Aeon Systems Inc.

Model 5488

GPIB Crate Controller

Model 5488
GPIB Crate Controller

INSTRUCTION MANUAL

AEON SYSTEMS INC.
P.O. Box 10300
ALBUQUERQUE, NEW MEXICO 87184
(505) 828-9120

A4880C

850130

CHANGE RECORD

Rev.	Date	Change
A	9/84	Initial Release
B	10/84	Add Section 4.4 to describe Transfer Order Jumper Correct Table of Contents
C	1/85	Rewrite Section 3.1 for clarity. Add Section 3.2.4, Block Transfers

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Description

The Aeon Systems 5488 GPIB Crate Controller allows a CAMAC crate to be controlled by any computer that supports the IEEE-488 GPIB (General Purpose Instrumentation Bus) interface. It can operate either as a master or auxiliary CAMAC controller. It is also addressable so that multiple 5488's (acting as crate masters) can be connected to one GPIB interface.

The 5488 handles all types of CAMAC transfers in single or block mode, with 8, 16 or 24 bits per transfer.

The 5488 occupies a double width CAMAC slot. The GPIB cable connects at a standard GPIB connector on the front panel.

1.2 Front Panel Controls and Indicators

In addition to the GPIB connector, the 5488 front panel includes the following controls and indicators:

Control switches

ON LINE/OFF LINE - in the ON LINE position, this locking toggle switch allows the 5488 to run CAMAC normally. When it is in the OFF LINE position, the 5488 can run no cycles except Z or C from the Z/C switch.

Z/C - when the 5488 is OFF LINE, the Z/C switch can be used to execute an initialize CAMAC cycle (Z) or clear cycle (C). This switch has no effect when the 5488 is ON LINE.

LED Indicators

BUSY	-	When on, indicates a CAMAC cycle has occurred
BLOCK	-	When on, indicates the GPIB is requesting a block transfer
RSV	-	When on, indicates the 5488 has requested an interrupt on the GPIB
LSUM	-	When on, indicates a masked LAM has requested a interrupt from the computer
NO X	-	When on, indicates the last CAMAC cycle had X=0
NO Q	-	When on, indicates the last CAMAC cycle had Q=0

1.3 Interface Connections

The GPIB connects to the 5488 through a connector on the front panel. This connector accepts the standard GPIB connector.

Multiple controllers in a single crate communicate through a flat cable that connects to each controller in the free access area at the back of the card.

In addition to these switches and indicators, the front panel includes connectors for Request and Grant lines. Refer to section 2.3 for information on connecting these lines.

2.3 Cabling

2.3.1 Request/Grant Chain Connectors

Cabling of the REQUEST, GRANT IN and GRANT OUT front panel connectors depends on the role of the 5488 in the crate, as follows:

If the 5488 is the only controller in the crate,

Connect REQUEST to GRANT IN

CAMAC cycles cannot be executed unless this connection is made.

If the 5488 is the master controller in a multi-controller crate,

- Connect REQUEST to GRANT IN and
  GRANT OUT to GRANT IN on the next lower-priority controller in the crate

If the 5488 is an auxiliary controller in a multi-controller crate,

Connect GRANT IN to GRANT OUT on the next higher-priority controller in the crate

2.3.2 GPIB Connector

The GPIB connector is keyed so that it can only be oriented one way. Screws at the ends of the cable connector engage threaded bosses on the front panel connector to secure the connection.

2.3.3 Auxiliary Controller Bus

If the 5488 is a master or auxiliary controller in a multi-controller crate, connect the auxiliary controller bus cable at the connector in the free access area at the back of the module. The 5488 uses the standard 40-pin CAMAC auxiliary controller bus connector.

3. OPERATION

3.1 Programming Considerations

All CAMAC commands in this manual are shown in so-called NAF form. For example, the Write Status Register command is shown as

```
N(30)·A(0)·F(17)
```

The number in parentheses following the N is the slot number, which is always 30 for the 5488. The value following A is the subaddress and the number following F is the function code. Commands are sent to the 5488 over the GPIB as three bytes in NAF order. For more information on CAMAC commands, refer to section 3.2.3.

Throughout this section, examples are given to show how these commands are expressed in an actual computer programming language. The programming examples use the syntax of Hewlett Packard BASIC, version 2.0. This version contains statements specially suited to data acquistion and control, and therefore allows concise expression of the 5488's functions. The principles can, however, be expressed easily in other dialects of BASIC. For example, in HP BASIC, the Write Status Register command (refer to section 3.2.1) is written as follows:

```
OUTPUT Cam USING "#,B";30,0,17,X,Y,Z
```

This statement is a formatted output command that transmits the data in the value list as bytes. Note that all transfers over the GPIB are sent one byte at a time. Cam is the device selector for the 5488. The device selector is composed of the select code of the GPIB interface--which depends on the computer being used--and the device address of the 5488. As supplied, the 5488 device address is 16 for single transfers and 17 for block transfers (refer to section 3.2.4 for more information on block transfers). These device addresses can be changed by means of the address switches on the 5488. Refer to section 4.2 for instructions on setting these switches.

If, for example, the select code of the GPIB interface is 7, and the device address for single transfers is 16, then the device selector Cam is 716. In practice, it may be helpful to include a statement that computes the device selector as follows:

```
Cam=Gpib*100+Device_address
```

where Gpib is the select code. For block transfers, the device selector can be expressed as

```
Cam=Gpib*100+Device_address
Block=Cam+1
```

Handling the device selector in this manner allows programs to be changed easily if the select code or device address is

changed.

The first three items in the OUTPUT statement list are N, A and F and the the last three items are variables that contain the values to be assigned to three bytes of the status register. Note that in HP BASIC, the data bytes are sent highest-order byte first. In some other computer systems the order is reversed. To use the 5488 in such a system, the Transfer Order Jumper must be removed on the 5488 (refer to section 4.4) and the order of the values in the OUTPUT statement (or equivalent) must be reversed from that in the examples. This is particularly crucial when handling data from multiple-byte CAMAC data transfers. Since the data are transmitted over the GPIB one byte at a time, keeping track of the correct order of transfer is essential.

No matter whether the Transfer Order Jumper is installed or not, however, the order of N, A and F is always the same.

3.2 Programming the 5488

3.2.1 Set-Up

After the GPIB interface has been configured for data transfers, commands can be sent to the 5488 to prepare it for data transfer. The first step is to program the 5488 status register for the required communication parameters. The status register consists of three bytes that can be written by the following CAMAC command:

N(30)·A(0)·F(17)

This command is followed by three bytes of data corresponding to the three bytes of the Status Register. To calculate the values of the first two of the bytes, determine the bits that are to be set in the status register. Table 3-1 lists the bits and their functions. For each byte, the value to be sent in the command is the sum of the numbers in the Value column corresponding to the bits to be set. The third data byte is a place holder corresponding to the third byte of the status register. Since this byte is read-only, the 5488 ignores its contents. It must be present, though, for correct operation.

Example:

To set the 5488 for 16-bit CAMAC transfers and UQC (Q Repeat) block transfers, bits 0 and 2 in Byte 2 of the status register must be set. From Table 3-1, the Value of bit 0 is 1 and the value of bit 2 is 4, so the value of byte 2 of the status register is 5. BASIC commands to set the status register are, then

```
OUTPUT Cam USING "#,B" 30,0,17,0,5,0
```

When power is first turned on, or when the GPIB interface asserts the CLEAR signal, the status register settings are as follows:

Interrupt Request Mask	-	all bits set to 0
Mode	-	24-bit CAMAC transfers
Block Transfers	-	UCC

In addition, all bits of the LAM Mask Register (refer to Section 3.2.2) are reset to 0.

Once the status register is set as required, the 5488 is ready for use.

TABLE 3-1
5488 Status Register

Byte 1 - Interrupt Request Mask

Bit	Value	Function
7	128	Z. When set, the 5488 executes an initialize CAMAC cycle. This is a write-only bit.
6	64	C. When set, the 5488 initiates a clear CAMAC cycle. This is a write-only bit.
5	32	LAM SUM ENABLE When set, any enabled LAM causes an interrupt request. This bit can be written to set or reset the function. Reading the bit returns the current status.
4	16	INH ENB. When set, enables bus interrupts when the dataway Inhibit signal is active for any controller in the crate.
3	8	ON LINE EN. When set, enables a bus interrupt when the ON LINE/OFF LINE switch is moved to the ON LINE postion.
2	4	NOT USED
1	2	NO X EN. When set, enables a bus interrupt when a CAMAC cycle terminates with X=0.
0	1	NO Q EN. When set, enables a bus interrupt when a CAMAC cycle terminates with Q=0.

Byte 2 - Mode

Bit	Value	Function
7	128	NOT USED
6	64	NOT USED
5	32	INH. Asserts the CAMAC Inhibit line.
4	16	MB2. DMA block mode control. See table below
3	8	MB1
2	4	MB0
1	2	BT1
0	1	BT0

	MB2	MB1	MB0
ACA	1	x	x
UCW	0	1	1
UCS	0	1	0
UQC	0	0	1
UCC	0	0	0

CAMAC Word Length control.
See table below:

	BT1	BT0
8-bit	1	x
16-bit	0	1
24-bit	0	0

Note that all transfers from the 5488 to the controller over the GPIB are in 8-bit bytes.

TABLE 3-1 continued
5488 Status Register

Byte 3 - Status Byte

Bit	Value	Function
7	128	NOT USED
6	64	NOT USED
5	32	IRT ENB. When set, indicates that GPIB interrupts are enabled.
4	16	INH. Indicates the Inhibit line on the dataway is active
3	8	ON LINE. Indicates the controller is on line.
2	4	NOT USED
1	2	X. Indicates the last CAMAC cycle ended with X=1
0	1	Q. Indicates the last CAMAC cycle ended with Q=1

This register is read-only. Writing the register does not change its contents.

3.2.2 LAM Handling

A. LAM Status Register

LAMs from the other modules in the crate are routed to the 5488's auxiliary connector at the back of the module in the free access area. These LAMs can be read at any time by means of the following command:

N(30)·A(12)·F(1)

In response to this command, the 5488 sends three data bytes that correspond to the three bytes of the LAM Status Register as shown in Table 3-2, below.

Table 3-2. LAM Status Register

Byte 1 (X)	Bit	Value	LAM
	7	128	24
	6	64	23
	5	32	22
	4	16	21
	3	8	20
	2	4	19
	1	2	18
	0	1	17
Byte 2 (Y)	Bit	Value	LAM
	7	128	16
	6	64	15
	5	32	14
	4	16	13
	3	8	12
	2	4	11
	1	2	10
	0	1	9
Byte 3 (Z)	Bit	Value	LAM
	7	128	8
	6	64	7
	5	32	6
	4	16	5
	3	8	4
	2	4	3
	1	2	2
	0	1	1

To determine which LAMS are being asserted, convert the returned values to eight-bit binary numbers. Each bit corresponds to one of the LAMs as shown in Table 3-2. For each LAM bit, one means the LAM is being asserted and 0 means that it is not.

B. LAM Mask Register

To generate LAM interrupt requests, the 5488 forms the logical product (AND) of the LAM Status Register and the LAM Mask Register. The 5488 requests an interrupt over the GPIB (using the RSV line) in response to a LAM if and only if the corresponding bit in the LAM Mask Register is set to one and the LAM interrupt bit is set in the Status Register. To read the current state of the LAM Mask Register, use the following command:

N(30)*A(13)*F(1)

As with the LAM Status Register, the three data bytes returned in response to this command correspond to the three bytes of the LAM Mask Register. The data bytes are coded in the same way as the LAM Status Register as shown in Table 3-2. One means the corresponding LAM can request an interrupt and zero means it cannot.

When power is first turned on, or when the GPIB interface asserts the CLEAR signal, all bits in the LAM Mask Register are reset to 0.

To change the LAM Mask Register, use the following command:

N(30)*A(13)*F(17)

This command is followed by three data bytes corresponding to the three bytes of the LAM Mask Register. To code these data bytes, determine the LAMs to be allowed to generate interrupt requests. Write out the contents of the LAM mask register with 1's for those LAMs and 0's for all the rest. Then divide the register in three bytes and express in decimal according to the scheme shown in Table 3-2.

Example:

Assume LAMs from slots 3, 6 and 16 are to be allowed to request interrupts. LAM 16 corresponds to a value of 64 in Byte 2 of the LAM Mask Register, while 3 and 6 correspond to a value of 36 (sum of 4 and 32) in Byte 3. Therefore, a BASIC command to enable these LAM interrupts is

```
OUTPUT Cam USING "#,B";30,13,1,0,64,36
```

where Cam is the device selector of the 5488 as described in the example in section 3.1.

C. LAM Request Register

The LAM Request Register stores the current logical product of the LAM Status Register and the LAM Mask Register. That is, if a LAM is true and its corresponding bit in the LAM Mask Register is 1, then the corresponding bit in the LAM Request Register is also 1.

To read the contents of the LAM Request Register, use the following command

N(30)*A(14)*F(1)

In response, the 5488 returns three data bytes that are coded in the same way as the LAM Status Register in Table 3-2.

D. LAM Interrupts

The 5488 can be configured to allow selected LAMs to request interrupts on the GPIB RSV line. The procedure for doing this is as follows:

NOTE

Remember that in order for this approach to work, an appropriate interrupt handling routine must be provided.

Step Procedure

1. Enable the LAM interrupts by setting the LAM SUM bit in the Interrupt Request Register Mask of the 5488 Status register. For example:

   OUTPUT Cam USING "#,B";30,0,17,32,1,0

   where Cam is the device selector as described in section 3.1. This command also puts the 5488 in 16-bit CAMAC transfer mode (optional).

2. Enable those LAMs to be allowed to request interrupts by setting the corresponding bits in the LAM Mask Register. Suppose LAMs 3, 5, 11, and 22 are to request interrupts. LAM 22 is bit 5 in Byte 1, which corresponds to a decimal value of 32. (Refer to Table 3-2 for the values of each LAM.) LAM 11 is 4. LAMs 3 and 5 are 4 and 16 respectively. The value for the third byte of the command is the sum of these values, 20. So to enable these LAMs, use the following command:

   OUTPUT Cam USING "#,B";30,13,17,32,4,20

3. Enable GPIB interrupts from the 5488 Crate Controller. An example of a BASIC command to do this is the following:

   ON INTR Gpib GOSUB Camac_interrupt

   where Gpib is the select code of the GPIB interface as described in section 3.1.

4. Enable the RSV interrupt from the 5488 Crate Controller. Make sure the interrupt mask allows interrupts from the RSV line on the GPIB interface. In this example, the mask is one. For example,

```
Mask = 1
ENABLE INTR Gpib;Mask
```

3.2.3 Other CAMAC Commands

NOTE

A summary of 5488 commands is in Appendix B.

The 5488 expects CAMAC commands to be a sequence of three bytes over the GPIB. The bytes of a CAMAC command have the following formats:

Byte 1 (CN) (Crate and Module Number)	Bit	Function
	7	crate (MSB)
	6	crate
	5	crate (LSB)
	4	N16
	3	N8
	2	N4
	1	N2
	0	N1

Byte 2 (A) (Subaddress)	Bit	Function
	7	not used
	6	not used
	5	not used
	4	not used
	3	A8
	2	A4
	1	A2
	0	A1

Byte 3 (F) (Function)	Bit	Function
	7	not used
	6	not used
	5	not used
	4	F16
	3	F8
	2	F4
	1	F2
	0	F1

After it receives the three command bytes, the 5488 determines the direction of data flow. If the command is a write command (F16 =1 and F8 = 0), the 5488 waits for the proper number of data transfers from the GPIB interface and then executes a CAMAC cycle. If it is a read command (F16 and F8 = 0) the 5488 executes a CAMAC cycle and then waits for the GPIB interface to read the proper number of data transfers.

After executing a command, the 5488 makes its status byte (byte 3, Table 3-1) available for reading. To retrieve this status byte, execute a GPIB serial poll at the 5488's device selector.

Example:

To execute a 24-bit CAMAC data read, the computer first must send the CAMAC command to the module, as follows:

```
OUTPUT Cam USING "#,B"; N,A,F
```

where Cam is the variable holding the GPIB select code, N is the variable holdng the number of the module to be interrogated, A contains the subaddress and F represents the function code. To retrieve the data, the following BASIC command could be executed:

```
ENTER Cam USING "#,B"; X,Y,Z
```

3.2.4 Block Transfers

Block transfer commands allow the 5488 to transfer large quantities of data between the GPIB interface and the CAMAC system without requiring the GPIB controller to manage each transfer. The 5488 has several types of block transfers, differing according to the synchronizing source, addressing technique and terminating method. These types are usually referred to in the CAMAC literature by three-letter designations as follows:

UCC - Word Count (Classic DMA) mode. The 5488 transfers words (or bytes) at full controller speed for a number of transfers specified by the GPIB controller. The 5488 does not contain a word counter since the GPIB DMA controller stops the transfers itself when the word count has been met.

UQC - Q Repeat mode. In this mode, the 5488 transfers data between the GPIB interface and one of the CAMAC modules. The GPIB controller specifies the module number and subaddress to or from which data is to be transferred and the module signals each successful transfer by setting Q=1. Unsuccessful transfers cause Q=0. When Q=0, the 5488 attempts the transfer again. This property of UQC transfers make them useful for reading from or writing to slow devices. The GPIB controller terminates the operation when all required data has been transferred.

ACA - Address Scan mode. The GPIB controller designates a module number and subaddress at which to start transfering data. After each transfer, the subaddress is incremented and another transfer is attempted. If Q=0 at any subaddress or after the transfer at A(15), the subaddress is reset to zero and the module number is incremented. Transfer continues until stopped by the GPIB controller, or until X=0.

UCS and UCW - Q Stop mode. Transfers continue at the GPIB controller-specified address until Q=0. In UCW mode, the word transferred when Q=0 is valid. In UCS mode, the last transfer is invalid.

Setting the appropriate bits in Byte 2 of the Status Register puts the 5488 in block mode. All block mode transfers are through the block mode device address, which is the device address set by the address switches (refer to section 4.2) plus one. As supplied, the block mode device address is 17.

The following is a general procedure for performing block transfers. It includes examples taken from HP BASIC to illustrate the principles involved. Following the general procedure are typical routines for performing the various types of block transfers.

Step Procedure

1. Set up a buffer to hold the data. For example,

```
INTEGER Da(4000) BUFFER
```

sets up a 4000-byte buffer.

2. Assign a path to the Controller and to the Buffer.

```
Gpib=7
Device_address=16
Cam=Gpib*100+Device_address
ASSIGN @Camac TO Cam+1
ASSIGN @Path to BUFFER Da(*)
```

NOTE

HP BASIC uses ASSIGN statements to set up paths between DMA input and output ports and program variables. Use of the paths allows the DMA controller to operate independently of the processor. The path may require certain attributes, depending on the type of transfer to be performed. The examples below show how these attributes are defined.

3. Send the 5488 Crate controller command that indicates the type of block mode to be performed. For example:

```
OUTPUT Cam USING "#,B";30,0,17,0,1,0
REM This puts the Controller in 16 bit, UCC mode
```

4. Send the read or write command to the 5488. For example, assume a block operation at module number 2:

```
OUTPUT Gpib USING "#,B";2,0,0        ! Typical read
                                     ! command
```

or

```
OUTPUT Gpib USING "#,B";2,0,17       ! Typical write
                                     ! command
```

5. Initiate the transfer. For a block read,

```
TRANSFER @Camac TO @Path             ! Read from 5488
                                     !  into buffer
```

or for a block write

```
TRANSFER @Path TO @Camac              !Writefrom
                                     !  buffer to
                                     !  5488
```

NOTE

The transfer may also have attributes. Refer to the examples.

To further illustrate the use of block transfer modes, the following examples show how they may be implemented in BASIC.

UCC mode

This routine writes 1000 values to a device in the crate.

```
Gpib=7                                  ! Select Code of GPIB
Device_address=16                       ! Device Address of 5488
Cam=Gpib*100+Device_address             ! Device Selector
INTEGER Ad(4000) BUFFER                 ! Transfer buffer
ASSIGN @Camac TO Cam+1                  ! Path for 5488 to GPIB
ASSIGN @Path TO BUFFER Ad(*)            ! Path for buffer
OUTPUT Cam USING "#,B";30,0,17,0,1,0    ! 16 bit word,UCC mode
                                        !  Note that this is a
                                        !  transfer
OUTPUT Cam USING "#,B";2,0,16           ! Send read command to
                                        !  module
TRANSFER @Path TO @Camac;COUNT 1000     ! Read in 1000 values
```

UCS and UCW Mode

This routine reads up to 4000 bytes from a device in slot 2 of the crate in Q-Stop mode.

```
Gpib=7                                  ! Select Code of GPIB
Device_address=16                       ! Device Address of 5488
Cam=Gpib*100+Device_address             ! Device Selector
INTEGER Ad(4000) BUFFER                 ! Set Up Transfer buffer
ASSIGN @Path TO BUFFER Ad(*)            ! Path for buffer
ASSIGN @Camac TO Cam+1                  ! Path for 5488
OUTPUT Cam USING "#,B";30,0,17,0,9,0    ! 16 bit word,UCS mode
OUTPUT Cam USING "#,B";2,0,0            ! Send read command to
                                        !  module
TRANSFER @Camac to @Path;WAIT           ! Write entire buffer
REM The WAIT in the above statement suspends
REM program execution until the transfer is complete
```

ACA Mode

This example routine transfers up to 48 values starting at N=2, A=0.

```
Gpib=7                                  ! Select Code of GPIB
Device_address=16                       ! Device Address of 5488
Cam=Gpib*100+Device_address             ! Device Selector
INTEGER Ad(47) BUFFER                   ! Sets Up 48 Value
                                        !  Transfer buffer
ASSIGN @Path TO BUFFER AD(*)            ! Path for buffer
ASSIGN @Camac TO Cam+1                  ! Path for Crate
OUTPUT Cam USING "#,B";30,0,17,0,17,0   ! Set Status Register for
                                        !  16 bit word,UCS mode
OUTPUT Cam USING "#,B";2,0,0            ! Send starting address
TRANSFER @Camac to @Path                ! Read block data
```

4. SPECIAL OPERATIONS

4.1 Making the 5488 an Auxiliary Controller

The 5488 can operate as an auxiliary controller with any controller that can handle auxiliaries. To convert the 5488 to an auxiliary controller, remove all of the CAMAC pull-up resistors. These resistors are in dual in-line (DIP) packages and mounted in sockets for easy removal. To remove the resistor packs, use the following procedure:

Step	Procedure
1	Remove two screws from the right side of the front panel and two screws from the left side of the rear panel.
2	Gently slide the right-hand board (as viewed from the front) back and away from the front panel.
3	Disconnect the flat cable from the right-hand board by lifting the latches at either end of the connector and pulling the cable connector back.
4	Remove the following resistor packs from the board. U9 - 390 ohm U17 - 680 ohm U22 - 680 ohm U28 - 680 ohm U33 - 680 ohm U34 - 680 ohm U39 - 680 ohm Save these resistor packs in case the controller is to be converted back to a master controller in the future.
5	To reassemble the module, reverse steps 1 - 4. During reassembly, make sure to seat the cable connector completely without twists or kinks in the cable. Carefully guide the LED indicators into their corresponding holes in the front panel.
6	Remove six screws from the left side cover of the module and remove the cover.
7	Remove the following resistor packs: U28 - 680 ohm U37 - 680 ohm U30 - 680 ohm U24 - 680 ohm U13 - 680 ohm
8	Replace the cover.

4.2 Connecting Multiple Crates to a GPIB Interface

The 5488 is designed to allow multiple crates, each with a 5488 controller, to be connected to a single GPIB interface.

The interface distinguishes between the controllers along the bus by means of GPIB device addresses. Each controller has a unique address as determined by switches on the module. When the interface sends a command, the modules compare their addresses with the address in the first byte of the command. If the addresses match, the controller executes the command. If not, it ignores the command.

As it comes from the factory, the 5488 command device address is set to 16 and its block transfer address is 17. To change these addresses, use the following procedure:

Step	Procedure
1	Orient the module so that the left side cover is up and the front panel is to the left.
2	The crate number switches are accessed through the hole in the upper center of the cover. To set a number, position the switches as described in Table 4-1, below. The ON position is in the direction of the arrow on the switch body.

Table 4-1

Address		Switch			
Command	Block Xfer	5	4	3	2
0	1	ON	ON	ON	ON
2	3	ON	ON	ON	OFF
4	5	ON	ON	OFF	ON
6	7	ON	ON	OFF	OFF
8	9	ON	OFF	ON	ON
10	11	ON	OFF	ON	OFF
12	13	ON	OFF	OFF	ON
14	15	ON	OFF	OFF	OFF
16	17	OFF	ON	ON	ON
18	19	OFF	ON	ON	OFF
20	21	OFF	ON	OFF	ON
22	23	OFF	ON	OFF	OFF
24	25	OFF	OFF	ON	ON
26	27	OFF	OFF	ON	OFF
28	29	OFF	OFF	OFF	ON
30	31	OFF	OFF	OFF	OFF

Switch 1 sets the 5488 for high or low speed mode, as follows:

ON = low speed
OFF= high speed

In low speed mode, the 5488 limits its transfer rate to about 500K bytes per second. Most GPIB interfaces will not run any faster than this, so the switch is set to ON at the factory. If your interface can handle transfer rates in excess of 500K bytes per second, set switch 1 to OFF.

4.3 Hold Line

The hold line allows the 5488 to hold (lengthen) a CAMAC cycle to accommodate slow modules. Bringing the hold line low holds the current CAMAC cycle until the line is brought high again. As it comes from the factory, the hold line is assigned to CAMAC dataway line P2. To change the hold line to another dataway line, use the following procedure:

Step	Procedure
1	Remove six screws from the left side cover and remove the cover.
2	Remove the jumper between E1 and E2 near the upper end of the CAMAC dataway connector.
3	Replace the cover and screws.
4	Disassemble the crate and power supply assembly to the extent necessary to expose the motherboard.
5	Install a jumper between the P2 line on the control slot and the dataway line to be used as the hold line.
6	Reassemble the crate and reinstall the controller.

4.4 Transfer Order Jumper

As supplied, the 5488 transfers the high-order byte of multi-byte data items first. This conforms to the convention used in Hewlett-Packard and other computers. To use the 5488 with those computers, such as the DEC PDP-11 series, that transfer the low-order byte first, a jumper must be removed from the module. To do this, use the following procedure:

Step	Procedure
1	Remove six screws from the left side cover and remove the cover.
2	Remove the jumper between E3 and E4 near the middle of the upper edge of the board.
3	Replace the cover and screws.

This jumper affects the order of all multi-byte data transfers in either direction between the GPIB controller and the 5488 except for N, A and F in CAMAC commands.

APPENDIX
Command Summary

Command	Mnemonic	N	A	F
Read Status Register	RD2	30	0	1
Read LAM Status Register	RD2	30	12	1
Read LAM Mask Register	RD2	30	13	1
Read LAM Status Register	RD2	30	14	1
Write Status Register	WT2	30	0	17
Write LAM Mask Register	WT2	30	13	17

After any CAMAC command has been executed, the controller status can be retrieved by executing a serial poll at the 5488's device selector.

ERRATA
Model 5488 GPIB Crate Controller
Operation Manual
Document No. A4880C

In section 3.2.4, Block Transfers (page 16), the paragraph concerning ACA mode block data transfers is in error. Replace this paragraph with the following:

ACA - Address Scan mode. The GPIB controller designates a module number and subaddress at which to start transfering data. After each transfer, the subaddress is incremented and another transfer is attempted. If Q=0 at any subaddress or after the transfer at A(15), the subaddress is reset to zero and the module number is incremented. After the last transfer from module number 24, transfer continues at module number 1. Transfer continues until stopped by the GPIB controller. Note that X=0 does not stop ACA mode transfers.