GO TO A TERMINAL, OR FROM RXC IF PORT B IS CONFIGURED TO GO TO A MODEM.

NOTE

Refer to paragraph 2.4.3, system considerations, for possible installation of resistor pack R23 terminators for port A signals.

2.4 INSTALLATION INSTRUCTIONS

The following paragraphs discuss installation of the MVME133 module into a VME chassis, connection of an RS-232C terminal and cable, and system considerations. Ensure that desired ROM/PROM/EPROM/EEPROM devices (such as those for the MVME133Bug debug monitor in sockets XU31 and XU46) are installed and configured, and that all other headers are configured for desired operation.

2.4.1 MVME133 Module Installation

Now that the MVME133 module is ready for installation, proceed as follows:

a. Turn all equipment power OFF.

CAUTION

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

b. The MVME133 module may be installed in any double-height unused card slot, if it is not configured as system controller. If the MVME133 is configured as system controller, it must be installed in the left-most card slot (slot 1) to correctly initiate the bus-grant daisy-chain.

c. Carefully slide the MVME133 module into the card slot. Be sure module is seated properly into connectors on the backplane. Fasten module in chassis with screws provided, making good contact with the transverse mounting rails to minimize RFI emissions.

d. Connect the required cables to the MVME133 module at the P2 backplane connector, to mate with peripherals at the RS-232C and/or RS-485 serial ports. These cables are not provided with the MVME133 module and must be made or provided by the user. (Motorola recommends using shielded cables for all connections to peripherals to minimize radiation.) Connect the peripherals to the cables. Install any other required VMEmodules in the system.

e. Turn equipment power ON.

2.4.2 Terminal Connection

The RS-232C port on the front panel is configured for DCE (to terminal) operation only. A 25-pin RS-232C cable may be connected to the front panel female connector J14, with the other end connected to a compatible terminal. This cable is not provided with the MVME133 module, and must be made or provided by the user. (Motorola recommends using shielded cables for all connections to peripherals to minimize radiation.) Note that J14 has a metal shell and jack posts that are electrically connected to the MVME133 front panel. If the MVME133 front panel is electrically connected to the chassis ground, then the shell and jack posts are connected to chassis ground. Refer to Table 5-3 and Appendix B for detailed information on the signals supported.

NOTE

The user may change J14 to a "to modem" configuration by providing a "null-modem" cable that switches certain signals.

2.4.3 System Considerations

The MVME133 may be used by itself or with other VMEbus controllers. The MVME133 is not intended to be used as an Intelligent Peripheral Controller (IPC). It is intended to be used as a VMEbus master.

Whether the MVME133 operates as a VMEbus master or as a VMEbus slave, it is configured for 24 bits of address and 32 bits of data (A24/D32). Many VME chassis are A24/D16 or A32/D32. The MVME133 is configured as a 16-bit data port or as a 32-bit data port depending on address alignment on a longword boundary, and the status of address bit A24. Refer to theory details in Chapter 4. There can also be system problems with bus error (BERR*) and Read-Modify-Write (RMW) cycles. These details are also discussed in Chapter 4.

The MVME133 uses the address modifier lines in such a way that it responds to short or standard addressing (AM = $3E, $3D, $3A, $39, $2D, or $29) when it is VMEbus master, but only to standard addressing (AM = $3E, $3D, $3A, or $39) when it is a VMEbus slave.

Note that the MVME133 contains no parallel ports. To use a parallel device, such as a printer, with the MVME133, it is necessary to add a module such as the MVME050 System Controller Module to your system.

A single resistor package, SIP, with four 120-ohm resistors, may be needed at R23 for proper and reliable system operations with the RS-485 port (port A). In systems where RS-485 multi-drop cable is used to connect many RS-485 ports (e.g., many MVME133s), noise on the cable may be read as spurious data and/or cause undesired interrupts unless the cable is terminated properly. The recommended method is to terminate each of the two ends of the cable with a 120-ohm resistor. For systems using MVME133s, proper termination can be accomplished by installing an eight-pin resistor pack with four 120-ohm resistors (actually, any value between 90 and 150 ohms is acceptable) at R23 on two of the MVME133 modules, one at each end of the RS-485 cable.

Termination may also be required if devices connecting to the MVME133 RS-485 port may be OFF or not online when the RS-485 port is enabled.

There may be some software problems using the MVME133, caused by mask errors in the MC68020 chip. Check the mask number of the MC68020 on your MVME133, and refer to Appendix A for the appropriate problem discussion.

CHAPTER 3

OPERATING INSTRUCTIONS

3.1 INTRODUCTION

This chapter provides the necessary information to use the MVME133 module in a system configuration. This includes controls and indicators, and memory map details.

3.2 CONTROLS AND INDICATORS

The MVME133 module has ABORT and RESET switches, and FAIL, HALT, and RUN indicators, all of which are located on the front panel of the module.

3.2.1 ABORT Switch S1

The ABORT switch is debounced and brought into the interrupt handler as a level seven interrupt. Refer to the interrupt handler description in Chapter 4 for details.

3.2.2 RESET Switch S2

The front panel RESET switch resets all onboard devices (including the MPU) and drives SYSRESET* low if the MVME133 is the system controller. (The MVME133 also drives SYSRESET* low at power up if it is configured as the system controller. Refer to the reset description in Chapter 4 for details.)

3.2.3 FAIL, HALT, and RUN Indicators DS1, DS2, and DS3

There are three LEDs on the MVME133: RUN, HALT, and FAIL. RUN is on (green) when [AS*] is low. HALT is on (red) when reset (any reset except the RESET instruction from the MPU) is true or when the [HALT*] line is low. FAIL is on (red) when the [BRDFAIL] line is high. Table 3-1 describes the module status for all possible combinations of these LEDs.

3.3 MVME133 MEMORY MAPS AND MAP DECODER

At the beginning of each MPU cycle, the map decoder determines what kind of cycle takes place and which device is selected within that cycle type. Cycle types are determined by the function code lines FC2-FC0, which are driven by the MC68020. Table 3-2 shows the cycle types and the devices that respond.

TABLE 3-1. Front Panel LEDs and MVME133 Status

FAIL DS1 RED	HALT DS2 RED	RUN DS3 GREEN	MVME133 STATUS
off	off	off	No power is applied to the module, or the MPU is not the current local bus master.
off	off	ON	Normal operation.
off	ON	off	MPU is halted.
off	ON	ON	MPU is running and VMEbus deadlocks or real-time-clock read cycles are occurring. Frequency of VMEbus deadlocks/real-time-clock reads determines intensity of HALT LED.
ON	off	off	MPU is not current local bus master. Also, [BRDFAIL] has not been cleared since reset or software has set [BRDFAIL].
ON	off	ON	[BRDFAIL] has not been cleared since reset or software has set [BRDFAIL].
ON	ON	off	MPU is halted and [BRDFAIL] has not been cleared since reset or software has set [BRDFAIL].
ON	ON	ON	MPU is running and VMEbus deadlocks or real-time-clock reads are occurring. Frequency of VMEbus deadlocks/real-time-clock reads determines intensity of HALT LED. Also [BRDFAIL] has not been cleared since reset or software has set [BRDFAIL].

TABLE 3-2. Cycle Types and Responding Devices

FC2	FC1	FC0	CYCLE TYPE	MVME133 DEVICES THAT RESPOND
0	0	0	reserved	None except dynamic RAM
0	0	1	User Data	All except interrupt handler and MC68881
0	1	0	User Program	All except interrupt handler and MC68881
0	1	1	reserved	None (causes local time-out)
1	0	0	reserved	None except dynamic RAM
1	0	1	Supervisory Data	All except interrupt handler and MC68881
1	1	0	Supervisory Program	All except interrupt handler and MC68881
1	1	1	CPU (IACK)	VMEbus, Z8530, MK68901, interrupt handler
1	1	1	CPU (coprocessor)	MC68881

3.3.1 Main Memory Map

The memory map of devices that respond in User Data, User Program, Supervisory Data, and Supervisory Program spaces is shown in Table 3-3.

TABLE 3-3. MVME133 Main Memory Map

ADDRESS RANGE	D31 D24 | D23 D16 | D15 D08 | D07 D00	NOTES
X000000 XX000007	Onboard ROM for first four memory cycles after reset. Onboard Dynamic RAM thereafter.	1
XX000008 XX0FFFFF	Onboard Dynamic RAM.	1
XX100000 XXEFFFFF	VMEbus	1
XXF00000 XXF1FFFF	Onboard ROM/PROM/ EPROM/EEPROM Bank 1	1,3
XXF20000 XXF3FFFF	Onboard ROM/PROM/ EPROM/EEPROM Bank 2	1,3
XXF40000 XXF5FFFF	Onboard Bank 1 repeats in this space.	1
XXF60000 XXF7FFFF	Onboard Bank 2 repeats in this space.	1
XXF80000	MSR | MFP GPIP	1,2
XXF80002	MSR | MFP AER	1,2
XXF80004	MSR | MFP DDR	1,2
XXF80006	MSR | MFP IERA	1,2
XXF80008	MSR | MFP IERB	1,2
XXF8000A	MSR | MFP IPRA	1,2
XXF8000C	MSR | MFP IPRB	1,2
XXF8000E	MSR | MFP ISRA	1,2
XXF80010	MSR | MFP ISRB	1,2
XXF80012	MSR | MFP IMRA	1,2

TABLE 3-3. MVME133 Main Memory Map (cont'd)

ADDRESS RANGE	D31 D24	D23 D16	D15 D08	D07 D00	NOTES
XXF80014	MSR	MFP IMRB			1,2
XXF80016	MSR	MFP VR			1,2
XXF80018	MSR	MFP TACR			1,2
XXF8001A	MSR	MFP TBCR			1,2
XXF8001C	MSR	MFP TCDCR			1,2
XXF8001E	MSR	MFP TADR			1,2
XXF80020	MSR	MFP TBDR			1,2
XXF80022	MSR	MFP TCDR			1,2
XXF80024	MSR	MFP TDDR			1,2
XXF80026	MSR	MFP SCR			1,2
XXF80028	MSR	MFP UCR			1,2
XXF8002A	MSR	MFP RSR			1,2
XXF8002C	MSR	MFP TSR			1,2
XXF8002E	MSR	MFP UDR			1,2
XXF80030 thru XXF9FFFF	The Status and MFP registers occur repeatedly in this space.				1
XXFA0000	SIOB RRO SIOB WRO				1
XXFA0001	SIOB RXDATA SIOB TXDATA				1
XXFA0002	SIOA RRO SIOA WRO				1
XXFA0003	SIOA RXDATA SIOA TXDATA				1

TABLE 3-3. MVME133 Main Memory Map (cont'd)

ADDRESS RANGE	D31 D24	D23 D16	D15 D08	D07 D00	NOTES
XXFA0004 XXFAFFFF	The above SIO registers appear repeatedly in this space.				1
XXFB0000	UUUU RTC00				1,4
XXFB0001	UUUU RTC01				1,4
XXFB0002	UUUU RTC02				1,4
XXFB0003	UUUU RTC03				1,4
XXFB0004	UUUU RTC04				1,4
XXFB0005	UUUU RTC05				1,4
XXFB0006	UUUU RTC06				1,4
XXFB0007	UUUU RTC07				1,4
XXFB0008	UUUU RTC08				1,4
XXFB0009	UUUU RTC09				1,4
XXFB000A	UUUU RTC10				1,4
XXFB000B	UUUU RTC11				1,4
XXFB000C	UUUU RTC12				1,4
XXFB000D	UUUU RTC13				1,4
XXFB000E	UUUU RTC14				1,4
XXFB000F	UUUU RTC15				1,4
XXFB0010 XXFBFFFF	The above RTC registers appear repeatedly in this space.				1,4

TABLE 3-3. MVME133 Main Memory Map (cont'd)

ADDRESS RANGE	D31 D24 | D23 D16 | D15 D08 | D07 D00	NOTES
XXFC0000 XXFEFFFF	VMEbus	1
XXFF0000 XXFFFFFF	VMEbus Short I/O Space	1

NOTES:
1. XX denotes "don't care". However, when [A24] is zero, the VMEbus is treated as a 32-bit data port for longword aligned transfers, and when [A24] is one, the VMEbus is treated as a 16-bit data port for all transfers. (Refer to paragraph 4.3.7 and Table 4-2.) Also, note that reading the Real-Time-Clock (RTC) with [A25] = one causes the VMEbus interrupter to activate a level three interrupt, while reading it with [A25] = 0 has no effect on the interrupter.
2. The Module Status Register (MSR) is read only. It should not be written to, because cycles that access the MSR also access the MC68901 Multi-Function Peripheral (MFP). The MSR is connected to D24-D31 and the MFP is connected to D16-D23.
3. Writes to EEPROM must always be 16-bit wide.
4. UUUU denotes four undefined bits. RTC = the Real-Time-Clock.

3.3.2 CPU Space Memory Map

The MVME133 responds to two types of CPU cycles: Coprocessor and Interrupt Acknowledge (IACK). Note that the MC68020 is capable of generating other types of CPU space cycles (using Breakpoint Acknowledge, Access level control, or MOVES instructions) which the MVME133 does not support.

3.3.2.1 Coprocessor Interface Register Map. The only coprocessor on the MVME133 module is the MC68881 Floating Point Coprocessor. The map decoder selects the MC68881 any time the MPU executes a coprocessor cycle (FC2-FC0 = %111 and A19-A16 = %0010). The recommended Coprocessor ID (Cp-ID) (bits 9-11 of the coprocessor instruction word) for the MC68881 is binary %001. However, the MVME133 selects the MC68881 regardless of what the Cp-ID is. The MC68881 registers are addressed by the A04-A01 as shown in Table 3-4.

TABLE 3-4. MC68881 Coprocessor Interface Register Map

A04-A00 (BINARY)	OFFSET (HEX)	MC68881 REGISTER D31	D16	D15	D00
%0000X	$00	Response	(Read Only)		
%0001X	$02	Control	(Write Only)		
%0010X	$04	Save	(Read Only)		
%0011X	$06	Restore	(Read/Write)		
%0100X	$08	Operation Word	(Read/Write)		
%0101X	$0A	Command	(Write Only)		
%0110X	$0C	(Reserved)			
%0111X	$0E	Condition	(Write Only)		
%100XX	$10	Operand		(Read/Write)	
%1010X	$14	Register Select	(Read Only)	(Reserved)	
%110XX	$18	Instruction Address		(Write Only)	
%111XX	$1C	Operand Address		(Read/Write)	

NOTES: 1. Write accesses to read-only registers are ignored, while read accesses to a write-only register always return all ones.

2. X means don't care.

3.3.2.2 Shared Memory Map. The user selects the address at which onboard RAM appears from the VMEbus, by using jumpers on J2 and/or reprogramming U22, PALDP. J2 selects which of the four programmable addresses in U22 is the shared memory address. (Refer to paragraph 2.3.2.) When U22 contains the default factory program, J2 selects the base addresses as given in Table 3-5.

TABLE 3-5. Shared Memory Map for Onboard RAM

J2 CONNECTIONS	SHARED MEMORY ADDRESS
1-2 AND 3-4	$000000 - $0FFFFF
1-2	$100000 - $1FFFFF
3-4	$200000 - $2FFFFF
none	$300000 - $3FFFFF

The onboard RAM responds to the VMEbus only when AM0-AM5 indicate standard, privileged or non-privileged, data or program space, and when this module (the MVME133) is not the VMEbus master.

3.3.2.3 Interrupt Acknowledge Map. The MC68020 distinguishes Interrupt Acknowledge (IACK) cycles from other CPU space cycles by placing the binary value %1111 on A19-A16. It also specifies the level that is being acknowledged using [A03] - [A01]. The interrupt handler selects which device within that level is being acknowledged. Refer to the interrupt handler description in Chapter 4 for further details.

CHAPTER 4

FUNCTIONAL DESCRIPTION

4.1 INTRODUCTION

This chapter provides the overall block diagram level description for the MVME133 module. The general description provides an overview of the module, followed by a detailed description of each section of the module. Figure 4-1 shows the simplified block diagram of the MVME133.

4.2 GENERAL DESCRIPTION

The MVME133 is a VMEbus CPU module. The MVME133 has an MC68020 MPU, 1Mb of dynamic RAM (accessible from the VMEbus), a real-time clock, a serial debug port, two multiprotocol serial ports (one with RS-232C interface and one with RS-485 interface), three 8-bit timers, four ROM sockets, an A24/D32 VMEbus interface, a simple single-level VMEbus interrupter, a seven-level VMEbus interrupt handler, and the VMEbus system controller functions.

4.2.1 Data Bus Structure

The data bus structure on the MVME133 is arranged to accommodate the 8-bit, 16-bit, 32-bit, and 16-/32-bit ports that reside on the module. Figure 4-2 shows the data bus structure of the MVME133.

4.2.2 Memory Map

The operation of the map decoder and a detailed discussion of the various memory maps in the MVME133 are given in Chapter 3. This includes the main memory map, [A24] function, [A25] function, and shared memory map.

4.2.3 MVME133 Timing

Table 4-1 and the following paragraphs give general characteristics of MVME133 module timing.

4.2.3.1 RAM Cycle Times. MC68020 accesses to onboard RAM require four MPU clock cycles (three minimum + one wait cycle), both at 12.5 MHz (MVME133) and at 16.67 MHz (MVME133-1). MC68020 RMW cycles to onboard RAM can require more than four MPU clock cycles if J5 pins 1-2 are connected. Refer to paragraph 4.3.3.1 for more details on local accesses.

4.2.3.2 VMEbus Access Time to Onboard RAM. Onboard RAM access time from the VMEbus (activation of DS0*/DS1* to activation of DTACK*) is typically 720 ns for 12.5 MHz modules (MVME133) and 540 ns for 16.67 MHz modules (MVME133-1).

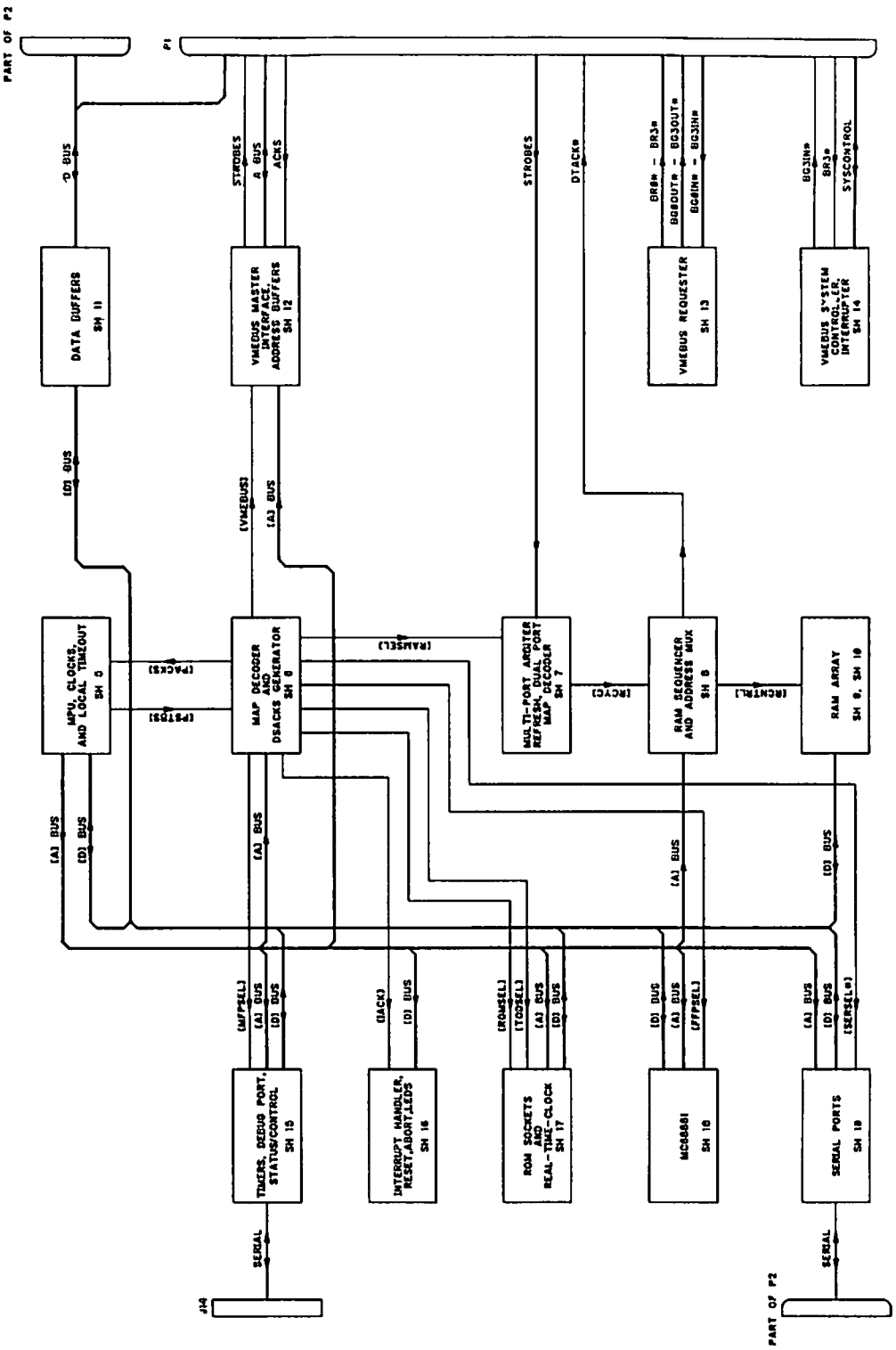


FIGURE 4-1. MVME133 Block Diagram

FIGURE 4-2. MVME133 Data Bus Structure

TABLE 4-1. MVME133 Timing

TYPE OF ACCESS	MVME133 READ	MVME133 WRITE	MVME133-1 READ	MVME133-1 WRITE	NOTES
Onboard MPU to MC68881	3 cycles	3 cycles	3 cycles	3 cycles	1,2
MPU to local ROM/ PROM/EPROM/EEPROM	6 cycles	6 cycles	6 cycles	6 cycles	1
MPU to onboard DRAM	4 cycles	4 cycles	4 cycles	4 cycles	1,3
Onboard MPU to	4+N cycles	5+N cycles	4+N cycles	5+N cycles	4,5
VMEbus to onboard DRAM	720 nsec	720 nsec	540 nsec	540 nsec	6,7

NOTES:
1. No arbitration overhead.
2. Except for read accesses to Response or Save Coprocessor Interface Register (CIR) which take 5 MPU clock cycles.
3. Except for RMW cycles where MVME133/133-1 must obtain VMEbus mastership before any RAM cycle may be started.
4. Assume MVME133/133-1 is the current VMEbus master.
5. N = (DS0*/DS1* to DTACK* time)/(MPU clock cycle time). N should be rounded up to the nearest integer.
6. DS0*/DS1* activated to DTACK* time.
7. Typical values. Actual values may be greater or less depending on the state of the MVME133.

4.2.3.3 ROM/PROM/EPROM/EEPROM Cycle Times. All ROM/PROM/EPROM/EEPROM accesses require three wait cycles to complete.

4.2.3.4 VMEbus Cycle Times. The following formula assumes that the MVME133 is the current VMEbus master and that all slaves have released DTACK* and BERR*. The time from the activation of DS0*/DS1* to the activation of DTACK* is ta, the number of wait cycles incurred is N, and T is the MPU clock period. All numbers are typical. N must always be rounded up to the next integer.

For read cycles	N = 1 + [ta/T]
For write cycles	N = 2 + [ta/T]

The following formula assumes that the MVME133 is not the current VMEbus master, but that it is the system controller. Also, it assumes that all previous slaves have released DTACK* and/or BERR* when the MVME133 receives VMEbus mastership. The delay from BR3* low (driven by MVME133) to BBSY* high and AS* high is tr.

For read cycles	N = 1 + [(ta + tr + 170 ns)/T]	typical
For write cycles	N = 3 + [(ta + tr + 170 ns)/T]	typical

The following formula assumes that the MVME133 is not the current VMEbus master, and it is not the system controller. Also, it assumes that all previous slaves have released DTACK* and/or BERR* when the MVME133 receives VMEbus mastership. The delay from BRX* low (driven by MVME133) to BGXIN* low and AS* high is tg.

For read cycles	N = 1 + [(ta + tg + 150 ns)/T]	typical
For write cycles	N = 3 + [(ta + tg + 150 ns)/T]	typical

4.2.3.5 VMEbus Arbitration Time. When the MVME133 is configured as system controller, the delay from BBSY* high and BR3* low to BG3OUT* low is 150 ns typical, and 230 ns maximum if the MVME133 is not requesting VMEbus mastership.

When the MVME133 is not configured as system controller, the delay from BGXIN* low to BGXOUT* low is 130 ns typical, and 180 ns maximum if the MVME133 is not requesting VMEbus mastership.

4.2.4 System Considerations

4.2.4.1 Memory Sizing. The MVME133 accesses the VMEbus as a longword port (32-bit data) when the MPU executes a longword aligned cycle with [A24] = 0. The MVME133 accesses the VMEbus as a word port (16-bit data) when the MPU executes a cycle other than longword aligned with [A24] = 0.

This has impact on memory sizing routines. In order to size the longword portion of VMEbus memory, the sizing routine should use longword aligned transfers with [A24] = 0. In order to size the word portion of VMEbus memory, it should use something other than longword aligned transfers with [A24] = 0.

Finally, to isolate the word only portion of the VMEbus memory, it should subtract the portion that responds to longword aligned cycles from the portion that doesn't. For example, if locations $100000 - $4FFFFC respond to longword aligned accesses, and locations $100000 - $7FFFFE respond to word accesses, then locations $500000 - $7FFFFE are word only locations and should be accessed as bytes or words only, or they should be accessed in the range $1500000 - $17FFFFF.

4.2.4.2 Sources of BERR*. There are three sources of bus error exceptions on the MVME133. They are: Local Bus Time-out, VMEbus [QVBERR*], and RMW-LOCK.

4

Local bus time-out occurs whenever an MPU access does not complete within 20 ms. If the system is configured properly, this should only happen if: software accesses a non-existent location within the onboard range, or something prevents this module from becoming the VMEbus master.

[QVBERR*] occurs when the BERR* signal line is activated on the VMEbus while the MVME133 is the VMEbus master. VMEbus BERR* should occur only if: an initialization routine samples to see if a device is present on the VMEbus, software accesses a non-existent device within the VMEbus range, software tries to access a device on the VMEbus incorrectly (such as driving LWORD* low to a 16-bit module), a hardware error occurs on the VMEbus, or a VMEbus slave reports an access error (such as parity error).

RMW-LOCK occurs when there is a VMEbus deadlock during an MPU RMW cycle. As noted in paragraph 4.3.3.2, whenever a VMEbus deadlock occurs, the multiport arbiter breaks the lock by activating both [BERR*] and [HALT*] at the same time. This sequence indicates to the MPU that it should abort the current cycle. Once the MPU aborts the current cycle, it relinquishes local bus mastership to the VMEbus, which in turn executes a RAM cycle. If the MC68020 happens to be executing an RMW cycle when the VMEbus deadlock occurs, it will not relinquish local bus mastership until it completes all portions of the RMW cycle. However, the RMW cycle cannot complete until the MVME133 obtains VMEbus mastership, and the MVME133 cannot obtain VMEbus mastership because another VMEbus module is master and is waiting at the MVME133 shared memory. This lockup condition is called RMW-LOCK.

When the multiport arbiter detects the RMW-LOCK condition, instead of pulling [BERR*] and [HALT*] low to cause a retry, it activates only [BERR*], which causes the MPU to abort the RMW and relinquish the local bus, thus breaking the RMW-LOCK condition, but causing a bus error exception.

If the system is constructed so that RMW cycles will never be generated to the VMEbus range, and that no multiple address RMW cycles can happen to the onboard RAM from the VMEbus side, then header J5 can be jumpered so that RMW cycles to onboard RAM do not require VMEbus mastership. This removes RMW-LOCK bus errors.

Because different conditions can cause bus error exceptions, the software must be able to distinguish the source. To aid in this, the MVME133 provides two status bits in the MC68901 MFP. They are [LTO] and [QVBERR*]. RMW-LOCK is not provided as a status bit, but software can determine that it has happened if a bus error exception occurs and neither [LTO] nor [QVBERR*] is set.

Generally, the bus error handler can interrogate [LTO] and [QVBERR*] and proceed with the result. However, an interrupt can happen during the execution of the bus error handler (before an instruction can write to the status register to raise the interrupt mask). If the interrupt service routine causes a second bus error, the status that indicates the source of the first bus error may be lost. The software must be aware of this and must be written to deal with it.

4.2.4.3 Use of RMW Instructions. The MC68020 RMW Instructions are TAS, CAS, and CAS2. These instructions cause indivisible cycle sequences to occur on the MC68020 local bus. TAS and single address CAS perform one read and then one write to the same address. Multiple address CAS and CAS2 perform reads and writes to multiple addresses. The VMEbus defines single address indivisible cycles as READ-MODIFY-WRITE cycles. The VMEbus does not define multiple address indivisible cycles. A scheme has been devised to allow indivisible multiple address cycles on the VMEbus. It is not part of the VMEbus specification. It is implemented on the MVME133 when J5 pins 1-2 are connected. The scheme has the following rules:

1. Locations that are accessed by multiple address indivisible cycles are called Multiple Address Interlock (MAI) locations.

2. All devices that access MAI locations must use indivisible cycle instructions (that is, TAS, CAS, or CAS2 of MC68020).

3. Any device that executes an indivisible cycle instruction must obtain VMEbus mastership before executing the first cycle of the instruction. In addition, it must retain VMEbus mastership until it has completed the last cycle of the instruction.

Rule number 1 is a definition, rule number 2 is a software requirement, and rule number 3 is taken care of automatically by the MVME133 requester.

The MVME133 does not support the above scheme when J5 pins 1-2 are not connected. In this configuration, the MVME133 does not obtain VMEbus mastership before executing indivisible cycle instructions. In fact, J5 pins 1-2 must only be disconnected if the software never executes RMW cycles within the VMEbus range. The advantage of using this jumper option is that, when it is used properly, RMW-LOCKs never occur.

4.3 DETAILED DESCRIPTION

The following paragraphs describe in detail the theory of operation for the MVME133 module. During this discussion, sheets referenced belong to the schematic diagram for the MVME133 module (see Figure 5-2).

4.3.1 MPU, Clocks, and Local Time-out (Sheet 5)

The MVME133 runs with a 12.5 MHz MC68020, but the MVME133-1 runs with a 16.67 MHz MC68020. (Known problems in the various masks of the MPU are described in Appendix A.) The two models of the module also differ in the frequency of the master crystal oscillator Y3, 25 MHz and 33.33 MHz, respectively.

The local bus time-out generator aborts any cycle that does not complete within 21 to 24 ms for the MVME133, or within 16 to 18 ms for the MVME133-1, by driving active low [BERR*] to the MPU.

4.3.2 Map Decoder and DSACKs Generator (Sheet 6)

The operation of the map decoder and a detailed discussion of the various memory maps in the MVME133 are given in Chapter 3. This includes the main memory map, [A24] function, [A25] function, and shared memory map.

4.3.3 Multiport Arbiter, Refresh, and Shared Memory Map Decoder (Sheet 7)

The 1Mb of onboard dynamic RAM is accessible by the onboard MPU, the refresh circuitry, and the VMEbus. Each of these three devices requests and is granted the RAM by the multiport arbiter.

Because the local address and data busses are used to access the onboard dynamic RAM, any device that uses the RAM must become the local bus master first. The MPU arbitration logic (BR*, BG*, BGACK*) is utilized by the multiport arbiter to transfer local bus mastership from the current master to the next. During normal operation, the MPU is the local bus master. When either the VMEbus or the refresh circuitry requests use of the RAM, the multiport arbiter activates [BR*] to the MPU. The MPU responds by activating [BG*], finishing its current cycle (if one is in progress), and giving up local bus mastership. At this point, the multiport arbiter activates [BGACK*], deactivates [BR*], and grants local bus mastership to the highest priority requesting device. (Refresh has higher priority than VMEbus.) The granted device executes one RAM cycle (read, write, or refresh) and relinquishes local bus mastership. If another device is requesting local bus mastership at this time, the multiport arbiter grants it to that device. Otherwise, it deactivates [BGACK*] and the MPU resumes local bus mastership.

4.3.3.1 Local Accesses. Once the MPU starts a RAM cycle, it does not wait for the multiport arbiter because it would already be the local bus master. However, there is one exception. When the MPU begins a RMW cycle to onboard RAM and J5 pins 1-2 are connected, the RAM sequencer requires that the VMEbus requester obtain VMEbus mastership before the first access of the RMW cycle can occur. Refer to paragraph 4.2.4.3, use of RMW instructions, for further details.

MPU to RAM access cycles complete in four MPU cycles (3 + 1 wait). The RAM array appears to the MPU as a 32-bit port.

4.3.3.2 VMEbus Accesses. When DS0* and/or DS1* go from false to true on the VMEbus, the MVME133 shared memory map decoder looks at the value on AM0-AM5 and at the address on A20-A23. If they are in the shared memory address range and if this module is not the current VMEbus master, then the shared memory map decoder requests the multiport arbiter for local bus mastership. (Refer to Chapter 3 for details of the shared memory map.) Once the multiport arbiter has granted local bus mastership, a RAM read or write cycle happens. When the RAM sequencer activates the DTACK* signal on the VMEbus, the multiport arbiter grants local bus mastership to the refresh circuitry if a refresh request is pending. Otherwise, it returns the local bus mastership to the MPU at the end of the VMEbus shared memory cycle. However, if the VMEbus master is executing a RMW cycle to the RAM, then the multiport arbiter does not restore local bus mastership to the MPU until both the read and write cycles are completed.

If the MPU is the current local bus master and is executing a cycle that requires the VMEbus when the shared memory map decoder requests local bus mastership, then a VMEbus deadlock condition occurs. To break this VMEbus deadlock condition, the multiport arbiter signals a retry to the MPU by activating both BERR* and HALT*. The MPU responds by aborting the current cycle, at which time it relinquishes local bus mastership so that the multiport arbiter can grant it to the VMEbus. Once the VMEbus has finished with the RAM, the multiport arbiter returns local bus mastership to the MPU. The MPU then retries the aborted cycle.

However, on some masks of the MC68020 (including 0A45J), the above sequence will not work if the MPU is executing an RMW cycle. Instead of indicating a retry to the MPU, the multiport arbiter must activate [BERR*] to break the lock-up condition. This creates some software implications which are covered in paragraph 4.2.4.2, sources of BERR*. Also refer to Appendix A for details of the MC68020 mask errors.

The RAM appears to the VMEbus as a 16-bit port for transfers with LWORD* deactivated, and as a 32-bit port for transfers with LWORD* activated. The MVME133 supports misaligned transfers to RAM from the VMEbus.

4.3.3.3 Refresh. The dynamic RAMs require that each of their 256 rows be refreshed once every 4 ms. To accomplish this, the refresh times requests the multiport arbiter for a RAM refresh cycle every 15 us. When the multiport arbiter issues a refresh grant, the RAM sequencer performs a CAS-before-RAS refresh cycle to the RAM.

Normally, when any device requests the use of the RAM and the MPU is the current local bus master, the multiport arbiter waits for the MPU to relinquish bus mastership before giving a grant to the new device. However, the multiport arbiter may issue a refresh grant without waiting for the MPU to release the local bus if it detects a refresh request when the MPU is executing a cycle to devices other than onboard RAM or when the MPU is waiting for VMEbus mastership before RMW to the local RAM. This allows the RAMs to be properly refreshed during long periods when the MVME133 is requesting VMEbus mastership, or when it is accessing very slow or non-existent devices.

4.3.4 RAM Sequencer and Address Multiplexers (Sheet 8)

MPU accesses to onboard dynamic RAM take four MPU clock periods. VMEbus access time is typically 720 ns for MVME133 and 540 ns for MVME133-1. Refer to local and VMEbus accesses descriptions in paragraphs 4.3.3.1 and 4.3.3.2.

4.3.5 RAM Array (Sheets 9 and 10)

The onboard dynamic RAM uses thirty-two 256K x 1 dynamic RAM ZIPs (zigzag-inline-packages), making a total of 1Mb of local RAM. It is accessible by the onboard MPU, the refresh circuitry, and the VMEbus, as described in paragraph 4.3.3.

4

4.3.6 VMEbus Data Buffers (Sheet 11)

Refer to paragraph 4.2.1 and Figure 4-2, which describe data bus structure.

4.3.7 VMEbus Master Interface and Address Buffers (Sheet 12)

The MVME133 has an A24/D32 master interface. Whenever the MVME133 executes a VMEbus cycle (read, write, or interrupt acknowledge) and its VMEbus requester has obtained VMEbus mastership, it drives the VME address bus with its local address bus and the VMEbus address modifiers with:

AM0 = [FC0],
AM1 = [FC1],
AM2 = [FC2],
AM3 = high,
AM4 = low in short I/O range or high in standard range,
AM5 = high.

(Refer to paragraph 3.3 and Table 3-2 for cycle types and responding devices.)

The MVME133 also activates IACK* if this is an interrupt acknowledge cycle. The MVME133 drives LWORD* according to Table 4-2. (Refer also to paragraph 3.3.1 and Table 3-3.)

TABLE 4-2. MVME133 using [A24] to Set VMEbus Data Width

MPU CYCLE TYPE	LWORD*	DATA WIDTH
Not longword aligned	high	VMEbus = 16-bit data port
Longword aligned and [A24] = 0	low	VMEbus = 32-bit data port
Longword aligned and [A24] = 1	high	VMEbus = 16-bit data port

Once A01-A23, AM0-AM5, IACK*, and LWORD* are driven to their appropriate levels on the VMEbus, the MVME133 activates AS*. If it is a read cycle, the MVME133 drives WRITE* high and enables D00-D15 onto [D16]-[D31] if LWORD* is high, or D00-D15 onto [D00]-[D15] and D16-D31 onto [D16]-[D31] if LWORD* is low. Then it activates DS0* and/or DS1* appropriately. If the cycle is a write cycle, then the MVME133 drives WRITE* low and enables [D16]-[D31] onto D00-D15 if LWORD* is high, or [D00]-[D15] onto D00-D15 and [D16]-[D31] onto D16-D31 if LWORD* is low. Then, after the appropriate delay, the MVME133 drives DS0*/DS1* low. (Refer also to the data bus structure in paragraph 4.2.1 and Figure 4-2.)

If the cycle terminates normally (with DTACK* driven to low), then the MVME133 DSACKs generator circuit activates [DSACK1*] and [DSACK0*] if LWORD* is low, or it drives only [DSACK1*] to low if LWORD* is high. If the cycle terminates with an error (with BERR* driven to low), then the MVME133 BERR generator circuit activates [BERR*] to the MPU. Once the handshake has occurred (either DTACK* or BERR*), the MPU removes [AS*], [DS*] and the MVME133 completes the cycle by disabling the data bus drivers and removing DS0*/DS1* and AS*.

The above sequence is altered slightly when the MPU executes RMW cycles. When the MPU starts an RMW cycle, the VMEbus master interface checks to see if it is a single or multiple address RMW by examining SIZ1 and SIZ0. If it is a multiple address RMW cycle, the VMEbus master interface operates normally. (Operation of the VMEbus requester is altered as shown in paragraph 4.3.8.) If it is a single address RMW cycle, then the VMEbus master interface leaves AS* active during the entire time from the beginning of the RMW read cycle to the end of the RMW write cycle. This makes a single address RMW cycle from the MPU appear on the VMEbus as a VMEbus-defined read-modify-write cycle.

4.3.8 VMEbus Requester (Sheet 13)

The VMEbus requester is used to obtain and relinquish mastership of the VMEbus. It can request VMEbus mastership on any one of the four request levels depending on the configurations of J3 and J4, and it fully supports the bus-grant daisy-chain. It requests mastership of the VMEbus any time the MVME133 module is not the current VMEbus master and the map decoder or the interrupt handler indicates that the MPU is executing a cycle that requires the VMEbus. It also requests mastership of the VMEbus when the MVME133 module is not the current VMEbus master and the MPU is starting to execute an RMW sequence to the onboard RAM with J5 pins 1-2 connected.

The VMEbus requester operates in the Early Release-On-Request (ROR) mode only. Once the MVME133 has obtained VMEbus mastership, the VMEbus requester maintains mastership until another VMEbus module requests VMEbus mastership and then only if an RMW sequence is not in process. It releases the VMEbus in one of two different ways, depending on the state of the MVME133 at the time.

If the MVME133 is in the middle of a VMEbus cycle (AS* already activated) when the VMEbus requester decides to relinquish VMEbus mastership, it releases BBSY* immediately. The transfer of VMEbus mastership occurs when the VMEbus master interface (refer to paragraph 4.3.7) deactivates and releases AS*.

If the MVME133 is not in the middle of a VMEbus cycle when the VMEbus requester decides to relinquish VMEbus mastership, the VMEbus master interface (refer to paragraph 4.3.7) releases all of the VMEbus lines, after which the VMEbus requester releases BBSY* to complete the transfer of VMEbus mastership.

4.3.9 VMEbus System Controller and Interrupter (Sheet 14)

4.3.9.1 System Controller and SYSRESET*. The system controller implements the following functions: level-three VMEbus arbiter, IACK* daisy-chain driver, global SYSCLK (16 MHz), and global VMEbus time-out that drives BERR*. All of the MVME133 system controller functions and the SYSRESET* driver are enabled/disabled by header J1. The position of the jumper on J1 appears as the SYSCON bit in the Module Status Register. (Refer to paragraph 4.3.10.3.)

The global bus time-out circuit starts the timing upon detecting the activation of DS0* and/or DS1*. If it reaches the time-out count before DS0* and DS1* return to high, it drives BERR*. The time-out count can be set for 82 to 92 us for MVME133 (61 to 69 us for MVME133-1) (J8 pins 1-2 connected), or for infinity (J8 pins 1-2 not connected).

The SYSCLK driver drives a periodic 16 MHz clock onto the SYSCLK line on the VMEbus if the system controller on the MVME133 is enabled.

The level-three arbiter is designed to meet the VMEbus specification requirements. In addition, it is designed to ignore possibly erroneous signals on BBSY* and to re-arbitrate if no VMEbus master responds to a grant within 1 ms.

The IACK* daisy-chain driver is designed to meet the VMEbus specification requirements.

Even though SYSRESET* is not a VMEbus system controller function, the MVME133 enables/disables its SYSRESET* function at the same time that it enables/disables its system controller functions. When configured as the system controller, the MVME133 drives the SYSRESET* signal line to low when the front panel RESET switch is depressed, when a watchdog time-out occurs, when the RRESET* line is activated, or when a power-up occurs.

NOTE

The MVME133 does not fully implement the SYSRESET* timing of a VMEbus power monitor.

4.3.9.2 VMEbus Single-Level Interrupter. The VMEbus single-level interrupter generates interrupt requests on IRQ3*. It provides the value $FF as its status ID byte. It is an 8-bit interrupter and consequently responds to all sizes of interrupt acknowledge cycles. The interrupter drives IRQ3* whenever the MPU reads the real-time clock with [A25] = 1. The state of the interrupter is reflected as the [OIRQ] bit of the Multi-Function Peripheral (MFP) GPIO port. (Refer to paragraph 4.3.10.3.) A typical sequence for interrupting is as follows:

a. Verify that the [OIRQ] bit is 0.

b. Set up the MFP to interrupt the MPU when [OIRQ] transitions from 1 to 0.

c. Read any register in the real-time clock with [A25] = 1 (for example, $2FB0000).

d. Continue with other processing until the [OIRQ] interrupt occurs.

e. The VMEbus interrupt has now been acknowledged.

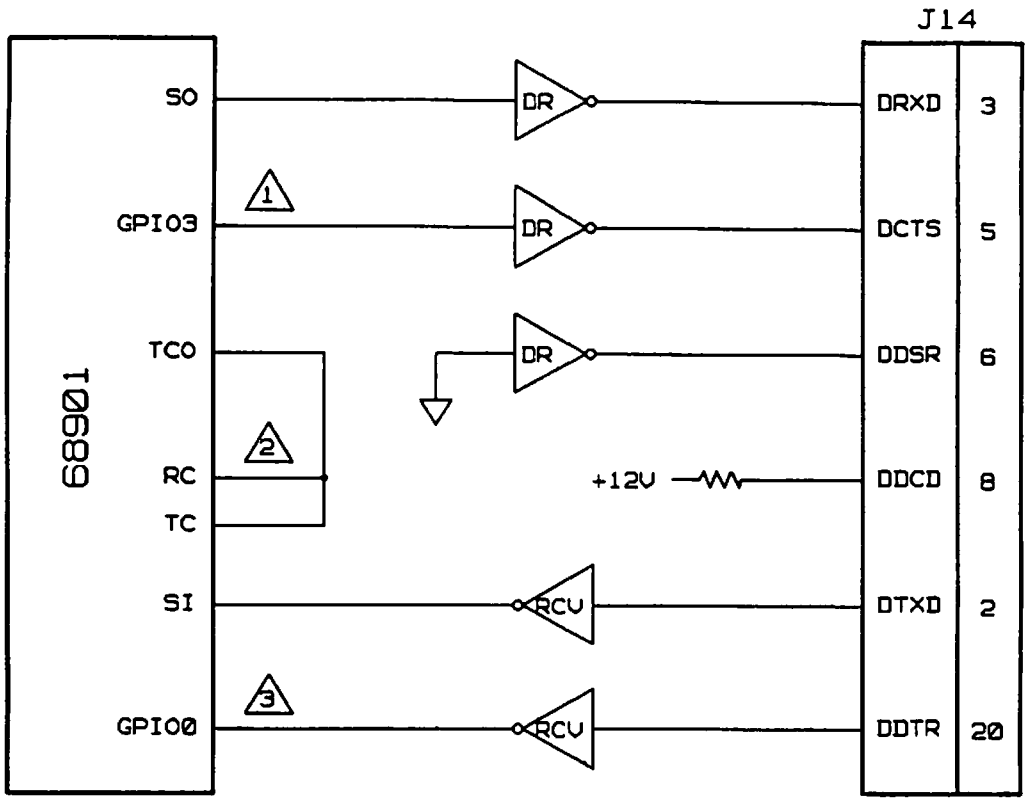
f. Continue with normal processing.

4.3.10 Timers, Debug Port, and Status/Control (Sheet 15)

The MVME133 uses the MFP MC68901 chip for its front panel debug port, its tick timers, its watchdog timer, and the status and control information. The MC68901 has the ability to interrupt the MPU on level five. (Refer to paragraph 4.3.11.1.) Its interrupt sources are from the timers, the debug port, and the GPIO (status) bits.

4.3.10.1 Debug Port. The front panel debug port (through J14) is a minimal implementation of a to-terminal-only RS-232C serial port. (Refer to Appendix B for a discussion of RS-232C signals.) It uses DRXD as its transmit data output and DTXD as its receive data input. It pulls DDSR and DDCD to +12 Vdc, controls DCTS with a software bit, and monitors DDTR with another software bit, providing minimal flow control. See Figure 4-3.

The baud rate generator for the serial port is timer C of the MC68901 MFP. The XTAL input to the MC68901 is 1.23 MHz. The baud rates supported are programmed as shown in Table 4-3.



NOTE:
1. GPIO3 PROGRAMMED AS OUTPUT.
2. TC AND RC ARE DRIVEN BY OUTPUT OF TIMER C.
3. GPIO0 PROGRAMMED AS INPUT.

FIGURE 4-3. MVME133 Debug Port (To Terminal Only)

TABLE 4-3. Debug Port Baud Rates

BAUD RATE	THEORETICAL 16 x CLOCK FREQUENCY	XTAL = 1.23 MHz PRE-SCALE	TIMER C COUNT	ACTUAL FREQUENCY	PERCENT ERROR
9600	153600	4	1	153846	0.16
4800	76800	4	2	76923	0.16
2400	38400	4	4	38462	0.16
1200	19200	4	8	19230	0.16
600	9600	4	$10	9615	0.16
300	4800	4	$20	4808	0.16
110	1760	4	$57	1768	0.47

4

4.3.10.2 Timers. There are four timers in the MC68901 MFP. They are assigned as follows:

TIMER C - Baud rate generator for the serial port.

TIMER A - Software tick timer. The tick timer is capable of generating a periodic interrupt.

TIMER B - Tick timer overflow/watchdog time-out. The watchdog timer is capable of generating a local/system reset when timer B output is high after a programmable interval.

TIMER D - Delay mode only. Unassigned by hardware.

4.3.10.3 Status and Control. The MC68901 MFP has eight General Purpose I/O (GPIO) pins. The MVME133 uses five of these pins as status inputs and three of them as control outputs. After a reset, the MC68901 makes all of the GPIO pins inputs. Therefore, after each reset, the software should initialize the control bits and make them outputs. MVME133 hardware defaults the control lines to high when they are not programmed as outputs. The assignment of pins GPIO0-GPIO7 is as follows:

GPIO0 - Input connected to [DDTR*]. General Purpose I/O Interrupt Port (GPIP) bit 0 is 0 when DDTR is high on the debug RS-232C interface. GPIP bit 0 is 1 when DDTR is low on the debug RS-232C interface. Bit 0 of the Interrupt Pending Register B (IPRB) may be initialized by software to detect the transitions of DDTR.

GPIO1 - Input connected to [QVBERR*]. When this module executes a cycle on the VMEbus, if BERR* goes low, [QVBERR*] goes low until the end of the cycle. Because [QVBERR*] always goes back high at the end of error cycle,

GPIP bit 1 always reads as 1 by the time software reads it. However, software may initialize IPRB bit 1 to latch the fact that [QVBERR*] has pulsed low. IPRB bit 1 may then be read and cleared by software.

GPIO2 - Input connected to [LTO]. When the MPU performs an access which does not complete within 20 ms, the MVME133 activates [BERR*] to the MPU. [LTO] goes high during that cycle and returns to low after the end of the cycle. Because [LTO] always goes back low at the end of the time-out cycle, GPIP bit 2 always reads as 0 by the time software reads it. However, software may initialize IPRB bit 2 to latch the fact that [LTO] has pulsed high. IPRB bit 2 then may be read and cleared by software.

GPIO3 - Control connected [DCTS*]. When bit 3 of GPIP is 0, DCTS is high on the debug RS-232C interface. When bit 3 of GPIP is 1 or when it is programmed as input, DCTS is low on the debug RS-232C interface.

GPIO4 - Control connected to [IE*]. When bit 4 of GPIP is 1 or when it is programmed as input, no interrupt requests reach the MPU. When bit 4 of GPIP is 0, interrupt requests may reach the MPU.

GPIO5 - Control connected to [BRDFAIL]. When bit 5 of GPIP is 1 or when it is programmed as input, the FAIL indicator is lit. Also, if the MVME133 is not the system controller, it drives the SYSFAIL* line on the VMEbus low during this time. When bit 5 of GPIP is 0, the SYSFAIL* line is not driven by the MVME133 and the FAIL indicator is not lit.

GPIO6 - Input connected to [OIRQ]. When [OIRQ] is 1, the MVME133 is driving IRQ3* on the VMEbus. When [OIRQ] is 0, it is not. [OIRQ] transitions from 0 to 1 when the MVME133 reads the real-time clock with [A25] high. [OIRQ] transitions from 1 to 0 when the MVME133 IRQ3* is acknowledged on the VMEbus. Transitions on [OIRQ] may be detected and latched in Interrupt Pending Register A (IPRA) bit 6. [OIRQ] is cleared by reset.

GPIO7 - Input connected to [SYSFAIL]. When SYSFAIL* is low, bit 7 of the GPIP is 1. When SYSFAIL* is high, bit 7 of the GPIP is 0. Transitions on SYSFAIL* may be detected and latched in IPRA bit 7.

In addition to the status and control bits that are implemented in the MC68901 MFP, the MVME133 has eight status bits that are read only, have no latching mechanism, and cause no interrupts (with one exception). These bits are called the Module Status Register (MSR). Because of hardware savings, the MSR and the MC68901 are grouped together and appear as a 16-bit word port to the MPU. (Refer to paragraph 3.3.1 and Table 3-3.) Therefore, it is important to note that even though the MSR ignores all write accesses, a write to the MSR will affect the MC68901.

The MC68901 appears on the lower byte of the word, and the MSR appears on the upper byte. The bit assignments for the MSR are:

BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
ACFAIL	SYSCON	PWRUP*	SRBIT4	SRBIT3	SRBIT2	SRBIT1	SRBIT0

ACFAIL - When ACFAIL* is low, this bit is 1. When ACFAIL* is high, it is 0. [ACFAIL] is also an input to the interrupt handler.

SYSCON - If this module is the system controller, this bit is 1. When it is not the system controller, this bit is 0.

PWRUP* - This bit is set to 0 only by Power-Up reset when power is first applied to the MVME133. (Refer to paragraph 4.3.11.2.) A read cycle to the real-time clock resets this bit to a 1.

SRBIT4 - This bit is 0 when J15 pins 9-10 are connected, and is 1 when they are open.

SRBIT3 - This bit is 0 when J15 pins 7-8 are connected, and is 1 when they are open.

SRBIT2 - This bit is 0 when J15 pins 5-6 are connected, and is 1 when they are open.

SRBIT1 - This bit is 0 when J15 pins 3-4 are connected, and is 1 when they are open.

SRBIT0 - This bit is 0 when J15 pins 1-2 are connected, and is 1 when they are open.

4.3.11 Interrupt Handler, Reset, Abort, and Status LEDs (Sheet 16)

4.3.11.1 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*, VMEbus SYSFAIL*, and the ABORT switch.

All VMEbus interrupts are enabled/disabled using header J12, ABORT is enabled/disabled using J10, and all interrupts that go through the MC68901 are enabled in the MC68901. Also, Z8530 and the real-time clock may be enabled/disabled individually. All interrupts are disabled when the [IE*] bit is 1.

When the MPU initiates an interrupt acknowledge cycle, the interrupt handler determines the acknowledge level by examining [A01] - [A03]. It then decides which device the cycle is for. Finally, it activates [AVEC*] to indicate to the MPU to generate the interrupt vector internally if the acknowledge cycle was for VMEbus ACFAIL*, ABORT switch, or the real-time clock. If the acknowledge cycle is for the Z8530, the MC68901 MFP, or the VMEbus, then it indicates a vector fetch cycle to the appropriate device.

If both onboard and VMEbus interrupts are activated on the acknowledge level, the interrupt handler acknowledges the onboard interrupt. Note also that VMEbus ACFAIL* and ABORT switch are both on level seven and have the same interrupt offset vector. Therefore, the software handler routine for this autovector must interrogate the [ACFAIL] bit in the MSR to determine the actual interrupt source.

Table 4-4 summarizes all the interrupt sources on the MVME133 (in ascending order of priority) and the associated interrupt vectors.

TABLE 4-4. Interrupt Sources and Vectors

INTERRUPT SOURCE	PATH	VECTOR PASSED	VECTOR OFFSET	LEVEL
VMEbus IRQ1*	Direct	From interrupting VMEbus slave	4 x vector	1
VMEbus IRQ2*	Direct	From interrupting VMEbus slave	4 x vector	2
VMEbus IRQ3*	Direct	From interrupting VMEbus slave	4 x vector	3
VMEbus IRQ4*	Direct	From interrupting VMEbus slave	4 x vector	4
MM58274A Real-Time Clock	Direct	NONE	$70	4
VMEbus IRQ5*	Direct	From interrupting VMEbus slave	4 x vector	5
DDTR*	MC68901 GPIO0	Refer to MC68901 data sheet	4 x vector	5
VMEbus BERR*	MC68901 GPIO1	Refer to MC68901 data sheet	4 x vector	5
LTO Local bus time-out	MC68901 GPIO2	Refer to MC68901 data sheet	4 x vector	5
MC68901 Timer D	MC68901	Refer to MC68901 data sheet	4 x vector	5
MC68901 Timer C	MC68901	Refer to MC68901 data sheet	4 x vector	5
MC68901 Timer B	MC68901	Refer to MC68901 data sheet	4 x vector	5
Transmit Error (Serial)	MC68901	Refer to MC68901 data sheet	4 x vector	5
Transmit Buffer Empty (Serial)	MC68901	Refer to MC68901 data sheet	4 x vector	5
Receive Error (Serial)	MC68901	Refer to MC68901 data sheet	4 x vector	5
Receive Buffer Full (Serial)	MC68901	Refer to MC68901 data sheet	4 x vector	5

TABLE 4-4. Interrupt Sources and Vectors (cont'd)

INTERRUPT SOURCE	PATH	VECTOR PASSED	VECTOR OFFSET	LEVEL
MC68901 Timer A	MC68901	Refer to MC68901 data sheet	4 x vector	5
OIRQ (Bus Interrupter)	MC68901 GPIO6	Refer to MC68901 data sheet	4 x vector	5
VMEbus SYSFAIL	MC68901 GPIO7	Refer to MC68901 data sheet	4 x vector	5
VMEbus IRQ6*	Direct	From interrupting VMEbus slave	4 x vector	6
Z8530 Serial Ports	Direct	Refer to Z8530 data sheet	4 x vector	6
VMEbus IRQ7*	Direct	From interrupting VMEbus slave	4 x vector	7
VMEbus ACFAIL*	Direct	NONE	$7C	7
ABORT*	Direct	NONE	$7C	7

4.3.11.2 Reset. The six sources of reset on the MVME133 are as follows:

a. SYSRESET* - Resets all onboard devices.

b. Power-Up Reset - Resets all onboard devices and drives SYSRESET* if this module is system controller.

c. Front Panel RESET Switch - Resets all onboard devices and drives SYSRESET* if this module is system controller.

d. Watchdog Time-out - Resets all onboard devices and drives SYSRESET* if this module is system controller.

e. RRESET* - Resets all onboard devices and drives SYSRESET* if this module is system controller.

f. MC68020 RESET Instruction - Resets only the Z8530 and the MC68901 MFP.

All resets wait until the MPU is between cycles before starting.

4.3.11.3 ABORT and RESET Switches. Refer to Chapter 3.

4.3.11.4 FAIL, HALT, and RUN Indicators. Refer to Chapter 3.

4.3.12 ROM/PROM/EPROM/EEPROM Sockets and Real-Time Clock (Sheet 17)

4.3.12.1 ROM/PROM/EPROM/EEPROM. The MVME133 has four 28-pin ROM/PROM/EPROM/EEPROM sockets that are organized as two banks with two sockets per bank. Each bank appears as a 16-bit word port to the MPU and can be separately configured for 8K x 8, 16K x 8, 32K x 8, or 64K x 8 ROM/PROM/EPROMs; or for 2K x 8, 8K x 8, or 32K x 8 EEPROMs. When a bank is configured for EEPROM, writes to that bank must always be 16-bit wide.

There are several different algorithms for erasing/writing to EEPROMs, depending on the manufacturer. The MVME133 supports only those devices which have a "static RAM" compatible erase/write mechanism. The MVME133 connects pin 1 of both devices in bank 1 to the DCDA input of the Z8530, and pin 1 of both devices in bank 2 to the CTSA input of the Z8530. DCDA and CTSA can be used as general purpose status inputs for monitoring those devices which provide RDY/BSY* status on pin 1. Note that the MVME133 requires that the EEPROMs must allow wired-OR on the RDY/BSY* pin.

Figure 4-4 shows the definitions of the ROM/PROM/EPROM/EEPROM socket pins, depending upon the configuration used.

CONFIGURATION 1	2	3	4	5			CONFIGURATION 5	4	3	2	1	NOTES
+5V	+5V	A15	RDY/BSY	A14	1	28	+5V	+5V	+5V	+5V	+5V	SOCKET A14 = BOARD [A15].
A12	A12	A12	A12	A12	2	27	WE*	WE*	A14	A14	VIH	
A7	A7	A7	A7	A7	3	26	A13	A13	A13	A13	A13	SOCKET A15 = BOARD [A16].
A6	A6	A6	A6	A6	4	25	A8	A8	A8	A8	A8	
A5	A5	A5	A5	A5	5	24	A9	A9	A9	A9	A9	
A4	A4	A4	A4	A4	6	23	A11	A11	A11	A11	A11	SEE SCHEMATIC
A3	A3	A3	A3	A3	7	22	OE*	OE*	OE*	OE*	OE*	DIAGRAM, FIGURE
A2	A2	A2	A2	A2	8	21	A10	A10	A10	A10	A10	5-2.
A1	A1	A1	A1	A1	9	20	CE*	CE*	CE*	CE*	CE*	
A0	A0	A0	A0	A0	10	19	D7	D7	D7	D7	D7	REFER TO PARA-
D0	D0	D0	D0	D0	11	18	D6	D6	D6	D6	D6	GRAPH 2.3.5.
D1	D1	D1	D1	D1	12	17	D5	D5	D5	D5	D5	
D2	D2	D2	D2	D2	13	16	D4	D4	D4	D4	D4	MVME133BUG USES
GND	GND	GND	GND	GND	14	15	D3	D3	D3	D3	D3	64K x 8 EPROMs.

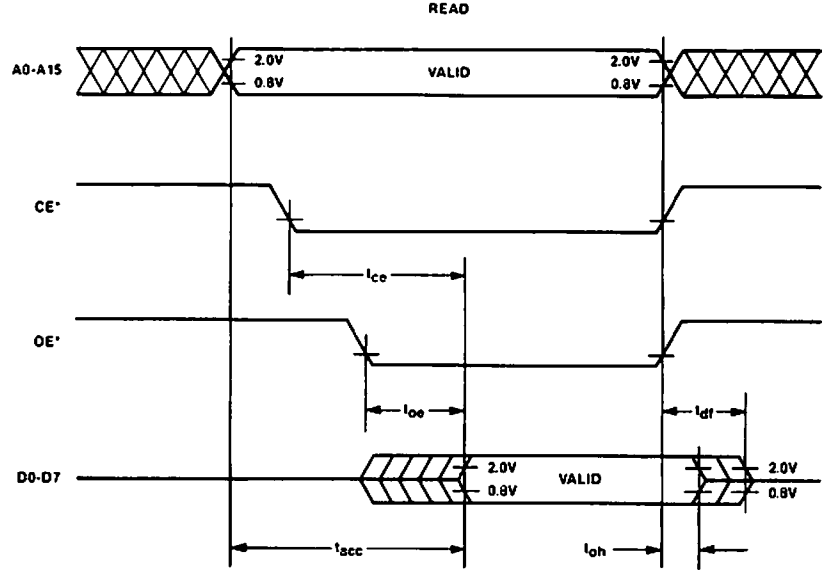
SOCKET

CONFIGURATION J6(BANK 1) OR J7 (BANK 2) CONNECTIONS

1	2 to 4	8K x 8 or 16K x 8 ROM/PROM/EPROM
2	1 to 3, and 2 to 4	32K x 8 ROM/PROM/EPROM
3	1 to 3, and 4 to 6	64K x 8 ROM/PROM/EPROM (FACTORY CONFIG.)
4	3 to 5	2K x 8 or 8K x 8 EEPROM
5	1 to 4, and 3 to 5	32K x 8 EEPROM

FIGURE 4-4. ROM/PROM/EPROM/EEPROM Sockets Configurations

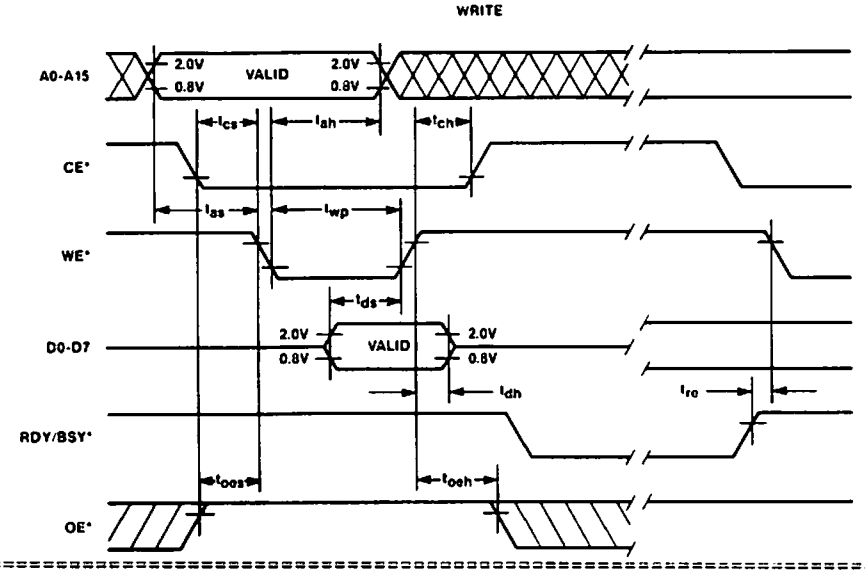
The ROM/PROM/EPROM/EEPROM devices must meet the timings shown in Figure 4-5, and are guaranteed the timings shown in Figure 4-6.



SYMBOL	DESCRIPTION	TIME IN NSEC MVME133 12.5 MHz MIN	MAX	MVME133-1 16.67 MHz MIN	MAX
tacc	Address valid to data valid	---	390	---	295
tce	CE* low to data valid	---	350	---	265
toe	OE* low to data valid	---	275	---	200
toh	Address invalid, CE* or OE* high to data not valid	0	---	0	---
tdf	CE* or OE* high to data high impedance	---	80	---	60

NOTE: The MVME133 does not guarantee a maximum transition time on address and data lines during the time that CE* is high.

FIGURE 4-5. ROM/PROM/EPROM/EEPROM Timings Required by MVME133



SYMBOL	DESCRIPTION	TIME IN NSEC MVME133 12.5 MHz MIN	MAX	MVME133-1 16.67 MHz MIN	MAX
tas	Address valid to WE* low	86	---	66	---
tcs	CE* low to WE* low	44	---	34	---
toes	OE* high to WE* low	93	---	63	---
tah	Address valid after WE* low	324	---	234	---
twp	WE* low pulse width	230	---	170	---
tds	Data valid to WE* high	245	---	185	---
tdh	WE* high to data not valid	93	---	63	---
toeh	WE* high to OE* low	221	---	161	---
tch	WE* high to CE* high	58	---	38	---
tre	RDY/BSY* high to first write to device (NOTE 1)	126	---	96	---

NOTES:
1. Assumes that software waits for RDY/BSY* to go high before writing to the device again.
2. The MVME133 does not guarantee a maximum transition time on address and data lines during the time that CE* is high.

FIGURE 4-6. ROM/PROM/EPROM/EEPROM Timings Guaranteed by MVME133

The MVME133 provides no protection against inadvertent writes to EEPROM that might happen during power on/off transitions. Most devices provide some level of internal protection. In order to gain "absolute protection", devices with additional "software protection" are recommended.

4.3.12.2 Real-Time Clock. The real-time clock on the MVME133 is an MM58274. It has a 4-bit data bus that is connected to [D24] - [D27]. It is capable of generating interrupts to the MPU on level four.

Note that read accesses to the real-time clock deactivate the [PWRUP*] status bit. Note also that any read cycle to the real-time clock with [A25] = 1 causes the VMEbus interrupter to request a VMEbus level-three interrupt. (Refer to the VMEbus single-level interrupter in paragraph 4.3.9.2.)

4.3.13 MC68881 Floating Point Coprocessor (Sheet 18)

The MC68881 shares the same clock with the MPU (12.5 MHz on the MVME133, and 16.67 MHz on the MVME133-1). The MVME133 connects the MC68881 as a 32-bit port. It imposes no delays on CE* or on [DSACK0*] or [DSACK1*] between the MPU and the MC68881. Hence, accesses to the MC68881 (other than a read to the response or save CIR) requires three MPU clocks (no wait cycle). Reading the response or save CIR is performed in five MPU clock cycles.

Refer to Chapter 3 for the memory map of the registers in the MC68881 floating point coprocessor.

4.3.14 Multiprotocol Serial Ports (Sheet 19)

The MVME133 uses the Z8530 to implement its two multiprotocol serial ports. Port A of the Z8530 is connected to RS-485 drivers and receivers. Port B of the Z8530 is connected to RS-232C drivers and receivers. Because of its internal structure, there are several means of obtaining the baud rate clocks for each of the two serial channels. Each channel within the Z8530 has a programmable baud rate generator. The baud rate generator input can be from the RTXC input or from PCLK. The hardware on the MVME133 allows the RTXC pin for each channel to be connected to an external source or to the onboard 1.23 MHz clock. Table 4-5 shows the values in the Z8530 channel time constant register that are required to create some common baud rates.

TABLE 4-5. Baud Rate Generator with Clock Source = RTXC Pin = 1.23 MHz

BAUD RATE	THEORETICAL 16 x CLOCK FREQUENCY	TIME CONSTANT REGISTER VALUE	ACTUAL FREQUENCY	PERCENT ERROR
19200	307200	0	307692	0.16
9600	153600	2	153846	0.16
4800	76800	6	76923	0.16
2400	38400	$E	38461	0.16
1200	19200	$1E	19231	0.16
600	9600	$3E	9615	0.16
300	4800	$7E	4808	0.16
110	1760	$15E	1748	0.67
BAUD RATE	THEORETICAL 1 x CLOCK FREQUENCY	TIME CONSTANT REGISTER VALUE	ACTUAL FREQUENCY	PERCENT ERROR
	N/A	0	307692	N/A
	N/A	1	205128	N/A
	N/A	2	153846	N/A
	N/A	3	123076	N/A
	N/A	4	102564	N/A
	N/A	5	87912	N/A
	N/A	6	76923	N/A
	N/A	7	68376	N/A
64000	64000	8	61538	3.85
56000	56000	9	55944	0.10
48000	48000	$B	47337	1.38
38400	38400	$E	38461	0.16

The Z8530 DPLL input can be either the BRG output or the RTXC pin. The DPLL operates at 32 times the data rate for NRZI and at 16 times the data rate for FM. Tables 4-6 and 4-7 give some of the data rates that are achievable with the MPU operating at 12.5 MHz (in the MVME133) and at 16.67 MHz (in the MVME133-1), respectively.

TABLE 4-6. Baud Rate Generator with Clock Source=PCLK = 3.125 MHz (MVME133)

BAUD RATE	THEORETICAL 32 x CLOCK FREQUENCY	TIME CONSTANT REGISTER VALUE	ACTUAL FREQUENCY	PERCENT ERROR
24414	N/A	0	781248	N/A
16276	N/A	1	520832	N/A
12207	N/A	2	390624	N/A
BAUD RATE	THEORETICAL 16 x CLOCK FREQUENCY	TIME CONSTANT REGISTER VALUE	ACTUAL FREQUENCY	PERCENT ERROR
48000	768000	0	781250	1.7
32552	N/A	1	520832	N/A
24414	N/A	2	390625	N/A

TABLE 4-7. Baud Rate Generator with Clock Source=PCLK = 4.167 MHz (MVME133-1)

BAUD RATE	THEORETICAL 32 x CLOCK FREQUENCY	TIME CONSTANT REGISTER VALUE	ACTUAL FREQUENCY	PERCENT ERROR
32552	N/A	0	1041667	N/A
21701	N/A	1	694444	N/A
16276	N/A	2	520833	N/A
BAUD RATE	THEORETICAL 16 x CLOCK FREQUENCY	TIME CONSTANT REGISTER VALUE	ACTUAL FREQUENCY	PERCENT ERROR
64000	1024000	0	1041667	1.7
43403	N/A	1	694444	N/A
32552	N/A	2	520833	N/A

If other frequencies than the ones available with 1.23 MHz as the BRG clock source are needed, the frequency of 1.23 MHz can be changed by reprogramming U18, PALSC, to divide the 16 MHz by a value other than 13. The user must be aware that both ports of the Z8530 and the MC68901 may be using the 1.23 MHz signal, and changing that frequency may make it impossible to create a desired frequency on the other port of the Z8530 and/or on the MC68901 MFP.

4.3.14.1 RS-485 Port. Port A of the Z8530 uses RS-485 drivers and receivers. The RS-485 signals are routed to P2. An external cable may be connected to P2 and the user must make a crossover cable to convert from the cable pinout of P2 on the MVME133 to the pinout of the user's serial network. The connector that is used to interface to the RS-485 network should take shielding into consideration.

4

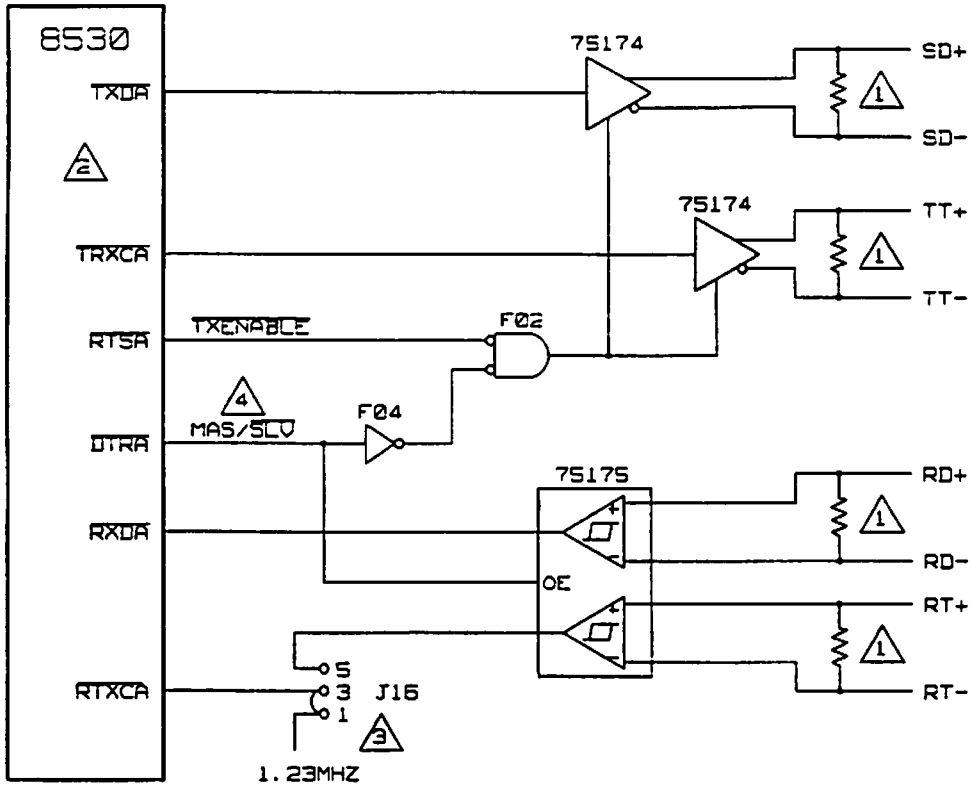
The RS-485 port can be configured by software to be either master or slave, with the DTR/REQA (DTRA) bit. Also, the RTSA bit may be used to enable/disable the RS-485 drivers. Table 4-8 and Figures 4-7 and 4-8 show the possible configurations for the RS-485 port.

TABLE 4-8. RS-485 Port Configurations

DTR/REQA	RTSA	CONFIGURATION	DESCRIPTION
low	low	Slave (drivers on)	TXDA drives RD+/-, SD+/- drives RXDA, TRXCA drives RT+/-, TT+/- drives RTXCA if J16 pins 3-5 are connected.
low	high	Slave (drivers off)	RD+/- not driven, SD+/- drives RXDA, RT+/- not driven, TT+/- drives RTXCA if J16 pins 3-5 are connected.
low	high	Half duplex receive	RD+/- not driven, SD+/- drives RXDA, RT+/- not driven, TT+/- drives RTXCA if J16 pins 3-5 are connected.
high	low	Master (drivers on)	TXDA drives SD+/-, RD+/- drives RXDA, TRXCA drives TT+/-, RT+/- drives RTXCA if J16 pins 3-5 are connected.
high	low	Half duplex send	TXDA drives SD+/-, RD+/- drives RXDA, TRXCA drives TT+/-, RT+/- drives RTXCA if J16 pins 3-5 are connected.
high (NOTE)	high (NOTE)	Master (drivers off) (NOTE)	SD+/- not driven, RD+/- drives RXDA, TT+/- not driven, RT+/- drives RTXCA if J16 pins 3-5 are connected.

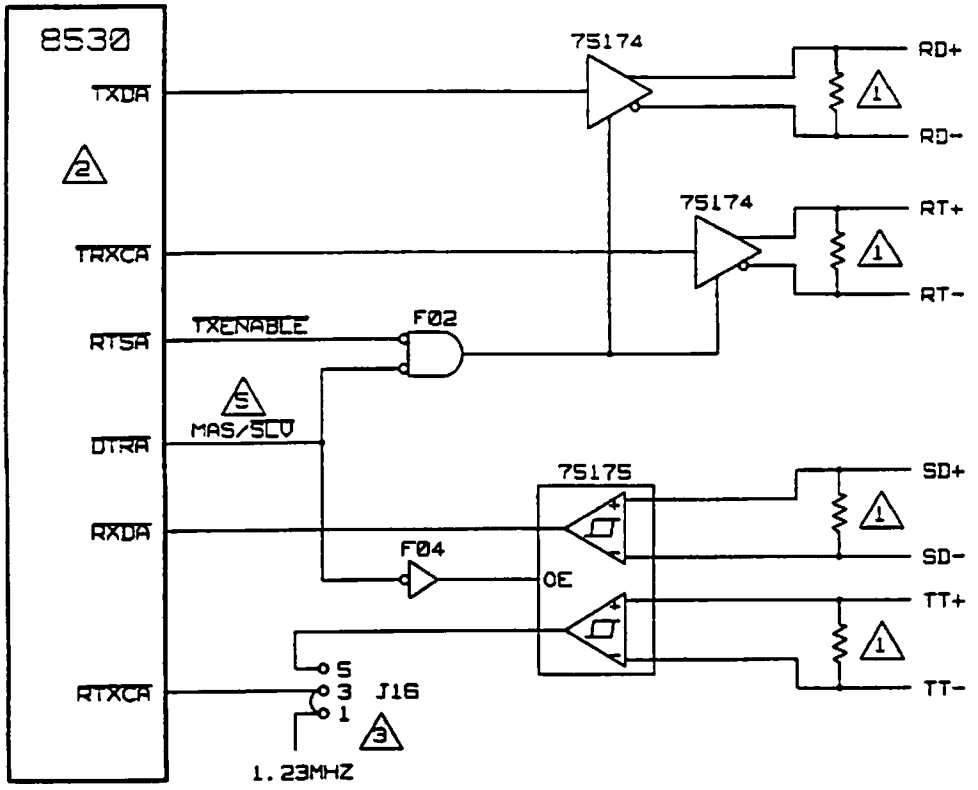
NOTE: RESET causes DTR/REQA and RTSA to enter this state.

Note that the MVME133 has only clocks and data. There are no hardware handshakes for the RS-485 port.



NOTE:
1. USER-SUPPLIED RESISTOR SIP, TYPICALLY 120 OHMS.
2. AUTO-ENABLE BIT FOR PORT 'A' MUST BE CLEARED.
3. FACTORY CONFIGURATION.
4. $\overline{DTRA}$ = 1 TO INDICATE 'MASTER'.

FIGURE 4-7. MVME133 RS-485 Port Configured as Master (To DCE)

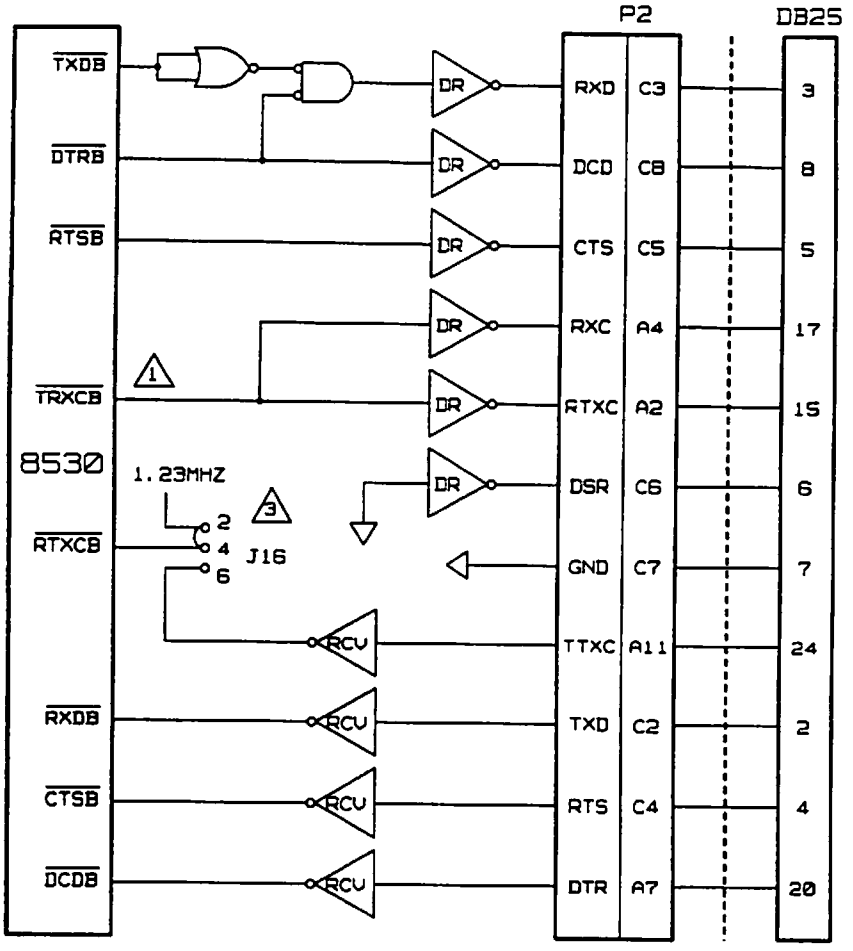


NOTE:
1. USER-SUPPLIED RESISTOR SIP, TYPICALLY 120 OHMS.
2. AUTO-ENABLE BIT FOR PORT 'A' MUST BE CLEARED.
3. FACTORY CONFIGURATION.
5. DTRA = 0 TO INDICATE 'SLAVE'.

FIGURE 4-8. MVME133 RS-485 Port Configured as Slave (To DTE)

4.3.14.2 RS-232C Port. Port B of the Z8530 uses RS-232C drivers and receivers. All of the buffers and the configuration headers are on the MVME133. This port may be configured either as a DTE or as a DCE, by using J13. Figure 4-9 shows the DCE configuration of the port, and Figures 4-10 through 4-12 show the DTE configurations of the port. The MVME133 also provides an external ability to disable the TXDB pin of the Z8530. When the DTRB pin of the Z8530 is high, the TXDB pin is disabled to the RS-232C port. When the DTRB pin is low, the TXDB pin is enabled to the RS-232C port. Note that the DTRB pin is also an RS-232C signal line. DTRB is set high by a reset to the Z8530.

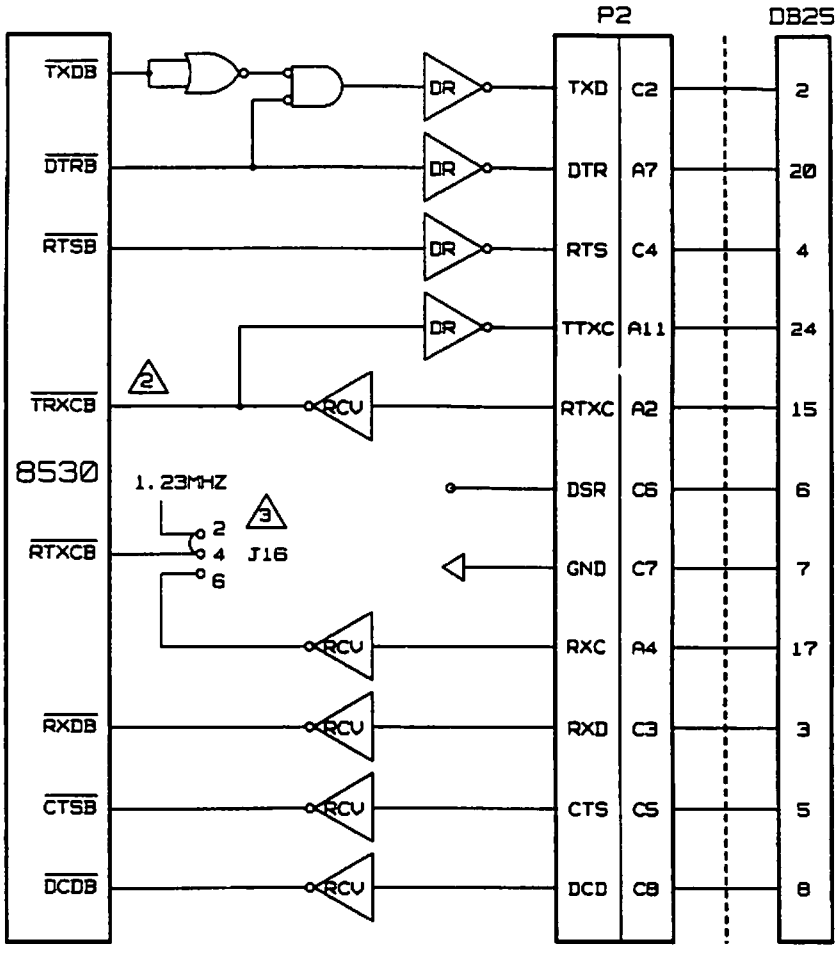
The RS-232C lines are routed to P2. An external cable may be connected to P2 and a DB-25 connector crimped directly onto it. For shielding purposes, the DB-25 should be mounted to a back panel that is mounted to the chassis, and connection should be made between the back panel and the shielding metal of the DB-25.



NOTE: 1 TRXCB MUST BE PROGRAMMED AS OUTPUT

3 FACTORY CONFIGURATION

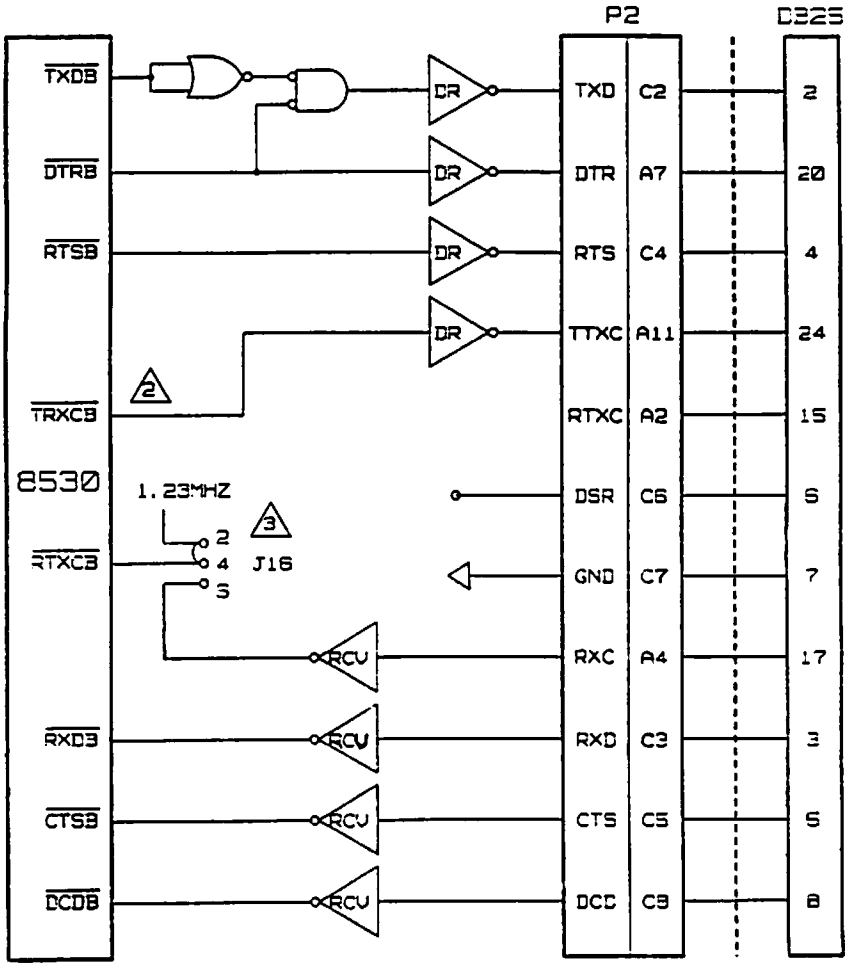
FIGURE 4-9. MVME133 RS-232C Port Configured as DCE (To Terminal)



NOTE: (2) TRXCB MUST BE PROGRAMMED AS INPUT

(3) FACTORY CONFIGURATION

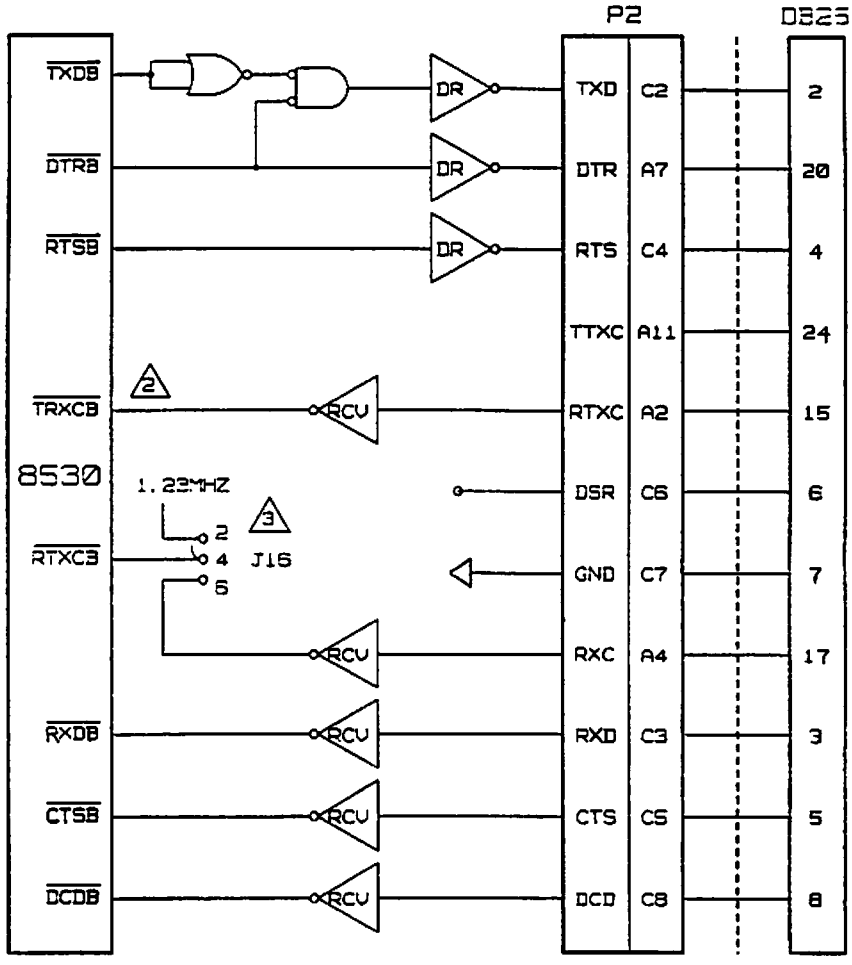
FIGURE 4-10. MVME133 RS-232C Port Configured as DTE (To Modem)



NOTE: (2) TRXCB MUST BE PROGRAMMED AS OUTPUT

(3) FACTORY CONFIGURATION

FIGURE 4-11. MVME133 RS-232C Port Configured as DTE (To Modem) with RTXC Not Used



NOTE: (2) TRXCB MUST BE PROGRAMMED AS INPUT

(3) FACTORY CONFIGURATION

FIGURE 4-12. MVME133 RS-232C Port Configured as DTE (To Modem) with TTXC Not used

THIS PAGE INTENTIONALLY LEFT BLANK.

CHAPTER 5

SUPPORT INFORMATION

5.1 INTRODUCTION

This chapter provides the interconnection signals, parts list with parts location illustration, and schematic diagram for the MVME133 module.

5.2 INTERCONNECT SIGNALS

The MVME133 module interconnects with the VMEbus through connector P1, with the VMEbus and serial ports through connector P2, and with an RS-232C device through connector J14.

5.2.1 Connector P1 Interconnect Signals

Connector P1 is a standard DIN 41612 triple-row, 96-pin male connector. The MVME133 interconnects with the VMEbus through rows A, B, and C of P1 and through row B of P2. 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 32 three-state bidirectional data lines that provide the data path between VMEbus master and slave.
A9	GND	GROUND - connected to the MVME133 ground plane.
A10	SYSCLK	SYSTEM CLOCK - a 16 MHz free-running clock that is driven by the MVME133 only when it is configured as system controller.
A11	GND	GROUND - connected to the MVME133 ground plane.
A12	DS1*	DATA STROBE 1 - signal that indicates which part of the data bus is transferring data. It is driven by the MVME133 when it is the VMEbus master. It is received by the MVME133 when it is a VMEbus slave.
A13	DS0*	DATA STROBE 0 - signal that indicates which part of the data bus is transferring data. It is driven by the MVME133 when it is the VMEbus master. It is received by the MVME133 when it is a VMEbus slave.

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

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
A14	WRITE*	WRITE - signal that specifies the direction of data transfers. It is driven by the MVME133 as a VMEbus master and received by the MVME133 as a slave.
A15	GND	GROUND - connected to the MVME133 ground plane.
A16	DTACK*	DATA TRANSFER ACKNOWLEDGE - signal that indicates that valid data is available on the data bus during a read cycle or that it has been accepted during a write cycle. It is received by the MVME133 as a VMEbus master and driven by the MVME133 as a VMEbus slave.
A17	GND	GROUND - connected to the MVME133 ground plane.
A18	AS*	ADDRESS STROBE - the falling edge of this signal indicates that a valid address, address modifier, LWORD*, and IACK* are available on the VMEbus. It is driven by the MVME133 as a VMEbus master and received by it as a VMEbus slave.
A19	GND	GROUND - connected to the MVME133 ground plane.
A20	IACK*	INTERRUPT ACKNOWLEDGE - signal that indicates an interrupt acknowledge cycle on the VMEbus. It is driven true by the MVME133 during an interrupt acknowledge to the VMEbus. The MVME133 does not respond to interrupt acknowledge cycles as a slave.
A21	IACKIN*	INTERRUPT ACKNOWLEDGE IN - IACKIN* and IACKOUT* form a daisy-chained signal. When the MVME133 is configured as system controller, it drives IACKOUT* according to the IACK daisy-chain driver specification. When the MVME133 is not configured as system controller, it connects IACKIN* directly to IACKOUT*.
A22	IACKOUT*	INTERRUPT ACKNOWLEDGE OUT - see IACKIN*.
A23	AM4	ADDRESS MODIFIER (bit 4) - one of the three-state lines that provide additional information about the address bus, such as size, and cycle type. It is driven by the MVME133 as a master and received by the MVME133 as a slave.

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

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
A24-A30	A07-A01	ADDRESS bus (bits 7-1) - seven of 23 three-state lines that specify an address in the memory map. They are driven by the MVME133 as a master and received by the MVME133 as a slave.
A31	-12 VDC	-12 Vdc power - used by the RS-232C drivers on the MVME133.
A32	+5 VDC	+5 Vdc power - used by the logic circuits on the MVME133. Connected to the MVME133 +5V plane.
B1	BBSY*	BUS BUSY - this signal is driven true by the MVME133 when it is VMEbus master. When the MVME133 is system controller, BBSY* is an input to the level three arbiter.
B2	BCLR*	BUS CLEAR - not used by the MVME133.
B3	ACFAIL*	AC FAILURE - the MVME133 monitors this signal line to detect ac power failure.
B4	BG0IN*	BUS GRANT IN (level 0) - this signal going true at the input to the MVME133 indicates that it may become VMEbus master if it is configured for level 0. If the MVME133 is not requesting VMEbus mastership, then it drives the BG0OUT* signal line low. The other three bus grant lines are tied directly to the corresponding bus grant out lines.
B5	BG0OUT*	BUS GRANT OUT (level 0) - see BG0IN*.
B6	BG1IN*	BUS GRANT IN (level 1) - same as BG0IN* on B4.
B7	BG1OUT*	BUS GRANT OUT (level 1) - same as BG0OUT* on B5.
B8	BG2IN*	BUS GRANT IN (level 2) - same as BG0IN* on B4.
B9	BG2OUT*	BUS GRANT OUT (level 2) - same as BG0OUT* on B5.
B10	BG3IN*	BUS GRANT IN (level 3) - same as BG0IN* on B4.
B11	BG3OUT*	BUS GRANT OUT (level 3) - same as BG0OUT* on B5.

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

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
B12	BR0*	BUS REQUEST (level 0) - signal line driven by the MVME133 when it desires to become VMEbus master (if it is configured for level 0), and received by the MVME133 to detect whether it should relinquish VMEbus mastership.
B13	BR1*	BUS REQUEST (level 1) - same as BR0* on B12.
B14	BR2*	BUS REQUEST (level 2) - same as BR0* on B12.
B15	BR3*	BUS REQUEST (level 3) - same as BR0* on B12, plus when the MVME133 is configured as system controller, BR3* is an input to the level three arbiter.
B16-B19	AM0-AM3	ADDRESS MODIFIER (bits 0-3) - same as AM4 on pin A23.
B20	GND	GROUND - connected to the MVME133 ground plane.
B21	SERCLK	Not used.
B22	SERDAT*	Not used.
B23	GND	GROUND - connected to the MVME133 ground plane.
B24-B27	IRQ7*-IRQ4*	INTERRUPT REQUEST (7-4) - four of the seven prioritized interrupt request inputs to the MVME133. Jumper enabled, level 7 is the highest priority.
B28	IRQ3*	INTERRUPT REQUEST (3) - one of the seven prioritized interrupt request inputs/outputs of the MVME133. Jumper enabled as input, and jumper enabled as output.
B29-B30	IRQ2*-IRQ1*	INTERRUPT REQUEST (2-1) - two of the seven prioritized interrupt request inputs to the MVME133. Jumper enabled, level 7 is the highest priority.
B31	+5V STDBY	Not used.
B32	+5 VDC	+5 Vdc power - same as +5 VDC on pin A32.
C1-C8	D08-D15	DATA bus (bits 8-15) - same as D00-D07 on pins A1-A8.

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

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
C9	GND	GROUND - connected to the MVME133 ground plane.
C10	SYSFAIL*	SYSTEM FAIL - signal driven by the MVME133 when [BRDFAIL] is true. Also can be monitored via the MC68901 MFP.
C11	BERR*	BUS ERROR - signal driven by the MVME133 bus time-out circuit when it is the system controller and a VMEbus data strobe cycle exceeds 120 us. Also monitored by the MVME133 when it is the VMEbus master. It causes a bus error exception in the MC68020 in this case.
C12	SYSRESET*	SYSTEM RESET - signal driven by the MVME133 when it is configured as system controller during power-up, when the front panel RESET switch is depressed, when a watchdog time-out occurs, or when remote reset becomes true. Also an input to the MVME133 that causes all of its devices to be reset.
C13	LWORD*	LONGWORD - signal driven true by the MVME133 when it does a 32-bit data transfer over the VMEbus. Also monitored by the MVME133 to distinguish 32-bit from 16-bit data accesses to its RAM from the VMEbus.
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.

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

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
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 RS-232C drivers on the MVME133.
C32	+5 VDC	+5 Vdc power - same as +5 VDC on pin A32.

5

5.2.2 Connector P2 Interconnect Signals

Connector P2 is a standard DIN 41612 triple-row, 96-pin male connector. The MVME133 interconnects with the VMEbus through rows A, B, and C of P1 and through row B of P2. Serial ports A (RS-485) and B (RS-232C) of the Z8530 and remote reset are brought out through rows A and C of P2. Table 5-2 lists each pin connection, signal mnemonic, and signal characteristic for the connector.

TABLE 5-2. Connector P2 Interconnect Signals

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
A1	--	Not used.
A2	RTXC	RS-232C Transmitter Signal Element Timing (DCE) - signal line driven by the TRXCB pin of the Z8530 when port B is configured as DCE, and optionally input to the same pin when port B is configured as DTE.
A3	--	Not used.

TABLE 5-2. Connector P2 Interconnect Signals (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
A4	RXC	RS-232C Receiver Signal Element Timing (DCE) - signal line driven by the TRXCB pin of the Z8530 when port B is configured as DCE, and optionally input to the RTXCB pin of the Z8530 when port B is configured as DTE.
A5-A6	--	Not used.
A7	DTR	RS-232C Data Terminal Ready - input to the DCDB pin of the Z8530 when port B is configured as DCE, and output from the DTR/REQB (DTRB) pin of the Z8530 when port B is configured as DTE.
A8-A10	--	Not used.
A11	TTXC	RS-232C Transmitter Signal element Timing (DTE) - signal line optionally driven by the TRXCB pin of the Z8530 when port B is configured as DTE, and optionally input to the RTXCB pin of the Z8530 when port B is configured as DCE.
A12	--	Not used.
A13	SD+	RS-485 Send Data - this signal line is half of the balanced differential pair that includes SD+ and SD-. The pair is buffered to the RXDA pin of the Z8530 when port A is configured as slave, and it is buffered from the TXDA pin of the Z8530 when port A is configured as master.
A14	TT+	RS-485 Terminal Timing - this signal line is half of the balanced differential pair that includes TT+ and TT-. The pair is buffered to the RTXCA pin of the Z8530 (depends on J16) when port A is configured as slave, and it is buffered from the TRXCA pin of the Z8530 when port A is configured as master.
A15	RD+	RS-485 Receive Data - this signal line is half of the balanced differential pair that includes RD+ and RD-. The pair is buffered from the TXDA pin of the Z8530 when port A is configured as slave, and it is buffered to the RXDA pin of the Z8530 when port A is configured as master.

TABLE 5-2. Connector P2 Interconnect Signals (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
A16	RT+	RS-485 Receive Timing - this signal line is half of the balanced differential pair that includes RT+ and RT-. The pair is buffered from the TRXCA pin of the Z8530 when port A is configured as slave, and it is buffered to the RTXCA pin of the Z8530 (depends on J16) when port A is configured as master.
A17	GND	GROUND - connects to MVME133 ground plane.
A18-A19	--	Not used.
A20	RRESET*	Remote Reset input - when this signal line is high, no reset function occurs on the MVME133. When it is low, the MVME133 is reset and remains in the reset state until it goes high again. (Note that this signal line is also connected to pin P2-A32.)
A21	--	Not used.
A22	GND	GROUND - connected to MVME133 ground plane.
A23-A30	--	Not used.
A31	GND	GROUND - connected to MVME133 ground plane.
A32	RRESET*	Remote Reset input - when this signal line is high, no reset function occurs on the MVME133. When it is low, the MVME133 is reset and remains in the reset state until it goes high again. (Note that this signal line is also connected to pin P2-A20.)
B1	+5 VDC	+5 Vdc power - used by the logic circuits on the MVME133. Connected to the MVME133 +5V plane.
B2	GND	GROUND - connected to the MVME133 ground plane.
B3	Reserved	Not used.
B4-B11	A24-A31	ADDRESS bus (bits 23-31) - Not used.
B12	GND	GROUND - connected to MVME133 ground plane.
B13	+5 VDC	+5 Vdc power - used by the logic circuits on the MVME133. Connected to the MVME133 +5V plane.

TABLE 5-2. Connector P2 Interconnect Signals (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
B14-B21	D16-D23	DATA bus (bits 16-23) - eight of 32 three-state bidirectional data lines that provide the data path between VMEbus master and slave.
B22	GND	GROUND - connected to MVME133 ground plane.
B23-B30	D24-D31	DATA bus (bits 24-31) - eight of 32 three-state bidirectional data lines that provide the data path between VMEbus master and slave.
B31	GND	GROUND - connected to MVME133 ground plane.
B32	+5 VDC	+5 Vdc power - used by the logic circuits on the MVMV133. Connected to the MVME133 +5V plane.
C1	--	Not used.
C2	TXD	RS-232C Transmitted Data - input to the RXDB pin of the Z8530 when port B is configured as DCE, and output from the TXDB pin of the Z8530 when port B is configured as DTE.
C3	RXD	RS-232C Received Data - output from the TXDB pin of the Z8530 when port B is configured as DCE, and input to the RDXB pin of the Z8530 when port B is configured as DTE.
C4	RTS	RS-232C Request To Send - input to the CTSB pin of the Z8530 when port B is configured as DCE, and output from the RTSB pin of the Z8530 when port B is configured as DTE.
C5	CTS	RS-232C Clear To Send - output from the RTSB pin of the Z8530 when port B is configured as DCE, and input to the CTSB pin of the Z8530 when port B is configured as DTE.
C6	DSR	RS-232C Data Set Ready - output from the Z8530 that is always high when port B is configured as DCE, and no connect when port B is configured as DTE.
C7	GND	RS-232C Signal Ground/Common Return - connected to the MVME133 ground plane. Not connected to chassis ground on the MVME133.

TABLE 5-2. Connector P2 Interconnect Signals (cont'd)

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
C8	DCD	RS-232C Received Line Signal Detector - output from the DTR/REQB (DTRB) pin of the Z8350 when port B is configured as DCE, and input to the DCDB pin of the Z8530 when port B is configured as DTE.
C9-C13	--	Not used.
C14	SD-	RS-485 Send Data - this signal is half of the balanced differential pair that includes SD+ and SD-. Refer to SD+ on pin A13.
C15	TT-	RS-485 Terminal Timing - this signal line is half of the balanced differential pair that includes TT+ and TT-. Refer to TT+ on pin A14.
C16	RD-	RS-485 Receive Data - this signal line is half of the balanced differential pair that includes RD+ and RD-. Refer to RD+ on pin A15.
C17	RT-	RS-485 Receive Timing - this signal line is half of the balanced differential pair that includes RT+ and RT-. Refer to RT+ on pin A16.
C18-C19	--	Not used.
C20	GND	GROUND - connected to MVME133 ground plane.
C21	--	Not used.
C22	GND	GROUND - connected to MVME133 ground plane.
C23-C28	--	Not used.
C29	GND	GROUND - connected to MVME133 ground plane.
C30	--	Not used.
C31-C32	GND	GROUND - connected to MVME133 ground plane.

5.2.3 Connector J14 Interconnect Signals

Connector J14 is a standard RS-232C DB-25 25-pin female connector. J14 provides the interconnection for the debug port of the MVME133. Table 5-3 lists each pin connection, signal mnemonic, and signal characteristic for the connector. Note that J14 mates with a 25-pin cable to connect to a terminal. For further details, refer to Appendix B.

TABLE 5-3. RS-232C Connector J14 Interconnect Signals

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
1	--	Not used.
2	DTXD	RS-232C Transmitted Data - input to the SI pin of the MC68901 Multi-Function Peripheral (MFP).
3	DRXD	RS-232C Received Data - output from the SO pin of the MC68901 MFP.
4	--	Not used.
5	DCTS	RS-232C Clear To Send - output from the GPIO3 pin of the MC68901 MFP.
6	DDSR	RS-232C Data Set Ready - output that is always driven high by the MVME133.
7	GND	RS-232C Signal Ground/Common Ground - connected to the MVME133 ground plane. Not connected to chassis ground by the MVME133.
8	DDCD	RS-232C Received Line Signal Detector - signal line that is pulled to +12 V by the MVME133.
9-19	--	Not used.
20	DDTR	RS-232C Data Terminal Ready - input to the GPIO0 pin of the MC68901 MFP.
21-25	--	Not used.

5.3 PARTS LIST

Table 5-4 lists the components of the MVME133 and MVME133-1. Unless otherwise indicated, all components are used in both the MVME133 and the MVME133-1. The parts locations are illustrated in Figure 5-1. These parts reflect the latest issue of hardware at the time of printing.

TABLE 5-4. MVME133 and MVME133-1 Module Parts List

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
--	84-W8434B01	Printed wiring board assembly, MVME133
C1	23NW9618A80	Capacitor, electrolytic, radial lead, 10 uF ± 20% @ 50 Vdc
C2,C3,C5-C10, C12,C16,C17, C20,C21,C23-C34,C36-C42, C44-C62	21NW9632A03	Capacitor, ceramic, axial lead, 0.1 uF ± 20% @ 50 Vdc
C4,C11,C15, C22,C35,C43, C63	23NW9618A71	Capacitor, electrolytic, radial lead, 47 uF ± 20% @ 10 Vdc
C13,C14	21SW992C043	Capacitor, ceramic, 20 pF ± 5% @ 50 Vdc
C18,C19	21SW992C042	Capacitor, ceramic, 330 pF ± 5% @ 50 Vdc
CR1,CR2	48NW9616A03	Diode, silicon, 1N4148/1N914
DS1,DS2	48NW9612A49	LED, red, right angle
DS3	48NW9612A59	LED, green, right angle
E1,E3,J1-J13, J15-J16	29NW9805C07	Pin, 0.025 inch square, gold, autoinsert, used on E1(1), E3(1), J1(2), J2(4), J3(6), J4(12), J5(2), J6(6), J7(6), J8(2), J9(2), J10(2), J11(2), J12(14), J13(33), J15(10), J16(6)
E2	--	Not used.
--	29NW9805B17	Jumper, insulated, shorting (43 req'd) (used with J1-J13, J15-J16)
J14	28NW9802G42	Connector, 25-pin, right-angle, D-subminiature, socket
--	47NW9405A28	Jackpost assembly, D-subminiature (used with J14)

TABLE 5-4. MVME133 and MVME133-1 Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
L1	76NW9810A04	Bead, ferrite, 0.146 inch x 0.126 inch (at P1-B1)
P1,P2	28NW9802E51	Connector, 96-pin, plug, PWB
--	03SW993D110	Screw, phillips, M2 x 0.4 x 10 (4 req'd) (usedwith P1 & P2)
--	02SW990D007	Nut, hexagonal, M2 x 0.4 x 1.6 (4 req'd) (used with P1 & P2)
R1,R12	06SW-124A97	Resistor, film, 100k ohms, 5%, 1/4 W
R2,R6,R24, R25	51NW9626B75	Resistor network, SIP, seven 10k ohm
R3	06SW-124A18	Resistor, film, 51 ohms, 5%, 1/4 W
R4,R5,R11	06SW-124A23	Resistor, film, 82 ohms, 5%, 1/4 W
R7,R10,R15	51NW9626B56	Resistor network, SIP, nine 10k ohm
R8,R9,R16, R18-R21	51NW9626B64	Resistor network, SIP, four 47 ohm
R13	51NW9626B59	Resistor network, SIP, nine 2.7k ohm
R14	51NW9626B63	Resistor network, SIP, nine 39k ohm
R17	06SW-125A53	Resistor, film, 1.5k ohms, 5%, 1/2 W
R22	51NW9626A69	Resistor network, SIP, seven 1k ohm
R23	--	Optional customer-installed resistor network, SIP, four 120 ohms (any value between 90 and 150 ohms). (Refer to paragraph 2.4.3.)
--	09-W4659B08	Socket, I.C., SIL, 8-pin (used with R23)
S1,S2	40NW9801B70	Switch, push, SPDT, momentary, PC, right-angle, gold
--	38NW9404C11	Cap, switch, black, for B70 (used with S1)
--	38NW9404C12	Cap, switch, red, for B70 (used with S2)
U1	--	Not used.

TABLE 5-4. MVME133 and MVME133-1 Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
U2,U40,U53, U59	51NW9615K70	I.C. 74F08PC
U3,U4,U8,U9, U30,U34,U35, U100,U102	51NW9615S43	I.C. SN74ALS623A-1N
U5	(NOTE)	I.C. Programmed PAL
--	09NW9811A78	Socket, I.C., DIL, 20-pin (used with U5, U18, U22, U42, U45, U58, U69)
U6,U11,U16	51NW9615F38	I.C. SN74LS393N
U7,U37,U43, U44,U47,U51, U55,U57,U60, U61,U62,U63	51NW9615J39	I.C. 74F74PC
U10	--	Not used.
U12	(NOTE)	I.C. Programmed PAL
--	09NW9811B01	Socket, I.C., DIL, 24-pin (used with U12, U52, U64, U75, U78)
U13	--	Not used.
U14,U15,U21	51NW9615R55	I.C. N74F38N
U17	51NW9615E93	I.C. SN74LS14N
U18	(NOTE)	I.C. Programmed PAL
U19	51NW9615K47	I.C. 74F244PC
U20	51NW9615R66	I.C. MM58274N
U22	(NOTE)	I.C. Programmed PAL
U23	51NW9615P48	I.C. SN74ALS240AN
U24,U31, U36,U46	--	Customer-supplied ROMs/PROMs/EPROMs/EEPROMs
XU24,XU31, XU36,XU46	09-W4659B14	Socket, I.C., SIL, 14-pin (2 each req'd per chip 8 total)

TABLE 5-4. MVME133 and MVME133-1 Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
U25,U50	51NW9615K73	I.C. 74F00PC
U26,U77	51NW9615K66	I.C. 74F32PC
U27	51NW9615N72	I.C. MC68881RC12 (on MVME133)
U27	51NW9615T12	I.C. MC68881RC16A (alternate is XC68881RC16A, part number 51NW9615T17) (on MVME133-1)
--	09NW9811A71	Socket, I.C., pin-grid-array, 68-pin (used with U27)
U28	51NW9615R99	I.C. XC68020RC12 (on MVME133)
U28	51NW9615R89	I.C. MC68020RC16B (on MVME133-1)
--	09NW9811B12	Socket, I.C., pin-grid-array, 124-pin (used with U28)
U29,U81	51NW9615K09	I.C. SN74ALS244N
U32	51NW9615F30	I.C. DM74S05N
U33,U41,U68	51NW9615K71	I.C. 74F04PC
U38,U54	51NW9615K68	I.C. 74F11PC
U39	51NW9615L74	I.C. 74F163APC
U42	(NOTE)	I.C. Programmed PAL
U45	(NOTE)	I.C. Programmed PAL
U48,U82	51NW9615K72	I.C. 74F02PC
U49	51NW9615N56	I.C. 74F174PC
U52	(NOTE)	I.C. Programmed PAL
U56	51NW9615K69	I.C. 74F10PC
U58	(NOTE)	I.C. Programmed PAL
U64	(NOTE)	I.C. Programmed PAL
U65	51NW9615N40	I.C. MK68901N (on MVME133)
U65	51NW9615T24	I.C. MK68901N-5D (on MVME133-1)

TABLE 5-4. MVME133 and MVME133-1 Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
--	09-W4659B24	Socket, I.C., SIL, 24-pin (2 req'd) (used with U65)
U66,U72,U73	51NW9615S83	I.C. MC145406P
U67	51NW9615J63	I.C. SN74S51N3
U69	(NOTE)	I.C. Programmed PAL
U70,U74,U79, U80	51NW9615K59	I.C. 74F175PC
U71,U76	51NW9615K60	I.C. 74F158PC
U75	(NOTE)	I.C. Programmed PAL
U78	(NOTE)	I.C. Programmed PAL
U83-U98, U103-U118	51NW9615S68	I.C. MB81256-12PZ (alternates are M5M4257L-12, part number 51NW9615T50; and M5M4257L-15, part number 51NW9615T56)
U99	51NW9615H37	I.C. SN75174N
U101	51NW9615H38	I.C. SN75175N
U119	51NW9615R68	I.C. Z8530APC
U120,U121	--	Not used.
Y1	48AW1015B09	Crystal oscillator, 16 MHz ± 0.01%
Y2	48NW9606A55	Quartz crystal, 32.768 kHz ± 0.002 %
Y3	48AW1015B11	Crystal oscillator, 25 MHz ± 0.01% (on MVME133)
Y3	48AW1015B17	Crystal oscillator, 33.3333 MHz ± 0.01% (on MVME133-1)
--	09NW9811A46	Socket, crystal oscillator, 4-lead (used with Y3)
--	67NW9415A17	Kit, ejector handle, 6U component
--	64-W5001B01	Panel, front, MVME133
--	33-W5089B20	Nameplate, Scanbe, MVME133 (on MVME133)

TABLE 5-4. MVME133 and MVME133-1 Module Parts List (cont'd)

REFERENCE DESIGNATION	MOTOROLA PART NUMBER	DESCRIPTION
--	33-W5089B39	Nameplate, Scanbe, MVME133-1 (on MVME133-1)
--	33-W5089B01	Nameplate, Scanbe, logo
--	42NW9401B14	Captive collar screw (2 req'd)
--	03NW9004B48	Screw, captive, M2.5 (2 req'd)
--	56NW9208A23	Bag, static shield, 12 inches x 18 inches

NOTE: When ordering, use number labeled on part.

5

5.4 SCHEMATIC DIAGRAM

Figure 5-2 illustrates the schematic diagram for the MVME133 and MVME133-1 modules.

THIS PAGE INTENTIONALLY LEFT BLANK.

A

APPENDIX A

MC68020 MASK SET PROBLEMS

1. ERRATA SHEET FOR MC68020 WITH MASK NUMBER 0A45J

The 0A45J mask of the MC68020 is used for many MVME133 modules. The 0A45J has the following known errors:

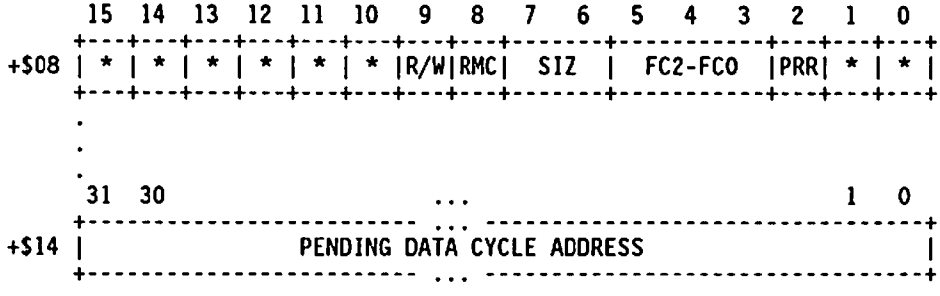
1.1 LINK A7,Disp will load A7 with an indeterminate value. Note that this problem must be accounted for in software.

1.2 During power on reset, the RMC pin may be driven high. If local bus arbitration occurs during a cycle which is terminated with a [BERR*], RMC may be driven high for several cycles of [RMPUCLK] before being three-stated. Note that this problem is handled by the MVME133 hardware.

1.3 The coprocessor primitive "Evaluate EA and Transfer Data", where EA is immediate addressing and the size is byte, passes the operand to the coprocessor as a word. As such, the byte operand is passed in bits D16-D23 rather than in bits D24-D31. To fix this, replicate the byte of data in the upper and lower bytes of the word in which the immediate address is stored. Currently, the upper byte is defined as 0, and it will not cause any problems to the machine to have the data in those bits. Note that this problem must be accounted for in software.

1.4 CMP.x (A7)+,(Ax)+ may not work correctly in a virtual environment. If a write cycle initiated by a previous instruction overlaps into this instruction and [BERR*] is driven active low, the bus error stack may overwrite the source value for the CMP instruction. Note that this problem must be accounted for in software.

1.5 There is a problem with the RTE instruction, when returning from a bus error or address error service routine, that may cause the MPU to hang up. Under certain conditions, an extra data cycle software fixup must be performed to permit proper execution of an RTE and avoid hangup. The bus cycle information for the software fixup is found in the internal register at the stack offset +$08, while the address for the bus cycle is in the internal register at the stack offset +$14.



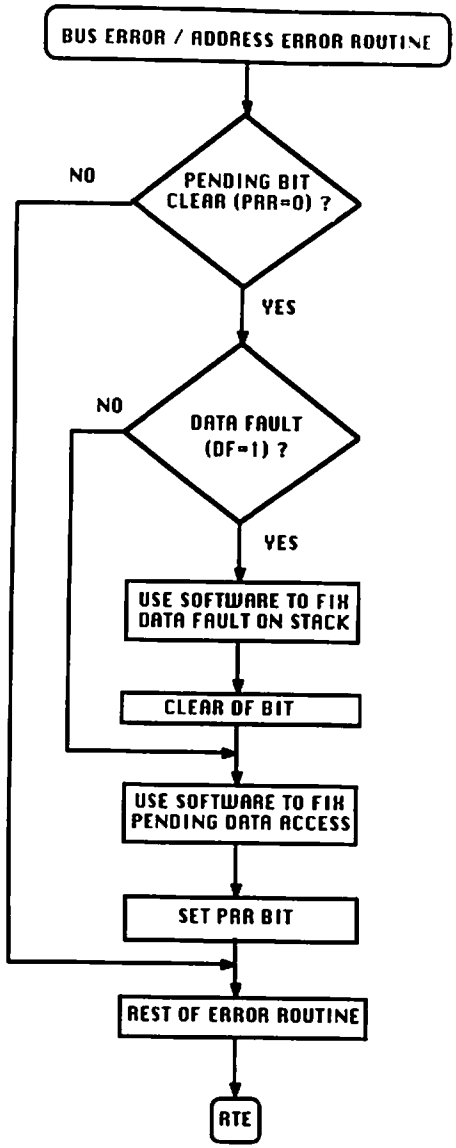
A

To determine if the extra fixup is required, check the Pending ReRun (PRR) bit. If the PRR bit is a zero, then the extra software fixup must be performed and the PRR bit must be set to a one before the RTE is executed; otherwise the processor will hang up. If the PRR bit is a one, no action is needed. The information for the extra data cycle fixup is shown on the previous page and is used in exactly the same way as the information in the Special Status Word. If the R/W bit indicates read (R/W = 1), use the address in offset +$14 to read data into the Data Input Buffer (offset +$2C). If the R/W bit indicates write (R/W = 0), use the data in the Data Output Buffer (offset +$18) to write to the Pending Data Cycle address found at offset +$14. Pending RMW cycles cannot occur in the current implementation, so the RMC bit can be ignored. SIZ encodings are:

%00 = 4 bytes,
%11 = 3 bytes,
%10 = 2 bytes,
%01 = 1 byte.

This is the same encoding as for the special status word. Do not alter any bits other than setting the PRR bit if it is a zero.

This fixup is for a pending bus cycle that was scheduled to be run after the access that caused the bus error or address error. Note that if the hardware fixup method is used to rerun the access that caused the [BERR*], the pending bus cycle access will be run before the hardware fixup access. This may cause bus sequence dependent data errors. To guarantee proper operation, the software fixup method should be used to rerun all data fault accesses, with the pending data fixup described above performed after the data fault fixup. The handling of instruction prefetch faults is not affected by this problem. The following flow should help clarify the fix procedure:



Note that the above problem must be accounted for in software.

A

2. ERRATA SHEET FOR MC68020 WITH MASK NUMBER 1A45J

The 1A45J mask of the MC68020 is used for many MVME133 modules. The 1A45J has the following known errors:

2.1 The CAS2.x instruction will not operate correctly if either of the operand effective addresses is specified to be data register indirect (Dn). The Rn1 and Rn2 registers must be address registers for the instruction to operate properly. If a data register is used, the address of the fetched operand(s) may be incorrect. In addition, the content of one or more of the data registers may be destroyed.

NOTE

The error described as item 2.2 applies to MC68020 chips with the 0A45J mask also.

2.2 The TAS instruction will not operate properly if either the read or the write portion of the RMC cycle is bus-errored. To correct this problem it is necessary to modify the content of one of the internal registers on the stack before an RTE is performed. The following code sequence shows one method of performing this fixup:

```
*  ... BUS ERROR HANDLER ROUTINE ...

        BTST.B      #0,$A(A7)          *TEST DATA FAULT BIT
        BEQ.S       NOPROB             *IF BIT IS 0
        BTST.B      #7,$B(A7)          *TEST RMC BIT
        BEQ.S       NOPROB             *IF BIT IS 0
        BFEXTU      $1C(A7){0:10),D0   *FIRST TEN BITS OF OPCODE
        CMPI.L      #$12B,D0           *TAS = 0100 1010 11XX XXXX
        BNE.S       NOPROB             *IF DOES NOT COMPARE
        BFTST       $1C(A7){10:3)      *OPCODE MODE BITS
        BEQ.S       NOPROB             *IF MODE = DN = 000
FIX     MOVE.L      #$80,$50(A7)       *FIX INTERNAL REGISTER
*  ...
NOPROB  RTE
```

2.3 The BTST instruction when used as follows will not operate correctly:

```
BTST  Dn,#< >
```

This will not do the bit test, but will perform an extraneous write.

APPENDIX B

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, a number of 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. 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 of them 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.

B

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

PIN NUMBER	SIGNAL MNEMONIC	SIGNAL NAME AND DESCRIPTION
5	CTS	CLEAR TO SEND - Clear to send is a function supplied to the terminal by the modem, and 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 - Data set ready 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 is active.
23		Not used.
24	TXC	TRANSMIT CLOCK - Same as TXC on pin 15.

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 for 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

B

with only three wires. This is based on the fact that 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.

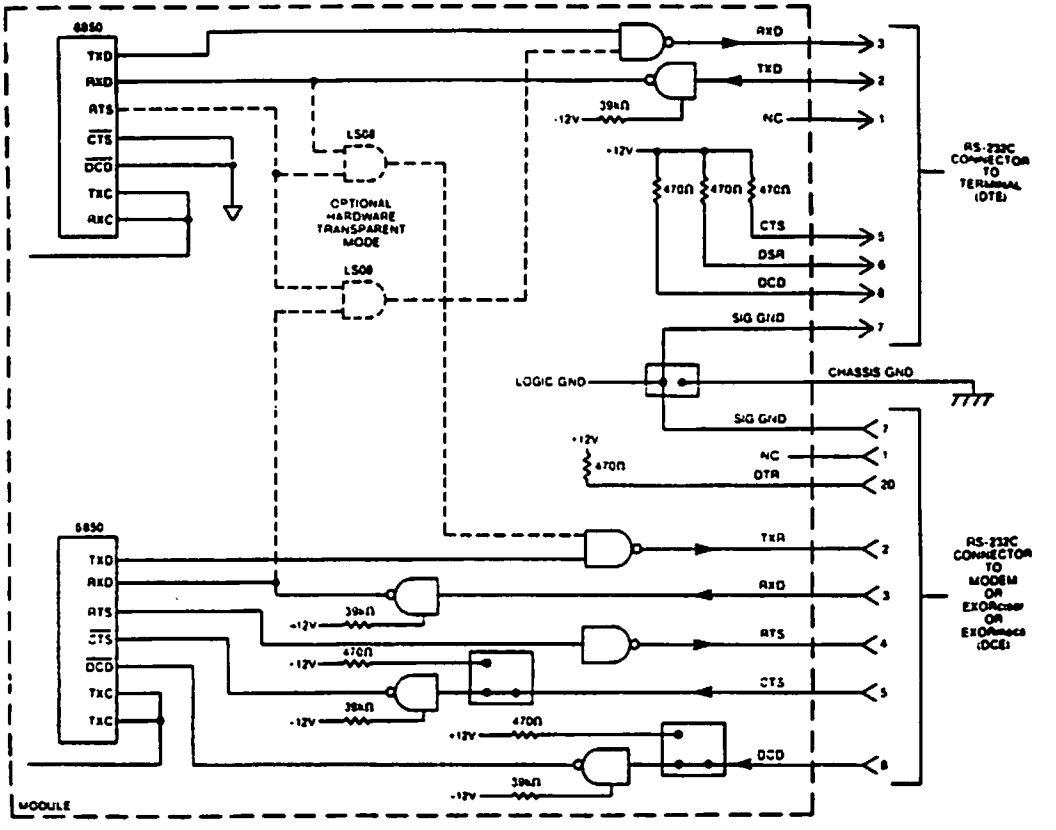


FIGURE 1. Middle-of-the-Road RS-232C Configuration

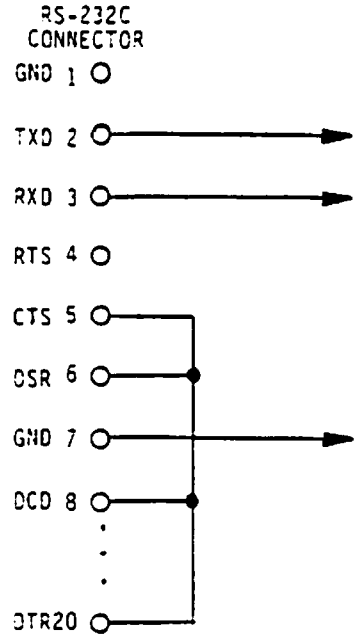


FIGURE 2. Minimum RS-232C Connection

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.

THIS PAGE INTENTIONALLY LEFT BLANK.

INDEX

THIS PAGE INTENTIONALLY LEFT BLANK.

SUGGESTION/PROBLEM REPORT

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

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

Product: _____ Manual: _____

COMMENTS: _____

(For additional comments use other side)

Please Print

Name _____ Title _____

Company _____ Division _____

Street _____ Mail Drop _____

City _____ Phone _____

State _____ Zip _____ Country _____

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

Motorola Field Service Division/Customer Support
(800) 528-1908
(602) 438-3100

MOTOROLA

COMMENTS: _____