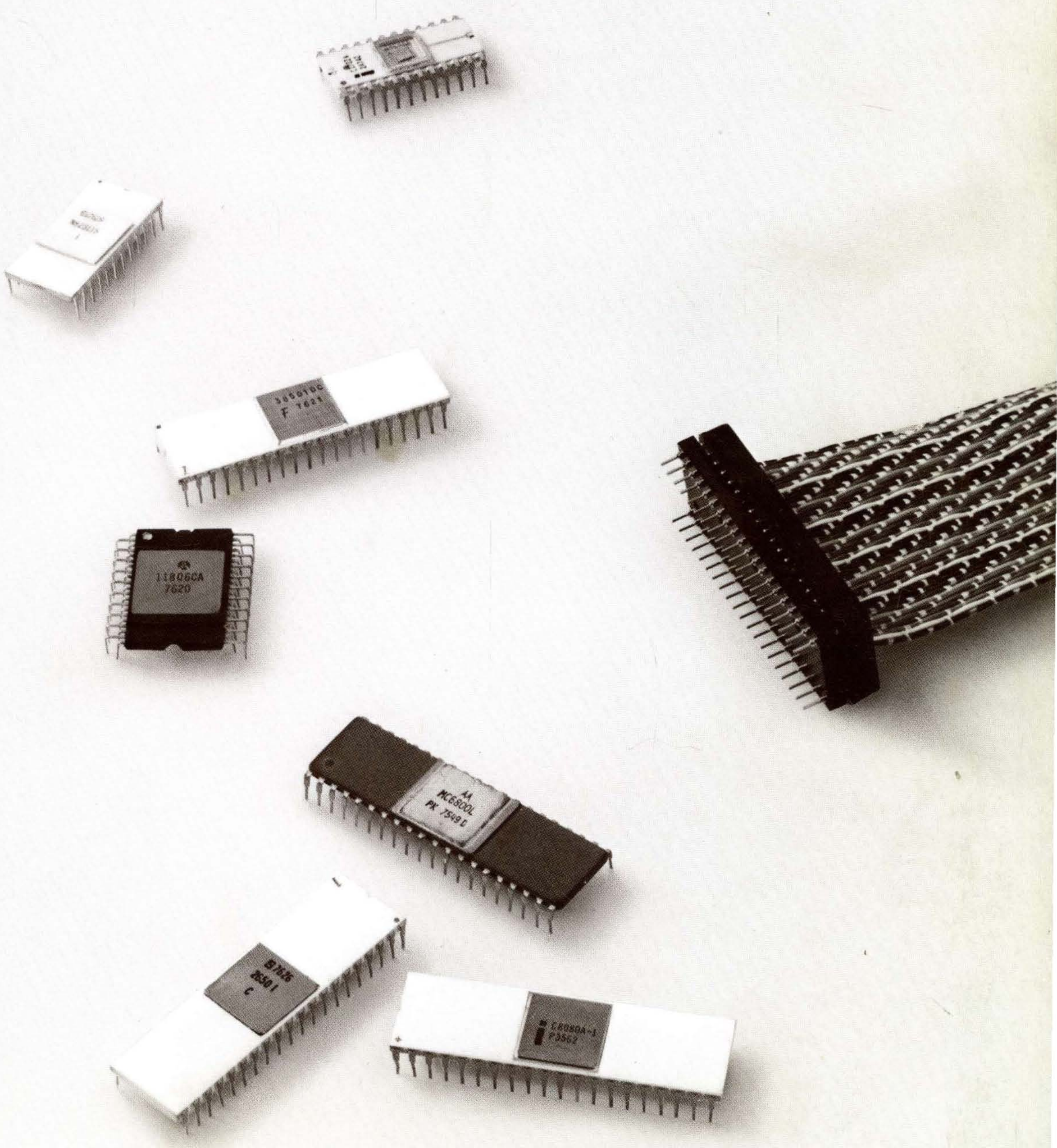
MILLENNIUM INFORMATION SYSTEMS, INC.

UNIVERSAL ONE

Microcomputer Development System

Hardware Reference Manual



UNIVERSAL ONE

Microcomputer Development System

Hardware Reference Manual

Document No. UO1-0000-11
Issued. Dec. 1976

MILLENNIUM INFORMATION SYSTEMS, INC.
420 MATHEW STREET, SANTA CLARA, CA 95050, 408-243-6652

TABLE OF CONTENTS

Chapter 1 INTRODUCTION

Page

1.0 Universal One System 1-1
1.1 Computer Components 1-2
1.2 Development Software 1-5
1.3 Universal Emulator 1-6
1.4 Universal Emulator Software 1-8
1.5 Purpose of this Manual 1-8
1.6 Manual Contents 1-8

Chapter 2 DEVELOPMENT COMPUTER DESCRIPTION

Page

2.0 Computer Architecture 2-1
2.1 Memory Organization 2-2
2.2 Bus Structure 2-4
2.3 Debug Hardware 2-6
2.4 I/O Provisions 2-10
2.5 Development Computer Subassemblies 2-10
2.5.1 Master CPU Module 2-11
2.5.2 Slave CPU and Emulation Interface 2-11
2.5.3 Debug and Front Panel I/O Module 2-12
2.5.4 Master Memory 2-13
2.5.5 Slave Memory 2-13
2.5.6 4K RAM/2K PROM Module 2-14
2.5.7 16K Dynamic RAM Module 2-14
2.5.8 General Purpose I/O Module 2-15
2.5.9 PROM Programmer Modules 2-15
2.5.10 Hardware Analyzer Module 2-16
2.5.11 Emulation Cable 2-17

Chapter 3 MASTER CPU MODULE

Page

3.0 Introduction 3-1
3.1 General Description 3-1
3.1.1 The CPU 3-2
3.1.2 I/O Logic 3-2
3.1.2 Interrupt Logic 3-4
3.1.4 DMA Logic 3-4
3.2 Detailed Description 3-5
3.2.1 2650 CPU Support 3-5
3.2.2 I/O Logic 3-8
3.2.3 Interrupt Logic 3-14
3.2.4 DMA Logic 3-14
3.2.5 Baud Rate and Timer Logic 3-18
3.3 Utilization 3-18
3.3.1 Installation 3-18
3.3.2 Switch and Jumper Options 3-22
3.3.3 Interface and Connector Pinouts 3-22

TABLE OF CONTENTS (Continued)

Chapter 4 DEBUG AND FRONT PANEL I/O MODULE **Page**

4.0 Introduction 4-1
4.1 General Description 4-1
4.1.1 Master/Slave Control 4-2
4.1.2 Debug Features 4-2
4.1.3 I/O Commands and Interrupts 4-6
4.1.4 Front Panel Interface 4-6
4.2 Detailed Description 4-8
4.2.1 Master/Slave Control and Halt Detector 4-8
4.2.2 I/O Commands and Interrupts 4-10
4.2.3 Debug Features 4-17
4.2.4 Front Panel Interface 4-21
4.3 Utilization 4-23
4.3.1 Installation 4-23
4.3.2 Switch and Jumper Options 4-25
4.3.3 Interface and Connector Pinouts 4-25

Chapter 5 RAM/PROM MEMORY MODULE **Page**

5.0 General Description 5-1
5.1 Detailed Description 5-1
5.1.1 Memory Elements 5-1
5.1.2 PROM Addressing 5-2
5.1.3 RAM Addressing 5-2
5.1.4 Timing 5-2
5.1.5 -9V Regulator 5-4
5.2 Utilization 5-4
5.2.1 Installation 5-4
5.2.2 Switch and Jumper Options 5-5
5.2.3 Interface and Connector Pinouts 5-7

Chapter 6 16K DYNAMIC RAM MODULE **Page**

6.0 Introduction 6-1
6.1 General Description 6-1
6.2 Detailed Description 6-6
6.2.1 RAM Storage 6-6
6.2.2 PROM Memory 6-6
6.2.3 Module Select and System Request Logic 6-8
6.2.4 Byte/Word Control 6-10
6.2.5 Address Control 6-10
6.2.6 Data-Parity Control 6-14
6.2.7 Refresh Request Logic 6-16
6.2.8 Operation Control and Timing 6-19
6.3 Utilization 6-20
6.3.1 Installation 6-20
6.3.2 Switch and Jumper Options 6-22
6.3.3 Interface and Connector Pinouts 6-26

TABLE OF CONTENTS (Continued)

Chapter 7 GENERAL PURPOSE I/O MODULE **Page**

7.0 Introduction 7-1
7.1 General Description 7-1
7.2 Detailed Description 7-3
7.2.1 Serial I/O Ports 7-3
7.2.2 Module and Port Address Control 7-8
7.2.3 Parallel Input and Output Ports 7-8
7.2.4 Interrupt Logic 7-10
7.3 Utilization 7-10
7.3.1 Installation 7-10
7.3.2 Switch and Jumper Options 7-12
7.3.3 Interface and Connector Pinouts 7-18

Chapter 8 1702A AND 82S115 PROM PROGRAMMER MODULES **Page**

8.0 Introduction 8-1
8.1 General Description 8-2
8.2 1702A Detailed Description 8-5
8.2.1 Operation 8-5
8.2.2 Data Control 8-6
8.2.3 Power Supplies 8-8
8.2.4 Sequencer and Voltage Switches 8-10
8.3 82S115 Detailed Description 8-14
8.3.1 Operation 8-14
8.3.2 Data Control 8-16
8.3.3 Power Supplies 8-17
8.3.4 Sequencer and Voltage Switches 8-18
8.4 Utilization: 1702A 8-22
8.4.1 Installation 8-22
8.4.2 Switch and Jumper Options. 8-24
8.4.3 Interface and Connector Pinouts. 8-24
8.5 Utilization 8-24
8.5.1 Installation 8-24
8.5.2 Switch and Jumper Options 8-26
8.5.3 Interface and Connector Pinouts 8-26

Chapter 9 SYSTEM BUS **Page**

9.0 General Description 9-1
9.1 Bus Signal Definitions 9-1
9.2 Motherboard Pin Lists 9-8

TABLE OF CONTENTS (Continued)

Chapter 10 FRONT PANEL **Page**

10.0 Introduction 10-1
10.1 Functional Description 10-1
10.1.1 Address, Data Entry and Display 10-3
10.1.2 System 10-4
10.1.3 CPU Control Swtiches 10-4
10.1.4 Bus Control Switches 10-5
10.1.5 Bus Status Indicators. 10-6
10.1.6 CPU Status Indicators. 10-6
10.1.7 System Status Indicators 10-6
10.1.8 PROM Programming Sockets and Power Switches 10-6
10.2 Theory of Operation 10-7
10.2.1 Data and Address Logic 10-10
10.2.2 Bus Access Logic 10-12
10.2.3 CPU Control Logic 10-13
10.2.4 System Control 10-13

Chapter 1
INTRODUCTION

1.0 UNIVERSAL ONE SYSTEM

The Universal One development system is a complete general purpose tool, supporting in turn - key fashion all phases of user software and hardware development. Its basic structure and capability is similar to a general purpose minicomputer system but with two very important added features. It provides multiple in-circuit emulation capability for exercising and debugging user programs and prototype systems and it contains both bipolar and MOS PROM programming capability integral to the system.

The system components, illustrated in figure 1-1 consist of the Development Computer, the Dual Floppy Disk unit, the Console (CRT or TTY) and one or more Emulation cables. Emphasis has been placed on universality and ease of operation. Due to its Master-Slave structure new microprocessors can be supported with the addition of a single printed circuit module. Basic software and hardware remains unchanged. Thus, the system becomes more valuable with time, supporting more processors rather than becoming obsolete as technology advances.

Ease of operation is insured through use of an interrupt driven floppy disk based operating system. The software features include:

- Floppy Disk Operating System
  - Comprehensive system monitor
  - Disk file manager
  - Text editor

- Comprehensive Debug Software
  - Breakpoint detection
  - Program tracing
  - Program stepping
  - Static state processor control

Comprehensive Diagnostic Package

- System peripherals
- Memory diagnostics
- CPU diagnostics
- System diagnostics

The dual expandable Disk System is controlled by its own internal microprocessor relieving the computer system of time consuming hardware related functions and error checking.

In-Circuit Emulation to debug the user prototype is as easy as unplugging the microprocessor chip in the user breadboard and plugging the Emulation Cable into its socket. Full debug capability with system RAM and user I/O or user memory and I/O is supported. The cable actually operates with the real microprocessor, not a simulation, allowing the user to isolate timing and interface problems related to the processor as well as his peripheral devices.

1.1 COMPUTER COMPONENTS

The Development Computer is completely modular with a bus oriented structure. The Computer contains the following components:

- MASTER CENTRAL PROCESSING UNIT, based on the 2650, which supervises the overall system and contains the basic system I/O interface (Console and Disk Subsystems).

- PROTECTED 16K BYTE MASTER MEMORY containing the system software.

- SLAVE PROCESSING UNITS, containing 8 or 16 bit microprocessors, which execute user programs and interface the Emulation Cable.

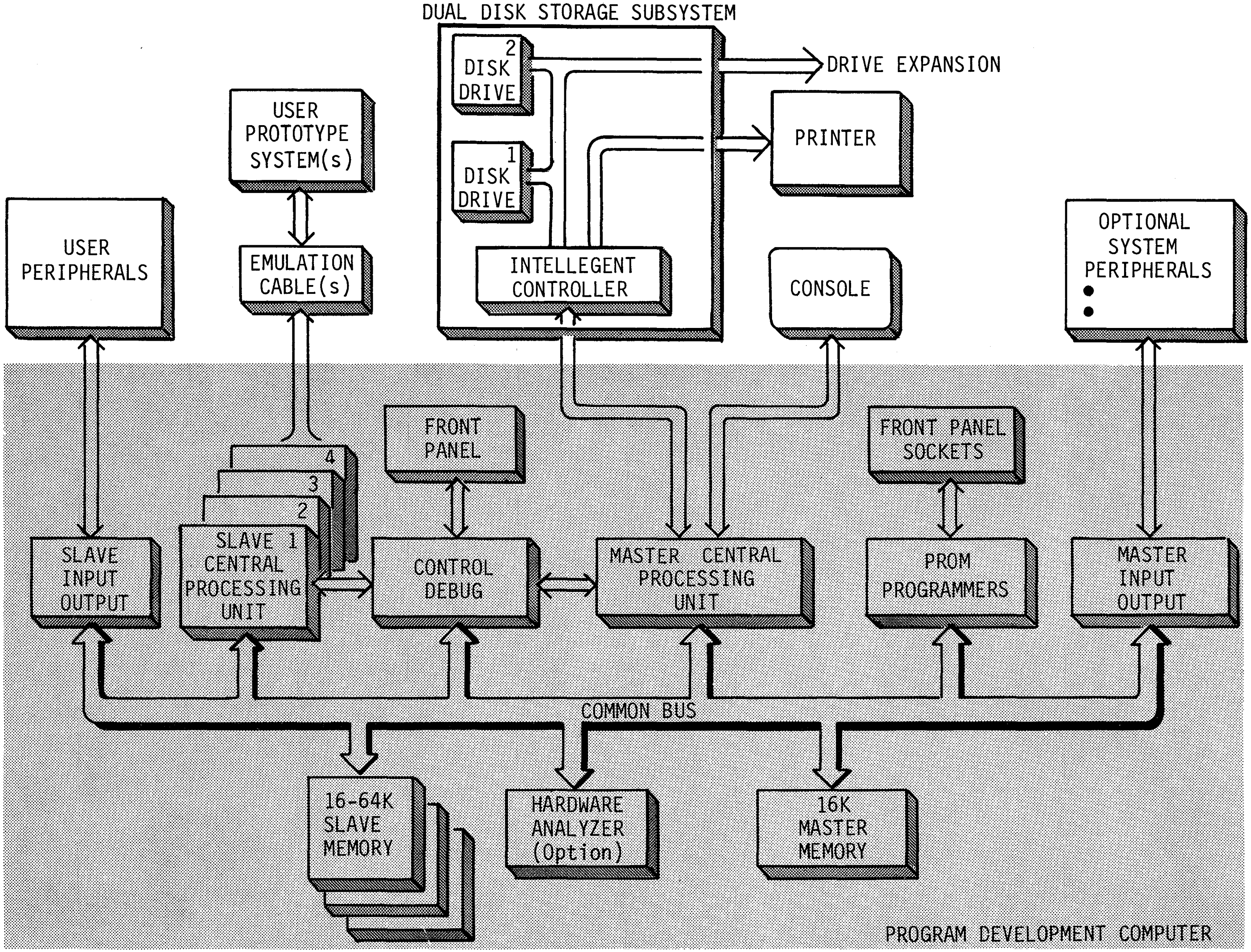


FIGURE 1-1 THE UNIVERSAL ONE SYSTEM

- 16K BYTE/WORD SLAVE RAM MEMORY expandable to 64K byte for user programs. Any mixture of RAM and ROM may be used.

- EMULATION CABLE(s) which replace user microprocessor with Slave processor to allow full debugging of user hardware.

- DEBUG MODULE which supports software debug features such as TRACE, BREAKPOINT and GO. It also interfaces both front panels.

- HARDWARE ANALYZER (option) Provides expanded breakpoints, 128 transaction real time trace storage and User memory mapping.

- GENERAL PURPOSE I/O MODULE (option) Contains RS-232 serial interface and custom configurable parallel 8 bit ports. It may be used for additional system peripherals such as a Modem or High Speed Paper Tape Reader, or for user peripherals.

- PROM PROGRAMMERS (option) Three front panel sockets and two independent programming modules provide erasable MOS PROM and bipolar PROM programming capability.

The unit physically consists of a 20 slot card file for 11" x 7" circuit modules, a snap on front panel, a power supply module containing reserve power for custom user cards, and a dual fan system for reliable cooling. The unit is extremely modular for ease of replacement of any component.

The card file contains a universal bus structure to all positions except that the bus is split into Master-Slave halves with control and interrupt lines unique to each half. Master and Slave processors and their memory and I/O may be placed anywhere within their designated halves. The motherboard contains an auxiliary power bus for special user voltage requirements and it contains a 16 bit data bus to accommodate 16 bit Slave processors and memory.

A Full Display Front Panel can be substituted for the Standard Front Panel at any time. Existing interface is provided by the debug module. The panel contains hexadecimal data and addresses displays, paddle handle address and data entry switches, memory and I/O examine and alter functions, as well as single step and system control functions. Back lit displays indicate system states.

1.2 DEVELOPMENT SOFTWARE

System software provided with the Prototype Development System includes the Millennium Universal Disk Operating System (UDOS) and the Text Editor. The Assembler and the Debug software are provided with each Slave supported by the system.

UDOS -- UDOS relieves the user of the necessity of understanding detailed internal operation of the system. It provides complete control over operation of all portions of the Prototype Development System. All functions relating to file handling, loading and execution are monitored and controlled by UDOS, including in-circuit emulation and PROM programming functions. UDOS resides in a dedicated memory consisting of 256 byte PROM and 16K RAM running under the Master CPU.

The UDOS software allows the user to create, edit, and assemble files; obtain object and listing outputs; load and execute programs; and, through the debug system, check out programs in an efficient manner.

UDOS provides a powerful command file capability that enables the user to create customized operating system commands.

UDOS controls the multi-drive Floppy Disk Subsystem, a Line Printer, and the CRT Console, which may be an ASR-33 TTY or an RS-232 compatible device. Software drivers are provided within UDOS for these I/O devices. The UDOS software provides a flexible input/output system that enables the user to dynamically assign any logical channel to any physical device or file within the system. In this way, system I/O devices may be dynamically assigned using UDOS commands either from the console or from within a user's program. Thus, the user may write his own driver for other peripheral devices and link them into the UDOS system by use of the optional General Purpose I/O Card.

TEXT EDITOR -- The Text Editor is a comprehensive software package that allows the user to enter and modify text files. The Text Editor is line oriented and accepts inputs from the terminal or a disk file, performs modifications in a work space, and outputs the revised text to a disk file.

ASSEMBLER -- The Prototype Development System Resident Assembler translates symbolic assembly language instructions into appropriate machine language code.

The Assembler generates absolute object code. This code is in hexadecimal format and may be loaded into the system for direct execution or may be converted by an UDOS command to SMS format for PROM or ROM programming.

DEBUG -- The Debug system is a software program which provides the user with real-time program debug capabilities within both a software and hardware environment. It uses special hardware features built into the program development system to control execution of the user's program. User programs operating under the Debug system have dynamic program trace, breakpoint capabilities, and memory modification capabilities. Status reporting on the memory, the program, and the processor is also provided.

1.3 UNIVERSAL EMULATOR

The Universal Emulator (figure 1-2) containing the same universal architecture as the Universal One System is oriented toward hardware prototyping. The Universal Emulator is supported by a software package contained in PROM Memory rather than a Floppy Disk Subsystem. All the basic software requirements are contained in PROM Memory with provisions for

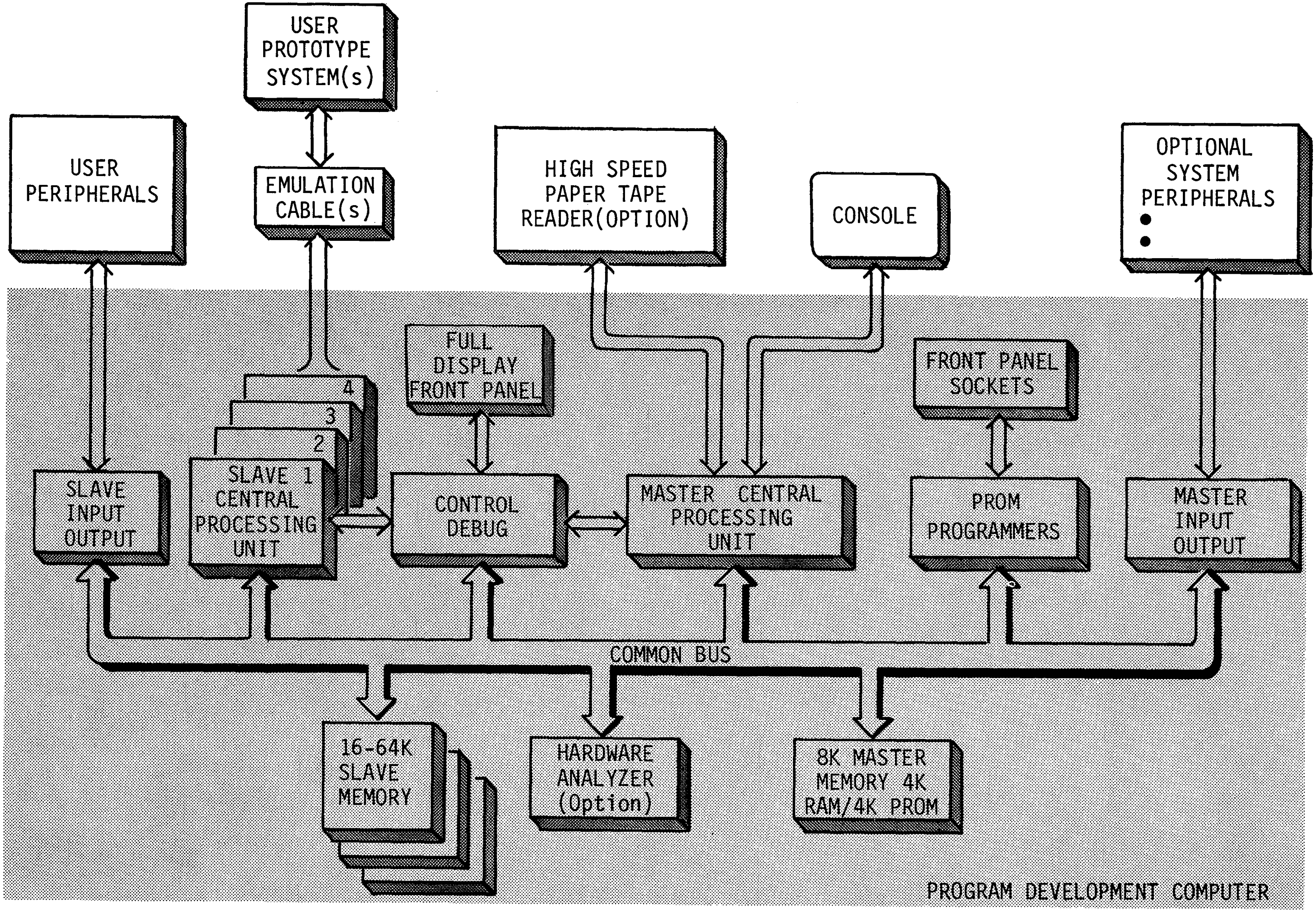

USER PROTOTYPE SYSTEM(s)
USER PERIPHERALS
EMULATION CABLE(s)
HIGH SPEED PAPER TAPE READER(OPTION)
CONSOLE
OPTIONAL SYSTEM PERIPHERALS
●
●
FULL DISPLAY FRONT PANEL
FRONT PANEL SOCKETS
4
3
2
SLAVE INPUT OUTPUT
SLAVE 1 CENTRAL PROCESSING UNIT
CONTROL DEBUG
MASTER CENTRAL PROCESSING UNIT
PROM PROGRAMMERS
MASTER INPUT OUTPUT
COMMON BUS
16-64K SLAVE MEMORY
HARDWARE ANALYZER (Option)
8K MASTER MEMORY 4K RAM/4K PROM
PROGRAM DEVELOPMENT COMPUTER

FIGURE 1-2 THE UNIVERSAL EMULATOR SYSTEM

additional capability to be loaded in the 4K of Master RAM Memory. The basic configuration consists of the Master CPU, 4K PROM Master Memory, 4K of RAM Master Memory, the Debug card, 4K of Slave Memory and a Slave CPU with the Emulation Cable. The Universal Emulator is also supplied with extended front panel capability.

1.4 UNIVERSAL EMULATOR SOFTWARE

The Universal Emulator software is broken down into three functional areas. Software features which are independent of the Slave processor are contained in the Master PROM memory. Software features which are optional and/or dependent on the Slave processor can either be loaded into Master RAM from the TTY or contained on PROM housed on the particular Slave CPU card.

1.5 PURPOSE OF THIS MANUAL

This manual is intended to acquaint an experienced logic designer with the Millennium Universal Microprocessor Development Systems, both in overall concept and architecture, and in detail. The descriptions of functions and the detailed descriptions of each module in the system should be sufficient to allow troubleshooting to the board level in case of failure. It is assumed that the reader has some knowledge of microcomputer development systems and knowledge of the characteristics of the Slave microprocessor being used in the product. For a detailed description of system operation refer to the Universal One or Universal Emulator Operator's Manual.

1.6 MANUAL CONTENTS

Chapter 2 of this manual provides a description of the Development Computer in general terms, and describes its architecture. The rest of the manual is devoted to describing each module used in the system, with functional descriptions and a theory of operation at the block diagram level. System peripherals are mentioned briefly in Chapter 2, but detailed descriptions and operating theory are not provided. The reader is referred to the manuals for each peripheral device for operating and maintenance.

Chapter 2

DEVELOPMENT COMPUTER DESCRIPTION

2.0 COMPUTER ARCHITECTURE

The Development Computer is based on a dual processor concept, as shown in figure 2-1. The Master CPU runs the Operating System and the Text Editor and performs all interactions with system I/O devices. The Slave CPU runs user programs as well as the Assembler. The Master CPU has completely separate memory and I/O, protected from user programs. This provides a crashproof Operating System and gives the user total use of the Slave Memory and I/O address space.*

The dual processor architecture permits different Slave CPU's to be supported by the system without requiring a re-write of the Operating System software. Only those portions of the system software unique to the Slave CPU being supported (the Assembler and portions of the Debugger) must be updated. Hardware provisions within the computer permit support of Slave CPU's of up to 16-bit word length and up to 64K address space.

Interaction of Master and Slave CPU's is controlled by the Debug module under Master CPU direction. Both CPU's share the basic bus structure, with only one processor active at any one time. Control is transferred to the Slave by a Master HALT. Control is returned to the Master by any system interrupt or a Debug interrupt such as a breakpoint.

The bus structure is essentially universal for all board locations in the card file, with minor exceptions. The Debug board splits the bus

* For the Slave CPU a 100 -1K byte segment of Slave Memory is required for Master/Slave interface when running the Slave CPU under Debug control. For the 8080, this is provided by a 1K PROM at the upper memory boundary.

into Master and Slave halves, with a separate interrupt structure for each half. The PROM Programmer boards have dedicated locations because of the special AC voltages required for programmer operation. Memory or I/O boards may be placed in any open position in the appropriate half of the card file.

The Emulator Cable assembly provides in-circuit emulation of user hardware. The cable connects to the Slave CPU on one end and essentially replaces the CPU in user hardware on the other end. This gives total debugging capability in an actual hardware environment.

The Master CPU contains provision for the basic system I/O which consists of the Floppy Disk and Printer interface and the system Console interface. General Purpose I/O modules can provide a full EIA RS-232-C interface and parallel I/O ports to interface additional system peripherals.

With UDOS, the Slave CPU can utilize the system peripherals through Service Calls (SVC's) to the operating system. SVC's are described in Chapter 9 of the Operator's Manual. Also special user peripherals under direct Slave control may be interfaced with the use of a General Purpose I/O module.

2.1 MEMORY ORGANIZATION

For UDOS, Master Memory consist of 16K of RAM, with the first 256 bytes overridden by PROM which contains the system bootstrap program. When the system is initialized, the Master accesses the bootstrap to load the operating system from the system diskette into Master Memory. For the Universal Emulator, Master Memory consists of 4K PROM and 4K RAM.

Slave Memory contains 16K bytes of RAM in the UDOS system and 4K bytes in the Emulator system, both expandable to 64K bytes. The Slave has access to Slave Memory only, while the Master has access to both memories. A 16K byte portion of Slave Memory is assigned under program control to

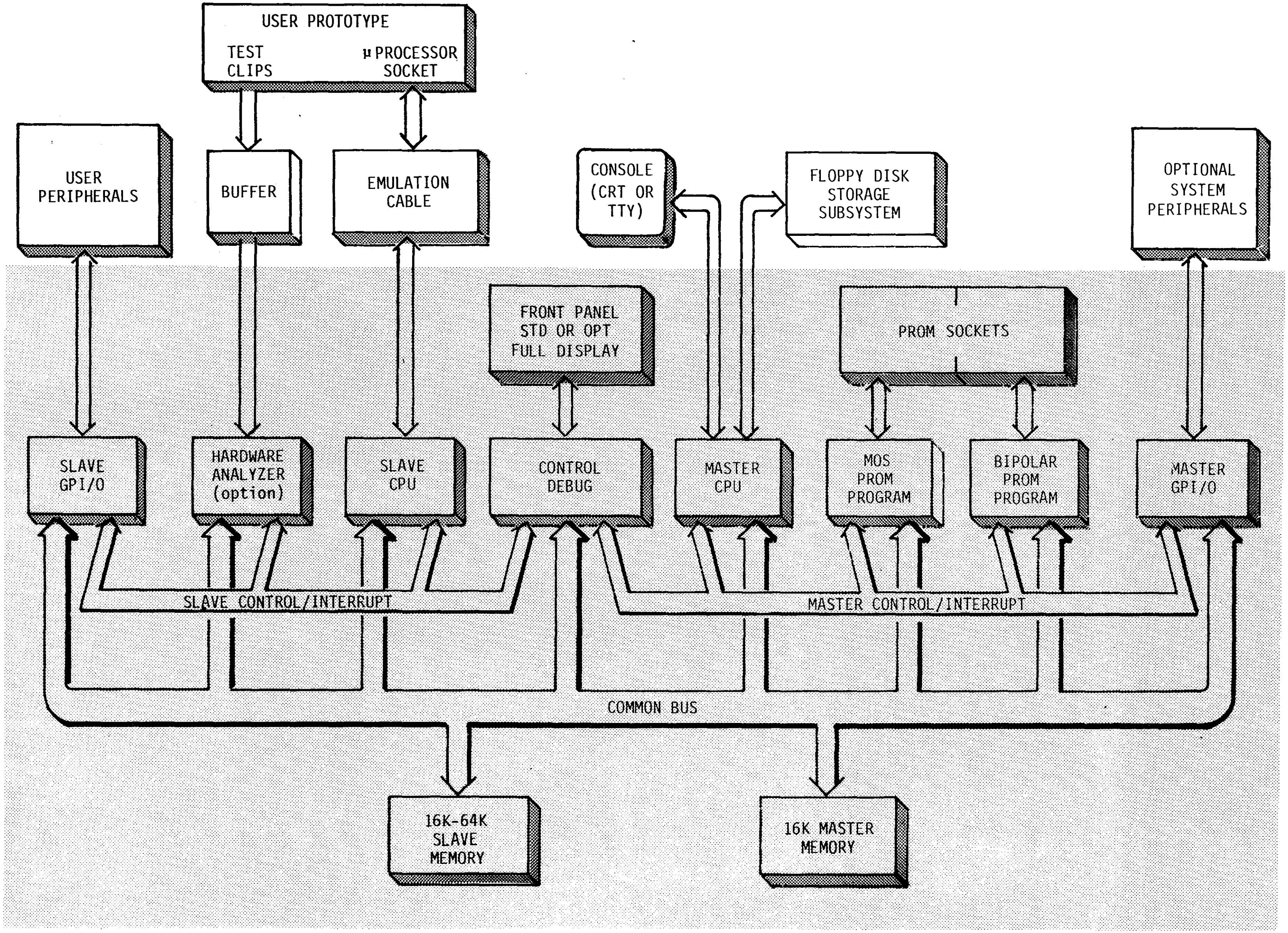


FIGURE 2-1 UNIVERSAL ONE SYSTEM BLOCK DIAGRAM

the Master's 16K-32K address space as illustrated in figure 2-2. The ability of the Master to "bank switch" Slave Memory into its address space allows communication for I/O Service Calls, Debug Trace operation, and Text Editor operation.

2.2 BUS STRUCTURE

The Computer bus structure is shown in figure 2-3. The bus is split into Master and Slave sections by the Debug and front panel I/O Logic board. The bus is essentially universal in that data, address, and control lines are paralelled to all boards. The exceptions to this structure are the independent Debug control and interrupt lines for the Master and Slave, and special PROM AC power lines. The bus structure allows freedom of Memory, CPU and I/O module placement within each half of the bus. I/O and Memory address assignment is switch selected on the individual boards.

The bus consists of 100 lines made up of the following signals:

<u>COMMON BUS LINES</u>

- 16 Address lines
- 16 Data lines
- 22 Control lines:
  - 9 Memory and I/O Control lines
  - 8 System Control lines
  - 3 Debug lines
  - 2 Sense/Flag lines
- 2 Interrupt
- 20 Power and Ground lines
- 3 Spare

<u>UNIQUE LINES</u>

<u>SLAVE HALF</u>

- 6 Debug lines
- 13 Interrupt-related lines
- 2 Spare lines

<u>MASTER HALF</u>

- 4 Debug lines
- 17 Interrupt-related lines

<u>PROM PROGRAMMER SECTION</u>

- 14 AC Voltage lines

Four of the 20 power and ground pins provide an auxiliary power bus which can be connected via rear-panel terminals to an external power supply for

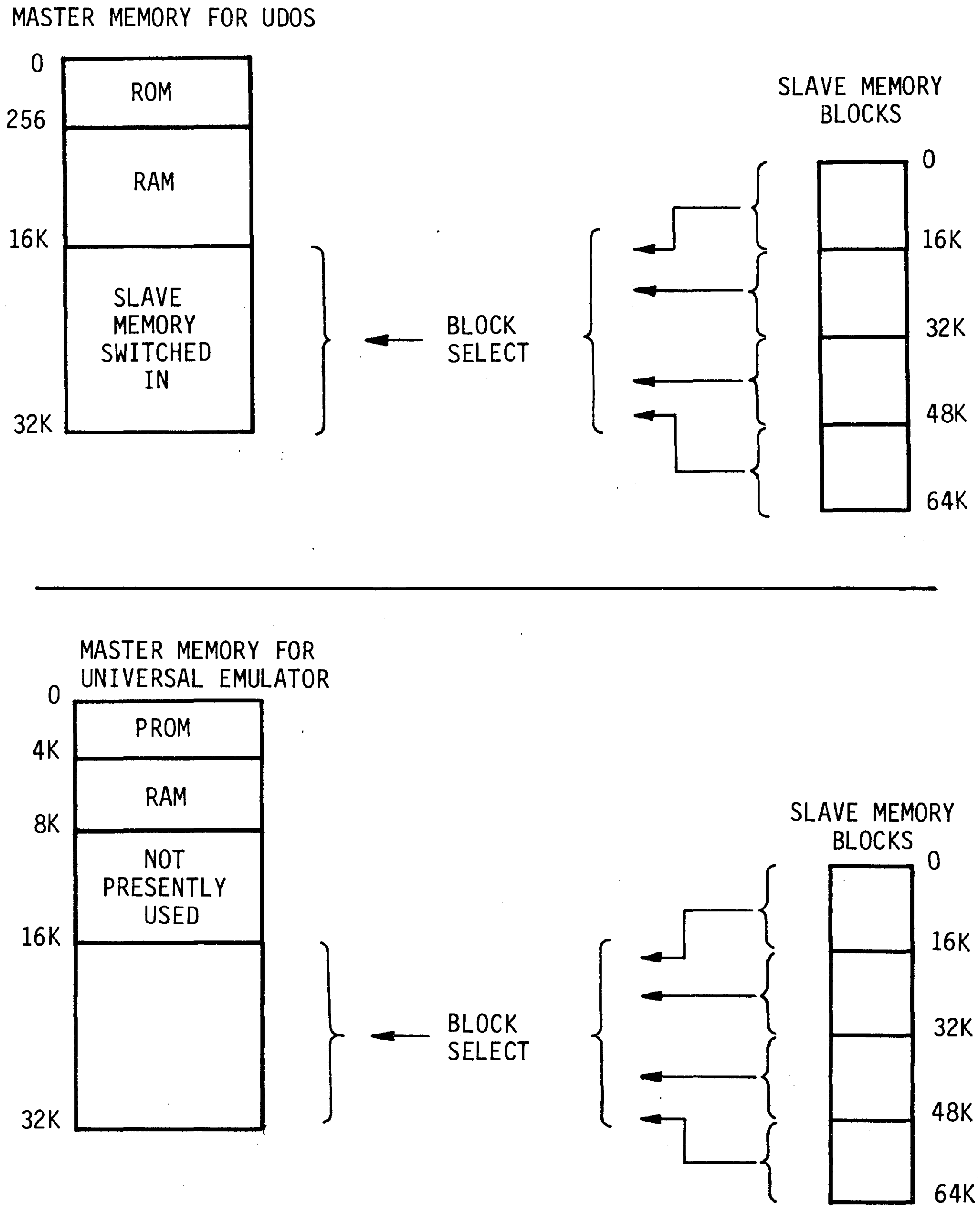


FIGURE 2.2 MEMORY ORGANIZATION

user requirements not met by the standard system supplies. The AC voltages required by the PROM programmers extend to the end board positions only and mechanical keying prevents plugging any other boards into those positions. All board connectors are offset to prevent plugging boards in backwards.

A complete description of all bus signals is contained in chapter 9.

2.3 DEBUG HARDWARE

Hardware is contained on both the Debug module and Slave CPU module to facilitate the following major system Debug features:

- Master/Slave interaction
- Breakpoints
- Forced Slave Jumps

Master/Slave Interaction -- All Debug hardware communication to the Master CPU is by way of interrupts. The Master reliquishes control to the Slave CPU by performing a HALT instruction. Control is regained by the Master when it receives an interrupt. Interrupts may be from system I/O or from the Debug hardware. The Debug module contains its own interrupt priority decoder and vector generator for Debug and Service call interrupts.

Breakpoints -- Two breakpoint registers are contained on the Debug module. The breakpoint addresses are loaded by the Master CPU under command from the user. When a breakpoint address is identical to the current Slave CPU address, the Master CPU is interrupted before the next Slave instruction fetch. Both the "last" instruction address and the "next" instruction address are stored. This allows the software to examine the Slave's program and interpret op-codes for the trace printout. It also allows the system to restore the Slave to its original address after executing a

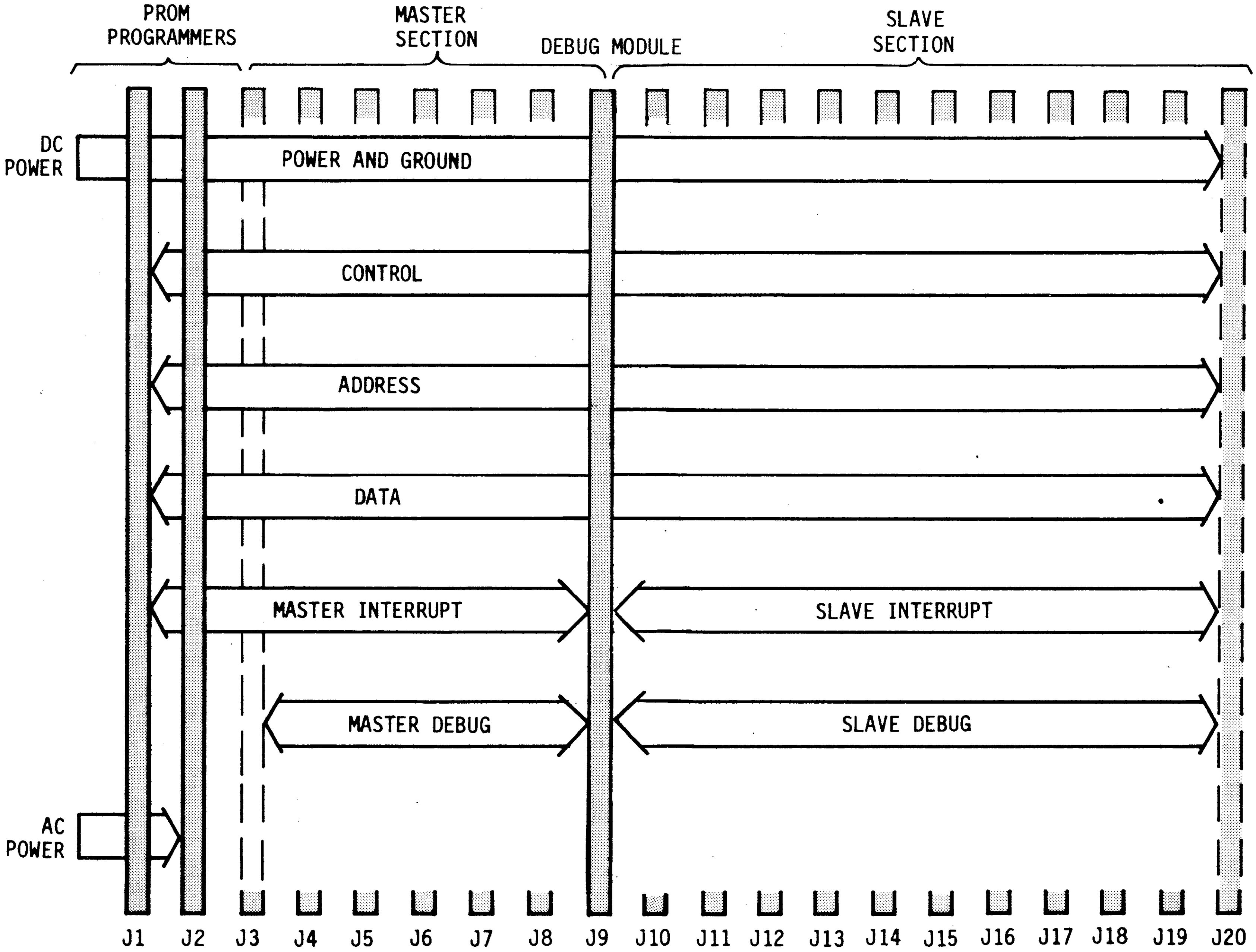


FIGURE 2-3 BUS STRUCTURE

register dump routine.

Forced Slave Jumps -- Forced Slave Jumps are required for three reasons:

1) To start execution of the user's program at any commanded location.

2) To allow Slave execution of register dump and register restore routines for trace.

3) To allow Slave execution of commands relating to user memory modification.

The jump address is contained in a register in the Debug module and the Forced Jump logic is contained on the Slave CPU module.

Emulation Cable -- The Emulation Cable allows the user to exercise his hardware while still retaining the Debug features of the development system. The cable connects to the Slave CPU on one end and plugs into the user's prototype microprocessor socket on the other. The cable contains line drivers and receivers close to the user end to insure signal integrity and minimize user circuit loading.

The Slave CPU essentially acts as the user's microprocessor. There are two basic in-circuit modes of operation illustrated in figure 2-4a, b and c. In Mode 1, the user's I/O is exercised while the program is executing out of the computer Slave Memory. Full Debug and trace operations are available. In Mode 2, both the user's I/O and memory are utilized, still allowing breakpoints, single step, and trace. Multiplexers on the Slave board switch between the cable interface and the computer bus for each mode. The cable can remain installed in the normal Slave CPU mode with no adverse effects. An optional Hardware Analyzer Module allows mapping Slave Memory in 256 byte blocks to either the user prototype system or the computer system while in Emulation mode 1.

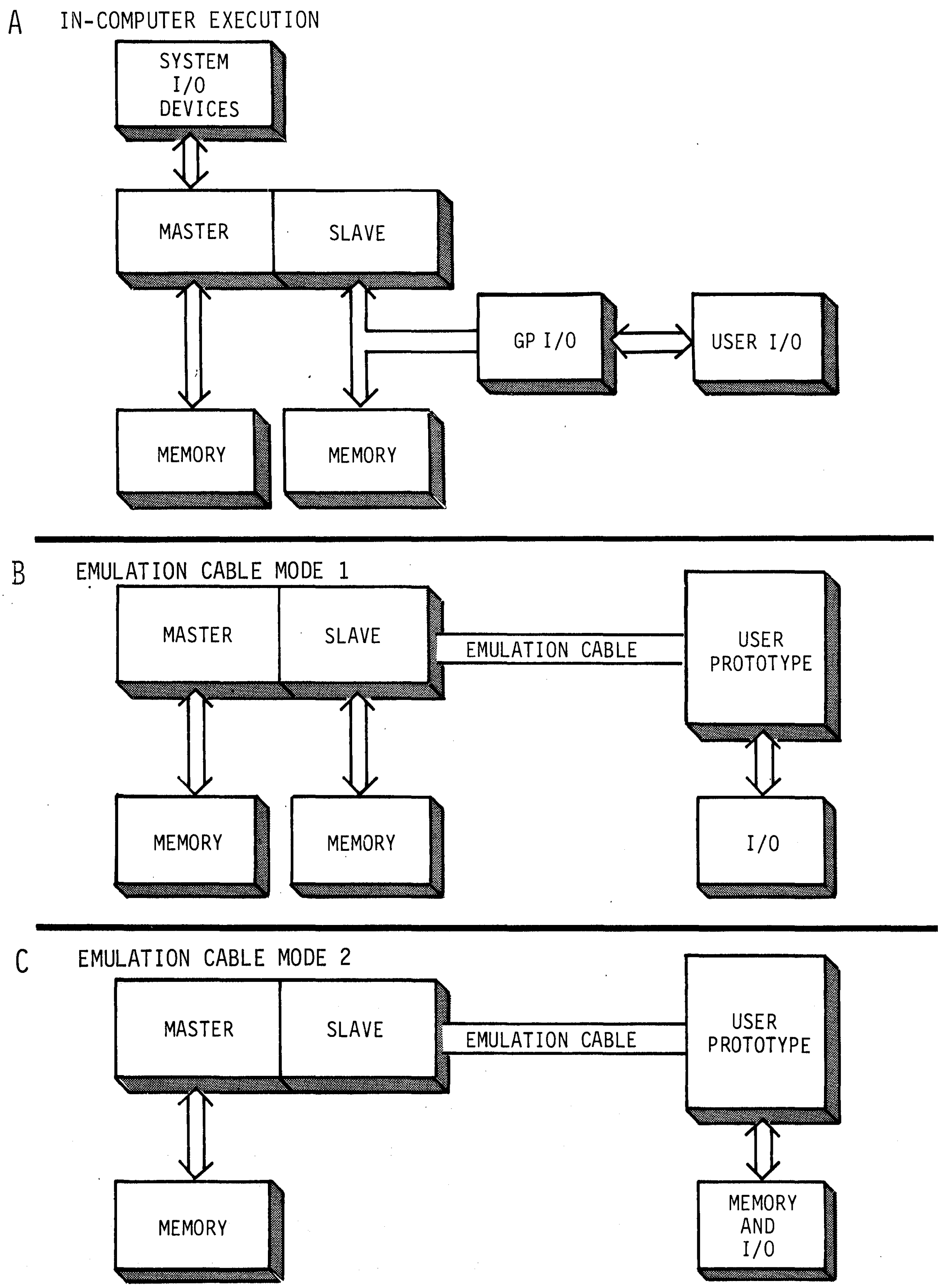


FIGURE 2-4 EMULATION MODES

2.4 I/O PROVISIONS

The basic computer I/O consists of the Floppy Disk and Console interface as shown in figure 2-1. These are both contained on the Master CPU. The Floppy Disk interface uses 8 bit parallel input and output ports for data and control. A 40 pin ribbon cable on the top edge of the Master connects directly to the Floppy Disk System.

The console (normally CRT or TTY) is a serial interface providing a limited EIA RS-232-C interface and a 20 ma TTY current loop interface. Baud rates and data formats are switch selectable for TTY or EIA operation. Additional switch positions select the EIA Baud rate for 150-2400. A 25 pin standard EIA connector on the rear panel connects through a ribbon cable to the top edge of the Master CPU board.

The General Purpose I/O board allows both system I/O expansion and user in computer I/O capability. The board contains a full RS-232-C interface, four sets of 8 bit parallel I/O ports, and eight interrupt flip-flops. This allows a hardware interface to most TTL-compatible peripherals. By inserting the card in the Slave side and setting a control switch, the card responds to Slave commands only and can operate the 8 level Slave priority interrupt structure.

2.5 DEVELOPMENT COMPUTER SUBASSEMBLIES

As shown in figure 2-1, the development computer is comprised of functional modules, with each module consisting of a separate printed circuit board. The computer is divided both functionally and physically into two sections: Master and Slave. The Master section includes the Master CPU, Master Memory (consisting of 4K or 16K memory boards) and PROM Programmer modules. Common to both Master and Slave sections is the Front Panel I/O and Debug module. The Slave section includes one or more Slave CPU modules and Slave Memory (consisting of 4K or 16K memory boards). General Purpose I/O modules may be included in the Master or Slave sections. All of the modules plug into the Motherboard which carries Power, Address, Data and Control buses.

2.5.1 Master CPU Module.

The Master CPU module functions as the control for the system. This module contains clock circuitry for its 2650 microporcessor as well as for the rest of the system. The 2650 is supported by auxiliary logic which buffers the address and data lines to drive the system buses without loading the chip. I/O functions include I/O ports for use with the Floppy Disk System and the Console. Interrupt circuits are provided to interrupt the CPU during I/O operations. An interval timer, driven by the system clock, is provided to give selectable 10 ms or 100 ms interrupts. The timer can be enabled or disabled under program control.

The Master CPU has full access to both Master and Slave Memory through use of bank switching previously discussed. It performs input output functions to all system peripherals through use of I/O ports located on the Master CPU module and optional General Purpose I/O modules. It directs all other computer modules, such as the Debug, Slave and PROM programmers through use of Extended I/O commands to ports located on the particular modules. The Master CPU is in control of the Computer Bus except during DMA operations or if it relinquishes control to a Slave CPU by executing a HALT instruction. Bus control is regained if any system or Debug interrupt is received.

2.5.2 Slave CPU and Emulation Interface

Execution of user programs, interface to user hardware and execution of the microprocessor related assembler are functions provided by the Slave CPU. The board contains the microprocessor and supporting logic to drive the system bus and the Emulation interface.

Essentially, the types of operations performed are the same as the Master CPU, except that the Slave CPU can access only Slave Memory or User Memory and I/O devices. When the Master CPU is running, the Slave is PAUSED waiting for control to be relinquished to continue executing instructions or to be jumped to an address indicated by the Debug circuitry.

Interrupt logic is contained on the board to allow interrupts as requested from the Slave interrupt bus (separate from the Master interrupt bus). 8 level priority encoding is provided by circuitry on the board. Interface to user hardware is accomplished by multiplexing the microprocessor data and control to the computer bus or to the Emulation Cable. There are three modes of Slave operation. (see figure 2-4)

1) The Slave uses the development computer memory and I/O, strictly to Debug software or user peripherals interfaced to the computer.

2) The Slave CPU exercises external user I/O while executing from development computer memory allowing checkout of prototype I/O.

3) The Slave uses external user memory and I/O. Providing Debug of the complete user prototype.

In user Mode 2 and 3, the user clock provides timing for the Slave CPU. In Mode 2, memory can be mapped in 256 byte blocks by the optional hardware analyzer module. This allows intermix of computer and user memory or memory mapped I/O.

2.5.3 Debug and Front Panel I/O Module The Debug and Front Panel I/O module performs three distinct functions within the development computer. It controls the interaction and bus time sharing of the Master and Slave CPU's. It supports all software debug features such as breakpoint, trace and Slave program start at any location. Finally, it provides interface to either the standard or Full Display Front Panel. The card is centrally located in the motherboard bus structure providing the dividing line for Master and Slave halves. Unique control lines connect to the Master CPU and Slave CPU in each half.

Debug features include storage of the Slave CPU address value, (P.C.) during instruction fetch, two Breakpoint registers with address comparison logic and Forced Jump logic including jump address storage. Debug interrupts which include breakpoint, single cycle, Slave halted as well as the eight Slave Service calls are priority vector encoded for presentation to the Master CPU.

The front panel interface routes bus address, and data to the front panel display and interfaces data and address values from the front panel during memory or I/O access. It also supports the various front panel control features such as single step.

2.5.4 Master Memory For the Universal One system, Master Memory consists of one 16K byte dynamic memory module with a 256 byte PROM overriding the first 256 RAM locations.

The Master Memory for the PROM based Universal Emulator system consists of 4K bytes of Static RAM and 4K bytes of PROM Memory. This is provided by one 4K RAM/2K PROM Master module and by the 2K PROM section of the Slave Memory module. These modules are further described in sections 2.5.6 and 2.5.7.

2.5.5 Slave Memory For the Universal One system, one to four 16K dynamic RAM modules provide 16K to 64K of Slave Memory.

For the Universal Emulator, the basic memory consists of a single 4K Static RAM module, expandable with 4K Static modules or 16K Dynamic modules to 64K bytes.

For either system, 2K PROM Modules may be intermixed with the RAM Memory if desired.

2.5.6 4K RAM/2K PROM Module This module contains two relatively independent sections: one containing 4K bytes of RAM with bus interface and the other containing 2K bytes of PROM with independent bus interface and timing.

The RAM section consists of 32 RAM chips, each capable of storing 1024 bits. These are arranged in 4 rows of 8 chips, providing an organization of 4K 8 bit words.

The PROM section contains up to eight 256 x 8 bit programmable read only memory (PROM) chips providing 2K bytes of storage. Sockets are provided for the PROM and a bank of "PROM RESIDENT" switches allows operation with any combination of PROMS plugged in.

Base address and Master-Slave identification is independently switch selectable for the PROM and RAM sections. The 4K RAM only module is a depopulated version of the 4K RAM/2K PROM module.

2.5.7 16K Dynamic RAM Module The 16K dynamic RAM module contains thirty-six 4K x 1 dynamic memory chips. It is organized in four rows of nine bits each with the ninth bit providing parity.

The memory contains input and output storage, transparent refresh support logic and byte word addressing and data storage. A pair of cards can provide full 16 bit word capability while still allowing the 8 bit byte oriented Master and byte oriented Slaves full access to both memories.

A socket is provided for a single 256 x 8 Boot PROM. This is utilized for the Master Memory and overrides the first 256 RAM locations.

2.5.8 General Purpose I/O Module Interface to peripheral devices is provided by the General Purpose I/O module board. The module can be used by either Master or Slave CPU's. This board consists of four pairs of input/output ports, selected by an Extended Read or Write operation commanded by the CPU. The module address is switch selectable with each port having a unique address. The module is designed to be connected to a variety of external parallel data devices through connectors on the top of the board. Additionally a full serial interface for either RS-232-C devices or a TTY is provided together with control and status ports for data interchange protocol. A UART (Universal Asynchronous Receiver Transmitter) is located on the board to perform the necessary serial to parallel conversion. Separate drivers are provided for the RS-232-C or TTY outputs, although only one device can be used at one time. Clock circuits, using the system clock as a reference, provide data rates of 100 Baud for the TTY and 300 to 1200 Baud for the RS-232-C interface.

2.5.9 PROM Programmer Modules Two PROM programmer modules are designed to be resident in the system: an erasable MOS PROM programmer and a bipolar fusable link programmer. The modules are connected to the system bus and to the front panel PROM sockets. Each board, although differing in detail, performs the same function. On command from the system software, an object file is loaded into Slave Memory. The data contained in the file is transferred, one word at a time from the memory to registers on the board. Sequencers and timing circuits address the chip to be programmed, and pulse the power supplies on the board to enter the data into the PROM. When the programming sequence is complete, the programmer interrupts the CPU and the next word is programmed following the same sequence. Circuitry is included to read the data stored on the PROM and output it back to the system for error checking or comparison with Data contained in Slave Memory. Current limiting, over voltage protection and low voltage sensing are provided on each module to prevent improper programming of the PROM IC's.

2.5.10 Hardware Analyzer Module This module operates in conjunction with the Debug and Slave Modules to provide these additional analysis features.

1) Storage of last 128 internal or Emulation cable bus transactions.

2) Expanded breakpoint capability

3) Memory mapping

The last 128 transactions are continuously stored and may be displayed upon encountering a breakpoint or at any point in program execution. 16 address, 16 data, bus control lines and the status of 8 auxiliary test clips are stored for each bus access. This allows real time execution up to a breakpoint event.

The expanded breakpoint consist not only of data comparisons, but include 16 data, 4 bus control functions (R/W, M/IO, FETCH, INTACK) and 8 auxiliary user test clips. Any combination of these groups may contribute to the breakpoint function. Don't cares are allowed for bus control and auxiliary user test clips. Provision is also made for selection of 8 or 16 bit data and 8 bit address during I/O transactions.

The breakpoint may be set to occur between two address limits, thus allowing trace printouts only between those limits.

Modifiers to the basic breakpoint include a pass count (so many events before the breakpoint is allowed) and a delay (allows positioning the stored trace informative window relative to the breakpoint event). Additionally real time or number of transactions may be measured between two breakpoint events.

Memory mapping consists of a 256 bit map which is loaded by the Master CPU under user command. The map divides the 64K Slave Memory space into 256 byte blocks. The state of the bit for each block determines whether a Slave Memory transaction is routed to user memory or computer memory. This control logic is utilized by the Slave CPU in Emulation mode 1 only.

2.5.11 Emulation Cable The Emulation Cable consists of two sections of cable and an in-line buffer card assembly.

The cable is attached to the Slave Module on one end and the users 40 pin microprocessor socket on the other. The buffer card contains line driver and receivers to interface to the Slave CPU over the main 8 foot cable length. A short twisted pair cable attaches to the 40 pin microprocessor plug. The interface is designed to minimize user circuitry loading and insure signal integrity.

Chapter 3
MASTER CPU MODULE

3.0 INTRODUCTION

Data transfer and processing tasks as well as overall system control are handled by the Master CPU and I/O module. In system operation, the board controls all major modules and can perform I/O transfers with the disk system and console device.

The central processor function is performed by a Signetics 2650 microprocessor and its supporting logic, consisting of crystal-controlled clock and bus drivers for the bi-directional data bus, the address bus, and the control bus.

The module contains the I/O ports to the Floppy Disk Storage Subsystem and the CRT or TTY console. It also provides all system clocks for use by modules interfacing to the bus.

This chapter outlines the function of the Master CPU module in the system, and describes the logic contained on the board to implement this function. The 2650 microprocessor itself is not described. Those readers wishing detailed information on this device are referred to the Signetics 2650 Reference Manual.

3.1 GENERAL DESCRIPTION

A simplified data flow block diagram of the board is shown in Figure 3-1. The board can be divided into four major sections for purposes of discussion. The first major block is the CPU itself and the supporting logic. The second section includes I/O logic for communication with the Disk System and the Console device. The third section consists of interrupt priority decoding and interrupt vector generation logic. The last major section consists of the baud rate and interval timer logic.

3.1.1 The CPU The 2650 microprocessor is the controlling element, accessing memory for instructions and executing these instructions. Logic surrounding the CPU device is intended to provide bus driving capability and timing for the CPU's operations. Secondary logic functions include Slave Memory bank switching and power-on reset.

The clock consists of 10 MHz crystal-controlled oscillator which runs continuously. The output of the oscillator drives a divide-by-eight circuit to give the required 1.25 MHz clock signal for the CPU. Part of the divider consists of a flip-flop which in turn is controlled by an external "hold" signal which effectively puts the CPU in a hold state during Front Panel single step or breakpoint.

The CPU's data lines are buffered by transceivers whose inputs and outputs are connected to the system data bus. Each of the address lines is buffered similarly, using tri-state drivers whose outputs may be floated to allow DMA operation in future applications.

The drivers and receivers provide the necessary system bus driving capability. When the CPU is in a "run" state, the drivers are enabled, and signals are passed to the bus or received from the bus. When the CPU is halted, the drivers are disabled and put into a high-impedance state so the buses can be controlled by external devices, or by the Slave CPU.

Logic is provided to initialize the system and restart the CPU upon power turn-on. When the +5 V supply comes up to its operating level, the power-on detector initiates a "reset" signal which resets the 2650 CPU and also sends a system-wide reset to the rest of the modules. This causes the CPU to fetch a ZBSR instruction from location 00 in Master memory and perform start up routines.

3.1.2 I/O Logic Communications between the system and peripherals such as the Floppy Disk or High Speed Paper Tape Reader and the Console

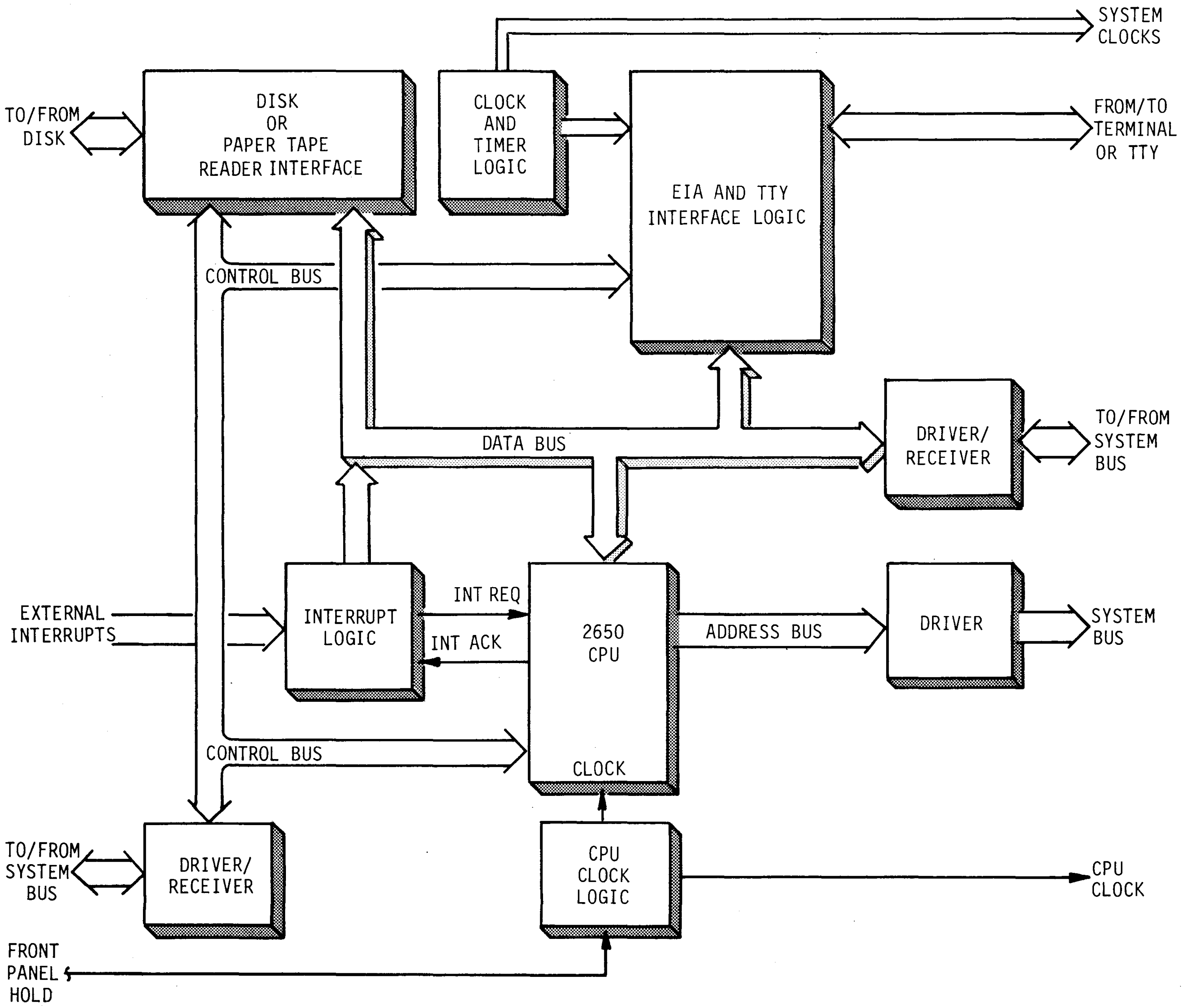


FIGURE 3-1 MASTER CPU BLOCK DIAGRAM

are handled by the CPU board I/O logic. There are two separate interfaces: a serial interface for the console device, any EIA RS-232 terminal or current loop TTY, and a parallel interface for the disk system. The serial interface incorporates a UART (Universal Asynchronous Receiver Transmitter) to perform the necessary serial-to-parallel and parallel-to-serial conversion for the TTY or terminal. Status and control registers are included to provide the data exchange protocol signals required by the console device and the CPU. Interface to the disk includes lines which allow control of the optional line printer which is interfaced to the disk system.

Clock generation logic provides a crystal-controlled 9.984 MHz system clock, 110 baud to 2400 baud TTY and CRT terminal clocks, a 38.4 kHZ I/O clock, and a 10 ms or 100 ms interval timer.

3.1.3 Interrupt Logic Since the system operates with an interrupt-driven structure, a means must be provided to assign priority to interrupts from devices capable of such interrupts, and to generate interrupt vectors when the CPU has accepted an interrupt. The interrupt logic accepts up to 16 bus interrupts, 6 of which are generated on the card. It latches all inputs when the processor is acknowledging an interrupt to prevent a higher priority line changing the vector while the CPU is receiving it. Priority encoders arbitrate priority among the devices requesting interrupts and enable the interrupt vector generation logic. An additional 16 Master interrupts vectors are generated on the debug logic module. Priority and enabling of these vectors is controlled by logic on the Master CPU card.

3.1.4 DMA Logic Capability for future implementation of DMA boards within the system is provided by logic on the CPU board which allows the 2650 to be PAUSED and the address and data buses to be tri-stated. In this way, the DMA boards can control the address and data buses, allowing an external device to operate directly with system memory, without utilizing the CPU and its I/O functions.

3.2 DETAILED DESCRIPTION

This section describes individual logic blocks within the module.

3.2.1 2650 CPU Support The 2650 microprocessor and its supporting logic, shown in figure 3-2, provide control of the system. Acting on instructions accessed from memory, the 2650 manipulates data and provides control using its arithmetic and logic functions. For reference, a list of 2650 signal definitions is given in Figure 3-3.

The CPU clock is generated by a 10 MHz oscillator whose output is divided by eight in two stages. The first stage divides the clock frequency by four. The second stage divides the clock by two and provides a means of stopping the clock in the low signal state for any number of half cycles. The front panel "hold" signal is generated by the Full Display front panel and stops the clock during single step or breakpoint operation. The basic 10 MHz oscillator uses a series resonant crystal as the feedback element and two inverters which are biased into the linear region as amplifiers. The 10 MHz output is also buffered onto the system bus for Slave CPU use.

The CPU address, data, and control outputs are buffered by a series of drivers and receiver/transmitters. Bidirectional bus drivers receive data from the system bus and transmit it to the 2650, or receive data from the 2650 and transmit it to the bus. These devices are enabled by ANDing the RUN and R/W outputs from the 2650 through inverters and gates. When the 2650 is not running, the transceivers are disabled and put into the high inpedance state. In this state the data bus is floating and can be driven by other modules in the system such as the Slave CPU or DMA module.

Unidirectional bus drivers buffer the address outputs from the 2650 and drive the system address bus. Each device is enabled by an inverted RUN output from the 2650. For the Master CPU to access Slave memory, any 16K

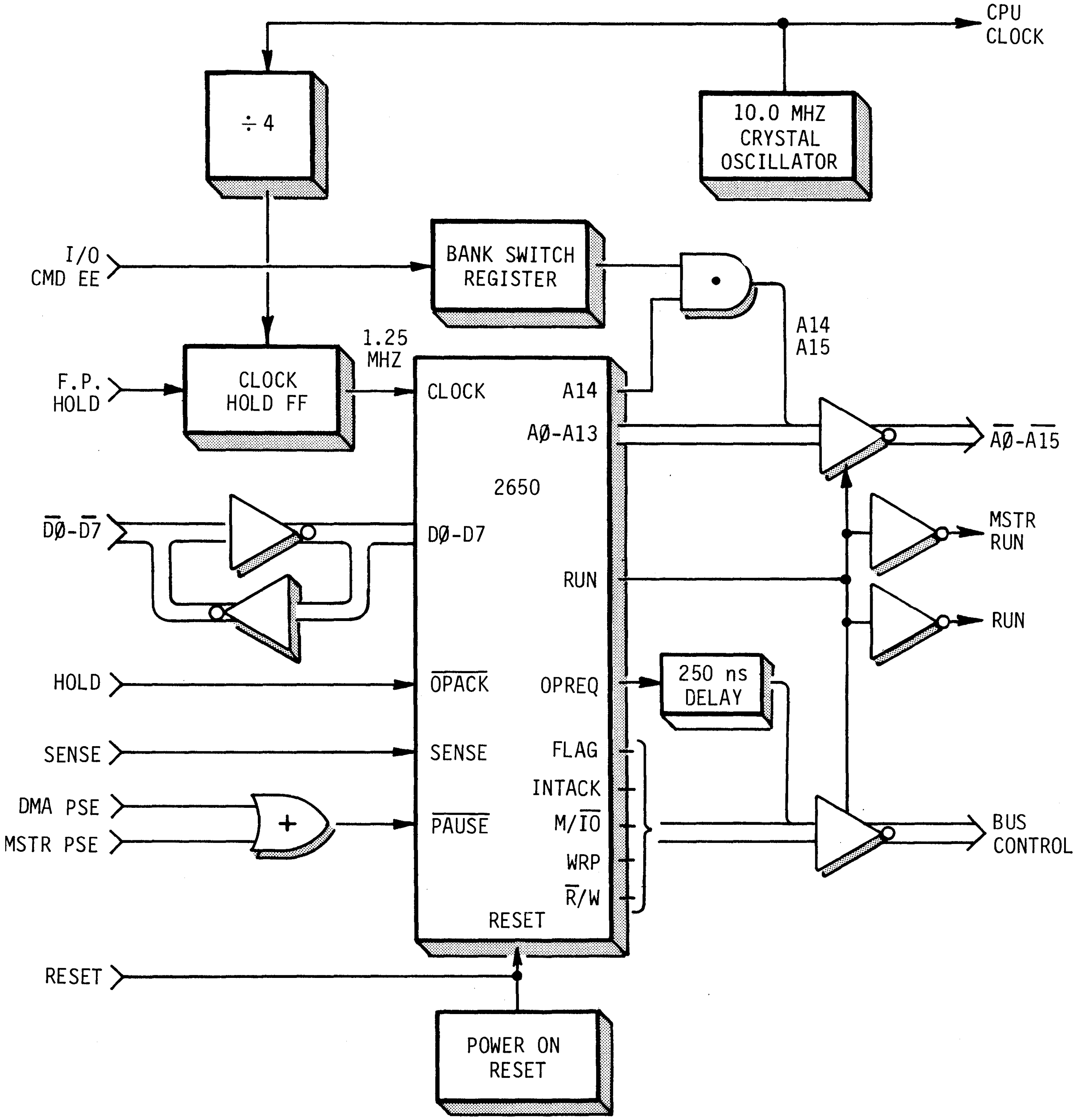


FIGURE 3-2 2650 PROCESSOR AND SUPPORT

ADR0 - ADR12

ADDRESS LINES. ADR0 carries the low order address bit, and ADR12 carries the high order address bit. ADR0 - ADR7 are also used for I/O device selection for extended I/O operations. These lines are tristated with $\overline{\mathrm{ADREN}}$.

ADR13 - E/$\overline{\mathrm{NE}}$

For memory operations, this line delivers the ADR13 address bit. For I/O operations, this line discriminates between EXTENDED and NON-EXTENDED I/O instructions.

ADR14 - D/$\overline{\mathrm{C}}$

For memory operations, this line delivers the ADR14 address bit. For I/O operations, this line discriminates between DATA and CONTROL non-extended I/O instructions.

$\overline{\mathrm{ADREN}}$

The ADDRESS ENABLE signal allows tristate control of address bus (always grounded).

DBUS0 - DBUS7

The DATA BUS pins form an 8-bit bidirectional data path "in" and "out" processor. All instructions and data are transferred using this bus.

$\overline{\mathrm{DBUSEN}}$

The DATA BUS ENABLE signal allows tristate control of the data bus (always grounded)

OPREQ

OPERATION REQUEST is the coordinating signal for all external operations.

$\overline{\mathrm{OPACK}}$

OPERATION ACKNOWLEDGE is an input to the processor indicating completion of an external operation. This allows asynchronous functioning of external devices. If not present, the processor stays in T1 low.

M/$\overline{\mathrm{IO}}$

MEMORY/$\overline{\mathrm{INPUT\text{-}OUTPUT}}$ is an output signal indicating a memory or I/O operation.

$\overline{\mathrm{R}}$/W

The $\overline{\mathrm{READ}}$/WRITE output signal indicates whether an operation is a read or write operation.

WRP

The WRITE PULSE output is a timing signal from the processor that provides a positive going pulse in the middle of each requested write operation.

SENSE

The SENSE line provides an input line to the 2650 that is independent of the normal I/O bus structures. The sense signal is connected directly to one of the bits in the program status word.

FLAG

The FLAG output indicates the state of the flag bit in the program status word.

$\overline{\mathrm{INTREQ}}$

The INTERRUPT REQUEST input is used by external devices to suspend normal program execution and enter into an interrupt sequence.

INTACK

The INTERRUPT ACKNOWLEDGE output signal is used by the processor to indicate to an external device that it is ready to receive the vector address.

$\overline{\mathrm{PAUSE}}$

The PAUSE input line is used to stop the processor and force it inot the "wait" state. It is used by the system for Front Panel Access and during Slave activity.

RUN/$\overline{\mathrm{WAIT}}$

The RUN/$\overline{\mathrm{WAIT}}$ output indicates the run/wait status of the processor.

RESET

The RESET input resets the program counter to zero and clears the interrupt inhibit bit.

CLOCK

The CLOCK input is a 1.25 MHz square wave that determines the instruction execution rate.

FIGURE 3-3 2650 SIGNAL DEFINITIONS

section of Slave memory is mapped into the Master 16K-32K address space. The 16K section is specified by two control bits in a Master I/O command. The bit formats are described in the paragraphs concerning I/O operations. When the Master address is 16K or above address bit 14 is on. This outputs the bank switching bits to the address bus as bits 14 and 15.

Control signals such as INTACK, FLAG, WRP, and M/IO are brought directly to a tri-state driver. Other signals such as RUN and OPREQ are conditioned by logic which is driven by the 2650 outputs. OPREQ is ANDed with INTACK and delayed 250 ns by an RC network. The resultant signal is squared by a Schmitt trigger and inverted before being transmitted to a driver which is enabled or disabled by the RUN output of the 2650. The 250 ns delay guarantees that all bus data, address and control lines will be valid when OPREQ is issued. HOLD and SENSE control inputs are inverted and drive the OPACK and SENSE inputs of the 2650.

The power-on reset circuit consists of an RC network which charges during power on and a Schmitt trigger which provides a 50 ms pulse to reset system logic. The RESET signal is ORed with the Front Panel RESET signal.

Basic CPU timing is shown in Figure 3-4 (a, b & c). Timing for Read, Write and Interrupt operations is shown for reference. The system bus uses a HOLD signal rather than an operation acknowledge (OPACK). If the memory or I/O device cannot return or take data within 200 ns of OPREQ, a hold is generated by the device until the data to be read is valid, or the write data has been received.

3.2.1 I/O Logic The I/O logic on the Master CPU board provides the following functions:

1) Console
2) Disk I/O
3) I/O decoding

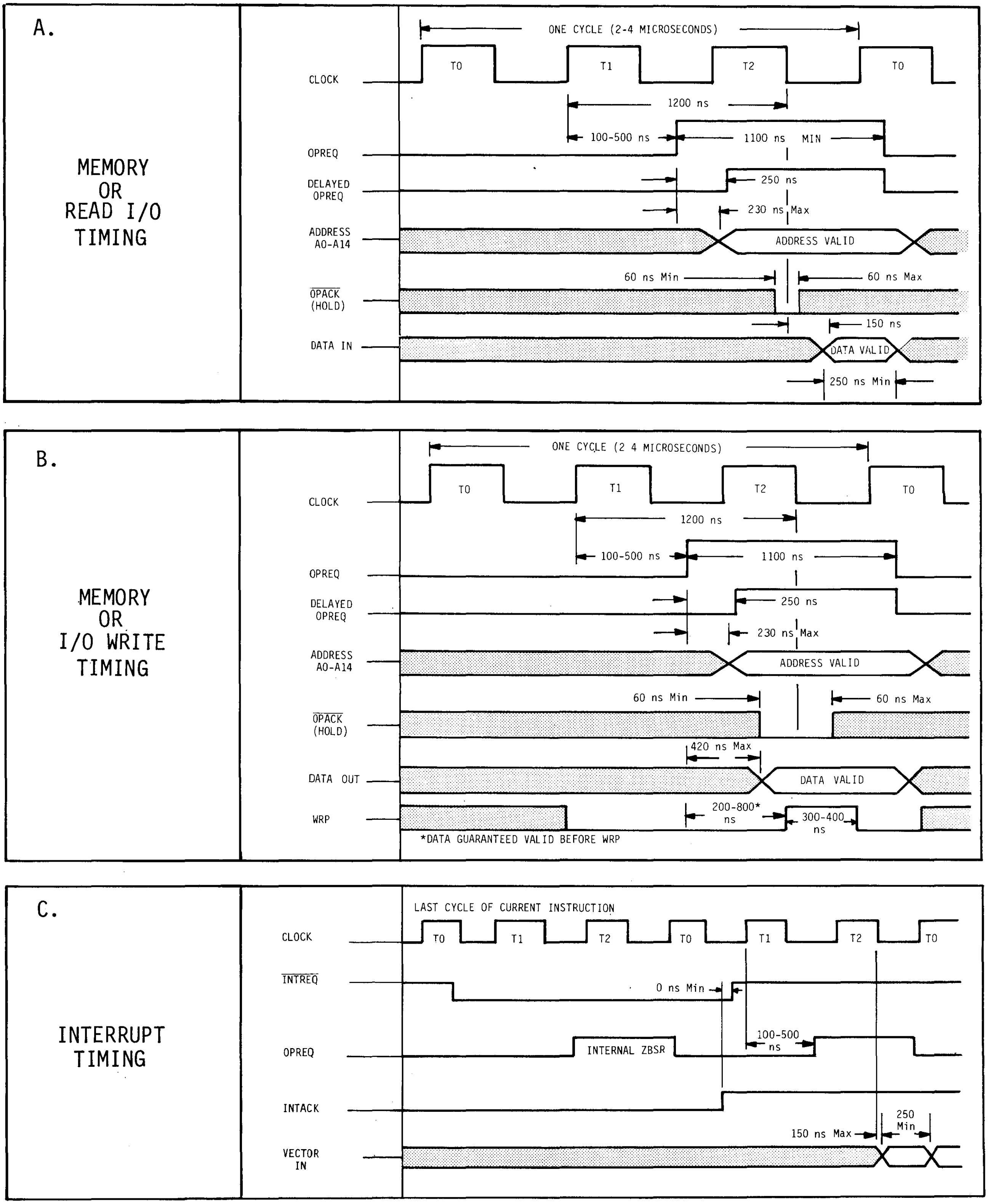


FIGURE 3-4 MASTER CPU TIMING

This section has a separate bi-directional I/O data bus and driver-receiver pairs to the bus. It contains all I/O decoding, I/O output storage registers, and input gating.

The I/O logic can be broken into sections for purposes of discussion. TTY and RS-232-C interfaces share the same logic, while the disk interface requires separate logic.

CONSOLE I/O This section, shown in Figure 3-5, provides both a limited EIA RS-232 serial interface and a TTY current loop interface. However, only one terminal can be connected to the board at any time. A UART is the heart of the circuit, providing parallel to serial and serial to parallel conversion, status, and programmable control functions. The UART clock rate and number of stop bits are switch-selected.

Serial data input received from the EIA receiver or the current loop interface is fed to the UART. The UART shifts the serial data in and sets the data available (DA) status when a complete character has been received. If interrupts are enabled, this triggers the TTY IN interrupt (level 4). The parallel data is enabled out of the UART through tri-state buffers onto the internal data bus is by reading port E8. This also resets DA.

Serial data out is transmitted by performing a write to port E8. The parallel data is presented to the UART from the internal bus and strobed into the UART by applying a pulse to its DS input. This initiates the serial output stream to an open collector driver for the TTY current loop interface and to a bipolar driver for the EIA interface. When the character is transmitted, TBMT (transmit buffer empty) is set by the UART. If interrupts are enabled this sets the TTY OUT interrupt (level 5).

Status indications consisting of DA, TBMT, OR (overrun), FE (framing error) or PE (parity error) are available by reading port E9. Tri-state buffers enable these signals onto the internal bus from the UART. Formats for the

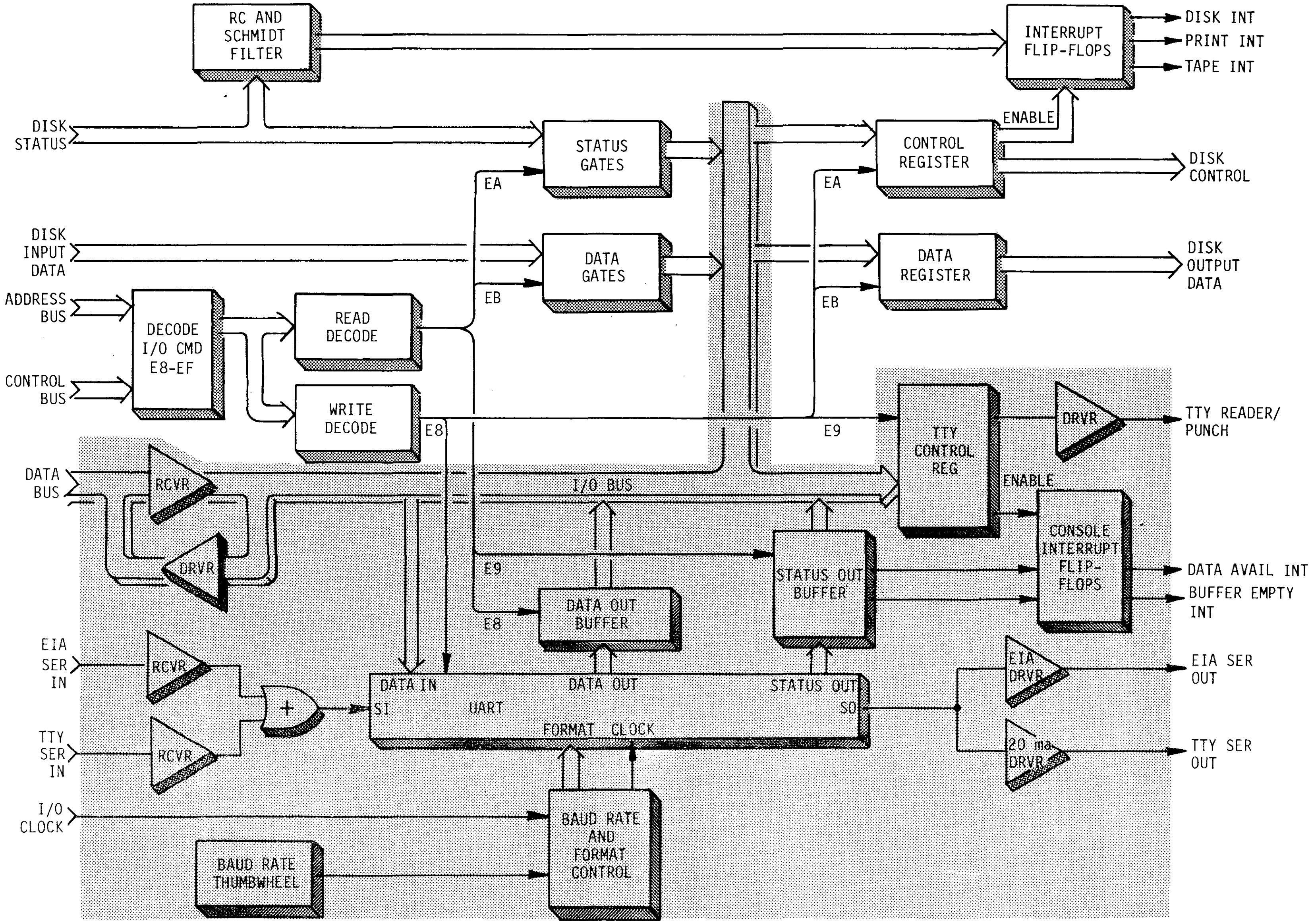


FIGURE 3-5 MASTER CPU CONSOLE I/O

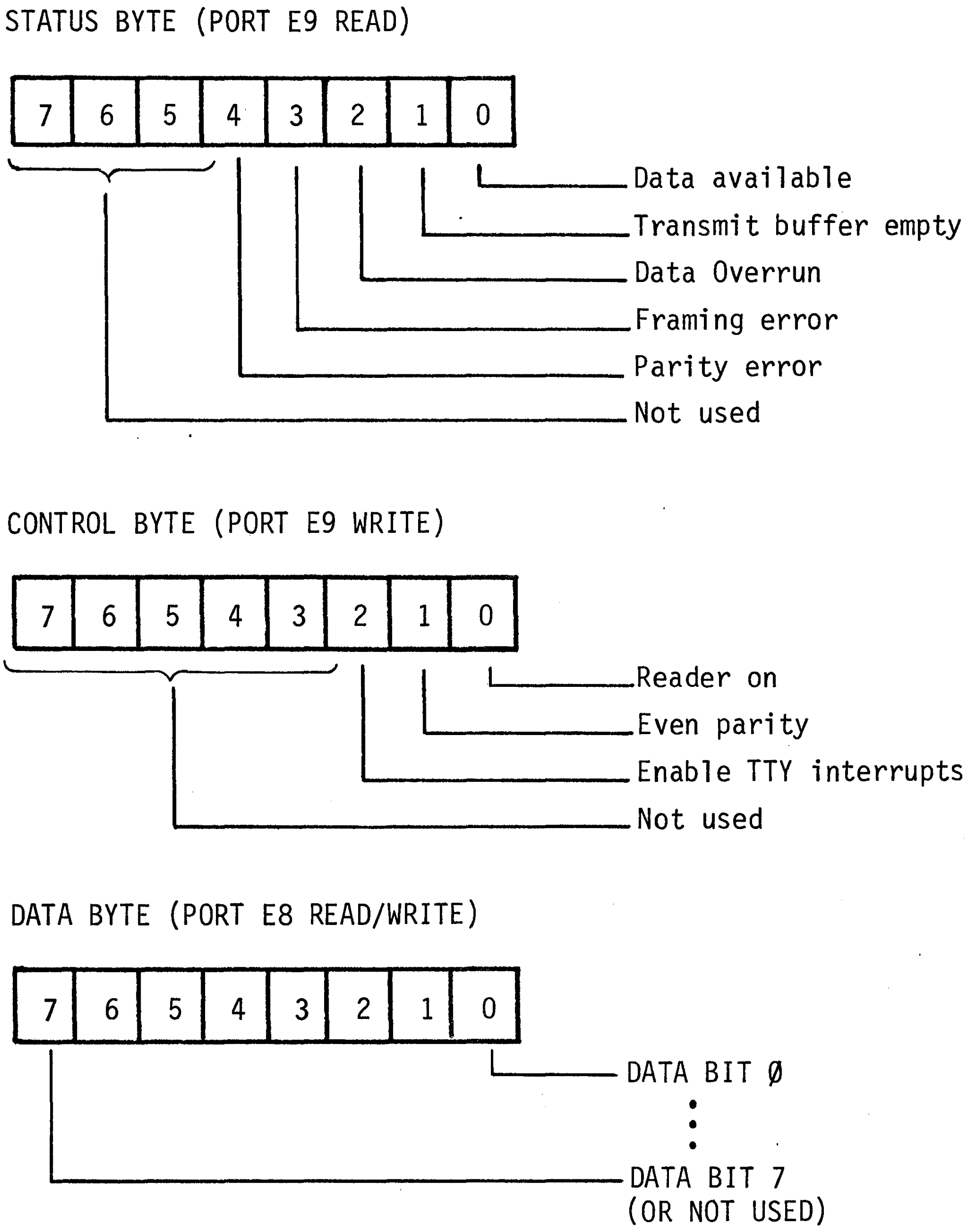


FIGURE 3-6 CRT/TTY FORMATS

data and control ports are shown in Figure 3-6.

A TTY-related function is control of the TTY paper tape reader. A separate reader-on flip-flop and open collector driver provide the interface for this function. The flip-flop is turned on under program control and is reset upon receipt of the start bit, thus reading a character at a time.

The status byte which controls the TTY paper tape reader, the UART parity circuits, and enables the TTY interrupts is output by a write to port E9.

The CONSOLE connector is a 25 pin connector conforming to EIA RS-232-C standards which is mounted on the development computer rear panel. This connector is connected to connector P2 on the top edge of the Master CPU board by a ribbon cable.

To interface to the CRT or other EIA peripherals, certain signals of the RS-232-C interface are pulled up permanently to a logic 1. These signals are clear to send, data set ready, carrier detected, data terminal ready, and request to send.

DISK I/O The disk I/O logic, shown in Figure 3-7, consists of a data I/O port, a control I/O port, and interrupt flip-flops. The byte formats are shown in Figure 3-8. The eight disk input data bits (DI DØ-7) are terminated with a pull up-pull down network and received with tri-state buffers. The buffers are gated onto the internal bus by reading port EB. Output data is stored in an eight-bit register by writing to port EB. The outputs are driven with tri-state buffers. Disk status input is enabled to the data bus with tri-state drivers by reading port EA. Disk controller signals PBZY and FLG are filtered with an RC network and shaped by a Schmitt trigger, then fed to interrupt flip-flops. A paper tape signal is provided for replacement of the disk drives by a high speed paper tape reader. Disk control is performed by a write to port EA. This outputs a control byte which consists of the three disk control bits, a spare output bit and interrupt enables for the disk and printer. The three output disk control bits are buffered with tri-state drivers.

For a detailed description of the floppy disk command formats and timing refer to Appendix A.

I/O DECODING The I/O decoding logic, shown in the shaded portion of figure 3-9, recognizes extended I/O commands to addresses E8-EF. The logic provides enable lines for the tristate input gates and clocks for the output registers. The logic also provides strobes for the bank switch and timer interrupt control discussed in other sections.

3.2.3 Interrupt Logic The interrupt logic, shown in Figure 3-10, resolves priority and generates the interrupt routine address vector for 16 system interrupts. It also handles the interaction of the 16 level Debug interrupt vector generator.

Interrupts are levels created by a dedicated interrupt flip-flop. The 16 interrupt lines are continuously sampled at a 10 MHz rate. If one or more interrupts are present, the CPU interrupt line is activated. When the CPU responds with INTACK, sampling is stopped and the interrupt address for the highest priority line present is encoded. The vector is placed on the data bus during INTACK for input to the CPU. I/O circuits can also recognize their vector on the data bus during INTACK to determine when their interrupt request has been acknowledged. On-board interrupts are reset with a vector decoder which is enabled during INTACK. The Debug module interrupts have lower priority and are given an enable line (DBG VEN) only if no system I/O interrupts are pending. Figure 3-11 lists priorities and vector addresses for the interrupts.

3.2.4 DMA Logic DMA operations require that the CPU be paused, and the address and data buses floated for the necessary DMA device access. This is accomplished by the DMA PAUSE signal. DMA PAUSE is ORed with

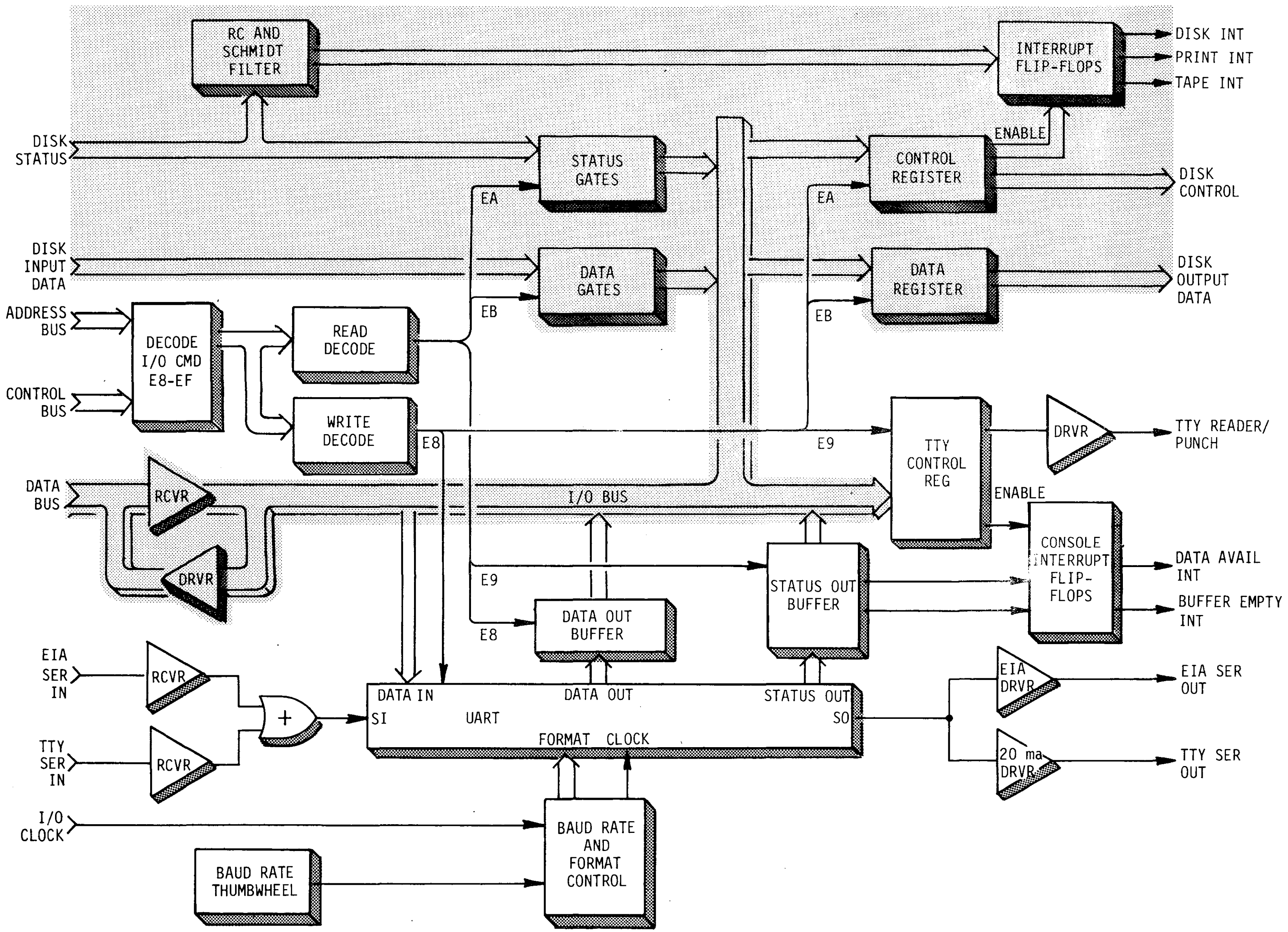
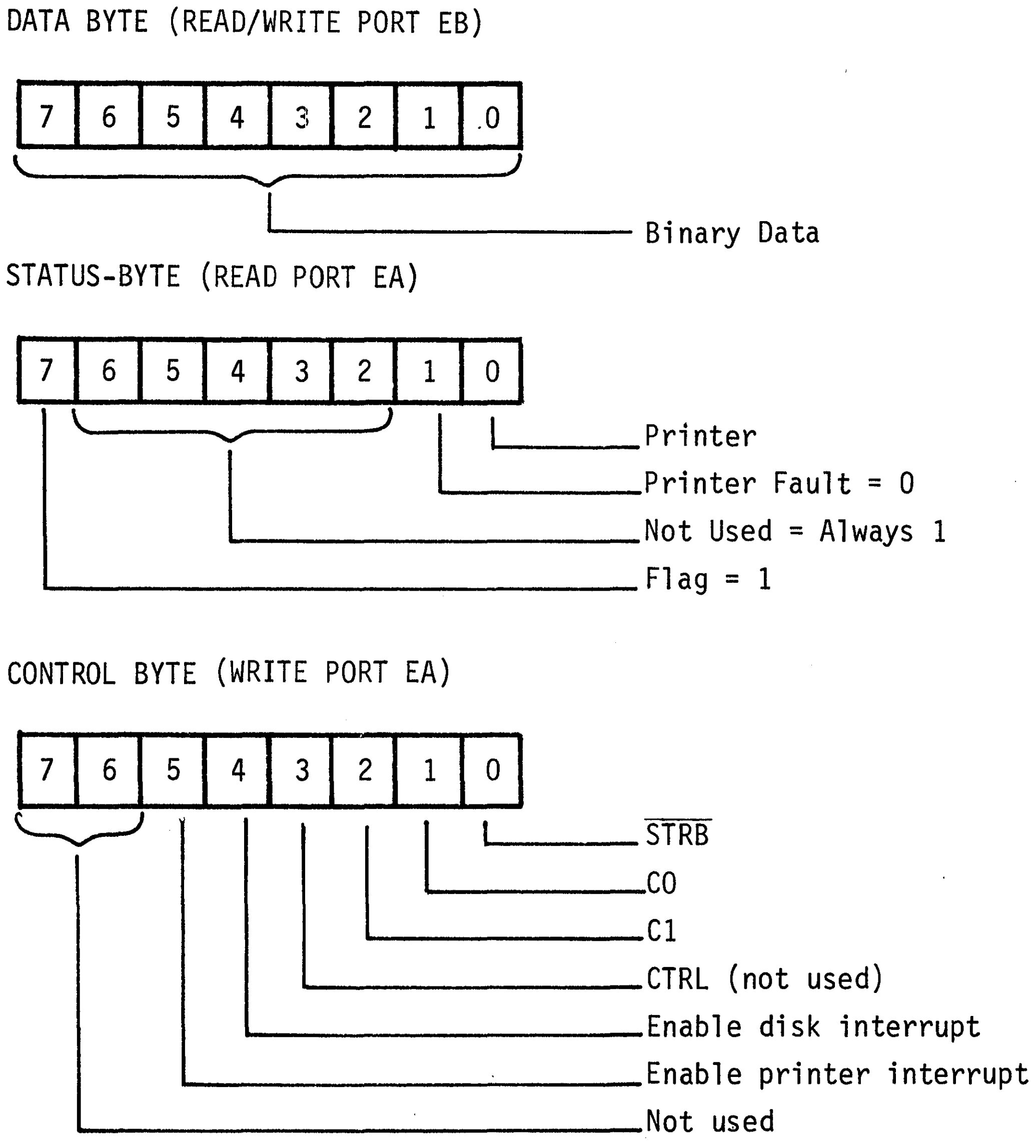


FIGURE 3-7 MASTER CPU DISK I/O BLOCK DIAGRAM



DATA BYTE (READ/WRITE PORT EB)

STATUS-BYTE (READ PORT EA)

CONTROL BYTE (WRITE PORT EA)

FIGURE 3-8   DISK FORMATS

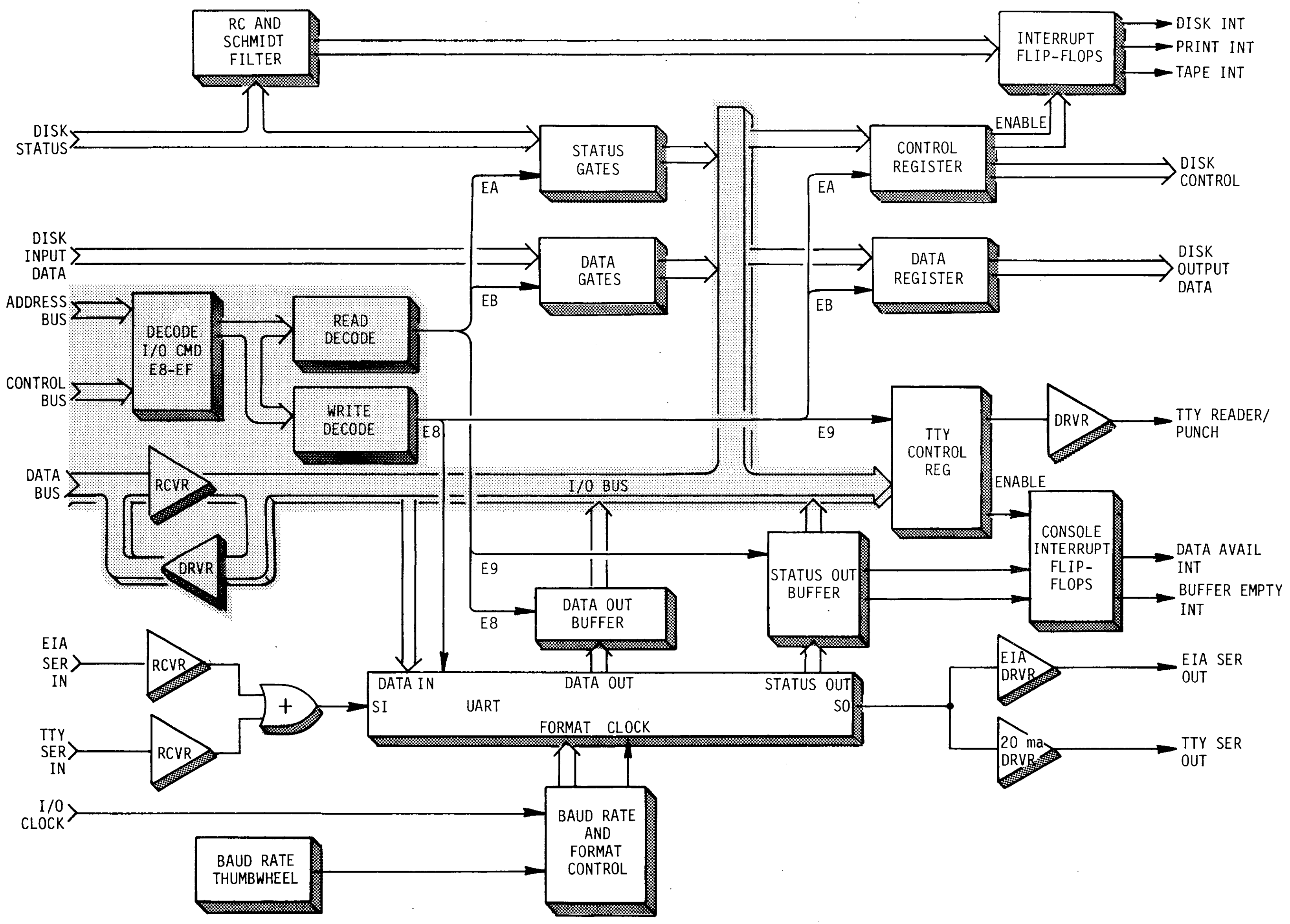


FIGURE 3-9 MASTER CPU I/O DECODING BLOCK DIAGRAM

MSTR PAUSE and drives the PAUSE input of the 2650. After completion of the current instruction, the 2650 RUN output will go high. This is used to disable the address and data bus drivers, which float the buses.

3.2.5 Baud Rate and Timer Logic The block diagram of this section is shown in Figure 3-12.

A crystal-controlled oscillator, identical (except for frequency) to that used in the CPU clock logic, is used for the baud rate and interval timer generators. The 9.984 MHz output of the oscillator is directed through a driver to the rest of the system through the system bus. The 38.4 KHz system I/O clock is provided by a divide by 260 circuit off the 9.984 MHz crystal. This is further divided by two and splits into three paths. One chain of dividers provides the 150-2400 (x16) baud rate clocks. The second path is a divide by eleven providing the I/O baud TTY rate. The third chain is a divide by 24 or 240 providing 10 or 100 ms interval timer clock.

3.3 UTILIZATION

3.3.1 Installation The module is designed to be plugged into the card file of the development computer, but may be used in other applications. Jumper and switch options are discussed in section 3.3.2. All interface connections and connector pinouts are detailed in section 3.3.3.

COMPUTER INSTALLATION The Master CPU module plugs into any Master section slot of the motherboard (J3-J8). For cabling convenience, it is normally plugged into J8. The card connector is offset, insuring that it cannot be plugged in backwards. All power requirements (+5V, ±12V) are supplied through the motherboard. Cable 90014011, a standard flat ribbon 25 pin I/O cable attaches to the top edge connector P2 for console input/output. Cable 90014021, the disk I/O cable attaches to the top edge connector P3. Care should be taken that pin 1 of the cable (Red Stripe) and pin 1 of the edge connector align. Pin 1 is to the left when

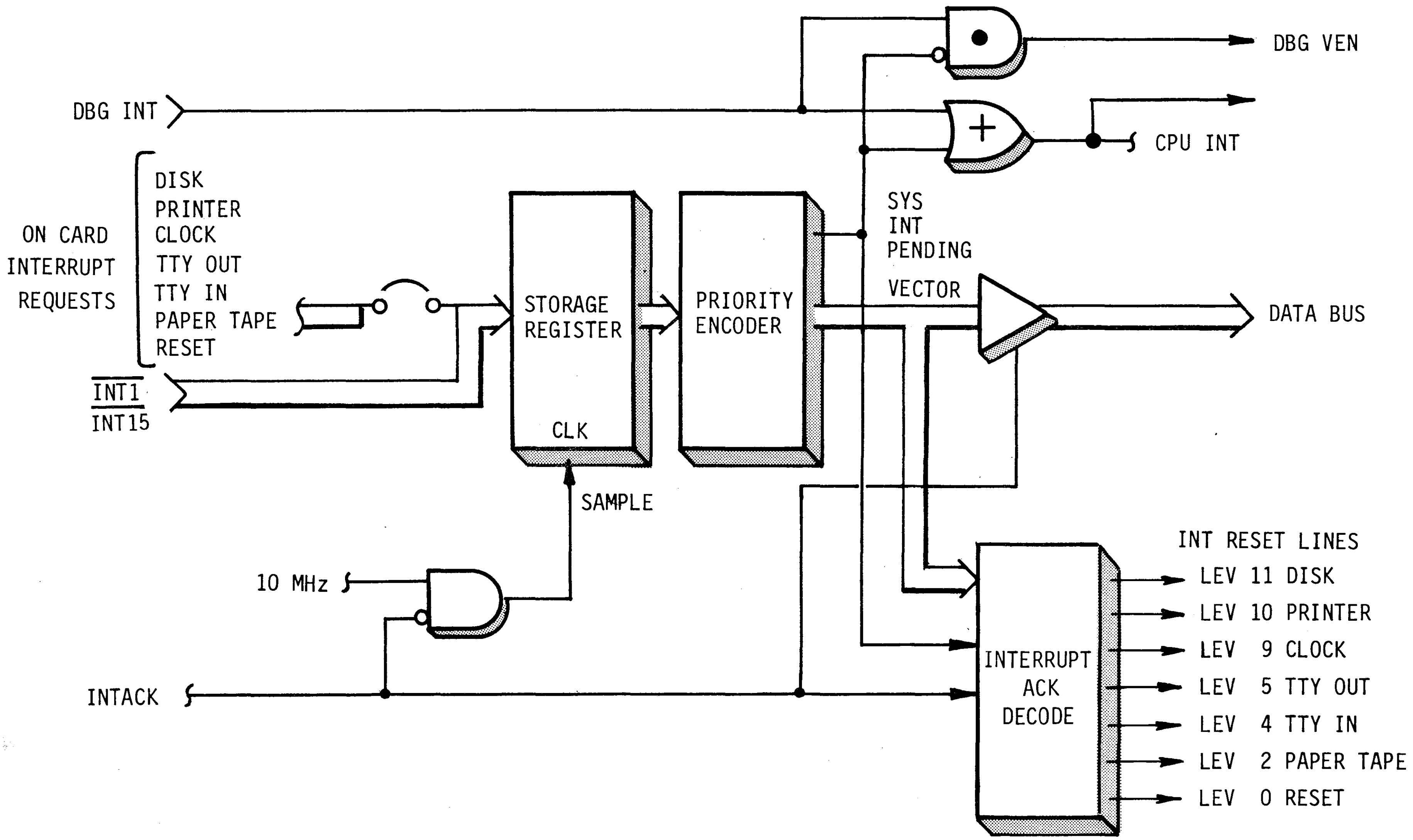


FIGURE 3-10 MASTER CPU INTERRUPT LOGIC

Priority	Vector Address	Function	
0	0000	Reset	MASTER I/O
1	0002	Master Memory Parity Error	
2	0004	Paper Tape Reader	
3	0006	Slave Memory Parity Error	
4	0008	TTY In	
5	000A	TTY Out	
6	000C	RS-232 In	
7	000E	RS-232 Out	
8	0010	NA	
9	0012	Timer	
10	0014	Printer	
11	0016	Floppy Disk	
12	0018	NA	
13	001A	PROM Program 1	
14	001C	PROM Program 2	
15	001E	NA	SERVICE REQUESTS
16	0020	Slave SVC 1	
17	0022	Slave SVC 2	
18	0024	Slave SVC 3	
19	0026	Slave SVC 4	
20	0028	Slave SVC 5	
21	002A	Slave SVC 6	
22	002C	Debug SVC 1	
23	002E	Debug SVC 2	
24	0030	Breakpoint 1	DEBUG HARDWARE
25	0032	Breakpoint 2	
26	0034	Single Cycle	
27	0036	Slave Halted	
28	0038	Diag. Int	
29	003A	Debug Int 29	
30	003C	" " 30	
31	003E	" " 31	

NA = NOT ASSIGNED

FIGURE 3-11 INTERRUPT PRIORITY ASSIGNMENTS AND ADDRESS VECTORS

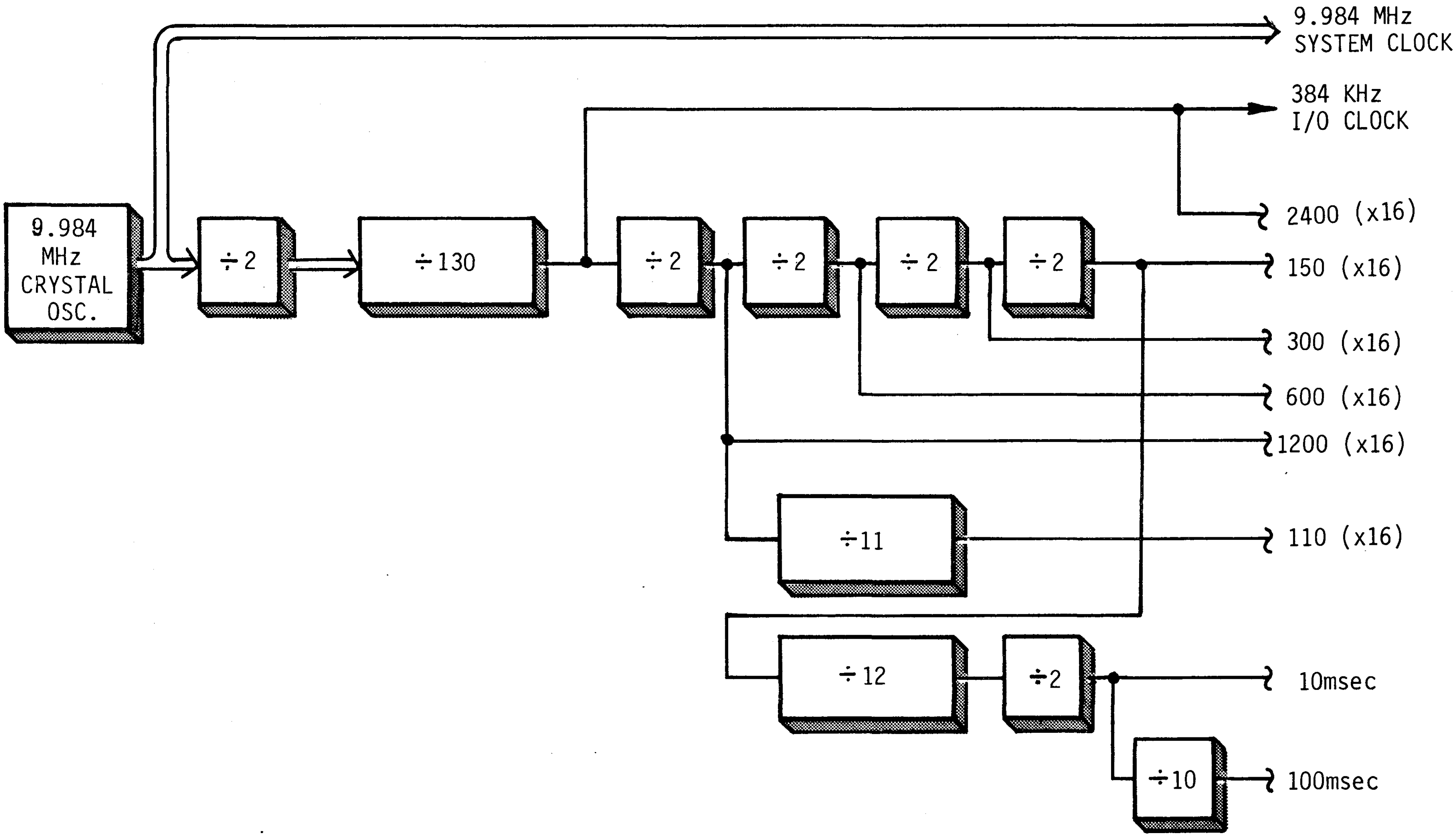


FIGURE 3-12 BAUD RATE AND TIMER LOGIC

viewed from the component side of the board.

INSTALLATION IN OTHER APPLICATIONS When using the module in other applications, the user must take into consideration environmental extremes, mounting, power requirements, interface connections and signal AC and DC requirements. Environmental limits, power requirements and card outline dimensions are listed in Figure 3-13.

For mounting, the board is designed to plug into a 100 pin edge card connector, CDC VPB-01C5-0D00A1 or equivalent. Card guide slots should be provided and the card should be mounted vertically or in some way clamped so that it does not vibrate out of its socket. Vertical mounting is also important to insure corrective cooling if no fan is used. The card dissipates approximately 10 watts, therefore adaquate space should be allowed between modules for air flow. It is desirable to have forced air flow between all modules if several modules are utilized within a confined area.

Interface connections and signal requirements are given in section 3.3.3.

3.3.2 Switch and Jumper Options

A diagram of the baud rate select thumbwheel switch is shown in Figure 3-14. Data format is selectable for binary or ASCII format at the 110 baud rate. A jumper provides optional binary format for all baud rates.

Other on card jumper options are for 10 or 100 ms timer selection and to disconnect interval interrupt lines. These affect system operations and should not normally be changed.

3.3.3 Interface and Connector Pinouts

A complete interconnection diagram for Master CPU use within the system is given in Figure 3-15. Figure 3-16 lists the motherboard connector (P1) pinouts and signal definitions. Figures 3-17 and 3-18 list P2 and P3 pinouts for the Console and Disk System I/O. The disk port is actually a general purpose parallel port and may be utilized for other applications.

ENVIRONMENTAL

Air Temp 0° - 70°C Operational
Surrounding Module

Storage Temp -55° - 125°C

Humidity 90% Non Condensing

POWER

	TYP	MAX
+5VDC	1.9A	2.4A
+12VDC	.05A	.08A
-12VDC	.05A	.08A

PHYSICAL OUTLINE

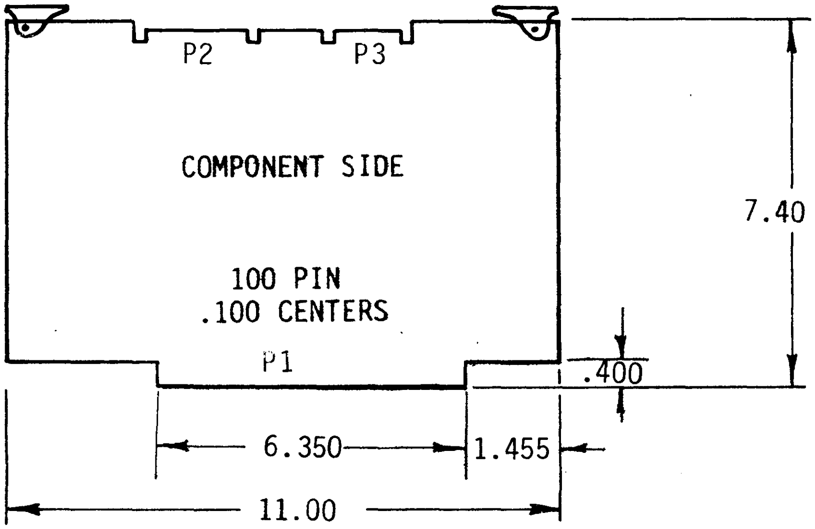
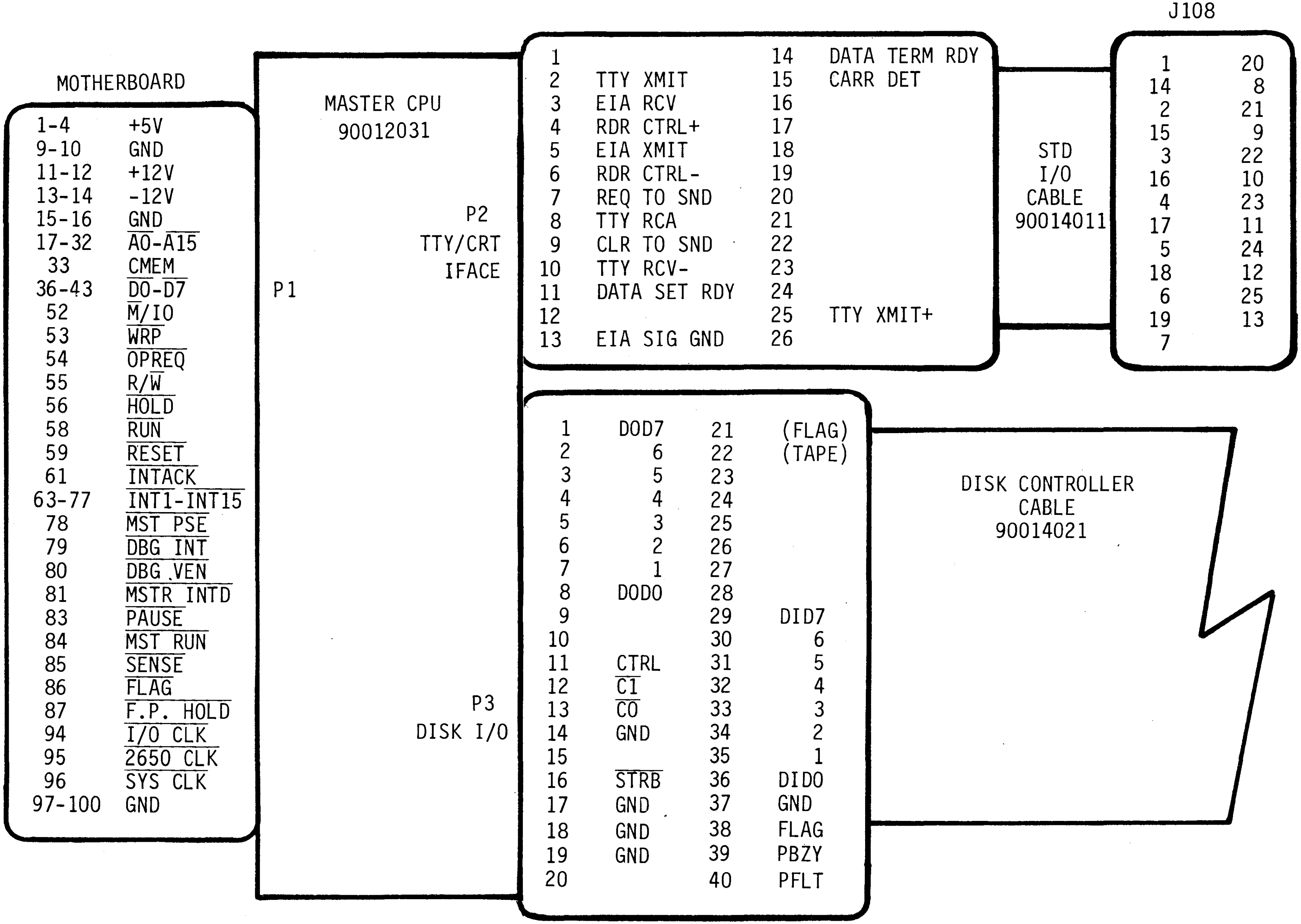


FIGURE 3-13 MASTER CPU ENVIRONMENTAL, POWER AND PHYSICAL REQUIREMENTS

Thumbwheel Position	Baud Rate	START	Data Format	STOP
9	110	S	0 1 2 3 4 5 6 P	S S
8	110	S	0 1 2 3 4 5 6 7	S S
7	150	S	0 1 2 3 4 5 6 P	S
6	300		same as 150	
5	600		same as 150	
4	1200		same as 150	
3	2400		same as 150	

FIGURE 3-14 BAUD RATE THUMBWHEEL SWITCH SELECTION



MOTHERBOARD

1-4	+5V
9-10	GND
11-12	+12V
13-14	-12V
15-16	GND
17-32	$\overline{\text{A0-A15}}$
33	$\overline{\text{CMEM}}$
36-43	$\overline{\text{D0-D7}}$
52	$\overline{\text{M/IO}}$
53	$\overline{\text{WRP}}$
54	$\overline{\text{OPREQ}}$
55	$\overline{\text{R/W}}$
56	$\overline{\text{HOLD}}$
58	$\overline{\text{RUN}}$
59	$\overline{\text{RESET}}$
61	$\overline{\text{INTACK}}$
63-77	$\overline{\text{INT1-INT15}}$
78	$\overline{\text{MST PSE}}$
79	$\overline{\text{DBG INT}}$
80	$\overline{\text{DBG VEN}}$
81	$\overline{\text{MSTR INTD}}$
83	$\overline{\text{PAUSE}}$
84	$\overline{\text{MST RUN}}$
85	$\overline{\text{SENSE}}$
86	$\overline{\text{FLAG}}$
87	$\overline{\text{F.P. HOLD}}$
94	$\overline{\text{I/O CLK}}$
95	$\overline{\text{2650 CLK}}$
96	$\overline{\text{SYS CLK}}$
97-100	GND

1		14	DATA TERM RDY
2	TTY XMIT	15	CARR DET
3	EIA RCV	16	
4	RDR CTRL+	17	
5	EIA XMIT	18	
6	RDR CTRL-	19	
7	REQ TO SND	20	
8	TTY RCA	21	
9	CLR TO SND	22	
10	TTY RCV-	23	
11	DATA SET RDY	24	
12		25	TTY XMIT+
13	EIA SIG GND	26	

J108

1	20
14	8
2	21
15	9
3	22
16	10
4	23
17	11
5	24
18	12
6	25
19	13
7	

1	DOD7	21	(FLAG)
2	6	22	(TAPE)
3	5	23	
4	4	24	
5	3	25	
6	2	26	
7	1	27	
8	DOD0	28	
9		29	DID7
10		30	6
11	CTRL	31	5
12	$\overline{\text{C1}}$	32	4
13	$\overline{\text{C0}}$	33	3
14	GND	34	2
15		35	1
16	$\overline{\text{STRB}}$	36	DID0
17	GND	37	GND
18	GND	38	FLAG
19	GND	39	PBZY
20		40	PFLT

FIGURE 3-15 MASTER CPU INTERCONNECT

PIN	SIGNAL	FUNCTION		PIN	SIGNAL	FUNCTION
1	+5V	Logic Power Input		51	D15	Data Bus
2	+5V			52	M/IO	Memory/IO OP Out
3	+5V			53	WRP	Write Pulse Out
4	+5V			54	OPREQ	Operation Request Out
5	Aux Bus	Auxiliary Power Input		55	R/W	Read/Write Command Out
6	Aux Bus			56	HOLD	HOLD IN (From Memory I/O)
7	Aux Bus			57	JMP CMD	Not Used
8	Aux Bus			58	RUN	Run Condition Output
9	GND	Ground		59	RESET	System Reset In/Out
10	GND			60	JMP ACK	Not Used
11	+12V	Power Inputs		61	INTACK	Interrupt Acknowledged
12	+12V			62	INT 0	Interrupt Level 0 In
13	-12V			63	INT 1	" " 1 In
14	-12V			64	INT 2	" " 2 In
15	GND	Ground		65	INT 3	" " 3 In
16	GND			66	INT 4	" " 4 In
17	A0	Address Bus		67	INT 5	" " 5 In
18	A1			68	INT 6	" " 6 In
19	A2			69	INT 7	" " 7 In
20	A3			70	INT 8	" " 8 In
21	A4			71	INT 9	" " 9 In
22	A5			72	INT 10	" " 10 In
23	A6			73	INT 11	" " 11 In
24	A7			74	INT 12	" " 12 In
25	A8			75	INT 13	" " 13 In
26	A9			76	INT 14	" " 14 In
27	A10			77	INT 15	" " 15 In
28	A11			78	MST PSE	Master Pause CMD from Debug
29	A12			79	DBG INT	Debug Interrupt In
30	A13			80	DBG VEN	Debug Vector Enable Out
31	A14			81	MST INTD	Master Interrupted Out
32	A15			82	FETCH	Not Used
33	CMEM	MSTR/SLV MEM Access Out		83	PAUSE	DMA Pause In
34	RAM INH	Not Used		84	MAST RUN	Master Running Out
35	WD ACCESS	Not Used		85	SENSE	CPU Sense Input
36	D0	Data Bus		86	FLAG	CPU Flag Output
37	D1			87	F.P. HOLD	Front Panel Hold Input
38	D2			88	SLMEMINT	Not Used
39	D3			89	RESERVED	Reserved
40	D4			90	"	
41	D5			91	"	
42	D6			92	"	
43	D7			93	"	
44	D8			94	I/O CLK	40 KHz Clock Out
45	D9			95	CPU CLK	Master CPU 10 MHz Clock Out
46	D10			96	SYS CLK	10 MHz System Clock Out
47	D11			97	GND	Main Bus Ground
48	D12			98	GND	
49	D13			99	GND	
50	D14			100	GND	

FIGURE 3-16 P1 CONNECTOR PIN LIST

CONSOLE INPUT-OUTPUT

P2 PIN	J108 PIN	SIGNAL	DESCRIPTION	COMMENTS
1	1	CHS GND	Not Used	
2	14	TTX	TTY Current Loop Input-	RCVR IN
3	2	EIA RCV DATA	EIA Serial Input	RCVR IN
4	15	TTRDR +	Tape Reader Control Out	47 ohm to +12
5	3	EIA XMIT DATA	EIA Serial Output	EIA Driver
6	16	TTRDR -	Tape Reader Control Out	Open Collector Driver
7	4	REQ TO SND	Request to Send	2.2K Pullup
8	17	TTRCV +	TTY Current Loop Output +	620 ohm to +12V
9	5	CLR TO SND	Clear to Send	2.2K Pullup
10	18	TTRCV -	TTY Current Loop Output	Open Collector Driver
11	6	DATA SET RDY	Data Set Ready	2.2K Pullup
12	19	———		
13	7	SIG GND	EIA Ground	
14	20	DATA TERM RDY	Data Terminal Ready	2.2K Pullup
15	8	CARR DET	Not Used	
16	21	———		
17	9	———		
18	22	RING IND	Not Used	
19	10	———		
20	23	———		
21	11	ORIG	Not Used	
22	24	———		
23	12	LOCAL	Not Used	
24	25	SND RESTRAINT	Not Used	
25	13	TTX +	TTY Current Loop Input +	620 ohm to +12V
26				

FIGURE 3-17a P2 AND P108 CONNECTOR PIN ASSIGNMENT

DISK INPUT-OUTPUT

P3 PIN	SIGNAL	DESCRIPTION
1	DOD7	DATA TO DISK SYSTEM
2	DOD6	
3	DOD5	
4	DOD4	
5	DOD3	
6	DOD2	
7	DOD1	
8	DODØ	
9		
10		
11	$\overline{CTRL}$	CONTROL TO DISK
12	$\overline{CI}$	
13	$\overline{CO}$	
14		
15		
16	$\overline{STRB}$	DATA STROBE TO DISK
17	GND	SIGNAL GROUND
18	GND	
19	GND	
20		

P3 PIN	SIGNAL	DESCRIPTION
21	FLG	(USED FOR TAPE RDR RDY)
22	TAPE	(TAPE STATUS BIT)
23		
24		
25		
26		
27	PFLT	(TAPE STATUS BIT)
28		
29	DID7	DATA FROM DISK SYSTEM
30	DID6	
31	DID5	
32	DID4	
33	DID3	
34	DID2	
35	DID1	
36	DIDØ	
37	GND	SIGNAL GROUND
38	FLG	STROBE ACK FRM DISK
39	PBZY	PRINTER BUSY
40	PFLT	PRINTER FAULT

FIGURE 3-17b P3 CONNECTOR PIN ASSIGNMENT

Chapter 4

DEBUG AND FRONT PANEL I/O MODULE

4.0 INTRODUCTION

The Debug and Front Panel I/O module performs three distinct functions within the development computer: a) It controls the interaction and bus time-sharing of the Master and Slave CPU's, b) It supports such software Debug features as breakpoint, trace, and start of Slave programs at any memory location, c) It provides interface to either the Standard or Full Display Front Panel and supports all Full Display Panel functions. The card is centrally located in the computer bus structure, providing a dividing line for Master and Slave halves. Unique control lines are connected to the Master CPU and Slave CPU in each half of the bus.

4.1 GENERAL DESCRIPTION

A block diagram of the board is shown in figure 4-1. Although the logic sections are interrelated, the board is described under four major topics:

1) Master/Slave control
2) Debug Features
3) I/O commands and interrupts
4) Front Panel interface

Master/Slave control section determines bus control between Master and Slave CPU's at all times. Debug features include storage of the Slave CPU address value (program counter) during instruction fetch of the last instruction and present instruction. Also included are two breakpoint registers with address comparison logic, and forced jump logic including jump address storage. The I/O consists of port decoding for storing of breakpoint values, loading the Debug command and control bytes, and reading the stored program counter address values. Slave I/O service requests are also decoded. Debug interrupts, breakpoint, single cycle, Slave halted as well as the eight service requests are priority vector encoded for presentation to the Master CPU. The Front Panel interface routes bus address and data to the Front Panel and interfaces data and address values from the Front Panel during memory or I/O access. It also supports various Front Panel control operations such as single step.

4.1.1 Master/Slave Control Master/Slave control, shown in the shaded portion of figure 4-2, is straightforward. The Master relinquishes bus control by executing a program HALT. The Slave is then allowed to RUN and the Master is PAUSED until a Master Interrupt occurs. This may be due to a breakpoint or other Debug interrupt, a service request, a console interrupt or an interval timer interrupt. The Slave is paused, and when off the bus, the Master is again allowed to RUN.

Both Master and Slave may be deactivated by a Front Panel access signal or by a DMA controller causing a PAUSE to both Master and Slave. When the PAUSE is removed, bus control is resumed in the state prior to the PAUSE. The Full Display Panel can also set the state of the Master/Slave control to either Master or Slave.

4.1.2 Debug Features Certain software Debug routines require hardware support that is not necessarily unique to a particular Slave processor. These features are contained on the Debug module. The module contains two breakpoint registers with a memory address comparator, two Slave program counter storage registers, and a forced jump address register. It also includes logic to provide single cycle interrupts during full trace operation, a forced Slave reset and forced Slave interrupt logic. The Debug Logic is indicated in the shaded portion of figure 4-3.

A central feature of the Debug Logic is use of an internal 8-bit bus to handle all address and data flow. File registers organized into four words by eight bits are used to write and read breakpoint values and Slave program counter values. A bus controller containing sequencing logic controls

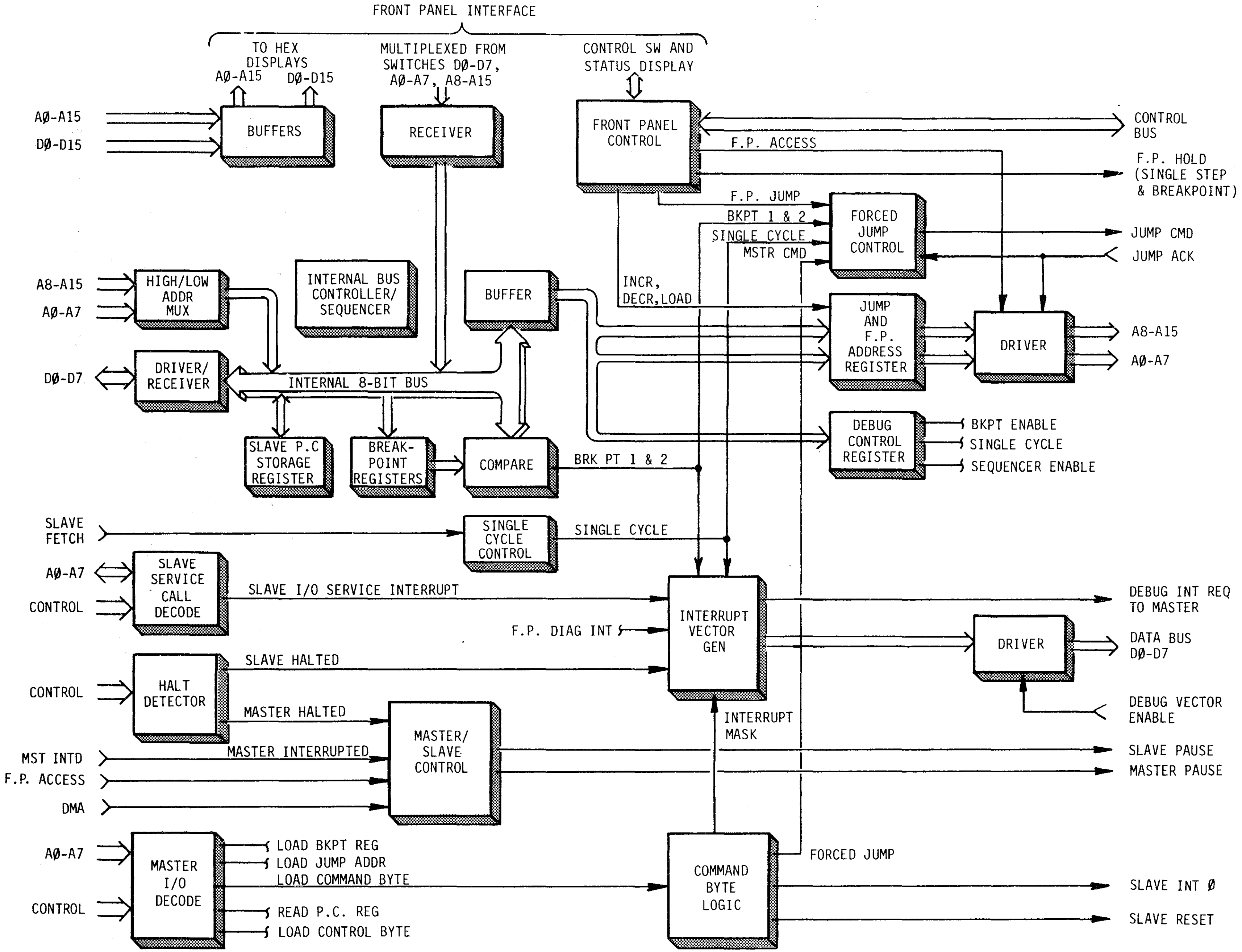


FIGURE 4-1 DEBUG AND FRONT PANEL I/O BLOCK DIAGRAM

bus flow and has several modes of operation. The Front Panel also makes use of this bus structure to load the dual-purpose address register and place data on the system bus. The sequencer operates only when commanded by Debug mode or during Front Panel access. A normal sequence is triggered by a Slave OPREQ. The low and high address bytes are sequentially placed on the internal bus and compared to breakpoint address 1. At the same time, the address bytes are loaded into one of the program counters. Then the two address bytes are again sequentially accessed for comparison to breakpoint address 2. If the breakpoint conditions specified by the command byte are met, the Slave is paused and the Master interrupted.

Forced jump logic is partially contained on the Slave CPU and partially on the Debug module. The Slave contains a jump sequencer and data multiplexing logic that when triggered, substitute an unconditional branch for the next instruction fetch and sequentially feed the following two jump address bytes to the CPU data bus. The jump address is stored in the dual purpose address register on the Debug module. Upon Master I/O commands, the address is loaded and a jump command trigger given to the Slave CPU. During the sequence, the Debug module places the jump address on the address bus for use by the Slave multiplexer.

Other Debug features include single cycle mode, forced Slave reset and forced Slave interrupt. During single cycle, the Slave is paused on execution of a single instruction and the Master is interrupted. Single cycle or breakpoint interrupts to the Master also initiate a forced jump sequence for the Slave CPU. The Slave processor, when paused, is designed to stop after fetching the first byte of the next instruction. Since it is necessary to jump the Slave to a register dumping routine immediately without an intervening instruction, the forced unconditional branch command must be received by the Slave for its next instruction before stopping. The Master takes control and loads the Debug module with the jump destination. The Master then halts, reenabling the Slave, which jumps to the register dump routine.

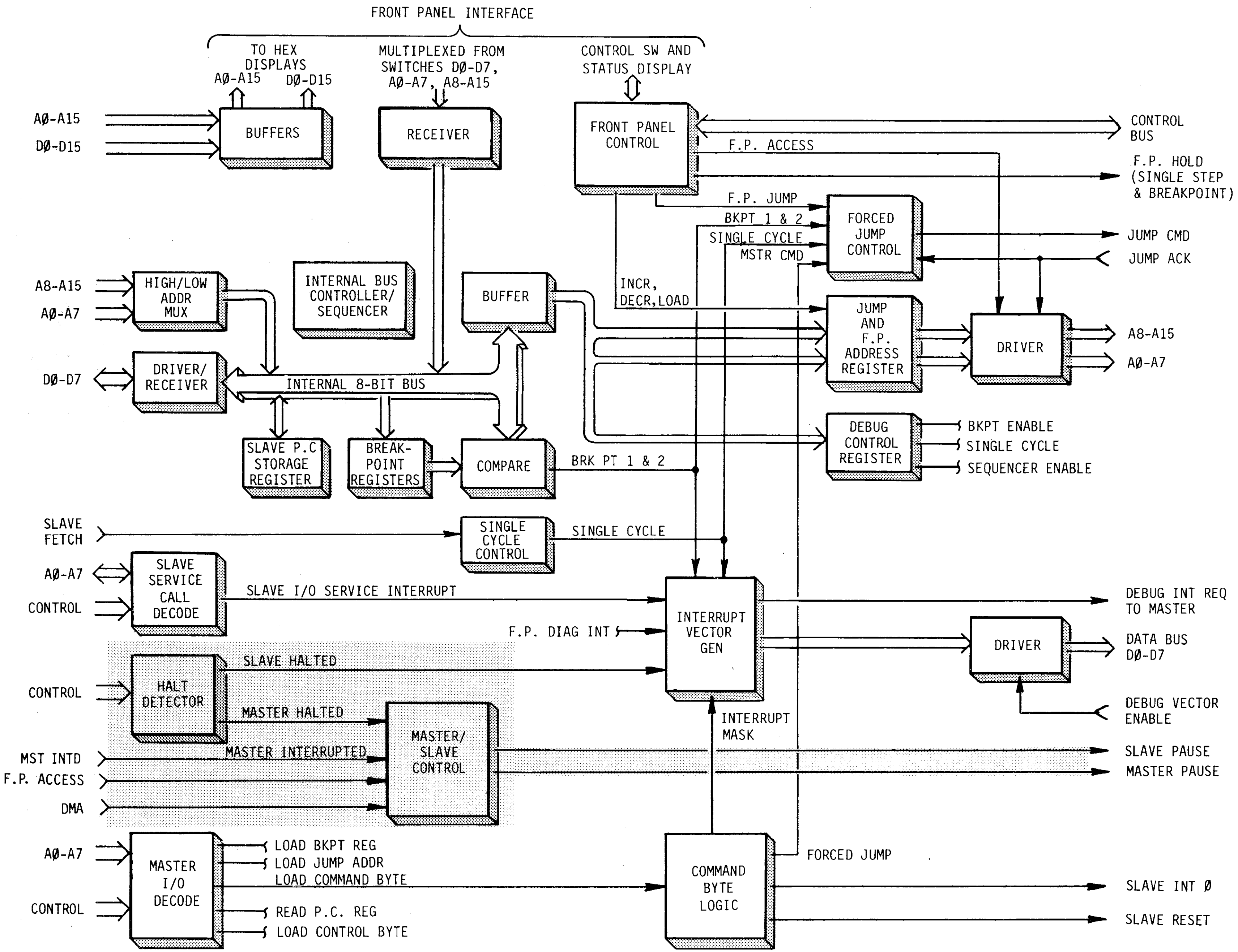


FIGURE 4-2 MASTER/SLAVE CONTROL

4.1.3 I/O Commands and Interrupts Figure 4-4 has this section shaded on the overall block diagram. I/O commands to the Debug module come only from extended Master CPU I/O commands. These include the following:

1) 4 bytes output breakpoint (1 and 2, low and high byte)
2) 4 bytes input program count (next and last, low and high byte)
3) 2 bytes output jump address
4) 1 byte output Debug command
5) 1 byte output Debug control

Breakpoints, program counter values, and jump address have been discussed in section 4.1.2. The command byte contains individual bits commanding a Slave forced reset, forced interrupt, forced jump and a Debug interrupt mask bit. The control byte sets the Debug mode to active or inactive mode and controls single cycle and breakpoint modes.

The interrupt structure can be divided into two sections. Slave I/O commands, FØ-F7, called service requests, directly produce interrupt vectors 0020 thru 002E (Priority levels 16-23). Debug interrupts consisting of the two breakpoints, single cycle, Slave halted and Front Panel diagnostic interrupt are priority vector encoded for addresses 0030-0038 (priority level 23-28). Presence of any interrupt produces a single Debug interrupt to the Master CPU. Priority is assigned below the 16 level Master interrupt I/O structures. If no higher level is pending, the CPU gives priority to the Debug card during INTACK allowing the Debug module to place the vector on the data bus and reset the interrupt.

4.1.4 Front Panel Interface The Debug module contains interface capability for the Full Display Front Panel, allowing substitution for the Standard Front Panel at any time. Since the console provides all operator interface and display necessary during normal operation, the Standard Front Panel contains minimal controls. The Full Display Panel,

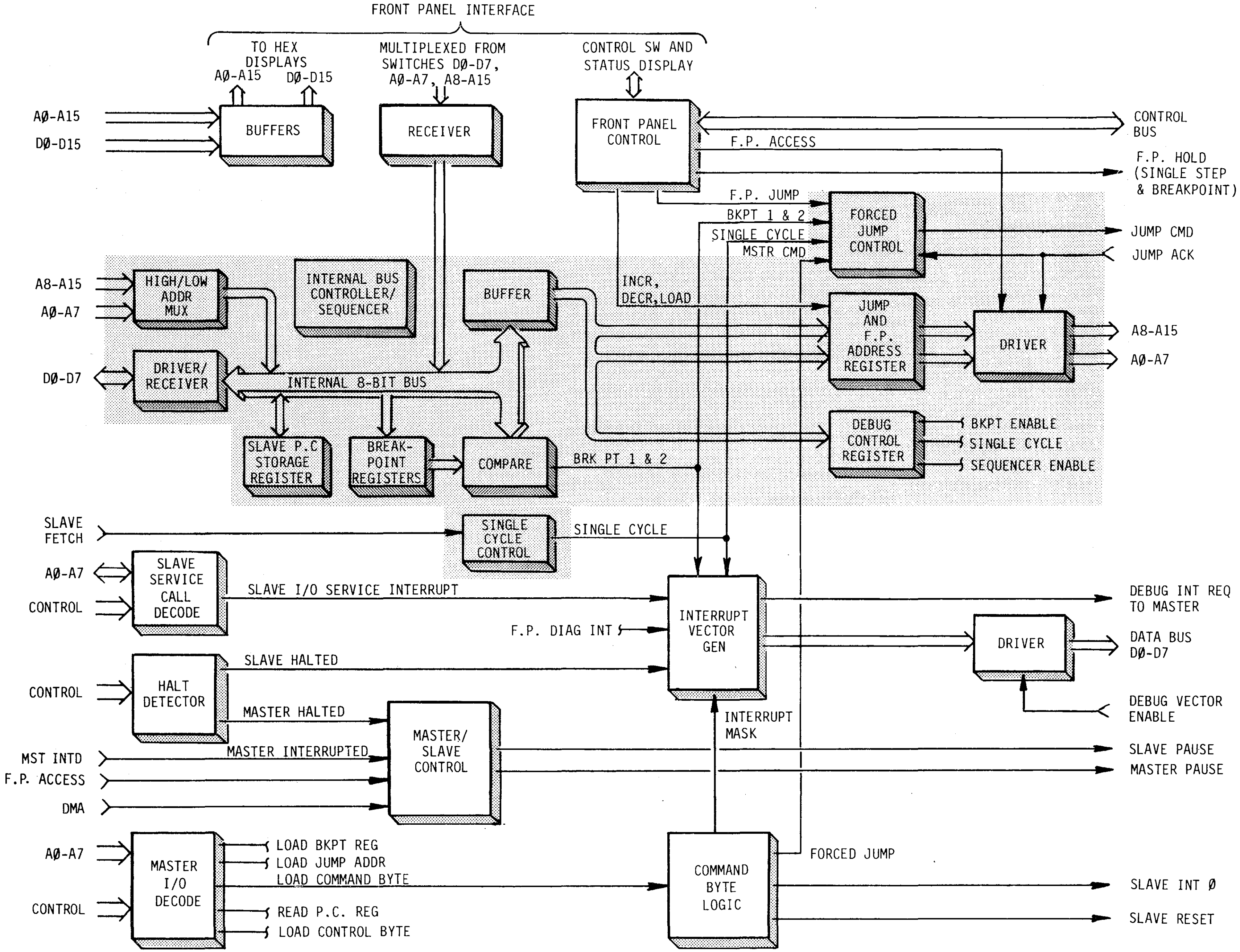


FIGURE 4-3 DEBUG LOGIC

on the other hand, contains full hexadecimal data and address displays and individual switches for data and address entry. Full control functions such as Front Panel Access of memory or I/O ports, single step and hardware breakpoint are included.

The Front Panel interface section, shaded in figure 4-5, contains buffers to route the 16 address and up to 16 data bits to the Front Panel displays. Data and address switch values are multiplexed in 8-bit bytes to the internal Debug module bus as previously discussed. Most of the logic necessary for the Front Panel functions is contained on the Front Panel circuit boards, but some interaction with the Debug control functions is necessary. These are Master/Slave control, the forced jump feature, diagnostic interrupt and a special function called Master Test which allows display of the last address value executed.

4.2 DETAILED DESCRIPTION

4.2.1 Master/Slave Control and Halt Detector

A simplified logic diagram of the Master/Slave Control Logic is shown in figure 4-6. One line to each half of the motherboard controls Master/Slave operation. A PAUSE to the Master or to the Slave causes it to relinquish bus control and causes the output of all address, data and control lines to go into a high-inpedance state at the end of the current instruction.

As previously discussed, both Master and Slave are PAUSED during Front Panel Access, or for DMA operations. Otherwise, the Debug control logic takes precedence. A single flip-flop stores Master/Slave status. When an event occurs to change the status, the active device is directly PAUSED. When out of the RUN state, only then is the STATUS flip-flop clocked to the new state allowing the inactive device to RUN. The Slave is PAUSED if a Master interrupt occurs or if the Front Panel "SET MSTR" switch is depressed. The Master is PAUSED is a Master HALT is detected or "SET SLV" switch on the Front Panel is depressed.

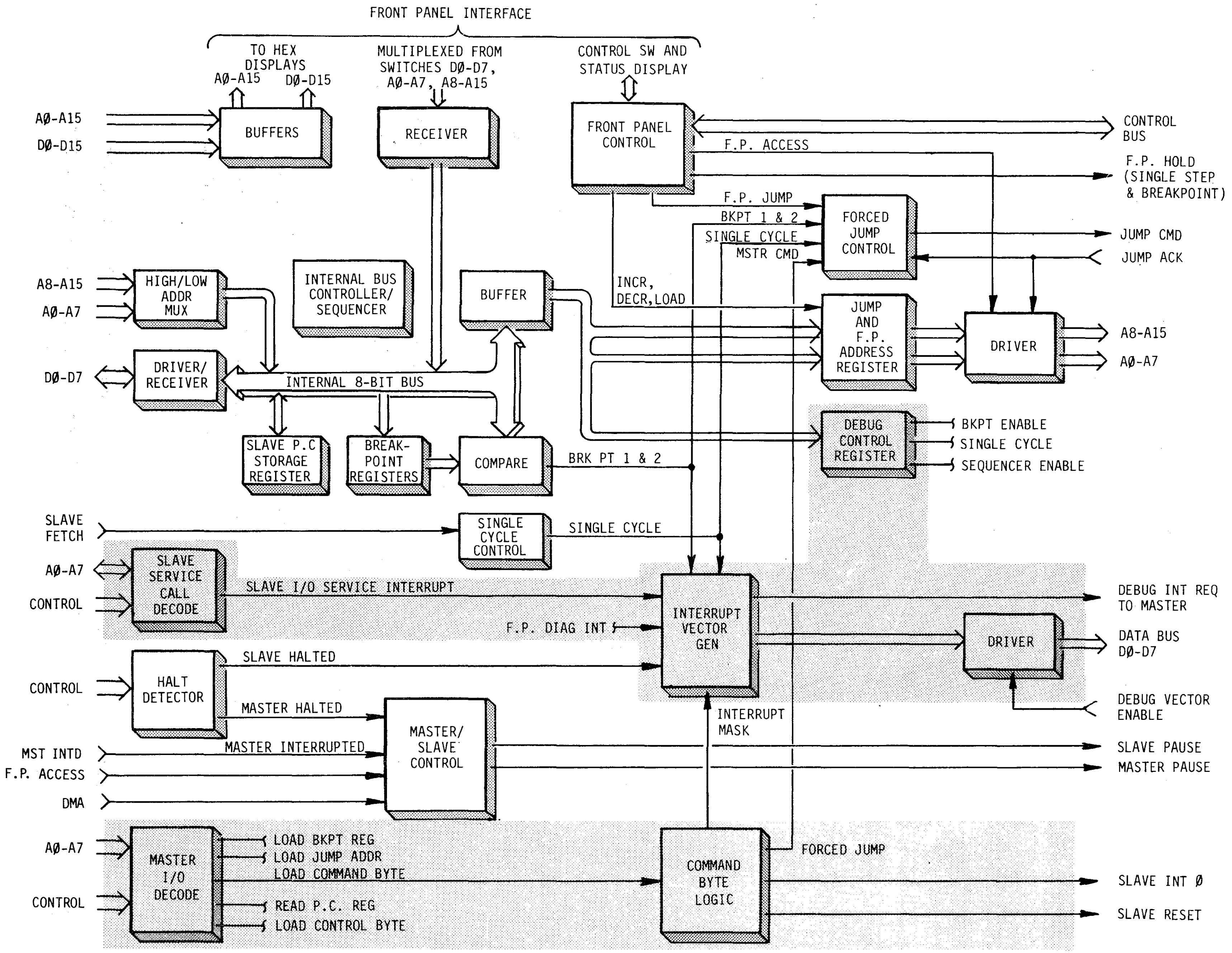


FIGURE 4-4 I/O COMMANDS AND INTERRUPTS

The Master halt detector is shown in figure 4-7. A Master HALT is the normal mechanism the Master uses to relinquish bus control. A Slave HALT indicates a programming error and immediately causes a Master interrupt, so the user can be notified. Operation of the halt detectors is identical. If either the Master or Slave is in RUN, the respective halt indicators are held in a reset condition. If the RUN indication is not present and no PAUSE has been issued for at least 26 μs, the halt detected flip-flop is set. Two flip-flops are required to provide the necessary 26 μs delay.

4.2.2 I/O Commands and Interrupts Debug features are controlled totally by the Master CPU module. To do this, I/O commands controlling the operation and the register contents are issued by the Master CPU. A list of input-output port assignments is shown in figure 4-8a. The breakpoint addresses are loaded by the Master CPU upon user request. The forced jump address may be loaded to execute a "GO" command from the user or to jump the Slave to or from a trace routine. The program counter contents are read only during trace or after detecting a breakpoint.

The command byte, shown in figure 4-8b, consists of four individual commands:

Forced Interrupt Command -- This command forces a Slave processor interrupt so that when the Slave processor is started, a forced ZBSR instruction at page 0 location 0 is issued.

Forced Jump Command -- This command forces the Slave processor to branch unconditionally to the address contained in the dual purpose address register. The address register must be set prior to issuing this command.

Forced Reset Command -- This command forces the Slave processor to be reset. The Slave processor will start instruction execution

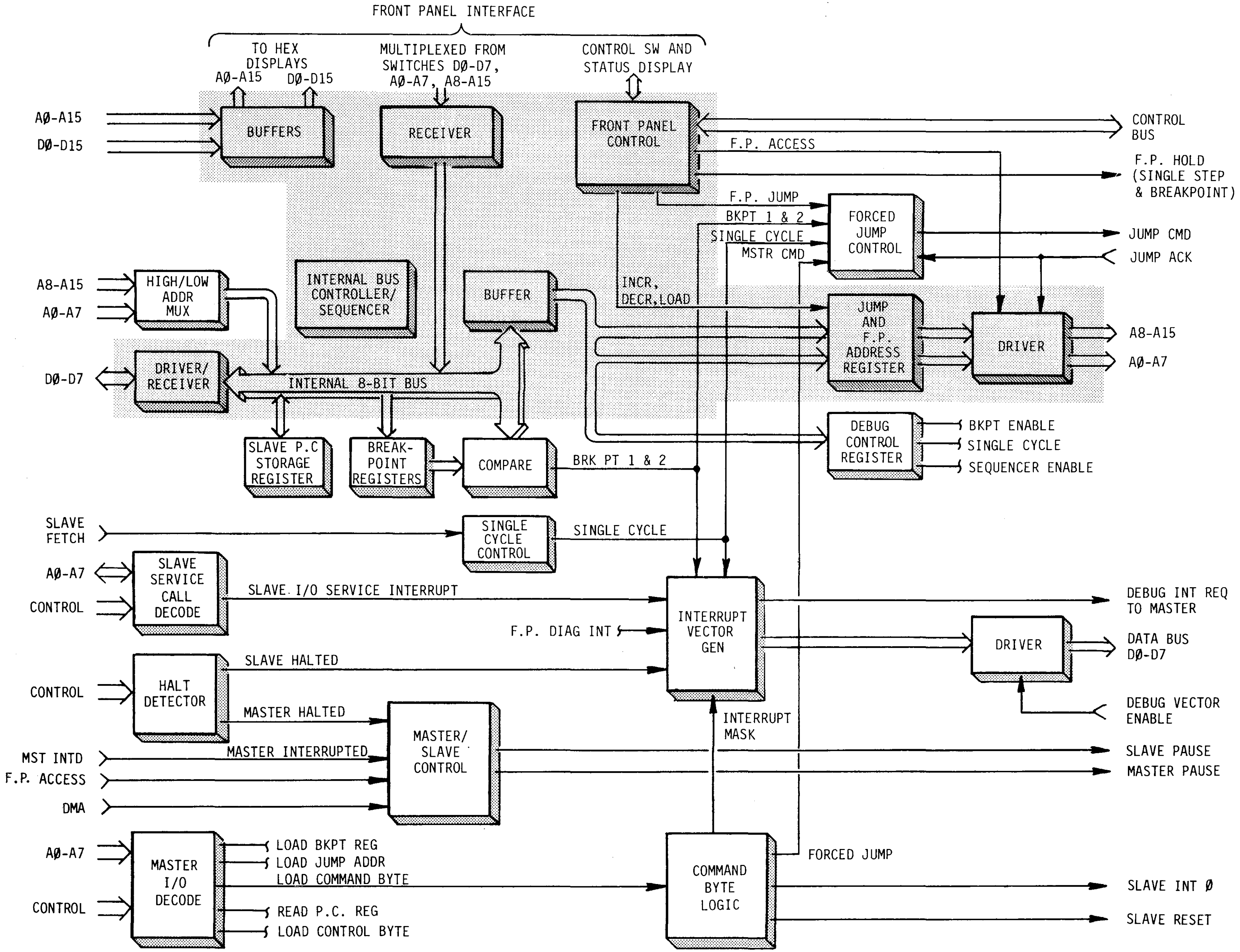


FIGURE 4-5 FRONT PANEL INTERFACE

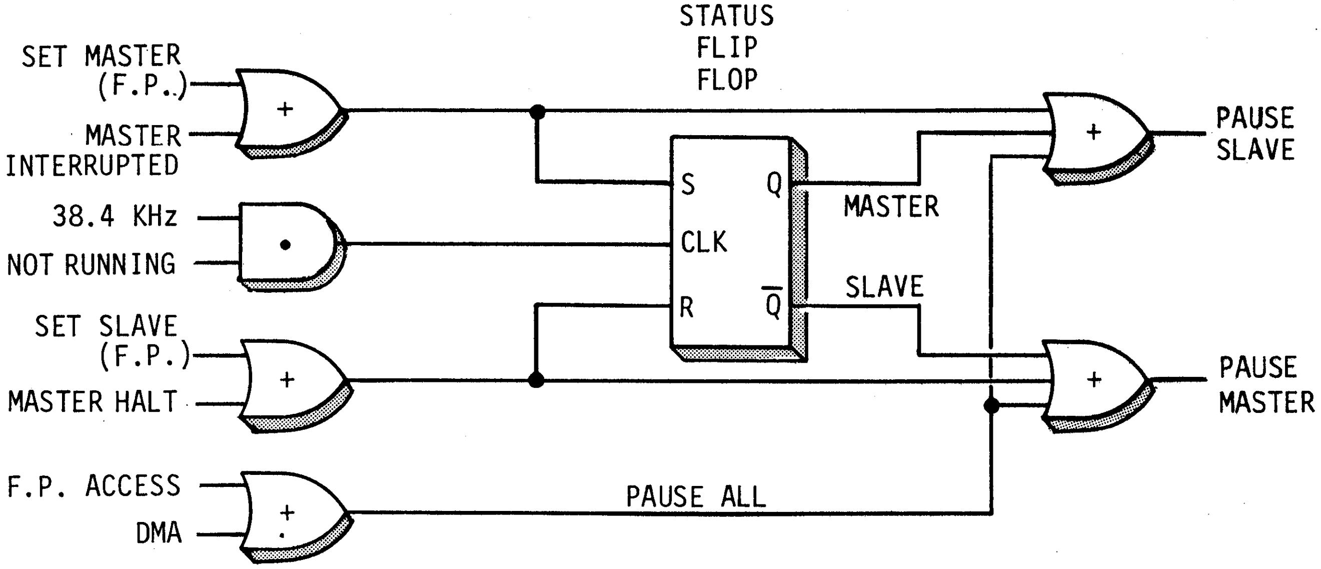


FIGURE 4-6 MASTER-SLAVE CONTROL

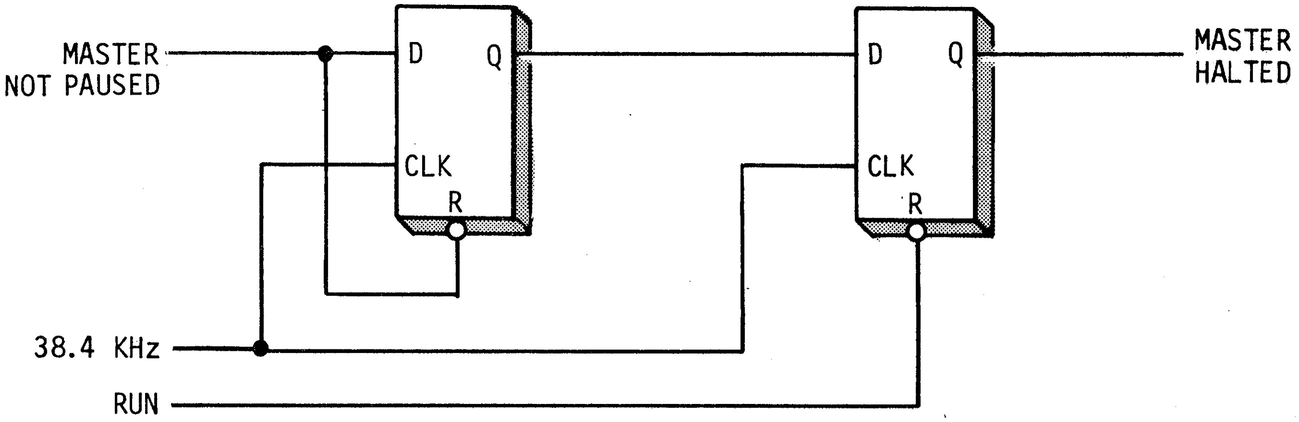


FIGURE 4-7 MASTER HALT DETECTOR

at page 0 and location 0. This command can be used together with a forced interrupt or forced jump command.

Mask Debug Interrupts Command -- This command sets a condition that will not allow any Debug hardware interrupt to occur. The masked interrupt is stacked and is allowed to occur when the mask is reset.

OUTPUT FROM CPU	Command byte	F9
	Control byte	F8
	Jump addr. low, high	FA, FB
	Breakpoint 1 low, high	FC, FD
	Breakpoint 2 low, high	FE, FF
INPUT TO CPU	Program counter LAST low, high	FC, FD
	Program counter NEXT low, high	FE, FF

FIGURE 4-8a INPUT OUTPUT PORT ASSIGNMENTS

7	6	5	4	3	2	1	0		
1	1	1	1	1	1	1	0	Forced Reset	FE
1	1	1	1	1	1	0	1	Forced Interrupt	FD
1	1	1	1	1	0	1	1	Forced Jump	FB
1	1	1	1	0	1	1	1	Interrupt Mask	F7

FIGURE 4-8b COMMAND BYTE FORMAT

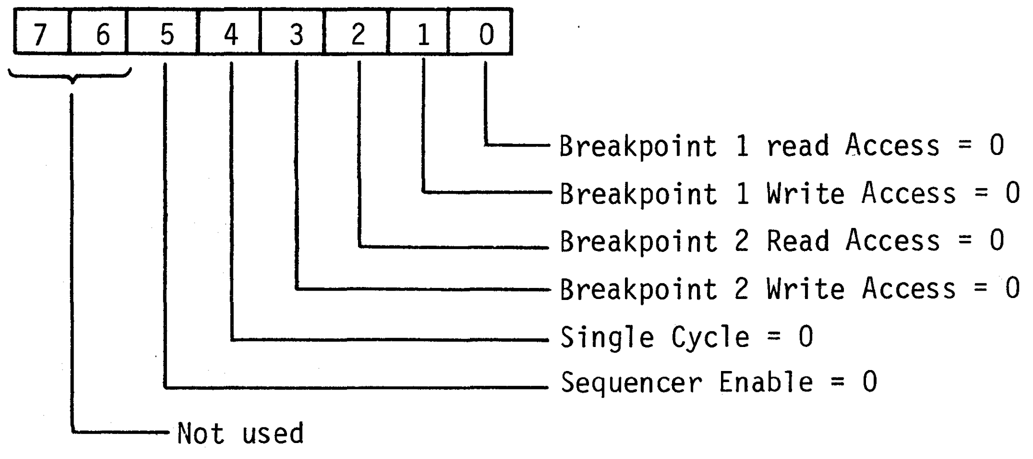

7
6
5
4
3
2
1
0
Breakpoint 1 read Access = 0
Breakpoint 1 Write Access = 0
Breakpoint 2 Read Access = 0
Breakpoint 2 Write Access = 0
Single Cycle = 0
Sequencer Enable = 0
Not used

FIGURE 4-8c DEBUG CONTROL BYTE FORMAT

The control byte contains six individual bits controlling the Debug mode of operation. The format is shown in figure 4-8c. The Debug mode enables or disables the sequencer loading the program counter values and comparing breakpoint address. The other bits are self-explanatory, involving trace or breakpoints upon memory read or write. The bits are non-exclusive and any combination of breakpoint bits may be used.

All commands decode the lower eight address bits to channel the data to or from a particular register and require Master active, Extended I/O, OPREQ and WRP (if an output).

Interrupts -- Interrupts from the Debug module are initiated by two sources: Slave Service Requests or by Debug Features. A simplified block diagram is shown if figure 4-9. Slave service requests utilize the I/O module address recognition, but require the Slave to be active rather than the Master. When a Slave I/O write from FØ-F7 is detected, a "DBG INT" goes to the Master CPU. Priority is established by a combination of the vector encoders on the Master CPU and on the Debug module. A list of Master interrupts is shown in figure 4-10. When the CPU acknowledges the interrupt with "DBG VEN" (Debug Vector Enable) the lower three address bits of the I/O service request (AØ-A2) are routed to the data bus as D1, D2, and D3 in the inverted form. Bit 5 is always set to "one" and bit 4 is set to "zero" for service requests and one for Debug interrupt. Note that a Slave I/O command to port 7 corresponds to Slave service request 1 and vector 0020. The trailing edge of "DBG VEN" clocks the register holding the service request vector back to zero.

The Debug feature interrupts start with individual flip-flops set by an event. Breakpoint 1, Breakpoint 2, Single Cycle, Slave halted, and the Front Panel DIAG INT switch each set a particular flip-flop.

A priority encoder receives the output of the flip-flops and generates a "DBG INT" to the Master. Action is as previously described for the

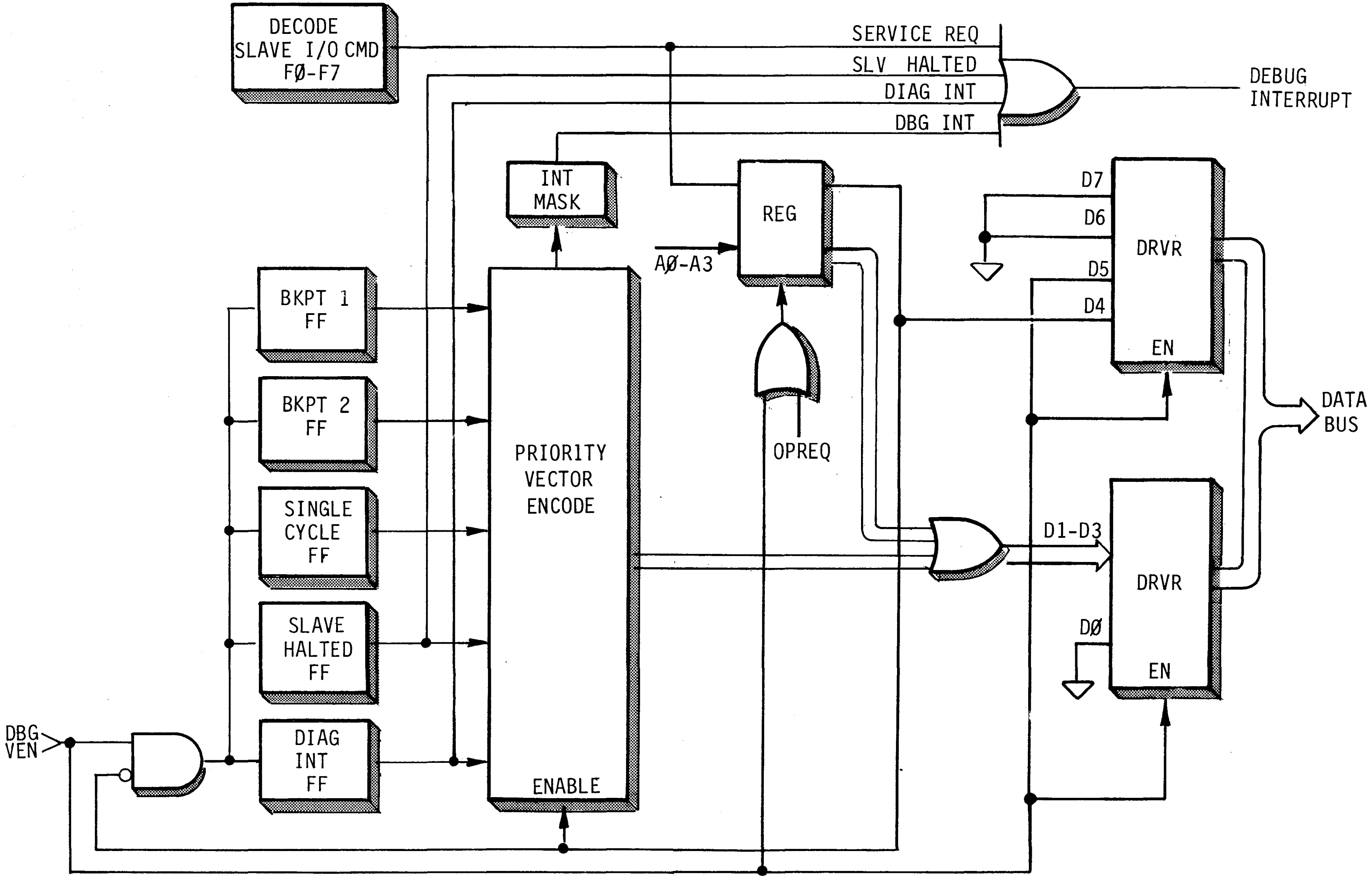


FIGURE 4-9 DEBUG INTERRUPT LOGIC

Priority	Vector Address	Function	
0	0000	Reset	MASTER I/O
1	0002	Master Memory Parity Error	
2	0004	Paper Tape Reader	
3	0006	Slave Memory Parity Error	
4	0008	TTY In(Console)	
5	000A	TTY Out(Console)	
6	000C	RS-232 In	
7	000E	RS-232 Out	
8	0010	NA	
9	0012	Timer	
10	0014	Printer	
11	0016	Floppy Disk	
12	0018	NA	
13	001A	PROM Program 1	
14	001C	PROM Program 2	
15	001E	NA	SERVICE REQUESTS
16	0020	Slave SVC 1	
17	0022	Slave SVC 2	
18	0024	Slave SVC 3	
19	0026	Slave SVC 4	
20	0028	Slave SVC 5	
21	002A	Slave SVC 6	
22	002C	Debug SVC 1	
23	002E	Debug SVC 2	DEBUG HARDWARE
24	0030	Breakpoint 1	
25	0032	Breakpoint 2	
26	0034	Single Cycle	
27	0036	Slave Halted	
28	0038	Diag. Int	
29	003A	Debug Int 29	
30	003C	" " 30	
31	003E	" " 31	

NA = NOT ASSIGNED

FIGURE 4-10 INTERRUPT PRIORITY ASSIGNMENTS AND ADDRESS VECTORS

service requests. The highest priority interrupt present when "DBG VEN" is received is encoded and placed on the data bus. Note that there is an interrupt mask, masking only the breakpoint and single cycle interrupts. This is controlled by bit 3 of the command byte. All Debug interrupt flip-flops are reset simultaneously, thus only the highest priority interrupt is seen by the Master CPU.

4.2.3 Debug Features As previously listed, the Debug hardware consists of two Slave instruction address registers (program counters), two breakpoint registers and associated address comparators, forced jump address storage and control, and Slave single cycle control. These functions are enabled or disabled by the Debug Control byte. Additionally covered under this topic are Slave forced reset and forced interrupt.

Since most of the logic revolves around the internal 8-bit bus and the bus controller or sequencer, this will be discussed first.

Internal Bus and Sequencer -- The internal bi-directional bus is connected to the motherboard data bus by bus driver-receiver pairs. It is connected to both the input and output of the file registers used for program counter storage. It is connected to the input of the breakpoint file registers and one input of the comparators. Through buffers it is further connected to the dual purpose (jump address/front panel) register and to the Debug control byte register. Refer to figure 4-3.

The data bus connection is used during the following operations:

1) Reading the P.C. file registers
2) Loading the Breakpoint file registers
3) Transferring Front Panel switch data to the bus
4) Loading the Debug Control byte
5) Loading the Jump Address register

These operations are controlled by the I/O decode logic except item 3. During Front Panel Access, an ENTER command causes data to be multiplexed from the Front Panel data switches, enabled onto the Debug internal bus and out to the motherboard bus.

During sequencer operation, the data bus connection is not used. Rather, the address bus is multiplexed onto the internal bus eight bits at a time. When the Debug mode is enabled, the sequencer, utilizing a 10 MHz clock, is triggered by the leading edge of "OPREQ". The timing diagram is shown in figure 4-11. A hold is issued to insure that the address data lasts for the duration of the sequencer operation. Four flip-flops generate the basic timing sequence and are listed as Q1-Q4. There are five basic intervals. During the first interval, the low half of the address bus is output to the internal bus. This is stored in the low byte of one of the program counter file registers. At the same time, the low address is accessed from breakpoint 1 file register and compared to the address on the internal bus with exclusive OR gates. The result of this comparison is clocked into the first stage of a compare circuit. In the second interval, the address byte is written into the high byte of the same program counter file register and the high address is accessed from the breakpoint 1 register and compared to the high address byte on the bus. If the last compare is valid, the result is clocked into the second stage of the compare circuit. During the third interval, the breakpoint 2 low address byte is accessed while the result of the last two breakpoint 1 comparisons is gated to interrupt circuitry. This sequence continues and on cycle 5 the result of the breakpoint 2 comparisons is gated to the interrupt circuitry. For logic simplicity, the program counter file register is written into five times, three times with the same low byte information and twice with the high byte. Program Counter Register "Next" is normally written into. Program Counter "Last" is only written into during an instruction FETCH when a Master interrupt is not pending. Thus, during an instruction where a breakpoint occurs, Program Counter Last is written into during the instruction fetch before the breakpoint occurs. Program Counter Next is written into on the subsequent cycles of the instruction through the next instruction FETCH. Thus, when the Master CPU is interrupted, Program Counter Last contains the address of the instruction that was just completed and Program Counter Next contains the address of the instruction that the Slave must return to after register dump and restore routines.

The sequencer is also activated in two other situations. During Front Panel Access, a "LOAD" command starts the sequencer. The normal address bus multiplexer is inhibited and instead, the Front Panel data containing the low and high bytes of the Front Panel address switches is enabled. Instead of loading the Program Counter file register, the low and high bytes of the address register are sequentially loaded (again five times). The other situation occurs when the Front Panel "MSTR CPU TEST" switch is activated. Operation is identical to normal sequencer operation except that the Address register is loaded from the address bus just as the Program Counter is loaded. Slave operation is not required. This feature allows Front Panel viewing of the last address executed by either CPU befor a halt and is used for detailed troubleshooting.

Breakpoint and Program Counter Registers -- As discussed in the last section, the breakpoints and Slave Program Counter values are stored in a four word by 8-bit file register. This structure economizes logic for busing, storage and comparison. As discussed, the low and high bytes of each breakpoint are sequentially compared. If the particular breakpoint is enabled for the Slave operation in progress, it is allowed to set the associated interrupt flip-flop. This occurs during cycle 3 of the sequencer for breakpoint 1 and cycle 5 for breakpoint 2. The compare is inhibited during forced jumps or if an interrupt is already pending. Breakpoint registers are loaded only by the Master CPU and cannot be read by an I/O command.

Program Counter value storage allows the Master CPU to trace Slave program execution. Some Slave CPU's have no provision for accessing the internal program counter directly, thus auxiliary logic must be used.

The 2650 Slave CPU stops when PAUSED after fetching the next instruction, making it difficult to determine the address of the last instruction where the breakpoint occurred. Thus the hardware implements storage of both the last instruction address values and the next or just fetched instruction address. The Program Counter file register contents are read only by the Master CPU.

Note that neither the breakpoint or Program Counter registers operate unless the Debug mode is enabled. Note also that the sequencer may slow the operation of Slave processors due to the required 600 ns HOLD time.

Single Cycle -- The Single Cycle Logic consists of a single flip-flop enabled by the control byte. A Slave FETCH clocks the flip-flop to an active state, causing a Master interrupt. This automatically PAUSES the Slave CPU at the end of the single instruction.

Forced Slave Jump -- The forced Slave jump is initiated by any one of several sources. The single cycle, either breakpoint, Front Panel jump logic, or a Master I/O command have the same effect. Two flip-flops generate a 100 ns "JMP CMD" to the Slave CPU which arms its jump mechanism. A JUMP PENDING flip-flop is also triggered to disallow a Single Cycle interrupt. After receipt of the Jump Command, the Slave outputs a forced branch instruction on the Slave data bus during the next instruction fetch. The absolute jump address is supplied by the Debug Jump Address register. On the leading edge of the following OPREQ in which the CPU receives the high byte of the jump address, "JMP ACK" is generated. The Debug card uses this signal to gate the address register ontp the address bus. The Slave multiplexes the high and low bytes of the address bus into the data bus as required for the particular Slaves JUMP instruction and resets "JMP ACK". Receipt of "JMP ACK" also resets the JMP PENDING flip-flop on the Debug module.

Forced Slave Reset and Forced Interrupt -- The Master CPU may command a Slave RESET at any time. This is one bit of the command byte. A flip-flop is set on receipt on this command or by system RESET. This immediately activates the Slave RESET line. The RESET is removed when the Master/Slave control flip-flop enables the Slave CPU. This avoids bus conflict since the Slave tries to fetch one byte after RESET even through PAUSED and must be held in a RESET state until the Master is off the bus.

Slave forced interrupt is simply a pulse generated by bit 1 of the I/O command byte. The interrupt is sent out on the Slave interrupt level zero line. It is stored and processed on the Slave CPU card.

4.2.4 Front Panel Interface It should be reemphasized that the majority of the Front Panel Logic is needed to support the full display panel. The discussion of the Debug module is in respect to the Full Display Panel, since only in so doing are all features discussed. The Standard Front Panel contains only the Reset switch, Diagnostic Interrupt switch and status lights for MSTR ACTIVE, SLV ACTIVE, RUN and POWER ON. Other Front Panel features common to both panels such as the PROM sockets, PROM Power switch primary AC power switch are not connected to the Debug module. Refer to figure 4-5 during the following discussion.

Data Paths -- Address and data display on the Front Panel comes directly from the computer motherboard bus. These signals are buffered by inverter gates on the Debug module and sent directly to the Front Panel. Data from the 8 bit data switches and 16 bit address switches is multiplexed by the Front Panel into a single eight bit bus. As previously discussed, the sequencer controls loading of the address switch data into the dual function address register. An "ENTER" or "ENTER NEXT" command from the Front Panel allows data onto the internal debug bus, then onto the system bus for writing into memory.

Status Display -- Most of the display functions to the Front

Panel are buffered directly from the system bus. These are R/$\overline{W}$ (read/write), $\overline{M}$/I/O (Memory/IO), $\overline{HOLD}$, CMEM, SLAVE INTACK and SLAVE FETCH. The RUN signal is further gated on the Front Panel by HOLD during single step. The Master and Slave active indicators come directly from the Master-Slave control flip-flop on the Debug module.

Control Switches -- The Front Panel Control switches are divided into three groups. System control, CPU control, and Bus or Access control. All control switches except CMEM, M/IO and MSTR TST are electrically debounced on the Front Panel.

The system control contains the RESET switch, BRKPT switch and RUN-STEP switch. The RESET signal, used to initialize all systems status, is driven directly onto the bus through a tristate driver. The BRKPT switch enables the Full Display Panel hardware comparator which compares the system address bus to the address swtiches and issues a HOLD to the CPU if a comparison is made. The RUNOSTEP switch puts a hold on the active CPU in the center position and when depressed to step, it resets the hold until the trailing edge of OPREQ. The step switch clocks the Front Panel flip-flop disabling the HOLD STATE on the Debug module. This allows the CPU to proceed to the next cycle. The Front Panel flip-flop is again set by OPREQ. Other Front Panel functions also disable the HOLD gate in order for the CPU to finish its instruction and get off the bus. HOLD is intended for short term holding devices such as memories. F.P. HOLD is generated by ANDing HOLD with OPREQ and is used by CPU's which must stop their clock for long duration HOLDs.

CPU control functions include the MSTR-SLV switch, JUMP switch, TEST switch and DIAG INT switch. The MSTR-SLV switch is enabled only in WAIT mode. This sets the Master-Slave Control flip-flop PAUSING one CPU and enabling the other.

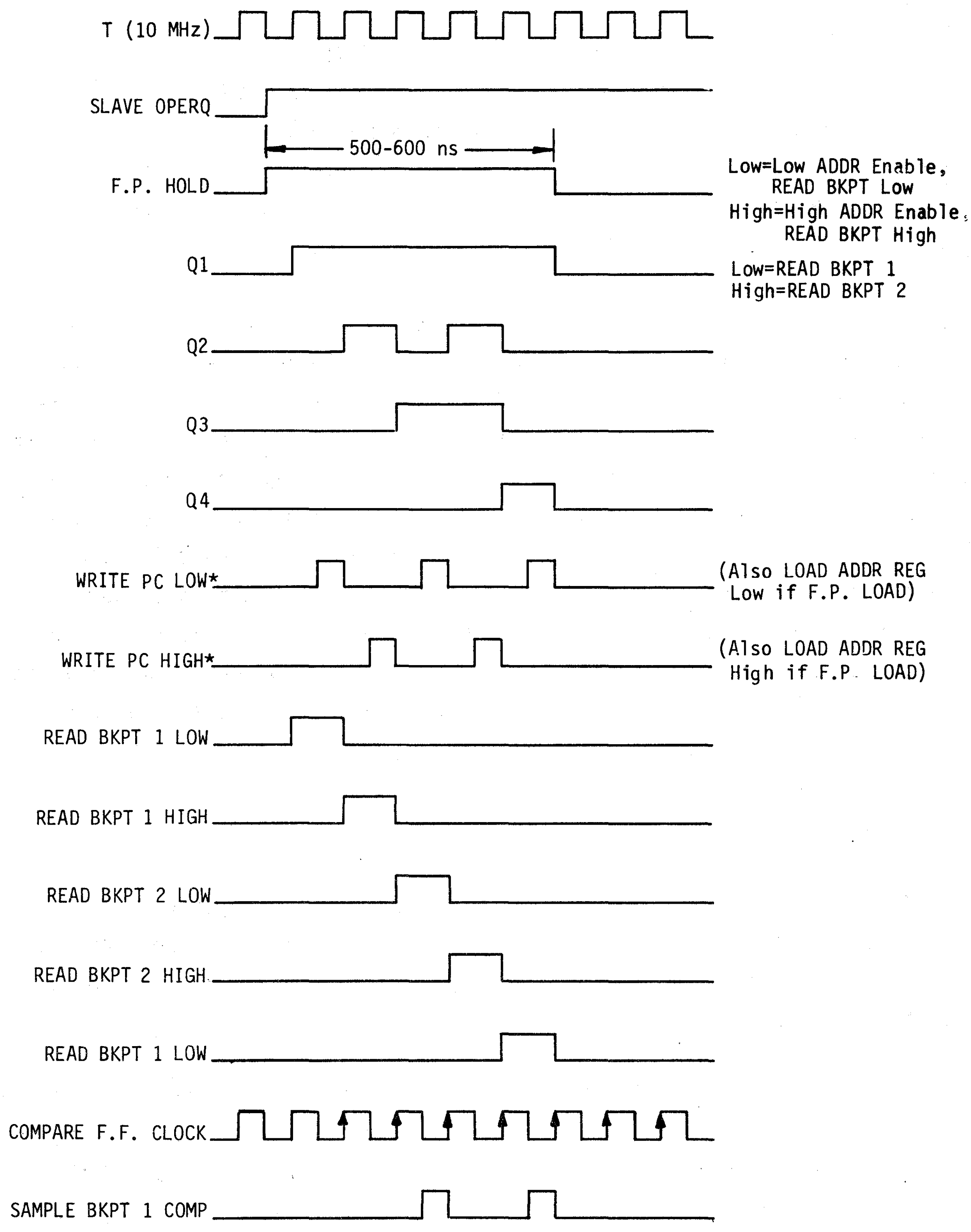
The JUMP switch, as previously discussed activates the INITIATE SLAVE JUMP flip-flop on the Debug module. The content of the address register is used as the Slave destination. The MASTER TEST switch activates the Debug sequencer in all modes, so that the address register always contains the last address accessed by a CPU. This was discussed under the sequencer section. The DIAG INT switch sets an INTERRUPT flip-flop on the Debug module which utilizes the interrupt vector generator on the Debug module.

Bus Control switches are used to access memory or I/O ports. The ACCESS switch enables all bus control switches and PAUSES both CPU's through the Debug CPU control logic. When the CPU's are out of RUN, the address register and bus control signals from the Front Panel and are gated through tristate devices onto the bus. CMEM and MEM/IO are gated directly from the Front Panel switches. The LOAD ADDR switch initiates the sequencer which loads the dual purpose address register with the Front Panel address switch values. INCR or DECR causes an up or down count to the address register. ENT (enter) triggers a WRITE signal to the bus as well as a WRP. At the same time, it utilizes the internal Debug bus to multiple the eight data switches onto the bus. ENT NXT (enter next) does the same thing except the address register is incremented first.

4.3 UTILIZATION

4.3.1 Installation The module is designed to be plugged into the card file of the development computer, but may be used in other applications. All interface connections and connector pinouts are detailed in section 4.3.3.

COMPUTER INSTALLATION The Debug and F.P. I/O module plugs into slot J9 of the motherboard. The card connector is offset, insuring that it cannot be plugged in backwards. The +5 logic power is supplied through the motherboard. Cable 90014081, the lower front panel cable, attaches to the top edge connector P3. When the full display panel is used, cable 90014091 - the upper front panel cable- attaches to the top edge connector



* PC LAST Written During FETCH and No DEBUG Interrupt
PC NEXT Written in All Other Cases

FIGURE 4-11 SEQUENCER TIMING

P2. Care should be taken that pin 1 of the cable (Red Stripe) and pin 1 of the edge connector align. Pin 1 is to the left when viewed from the component side of the board.

INSTALLATION IN OTHER APPLICATIONS. When using the module in other applications, the user must take into consideration environmental extremes, mounting, power requirements, interface connections and signal AC and DC requirements. Environmental limits, power requirements are card outline dimensions are listed in Figure 4-12.

For mounting, the board is designed to plug into a 100 pin edge card connector, CDC VPB-01C5-0D00A1 or equivalent. Card edge guides should be provided and the card should be mounted vertically or in some way clamped so that it does not vibrate out of its socket. Vertical mounting is also important to insure convective cooling if no fan is used. The card dissipates approximately 10 watts, therefore adequate space should be allowed between modules for air flow. It is desirable to have forced air flow between all modules if several modules are utilized within a confined area.

Interface connections and signal requirements are given in section 4.3.3.

4.3.2 Switch and Jumper Options There are no switch or jumper options on the Debug and F.P. I/O Module.

4.3.3 Interface and Connector Pinout A complete interconnection diagram for Debug and F.P. I/O Module use within the system is given in figure 4-13. Figure 4-14 lists the motherboard connector (P1) pinouts. For a complete description of signal mnemonics refer to chapter 9. Figures 4-15 and 4-16 list P3 and P2 pinouts to the lower and upper front panel cards.

ENVIRONMENTAL

Air Temp 0^o - 70^oC Operational
Surrounding Module

Storage Temp -55^o - 125^oC

Humidity 90% Non Condensing

POWER

	TYP	MAX
+5VDC	1.9A	2.4A

PHYSICAL OUTLINE:

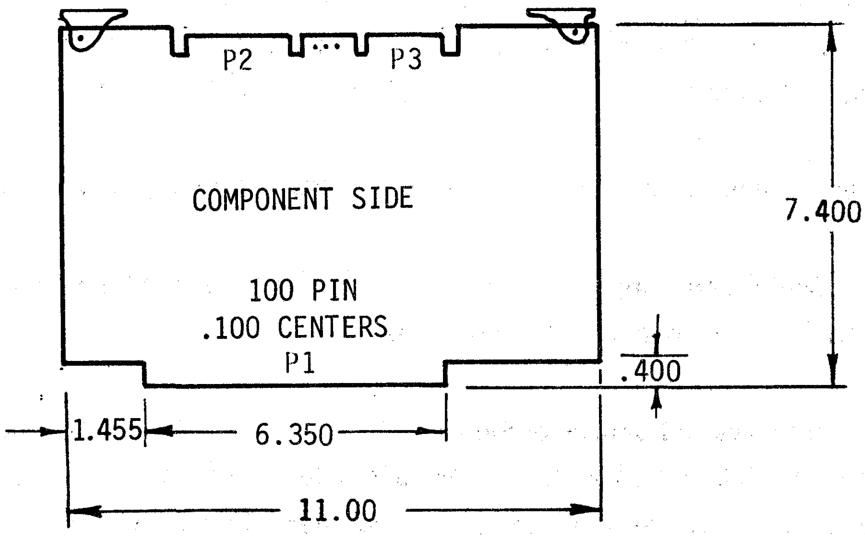


FIGURE 4-12 DEBUG & F.P. I/O MASTER CPU ENVIRONMENTAL
POWER AND PHYSICAL REQUIREMENTS

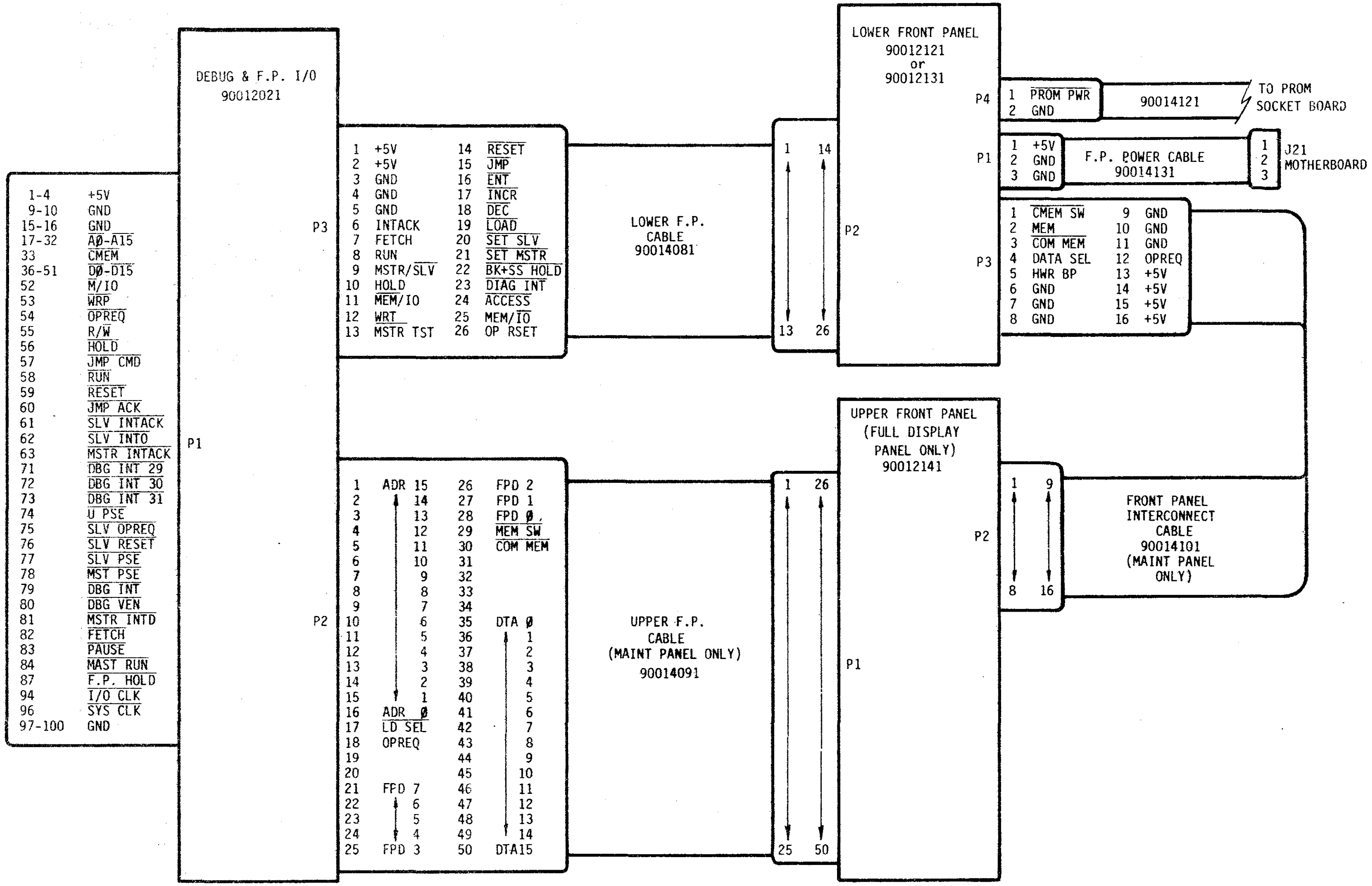


FIGURE 4-13 DEBUG AND FRONT PANEL INTERCONNECT

	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	+5V	+5VDC In	2	+5V	+5VDC In
	3	+5V	+5VDC In	4	+5V	+5VDC In
	5	AUX BUS	Not Used	6	AUX BUS	Not Used
	7	AUX BUS	Not Used	8	AUX BUS	Not Used
	9	GND	Signal & Power GND	10	GND	Signal & Pwr GND
	11	+12V	Not Used	12	+12V	Not Used
	13	-12V	Not Used	14	-12V	Not Used
	15	GND	Signal & Pwr GND	16	GND	Signal & Pwr GND
ADDRESS BUS	17	$\overline{\text{A0}}$	Address Bus In/Out	18	$\overline{\text{A1}}$	Address Bus In/Out
	19	$\overline{\text{A2}}$		20	$\overline{\text{A3}}$	
	21	$\overline{\text{A4}}$		22	$\overline{\text{A5}}$	
	23	$\overline{\text{A6}}$		24	$\overline{\text{A7}}$	
	25	$\overline{\text{A8}}$		26	$\overline{\text{A9}}$	
	27	$\overline{\text{A10}}$		28	$\overline{\text{A11}}$	
	29	$\overline{\text{A12}}$		30	$\overline{\text{A13}}$	
	31	$\overline{\text{A14}}$		32	$\overline{\text{A15}}$	
MEMORY CONTROL	33	CMEM	Slave/MSTR MEM In/Out	34	$\overline{\text{RAM INH}}$	Not Used
	35	$\overline{\text{WD ACCESS}}$	Not Used			
DATA BUS	37	$\overline{\text{D1}}$	Data Bus In/Out	36	$\overline{\text{D0}}$	Data Bus In/Out
	39	$\overline{\text{D3}}$		38	$\overline{\text{D2}}$	
	41	$\overline{\text{D5}}$		40	$\overline{\text{D4}}$	
	43	$\overline{\text{D7}}$		42	$\overline{\text{D6}}$	
	45	$\overline{\text{D9}}$		44	$\overline{\text{D8}}$	
	47	$\overline{\text{D11}}$		46	$\overline{\text{D10}}$	
	49	$\overline{\text{D13}}$		48	$\overline{\text{D12}}$	
	51	$\overline{\text{D15}}$		50	$\overline{\text{D14}}$	
BUS CONTROL	53	$\overline{\text{WRP}}$	Write Pulse In/Out	52	$\overline{\text{M}}$/IO	$\overline{\text{Memory}}$/I/O In/Out
	55	R/$\overline{\text{W}}$	Read/Write In/Out	54	$\overline{\text{OPREQ}}$	Operation Request In/Out
	57	$\overline{\text{JMP CMD}}$	Jump Command Out	56	$\overline{\text{HOLD}}$	Hold CPU In/Out
	59	$\overline{\text{RESET}}$	System Reset In/Out	58	$\overline{\text{RUN}}$	CPU Running In
				60	$\overline{\text{JMP ACK}}$	Jump Acknowledge In
SLAVE INTERRUPT	61	SLV INTACK	Slave Interrupt Ack. In	62	SLV INT 0	Slave Interrupt 0 Out
	63	MSTR INTACK	Master Interrupt Ack. In	64	SLV INT 2	Not Used
	65	SLV INT 3	Not Used	66	SLV INT 4	Not Used
	67	SLV INT 5	Not Used	68	SLV INT 6	Not Used
	69	SLV INT 7	Not Used	70	SLV CPU INT	Not Used
SLAVE DEBUG	71	DBG INT 29	Debug Interrupt 29 In	72	DBG INT 30	Debug Interrupt 30 In
	73	DBG INT 31	Debug Interrupt 31 In	74	U PSE	User Pause In
	75	SLV OPREQ	Slave OPREQ In	76	SLV RESET	Slave Reset Out
	77	SLV PSE	Slave PAUSE Out			
MASTER DEBUG	79	DBG INT	Debug Interrupt Out	78	MST PSE	Master Pause Out
	81	MST INTD	Master Interrupted In	80	DBG VEN	Debug Vector Enable In
BUS CONTROL (Except J1-J2)	83	$\overline{\text{PAUSE}}$	Pause CPU's In	82	$\overline{\text{FETCH}}$	Slave CPU Fetch In
	85	$\overline{\text{SENSE}}$	Not Used	84	$\overline{\text{MAST RUN}}$	Master CPU Running In
	87	$\overline{\text{F.P. HOLD}}$	Front Panel Hold Out	86	$\overline{\text{FLAG}}$	Not Used
	89	$\overline{\text{MST INT 3}}$	Not Used	88	$\overline{\text{MST INTACK}}$	Not Used
	91	SPARE	Not Used	90	SPARE	Not Used
				92	SPARE	Not Used
CLOCKS (Except J1-J2)	93	SLV CLK	Not Used	94	$\overline{\text{I/O CLK}}$	38.4 KHz I/O Clock In
	95	2650 CLK	Not Used	96	$\overline{\text{SYS CLK}}$	System 10 MHz Clock In
GROUND	97	GND	Signal & Pwr GND	98	GND	Signal & Pwr GND
	99	GND	Signal & Pwr GND	100	GND	Signal & Pwr GND

FIGURE 4-14 P1 (MOTHERBOARD) PINOUTS

PIN	SIGNAL	DESCRIPTION	PIN	SIGNAL	DESCRIPTION
1	+5V	Logic Power to Front Panel	2	+5V	Logic Power to Front Panel
3	GND	Ground to Front Panel	4	GND	Ground to Front Panel
5	GND	" " " "	6	INTACK	Interrupt Acknowledge Out
7	FETCH	Fetch Indication Out	8	RUN	CPU Run Indication Out
9	MSTR/SLV	Master/Slave Flip-Flop Out	10	HOLD	CPU Hold Indication Out
11	MEM/IO	Memory/IO Indication Out	12	WRT	Write/Read Indication Out
13	MSTRST	Master CPU Test Switch In	14	RSET	Reset Switch In
15	JMP	Jump Switch In	16	ENT	Enter In
17	INC	Increment In	18	DEC	Decrement In
19	LOAD	Load Switch In	20	SET SLV	Slave Select Switch In
21	SET MSTR	Master Select Switch In	22	BK+SS HOLD	Breakpoint or Single Step Hold In
23	DIAG INT	Diagnostic Interrupt Switch In	24	ACCESS	Access Switch In
25	MEM/IO	Memory/IO Switch In	26	OP RSET	Operation Reset In

FIGURE 4-15 PIN LIST, EDGE CONNECTOR P3

PIN	SIGNAL	DESCRIPTION	PIN	SIGNAL	DESCRIPTION
1	ADR 15	Address Display Bit 15 Out	2	ADR 14	Address Display Bit 14 Out
3	ADR 13	13	4	ADR 12	12
5	ADR 11	11	6	ADR 10	10
7	ADR 9	9	8	ADR 8	8
9	ADR 7	7	10	ADR 6	6
11	ADR 5	5	12	ADR 4	4
13	ADR 3	3	14	ADR 2	2
15	ADR 1	Address Display Bit 1 Out	16	ADR Ø	Address Display Bit Ø Out
17	LD SEL	Load Select to MUX Out	18	OPREQ	Operation Request Out
19	NOT USED		20	NOT USED	
21	FPD 7	Front Panel MUX Data Bit 7 In	22	FPD 6	Front Panel MUX Data Bit 6 In
23	FPD 5	5	24	FPD 4	4
25	FPD 3	3	26	FPD 2	2
27	FPD 1	Front Panel MUX Data Bit 1 In	28	FPD 0	Front Panel MUX Data Bit Ø In
29	MEM SW	Slave Memory Switch In	30	COM MEM	Slave Memory Select Line Out
31	NOT USED		32	NOT USED	
33	NOT USED		34	NOT USED	
35	DTA Ø	Data Display Bit Ø Out	36	DTA 1	Data Display Bit 1 Out
37	DTA 2	2	38	DTA 3	3
39	DTA 4	4	40	DTA 5	5
41	DTA 6	Data Display Bit 6 Out	42	DTA 7	Data Display Bit 7 Out
43	NOT USED		44	NOT USED	
45	NOT USED		46	NOT USED	
47	NOT USED		48	NOT USED	
49	NOT USED		50	NOT USED	

FIGURE 4-16 PIN LIST, EDGE CONNECTOR P2

Chapter 5
RAM/PROM MEMORY MODULE

5.0 GENERAL DESCRIPTION

The RAM/PROM Memory Module provides 4K eight bit words of read-write (RAM) Memory and up to 2K eight bit words of read only (1702A PROM) Memory. The board contains Module address select switches to allow the 4K of RAM to be based at a selected 4K segment of the memory space, and to allow the PROM Memory to be based at a selected 2K segment of the memory space. The PROM and RAM can also be designated independently as Master or Slave Memory. A means for disabling portions of the RAM when PROM is installed in the same memory space is also included. Figure 5-1 is a block diagram of the board.

Communications with the CPU is accomplished through the system data bus, which is driven by a set of drivers and receivers for the RAM array and a set of drivers for the PROM array. The drivers and receivers are enabled only when the board is accessed.

Memory is enabled when the CPU is executing an instruction requiring a read from, or write to memory. During each CPU access, control signals are active to control transfer of data to or from the data bus.

5.1 DETAILED DESCRIPTION

5.1.1 Memory Elements The RAM is composed of 32 type 2102-2 memory devices, each capable of storing 1024 bits. These are organized into a four row by eight bit array, for a total capacity of 4,096 eight bit words. Read and Write access time for the 2102-2 is 650 ns.

Sockets are provided for installation of up to eight 1702A PROMS, each containing 256 eight bit words, for a total capacity of 2048 eight bit words. Access time of the 1702A PROM is 1.0 μs.

The board contains two sets of four switches which allow the base address of the RAM to be placed at any multiple of 4K between Ø and 60K and similarly, allow the base address of the PROM to be placed at any multiple of 2K between 0 and 62K.

5.1.2 PROM Addressing The sixteen address bus bits (AØ-A15) are divided into three groups to address the PROM. The five most significant bits (A11-A16) are applied to the PROM base address comparator. The other inputs to the comparator are the five base address programming switches, the CMEM bus signal, and a CMEM programming jumper. This jumper allows the board to be used for either the Master Memory or the Slave Memory in the computer. The comparator output enables the PROM read circuits. The second group of address bits, A8-A10, are decoded into eight lines to enable one of the eight PROMS. These signals are also applied, via switches, to the PROM read circuits. These switches are used to enable the PROM read circuits only for those PROMS that are actually installed in the board. The third group of address lines, AØ-A7, select one of the 256 bytes of the enabled PROM to be output to the data bus drivers.

5.1.3 RAM Addressing The address lines are also divided into three groups to address the RAM. A12 to A15 are applied to the RAM base address comparator, the output of which enables the RAM read and write circuits. A10 and A11 are decoded to select one of the four "rows" of the RAM, while A0-A9 select the proper bit within the selected row.

The PROM read enable signal is also supplied to the bus as a RAM Inhibit (RAM INH) signal. This signal is used to inhibit the output of the RAM base address comparator. Thus, if PROM is installed in address space where RAM also exists, the RAM will be disabled when the PROM at the same address is being read.

5.1.4 Timing Memory is enabled when the CPU chip is executing an instruction requiring a read or write involving memory. During the CPU

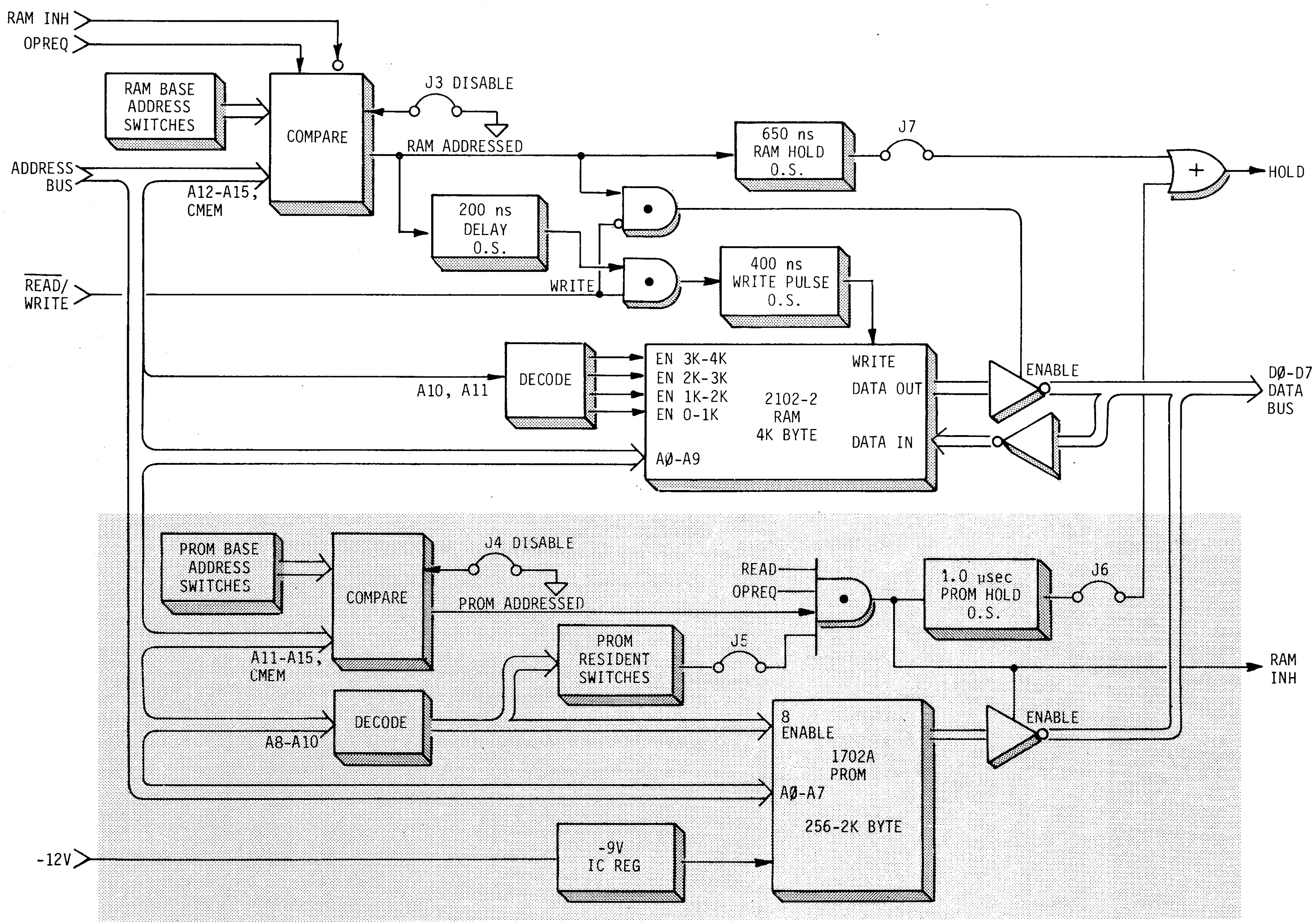


FIGURE 5-1 RAM/PROM MODULE BLOCK DIAGRAM

cycle, control signals become active to control transfer of data to or from memory. OPREQ synchronizes memory and I/O units during data transfers. MEM/IO signifies whether a memory unit or an I/O unit is to be activated and R/W determines whether the requested operation is a read or write.

On the board, MEM/IO and OPREQ are ANDed to generate the basic memory timing signal. When PROM is addressed, the output of PROM read enable circuit is used to enable data bus drivers which place the PROM outputs on the data bus. At the same time, a 1.Ø μs HOLD signal is generated. This provides the delay necessary to insure that valid PROM data is on the data bus before the CPU proceeds.

If the RAM is addressed (and no PROM occupies the same address space), and a read operation is requested, the RAM data output drivers are enabled to place the RAM data on the data bus. If the operation is a write, a 400 ns write pulse is generated and applied to the RAM write circuits to write the data bus contents into the RAM. This write pulse occurs after a 200 ns delay as required for proper setup time of the RAM chips. In either case, a 650 ns HOLD signal is generated to HOLD the CPU until the operation is completed.

5.1.5 -9V Regulator The -12V DC input is applied to a 3 terminal regulator to produce the -9V DC power required for operation of the PROMS.

5.2 UTILIZATION

5.2.1 Installation The module is designed to be plugged into the card file of the development computer, but may be used in other applications. Jumper and switch options are discussed in section 5.2.2. All interface connections and connector pinouts are detailed in section 5.2.3.

COMPUTER INSTALLATION The RAM/PROM Memory Module may be plugged into any available slot of the motherboard (J3-J8, J10-J20). The card connector is offset,ensuring that it cannot be plugged in backwards. All power requirements (+5V, -12V) are supplied through the motherboard.

INSTALLATION IN OTHER APPLICATIONS When using the module in other applications, the user must take into consideration environmental extremes, mounting, power requirements, interface connections and signal AC and DC requirements. Environmental limits, power requirements and card outline dimensions are listed in figure 5-2.

For mounting, the board is designed to plug into a 100 pin edge card connector, CDC VPB-01C5-0D00A1 or equivalent. Card guide slots should be provided and the card should be mounted vertically or in some way clamped so that it does not vibrate out of its socket. Vertical mounting is also important to insure convective cooling if no fan is used. The card dissipates approximately 7 watts, therefore adequate space should be allowed between modules for air flow. It is desirable to have forced air flow between all modules if several modules are utilized within a confined area.

Interface connections and signal requirements are given in section 5.2.3.

5.2.2 Switch and Jumper Options The RAM/PROM Memory Board contains several switches and jumpers which provide user control over the operation of the board.

RAM Base Address Selection -- Switch E8A is used to set the base address of the 4K RAM. This may be set at any 4K multiple between Ø and 60K. The switch weights and sample settups for a base address of 20,480 are as follows:

Switch	Weight	Setting for Base Address of 20,480 (4096 x 5)
E8A-1	4,086	ON
E8A-2	8,192	OFF
E8A-3	16,384	ON
E8A-4	32,768	OFF

PROM Base Address Selection -- E8B and jumper J1 select the base address of the 2K PROM as any 2K multiple between Ø and 62K. The weights and sample settings for a base address of 12,288 are as follows:

Switch or Jumper	Weight	Setting for Base Address of 12,288 (2048 x 6)
E8B-1	2,048	OFF
E8B-2	4,096	ON
E8B-3	8,192	ON
E8B-4	16,384	OFF
J1	32,768	OUT

Resident PROM Selection -- An 8 position switch, D14, is used to indicate which PROM chips have been plugged in. Jumper J5 can be left open to enable all 8 PROMS or jumper J4 can be added to disable all PROMS. The switch positions, the PROM location, and the corresponding address spaces are as follows, where BA refers to the PROM base address:

Switch	PROM Location	Address Range
D14-1	A10	BA - BA + 255
D14-2	B10	BA + 256 - BA + 511
D14-3	A12	BA + 512 - BA + 767
D14-4	B12	BA + 768 - BA + 102
D14-5	A13	BA + 1024 - BA + 1279
D14-6	B13	BA + 1280 - BA + 1535
D14-7	A15	BA + 1536 - BA + 1791
D14-8	B15	BA + 1792 - BA + 2047

Other Jumpers -- J2 identifies the baord as Master or Slave Memory and determines whether the RAM is selected when the CMEM bus signal is low or when CMEM bus signal is low or when it is high. J8 performs the same function for the PROM area. J3 can be used to disable the RAM completely and J4 will disable the PROM completely. J6 determines if the one microsecond PROM hold signal is applied to the HOLD bus line, and J7 does the same for the 650 μs RAM HOLD signal.

Below is a complete listing of all jumpers:

J1	IN	=	PROM A15
J2	IN	=	RAM Slave Memory
J3	IN	=	Disable RAM
J4	IN	=	Disable PROM
J5	OUT	=	Enable all PROM
J6	IN	=	Enable 1 μs PROM HOLD
J7	IN	=	Enable 650 ns RAM HOLD
J8	IN	=	PROM Slave Memory

5.2.3 Interface and Connector Pinouts Figure 5-3 lists the motherboard connector (P1) pinouts and signal definitions.

ENVIRONMENTAL

Air Temp 0^o - 70^oC Operational
Surrounding Module

Storage Temp -55^o - 125^oC

Humidity 90% Non-condensing

POWER

	TYP	MAX
+5VDC	1.2A	1.8
-12VDC	.32A	.52

PHYSICAL OUTLINE

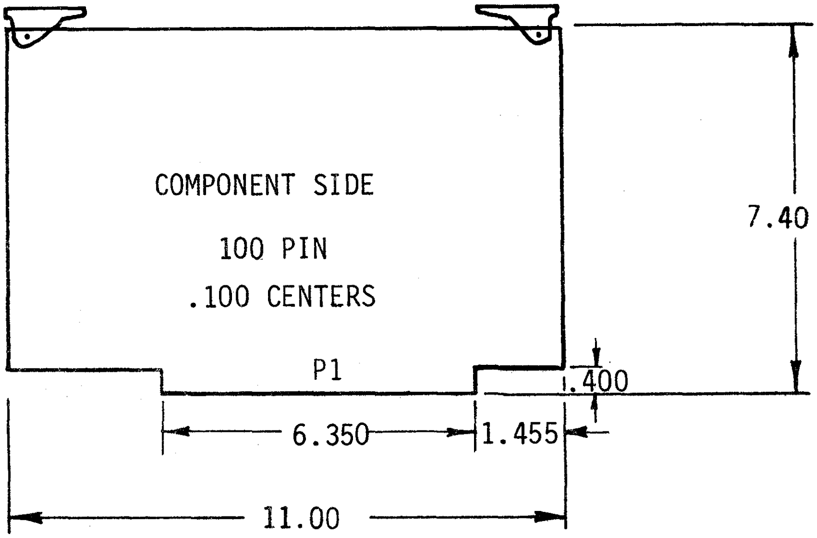


FIGURE 5-2 RAM PROM MEMORY ENVIRONMENTAL, POWER AND PHYSICAL REQUIREMENTS

	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	+5V	+5VDC In	2	+5V	+5VDC In
	3	+5V	+5VDC In	4	+5V	+5VDC In
	5	AUX BUS	Not Used	6	AUX BUS	Not Used
	7	AUX BUS	Not Used	8	AUX BUS	Not Used
	9	GND	Signal & Pwr GND	10	GND	Signal & Pwr GND
	11	+12V	Not Used	12	+12V	Not Used
	13	-12V	-12VDC In	14	-12VDC	-12VDC In
	15	GND	Signal & Pwr GND	16	GND	Signal & Pwr GND
ADDRESS BUS	17	$\overline{\text{A0}}$	Address bus In	18	$\overline{\text{A1}}$	Address Bus In
	19	$\overline{\text{A2}}$		20	$\overline{\text{A3}}$	
	21	$\overline{\text{A4}}$		22	$\overline{\text{A5}}$	
	23	$\overline{\text{A6}}$		24	$\overline{\text{A7}}$	
	25	$\overline{\text{A8}}$		26	$\overline{\text{A9}}$	
	27	$\overline{\text{A10}}$		28	$\overline{\text{A11}}$	
	29	$\overline{\text{A12}}$		30	$\overline{\text{A13}}$	
	31	$\overline{\text{A14}}$		32	$\overline{\text{A15}}$	
MEMORY CONTROL	33	$\overline{\text{CMEM}}$	Slave/$\overline{\text{MSTR}}$ MEM In	34	$\overline{\text{RAM INH}}$	RAM Memory Inhibit In/Out
	35	$\overline{\text{WD ACCESS}}$	Not Used			Not Used
DATA BUS	37	$\overline{\text{D1}}$	Data Bus In/Out	36	$\overline{\text{D0}}$	Data Bus In/Out
	39	$\overline{\text{D3}}$		38	$\overline{\text{D2}}$	
	41	$\overline{\text{D5}}$		40	$\overline{\text{D4}}$	
	43	$\overline{\text{D7}}$		42	$\overline{\text{D6}}$	
	45	$\overline{\text{D9}}$	Not Used	44	$\overline{\text{D8}}$	Not Used
	47	$\overline{\text{D11}}$		46	$\overline{\text{D10}}$	
	49	$\overline{\text{D13}}$		48	$\overline{\text{D12}}$	
	51	$\overline{\text{D15}}$		50	$\overline{\text{D14}}$	
BUS CONTROL	53	$\overline{\text{WRP}}$	Write Pulse In	52	M/IO	Memory/I/O In
	55	R/$\overline{\text{W}}$	Read/$\overline{\text{Write}}$ In	54	$\overline{\text{OPREQ}}$	Operation Request In
	57	JMP CMD	Not Used	56	$\overline{\text{HOLD}}$	Hold CPU Out
	59	RESET	Not Used	58	$\overline{\text{RUN}}$	Not Used
				60	$\overline{\text{JMP ACK}}$	Not Used
SPLIT BUS	61		Not Used	62		Not Used
	63			64		
	65			66		
	67			68		
	69			70		
	71			72		
	73			74		
	75			76		
	77			78		
	79			80		
	81					
BUS CONTROL (Except J1-J2)	83	$\overline{\text{PAUSE}}$	Not Used	82	$\overline{\text{FETCH}}$	Not Used
	85	$\overline{\text{SENSE}}$		84	$\overline{\text{MAST RUN}}$	
	87	$\overline{\text{F.P. HOLD}}$		86	$\overline{\text{FLAG}}$	
	89	$\overline{\text{MST INT 3}}$		88	$\overline{\text{MST INTACK}}$	
	91	SPARE		90	SPARE	
				92	SPARE	
CLOCKS (Except J1-J2)	93	SLV CLK	Not Used	94	I/O CLK	Not Used
	95	2650 CLK		96	SYS CLK	
	97	GND	Signal & Pwr GND	98	GND	Signal & Pwr GND
	99	GND	Signal & Pwr GND	100	GND	Signal & Pwr GND

FIGURE 5-3 RAM/PROM MODULE MOTHERBOARD (P1) PINOUTS

Chapter 6

16 K DYNAMIC RAM MODULE

6.0 INTRODUCTION

The 16K Dynamic RAM Module provides 16,384 (16K) x 8 bit words of dynamic random access memory (read/write). Up to four modules can be used in the development computer for Slave Memory providing a total of 65,536 bytes of read/write memory. In the Universal One system, the module is also used for the 16K Master Memory card 256 byte Boot PROM.

The module has full parity generation, storage, and checking for each 8 bit byte. Separate interrupts are generated for Master or Slave Memory if a parity error is detected.

A pair of modules provides 16 bit read/write storage for 16 bit word oriented Slaves. The same modules will respond to a byte oriented Slave or to the Master CPU in an 8 bit ping-pong fashion.

The RAM module completes a read or write cycle in 400 ns. Refresh is transparent to most Slave processors since it is interleaved between bus transactions.

6.1 GENERAL DESCRIPTION

The Dynamic RAM Module can be divided into eight functional units:

1) RAM Memory
2) PROM Memory
3) Module Selection
4) Byte/Word Control
5) Address Control
6) Data and Parity Control
7) Refresh Request Logic
8) Operation Control and Timing

A block diagram of the Module, illustrating the eight functional units is shown in figure 6-1.

The RAM Memory block consists of thirty-six 4K x 1 dynamic random access Memory IC's. The Memory is organized into 4 banks of 9 IC's providing a total storage of 16K 9 bit words. The ninth bit is used exclusively for parity.

The PROM Memory, installed only on the Master Memory module, consists of a single 1702A, 256 x 8 PROM which provides the system Boot. The Boot PROM has separate address lines directly from the bus, but shares the output data bus with the RAM.

The Module select block compares the most significant address bits and the Slave Master Memory line to programmable switch settings to determine whether the module has been selected. It also determines if locations 0-256 on the module are addressed to activate the PROM rather than the RAM Memory (Master Memory only).

The Byte/Word Control allows the memory to be used in a byte oriented fashion only or to be used as the lower or upper byte of a 16 bit word. It also allows the memory, however configured, to be accessed by the byte oriented Master CPU or a byte oriented Slave. A single jumper identifies the module as a byte or word memory, while a slide switch identifies it as the upper or lower byte. For word operation, modules must be used in pairs with one module providing the low byte and the other providing the high byte of the 16 bit word. When a byte oriented processor accesses the memories, its address is shifted, with the least significant address bit providing the low - high byte selection. Data, whether low or high byte, is routed to the lower eight bits. This allows all memory installed to be accessed by any Slave or Master processor with no change in jumper or switch settings. This provides the communication between the 8 bit Master and a 16 bit Slave for I/O service requests and debug operation.

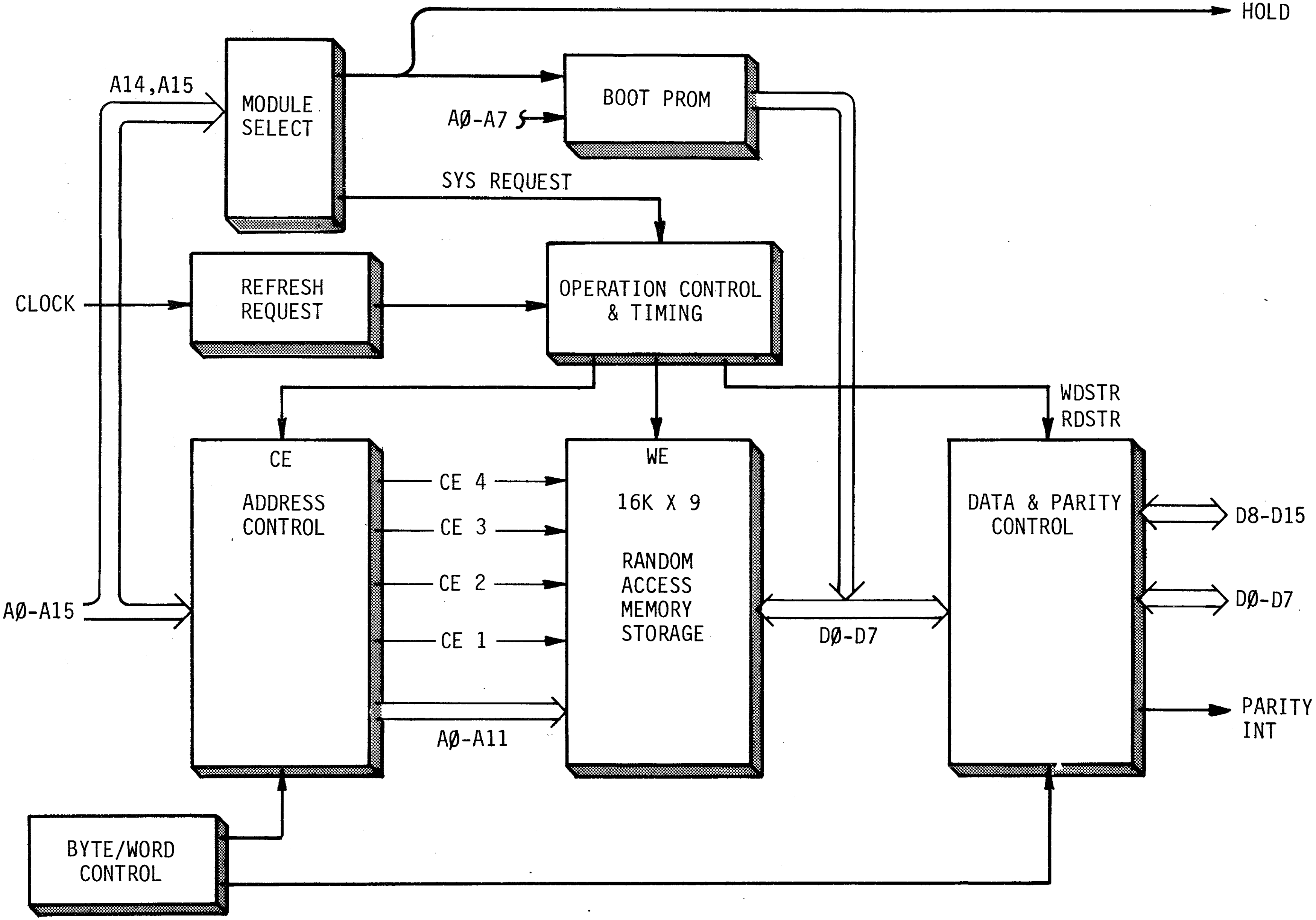


FIGURE 6-1 16K DYNAMIC RAM SIMPLIFIED BLOCK DIAGRAM

The Address Control section consists of a multiplexer which selects the normal address, a shifted address, or the refresh address. During byte access of a word jumpered memory, the selected address is shifted one bit from the bus address. Thus A15 becomes A14, A14 becomes A13, etc. A0 is compared to the high - low byte switch to determine whether the module is enabled. During front panel access, the address is not shifted and A15 is used for high-low byte selection. During refresh a five bit refresh counter is selected as A0-A5 while A6-A11 are held low.

The Data and Parity section contains bus driver-receivers, data registers and parity generation and checking circuitry. During a write operation, data is received from D0-D7 or D8-D15 and stored in an eight bit register. A parity generator IC provides an even parity output for the ninth bit based on the register contents. During read operations, the data from the memory is stored in an eight bit register which feeds bus drivers to D0-D7 or D8-D15. At the same time a parity checker looks at all nine bits for even parity. If parity is odd, indicating an error, a parity flip flop is clocked to the active state. This sends an interrupt to Master interrupt level 1 for a Master Memory card or to Master interrupt level 3 for a Slave Memory card.

The Refresh Request logic controls the basic refresh rate and provides timing for refresh requests such that they are invisible to most processors using the memory. Refresh is triggered by a system clock every 26 μs. On the completion of the next OPREQ, the refresh request is sent to the operation control logic. If no OPREQ occurs within 20 μs the refresh request is unconditionally sent. This effectively interleaves the refresh between processor bus transactions.

The Operation Control and Timing section resolves contention between system and refresh requests and generates all timing for READ, WRITE and REFRESH operations. The basic timing cycle is controlled by a tapped 200 ns delay line which provides the basic 400 ns memory cycle time. The chip enable pulse, write enable pulse and strobes for the read data register,

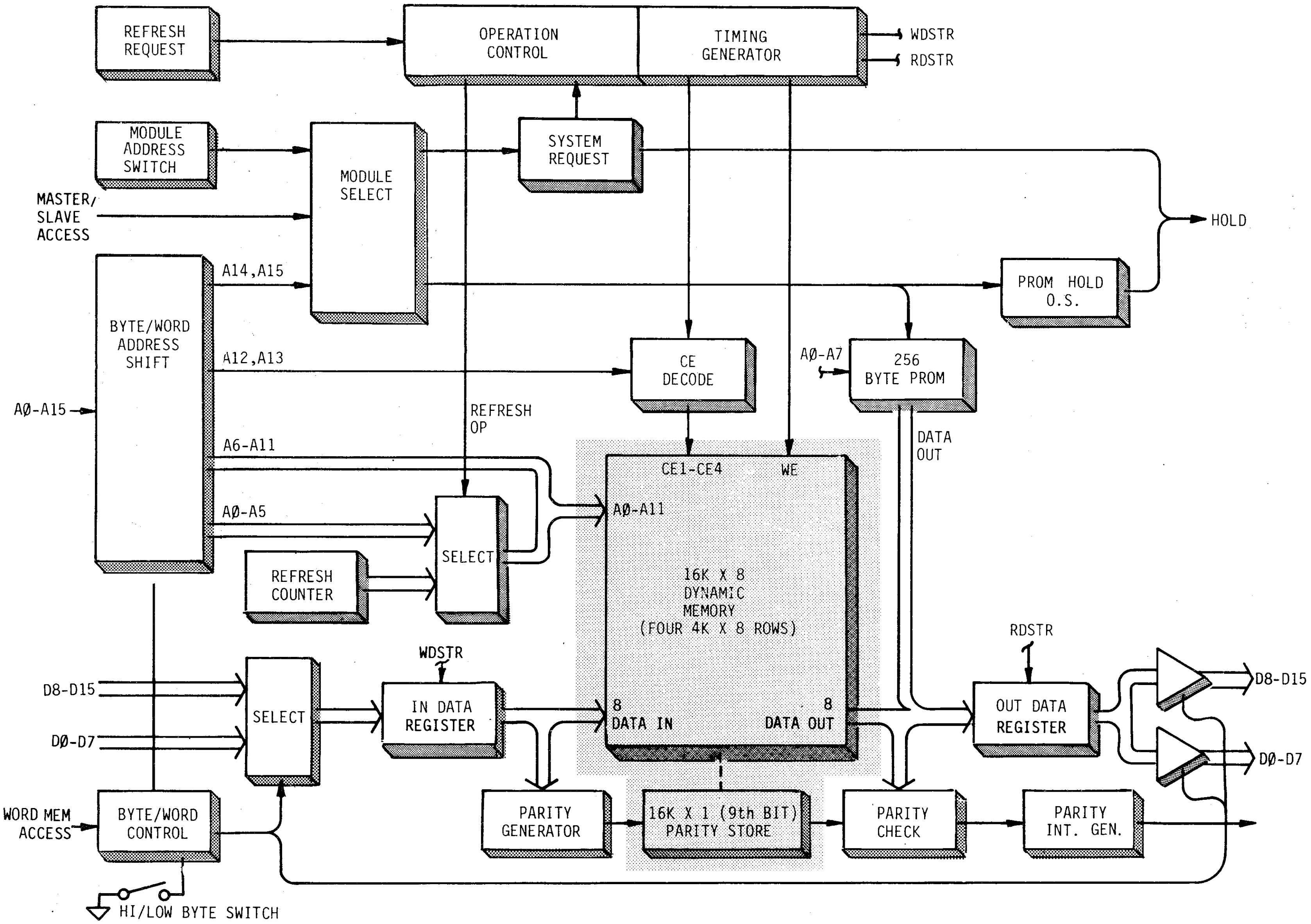


FIGURE 6-2 RAM STORAGE

write data register as well as the parity flip-flop are generated. In addition, the hold signal is reset at 120 ns into a Write cycle and 260 ns into a READ cycle allowing the processor to disengage the memory at the earliest possible time.

6.2 DETAILED DESCRIPTION

6.2.1 RAM Storage The RAM storage shown in the shaded area of figure 6-2 consists of 36 Signetics 2680 or National 5280 22 pin dynamic random access memory chips. Each chip has a 4096 by 1 bit storage capacity. The chips are organized in four banks of nine chips or bits each providing eight data bits plus parity for 16,536 locations. Each chip has 12 address inputs (A0-A11), a chip enable (CE), inverted write enable (WE), inverted chip select (CS), an inverted data in (DI) and a data output (DO). Power inputs include +12, +5, -5 and ground.

The chips are internally organized as a 64 row by 64 column matrix with A0-A5 identifying the row address. Each memory chip requires refresh of all row addresses every 2 ns or one row address every 31.25 µs. The chip has a read access time of 220 ns and a cycle time for both read and write of 400 ns.

6.2.2 PROM Memory A socket and interface logic is included on all memory cards for a single 1702A PROM shown in the shaded area of 6-3. The purpose of the PROM is to provide the system Boot routine in Master Memory, executed upon power-on or reset. The memory is fixed addressed at the first 256 locations of the card and when enabled, overrides the RAM at that location. The Slave/Master Memory switch automatically enables the PROM operation when in the Master Memory position. Two jumpers, J2 and J4, are provided to permanently enable or disable this function regardless of switch position.

The PROM addresses A0-A7 are buffered directly from the bus so that refresh can run concurrently with a PROM access. A PROM operation is initiated

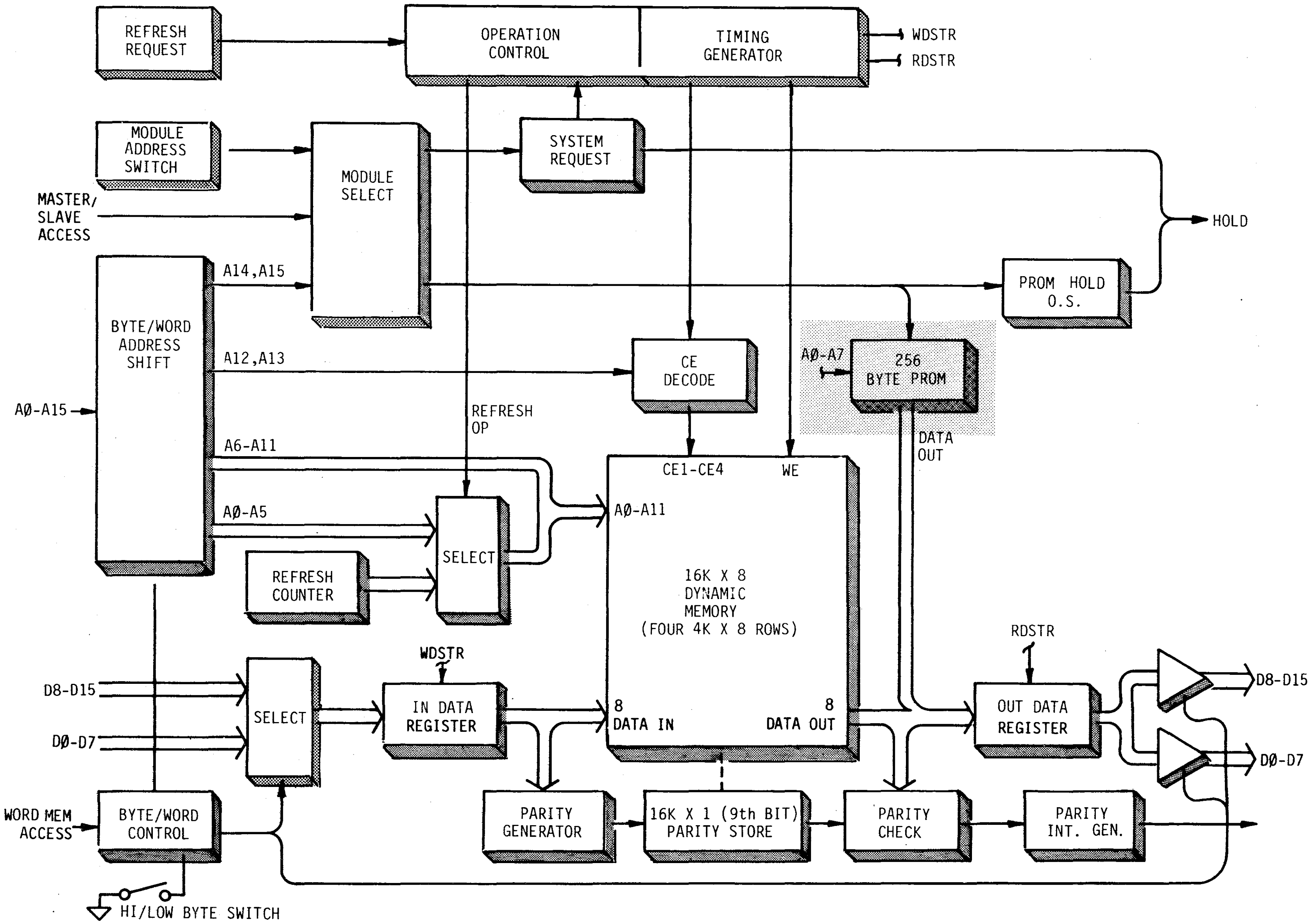


FIGURE 6-3 PROM MEMORY

on the leading edge of OPREQ if the module is selected and the address range is 0-255.

OPREQ triggers the one μs PROM ACCESS one shot which enables the PROM, provides the hold signal to the active processor, and provides a strobe on the trailing edge for output data. The PROM data out shares the RAM data output path which includes the data register and the bus drivers. The parity checker is not activated during a PROM read.

6.2.3 Module Select and System Request Logic The module select and system request logic is shown in the shaded portion of figure 6-4. Module identification is provided by a four position slide switch. Selection includes A15, A14, Master-Slave Memory and low or high byte (if WORD memory). Placing the Master-Slave switch in the Master position automatically enables the Boot PROM circuitry.

For module selection of a byte oriented memory, the transaction must be a memory access with the RAM inhibit line inactive and the address and common memory lines must compare to the switch settings. For a word oriented memory (J9 IN) additional conditions must be met. Either the word access line must be active, or during a byte access the hi-low byte switch must compare with the address bit specifying the low byte of memory.

The module select signal is routed as an enable to either to the PROM ACCESS one shot or to the SYSTEM PENDING flip flop depending on the state of the PROM address decode. On the leading edge of OPREQ, the enabled function is triggered.

If the SYSTEM PENDING flip flop is triggered a read request is allowed to set the SYSTEM REQUEST flip flop immediately. If a write request occurs, the SYSTEM REQUEST flip flop is held from setting until the leading edge of the Write Pulse (WRP). The System Request directly feeds the operation control logic discussed in outline 6.2.8.

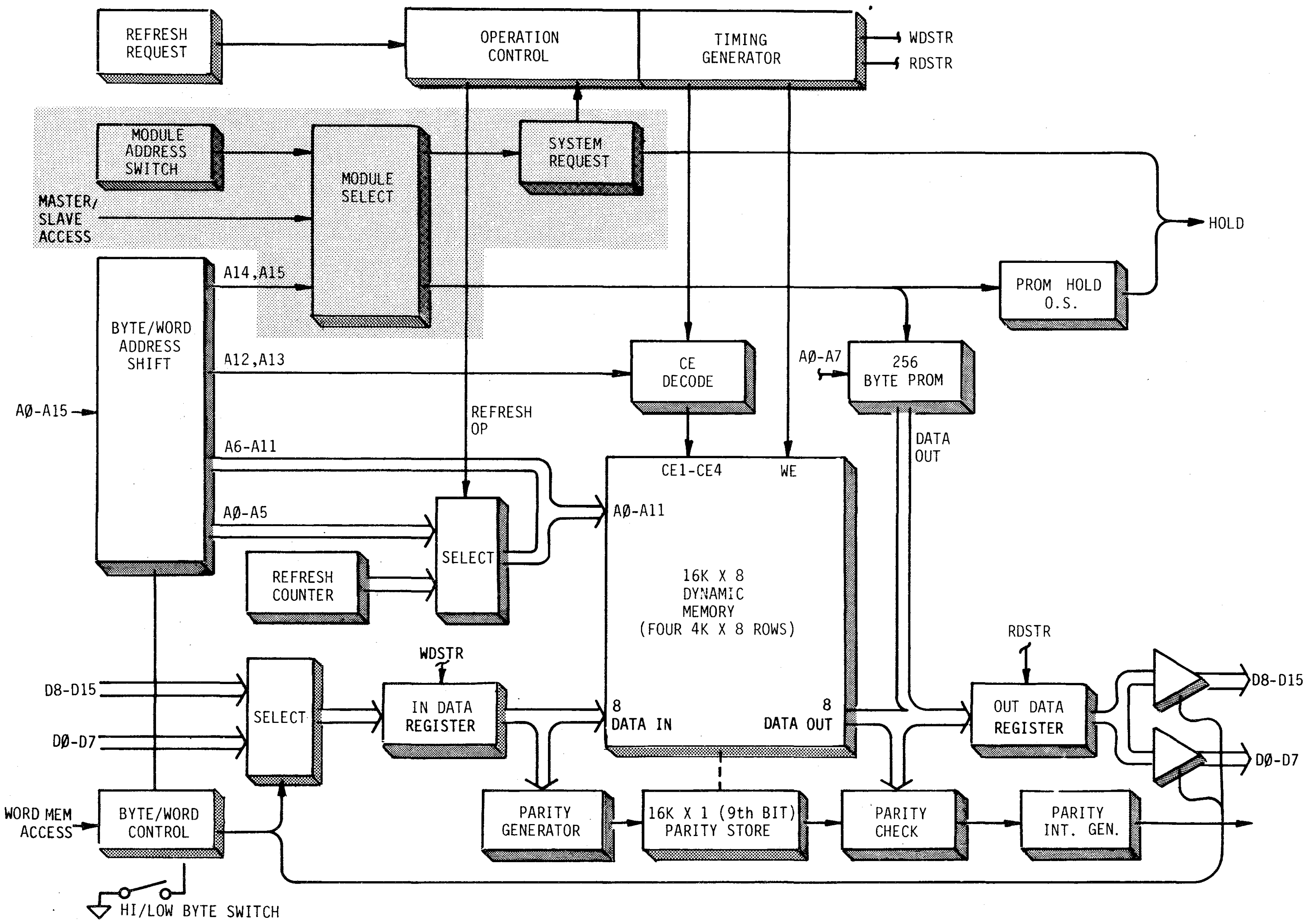


FIGURE 6-4 MODULE SELECT AND SYSTEM REQUEST LOGIC

6.2.4 Byte/Word Control The byte-word control shown in the shaded portion of figure 6-5 is activated only if the memory is designated as a word memory by installing J9 and deleting J10. Memory Cards must be installed in pairs to provide a 16 bit word. During access by a word oriented processor, signified by the WD ACCESS line, both memories of the pair are enabled onto the bus, one providing the low byte, the other the high byte. During access by a byte oriented processor only one memory of the pair is enabled at a time and both memories access only the low data byte. The addresses are shifted on the memory card leaving A0 to enable the high or low byte card. Figure 6-6 shows the address relationship between byte and word addressing. A special operation has been designed for Front Panel access to memory. The Front Panel is byte oriented in that it has only eight data switches for writing data. It is undesirable to have to shift the address byte and use A0 for low-high byte selection since the address would not correspond to program listing in HEX. Therefore, A15 is used to specify the low or high byte of the word and the address is unshifted.

The byte/word control affects module selection, data paths and address shifting. As previously discussed under module selection, if a work memory is accessed by a byte processor, the high-low byte switch must match A0 for a processor access or A15 for a Front Panel access. Data paths, to and from the low byte of the bus, are used unless the card is specified as a high byte word memory and a word access is in process.

Address shifting occurs only for a word memory during a byte processor access. No shift occurs for a Front Panel byte access, but A15 is substituted for A0 in the high-low byte comparison.

6.2.5 Address Control The address control shown in the shaded portion of figure 6-7 performs the byte/word shifting, the refresh address control and the chip enable decoding.

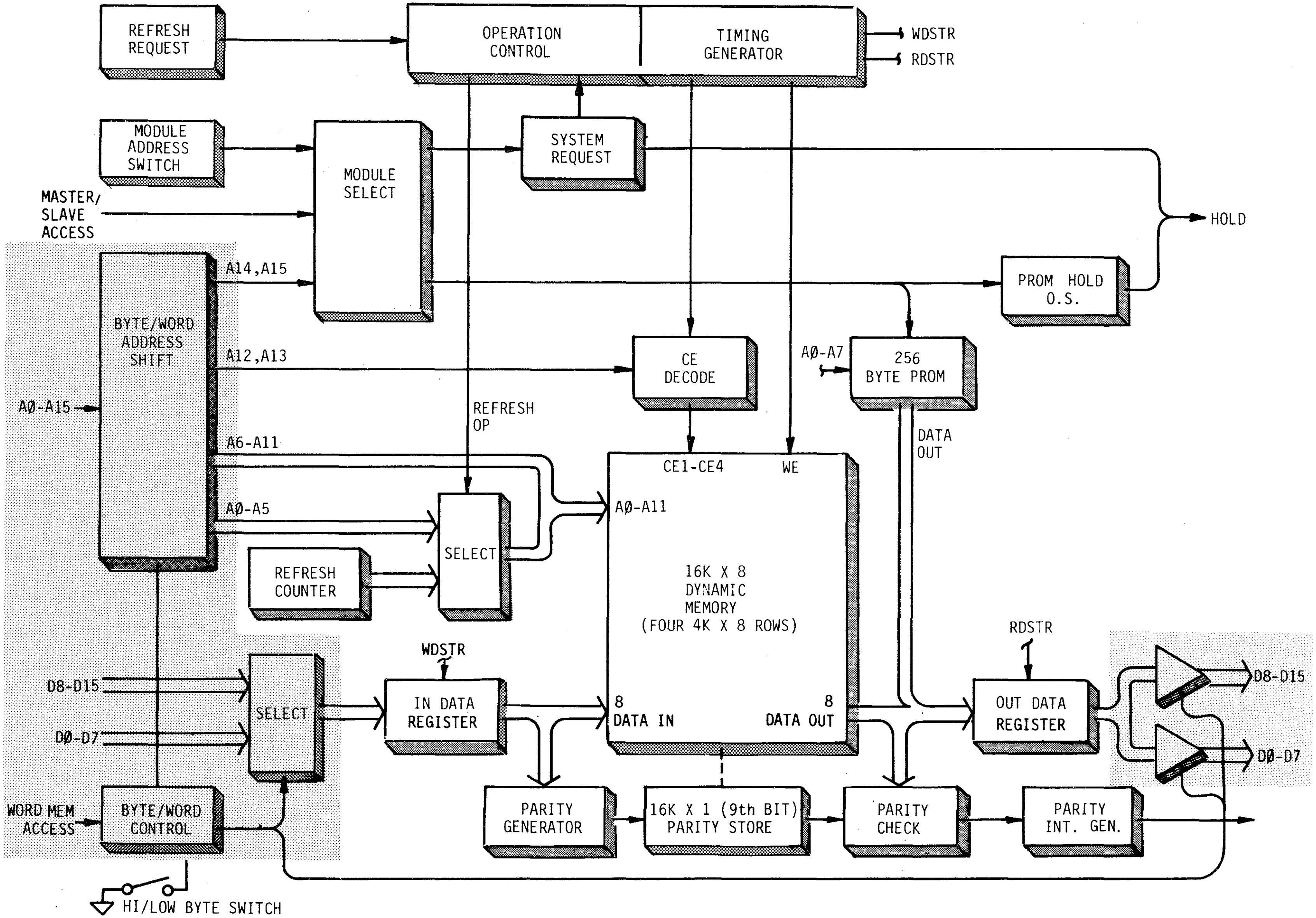


FIGURE 6-5 BYTE WORD CONTROL

ADDRESS		DATA	
WORD PROC	BYTE PROC	0 7	8(0) 15(7)
0000	0000,0001		
0001	0002,0003		
0002	0004,0005		
0003	0006,0007		
0004	0008,0009		
0005	000A,000B		
0006	000C,000D		
0007	000E,000F		

FIGURE 6-6 BYTE AND WORD PROCESSOR ADDRESSING

A6 through A15 are shifted or fed straight through four quadruple 2 to 1 multiplexers. During refresh the outputs of the multiplexers are inhibited for A6 through A13 providing a stable address to the RAMS.

A0-A6 to the RAMS are selected by dual four input multiplexers which select the normal address, the shifted address or a 6 bit refresh counter. The refresh counter is incremented after each refresh cycle, and counts continuously, rolling over after a 63 count.

A0 through A11 out of the multiplexers are fed straight to the RAM chips. Note that the addresses are complemented, saving extra inverter chips. 51 ohm damping resistors are used to critically damp the address lines so no undershoot occurs. A12 and A13 provide inputs to the chip enable decoder

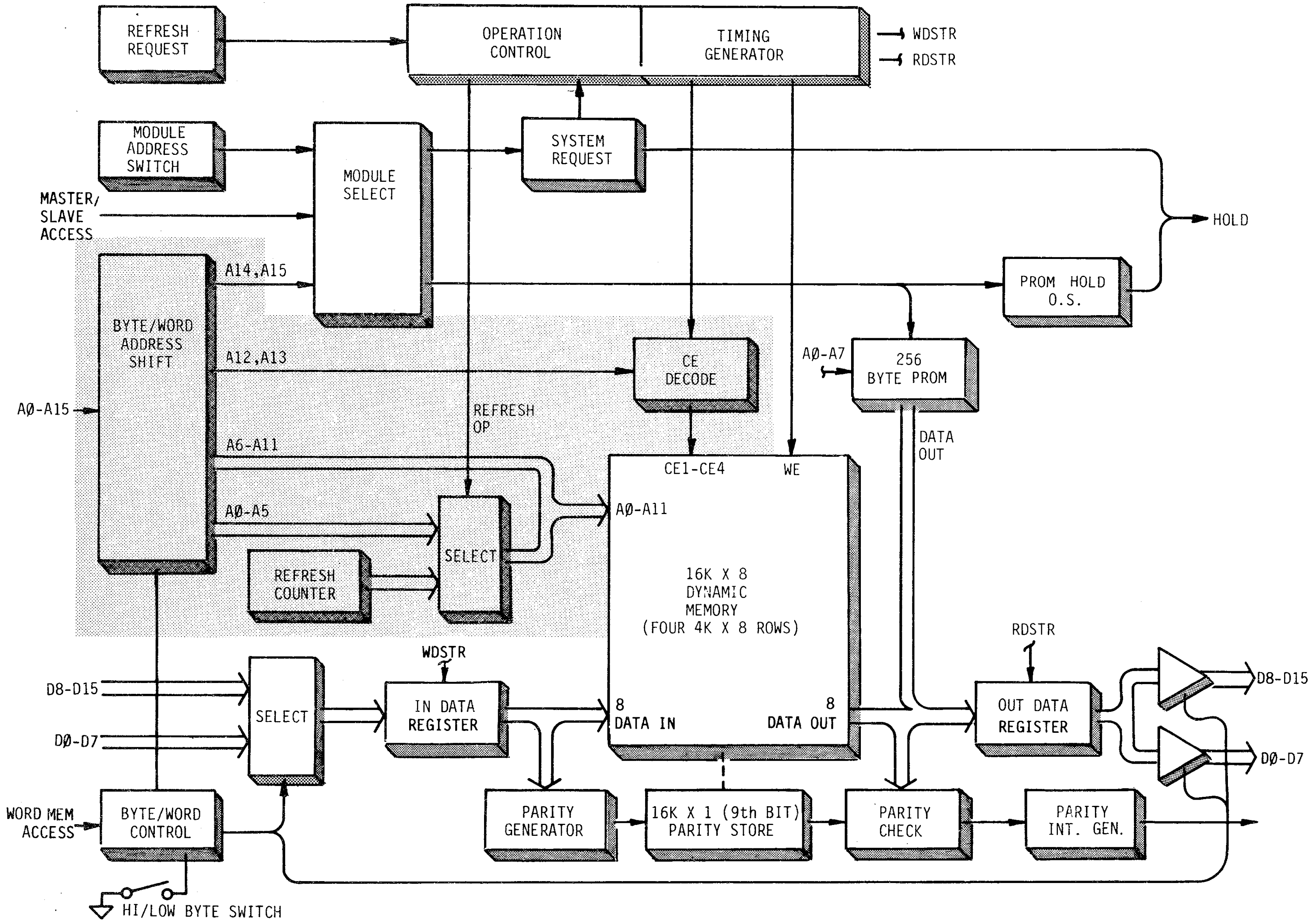


FIGURE 6-7 ADDRESS CONTROL

which selects one of the four 4K x 9 rows during system access. The decoder feeds a quadruple chip enable (CE) driver which provides the 0 to 12V swing required. 280 OHM damping resistors are used to eliminate ringing. The CE has a driver enable line which gates the CE1-CE4 lines with the CE signal from the timing section. A refresh input is provided which activates all four chip enable lines during refresh regardless of the decoder inputs.

A14 and A15 provide inputs to the module select comparators. Note that if a shift is performed A15 is no longer valid, thus for word memories, J10 removes A15 from the module comparison. The multiplexers also supply the high-low byte control line. If a shift occurs, A0 is selected, if not (during F.P. access) A15 is selected.

6.2.6 Data-Parity Control Data buffering to and from the bus is handled by 8T26 transceivers. On the card, separate input data and output data paths are maintained as shown in figure 6-8. For a write operation, data is received through the transceivers from either the high or low data byte. Unless the module is a high byte word memory and is accessed by a word oriented processor, data is received from the low byte. It is clocked into dual four bit registers with the write data strobe (WDSTR). The inverted outputs are used to provide the necessary inverted data to the memory. The register outputs feed the memory matrix with each bit connected to four chips, one for each row. The register also feeds a parity generator chip which provides an even parity bit to the ninth chip of each row.

For a read operation, output from the RAMS is enabled to two HEX registers of which only four bits are used. The registers are clocked with the read data strobe (RDSTR) and feed the 8T26 drivers to both the low and high byte of the data bus. The same control function as described above for the write operation, determines which set of 8T26's is enabled. A parity checker receives the eight RAM data outputs and the parity bit and checks that even parity is present. If parity is odd, indicating an error, the PARITY INTERRUPT flip flop is clocked to a 1. Note that the parity is checked

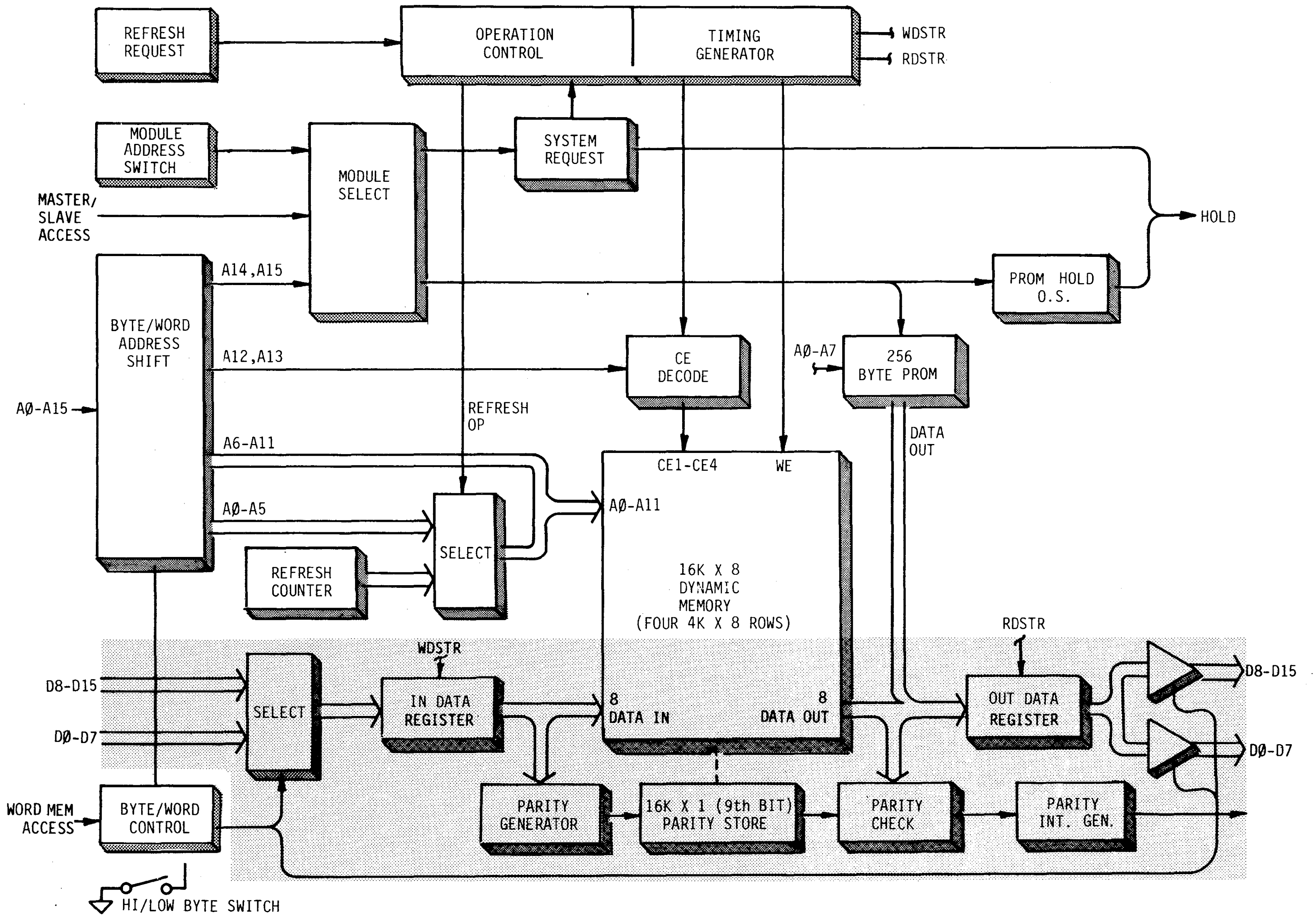


FIGURE 6-8 DATA AND PARITY CONTROLFIGURE

for a RAM read only. The parity interrupt is channeled to Master interrupt level 1 for a Master Memory card or Master interrupt level 3 for a Slave Memory card. Reset of the interrupt is accomplished by decoding the appropriate vector (address 2 or 6) on the data bus during a Master interrupt acknowledgement (INTACK).

During a PROM read, the PROM data out utilizes the RAM data registers and is clocked with the trailing edge of the PROM ACCESS one shot. Since the RAM Chip Select (CE) is off during refresh, refresh may occur concurrently with a PROM access.

6.2.7 Refresh Request Logic The purpose of the refresh request logic, shown in the shaded portion of figure 6-9, is to provide a periodic request (every 26 μs) to the operation control logic that will appear transparent to the processor accessing the memory. The timing is shown in figure 6-10a.

The system I/O clock provides a 100 ns pulse every 26 μs. The leading edge triggers a 20 μs one shot and the trailing edge triggers the REFRESH PENDING flip flop. The REFRESH PENDING flip flop allows the REFRESH REQUEST flip flop to go active on the next trailing edge of OPREQ. This effectively interleaves the refresh between processor accesses. If no OPREQ occurs for the duration of the 20 μs one shot, the REFRESH REQUEST flip flop is automatically set. When the Operation Control starts a refresh operation the REFRESH PENDING flip flop is reset. Near the end of the refresh cycle, the REFRESH REQUEST flip flop is reset.

If the module is used in another application where no 25 μs clock is available, the refresh request logic may be reconfigured to run on its own. J5 and J8 are cut while J6 and J7 are added. The 20 μs one shot thus retriggers itself and refresh is requested unconditionally every 20 μs. The timing is shown in figure 6-10b.

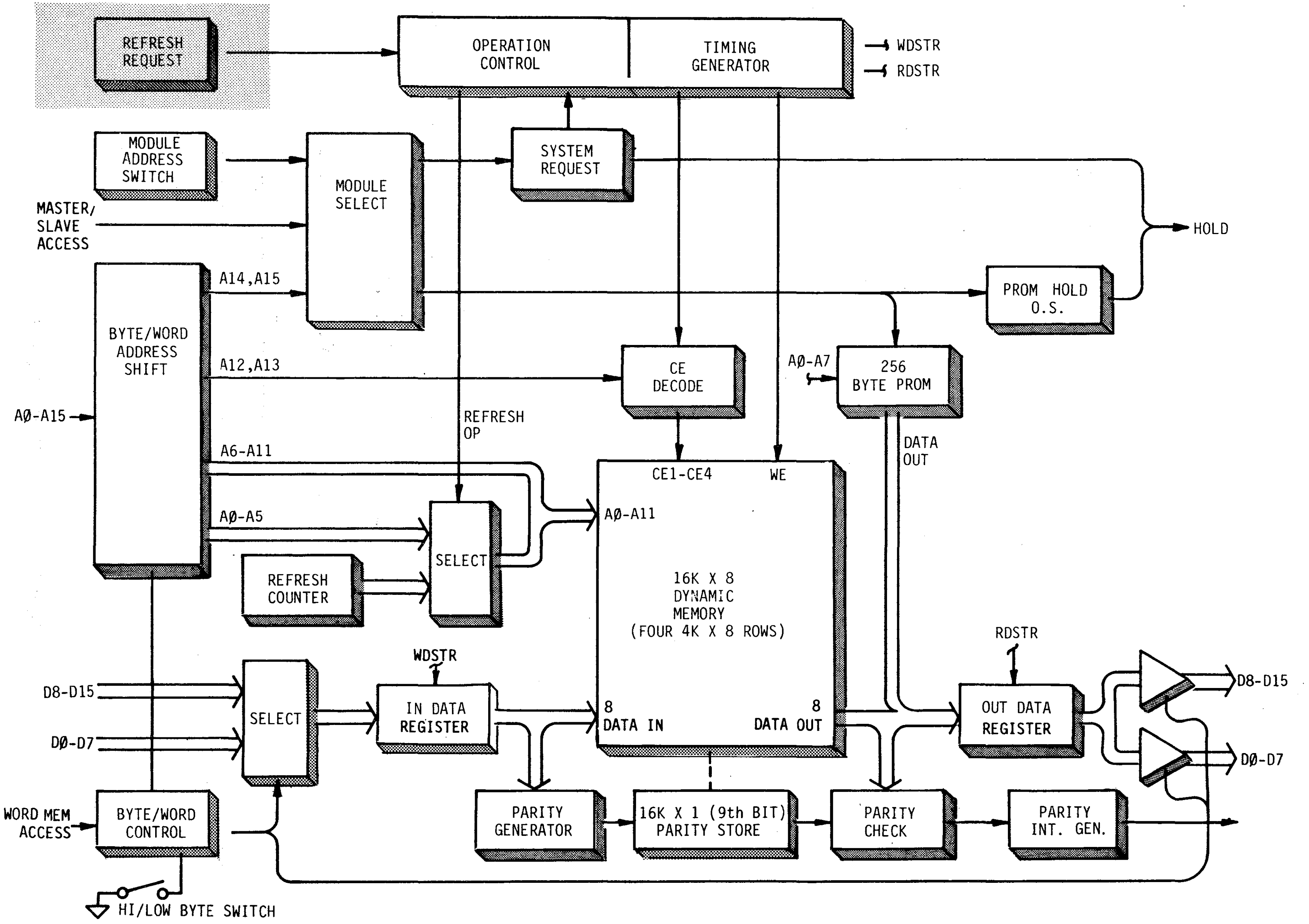


FIGURE 6-9 REFRESH REQUEST

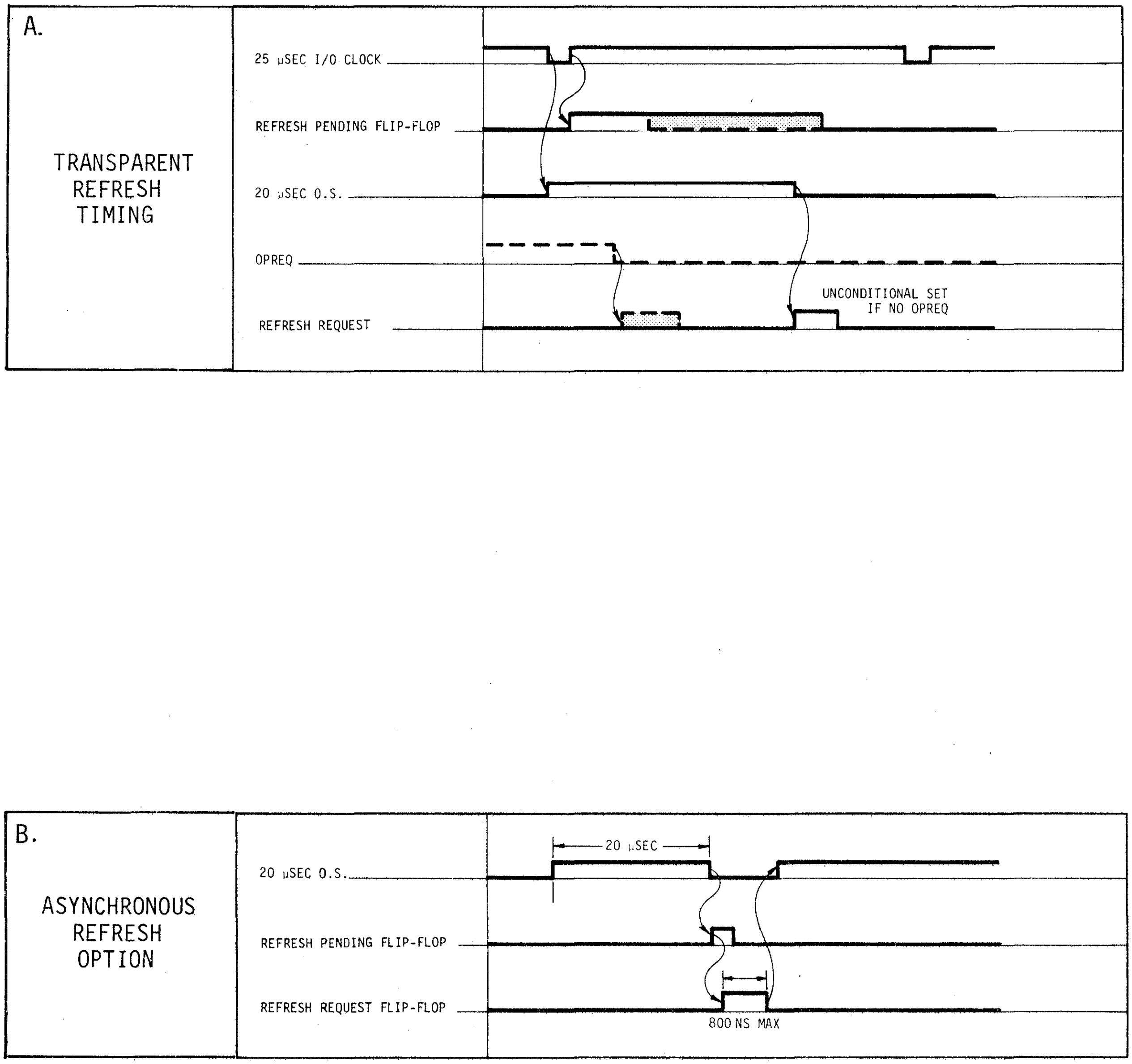

A.
TRANSPARENT REFRESH TIMING
25 μSEC I/O CLOCK
REFRESH PENDING FLIP-FLOP
20 μSEC O.S.
OPREQ
REFRESH REQUEST
UNCONDITIONAL SET IF NO OPREQ
B.
ASYNCHRONOUS REFRESH OPTION
20 μSEC
20 μSEC O.S.
REFRESH PENDING FLIP-FLOP
REFRESH REQUEST FLIP-FLOP
800 NS MAX

FIGURE 6-10 TIMING DIAGRAMS

6.2.8 Operation Control and Timing Figure 6-11 has this section shaded while figure 6-12 is the block diagram. The Operation Control receives system access requests and refresh requests, resolves contention, and generates all timing signals for writes, read and refresh.

An OR function of REFRESH REQUEST and SYSTEM REQUEST is generated and allowed to clock the OPERATION CONTROL flip flop if the memory is not presently being accessed as indicated by the busy signal. REFRESH REQUEST always takes priority and determines the state of the OPERATION flip flop.

The same basic timing generator is used to generate read, write and refresh timing. The generator consists of a PULSE flip flop driving a delay line with various taps of the delay line selected depending on the operation. When a system or refresh request is received the PULSE flip flop sends a low going edge down the 200 ns delay line. When the edge reaches the end of the delay line it resets the PULSE flip flop sending a positive edge down the line. This gives the basic 400 ns cycle time required by the memory. Timing diagrams for WRITE, READ and REFRESH operations are shown in figure 6-13 a, b and c. A multiplexer and gates operating from the tapped delay line generate and select the basic timing pulses. The select line on the multiplexer is controlled by the WRITE flip flop and the gate input is used to inhibit timing pulses during the refresh operation. The following paragraphs discuss, in detail, the READ, WRITE and REFRESH operation.

For a READ operation, the address is decoded to provide a MODULE SELECTED signal. OPREQ clocks the SYSTEM PENDING flip flop which is directly gated through as a SYSTEM REQUEST. If the memory is not busy, this sets the PULSE flip flop and clocks the OPERATION CONTROL and WRITE flip flips. If a refresh is pending, this takes precedence and a refresh cycle occurs. Due to the transparent refresh request, this will normally not occur and a read operation will start. BUSY is generated prohibiting another memory cycle starting before the full 400 ns. CHIP ENABLE is generated for 240 ns to the appropriate row and the memory data out is sampled 220 ns into the cycle. At the same time, the HOLD signal is reset, signaling the processor that data is valid. Total access time from receipt of OPREQ to bus data valid is 380 ns. After 400 ns into the cycle, the BUSY line is lowered allowing another memory cycle if desired.

Write timing is very similar except that the operation is not allowed to start until the receipt of WRP indicating that write data is valid. OPREQ clocks the SYSTEM PENDING flip flop which generates the HOLD signal. SYSTEM REQUEST is held off until receipt of WRP. As in a read, SYSTEM REQUEST is allowed to set the PULSE flip flop, clock the OPERATION CONTROL flip flop, and clock the WRITE flip flop if the BUSY line is not active. The CHIP ENABLE is generated and write data is sampled 40 ns into the operation. The WRITE ENABLE (WE) is generated from 60 to 260 ns. Because the write data is stored and the memory chips store the address, the HOLD signal is reset early in the cycle allowing the processor to proceed. The memory is still busy for the full 400 ns cycle..

Refresh timing starts with an I/O clock which triggers the 20 µs one shot on the leading edge and the REFRESH PENDING flip flop on the trailing edge. The next OPREQ trailing edge clocks the REFRESH REQUEST flip flop. If the memory is not busy the PULSE flip flop is set and OPERATION flip flop is clocked to a REFRESH cycle. This inhibits the WRITE ENABLE, and all read and write strobes. It also switches the address multiplexer to the refresh address and enables all four CHIP ENABLE lines. The timing is the same as a normal read cycle and takes 400 ns. The REFRESH REQUEST and OPERATION CONTROL flip flop are reset at 320 ns to allow maximum address setup time for a system request.

6.3 UTILIZATION

6.3.1 Installation

The module is designed to be plugged into the card file of the development computer, but may be used in other applications. Jumper and switch options are discussed in section 6.3.2. All interface

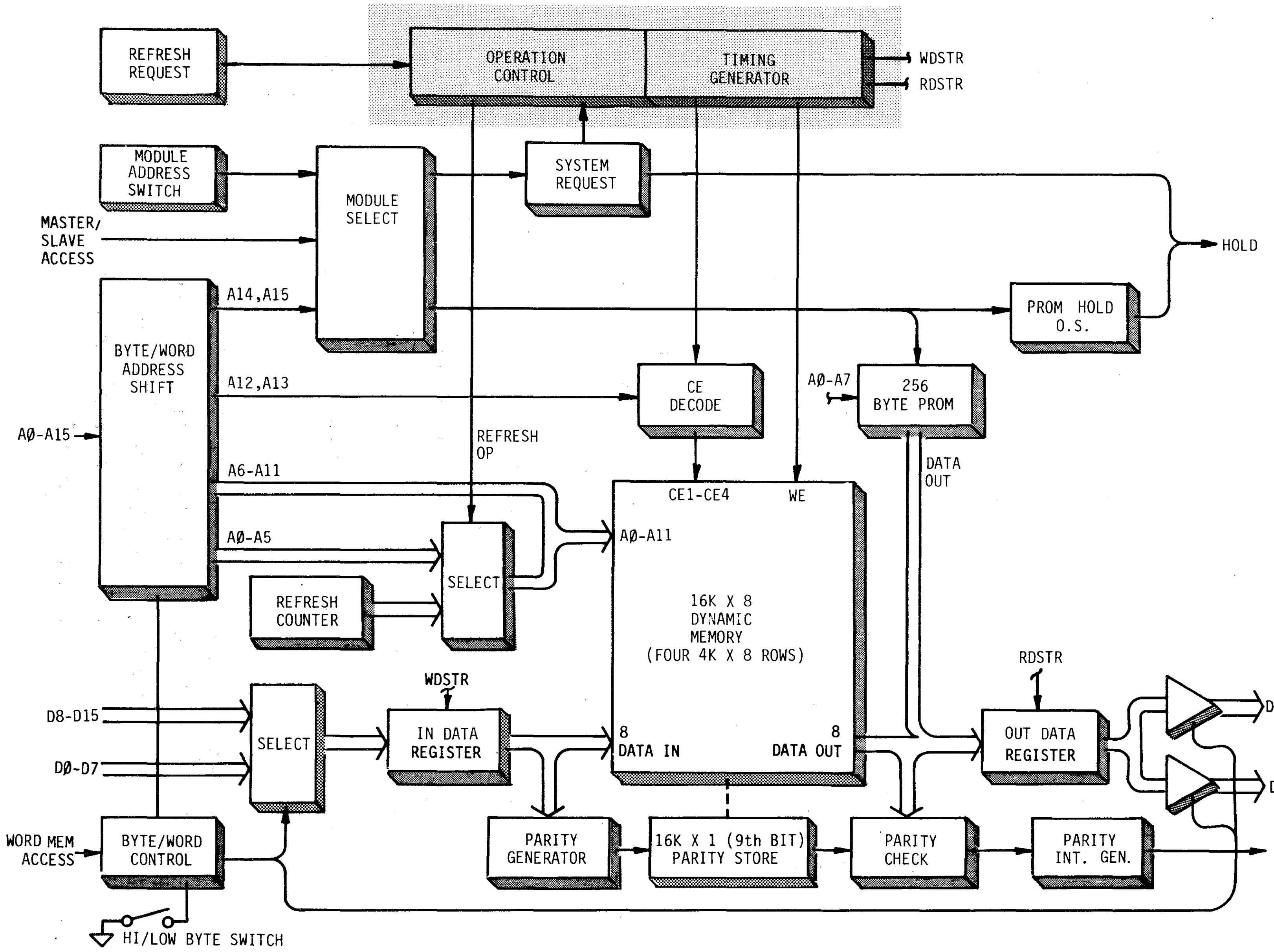


FIGURE 6-11 OPERATION CONTROL AND TIMING

connections and connector pinouts are detailed in section 6.3.3.

COMPUTER INSTALLATION A Master Dynamic RAM Module may be plugged into any available Master section slot of the motherboard (J3-J8). A Slave designated Dynamic RAM module may be plugged into any available slot of the motherboard (J3-J8, J10-J20). The card connector is offset, insuring that it cannot be plugged in backwards. All power requirements (+5V, +12V) are supplied through the motherboard.

INSTALLATION IN OTHER APPLICATIONS When using the module in other applications, the user must take into consideration environmental extremes, mounting, power requirements, interface connections and signal AC and DC requirements. Environmental limits, power requirements and card outline dimensions are listed in figure 6-14.

For mounting, the board is designed to plug into a 100 pin edge card connector, CDC VPB-01C5-0D00A1 or equivalent. Card guide slots should be provided and the card should be mounted vertically or in some way clamped so that it does not vibrate out of its socket. Vertical mounting is also important to insure convective cooling if no fan is used. The card dissipates up to 13 watts, therefore adequate space should be allowed between modules for air flow. It is desirable to have forced air flow between all modules if several modules are utilized within a confined area.

6.3.2 Switch and Jumper Options User selected switch or jumper options include:

1) Module address selections (D16-1, D16-2)
2) Master-Common Memory designation (D16-3)
3) Byte-Word Memory designation (J9, J10)
4) Hi-Low byte designation for Word Memory (D16-4)
5) Refresh timing option (J5-J8)

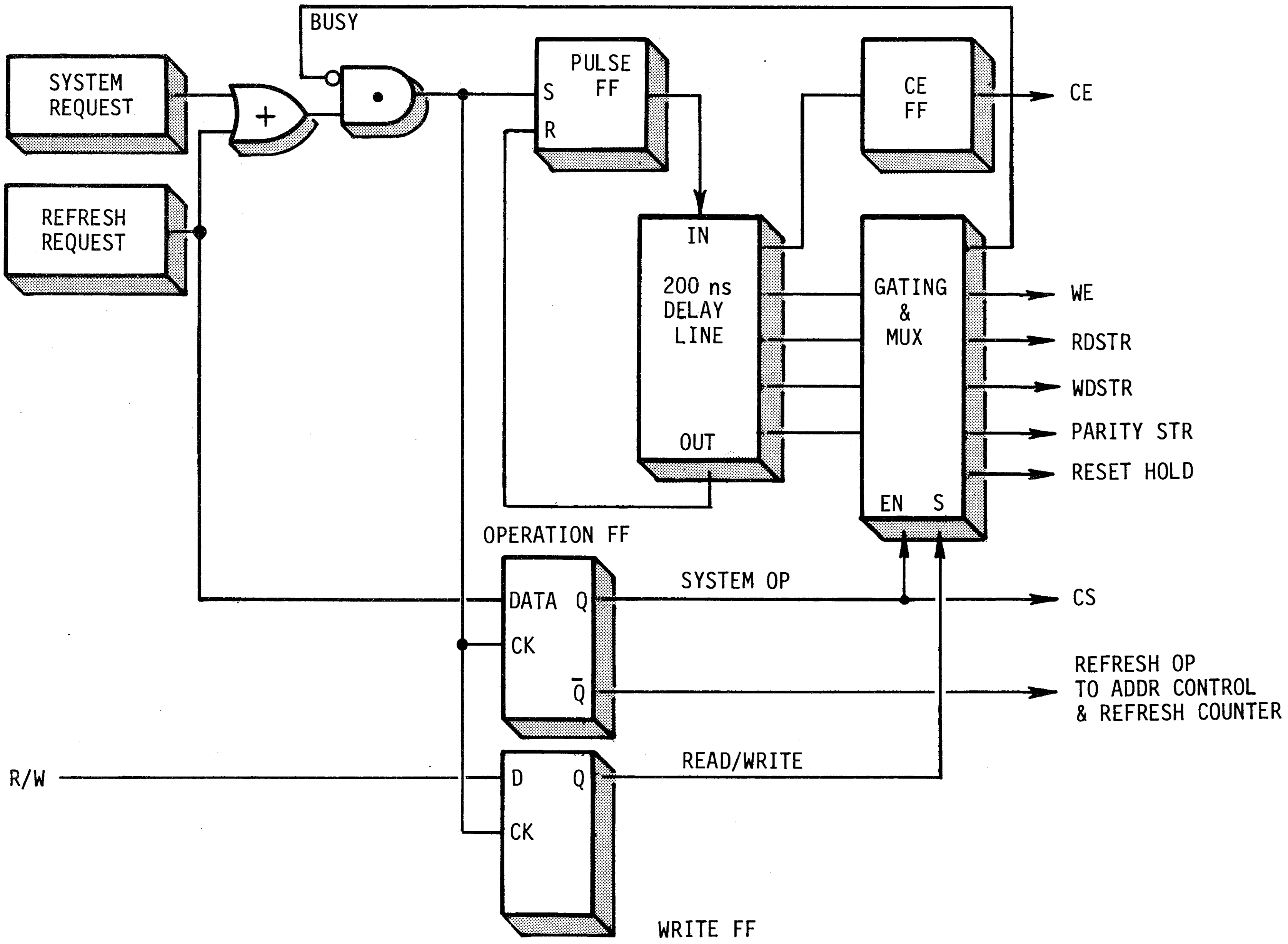
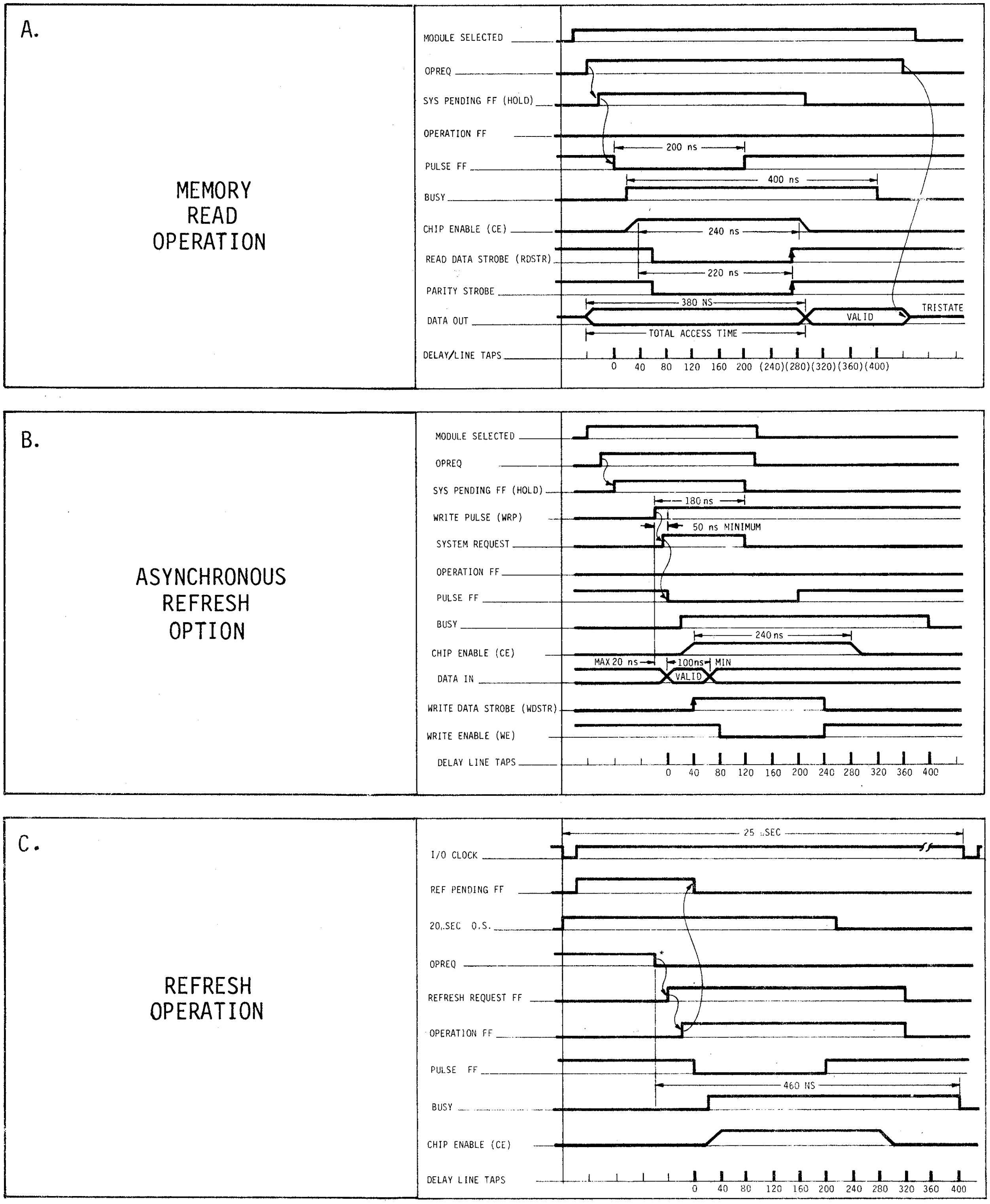


FIGURE 6-12 OPERATION CONTROL AND TIMING BLOCK DIAGRAM



*IF NO OPREQ TRAILING EDGE, REF REQ SET ON TRAILING EDGE OF 20 μSEC ONE SHOT.

FIGURE 6-13 TIMING DIAGRAMS

6) PROM Disable (J2 & J4)
7) HOLD Disable (J1)
8) Parity Interrupt Disable (J3)

Module Address and Master-Slave Selection -- A four position slide switch at position D16 allows setting of module or base address (A15, A14), Master or Common selection, and High or Low byte designation if Word Memory.

Switch	Description	Switch On Indicates
D16-1	A14 (16K weight)	A14 = 1
D16-2	A15 (32K weight)	A15 = 1
D16-3	Master/Slave	Slave
D16-4	High/Low	High

Word Memory Usage -- For use as a word memory with 16 bit Slave processors, the memory must be used in pairs with the same module or base address. The modules are designated as Word Memory by adding J9 and deleting J10. J9 is located under switch D17 and J10 is located between E15 and E16. One module must be selected High byte and the other Low byte (switch D16-4). Byte oriented processors are still able to access both memories in a ping-pong fashion.

Refresh Timing -- The board is configured to operate from a 38.4 KHz clock provided by the system bus. If used in an application where a clock similar to this is unavailable, the card may be reconfigured to operate off an internal multivibrator. The Refresh is then no longer interleaved with OPREQ, but becomes an asynchronous request at a 20 μs rate. To reconfigure, J5 and J8 are cut and J7 and J6 are added. These jumpers are all located between C16 and C17.

PROM Disable -- The PROM is normally disabled by the Master/Slave select switch when in the Slave position. The PROM may be permanently disabled by cutting J4 and adding J2. These jumpers are near B11.

HOLD Disable -- The HOLD signal during a RAM access may be disabled by cutting J1 located between A15 and A16.

Parity Interrupt Disable -- All parity interrupts may be disabled by cutting J3 located between B11 and B12

6.3.3 Interface and Connector Pinouts Figure 6-15 lists the motherboard connector (P1) pinouts and signal definitions.

ENVIRONMENTAL

Air Temp 0° - 70°C Operational
Surrounding Module

Storage Temp -55° - 125°C

Humidity 90% Non-condensing

POWER

	TYP	MAX
+5VDC	1.2A	1.5A
+12VDC	85ma	600ma

PHYSICAL OUTLINE

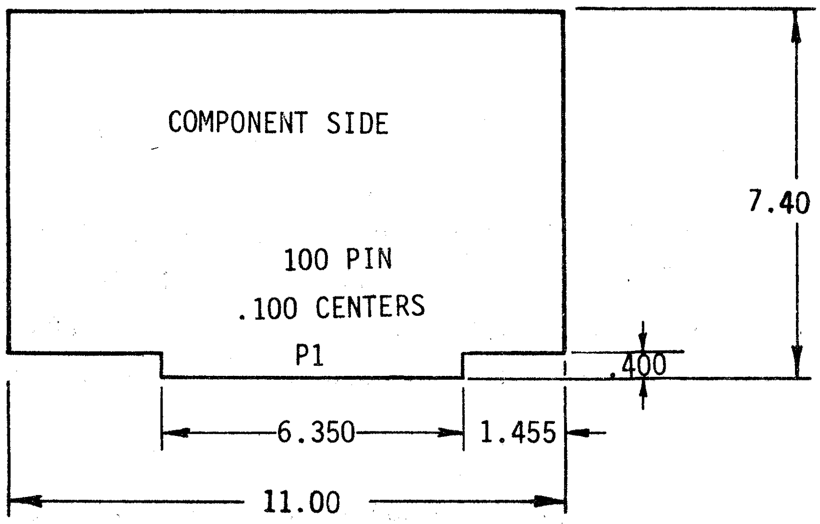


FIGURE 6-14 16K DYNAMIC RAM ENVIRONMENTAL, POWER AND PHYSICAL REQUIREMENTS

	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	+5V	+5VDC In	2	+5V	+5VDC In
	3	+5V	+5VDC In	4	+5V	+5VDC In
	5	AUX BUS	Not Used	6	AUX BUS	Not Used
	7	AUX BUS	Not Used	8	AUX BUS	Not Used
	9	GND	Signal & Pwr GND	10	GND	Signal & Pwr GND
	11	+12V	+12VDC In	12	+12V	+12 VDC In
	13	-12V	Not Used	14	-12V	Not Used
	15	GND	Signal & Pwr GND	16	GND	Signal & Pwr GND
ADDRESS SYSTEM	17	$\overline{A0}$	Address Bus In	18	$\overline{A1}$	Address Bus In
	19	$\overline{A2}$		20	$\overline{A3}$	
	21	$\overline{A4}$		22	$\overline{A5}$	
	23	$\overline{A6}$		24	$\overline{A7}$	
	25	$\overline{A8}$		26	$\overline{A9}$	
	27	$\overline{A10}$		28	$\overline{A11}$	
	29	$\overline{A12}$		30	$\overline{A13}$	
	31	$\overline{A14}$		32	$\overline{A15}$	
MEMORY CONTROL	33	$\overline{\text{CMEM}}$	Slave/MSTR MEM In	34	$\overline{\text{RAM INH}}$	RAM Memory Inhibit In
	35	$\overline{\text{WD ACCESS}}$	Word Processor Access In			
DATA BUS	37	$\overline{D1}$	Data Bus In/Out	36	$\overline{D0}$	Data Bus In/Out
	39	$\overline{D3}$		38	$\overline{D2}$	
	41	$\overline{D5}$		40	$\overline{D4}$	
	43	$\overline{D7}$		42	$\overline{D6}$	
	45	$\overline{D9}$		44	$\overline{D8}$	
	47	$\overline{D11}$		46	$\overline{D10}$	
	49	$\overline{D13}$		48	$\overline{D12}$	
	51	$\overline{D15}$		50	$\overline{D14}$	
BUS CONTROL				52	M/IO	Memory/In
	53	$\overline{\text{WRP}}$	Write Pulse In	54	OPREQ	Operation Request In
	55	R/$\overline{\text{W}}$	Read/Write In	56	HOLD	Hold CPU Out
	57	$\overline{\text{JMP CMD}}$	Not Used	58	RUN	Not Used
	59	$\overline{\text{RESET}}$	System Reset In	60	JMP ACK	Not Used
SPLIT BUS	61			62		
	63	$\overline{\text{MSTR INT 1}}$	Master Interrupt 1 Out	64		
	65			66		
	67			68		
	69			70		
	71			72		
	73			74		
	75			76		
	77			78		
	79			80		
	81					
BUS CONTROL (Except J1-J2)				82	FETCH	Not Used
	83	$\overline{\text{PAUSE}}$	Not Used	84	MAST RUN	Not Used
	85	$\overline{\text{SENSE}}$	Not Used	86	FLAG	Not Used
	87	$\overline{\text{F.P. HOLD}}$	Not Used	88	MST INTACK	MSTR Interrupt Ack In
	89	$\overline{\text{MST INT 3}}$	Master Interrupt 3 Out	90	SPARE	Not Used
	91	SPARE	Not Used	92	SPARE	Not Used
CLOCKS (Except J1-J2)	93	$\overline{\text{SLV CLK}}$	Not Used	94	I/O CLK	38.4 KHz I/O Clock In
	95	$\overline{\text{2650 CLK}}$	Not Used	96	SYS CLK	Not Used
GROUND	97	GND	Signal & Pwr GND	98	GND	Signal & Pwr GND
	99	GND	Signal & Pwr GND	100	GND	Signal & Pwr GND

FIGURE 6-15 16K DYNAMIC RAM MOTHERBOARD PINOUTS (P1)

Chapter 7

GENERAL PURPOSE I/O MODULE

7.0 INTRODUCTION

The General Purpose Input/Output module (G.P.I/O) is a system option designed to provide the user with a method of interfacing a processor (either Master or Slave) to a wide variety of peripheral devices. The board includes a complete EIA RS-232 standard serial interface with all serial-parallel conversion and status and control functions. It has four eight bit output ports, and four eight bit input ports. For interrupt driven peripherals, eight interrupt circuits are provided.

Emphasis has been placed on ease of interface. Four edge connectors are provided at the top edge of the board for connection to external peripheral devices. One connector provides the standard RS-232 port and another contains two input and two output ports. Two 25 pin connectors are unassigned and may be custom wirewrapped to any of the four output or four input ports as well as to the interrupt circuit triggers. When used in conjunction with a standard flat I/O cable, a rear panel 25 pin "D" connection is obtained. One board can easily be configured to interface several peripheral devices simultaneously.

7.1 GENERAL DESCRIPTION

Figure 7-1 is a simplified block diagram of the module. The module performs its functions in four basic blocks: The RS-232 interface, the four sets of parallel Input-Output ports, the eight interrupt circuits, and the module and port address decode.

The RS-232 interface is provided through two sets of I/O ports and a Universal Asynchonous Receiver-Transmitter (UART) device. The UART performs serial to parallel conversion functions and provides storage for both the input and output characters transmitted to and from the modules. Full

EIA RS-232 control output and status input with EIA level transmitters and receivers are for use with a wide variety of modules and terminals. A 20 ma current loop TTY interface is provided in parallel utilizing the same UART chip. Separate control lines for the TTY paper tape Reader/Punch are also provided. A single switch controls baud rate and stop bits required for TTY or EIA operation. The EIA baud rate is jumper selectable at 150, 300, 600 or 1200 baud while the TTY baud rate is fixed at 110.

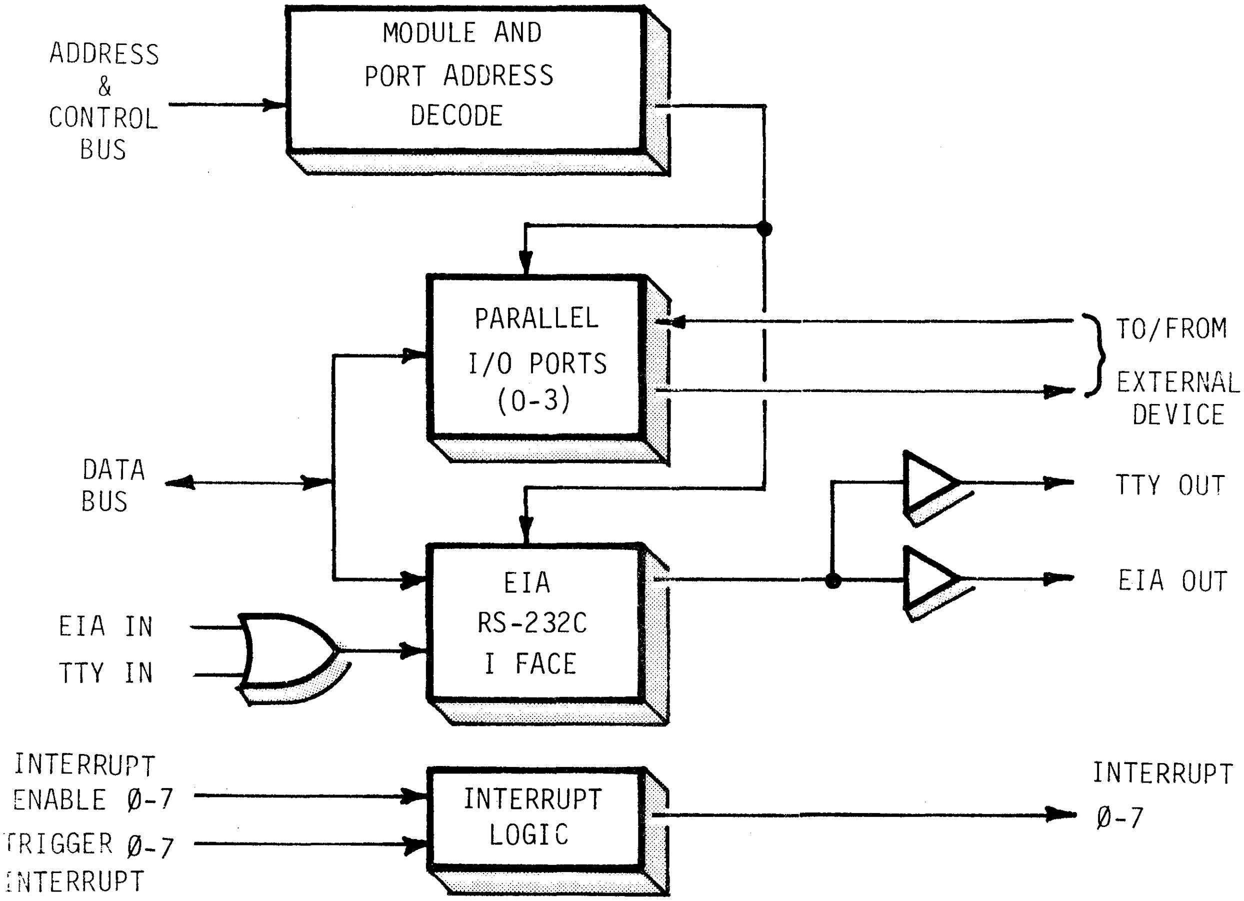


FIGURE 7-1 GP I/O SIMPLIFIED BLOCK DIAGRAM

The parallel ports consist of eight bit registers and driver gates for the output ports. Terminated input tristate gates for multiplexing the selected input port to the system bus are also provided. Independent lines are provided for input and output. Bidirectional driver gates interface the module to the system data bus.

The eight interrupt circuits include individual flip flops with selectable triggering on either signal edge and a program or externally controlled enable function. Logic is also included to reset the interrupt upon acknowledgement by the CPU.

The G.P.I/O card contains address decoding which consists of module address recognition and port decoding. There are 32 possible module addresses and each module is assigned eight port addresses. Three port addresses are assigned to the RS-232 interface and four to the parallel ports. The module address and type (Master or Slave) is switch selectable.

7.2 DETAILED DESCRIPTION

This section provides detailed information on how the General Purpose I/O board performs its communications function.

There are essentially four major functional areas:

1. RS-232 serial interface and associated I/O ports
2. Module and Port address decode
3. Parallel port input/output
4. External interrupt service

7.2.1 Serial I/O Ports The serial I/O port is a full EIA RS-232 communications interface. This port is utilized for EIA or TTY operation. Refer to the shaded portion figure 7-2.

Parallel to serial transmit operation and serial to parallel receive operation are performed by a UART (universal asychronous receiver/transmiter) device. The eight transmit data bits (DB1-DB8) are connected to the board's internal output data bus. The eight receive data bits (RD1-RD8) are connected to the internal input data bus. Both buses are connected to the system data bus via tristate drivers and receivers. Formats and port address for the serial interface are shown in figure 7-3. Signals are true at logic level "1".

UART initialization is accomplished during system reset. Transmit Buffer Empty, End of Character, and Serial Out are set true during this operation, as well as resetting Data Available. The device is hardwired for seven bits per character. The No Parity option is defeated and odd or even parity is under program control. The number of stop bits is determined by the operating mode (EIA or TTY) and is set by switch E1. The UART device monitors the number of stop bits and parity and compares these to the previously programmed control bits.

Transmit operation is initiated under CPU control by first checking the status register for a Transmit Buffer Empty indication. Data is then strobed into the internal UART data register by a Write Port 7 strobe. Transmit Buffer Empty will then go to a logic "0" at this time and the contents of the data buffer will be loaded into a UART shift register. Completion of the transfer is signified by Serial Out and End of Character both going to a logic "0". Serial transmission starts and Transmit Buffer Empty goes true immediately thereafter, generating Interrupt 7 which signifies that the next data byte may be loaded into the data register. Completion of transmission is indicated by End of Character going true.

Receive operation begins with verification of a valid start bit at the serial data input to the UART. Serial data, Parity, and Stop bits are now received and monitored. Status is latched internally and made available at this time. Internal logic checks Data Available. If Data Available is true, the Overrun bit is set. If Data Available is false, received data will then

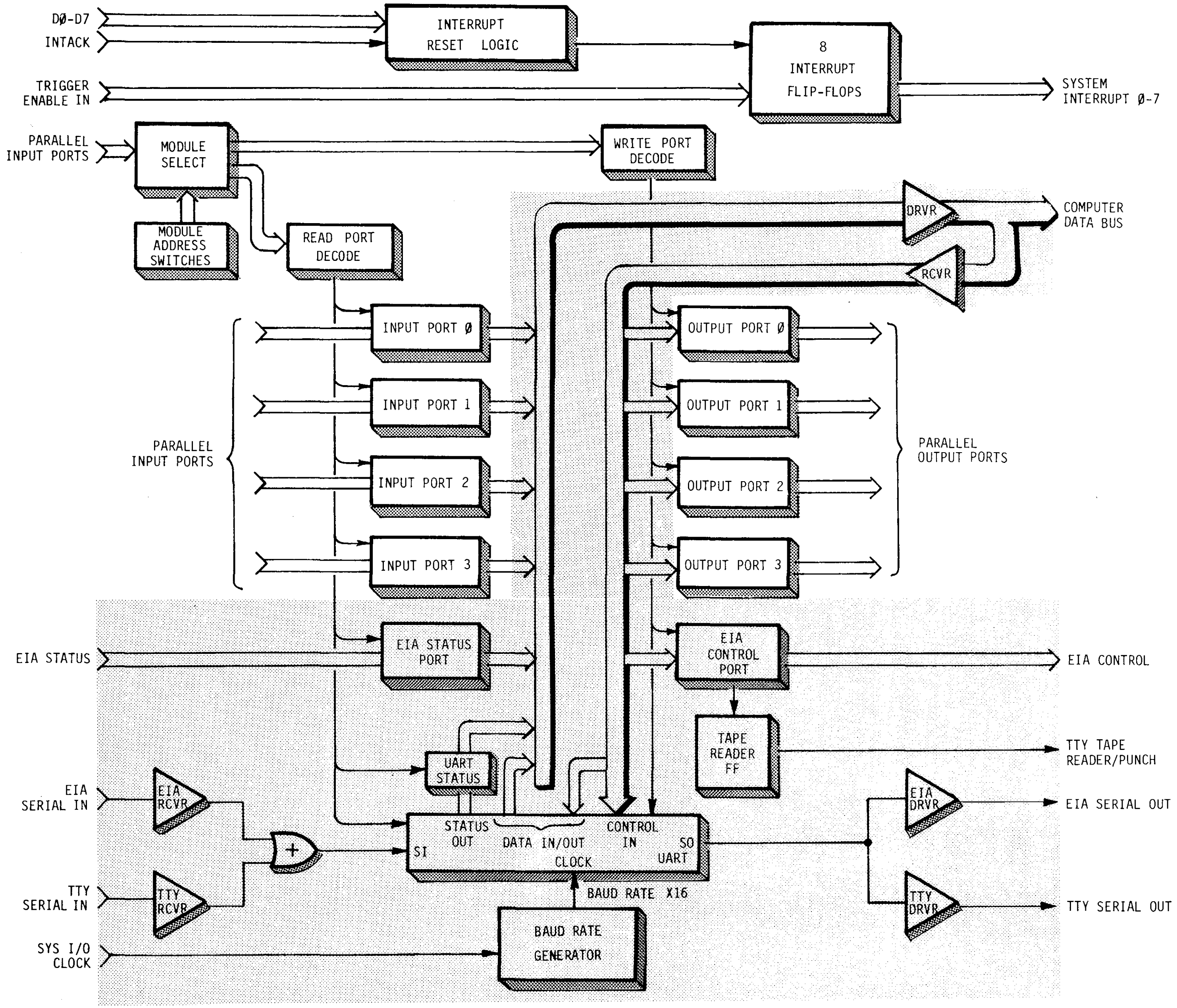


FIGURE 7-2 SERIAL I/O PORTS

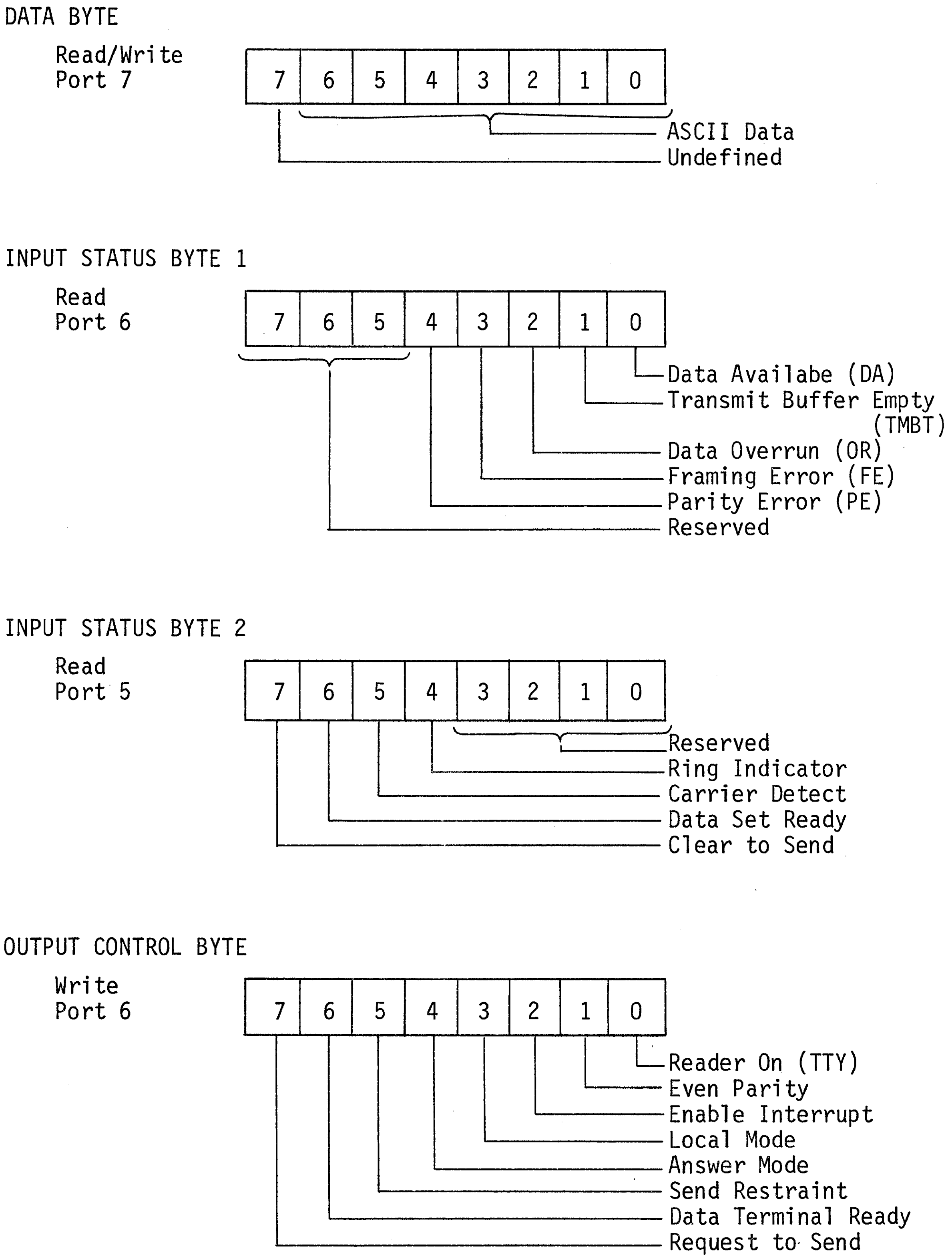


FIGURE 7-3 SERIAL PORT BYTE FORMATS

be transferred from the serial input register to the received data holding register. Completion of this transfer is indicated by Data Available going true which generates interrupt 6. Receive status is now checked with a read of port 6. The next character can now be loaded into the serial input register, with one full character time allowed to remove the previous character from the holding register. Read port 7 generates a Receive Data Enable which transfers data from the holding register to the system data bus for processing. This action also resets Data Available which indicates completion and allows the next byte to be transferred from the serial input register to the holding register.

External EIA receive status is converted from EIA levels to TTL levels by use of EIA line receivers. These are connected to the internal data bus via tri-state drivers which are enabled by a read port 5 command. A write port 6 will latch the designated EIA control data from the module's output data bus to the appropriate EIA line drivers. Refer to figure 7-4 for formats.

Various baud rates are available for serial I/O operation. In low speed mode, the baud rate is 110. This is used for TTY operation. Switch E1 #8 is used to set high or low baud rate and also sets the number of stop bits used -- one bit for high speed operation and two bits for low speed operation. In high speed mode, the rate may be jumper-selected for 150, 300, 600 and 1200 baud.

Teletype operation also utilizes the UART chip. Serial out (SO) is paralleled to an open collector driver and serial in (SI) is "OR" connected for 20 ma operation. Because of the paralleled operation, only one can be used at a time. Teletype baud rate and number of stop bits is controlled by the baud rate switch as previously described. A paper tape reader control is also available for ASR paper tape operation. It is set at write port 6 time using bit Ø of this control byte. It is reset upon receipt of a start bit on the serial stream or under program control.

7.2.2 Module and Port Address Control Because there may be more than one I/O module in the system, a means of selecting individual modules is required. This logic block is shown in the shaded portion of figure 7-4. Address control is responsible for module address decoding, port decoding, Master or Slave operation, and read or write functions.

Module address selection is accomplished by comparison of the board-mounted address switch settings and the contents of the system address bus. Module address is set with the address select switch located at E1 on the printed circuit board. Module select logic also determines Master or Slave operation and is done by comparison with the MSTR RUN signal from the system bus. Two three-line to eight-line decoders are used to determine the address of the required port. One decoder is used for read port operation and the other for write port operation. Module Select ANDed with Read or Write is used to enable the appropriate decoding of address bits AØ-A2 to determine which port is to be used. Address bit 13 (extended I/O) in conjunction with an I/O command and OPREQ enables the Module Select line for transmission to the decoders. In addition, the Write Decoder uses the CPU write pulse (WRP) as a strobe signal.

7.2.3 Parallel Input and Output Ports Four discrete eight bit parallel output and input ports shown in the shaded portion of figure 7-5 are available. The output ports have separate registers whose inputs are directly connected to the internal output data bus. These ports are accessed via write commands to each individual port address relative to the module address setting.

The input ports are terminated tri-state gates connected to the internal input data bus. Data is enabled onto the input data bus with individual read port commands.

Ports Ø and 1 are hardwired to card edge connector P3. In addition, these two ports are also connected to IC locations A6 and A7. Ports 2 and 3 are

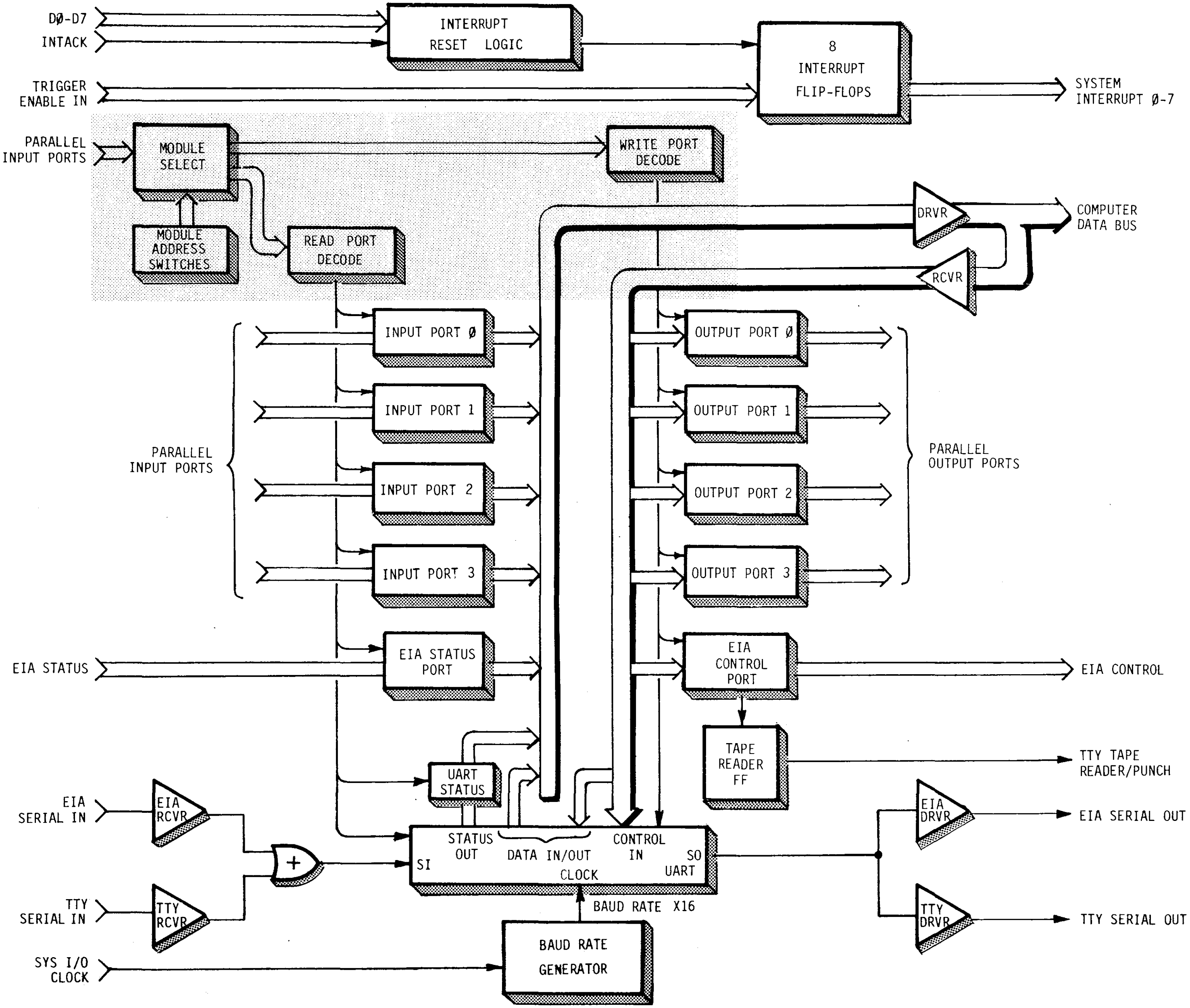


FIGURE 7-4 MODULE AND PORT ADDRESS CONTROL

hardwired to IC locations A9 and A10. Card edge connectors P4 and P5 are unassigned. They may be wirewrapped to any combination of the parallel ports. See section 7.3 for detailed wirewrapping information.

7.2.4 Interrupt Logic

This logic is shown in the shaded portion of figure 7-6. There are eight identical circuits, consisting of a schmidt trigger, an exclusive or gate for edge trigger selection and a flip-flop.

The interrupt flip-flops are set to non-active or false condition on power up. The system clock and the three line to eight line decoders sequence reset to each flip-flop at that time.

The interrupt flip-flops may be enabled under program control by wiring the desired output port bit to the enable input. The flip-flops normally trigger off a low going edge. Jumpers connected to an exclusive "OR" may be cut to provide positive edge triggering. When triggered, the interrupts remain set until the interrupt is acknowledged by its vector on the data bus and Interrupt Acknowledge.

Interrupts 6 and 7 are hardwire jumpered to Serial Data Available and Transmit Buffer Empty respecitvely. They may be cut and reconfigured. The remaining interrupt flip-flop trigger and enable inputs can be wirewrapped to available ports dependent on user application. A selected eight of the sixteen Master or eight Slave interrupts can be configured in this manner.

7.3 UTILIZATION

7.3.1 Installation

The module is designed to be plugged into the card file of the development computer, but may be used in other applications. Jumper and switch options are discussed in section 7.3.2. All interface connections and connector pinouts are detailed in section 7.3.3.

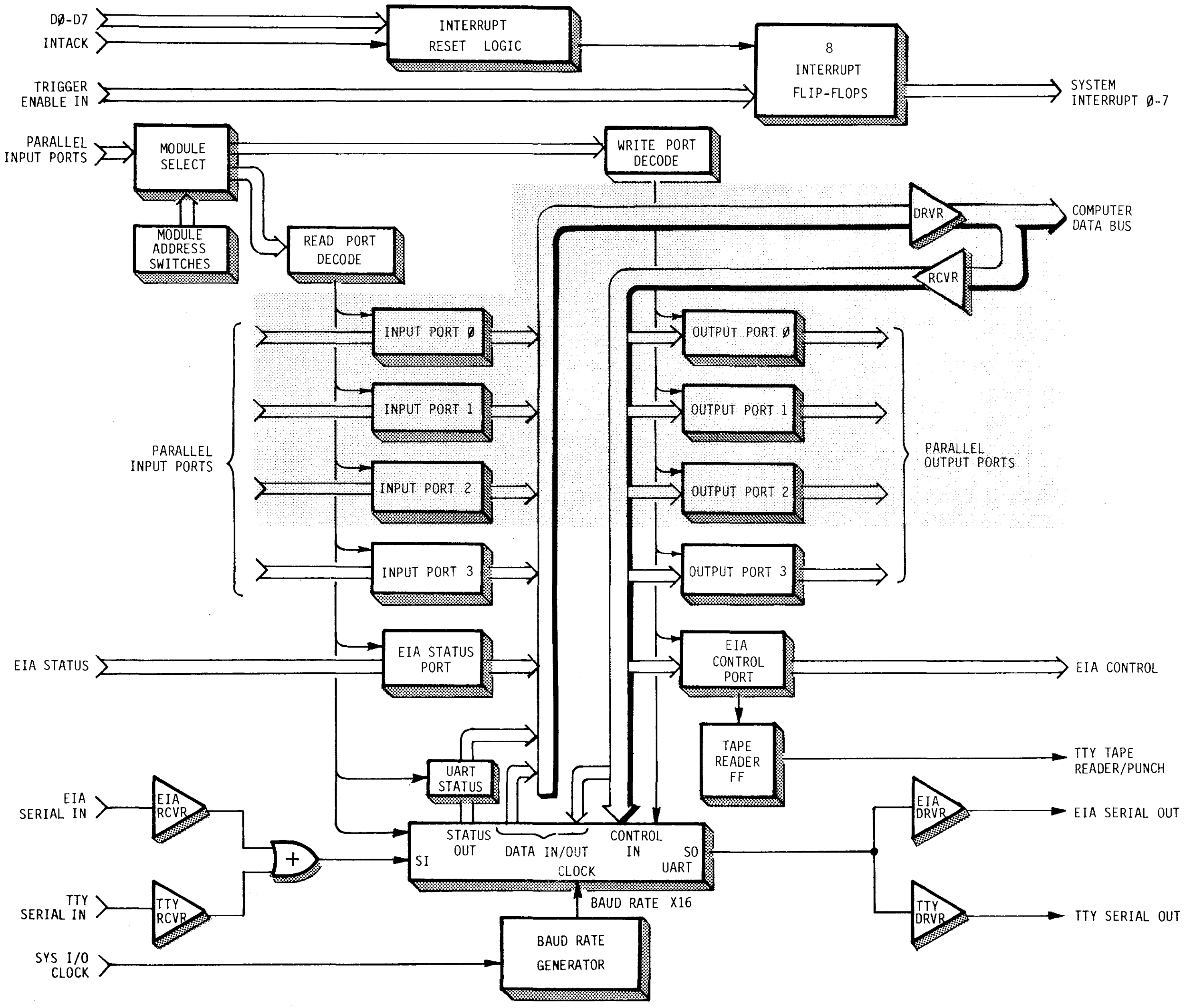


FIGURE 7-5 PARALLEL INPUT-OUTPUT PORTS

COMPUTER INSTALLATION For Master I/O operation with interrupts, the GPI/O Module plugs into any available Master section slot of the motherboard (J3-J8). For Slave I/O operation with interrupts, the module plugs into any available Slave section slot (J10-J20). The card connector is offset, insuring that it cannot be plugged in backwards. All power requirements (+5V, ±12V) are supplied through the motherboard. Cable 90014011, a standard flat ribbon 25 pin I/O cable can be attached to the top edge connector P2 for serial RS-232 input/output or P4 or P5 for custom I/O. This cable provides a rear panel 25 pin bulkhead connection. Care should be taken that pin 1 of the cable (Red Stripe) and pin 1 of the edge connector align. Pin 1 is to the left when viewed from the component side of the board.

INSTALLATION IN OTHER APPLICATIONS When using the module in other applications, the user must take into consideration environmental extremes, mounting, power requirements, interface connections and signal AC and DC requirements. Environmental limits, power requirements and card outline dimensions are listed in figure 7-7.

For mounting, the board is designed to plug into a 100 pin edge card connector, CDC VPB-01C5-0D00A1 or equivalent. Card guide slots should be provided and the card should be mounted vertically or in some way clamped so that it does not vibrate out of its socket. Vertical mounting is also important to insure convective cooling if no fan is used. The card dissipates approximately 10 watts, therefore adequate space should be allowed between modules for air flow. It is desirable to have forced air flow between all modules if several modules are utilized within a confined area.

Interface connections and signal requirements are given in section 7.3.3.

7.3.2 Switch and Jumper Options Switch Settings -- There is a single DIP switch mounted on the General Purpose I/O board at location E1. This switch selects module address and baud rate. The upper five switch sections select module address; the lower switch selects baud rate and number

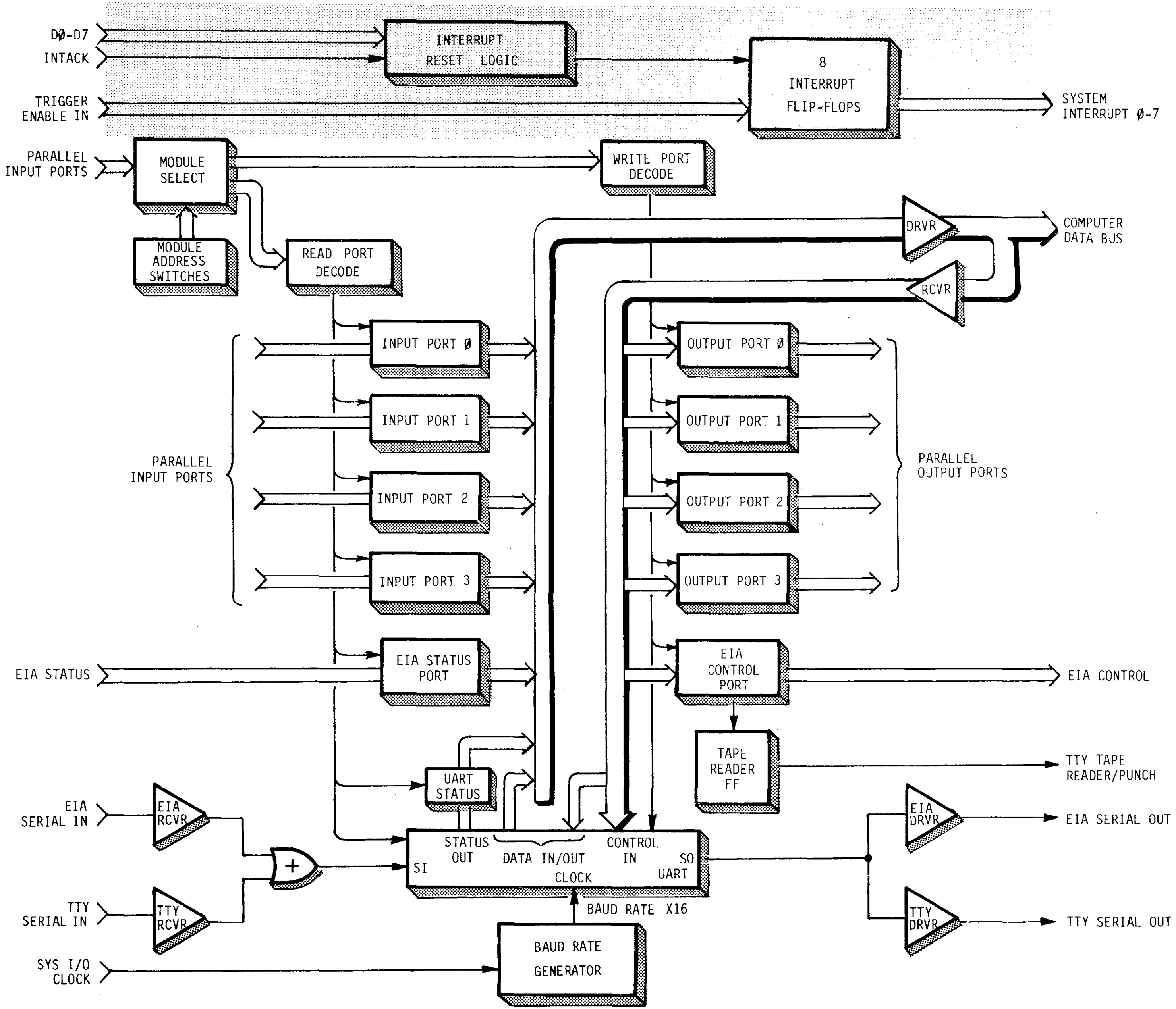


FIGURE 7-6 INTERRUPT LOGIC

ENVIRONMENTAL

Air Temp 0° - 70°C Operational
Surrounding Module

Storage Temp -55° - 125°C

Humidity 90% Non-condensing

POWER

	TYP	MAX
+5VDC	1.8A	2.3A
+12VDC	.05A	.08A
-12VDC	.05A	.08A

PHYSICAL OUTLINE

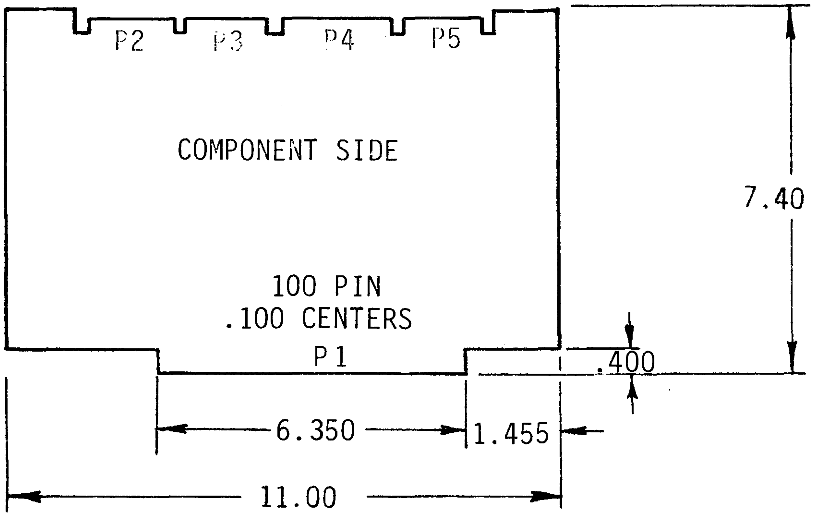


FIGURE 7-7 GPI/O ENVIRONMENTAL, POWER AND PHYSICAL REQUIREMENTS

of stop bits; switch number 1 selects Master or Slave I/O. The table below shows settings for address, Master/Slave, and baud rate.

Switch	Function	ON Position Indicates
E1-1	A3	A3 = 1 (weight 8)
E1-2	A4	A4 = 1 (weight 16)
E1-3	A5	A5 = 1 (weight 32)
E1-4	A6	A6 = 1 (weight 64)
E1-5	A7	A7 = 1 (weight 128)
E1-6	Not Used	
E1-7	Master/Slave	SLAVE
E1-8	TTY/EIA	TTY

NOTE: In the table above, "A5" refers to address line 5, "A3" refers to address line 3, etc. The board can be switched for any combination of the five lines, and this address is used in the extended read or write command in the user program.

Jumpers and User Options -- There are three possibilities for jumper installation on the board. Edge connectors P4 and P5 are not assigned to any I/O port. These can be wired by installing wirewrap pins in the holes provided and wiring to sockets mounted in board grid locations A6 (Port 0), A7 (Port 1), A8 (interrupts), A9 (Port 2), and A10 (Port 3). Figure 7-8 details the pinout for A6-A10. The second possibility is wiring the board for interrupt levels 8-15 by cutting the board traces for levels 0-7 and wiring to the proper holes in the board. The third possibility is selection of the EIA baud rate. Figure 7-9 shows jumper options and baud rate jumper placement.

Terminations have been provided on the parallel input ports. If not desired, they may be eliminated by removing the SIPS located at B6 through B10.

Four spare IC sockets for user breadboarding needs have been provided. They are located at A12 through A15, next to wirewrap connectors P4 and P5.

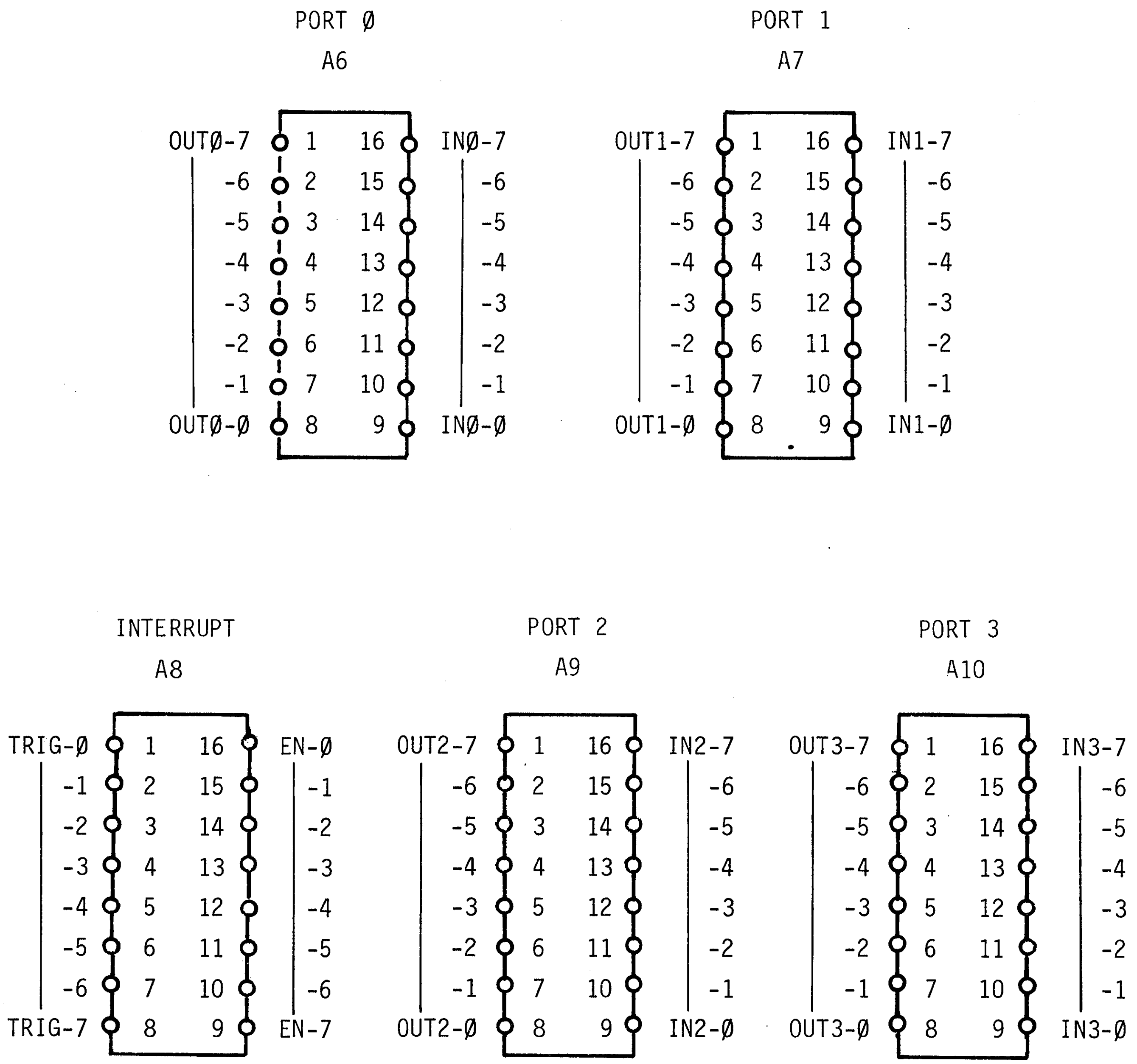


FIGURE 7-8 PARALLEL I/O AND INTERRUPT SIGNAL LOCATIONS

1) Interrupt Inputs

J1
J2
J3
J4

2) Interrupt Trigger Polarity
(cut for Positive Edge)

J5	Trig	- 7
J6		- 5
J7		- 1
J8		- 4
J9		- 6
J10		- 2
J11		- 3
J12	Trig	- Ø

3) Interrupt Resets

J13 - Reset 8

J20 - Reset 15
J21 - Reset Ø

J28 - Reset 7

4) Interrupt Outputs

J33 - INT Ø

J48 - INT 16

5) EIA Baud Rate

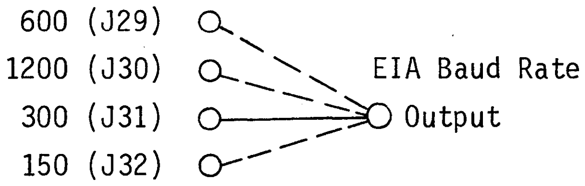


FIGURE 7-9 JUMPER OPTIONS

7.3.3 Interface and Connector Pinouts A complete interconnection diagram for GPI/O use is given in figure 7-10. Figure 7-11 lists the motherboard connector (P1) pinouts and signal definitions. Figures 7-12a and 7-12b list pinouts for the RS-232 connector (P2) and the 40 pin parallel I/O connector (P3) P4 and P5 are unassigned.

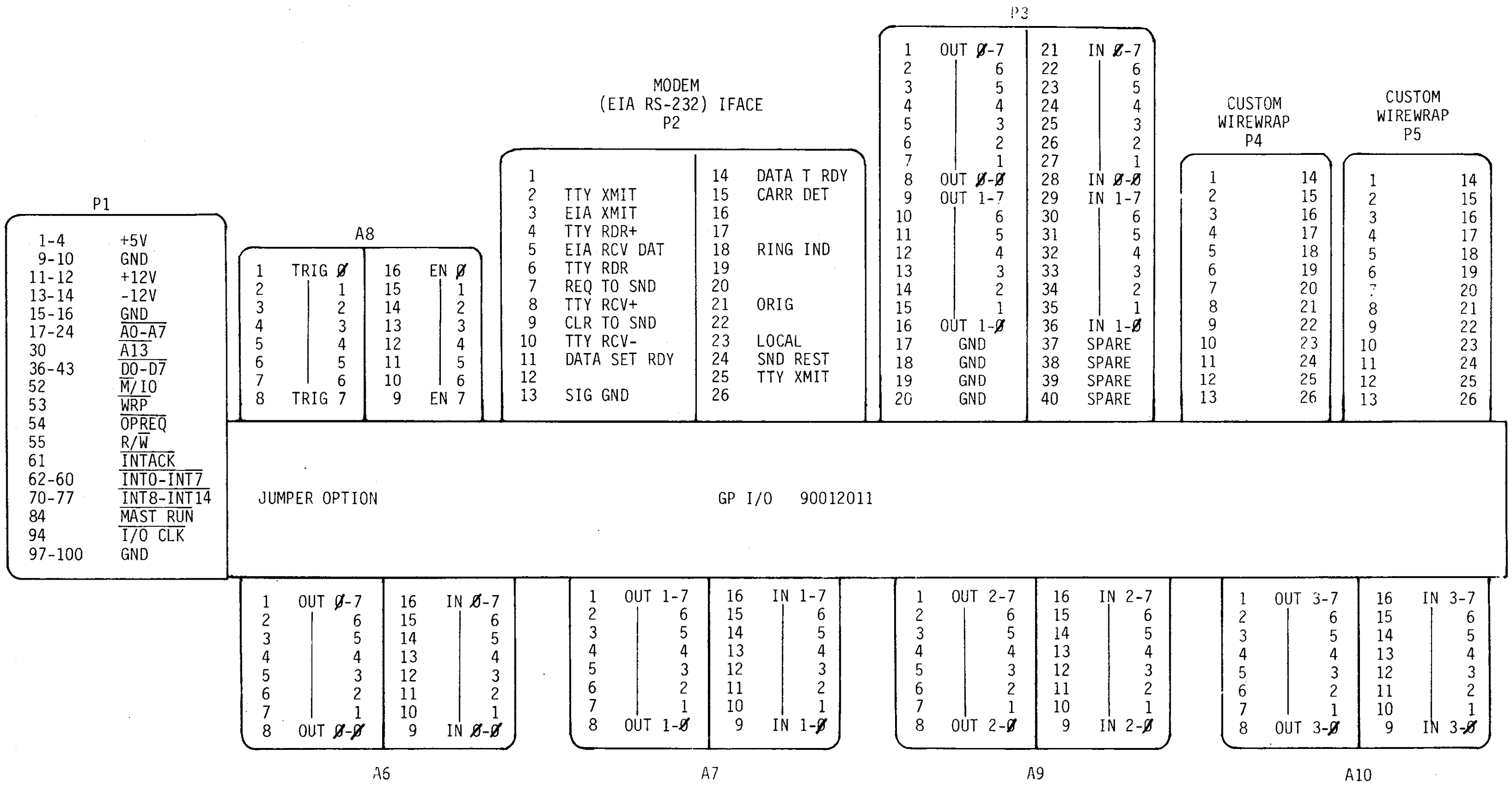


FIGURE 7-10 GP I/O INTERCONNECT

	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	+5V	+5VDC In	2	+5V	+5VDC In
	3	+5V	+5VDC In	4	+5V	+5VDC In
	5	AUX BUS	Not Used	6	AUX BUS	Not Used
	7	AUX BUS	Not Used	8	AUX BUS	Not Used
	9	GND	Signal & Pwr GND	10	GND	Signal & Pwr GND
	11	+12V	+12VDC In	12	+12V	+12VDC In
	13	-12V	-12VDC In	14	-12V	-12VDC In
	15	GND	Signal & Pwr GND	16	GND	Signal & Pwr GND
ADDRESS BUS	17	$\overline{\text{A0}}$	Address Bus In	18	$\overline{\text{A1}}$	Address Bus In
	19	$\overline{\text{A2}}$		20	$\overline{\text{A3}}$	
	21	$\overline{\text{A4}}$		22	$\overline{\text{A5}}$	
	23	$\overline{\text{A6}}$		24	$\overline{\text{A7}}$	
	25	$\overline{\text{A8}}$	Not Used	26	$\overline{\text{A9}}$	Not Used
	27	$\overline{\text{A10}}$		28	$\overline{\text{A11}}$	Not Used
	29	$\overline{\text{A12}}$		30	$\overline{\text{A13}}$	Extended I/O
	31	$\overline{\text{A14}}$		32	$\overline{\text{A15}}$	Not Used
MEMORY CONTROL	33	$\overline{\text{CMEM}}$	Not Used	34	$\overline{\text{RAM INH}}$	Not Used
	35	$\overline{\text{WD ACCESS}}$	Not Used			
DATA BUS	37	$\overline{\text{D1}}$	Data Bus In/Out	36	$\overline{\text{D0}}$	Data Bus In/Out
	39	$\overline{\text{D3}}$		38	$\overline{\text{D2}}$	
	41	$\overline{\text{D5}}$		40	$\overline{\text{D4}}$	
	43	$\overline{\text{D7}}$		42	$\overline{\text{D6}}$	
DATA BUS 8-15	45	$\overline{\text{D9}}$	Not Used	44	$\overline{\text{D8}}$	Not Used
	47	$\overline{\text{D11}}$		46	$\overline{\text{D10}}$	
	49	$\overline{\text{D13}}$		48	$\overline{\text{D12}}$	
	51	$\overline{\text{D15}}$		50	$\overline{\text{D14}}$	
BUS CONTROL	53	$\overline{\text{WRP}}$	Write Pulse In	52	$\overline{\text{M}}$/IO	Memory/In
	55	R/$\overline{\text{W}}$	Read/Write In	54	$\overline{\text{OPREQ}}$	Operation Request In
	57	$\overline{\text{JMP CMD}}$	Not Used	56	$\overline{\text{HOLD}}$	Not Used
	59	$\overline{\text{RESET}}$	Not Used	58	$\overline{\text{RUN}}$	Not Used
				60	$\overline{\text{JMP ACK}}$	Not Used
INTERRUPT CONTROL	61	$\overline{\text{INTACK}}$	Interrupt Acknowledge	62	$\overline{\text{INT 0}}$	Interrupt Level 0
	63	$\overline{\text{INT 1}}$	Interrupt Level 1	64	$\overline{\qquad 2}$	" " 2
	65	$\overline{\qquad 3}$	" " 3	66	$\overline{\qquad 4}$	" " 4
	67	$\overline{\qquad 5}$	" " 5	68	$\overline{\qquad 6}$	" " 6
	69	$\overline{\qquad 7}$	" " 7	70	$\overline{\qquad 8}$	" " 8
	71	$\overline{\qquad 9}$	" " 9	72	$\overline{\qquad 10}$	" " 10
	73	$\overline{\qquad 11}$	" " 11	74	$\overline{\qquad 12}$	" " 12
	75	$\overline{\qquad 13}$	" " 13	76	$\overline{\text{INT 14}}$	" " 14
	77	$\overline{\text{INT 15}}$	" " 15			
DEBUG CONTROL	79		Not Used	78		Not Used
	81			80		
BUS CONTROL (Except J1-J2)	83	$\overline{\text{PAUSE}}$	Not Used	82	$\overline{\text{FETCH}}$	Not Used
	85	$\overline{\text{SENSE}}$		84	$\overline{\text{MAST RUN}}$	Master CPU Running In
	87	$\overline{\text{F.P. HOLD}}$		86	$\overline{\text{FLAG}}$	Not Used
	89	$\overline{\text{MST INT 3}}$		88	$\overline{\text{MST INTACK}}$	Not Used
	91	SPARE		90	SPARE	Not Used
				92	SPARE	Not Used
CLOCK (Except J1-J2)	93	SLV CLK	Not Used	94	I/O CLK	38.4 KHz I/O Clock
	95	2650 CLK	Not Used	96	SYS CLK	Not Used
GROUND	97	GND	Signal & Pwr GND	98	GND	Signal & Pwr GND
	99	GND	Signal & Pwr GND	100	GND	Signal & Pwr GND

FIGURE 7-11 GP I/O MOTHERBOARD (P1) PINOUTS

RS-232 INPUT-OUTPUT

P2 PIN	BULKHEAD CONNECTOR	SIGNAL	DESCRIPTION	COMMENTS
1	1	CHS GND	Not Used	
2	14	TTX -	TTY Current Loop Input -	RCVR IN
3	2	EIA RCV DATA	EIA Serial Input	RCVR IN
4	15	TTRDR +	Tape Reader Control Out +	47K Ω to +12V
5	3	EIA XMIT DATA	EIA Serial Output	EIA Driver
6	16	TTRDR -	Tape Reader Control Out -	Open Collector Driver
7	4	REQ TO SND	Request to Send	EIA Driver
8	17	TTRCV +	TTY Current Loop Output +	620 Ω to +12V
9	5	CLR TO SND	Clear to Send	EIA Receiver, 15K Ω to +12V
10	18	TTRCV -	TTY Current Loop Output -	Open Collector Driver
11	6	DATA SET RDY	Data Set Ready	EIA Receiver, 15K Ω to +12V
12	19	----		
13	7	SIG GND	EIA Ground	
14	20	DATA TERM RDY	Data Terminal Ready	EIA Driver
15	8	CARR DET	Carrier Detected	EIA Receiver, 15K Ω to +12V
16	21	----		
17	9	----		
18	22	RING IND	Ring Indicator	EIA Receiver, 15K Ω to +12V
19	10	----		
20	23	----		
21	11	ORIG	Originate Mode	EIA Driver
22	24	----		
23	12	LOCAL	Local Mode	EIA Driver
24	25	SND RESTRAINT	Send Restraint	EIA Driver
25	13	TTX +	TTY Current Loop Input +	620 Ω to +12V
26		NOT USED		

FIGURE 7-12a P2 AND REAR PANEL BULKHEAD CONNECTOR PIN ASSIGNMENT

PORTØ, PORT1, PARALLEL I/O

P3 PIN	SIGNAL	DESCRIPTION	P3 PIN	SIGNAL	DESCRIPTION
1	OUTØ-7	OUTPUT PORT Ø	21	INØ-7	INPUT PORT Ø
2	-6		22	-6	
3	-5		23	-5	
4	-4		24	-4	
5	-3		25	-3	
6	-2		26	-2	
7	-1		27	-1	
8	OUTØ-Ø		28	INØ-Ø	
9	OUT1-7	OUTPUT PORT 1	29	IN1-7	INPUT PORT 1
10	-6		30	-6	
11	-5		31	-5	
12	-4		32	-4	
13	-3		33	-3	
14	-2		34	-2	
15	-1		35	-1	
16	OUT1-Ø		36	IN1-Ø	
17	GND	SIGNAL GROUND	37	SPARE	SPARE
18	GND		38	SPARE	
19	GND		39	SPARE	
20	GND		40	SPARE	

FIGURE 7-12b P3 CONNECTOR PIN ASSIGNMENT

Chapter 8

1702A AND 82S115 PROM PROGRAMMER MODULES

8.0 INTRODUCTION

Two PROM programmer modules may be used in the system at any one time. This chapter describes the operation of the 82S115 bipolar PROM programmer and the 1702A MOS PROM programmer. These boards plug into the motherboard and are connected through flat cable to the three PROM programmer sockets mounted on the front panel of the development computer. Through appropriate software commands, an assembled program can be "burned" into the PROM for the user's prototype hardware.

Operation of the programmer modules is under control of the computer system and its associated programmer software. Once a PROM has been inserted in the appropriate socket and the PROM POWER switch has been turned on; the sequence of events is controlled by the operator interaction with the system. Commands available are WRITE PROM, READ PROM, and COMPARE PROM. These commands are all referenced to Slave Memory. The users assembled object code would normally be programmed into his PROMS. He can specify the memory limits and write into part or all of the PROM. The PROM can be read into memory and the values dumped on the console display or a disk file created for copying PROMS. The compare command allows verification of an unknown PROM by comparing the PROM contents to memory contents.

The program executed by the Master CPU controls operation in a byte by byte fashion. For a write operation the address and the data for each byte to be written is sent to the programmer. The programmer module executes a programming sequence for that byte and then informs the CPU through an interrupt and through the status word. For read operations the address is sent out by the CPU and the data is directly gated from the PROM to the CPU.

The 1702A PROM is electrically erasable by exposure to an ultraviolet source through its transparent cover. Erasing resets all bits to the zero state (output low). Programming is performed by selectively writing "1" s into the desired bit locations. Once programmed to a "1" state, the bit can be changed only by ultraviolet erasing.

The 82S115 family of PROMs are "fusible link" devices which have all bits initially set to logic "0". The PROM programmer "fuses" the links by applying high-voltage pulses setting the appropriate bits to logic "1". This is a non-reversible process, so care must be taken to ensure that the program to be burned into the PROM is correct. As with the 1072A programmer, the 82S115 family is programmed by applying address and data to the appropriate device pins and cycling through each address until the PROM is completely programmed. The basic difference in programming the two PROM types (besides non-reversible programming) is the fact that the 82S115 family requires serial programming of each bit within an 8 bit byte to avoid overstressing the device. This serial programming is provided by the hardware, allowing both programmers to appear very similar to the software.

8.1 GENERAL DESCRIPTION

A simplified block diagram applicable to both programmers is shown in figure 8-1. There are three distinct sections to each programmer consisting of the Data Handling Logic, the Programming Power Supplies, and the Write Voltage Sequencer.

The Data Handling section is almost identical for both programmers. Master CPU I/O write commands are recognized and decoded to provide strobes to the write data register and to the address and control registers. Data is received from the bus through the receiver portion of the transceiver to an internal output data bus which connects to the registers. Master CPU I/O read commands are recognized and decoded to provide multiplexing of PROM read data or write status to the internal input bus. This is buffered into the system bus by the driver portion of the transceiver. Master CPU I/O command formats are shown in figure 8-2.

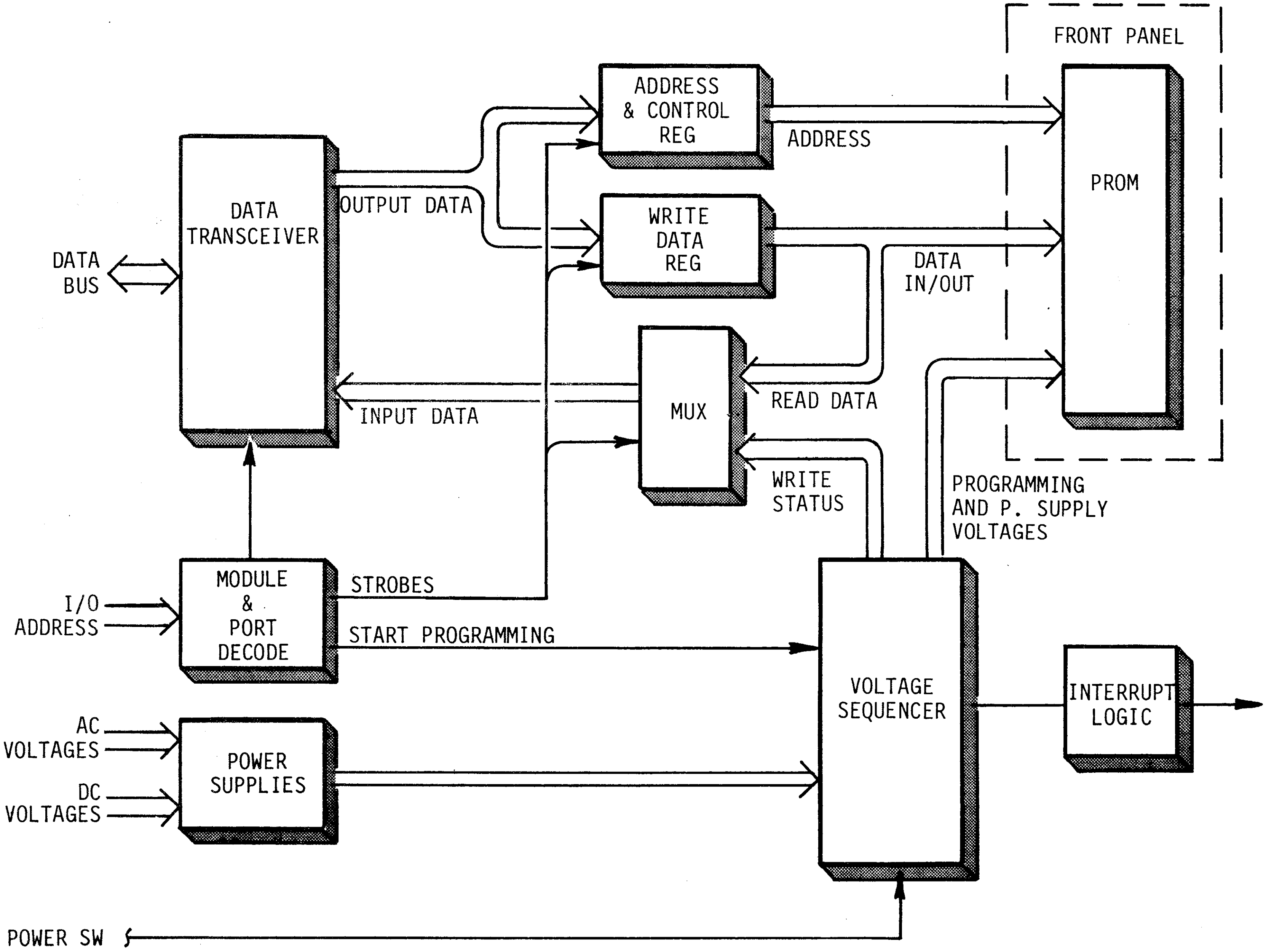
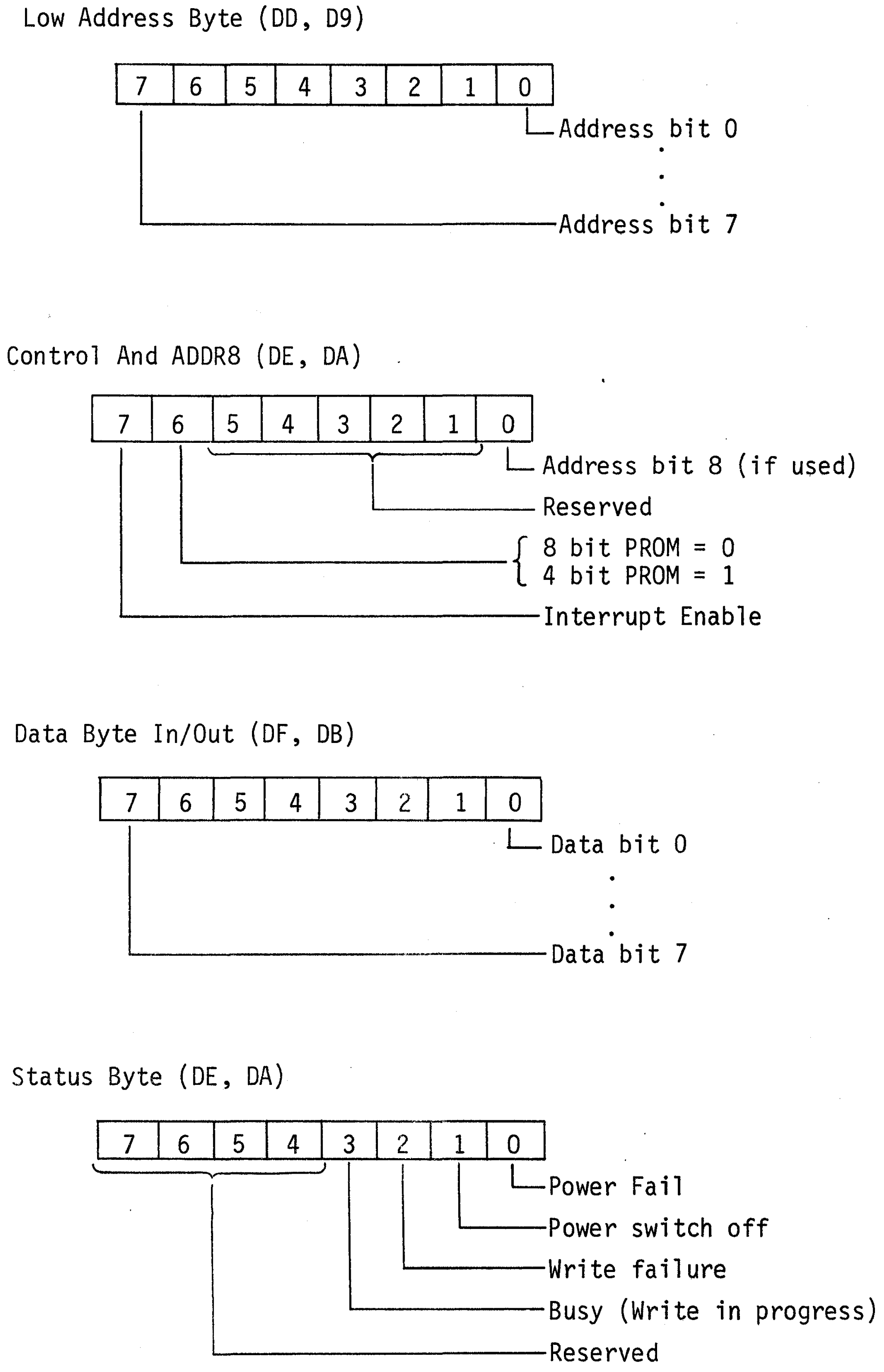


FIGURE 8-1 SIMPLIFIED PROM PROGRAMMER BLOCK DIAGRAM


Low Address Byte (DD, D9)

Control And ADDR8 (DE, DA)

Data Byte In/Out (DF, DB)

Status Byte (DE, DA)

FIGURE 8-2 COMMAND FORMATS

On board Power Supplies provide the higher voltages required for programming. AC is received, rectified, and then filtered. Series pass regulators operating on both DC bus inputs and the rectified AC provide controllable voltages for Write, Read, and Power Off functions. Many protection features such as OVP, current limit, and low voltage sense are included.

The Write Voltage Sequencer has its own clock and initiates a write sequence when each write data byte is received. Control lines and voltages are sequenced per manufacturer specifications for each PROM. Address complementing for the 1702A and serial data sequencing for 82S115 are also controlled. The end of the byte programming sequence triggers an interrupt flip-flop. Write status is available to the Master CPU at any time and consists of Busy, Power Fail, Power Switch Off, and Write Fail (82S115 only).

8.2 1702A DETAILED DESCRIPTION

8.2.1 Operation As previously discussed, the programmer software monitors and controls the programmer module in a byte by byte fashion. The programming sequence is as follows:

1) The operator inserts the 1702A PROM into the Front Panel Socket 1 insuring that the PROM power switch is off. This clamps voltages at the socket so no damage can occur.

2) The operator turns the PROM power switch on and enters a WRITE PROM command.

3) The software takes over and first reads the program status port to insure the power switch in on and that the correct module is installed.

4) The control byte is issued, enabling interrupts.

5) The address and data for the first byte is sent out.

6) The programmer writes the byte into the PROM which takes approximately 3 ms and signals completion with an interrupt.

7) The CPU reads the status byte to check for a power fail status.

8) The CPU reads the PROM data and compares it to the requested write data.

9) The 1702A requires that the programming operation be repeated five times the number of tries to the first successful write. If the PROM cannot be successfully written in 16 tries, it is assumed faulty. This repeated programming of the same byte is under software control.

10) When a programming sequence is successful, the software increments the address and programs the next location. This sequence continues until the software reaches the operator specified ending address, or the last PROM address (the default value).

8.2.2 Data Control Figure 8-3 is a detailed block diagram for the 1702A programmer. The data control section consists of a module and command decoder, output registers, and input data and status multiplexing.

The Module Decoder consists of gating to recognize an extended Master I/O command to DC- DF. A write decoder produces individual strobes for commands DD, DE, and DF to the low address, control and write data register respectively. The DF command also triggers the sequencer. For read operations,

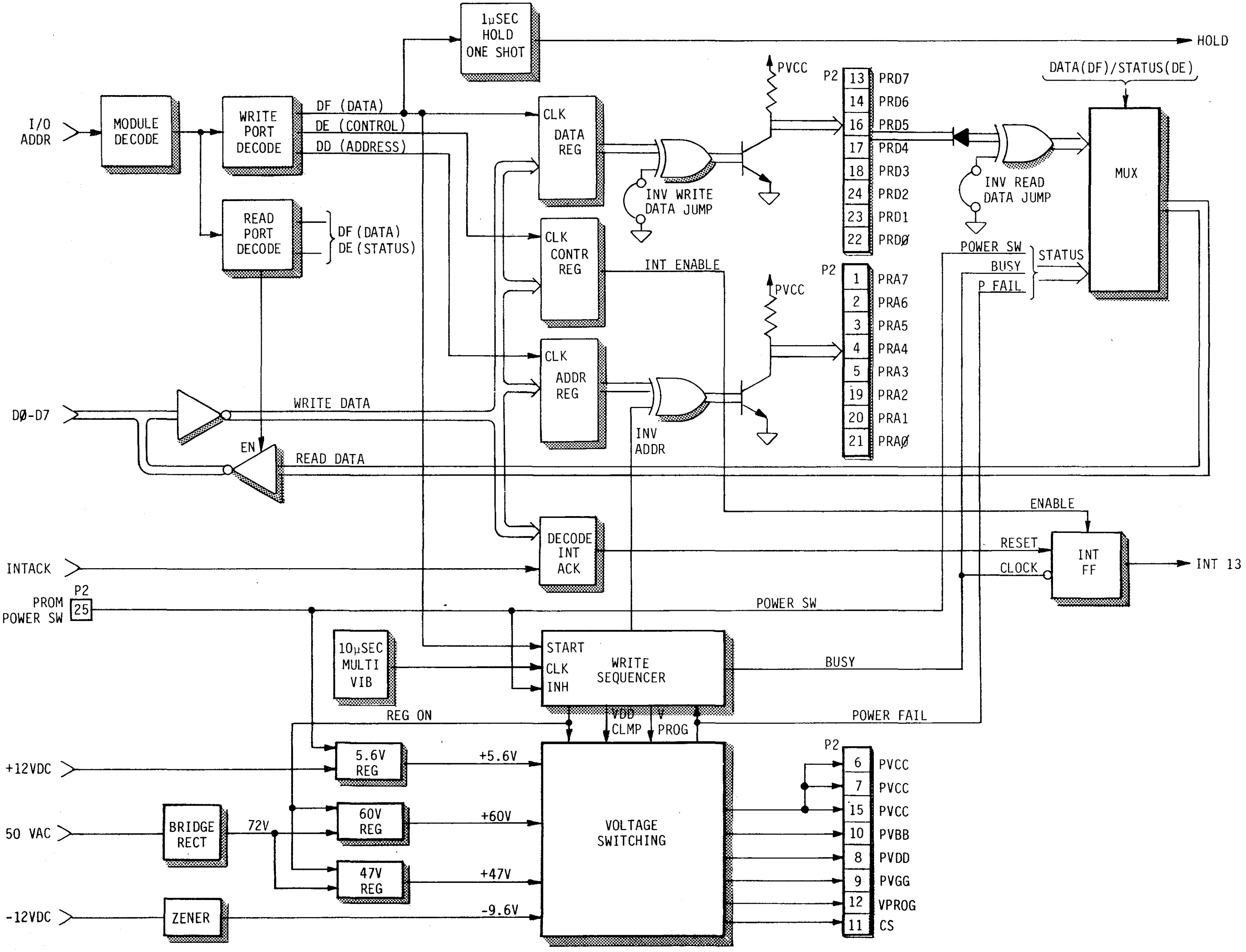


FIGURE 8-3 1702A PROM PROGRAMMER BLOCK DIAGRAM the module selected signal is further decoded to recognize a read to DE or DF. This enables the data drivers onto the system bus. Address bit 0, signifying a DE or DF command controls the MUX, routing PROM data or write status to the drivers.

The 8 bit write data register consists of a HEX register and two "D" connected flip-flops. Optional hardware data inversion is provided by exclusive OR gates. These feed NPN transistor drivers connecting through a flat cable to the data pins of the PROM socket. The discrete transistors are required to provide breakdown resistance to the +47V programming voltage. In order to disable these outputs during a read operation, the data register is reset and the exclusive OR gates are overriden to a zero output.

The address register is identical to the data register. An additional control line feeds the exclusive OR gates, however, providing the required address inversion during programming.

The control register is a single bit enabling or disabling the clocking of the interrupt flip-flop.

Read data lines from the PROM (same lines as write data) are received by exclusive OR gates. Series diodes are employed to protect the gates from the high-voltage levels applied during the programming cycle. The exclusive OR gates provide optional hardware read data inversion.

Status from the write sequencer and power supplies connects to one input of the data multiplexer and the PROM read data to the other. Status bits will be discussed under the sequencer section. The multiplexer connects to the driver section of the data bus transceivers which are enabled for a read status or read data command from the CPU.

8.2.3 Power Supplies The shaded portion of figure 8-4 illustrates the power supply section. Figure 8-5 shows this section in detail. Two high voltages, +47V and +60V, are produced for programming. 50 VAC is

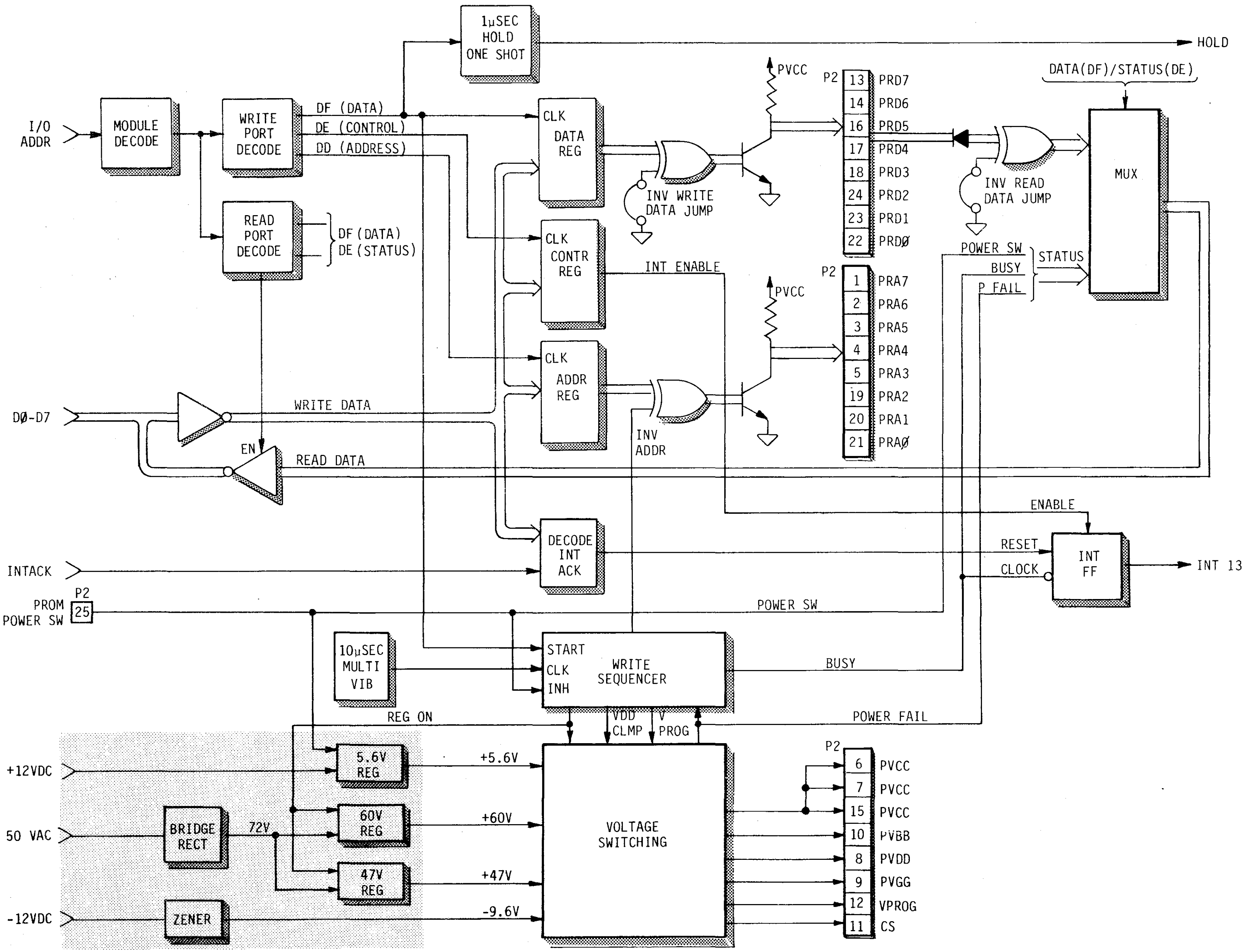


FIGURE 8-4 1702A PROM PROGRAMMER BLOCK DIAGRAM provided from a special winding in the system 5V power supply transformer. This is fused at 1 ampere on the rear panel and connected to the PROM module sockets, J1 and J2, on the motherboard. The 1702A Programming Module bridge rectifies, then filters this voltage producing approximately +80VDC. In one path a series pass regulator provides the +47V output. The other path has further filtering isolated by a diode to provide a separate source for the +60VDC series regulator. This regulator is directly referenced to the +47V by a 12V zener. Both regulators are individually current limited, the +47V at 300 milliamps and the +60V at 100 milliamps. Over voltage protection is provided by a Crowbar SCR which fires when the +47V exceeds +55V. The Crowbar is designed to blow the rear panel 1 ampere fuse, protecting the module and the PROM from further damage. The +47V regulator has a potentiometer for precise factory setup of the +47V output.

Two additional voltages are provided by regulators which utilize the available system DC. A 5.6V DC series regulator operates from the +12V DC line providing a controllable 5V source. -9V is provided to the PROM for read operation by a series zener on the -12V line.

Three sets of requirements are placed on the voltages to the PROM socket. When the PROM Power switch is in the Off position, the socket voltage must be zero. When reading the PROM, normal +5V and -9V voltages must be supplied. A third set of voltages with complex sequencing is required during programming. The sequencer and the PROM Power Switch control transistor voltage switches to meet these diverse requirements.

8.2.4 Sequencer and Voltage Switches Figure 8-5 details the components of the sequencer and voltage switches as well as the previously discussed power supply section. A 10 μs multivibrator drives a divider chain which is activated by receipt of a write data strobe. This chain provides all of the required timing. Figure 8-6 is a detailed timing diagram which should be referred to during the following discussion.

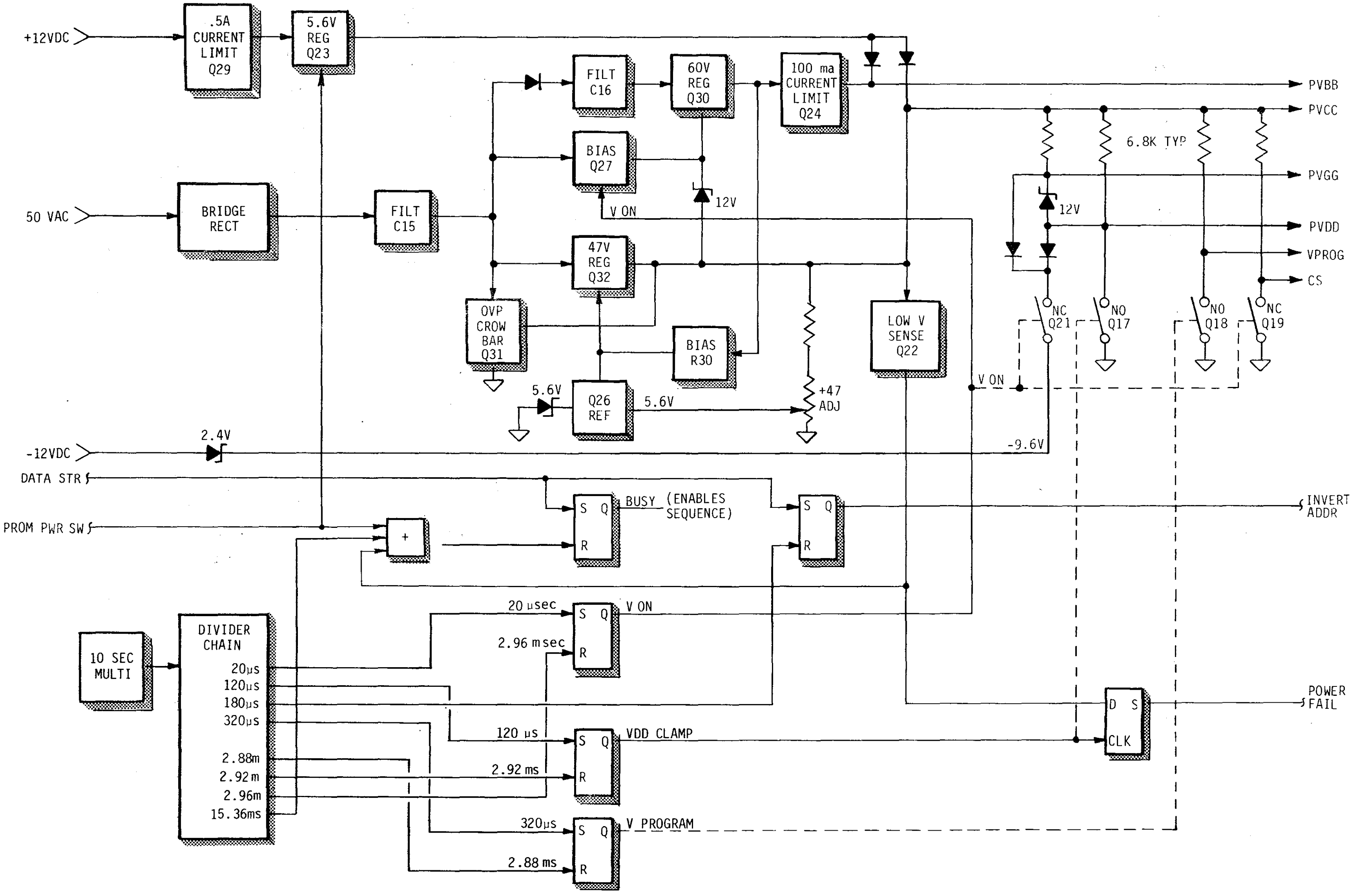


FIGURE 8-5 1702A PROM PROGRAMMER WRITE WAVEFORM GENERATOR DETAILED BLOCK DIAGRAM

The manufacturer specified operation is as follows:

1) Present the complement of the address (Logic 0/1=-47V/0V)

2) Force the data pins to the required value (Logic 0/1=-45V/0V)

3) Pulse VDD from 0V to -47V and VGG from 0V to -35V.

4) Present the true address.

5) Pulse the program line from 0V to -47V for three ms; then back to 0V.

6) Restore VDD and VGG back to 0V.

7) Reset for 12 ms (20% Duty cycle)

For convenience in programmer module design, all pins are shifted positively by +47V, thus enabling a simpler switching arrangement. The voltages are first shifted and allowed to stablize, then the recommended programming sequence is performed. Finally the voltages are restored to their original levels.

Referring to the timing diagram figure 8-6; the sequence is as follows:

1) The Write Data Strobe sets the INV ADDR flip-flop. Note that the PROM Power Switch must be on.

2) The BUSY flip-flop enables the divider chain, the other flip-flops, and the write data.

3) The VON flip-flop is clocked, which turns off the clamp on the 47V regulator. This turns on the +47V and +60V regulators. VBB goes to +60V. VGG, VDD, VCC, VPROG, CS and all data and address at logic 1 go to +47V.

4) The VDD CLAMP flip-flop is clocked which clamps VDD to 0V and VGG to +12V and turns off the -9V switch. This also enables the (+47V) POWER FAIL flip-flop. If the +47V is under +41V this flip-flop is set which resets the BUSY flip-flop and stops the sequencer.

5) The INV ADDR flip-flop is reset, restoring the exclusive OR gate address outputs to their true value.

6) The VPROG flip-flop is clocked on, clamping VPROG to 0V which starts the actual byte programming.

7) After 2.88 ms, the VPROG flip-flop is reset, turning off the VPROG clamp.

8) The VDD clamp flip-flop is reset, allowing VDD and VGG to go back to +47V.

9) The VON flip-flop is reset, shifting all voltages back to their read values. Note that VDD and VGG are switched to -9V. VCC and VBB are not allowed below +5V.

10) The BUSY flip-flop is reset after 15.36 ms. This provides the required programming duty cycle.

11) The trailing edge of BUSY triggers the INTERRUPT flip-flop, notifying the Master CPU.

12) The Master CPU acknowledges the interrupt with a vector on the data bus and INTACK, resetting the INTERRUPT flip-flop.

Status consisting of BUSY, POWER FAIL, and POWER SWITCH status are always available, but are normally read after each programming operation. The PROM data cannot be read during a Write operation since the Write register is forced into the PROM data output.

8.3 82S115 DETAILED DESCRIPTION

8.3.1 Operation The programming sequence under direct software control is as follows:

1) The operator inserts the 82S115 PROM into the front panel socket PROM 2 with the PROM POWER switch Off.

2) The operator turns the PROM POWER switch on and enters a WRITE PROM command on the console.

3) The status port is read to check power switch status.

4) The address to be programmed is sent out. This takes two bytes since the 82S115 has nine bits. The second byte includes the interrupt enable which is set on.

5) The data to be programmed for that address is sent out. This starts the write sequence.

6) The programmer hardware takes over and writes the data byte serially into the PROM. This takes 20.5 ms which includes a rest period of 10.25 ms.

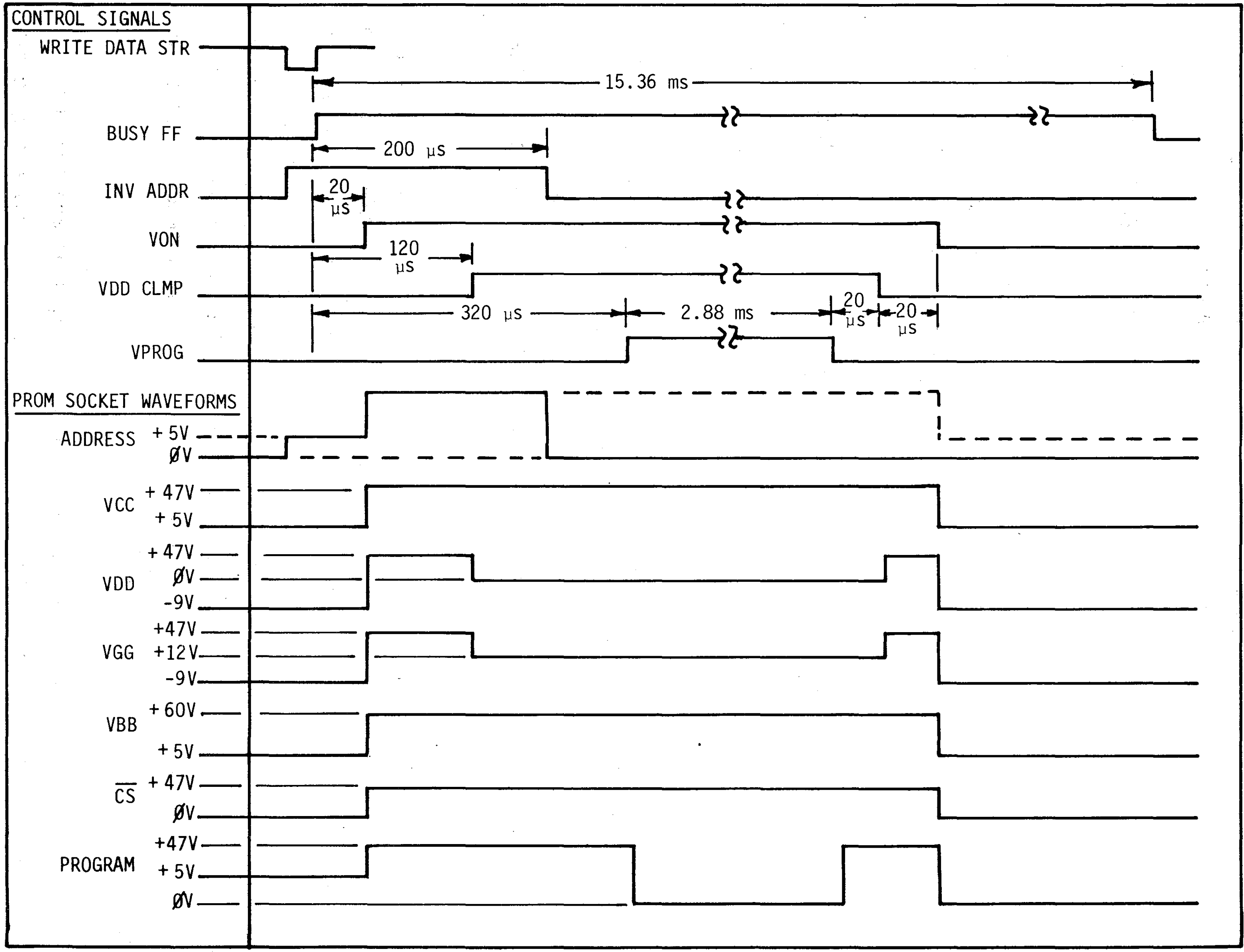


FIGURE 8-6 1702A SEQUENCER TIMING

7) After the hardware verifies the data at high and low VCC values, the CPU is interrupted.

8) The CPU reads the status byte for POWER FAIL or WRITE FAIL.

9) The CPU does a redundant verification of data. The 82S115 should program in one try, but the software is designed to try 8 times before reporting failure.

10) When a program sequence is successful, the software increments the address and programs the next location. This sequence continues until the software reaches the operator-specified ending address or the last PROM address.

8.3.2 Data Control Figure 8-7 is a detailed block diagram of the 82S115 programmer module. The data control section consists of a module and command decoder, output register, and input data and status multiplexing. In addition, a data commutator and data comparator are controlled by the write sequencer.

The module decoder consists of gating to recognize an extended Master I/O command to D8-DB. A write decoder produces individual strobes for commands D9, DA, and DB to the low address, control and write data register respectively. The DB command also triggers the write sequencer. For read operations, the module selected signal is further decoded to recognize a read to DA or DB, which enables the data driver onto the system bus. Address bit 0, signifying a DA or DB command controls the multiplexer which routes PROM data or write data to the drivers.

The 8 bit write data register consists of a HEX register and two "D" connected flip-flops. Optional hardware data inversion is provided by exclusive

OR gate. These feed commutator gates controlled by the sequencer which enables bits 0 through 7 sequentially through PNP switches onto the PROM data output pins.

The PNP switches allow +17V at 200 milliamps to those data cells to be programmed to a Logic 1.

The low address register (D9) consists of a HEX register and two D flip-flops. Address bit 9 is fed from a HEX register containing 6 bits of the control byte (DA). Exclusive OR gates provide jumperable hardware inversion followed by open collector gates which drive the PROM address pins directly.

Data input from the PROM is through comparators which have a 2.5V threshold level. These provide the dual function of a precise comparison level and protection against the +17V during programming. Exclusive OR gates follow the comparators to provide hardware jumperable data inversion. Status from the write sequencer or the PROM data input is selected by a 2 input multiplexor which feeds the data bus transceiver.

8.3.3 Power Supplies Figure 8-8 shows both the power supplies and the write sequencer section in detail. Power supplies consist of the +17V and the VCC supply.

Center tapped 60VAC is full wave rectified and capacitively filtered to provide approximately +42VDC unregulated. This feeds a 200 MA current source which utilizes a 5V drop across a resistor compared to a 5V zener to maintain 200 MA maximum. The current source stage feeds a 17V series regulator which is controlled by the sequencer. A RC on the +17V clamp circuit provides the required 20 μs rise time on turn on. An over voltage circuit triggers a SCR Crowbar if the voltage exceeds 20.6V.

The VCC supply is a series regulator fed by the +12V bus power. This supplies 5.0V for the 82S115, 8.75V for the four-bit, 16 pin PROMS in the same family, and 4.4V and 5.6V required during verification. A combination of zeners and diodes provide the reference voltage. Open collector gates clamp various points of the reference for each requirement. The current limit consists of a PNP transistor comparing the voltage drop across a diode and a 1.5 ohm resistor in the 12V line. This provides a limit of 400

8.3.4 Sequencer and Voltage Switches Figure 8-8 details the sequencer components. The write data strobe starts the sequencer operation if the PROM power switch is ON. The timing diagram is given in figure 8-9.

Programming for each bit follows an identical sequence. Fuse Enable 1 is turned on, then +17V applied to the data bit if it is to be programmed to a Logic 1. Fuse Enable 2 starts the actual bit programming and is held for 1.24 μs; then the sequence is reversed. Fuse Enable 2 is turned off; the 17V turned off; then the PROM data out is compared to the write data for VCC high (4.4V) and VCC low (5.6V) and the result stored in the write fail bit. A rest period of 10.24 μs with VCC off follows. For the four bit PROMS the rest period is 15.36 ms since only four bits are programmed.

Basic timing is provided by a 2.5 μs multivibrator feeding a divider chain. For commutating the data bits sequentially, a divide by 16 counter clocked at a 1.28 μs rate, and an octal decoder provide enable lines to the individual data gates. Thus each bit takes 1.28 μs to program. Discrete flip-flops provide a bidirectional shift register for sequencing and de-sequencing FE1, +17V and FE2.

Detailed timing is as follows. The DATA STROBE triggers the BUSY flip-flop enabling the divider chain and counter. Note that BUSY holds all flip-flops and counters in their initial state. After 5 μs, the VCC flip-flop is clocked on enabling either 5.0V or 8.76V to the PROM. (5.0V is normally present with the power switch on). After 20 μs, the FE1 flip-flop is

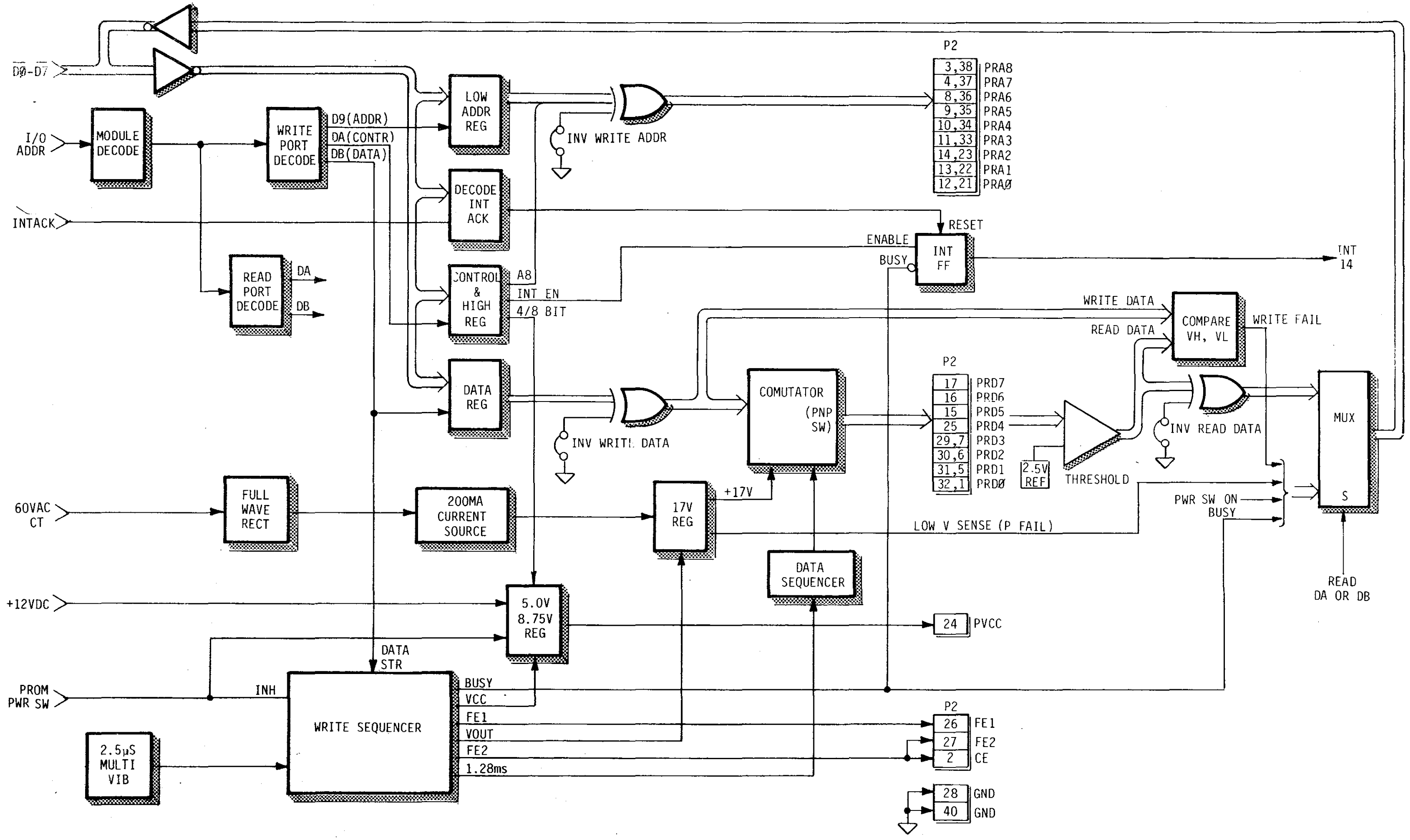


FIGURE 8-7 82S115 PROMGRAMMER
BLOCK DIAGRAM

clocked. The VOUT flip-flop enables the +17V regulator with a rise time of 20 μs. Note that the commutator enables the data bit 0 PNP switch if the write data requested is a Logic 1. FE2 flip-flop is clocked 40 μs later providing adequate settling time before fusing. The FE2 flip-flop clocks the POWER FAIL flip-flop which samples the +17V and VCC low voltage sense circuitry. If a power fail occurs, the sequence is terminated. The FE2 programming pulse lasts for 1.24 ms. Then in 20 μs clock periods, FE2, +17V, and FE1 are removed. Note that the four bit PROMS use only $\overline{\text{CE}}$ during the FE1 period. Following the de-sequencing the four bit counter is clocked producing an enable to data bit 1. The above sequence is repeated for data bits 1 through 7. (1-3 for four bit PROMS).

After data bit 7 is programmed, the VERIFY flip-flops come into operation. VCC is first switched to 4.4V and the PROM read data and the programmer write data register compared. This is stored in the WRITE FAIL flip-flop if the comparison fails. Then VCC is switched to 5.6V and the comparison again clocked to the WRITE FAIL flip-flop. This bit is available in the status word.

After verification, a rest period of 10.24 ms follows with VCC off. BUSY is then reset, triggering the INTERRUPT flip-flop and signaling the end of the programming sequence. The Master CPU acknowledges the interrupt with a vector on the data bus and INTACK, resetting the INTERRUPT flip-flop.

Status consisting of BUSY, POWER FAIL, WRITE FAIL, and POWER SWITCH STATUS is always available but usually read only after an interrupt. The PROM data cannot be read during a WRITE operation since the output bits are being forced by the commutator. However, the Serial Write Data can be read, providing loop-back for the PROM Programmer Diagnostic.

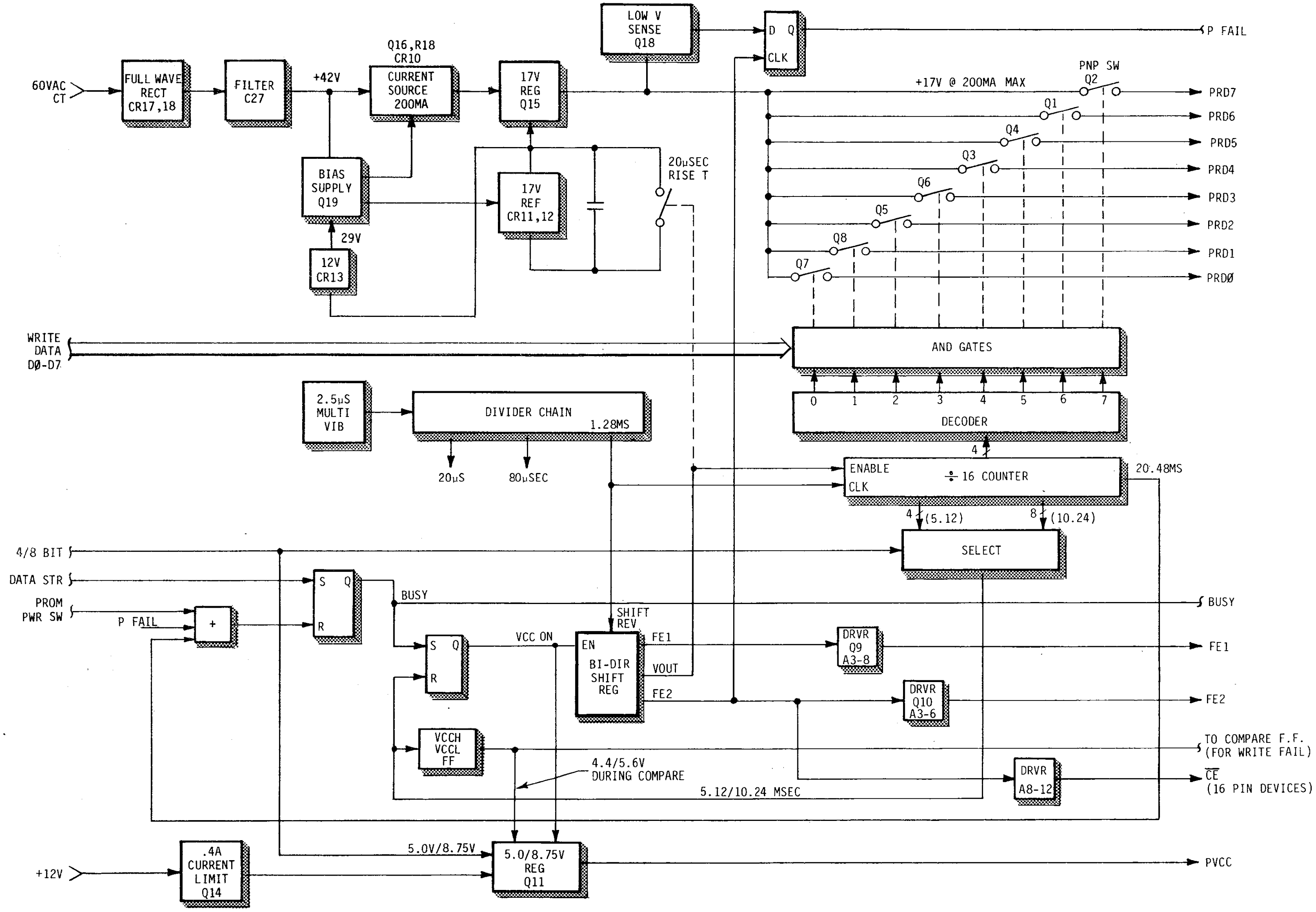


FIGURE 8-8   82S115 PROM PROGRAMMER WRITE WAVEFORM GENERATOR

8.4 UTILIZATION: 1702A

8.4.1 Installation The module is designed to be plugged into the card file of the development computer, but may be used in other applications. Jumper and switch options are discussed in section 8.4.2. All interface connections and connector pinouts are detailed in section 8.4.3.

COMPUTER INSTALLATION The 1702A PROM Programmer Module plugs into one of two special programmer slots in the motherboard (J1, J2). These slots are mechanically keyed preventing other computer modules from being accidentally plugged in. The card connector is offset, insuring that it cannot be plugged in backwards. All power requirements (+5V, ±12V, and 50 VAC) are supplied through the motherboard. Cable 90014061, the flat ribbon 26 pin PROM socket cable, attaches to the top edge connector P2. This cable connects directly to front panel socket PROM 1. Care should be taken that pin 1 of the cable (Red Stripe) and pin 1 of the edge connector align. Pin 1 is to the left when viewed from the component side of the board.

INSTALLATION IN OTHER APPLICATIONS When using the module in other applications, the user must take into consideration environmental extremes, mounting, power requirements, interface connections and signal AC and DC requirements. Environmental limits, power requirements and card outline dimensions are listed in figure 8-10.

For mounting, the board is designed to plug into a 100 pin edge card connector, CDC VPB-01C5-0D00A1 or equivalent. Card guide slots should be provided and the card should be mounted vertically or in some way clamped so that it does not vibrate out of its socket. Vertical mounting is also important to insure convective cooling if no fan is used. The card dissipates approximately 10 watts, therefore adequate space should be allowed between modules for air flow. It is desirable to have forced air flow between all modules if several modules are utilized within a confined area.

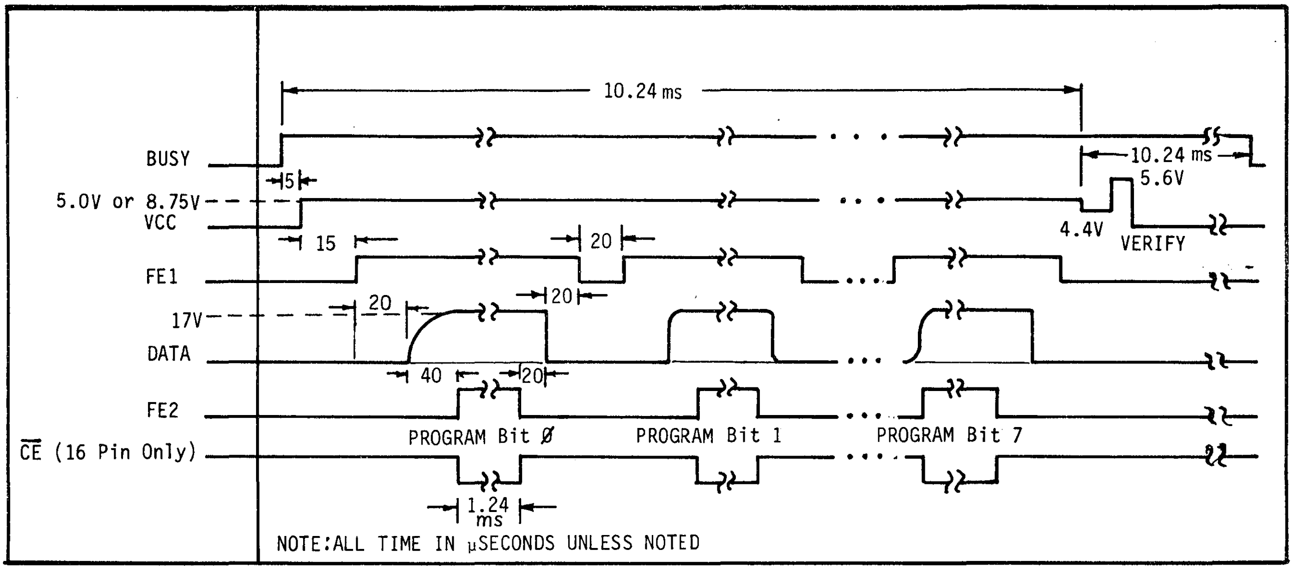


FIGURE 8-9 82S115 WRITE SEQUENCER TIMING DIAGRAM

ENVIRONMENTAL

Air Temp 0° - 70°C Operational
Surrounding Module

Storage Temp -55° - 125°C

Humidity 90% Non-condensing

POWER

	TYP	MAX
+5VDC	.9A	1.2A
+12VDC	60ma	90ma
-12VDC	40ma	70ma
50VAC		.3A PEAK

PHYSICAL OUTLINE

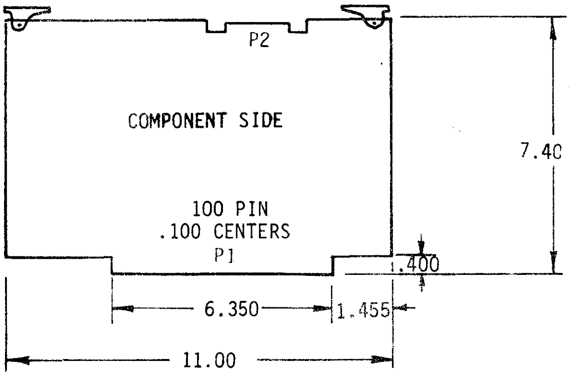


FIGURE 8-10 1702A PROM PROGRAMMER ENVIRONMENTAL, POWER AND PHYSICAL REQUIREMENTS

Interface connections and signal requirements are given in section 8.4.3.

8.4.2 Switch and Jumper Options The only jumper options are for hardware data and address inversion. If for some reason the card is used with other programming software that does not provide correct address or data polarity, the module may be configured to invert Read data, Write data, or address. Addition of J4 (Between C5 and C6 provides Read data inversion Cutting J1 and adding J2 (below TP4) provides address inversion. Addition of J3 (Under B1) provides Write data inversion.

8.4.3 Interface and Connector Pinouts NOUTS A complete interconnection diagram for 1702A PROM Programmer use within the system is given in figure 8-11. Figure 8-12 lists the motherboard connector (P1) pinouts and signal definitions. Figures 8-13 lists P2 pinouts to the front panel PROM socket.

8.5 UTILIZATION: 82S115

8.5.1 Installation The module is designed to be plugged into the card file of the development computer, but may be used in other applications. Jumper and switch options are discussed in section 8.5.2. All interface connections and connector pinouts are detailed in section 8.5.3.

COMPUTER INSTALLATION The 82S115 PROM Programmer Module plugs into any one of two special programmer slots in the motherboard (J1-J2). For cabling convenience, the slots are mechanically keyed to prevent other modules plugging into the AC. The card connector is offset, insuring that it cannot be plugged in backwards. All power requirements (+5V, +12V, 60 VAC) are supplied through the motherboard. Cable 90014071, the flat ribbon 40 pin PROM socket cable attaches to the top edge connector P2. The 40 pin cable attaches to both the 24 pin PROM 2 socket and the 16 pin PROM 3 socket. Care should be taken that pin 1 of the cable (Red Stripe) and pin 1 of the edge connector align. Pin 1 is to the left when viewed from the component side of the board.

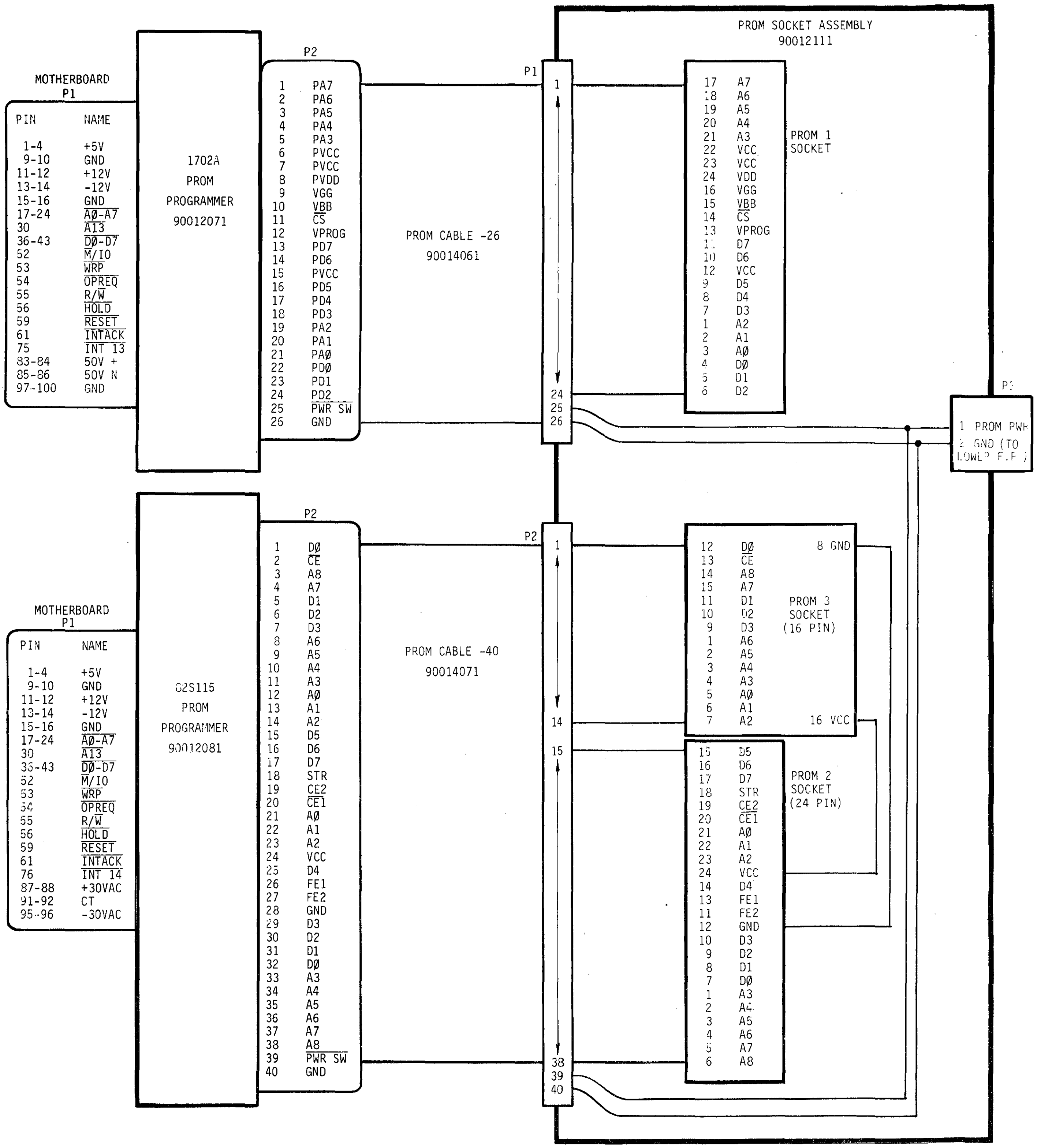


FIGURE 8-11 PROM PROGRAMMER INTERCONNECT

INSTALLATION IN OTHER APPLICATIONS When using the module in other applications, the user must take into consideration environmental extremes, mounting, power requirements, interface connections and signal AC and DC requirements. Environmental limits, power requirements and card outline dimensions are listed in figure 8-14.

For mounting, the board is designed to plug into a 100 pin edge card connector, CDC VPB-01C5-0D00A1 or equivalent. Card guide slots should be provided and the card should be mounted vertically or in some way clamped so that it does not vibrate out of its socket. Vertical mounting is also important to insure convective cooling if no fan is used. The card dissipates approximately 10 watts, therefore adequate space should be allowed between modules for air flow. It is desirable to have forced air flow between all modules if several modules are tuilized within a confined area.

Interface connections and signal requirements are given in section 8.5.3.

8.5.2 Switch and Jumper Options The only user jumper options are for hardware data and address inversion. If for some reason, the card is used with other programming software that does not provide correct address or data polarity, the module may be configured to invert Read data, Write data, or address. Cutting J3 (Between C4 and C5) provides Read data inversion. Addition of J4 (above D3 & D4) provides Write data inversion. Adding J1 (below A7) provides address inversion.

8.5.3 Interface and Connector Pinouts A complete interconnection diagram for PROM Programmer use within the system is given in figure 8-11. Figure 8-15 lists the motherboard connector (P1) pinouts and signal definitions. Figure 8-16 lists the pinouts for top edge connector P2.

	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	+5V	+5VDC In	2	+5V	+5VDC In
	3	+5V	+5VDC In	4	+5V	+5VDC In
	5	AUX BUS		6	AUX BUS	
	7	AUX BUS		8	AUX BUS	
	9	GND	Signal & Pwr GND	10	GND	Signal & Pwr GND
	11	+12V	+12VDC In	12	+12V	+12VDC In
	13	-12V	-12VDC In	14	-12V	-12VDC In
	15	GND	Signal & Pwr GND	16	GND	Signal & Pwr GND
ADDRESS BUS	17	$\overline{A0}$		18	$\overline{A1}$	
	19	$\overline{A2}$	Address Bus	20	$\overline{A3}$	Address Bus
	21	$\overline{A4}$	In	22	$\overline{A5}$	In
	23	$\overline{A6}$		24	$\overline{A7}$	
	25	$\overline{A8}$		26	$\overline{A9}$	
	27	$\overline{A10}$	Not Used	28	$\overline{A11}$	Extended I/O
	29	$\overline{A12}$		30	$\overline{A13}$	In
	31	$\overline{A14}$		32	$\overline{A15}$	
MEMORY CONTROL	33	$\overline{CMEM}$		34	$\overline{RAM\ INH}$	Not Used
	35	$\overline{WD\ ACCESS}$				
DATA BUS 0-7	37	$\overline{D1}$		36	$\overline{D0}$	
	39	$\overline{D3}$	Data Bus	38	$\overline{D2}$	Data Bus
	41	$\overline{D5}$	In/Out	40	$\overline{D4}$	In/Out
	43	$\overline{D7}$		42	$\overline{D6}$	
DATA BUS 8-15	45	$\overline{D9}$		44	$\overline{D8}$	
	47	$\overline{D11}$	Not Used	46	$\overline{D10}$	Not Used
	49	$\overline{D13}$		48	$\overline{D12}$	
	51	$\overline{D15}$		50	$\overline{D14}$	
BUS CONTROL	53	$\overline{WRP}$	Write Pulse In	52	$\overline{M}/IO$	Memory/In
	55	$R/\overline{W}$	Read/$\overline{\text{Write}}$ In	54	$\overline{OPREQ}$	Operation Request
	57	$\overline{JMP\ CMD}$		56	$\overline{HOLD}$	Hold CPU Out
	59	$\overline{RESET}$		58	$\overline{RUN}$	
				60	$\overline{JMP\ ACK}$	
SPLIT BUS	61	$\overline{MSTR\ INTACK}$	Master Interrupt Ack	62		
	63			64		
	65			66		
	67			68		
	69			70		
	71			72		
	73			74		
	75	$\overline{MSTR\ INT\ 13}$	Master Interrupt 13	76		
	77			78		
	79			80		
	81			82		
PROM PROGRAMMER AC	83	50V +	50VAC	84	50V +	50VAC
	85	50V N		86	50V N	
	87	+30VAC		88	+30VAC	
	89	+15VAC		90	+15VAC	
	91	CT	Not Used	92	CT	Not Used
	93	-15VAC		94	-15VAC	
	95	-30BAC		96	-30VAC	
GROUND	97	GND	Signal & Pwr GND	98	GND	Signal & Pwr GND
	99	GND	Signal & Pwr GND	100	GND	Signal & Pwr GND

FIGURE 8-12 1702A MOTHERBOARD (P1) PINOUTS

PIN	SIGNAL	FUNCTION	PIN	SIGNAL	FUNCTION
1	PA7	PROM 1 ADDRESS 7	2	PA6	PROM 1 ADDRESS 6
3	PA5	PROM 1 ADDRESS 5	4	PA4	PROM 1 ADDRESS 4
5	PA3	PROM 1 ADDRESS 3	6	PVCC	PROM 1 VCC
7	PVCC	PROM 1 VCC	8	PVDD	PROM 1 VDD
9	PVGG	PROM 1 VGG	10	VBB	PROM 1 VBB
11	$\overline{\text{CS}}$	CHIP SELECT PROM 1	12	VPROG	PROM 1 PROGRAM PIN
13	PD7	PROM 1 DATA 7	14	PD6	PROM 1 DATA 6
15	PVCC	PROM 1 VCC	16	PD5	PROM 1 DATA 5
17	PD4	PROM 1 DATA 4	18	PD3	PROM 1 DATA 3
19	PA2	PROM 1 ADDRESS 2	20	PA1	PROM 1 ADDRESS 1
21	PAØ	PROM 1 ADDRESS 0	22	PDØ	PROM 1 DATA Ø
23	PD1	PROM 1 DATA 1	24	PD2	1 2
25	$\overline{\text{PWR SW}}$	PROM POWER SWITCH IND.	26	GND	POWER SWITCH RETURN

FIGURE 8-13 1702A P2 PINOUTS

ENVIRONMENTAL

Air Temp 0° - 70°C Operational
Surrounding Module

Storage Temp -55° - 125°C

Humidity 90% Non-condensing

POWER

	TYP	MAX
+5VDC	.9A	1.2A
+12VDC	.15A	.20A
60VAC		.3A PEAK

PHYSICAL OUTLINE

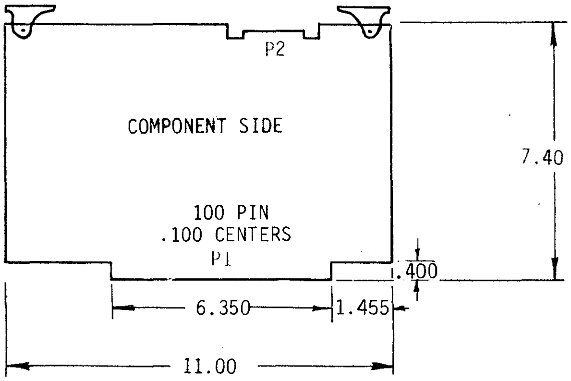


FIGURE 8-14 82S115 PROM PROGRAMMER ENVIRONMENTAL, POWER AND PHYSICAL REQUIREMENTS

	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	+5V	+5VDC In	2	+5V	+5VDC In
	3	+5V	+5VDC In	4	+5V	+5VDC In
	5	AUX BUS	Not Used	6	AUX BUS	Not Used
	7	AUX BUS	Not Used	8	AUX BUS	Not Used
	9	GND	Signal & Pwr GND	10	GND	Signal & Pwr GND
	11	+12V	+12VDC In	12	+12V	+12VDC In
	13	-12V	-12VDC In	14	-12V	-12VDC In
	15	GND	Signal & Pwr GND	16	GND	Signal & Pwr GND
ADDRESS BUS	17	$\overline{\text{A0}}$	Address Bus In	18	$\overline{\text{A1}}$	Address Bus In
	19	$\overline{\text{A2}}$		20	$\overline{\text{A3}}$	
	21	$\overline{\text{A4}}$		22	$\overline{\text{A5}}$	
	23	$\overline{\text{A6}}$		24	$\overline{\text{A7}}$	
	25	$\overline{\text{A8}}$	Not Used	26	$\overline{\text{A9}}$	Not Used
	27	$\overline{\text{A10}}$		28	$\overline{\text{A11}}$	Not Used
	29	$\overline{\text{A12}}$		30	$\overline{\text{A13}}$	Extended I/O In
	31	$\overline{\text{A14}}$		32	$\overline{\text{A15}}$	Not Used
MEMORY CONTROL	33	$\overline{\text{CMEM}}$	Not Used	34	$\overline{\text{RAM INH}}$	Not Used
	35	$\overline{\text{WD ACCESS}}$	Not Used			
DATA BUS 0-7	37	$\overline{\text{D1}}$	Data Bus In/Out	36	$\overline{\text{D0}}$	Data Bus In/Out
	39	$\overline{\text{D3}}$		38	$\overline{\text{D2}}$	
	41	$\overline{\text{D5}}$		40	$\overline{\text{D4}}$	
	43	$\overline{\text{D7}}$		42	$\overline{\text{D6}}$	
DATA BUS 8-15	45	$\overline{\text{D9}}$	Not Used	44	$\overline{\text{D8}}$	Not Used
	47	$\overline{\text{D11}}$		46	$\overline{\text{D10}}$	
	49	$\overline{\text{D13}}$		48	$\overline{\text{D12}}$	
	51	$\overline{\text{D15}}$		50	$\overline{\text{D14}}$	
BUS CONTROL				52	$\overline{\text{M}}\text{/IO}$	Memory/IO In
	53	$\overline{\text{WRP}}$	Write Pulse In	54	$\overline{\text{OPREQ}}$	Operation Request In
	55	R/$\overline{\text{W}}$	Read/Write In	56	$\overline{\text{HOLD}}$	Hold CPU Out
	57	$\overline{\text{JMP CMD}}$	Not Used	58	$\overline{\text{RUN}}$	Not Used
	59	$\overline{\text{RESET}}$	Not Used	60	$\overline{\text{JMP ACK}}$	Not Used
SPLIT BUS	61	$\overline{\text{MSTR INTACK}}$	Master Interrupt Ack	62		
	63			64		
	65			66		
	67			68		
	69			70		
	71			72		
	73			74		
	75			76	$\overline{\text{MSTR INT 14}}$	Master Interrupt 14
	77			78		
	79			80		
	81			82		
PROM PROGRAMMER AC	83	50V+	Not Used	84	50V +	Not Used
	85	50V N	Not Used	86	50V N	Not Used
	87	+30VAC	30VAC +	88	+30VAC	30VAC +
	89	+15VAC	Not Used	90	+15VAC	Not Used
	91	CT	Center Tap	92	CT	Center Tap
	93	-15VAC	Not Used	94	-15VAC	Not Used
	95	-30VAC	30VAC -	96	-30VAC	30VAC -
GROUND	97	GND	Signal & Pwr GND	98	GND	Signal & Pwr GND
	99	GND	Signal & Pwr GND	100	GND	Signal & Pwr GND

FIGURE 8-15 82S115 P2 PINOUTS

PIN	SIGNAL	FUNCTION	PIN	SIGNAL	FUNCTION
1	DØ	PROM 3 Data 0	2	$\overline{\text{CE}}$	PROM 3 Chip Enable
3	A8	" " Address 8	4	A7	" " Address 7
5	D1	" " Data 1	6	D2	" " Data 2
7	D3	" " Data 3	8	A6	" " Address 6
9	A5	" " Address 5	10	A4	" " Address 4
11	A3	" " Address 3	12	A0	" " Address 0
13	A1	" " Address 1	14	A2	" " Address 2
15	D5	PROM 2 Data 5	16	D6	PROM 2 Data 6
17	D8	" " Data 7	18	STR	" " Strobe
19	CE2	" " Chip Enable 2	20	CE1	" " Chip Enable 1
21	A0	" " Address 0	22	A1	" " Address 2
23	A2	" " Address 2	24	VCC	PROM 2 & 3 VCC
25	D4	" " Data 4	26	FE1	PROM 2 Fuse Enable 1
27	FE2	" " Fuse Enable 2	28	GND	PROM 2 & 3 Ground
29	D3	" " Data 3	30	D2	PROM 2 Data 2
31	D1	" " Data 1	32	D0	" " Data 0
33	A3	" " Address 3	34	A4	" " Address 4
35	A5	" " Address 7	36	A6	" " Address 6
37	A7	Power Switch Ind.	38	A8	" " Address 8
39	$\overline{\text{PWR SW}}$		40	GND	Ground

FIGURE 8-16 82S115 P2 PINOUTS

Chapter 9
SYSTEM BUS

9.0 GENERAL DESCRIPTION

The computer bus ties all modules together providing signal connections between modules, as well as supplying all power requirements. The bus is essentially universal with DC Power, Data, Address and Control lines going to all modules. Additionally, due to the Master-Slave architecture of the computer, the bus is split roughly in halves with special control lines and interrupt lines for the Master and Slaves halves. The Debug module (J9) provides the dividing line and interacts with control lines on both halves. The PROM Programmer modules are restricted to J1 and J2 since special AC voltages are bussed to these slots only.

Figure 9-1 illustrates the bus structures with groups of bus lines labeled. Figure 9-2 lists the signals within each group. 8 levels of Master and Slave interrupt lines are split by the Debug Module and use the same pin assignment on each half. This allows use of GPI/O cards in both Master and Slave halves. Special Slave Debug lines run opposite the remaining 8 Master interrupt lines connecting the Debug module to the Slave modules. Master Debug lines connect the Debug module and Master CPU. Opposite these lines the Slave bus interconnects all Slave modules and the Hardware Analyzer Module.

The bus is physically contained on a printed circuit motherboard. A heavy duty terminal strip is used to connect DC while a Molex connector provides PROM AC voltages. The edge card connectors contain 100 pins. A ground plane covers most of the circuit side of the board and heavy +5V and ground buses carry power to the modules. Protection from accidental plug in of other modules in the PROM Programmer section is provided by a substantial sheet-metal key.

9.1 BUS SIGNAL DEFINITIONS

The following signal definitions are split into four groups: Common bus signals, unique Master side bus signals, unique Slave side bus signals and unique PROM Programmer section signals.

COMMON BUS SIGNALS

$\overline{\text{AØ}}$-$\overline{\text{A15}}$

16 Address Lines; used to transmit the address of the memory location or I/O port to be accessed. A15 is the most significant bit.

$\overline{\text{DØ}}$-$\overline{\text{D15}}$

16 Bidirectional Data lines; used to transmit and receive information to and from a memory location or I/O port. D15 is the most significant bit. Eight-bit processors utilize only DØ-D7.

CMEM/($\overline{\text{MST}}$ $\overline{\text{MEM}}$)

SLAVE/$\overline{\text{MASTER}}$ MEMORY ACCESS; used only by the Master CPU to select Master or Slave memory.

$\overline{\text{RAM}}$ $\overline{\text{INH}}$

RAM MEMORY INHIBIT; normally used by PROM memory circuıtry to disable RAM that occupies the same address space.

$\overline{\text{WD}}$ $\overline{\text{ACCESS}}$

WORD ACCESS; control line from 16-bit Slave to Slave memory, forcing it to operate in a 16-bit mode.

$\overline{\text{M}}$/IO

$\overline{\text{Memory}}$/IO control line enables either memory or I/O devices.

$\overline{\text{WRP}}$

WRITE PULSE used by Memory and I/O devices to strobe Write Data.

$\overline{\text{OPREQ}}$

OPERATION REQUEST is the synchronizing command for all bus transactions signifying that all address and control lines are valid.

R/$\overline{\text{W}}$

READ/$\overline{\text{WRITE}}$ control line signifies a READ or a WRITE operation to Memory and I/O devices.

$\overline{\text{HOLD}}$

HOLD request to CPU used by memory or I/O devices to hold the CPU in a WAIT state during Memory or I/O transfers if required.

$\overline{\text{F.P. HOLD}}$

FRONT PANEL HOLD request to CPU used by front panel during breakpoint or single step operation signifying a long hold period. May be used by memory or I/O devices requiring a long Hold period.

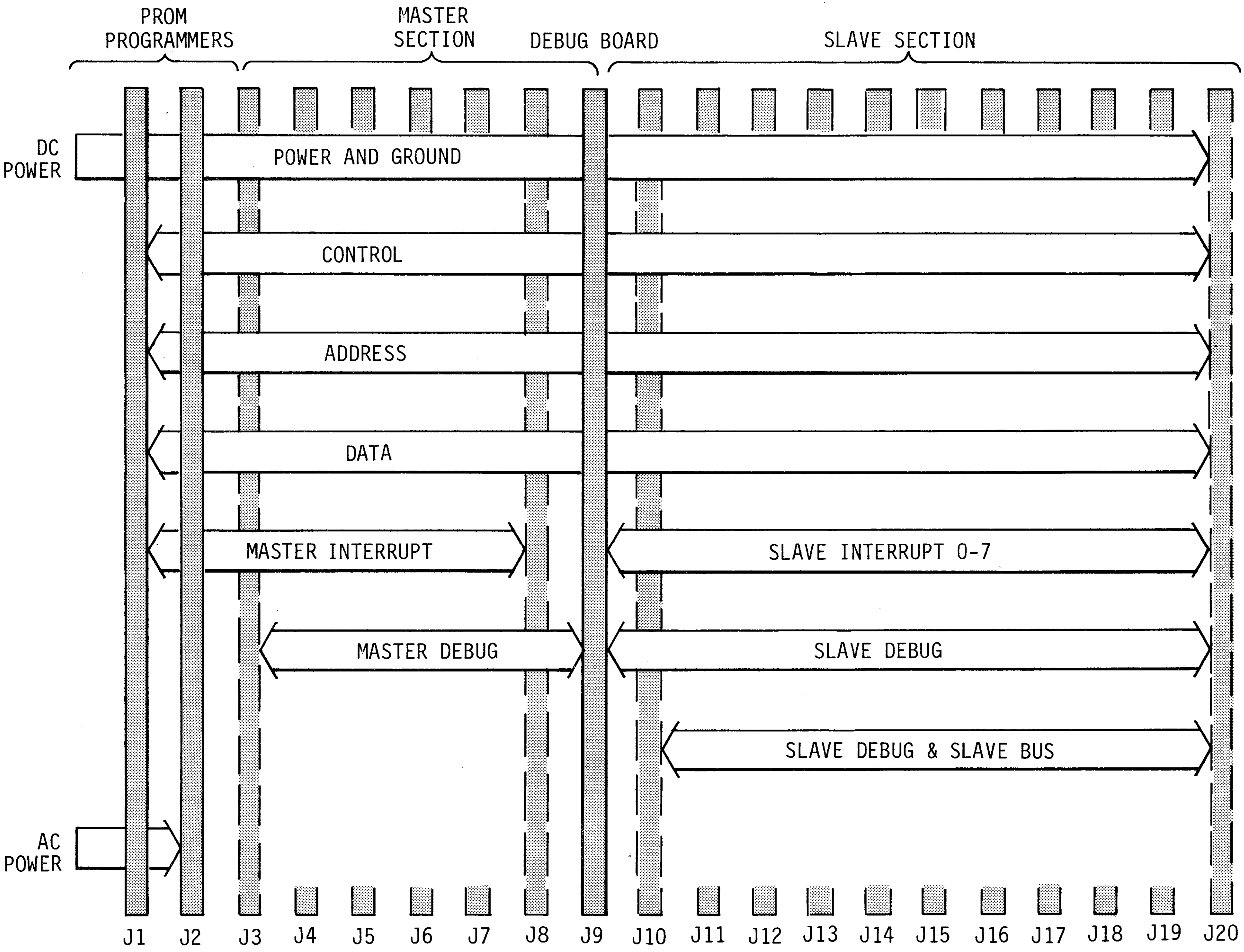


FIGURE 9-1 BUS STRUCTURE

COMMON BUS LINES

1. Power And Ground

   +5VDC
   +12VDC
   -12VDC
   GND
   AUX BUS

2. Control

   a. General Bus Control

      CMEM
      RAM INH
      WD ACCESS
      M/IO
      WRP
      OPREQ
      R/W
      HOLD
      F.P. HOLD
      RUN
      MAST RUN
      RESET
      PAUSE (DMA)

   b. Clock Lines

      I/O CLOCK
      SYS CLOCK
      2650 CLOCK
      SLV CLOCK

   c. Debug Related

      FETCH
      JMP CMD
      JMP ACK

   d. Special Interrupt

      MSTR INTACK
      MSTR INT 3

   e. Other

      SENSE

3. 16 ADDR

   DO-D15

4. 16 DATA

   AO-A15

PARTIAL BUS LINES

1. Master Interrupt

   MSTR INTØ-INT15
   MSTR INTACK

2. Master Debug

   MST PSE
   MST INTD
   DBG INT
   DBG VEN

3. AC POWER (PROM PROGRAM)

   50VAC +
   50VAC N
   30VAC +
   15VAC +
   CT
   15VAC -
   30VAC -

4. Slave Interrupt

   SLV INTØ-INT7
   SLV INTACK
   SLV CPU INT

5. Slave Debug

   SLV OPREQ
   SLV RESET
   SLV PAUSE
   DBG INT 29 - INT 31
   U PSE

6. Slave Bus

   U MAP
   SLV INOP

9-2 BUS SIGNAL GROUPS

$\overline{\text{RUN}}$ (CPU) RUNNING shared bus line indicating active processor is in the RUN state.

$\overline{\text{MAST RUN}}$ MASTER RUNNING line indicating the Master CPU is in the RUN state. Used by I/O devices to differentiate between Master and Slave I/O commands.

$\overline{\text{RESET}}$ (SYSTEM) RESET initializes all computer logic. This line represents both Power-On Reset and Front Panel Reset. The Master CPU starts execution at location zero.

$\overline{\text{PAUSE}}$ (DMA) PAUSE causes all processors to go to an inactive state with control, address and data lines tristated. This allows other devices, such as a DMA controller, to gain bus control.

$\overline{\text{FETCH}}$ FETCH line from active Slave CPU signifying an instruction fetch.

$\overline{\text{SENSE}}$ SENSE input line to 2650 Master or 2650 Slave directly connected to the Program Status Word.

$\overline{\text{FLAG}}$ FLAG output line from 2650 Master or 2650 Slave directly from the Program Status Word.

$\overline{\text{I/O CLK}}$ Input-Output 38.4 KHz Clock from Master CPU for use by any system module. Presently used by GPI/O cards for Baud Rate generation and Dynamic Memory Module for refresh.

$\overline{\text{SYS CLK}}$ 9.984 MHz SYSTEM CLOCK produced by a crystal oscillator on the Master CPU module. Used by the Debug module sequencer.

$\overline{\text{2650 CLK}}$ 10 MHz CLOCK produced by a crystal oscillator on the Master CPU module. Used to produce the 2650 Slave CPU clock.

$\overline{\text{SLV CLK}}$ SLAVE CLOCK output from the active Slave module.

$\overline{\text{JMP CMD}}$ JUMP COMMAND issued from Debug Module to initiate a forced Slave CPU jump.

$\overline{\text{JMP ACK}}$ JUMP ACKNOWLEDGE response from Slave CPU used by Debug Module to enable the jump address onto the system address bus.

$\overline{\text{MSTR INTACK}}$ MASTER INTERRUPT ACKNOWLEDGE from the Master CPU in response to an interrupt request. Signals interrupt logic to place vector on the Data Bus and is used to reset the interrupt.

$\overline{\text{MSTR INT 3}}$ MASTER INTERRUPT LEVEL 3 used by the 16K Dynamic memory module for Slave Memory Parity error interrupt.

+5V +5V LOGIC POWER BUS from 25 ampere supply.

$\overline{\text{AUX BUS}}$ AUXILIARY POWER BUS connected to a rear panel terminal strip. Used for special User Card requirements.

+12V +12V POWER BUS from dual 3 ampere supply.

-12V -12V POWER BUS from dual 3 ampere supply.

GND Logic and power GROUND BUS.

MASTER SECTION UNIQUE BUS LINES

$\overline{\text{INTØ-INT15}}$ 16 MASTER INTERRUPT lines which are priority encoded on the Master CPU module. Interrupt level Ø has highest priority.

$\overline{\text{MST PSE}}$ MASTER PAUSE control lines from the Master-Slave control logic on the Debug Module. This line holds the Master in an inactive state with its Data, Address and Control lines tristated.

$\overline{\text{MST INTD}}$ MASTER INTERRUPTED line from the Master CPU interrupt logic to the Debug Module. This signal causes the Slave CPU to be PAUSED and the Master CPU allowed to RUN.

$\overline{\text{DBG INT}}$ DEBUG INTERRUPT line from the Debug module to the Master CPU interrupt logic requesting an interrupt to the CPU.

$\overline{\text{DBG VEN}}$ DEBUG VECTOR ENABLE line from the Master CPU interrupt Logic acknowledging the DBG INT and enabling the Debug interrupt vector.

SLAVE SECTION UNIQUE CONTROL LINES

$\overline{\text{INTØ}}$-$\overline{\text{INT7}}$

8 SLAVE INTERRUPT lines to Slave CPU priority interrupt logic. Level 0 is highest priority.

$\overline{\text{SLV CPU INT}}$

SLAVE CPU INTERRUPT is a direct line to the processor bypassing all priority logic.

$\overline{\text{SLV INTACK}}$

SLAVE INTERRUPT ACKNOWLEDGE from the Slave CPU in response to an interrupt request. Signals interrupt logic to place highest priority vector on the Data Bus and is used to reset the interrupt.

$\overline{\text{SLV OPREQ}}$

SLAVE OPERATION REQUEST from active Slave CPU, valid in all Emulation modes and used by Debug Logic for breakpoint and P.C. storage operation.

$\overline{\text{SLV RESET}}$

SLAVE RESET line from Debug Module to Slave CPU's. Controlled by the Master CPU. This line holds the Slave CPU in an inactive state with data address and control lines tristated.

$\overline{\text{SLV PAUSE}}$

SLAVE PAUSE control line from the Master-Slave logic on the Debug Module.

$\overline{\text{UMAP}}$

USER MEMORY MAP line from the Hardware analyzer module directing an access from User Memory rather than System Memory.

$\overline{\text{DBG INT 29-31}}$

3 DEBUG INTERRUPT lines to Debug priority interrupt vector encoder. Used by the Hardware Analyzer module to interrupt the Master CPU.

$\overline{\text{UPSE}}$

USER PAUSE command from the active Slave CPU to the Debug module. Used to prohibit false Slave Halted detection.

$\overline{\text{SLV INOP}}$

SLAVE INOPERATIVE line from the Hardware Analyzer Module to the active Slave. This line sets all Slaves inactive allowing Master use of the bus during a Slave error condition.

PROM PROGRAMMER SECTION BUS LINES (J1 & J2)

50 VAC + 50 VAC N	50 VAC from winding on +5V supply transformer
30 VAC + 15 VAC + CT 15 VAC - 30 VAC -	Tapped 60 VAC winding on +5V supply transformer

9.2 MOTHERBOARD PIN LIST

Figures 9-3 a, b, and c, list PIN allocations for each line of the system bus. Figure 9-3a lists all common bus signals while figure 9-3b lists the split Master-Slave Bus signal lines. Figure 9-3c gives PIN allocations for PROM programmer AC to slots J1 and J2 only.

	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	+5V	+5VDC In	2	+5V	+5VDC In
	3	+5V	+5VDC In	4	+5V	+5VDC In
	5	AUX BUS	User Power Bus	6	AUX BUS	User Power Bus
	7	AUX BUS	User Power Bus	8	AUX BUS	User Power Bus
	9	GND	Signal & Pwr GND	10	GND	Signal & Pwr GND
	11	+12V	+12VDC In	12	+12V	+12VDC In
	13	-12V	-12VDC In	14	-12V	-12VDC In
	15	GND	Signal & Pwr GND	16	GND	Signal & Pwr GND
ADDRESS BUS	17	$\overline{\text{A0}}$	Address Bus	18	$\overline{\text{A1}}$	Address Bus
	19	$\overline{\text{A2}}$		20	$\overline{\text{A3}}$	
	21	$\overline{\text{A4}}$		22	$\overline{\text{A5}}$	
	23	$\overline{\text{A6}}$		24	$\overline{\text{A7}}$	
	25	$\overline{\text{A8}}$		26	$\overline{\text{A9}}$	
	27	$\overline{\text{A10}}$		28	$\overline{\text{A11}}$	
	29	$\overline{\text{A12}}$		30	$\overline{\text{A13}}$	
	31	$\overline{\text{A14}}$		32	$\overline{\text{A15}}$	
MEMORY CONTROL	33	CMEM	Slave/$\overline{\text{MSTR}}$ MEM	34	$\overline{\text{RAM INH}}$	RAM Memory Inhibit
	35	$\overline{\text{WD ACCESS}}$	Word Processor Access			
DATA BUS	37	$\overline{\text{D1}}$	Data Bus	36	$\overline{\text{D0}}$	Data Bus
	39	$\overline{\text{D3}}$		38	$\overline{\text{D2}}$	
	41	$\overline{\text{D5}}$		40	$\overline{\text{D4}}$	
	43	$\overline{\text{D7}}$		42	$\overline{\text{D6}}$	
	45	$\overline{\text{D9}}$		44	$\overline{\text{D8}}$	
	47	$\overline{\text{D11}}$		46	$\overline{\text{D10}}$	
	49	$\overline{\text{D13}}$		48	$\overline{\text{D12}}$	
	51	$\overline{\text{D15}}$		50	$\overline{\text{D14}}$	
BUS CONTROL	53	$\overline{\text{WRP}}$	Write Pulse	52	$\overline{\text{M}}$/IO	$\overline{\text{M}}$emory/Input-Output
	55	R/$\overline{\text{W}}$	Read/Write	54	$\overline{\text{OPREQ}}$	Operation Request
	57	$\overline{\text{JMP CMD}}$	Jump Command	56	$\overline{\text{HOLD}}$	Hold CPU
	59	$\overline{\text{RESET}}$	System Reset	58	$\overline{\text{RUN}}$	CPU Running
				60	$\overline{\text{JMP ACK}}$	Jump Acknowledge
SPLIT BUS see separate listing	61			62		
	63			64		
	65			66		
	67			68		
	69			70		
	71			72		
	73			74		
	75			76		
	77			78		
	79			80		
	81					
BUS CONTROL (Except J1-J2)	83	$\overline{\text{PAUSE}}$	Pause CPU's	82	$\overline{\text{FETCH}}$	Slave CPU Fetch
	85	$\overline{\text{SENSE}}$	Sense	84	$\overline{\text{MAST RUN}}$	Master CPU Running
	87	$\overline{\text{F.P. HOLD}}$	Front Panel Hold	86	$\overline{\text{FLAG}}$	Flag
	89	$\overline{\text{MST INT 3}}$	MSTR Interrupt 3	88	$\overline{\text{MST INTACK}}$	MSTR Interrupt Ack
	91	SPARE		90	SPARE	
				92	SPARE	
CLOCKS (Except J1-J2)	93	$\overline{\text{SLV CLK}}$	Active Slave Clock	94	$\overline{\text{I/O CLK}}$	38.4 KHz I/O Clock
	95	$\overline{\text{2650 CLK}}$	2650 Clock	96	$\overline{\text{SYS CLK}}$	System 10 MHz Clock
GROUND	97	GND	Signal & Pwr GND	98	GND	Signal & Pwr GND
	99	GND	Signal & Pwr GND	100	GND	Signal & Pwr GND

FIGURE 9-3a PIN ASSIGNMENENT OF MOTHERBOARD BUS SIGNALS

	PIN	MASTER SECTION (J1-J8)		DEBUG MODULE (J9)		SLAVE SECTION (J10-J20)	
		MNEMONIC	DESCRIPTION	MNEMONIC	DESCRIPTION	MNEMONIC	DESCRIPTION
SPLIT BUS SECTION	61	MSTR INTACK	MASTER INTERRUPT ACK.	SLV INTACK	SLAVE INTERRUPT ACK.	SLV INTACK	SLAVE INTERRUPT ACK.
	62	MSTR INTØ	MASTER INTERRUPT Ø	SLV INTØ	SLAVE INTERRUPT Ø	SLV INTØ	SLAVE INTERRUPT Ø
	63	1	1	MSTR INTACK	MASTER INTERRUPT ACK.	1	1
	64	2	2	SLV INT2	SLAVE INTERRUPT 2	2	2
	65	3	3	3	3	3	3
	66	4	4	4	4	4	4
	67	5	5	5	5	5	5
	68	6	6	6	6	6	6
	69	7	7	SLV INT7	SLAVE INTERRUPT 7	SLV INT7	SLAVE INTERRUPT 7
	70	8	8	SLV CPU INT	DIRECT SLAVE CPU INTERRUPT	SLV CPU INT	DIRECT SLAVE CPU INT.
	71	9	9	DBG INT29	DEBUG INTERRUPT 29	DBG INT29	DEBUG INTERRUPT 29
	72	10	10	DBG INT30	DEBUG INTERRUPT 30	DBG INT30	DEBUG INTERRUPT 30
	73	11	11	DBG INT31	DEBUG INTERRUPT 31	DBG INT 31	DEBUG INTERRUPT 31
	74	12	12	UPSE	USER SLAVE PAUSE	UPSE	USER SLAVE PAUSE
	75	13	13	SLV OPREQ	SLAVE OPERATION REQUEST	SLV OPREQ	SLAVE OPERATION REQ
	76	14	14	SLV RESET	SLAVE RESET	SLV RESET	SLAVE RESET
	77	MSTR INT15	MASTER INTERRUPT 15	SLV PSE	SLAVE PAUSE	SLV PSE	SLAVE PAUSE
	78	MST PSE	MASTER PAUSE	MST PSE	MASTER PAUSE	UMAP	USER MEMORY MAP
	79	DBG INT	DEBUG INTERRUPT	DBG INT	DEBUG INTERRUPT	SLV INOP	SLAVE INOPERATIVE
	80	PBG VEN	DEBUG VECTOR ENABLE	DBG VEN	DEBUG VECTOR ENABLE	SPARE	
	81	MST INTD	MASTER INTERRUPTED	MST INTD	MASTER INTERRUPTED	SPARE	

FIGURE 9-3b SPLIT BUS PIN ASSIGNMENTS

	PIN	MNEMONIC	DESCRIPTION	PIN	MNEMONIC	DESCRIPTION
PROM PROGRAMMER AC	83	50 VAC +	50 VAC	84	50 VAC +	50 VAC
	85	50 VAC N		86	50 VAC N	
	87	30 VAC +	TAPPED 60 VAC	88	30 VAC +	TAPPED 60 VAC
	89	15 VAC +		90	15 VAC +	
	91	C.T.		92	C.T.	
	93	15 VAC -		94	15 VAC -	
	95	30 VAC -		96	30 VAC -	

FIGURE 9-3c PROM PROGRAMMER A.C. PIN ASSIGNMENTS (J1 AND J2 ONLY)

Chapter 10
FRONT PANEL

10.0 INTRODUCTION

Two different front panels can be used with the Universal One and Universal Emulator computers. The Standard Front Panel contains minimum control and display functions, while the Full Display Front Panel provides complete monitoring and control functions. Both panels contain PROM Programming sockets. The panels physically snap-off and on with ball studs and electrically connect to the system modules through flat ribbon cables for signal functions and Molex connections for Power.

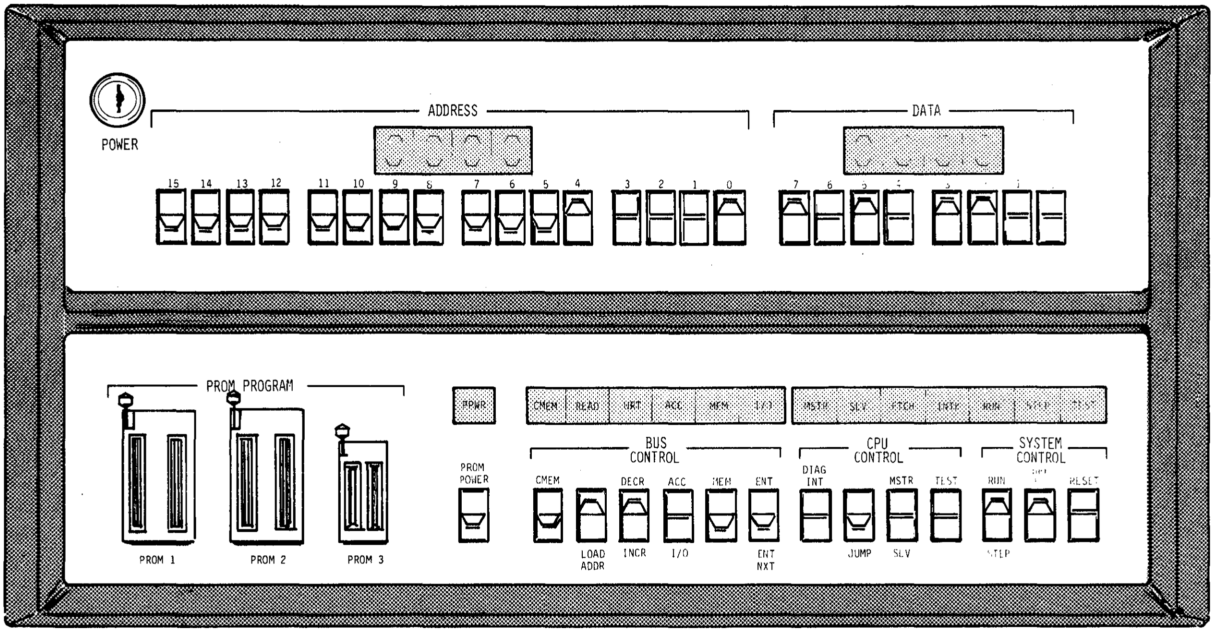
The Debug module supports all control and display functions of both panels. Two flat cables attach to the top edge of the Debug Module for the Full Display panel while only one cable is needed for the Standard Front Panel. Both panels have two PROM socket cables attaching to optional Bipolar and MOS programmer modules. The panels are completly interchangeable allowing field replacement of one with the other at any time.

Since many of the logic functions directly interact with logic on the Debug Module, it is necessary to review chapter 4.14 and 4.24 during the theory of operation discussion.

10.1 FUNCTIONAL DESCRIPTION

Since the Standard Front Panel contains a subset of the functions of the Full Display panel, the following discussion will relate to the Full Display panel only. The panel contains the following features as shown in figure 10-1.

16 ADDRESS ENTRY SWITCHES
8 DATA ENTRY SWITCHES
4 DIGIT HEXADECIMAL ADDRESS DISPLAY
2/4 DIGIT HEXADECIMAL DATA DISPLAY



- 6 BUS CONTROL SWITCHES

  CMEM
  LD ADR
  ACC
  INCR-DECR
  MEM - I/O
  ENT - ENT NXT

- 6 BUS STATUS DISPLAY

  CMEM
  READ
  WRT
  ACC
  MEM
  I/O

- 4 CPU CONTROL SWITCHES

  DIAG INT
  JUMP
  MSTR SLV
  TEST

- 4 CPU STATUS DISPLAY

  MSTR
  SLV
  FTCH
  INTK

- 3 SYSTEM CONTROL SWITCHES

  RUN - WAIT
  BRKPT
  RESET

- 3 SYSTEM STATUS DISPLAY

  RUN
  STEP
  TEST

- PROM PROGRAMMER SECTION

  2-24 PIN SOCKETS
  1-16 PIN SOCKET
  PROM POWER SWITCH
  P PWR STATUS DISPLAY

FIGURE 10-1   FULL DISPLAY FRONT PANEL

The panel supports full access and display of Master and Slave Memory and Master and Slave I/O ports. A hardware comparator provides a breakpoint or scope sync function based on the address switch settings. System control functions include System Reset, Hardware Breakpoint and CPU Single Step (address and data are displayed for every bus access). Other more specialized functions, referred to as CPU control, include a Diagnostic Interrupt switch, Slave Jump, Set Master active-set Slave active switch, and Master Test The Diagnostic Interrupt is used in the diagnostic routines to return control to the Monitor. The Slave Jump switch forces the active Slave to jump to the address switch setting. The set Master-set Slave switch gives bus control to either the Master or Slave Processors. The Master Test switch allows storage and display of the last Master CPU address for test purposes. Refer to figure 10-2 during the following discussion.

10.1.1 Address, Data Entry, and Display Address Entry -- 16 address switches can be used to control the address bus in the memory or I/O access mode, to provide a CPU jump address for the JUMP command and to provide a breakpoint address when the breakpoint switch is on. The switch value is loaded into the address register with the LOAD switch and enabled onto the address bus. In addition a Slave-Master memory switch (CMEM) selects either Master or Slave Memory and I/O during the access.

Address Bus Display -- A four digit L.E.D. hexadecimal display reads directly from the 16 bit computer address bus. This allows display of CPU addressing during single step or displays the address register contents during front panel access.

Data Entry -- 8 data switches are enabled onto the data bus during an ENTER or ENTER NEXT operation, allowing writing into memory or to an I/O device port. When a word oriented 16-bit memory is accessed, the most significant address bit (A15) specifies the low or high byte. This allows loading memory a byte at a time.

Data Bus Display -- 2 hexadecimal L.E.D. readouts directly display the 8-bit computer data bus. 2 additional optional L.E.D. readouts may be added to provide a 4-digit display for use with 16-bit processors.

10.1.2 System Control Switches

Power Switch (POWER) -- A key operated off-on switch to the upper left controls primary power to the unit.

RUN/STEP (RUN/STEP) -- Three position switch that allows the active CPU to RUN in the upward position and places the active CPU's in a HOLD condition while in the center position. Momentarily depressing the switch downward allows single CPU cycles in which OPREQ is active. The HOLD line is activated until the single step switch is depressed which allows one cycle, then reactivates the HOLD.

Breakpoint (BRKPT) -- Alternate action switch which allows use of address switches to stop program execution when that location in memory is accessed by the selected CPU. A compare during memory access initiates a HOLD, freezing the CPU during the access cycle.

Reset (RESET) -- Momentary upward action switch resets all system control FF's to their initial state. It also sets both CPU's to location Ø, enables the Master CPU and "PAUSES" the Slave CPU.

10.1.3 CPU Control Switches

Diagnostic Interrupt (DIAG INT) -- Momentary upward action switch sends a front panel interrupt to the Master CPU. (Master is activated automatically if any I/O interrupt is received).

Jump Switch (JUMP) -- Issues a hardware Jump Command to the selected CPU. The address switches are loaded on the bus and used by the CPU card as the second two bytes of an unconditional jump command. The selected CPU card disables the CPU address outputs, jams in the unconditional jump command and multiplexes the two addres bytes sequentially onto the data bus for bytes 2 and 3 of the instruction.

Set Master/Set Slave (MSTR/SLAVE) -- Two way momentary switch selects which CPU has control of the bus and is mainly used during single step.

Test (TEST) -- Is used only for Primitive CPU tests. The compliment of the last address executed by the CPU is forced into the address register.

10.1.4 Bus Control Switches

Access Switch (ACCESS) -- Alternate action switch pauses both CPU's and enables the address register onto the computer bus. Activates Front Panel MEM-I/O line and OPREQ line. Also enables the ENT/ENT Next switch.

Load Address Switch (LD ADR) -- Momentary switch downware loads the address switches into the switch register.

Increment, Decrement Switch (INCR-DECR) -- Two way momentary switch which increments or decrements the address register.

Enter/Enter Next (ENT/ENT NXT) -- Two way momentary switch which causes the contents of the data switches to be written into the memory or I/O location on the address bus. Enter Next increments the address register and then writes the contents of the data switches into the memory or I/O location.

10.1.5 Bus Status Indicators

CMEM Indicates, when lit, that the bus is accessing Slave (Common) Memory.

READ A read instruction is being executed.

WRT A write instruction is being executed.

ACC The front panel is in ACC (access) mode and the CPU's are PAUSED.

MEM A memory access is being executed.

I/O An I/O operation is being executed.

10.1.6 CPU Status Indicators

MSTR The Master CPU is active. (If RUN or STEP also lit)

SLV The Slave CPU is active. (It RUN or STEP also lit)

FTCH The Slave CPU is executing the first byte of an instruction FETCH.

INTK A Master or Slave interrupt acknowledge (INTACK) cycle is being executed.

10.1.7 System Status Indicators

RUN Indicates the Master or Slave CPU is running.

STEP Indicates the Master or Slave is being single stepped under front panel control.

TEST Indicates the status of the test switch.

10.1.8 Prom Programming Sockets And Power Switches Three zero insertion sockets are provided on the front panel for PROM programming purposes. One 24 pin socket labled PROM 1 connects to a MOS programming module through ribbon cable. The second 24 pin socket (PROM 2) and a 16 pin socket (PROM 3) connect through flat cable to a Bipolar PROM programming module.

The power switch (PROM Power) sends a logic level signal to both modules commanding a clamp of all voltages to the sockets. This allows insertion and extraction of the PROMS without shutting down computer power. An indicator (PPWR) shows the power switch status.

10.2 Theory of Operation

The front panel logic and indicators are contained on three separate printed circuit assemblies: The Upper Front Panel, the Lower Front Panel and the PROM Socket board. The Standard Front Panel has a depopulated version of the Full Display lower printed circuit board and does not contain an Upper Front Panel board.

The logic contained on the Full Display Front Panel can be divided into the following sections:

1) Data and Address Logic
2) Bus Access Logic
3) CPU Control Logic
4) System Control Logic
5) BUS, CPU and System State Indicators
6) PROM Socket and Power Switch

A block diagram showing components of both upper and lower boards is contained in figure 10-2.

The Data and Address logic contains 4 hexadecimal address displays and 2 or 4 hexadecimal data displays directly driven from buffers on the Debug module. The address and data switch settings are multiplexed to the Debug Module in two stage: The first stage selects high or low address and the second stage selects the address or 8-bit data. The 8-bit bus information feeds the Debug module internal 8-bit bus. The address multiplexer is controlled by the Debug sequencer, while the data/address multiplexer is controlled by the Enter Logic on the Lower Front Panel. Additionally, an address comparator looks for a match of bus address and the address switch setting. This produces a sync signal available for scope sync at any time, and when the breakpoint switch is on, freezes the processor with a "F.P. HOLD" command.

The Bus Access logic allows the front panel to directly control the bus to access memory or I/O modules. The ACCESS switch PAUSES both CPU's and enables the address register and control signals onto the bus as well as enabling the other access switch functions. The address register on the Debug Module can be loaded by depressing the LOAD switch. This activates the Debug sequencer and the address multiplexer on the Upper Front Panel. The register may be incremented or decremented causing an up or down count. This also removes and reasserts the OPREQ. The ENTER/ENTER NEXT function triggers two one shots, one removes and reasserts OPREQ and the other enables the front panel data switches onto the bus and issues a Write Pulse (WRP).

The CPU control logic contains the Set Master-Set Slave switch, the JUMP switch, the DIAG INT and the Test switch. The Set Master-Set Slave switch sends out pulses which change the state of the Master-Slave control flip-flop, PAUSING one CPU and enabling the other. It should be noted that any Master interrupt or Slave Halted indication could subsequently change the state of the Flip-Flop. Thus, the Slave must be enabled and no Master interrupts must be pending to successfully switch to Slave operation. The Jump switch sends a pulse to the Debug Jump initiation logic. The Slave must be enabled, and the Jump destination must have previously been loaded into the address register. If these conditions are met, the Slave will be force jumped and start execution at the destination address. The DIAG INT switch sends out a pulse which sets an interrupt flip-flop on the Debug Module. The Test switch activates a control line to the Debug sequencer which causes the sequencer to load all Master CPU addresses into the address register.

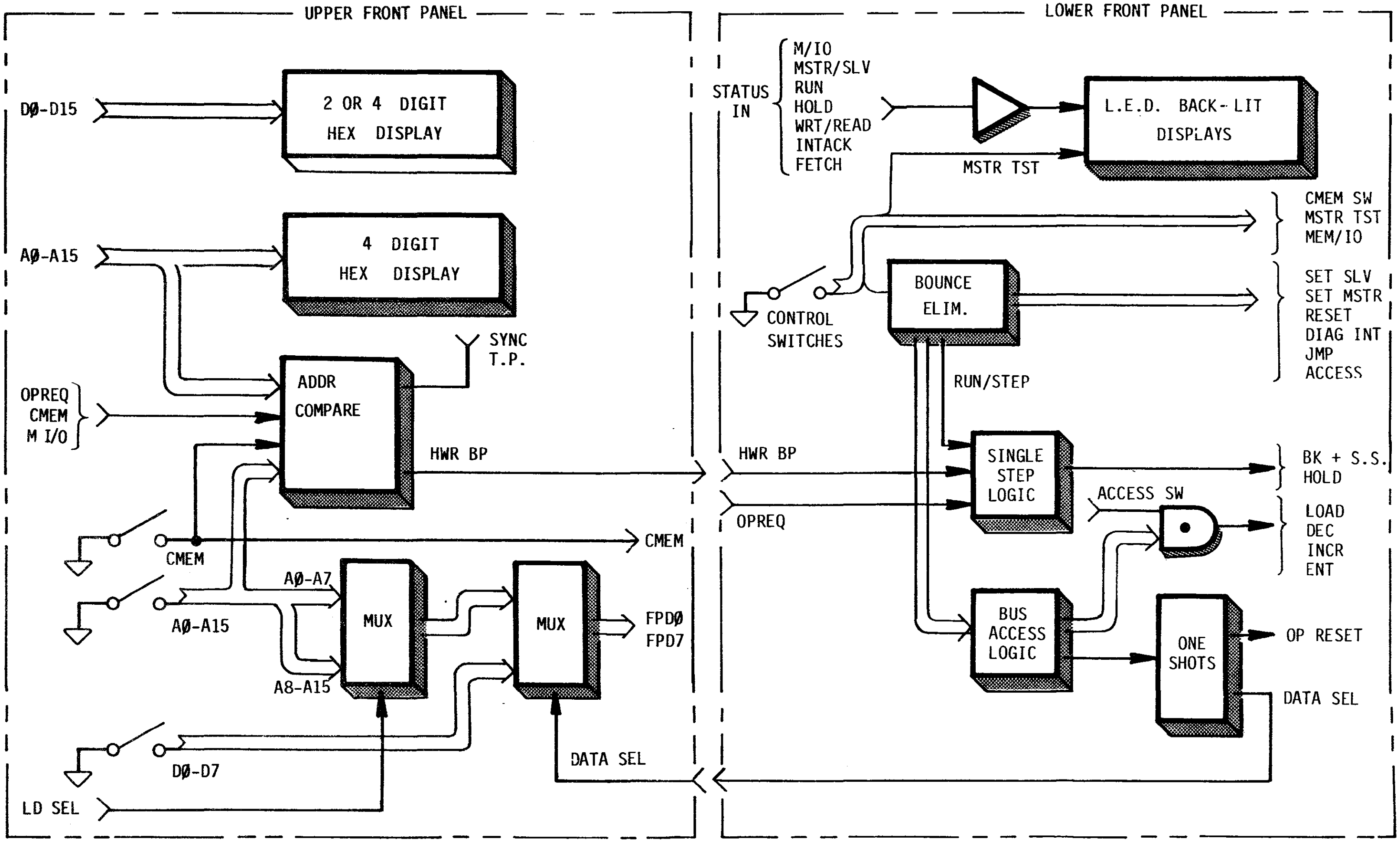


FIGURE 10-2 FULL DISPLAY FRONT PANEL BLOCK DIAGRAM

The System Logic consists of the Reset Switch, the Breakpoint switch and the RUN-Single Step switch. The Reset switch is buffered on the Debug Module and activates the bus Reset line initializing all system logic. The breakpoint switch allows a front panel HOLD signal to the bus when an address comparison is detected. The RUN-Single Step switch puts a front panel HOLD on the bus in the counter position. Depressing the switch triggers a flip-flop removing the HOLD. The HOLD is resumed on the trailing edge of the CPU OPREQ.

The BUS, CPU and System state indicators are mostly buffered signals directly from the bus. The exceptions are the Master and Slave Running, RUN-STEP, and Test indicators. The Master-Slave indicators are driven from the Master-Slave flip-flop on the Debug Module. The RUN indicator is inhibited during Step and the Step indicator lit. The TEST indicator is driven directly from the MSTR TST switch.

The PROM socket board contains only the three sockets and the two flat cables connecting to the PROM programmers. Jumper wires connect to the PROM power switch which is physically contained on the Lower Front Panel board. The power switch signal feeds the "PPWR" indicator directly and is routed through the flat cables to the PROM programmer modules.

10.2.1 Data And Address Logic This logic, shown in figure 10-3, is wholly contained on the Upper Front Panel. The Upper Front Panel is layed out to accept sockets for either Monsanto or Hewlett Packard Hexadecimal displays. The address and data displays are feed directly from inverter gates on the Debug Module which buffer the bus data.

The address lines feed 16 exclusive or gates which have the pulled up address switches as their other input. A 17th exclusive or gate has bus CMEM as one input and the CMEM switch as the other. The gates are open collector and OR tied to produce a breakpoint signal. This signal is gated to the bus on the Lower Front Panel card by the BKPT switch. The breakpoint

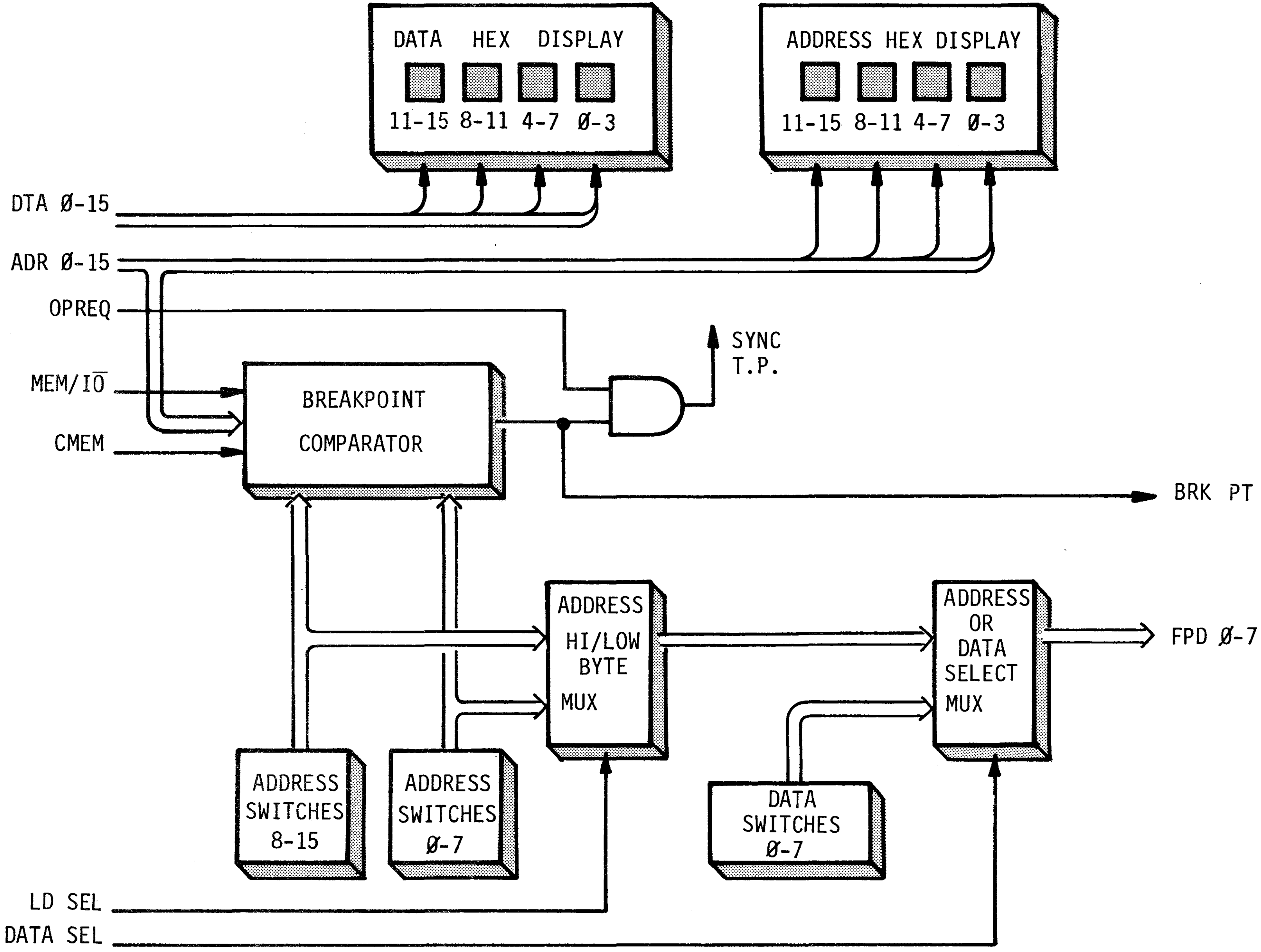


FIGURE 10-3 DATA AND ADDRESS LOGIC DETAILED BLOCK DIAGRAM

signal is ANDed with OPREQ to produce a SYNC test point for scope synchronization.

As previously discussed, the data and address switches are multiplexed in two states to produce an 8-bit bus to the Debug card. Tristate gates on the Debug card enable these 8 lines to the internal Debug bus during LOAD or ENTER operations. ENTER causes the data switches to be selected, while LOAD activates the sequencer which controls Low-High address multiplexing.

10.2.2 Bus Access Logic The bus access logic, shown in figure 10-4, allows front panel READ and WRITE to memory or I/O. The ACCESS switch feeds both the Master Pause and Slave Pause logic on the Debug Module. The active CPU reliquishes the bus after the current instruction. When RUN goes false, the address register and front panel initiated control lines (R/W, M/IO, OPREQ, WRP) are enabled onto the bus. ACCESS also enables the LOAD, DECR/INCR and ENT/ENT NXT functions.

Normally, the address switches are set to the desired value and then entered into the address register with LOAD. This pulse activates a special flip-flop on the Debug Module. It, in turn, activates the sequencer and allows load pulses to the low and high bytes of the address register. The sequencer sends a load select line to the Upper Front Panel address multiplexer controlling low or high byte selection.

Increment (INCR) or decrement (DECR) directly up or down counts the address register. It also triggers a one shot (OPRST) which removes the bus OPREQ for 1 μs allowing memory to recognize a new operation.

ENTER or ENTER NEXT triggers a 10 μs DATA SEL one shot as well as the OPRST one shot.

The DATA SEL line controls the data/address auxiliary on the Upper Front Panel and also serves as the WRITE and WRP lines to the bus. ENTER NEXT causes an increment pulse to the address register before the memory write operation.

The CMEM switch selects Slave (CMEM) or Master Memory and the MEM/IO switch selects a memory or I/O access. Both are gated through tristate buffers on the Debug module.

10.2.3 CPU Control Logic The CPU control logic, shown in figure 10-5, is very straightforward. The Set Master-Set Slave switch is bounce eliminated and sends a Set Slave Pulse or Set Master Pulse to the Master-Slave control section of the Debug Module. Either signal releases the Front Panel HOLD allowing the active processor to finish its instruction and relinquish control. Refer to the Debug Module writeup for a detailed description of the Master-Slave control logic.

The Diagnostic Interrupt switch sends a pulse to the Diagnostic Interrupt flip-flop which clocks it active. The flip-flop is reset along with all other Debug interrupts when acknowledged by the Master.

The Master Test switch sends a level to the sequencer allowing it to operate for any OPREQ rather than just Slave OPREQ's. It also enables the address register load pulses so that the bus address is sequentially loaded into the address register. Thus, when the Master HALTS in an error routine, the address register displays the compliment of the last executed address.

The Slave Jump switch sends a pulse to the Jump logic on the Debug Module. The Slave, when activated, will then jump to the address previously loaded into the address register.

10.2.4 System Control The system control logic, shown in figure 10-6, contains the Reset, Breakpoint, and RUN-Single-Step functions.

The Reset switch is debounced and feeds the system bus through a tristate driver on the Debug Module. The breakpoint switch gates the Upper Front Panel address comparison signal to the bus as Front Panel HOLD which freezes the active CPU during the bus transaction.

The RUN-STEP logic consists of the RUN-STEP switch, a flip-flop and gating. The center position ($\overline{\text{RUN}}$) allows a S. S. HOLD to the Debug Module. Depressing the switch clocks the Step flip-flop, disabling the HOLD and allowing the processor to proceed to the next bus access. $\overline{\text{OPREQ}}$ resets the flip-flop, effectively resuming HOLD on the trailing edge of OPREQ. Two HOLD functions are used on the system bus. Front Panel HOLD is used to disable the clock to certain processors (I.E., 2650) and thus cannot be asserted until OPREQ occurs. Otherwise, the processor will stop in a non-access state. This is accomplished by ANDing S. S. HOLD with OPREQ to produce the Front Panel HOLD signal. The other bus HOLD is asserted immediately on the trailing edge of OPREQ so that processors, such as the 8080, can recognize a HOLD early in its cycle and initiate a WAIT state.

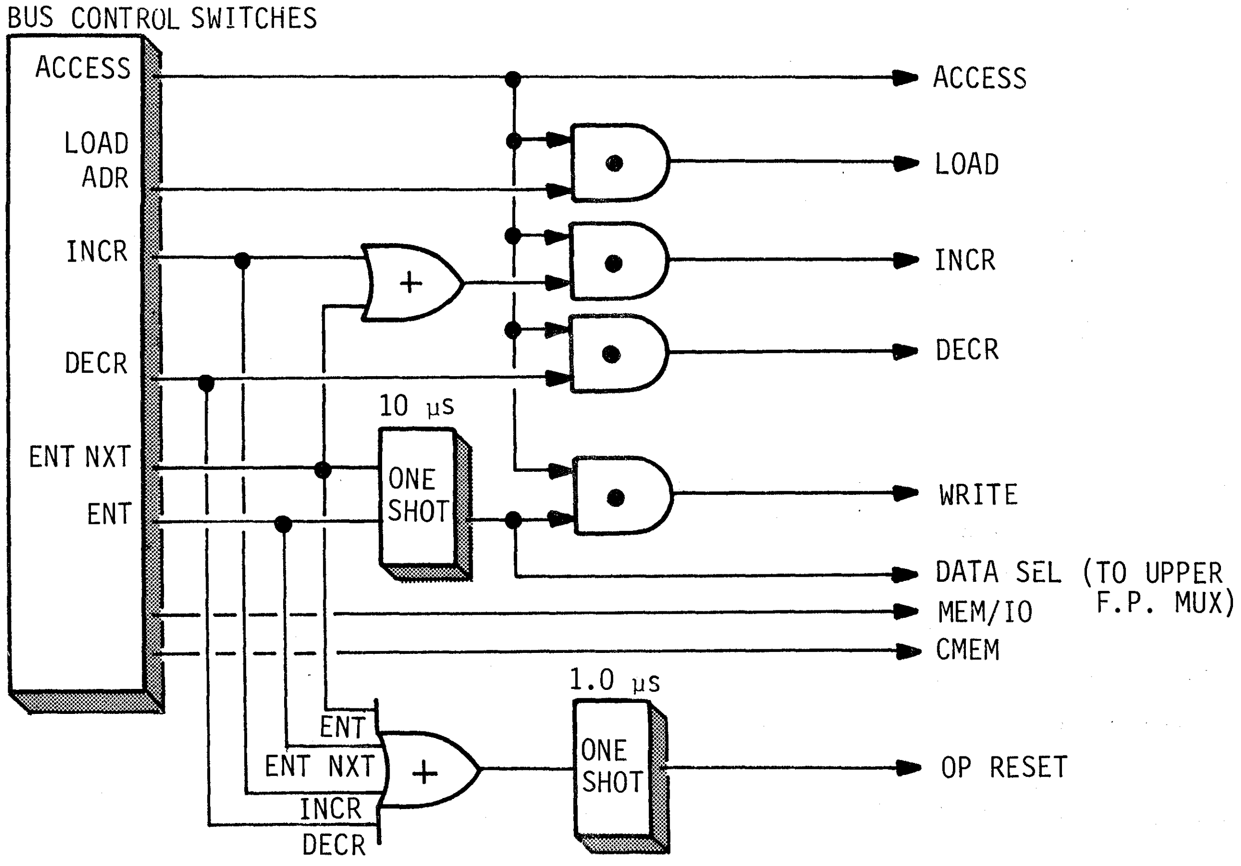


FIGURE 10-4 BUS ACCESS LOGIC

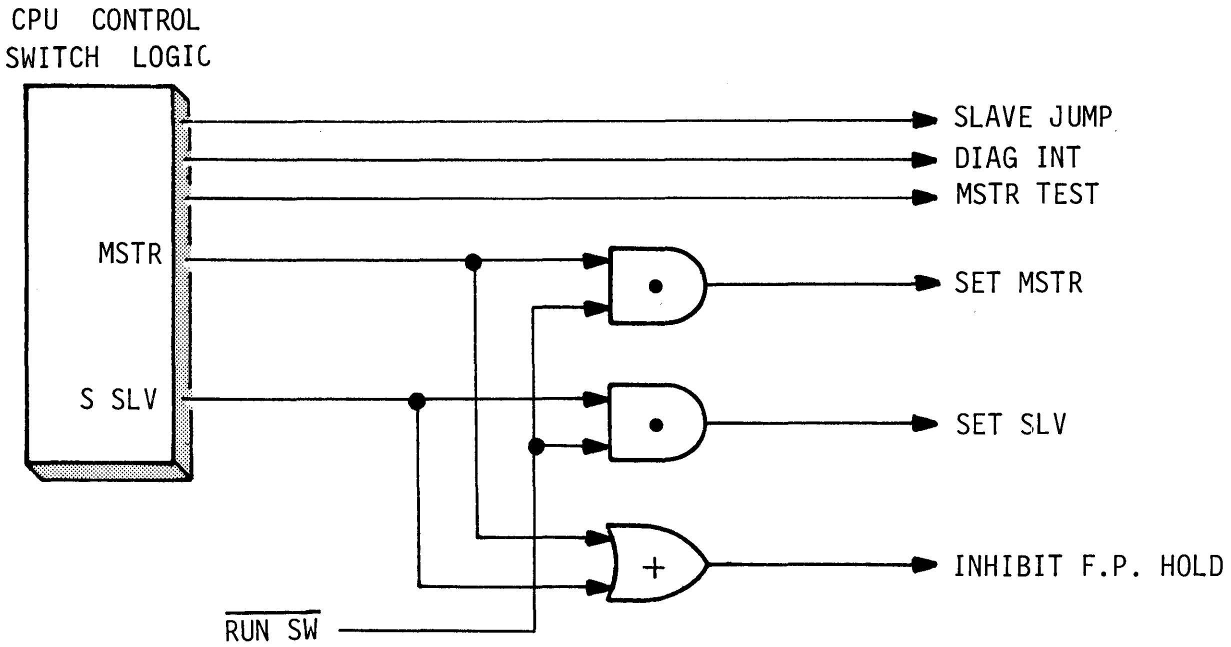


FIGURE 10-5 CPU CONTROL

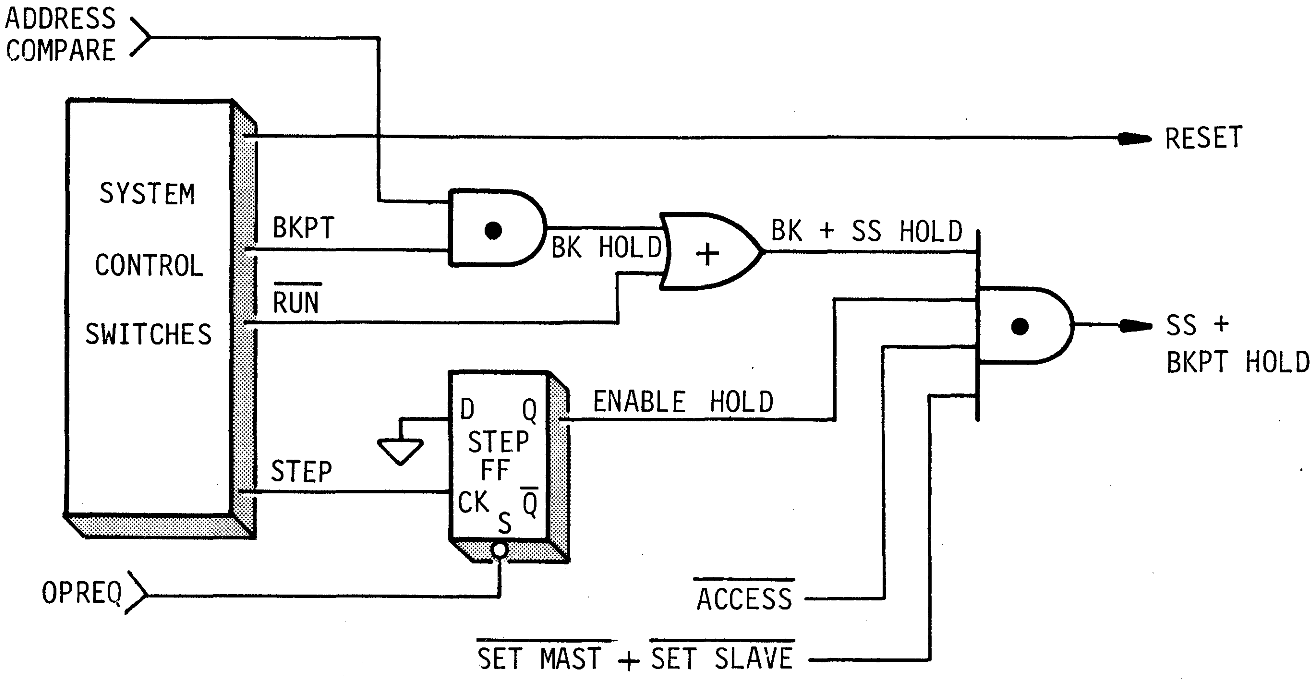


FIGURE 10-6 SYSTEM CONTROL LOGIC