



PRIORITY INTERRUPT MODULE

an option for the
Varian Data Machines
Computer Systems

The statements in this publication are not intended to create any warranty, express or implied. Equipment specifications and performance characteristics stated herein may be changed at any time without notice. Address comments regarding this document to Varian Data Machines, Publications Department, 2722 Michelson Drive, P.O. Box C-19504, Irvine, California, 92713.





varian data machines

98 A 9902 427

March 1976



TABLE OF CONTENTS

SECTION 1 GENERAL DESCRIPTION

SECTION 2 INSTALLATION

2.1 INSPECTION..... 2-1

2.2 PHYSICAL DESCRIPTION 2-1

2.3 PIM INTERRUPT LINES..... 2-1

2.4 DEVICE ADDRESS ASSIGNMENT..... 2-1

2.5 INTERCONNECTION 2-1

SECTION 3 OPERATION

3.1 I/O INSTRUCTIONS 3-1

3.2 PROGRAMMING CONSIDERATIONS..... 3-1

3.3 UNINTERRUPTABLE INSTRUCTIONS AND
CONDITIONS 3-1

3.4 PROGRAM EXAMPLE..... 3-2

SECTION 4 THEORY OF OPERATION

4.1 GENERAL..... 4-1

4.2 FUNCTIONAL DESCRIPTION 4-1

4.2.1 Control Logic..... 4-2

4.2.2 Interrupt Address Generator..... 4-3

4.2.3 Interrupt-Line Register 4-3

4.2.4 Sync Register and Line Priority..... 4-3

4.2.5 Mask Register..... 4-3

SECTION 5 MAINTENANCE

5.1 TEST EQUIPMENT..... 5-1

5.2 CIRCUIT-BOARD REPAIR 5-1

SECTION 6 MNEMONICS



LIST OF ILLUSTRATIONS

Figure 2-1. PIM Board (Component Side)	2-1
Figure 2-2. Device Address Connections	2-2
Figure 4-1. PIM Functional Block Diagram	4-1
Figure 4-2. Timing for Reception and Servicing of a Single Interrupt	4-2

LIST OF TABLES

Table 1-1. PIM Specifications	1-1
Table 2-1. Pin Assignments for Interrupt Lines	2-1
Table 3-1. I/O Instructions	3-1
Table 3-2. Program Example	3-2



SECTION 1 GENERAL DESCRIPTION

The Priority Interrupt Module is an I/O option available with Varian 70 series and 620 computer systems. This manual is divided into six sections:

- Features and specifications
- Installation and interconnection
- Operation
- Theory of Operation
- Maintenance
- Mnemonics list

Documents such as logic diagrams, schematics, and parts lists are supplied in a System Maintenance Manual. This manual is assembled when the equipment is shipped, and reflects the configuration of a specific system.

The following list contains the part numbers of other manuals pertinent to the Varian 70 series computers (the x at the end of each document number is the revision number and can be any digit 0 through 9):

V72 System Handbook	98 A 9906 20x
V73 System Handbook	98 A 9906 01x
V74 System Handbook	98 A 9906 21x
V75 System Supplement	98 A 9906 22x
Processor Manual	98 A 9906 02x
8K Core Memory Manual	98 A 9906 03x
8K Semiconductor Memory Manual	98 A 9906 04x

16K Core Memory Manual (1200 ns)	98 A 9906 24x
16K Core Memory Manual (990 ns)	98 A 9906 25x
Option Board Manual	98 A 9906 05x
Power Supply (Universal) Manual	98 A 9906 06x
V72 Power Supply Manual	98 A 9906 12x
Microprogramming Guide	98 A 9906 07x
Writable Control Store Manual	98 A 9906 08x
Memory Map Manual	98 A 9906 10x
MAINTAIN III Manual	98 A 9952 07x

The priority interrupt module (PIM) provides for the orderly servicing of peripheral-initiated interrupts of a program in progress. It does so by:

- a. Establishing up to eight levels of interrupt priority for selected peripheral controllers.
- b. Storing interrupt requests originated by associated peripheral controllers and placing the requests on the I/O bus in the order of the established priority.

In effect, the PIM organizes a "priority-within-a-priority" system. Peripheral controllers that cannot normally initiate an interrupt because of their inability to generate memory addresses can do so when connected to the PIM. PIM-controlled priority assignments are prewired at the factory to user specifications.

Table 1-1 lists the PIM specifications.

Table 1-1. PIM Specifications

Parameter	Description
Organization	Contains line synchronization, and mask registers; an interrupt address generator; priority and control logic; and line drivers and receivers
Control Capability	Establishes and implements eight levels of interrupt priority (user-assigned) for system peripheral controllers.
I/O Capability	Five external control and three transfer instructions
Standard Device Address	040 through 043
Interrupt Addresses	First PIM: 0100 through 0117 Second and succeeding PIMs: 0120 through 0177



Table 1-1 PIM Specifications (continued)

System Priority Assignment	Determined by location in the system priority chain (user-selected)
Logic levels (internal)	High = +2.4 to +5.0V dc Low = 0 to +0.4V dc
Logic levels (I/O bus)	High = +2.8 to +3.6V dc Low = 0 to +0.5V dc
Size	Contained on one 7-3/4-by-12-inch (19.7 x 30.3 cm) printed-circuit board.
Power	5V dc at 0.45A
Operating Environment	0 to 50 degrees C, 0 to 90 percent relative humidity without condensation.



SECTION 2 INSTALLATION

2.1 INSPECTION

The PIM has been packed and inspected to ensure its arrival in good working order. To prevent damage, take care during unpacking and handling. Check the shipping list to ensure that all equipment has been received. Immediately after unpacking, inspect the equipment for shipping damage. If damage exists:

- a. Notify the transportation company.
- b. Notify Varian Data Machines.
- c. Save all packing material.

2.2 PHYSICAL DESCRIPTION

The PIM circuits are contained on a single printed-circuit (PC) board (p/n 44P0683). As illustrated in figure 2-1, the board contains three connectors P1, J1, and J2. Connectors J1 and J2 each contain eight interrupt lines (IL00- through IL07-) that can be connected to selected peripheral controllers. Connector P1 also contains the same eight interrupt lines as well as all I/O bus control signals for the PIM.

2.3 PIM INTERRUPT LINES

The PIM has eight interrupt lines that enable up to eight peripheral controllers to be connected in the desired order of priority. The interrupt lines are designated IL00- through IL07-, where IL00- has the highest priority and IL07- the lowest. For controllers that are installed in the same chassis as the PIM, the interrupt lines are connected at the computer backplane connector that mates with P1 of the PIM board. For controllers in a different chassis, the

interrupt lines are contained in either an I/O expansion cable that connects to P1 of the PIM (via computer backplane) or in an interrupt cable that connects to J1 or J2 of the PIM. Pin assignments for the interrupt lines are listed in table 2-1.

Table 2-1. Pin Assignments for Interrupt Lines

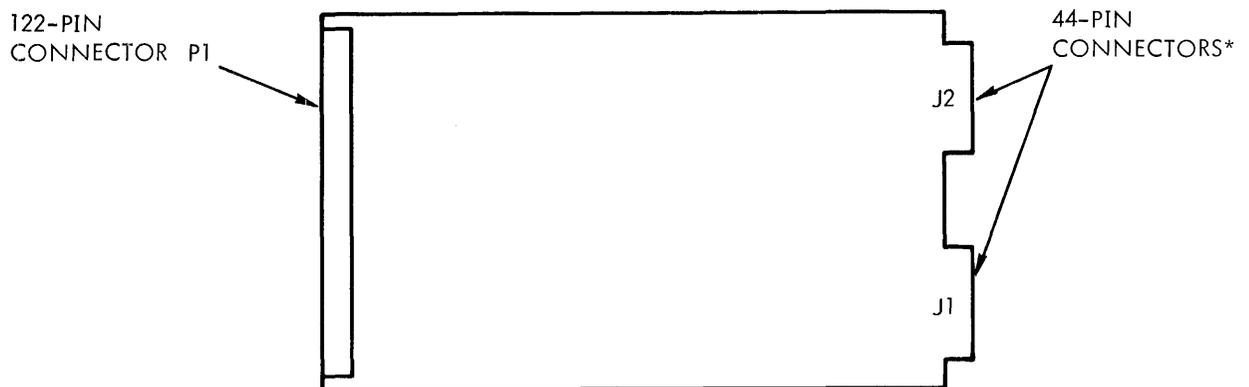
Interrupt Lines	P1			Interrupt Lines	P1		
	J1	J2	J1		J2		
IL00-	108	3	3	IL04-	102	11	11
IL01-	114	13	13	IL05-	88	5	5
IL02-	104	9	9	IL06-	112	1	1
IL03-	110	15	15	IL07-	86	7	7

2.4 DEVICE ADDRESS ASSIGNMENT

The device address for each PIM is implemented with jumper wires installed on the computer backplane connector that mates with P1 of the PIM board (figure 2-2). Device addresses 040 through 043 are reserved for PIM. Normally 040 is assigned to the first PIM, 041 to the second, 042 to the third, and 043 to the fourth. Address 044 affects all PIMs simultaneously and is used to enable or disable all PIMs.

2.5 INTERCONNECTION

In the V70 series systems, the PIM is installed in a designated slot of an I/O chassis. Refer to the appropriate Varian 70 Series System Handbook for further installation information.



* CONNECTORS J1 AND J2 ARE PARALLEL WIRED

VTH-1792

Figure 2-1. PIM Board (Component Side)

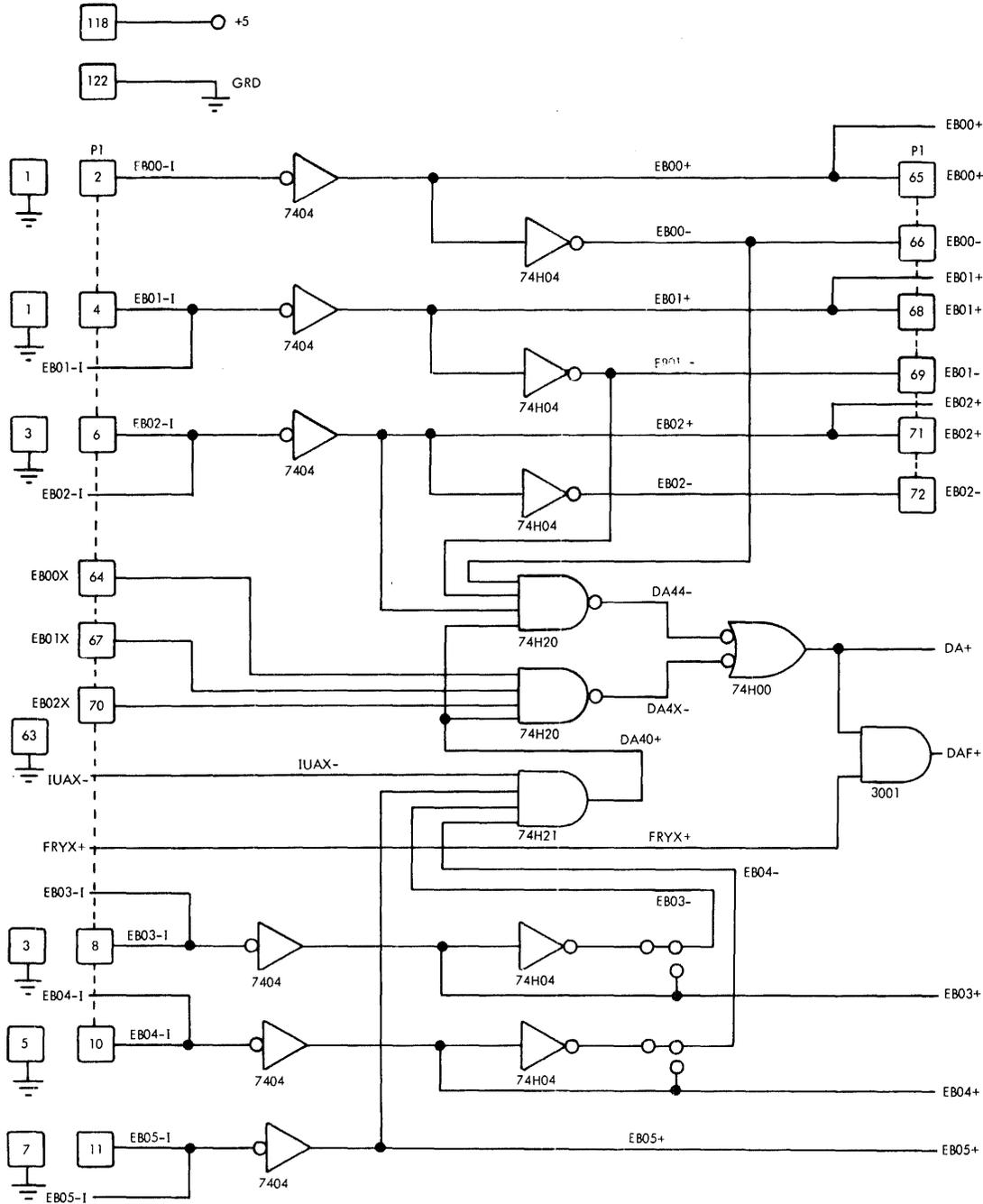


INSTALLATION

The PIM can be installed in designated slots of the main-frame and expansion chassis for 620/L systems, and in designated slots of the memory and I/O expansion chassis for 620/f-100 systems. Further installation information for these computer systems, can be found in the 620/L Maintenance Manual (document number 98 A 9905 15x) and the

620/f-100 Maintenance Manual (document number 98 A 9908 15x).

The pin assignments for the connectors on the PIM board are provided in logic diagram 91D0016.



VT11-2023

Figure 2-2. Device Address Connections



SECTION 3 OPERATION

The PIM has no operating controls or indicators. It operates under program control.

3.1 I/O INSTRUCTIONS

The PIM responds to the five external control and three data transfer instructions listed in table 3-1.

loops. Two NOP instructions are required after an external control (EXC) instruction.

For computer systems containing the memory protection (MP) option in addition to the PIM, the MP is disabled everytime an interrupt is serviced. Therefore at the end of the PIM service program, the MP must be enabled again.

Table 3-1. I/O Instructions

Mnemonic	Program Code	Functional Description
External Control		
EXC 014*	10014*	Clear interrupt registers.
EXC 024	10024	Enable PIM.
EXC 0244	100244	Enable all PIMs in system.
EXC 034	10034	Clear interrupt registers and enable PIM.
EXC 044	10044	Disable PIM.
EXC 0444	100444	Disable all PIMs in system.
EXC 054	10054	Clear interrupt registers and disable PIM.
Data Transfer		
OME 04	10304	Transfer contents of memory to mask register.
OAR 04	10314	Transfer contents of A register to mask register.
OBR 04	10324	Transfer contents of B register to mask register.

* represents the last octal digit of the device address.

3.2 PROGRAMMING CONSIDERATIONS

When preparing a PIM program, clear the interrupt registers to establish initial conditions. To mask peripheral controllers, write a mask word in the program. The eight least significant bits of the mask word correspond to the eight priority interrupt lines. Setting bit 0 inhibits the highest priority line, setting bit 1 inhibits the second-highest priority line, etc. The mask register must be loaded by the program after any power-up sequence, including the power-up cycle of the power failure/restart (PF/R) feature. System reset does not clear the PIM mask register.

When program loops contain only uninterruptable instructions, interrupts cannot occur. Thus, when recognition of an interrupt is imperative (such as with the PF/R), at least one no-operation (NOP) instruction must be added to such

3.3 UNINTERRUPTABLE INSTRUCTIONS AND CONDITIONS

An interrupt can only be detected during the last cycle of an instruction execution. In V70 and 620/f-100 systems using the memory protection feature, interrupts can be detected immediately following all instructions except:

- Halt (HLT) instructions
- Any external control (EXC) I/O instruction
- Any execution instruction. If the condition is met, interrupts are inhibited between executions of an execution instruction and the instruction at the execution address. If the condition is not met,



OPERATION

interrupts are inhibited between executions of an execution instruction and the instruction following in sequence.

d. Any instruction executed in the step mode.

In Varian 70 series and 620/f-100 systems without the memory protection feature, detection of an interrupt is inhibited during the following types of instructions:

- a. Halt (HLT) instructions
b. All shift instructions
c. All I/O instructions
d. All double-word instructions
e. All multiplication or division instructions
f. Any instruction executed in the step mode

- c. All I/O instructions
d. All shift or rotation instructions
e. All multiplication or division instructions
f. During the processor cycle immediately following an external control (EXC) I/O instruction.
g. During the processor cycle immediately following a shift, rotation, multiplication, or division instruction during which a trap occurred (DMA operation).
h. During the first instruction executed after entering run mode if that instruction is a single-word instruction.
i. During a manual step operation
j. During a halt condition

In all 620/L systems, detection of an interrupt is inhibited during the following types of instructions and conditions:

- a. Halt (HLT) instructions
b. All jump, jump and mark, or execution instructions when the jump condition is met. (When the jump condition is not met these instructions are interruptable).

3.4 PROGRAM EXAMPLE

Table 3-2 shows a typical program using the PIM. In this program, 256 descending binary frames are transferred to the high-speed paper tape punch. Memory locations 01000 to 01023 are used in the program as are computer mnemonic codes with corresponding machine language codes.

Table 3-2. Program Example

Table with 6 columns: Machine Code (Location, Instruction), Source Code (Label, Mnemonic, Operand), and Description. It lists assembly instructions like STRT, LDA, OAR, LDAL, JMP, and HLT with their corresponding machine codes and descriptions.



SECTION 4 THEORY OF OPERATION

4.1 GENERAL

Communication between the PIM and the processor is similar to that of any peripheral controller except that the PIM can request a program interrupt. When the computer acknowledges the interrupt, the PIM specifies the memory location of the instruction to be executed.

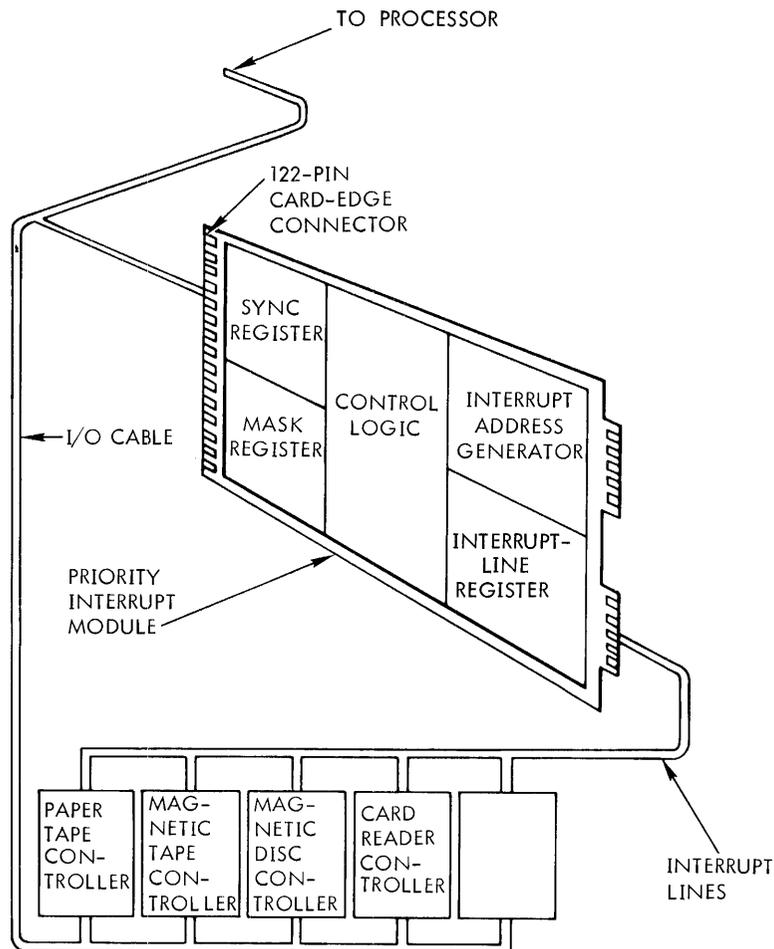
The PIM scans the interrupt lines with every cycle of the interrupt clock (IUCX-1). If signals are detected on more than one line, the highest-priority signal is acknowledged. The remaining interrupt requests are stored in the PIM interrupt-line register until acknowledged. The PIM has an eight-bit mask register that selectively inhibits interrupt requests from the interrupt-line register. When a given flip-flop of the mask register is set, corresponding interrupt requests from the interrupt-line register are inhibited. When the mask register flip-flop is reset, the interrupt

request is not inhibited. The mask register is unaffected by system reset and must be loaded under program control using data-transfer instructions.

Acknowledgment of an interrupt by the processor causes execution of the instruction located at the memory address specified by the PIM. Any instruction can be executed except an I/O type. An interrupt is thus serviced in one instruction period.

4.2 FUNCTIONAL DESCRIPTION

The following subsections describe the five functional circuits of the PIM (figure 4-1). Refer to the timing waveforms of figure 4-2 and the PIM logic diagram.



VT11-1794 A

Figure 4-1. PIM Functional Block Diagram



THEORY OF OPERATION

4.2.1 Control Logic

The control logic circuit directs and sequences the response of the PIM to external control, data transfer, and interrupt operations. When a computer program interrupt is requested, the processor requests the PIM to place the interrupt address on the I/O bus. If the PIM has system priority, the address line drivers are enabled. The PIM interrupt-request signal will remain active until all interrupts stored (but not inhibited) by the PIM have been acknowledged, or until PIM priority is lost.

The control logic circuit consists of flip-flops PRME, DTOX, IURM and associated gates. During the execution of an output data transfer command, signals FRYX and DA set DTOX. DTOX + high and DRYX + generate SMR1- low, which transfers the mask word from the I/O bus to the mask register. At the end of the transfer, DRYX + resets flip-flop DTOX. DA, in conjunction with FRYX +, indicates that the central processor is communicating with the PIM.

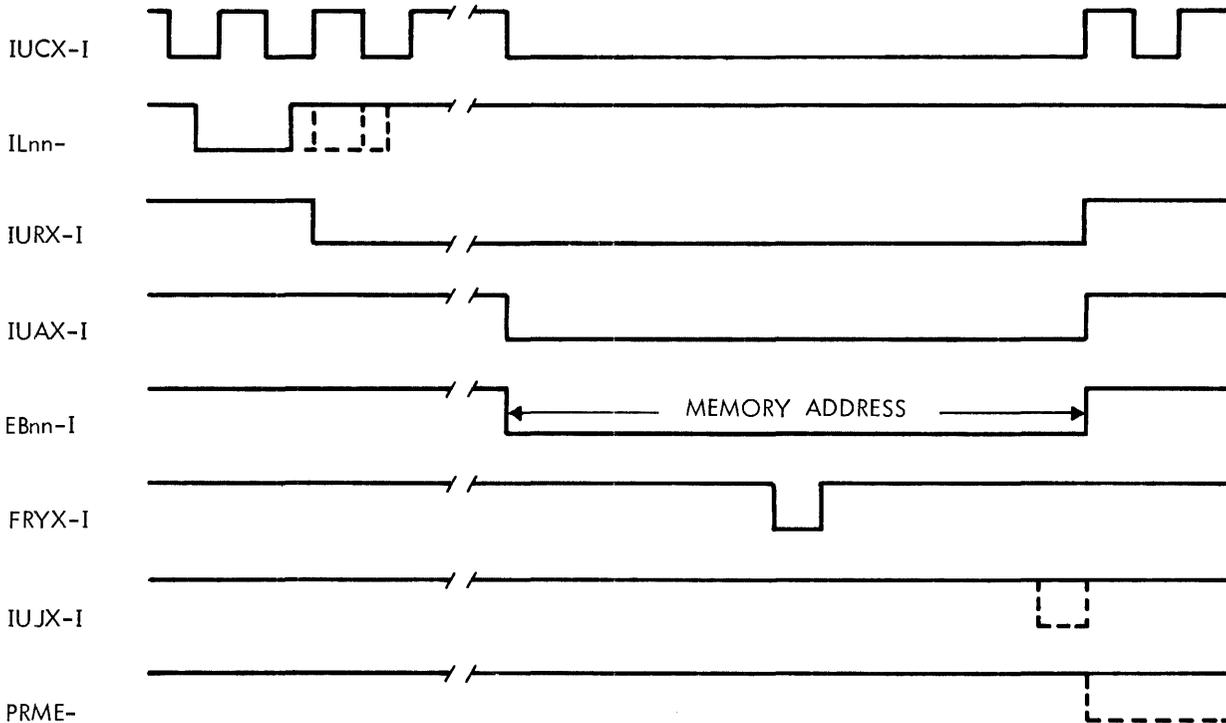
External control commands for the PIM are indicated by EXCX + low. EB06 +, EB07 +, and EB08 + are combined

with EXCX + to actuate the five external control commands.

Flip-flop PRME provides master enable/disable for the entire PIM. When PRME is reset, computer program interrupts cannot be requested by the PIM; however, the PIM continues to receive and store interrupts from external devices. When PRME + is high, flip-flop IURM is set by INR1 + high, one IUCX- I clock period after an interrupt is clocked into the sync register. IURM + and PRME + high plus PRMX- I low generate IURX- I low, requesting an interrupt.

When a computer program interrupt is requested by the PIM, the processor responds with signal IUAX- I low. This signal requests that PIM place the interrupt instruction address on the I/O bus.

If the PIM has system priority, the address line drivers are enabled as long as IUAX- I is low. When IUAX- I goes low, IUCX- I is held low. Interrupt conditions are therefore held static during the time that IUAX- I is low. Flip-flop IURM is cleared when all interrupts received by the PIM have been acknowledged by the central processor.



VT11-347B

Figure 4-2. Timing for Reception and Servicing of a Single Interrupt



THEORY OF OPERATION

4.2.2 Interrupt Address Generator

The interrupt address generator consists of coding logic that generates the binary number of the interrupt line requesting the interrupt. A pair of memory locations is reserved for each interrupt line. The interrupt addresses are normally in memory locations 0100 through 0117. However, the 16 interrupt addresses can be placed anywhere within the first 128 memory locations except 040 through 047, and may be placed at other addresses by special request.

The interrupt address generator consists of three gates that use line priority signals to generate signals A0XX+, A1XX+, and A2XX+. These signals constitute three bits of the address code for the interrupt line to be serviced. They are placed on the I/O bus by IAEX+ low. The least-significant bit of the address code is always low, resulting in an even address. This bit is supplied by the processor when the second word of a double-word interrupt instruction is accessed.

Signal IAEX+ low is generated by signals PRMX-I and IUAX-I low and FF set signal PRME+ and IURM+ high. These four signals indicate, respectively, that: the PIM has priority; the PIM interrupt has been acknowledged; the PIM is activated; and interrupt awaits on one of the interrupt lines.

4.2.3 Interrupt-Line Register

The interrupt-line register consists of eight flip-flops that asynchronously accept interrupt inputs. An interrupt is stored in the register until the interrupt is serviced or until the entire register is cleared by command.

The interrupt-line register consists of flip-flops LR00 through LR07. An interrupt is generated when an interrupt line signal, ILnn-, goes low. IL00- through IL07- are each connected to the clock input of an interrupt-line register flip-flop. ILnn- may remain low for any period of time greater than 0.2 microseconds; a constant low does not produce repetitive interrupts. If the interrupt is serviced, the flip-flop is reset by IUCP+ and a line priority signal,

LPnn+. Although IUCP+ is sent to all register flip-flops, only the flip-flop that was serviced is reset.

4.2.4 Sync Register and Line Priority

The sync register consists of eight flip-flops. At each interrupt clock period, the outputs of the interrupt-line register are clocked into the sync register. The sync register, therefore, samples the status of the eight interrupt lines synchronously with the computer interrupt clock. The sync register outputs activate the priority logic used to generate the interrupt address.

If two or more interrupts occur simultaneously, the line with the highest priority is given precedence and all other lines are temporarily inhibited. An interrupt line can request a computer program interrupt only if the line has not been inhibited and a higher-priority line is not active.

The sync register consists of flip-flops IR00 through IR07. IUC1- clocks the contents of the interrupt-line register into the sync register. The outputs of the sync register are fed into a network of gates that determine the priority of each interrupt line signal. The line priority circuit generates signal INR1+, which enables the generation of signal IURX-I. In addition, LP00+ through LP07+ determine the interrupt address and the interrupt-line register flip-flop to be reset after servicing its interrupt. IL00- has the highest priority, and signal IL07- has the lowest.

4.2.5 Mask Register

The mask register inhibits interrupt requests from selected interrupt lines to be disarmed while other lines are permitted to cause a program interrupt. The eight flip-flops of the mask register inhibit corresponding interrupt requests from the interrupt-line register. The mask register must be loaded under program control to establish which interrupts are to be inhibited.

EB00+ through EB07+ are loaded into the mask register when a particular interrupt is to be masked (inhibited). SMR1+ clocks this mask word into the register. The register outputs are fed into the line priority circuit. If one or more interrupts are inhibited by the mask register, the highest-priority unmasked interrupt is serviced.



varian data machines



SECTION 5 MAINTENANCE

Maintenance personnel should be familiar with the contents of this manual before attempting PIM troubleshooting. The Varian MAINTAIN III test program system (Test Programs Manual, 98 A 9952 07x) contains a PIM test program used to test various phases of PIM operation. Further diagnosis can then be made by referring to this manual.

5.1 TEST EQUIPMENT

The following test equipment and tools are recommended for maintenance:

- a. Oscilloscope, Tektronix type 547 with dual-trace plug-in unit, or equivalent.
- b. Multimeter, Triplet type 630 or equivalent.
- c. Soldering iron, 39-watt pencil type.

5.2 CIRCUIT-BOARD REPAIR

If it has been determined that circuit-board repair is required, it is recommended that the Varian Data Machines customer service department be contacted so that a new circuit board can be installed in the user's system and the faulty one returned to the factory for repairs. However, if the user decides to perform his own repairs, caution should be used so that the circuit board is not permanently damaged. Approved repair procedures should be followed such as the ones described in document IPC-R-700A prepared by the Institute of Printed Circuits.



varian data machines



SECTION 6

MNEMONICS

Mnemonics	Description
AnXX	Address code of device with priority interrupt.
CACR	Clear ac register. Generates signal CILR on receipt of signal EXCX.
CILR	Clear line register. Clears the line and sync registers.
DA	Decoded device address.
DRYX	Data ready pulse that resets flip-flop DTOX; enables signal SMR1.
DTOX	Data transfer out flip-flop. Stores the occurrence of an output command from the processor.
EBnn-I	Address or function code bit from I/O bus.
EXCX	Enables initialization of PIM upon external control command.
FRYX	Function ready pulse that sets flip-flop DTOX.
IAEX	Interrupt address enable. Gates address of interrupt line onto I/O bus.
ILnn	Interrupt line from peripheral controller.
INRn	Interrupt request. Indicates a request from one or more interrupt lines.
IRnn	Sync register outputs. Stores the status of the line register in synchronism with the interrupt clock signal.
IUAX	Interrupt acknowledgment. Enables servicing of PIM interrupts.
IUCP	Interrupt completion resets line register flip-flop after interrupt is serviced.
IUCX	Interrupt clock. Provides timing for servicing of PIM interrupt request.
IUCI	Interrupt clock inverted. Clocks contents of line register into sync register.
IUDX	Interrupt detection. Sets flip-flop IURM.
IUJX	Interrupt jump. Inhibits the PIM after a jump and mark command.
IURM	Interrupt request memory flip-flop. Stores a request for an interrupt from an interrupt line.
IURX-I	Interrupt request. Sent to the processor to request signal IUAX-I.
KPRME	K-input to PRME
LPii	Line priority signals. Indicates the eight PIM priorities.
LRnn	Line register flip-flop outputs. Stores request for an interrupt from a device connected to an interrupt line.
PRME	PIM enabling flip-flop. Stores the activation of the PIM.
PRMX	Priority input. Gives priority to PIM.
PRNX	Priority output. Passes priority to next in line after interrupts are serviced.
SMR1	Clocks mask word into mask register.
SYRT	System reset. Clears flip-flops DTOX and PRME and generates signal CILR when control-panel reset switch is pressed.



varian data machines



NOTES



varian data machines

NOTES