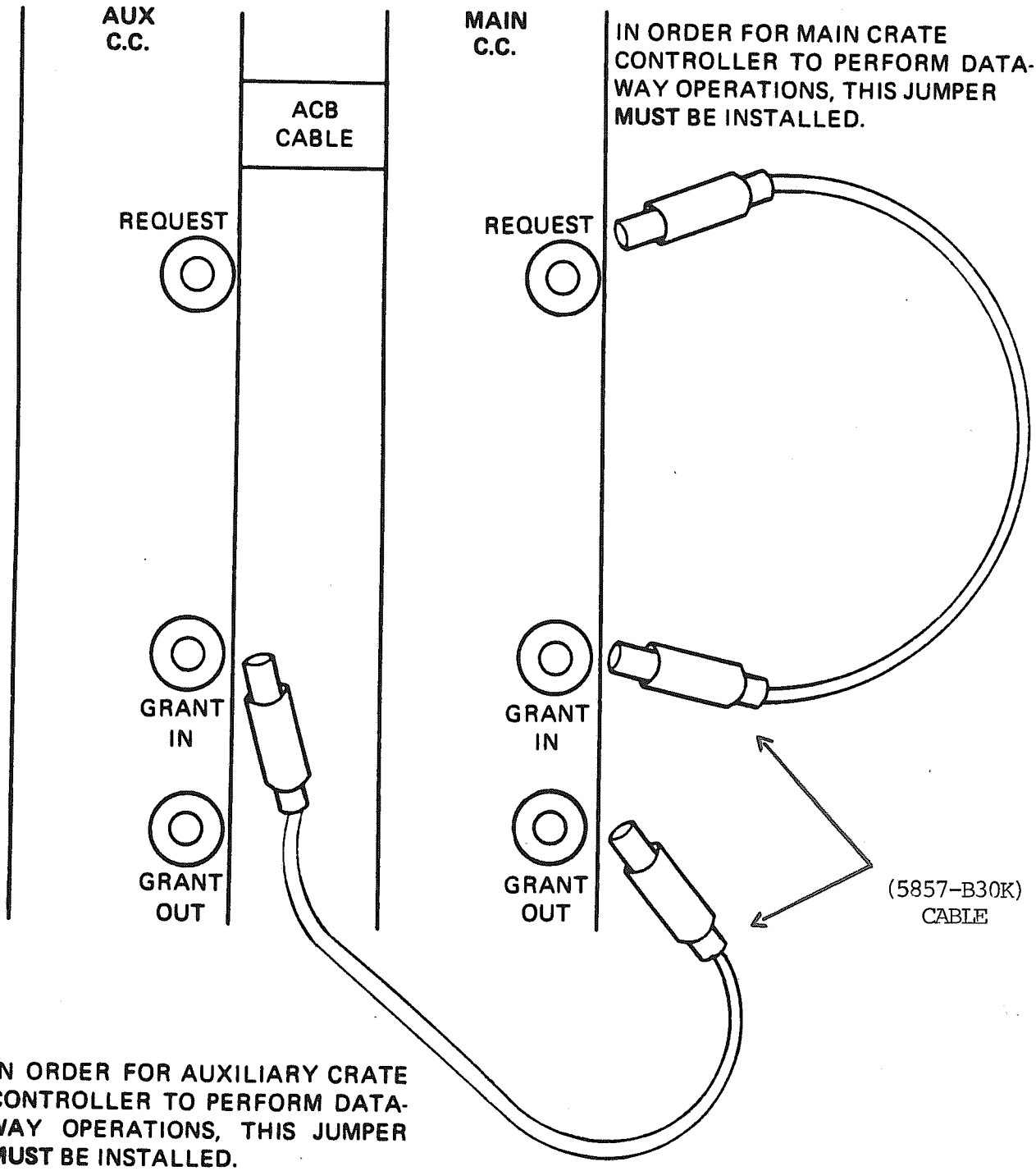
Model 3988-G3A/D3A

GPIB Crate Controller

INSTRUCTION MANUAL

December, 1997

SECTION OF FRONT PANEL



IN ORDER FOR AUXILIARY CRATE CONTROLLER TO PERFORM DATA-WAY OPERATIONS, THIS JUMPER MUST BE INSTALLED.

AND THE ACB CABLE MUST BE CONNECTED BETWEEN THE MAIN AND AUXILIARY CONTROLLERS.

NOTICE:

Some GPIB controllers, notably the Hewlett Packard desktop terminals, automatically insert a carriage return/line feed at the end of a message string. The 3988 uses only EOI true for message termination. A carriage return/line feed will be seen by the 3988 as part of another message and must therefore be suppressed.

Typical HP-85 input/output statements:

```
OUTPUT 701 USING "#, B"; 2, 0, 16, 255, 0, 64

     #--Suppress CR, LF
     B--Binary string

     in this example N=2, A=0, F=16
          high data   = 255
          middle data =   0
          low data    =  64

ENTER 701 USING "#, B"; A, B, C

WHERE
     A = High byte
     B = Middle byte
     C = Low byte
```

TABLE OF CONTENTS

GPIB Crate Controller

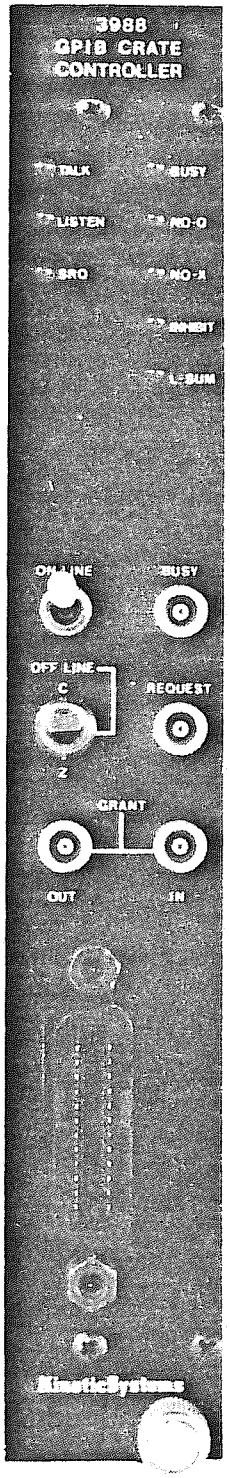
Interfaces an IEEE-488 controller to a CAMAC crate **3988**

Features

- **Provides an interface between a GPIB system and a CAMAC crate**
- **Supports data transfer rates up to 600 Kilobytes per second**
- **Acts as a main or auxiliary crate controller**
- **Meets all IEEE-488 requirements**
- **Supports Q-Scan, Q-Stop, and Q-Repeat block data transfers**
- **Contains a switch-selectable talk/listen address**

Typical Applications

- **Laboratory automation**
- **Process control**
- **CAMAC I/O in a GPIB-controlled system**
- **Local GPIB control of a distributed CAMAC system**



(Product specifications and descriptions in this document subject to change without notice.)

General Description

The 3988 is a double-width CAMAC crate controller that acts as a device on the General Purpose Interface Bus (GPIB). This crate controller conforms to all applicable CAMAC specifications as well as those for the GPIB (IEEE Standard 488). The 3988 allows GPIB system controllers to communicate with up to 23 CAMAC I/O modules in a crate.

Two new ordering options are provided. The Model 3988-G3A uses the IEEE-488 24-contact ribbon connector, while the 3988-D3A uses the IEC (European) 25-contact "D" connector. These new options support block transfers at data rates up to 600 kilobytes per second with the actual rate limited only by the controller or the host computer. The 3988-G3A/D3A supersede the 3988-G2A/D2A and are software-compatible with these earlier lower-speed units.

The 3988 functions as a main or auxiliary crate controller. When used as a main crate controller, it can support one or more auxiliary crate controllers. When used as an auxiliary crate controller, the 3988 shares control of the crate with the main crate controller.

The 3988 is configured at the factory as a main crate controller. It can be changed in the field to function as an auxiliary crate controller by removing plug-in resistor packages. A 40-contact, rear-panel connector is then used for the Auxiliary Controller Bus.

Internal Function Codes (N = 30)

Command		Action
F(0)·A(0)	RD1	Reads Transfer Count register.
F(1)·A(0)	RD2	Reads Control/Status register.
F(1)·A(12)	RD2	Reads LAM Request register.
F(16)·A(0)	WT1	Writes Transfer Count register.
F(16)·A(1)	WT1	Writes SRQ Mask register.
F(17)·A(0)	WT2	Writes Control/Status register.
F(17)·A(13)	WT2	Writes LAM Mask register.
Note: The Invalid Transfer bit (IT) of the status byte = 0 for all valid addressed commands; IT = 1 otherwise.		

Operating Modes

All command and data information are passed between the host computer and the 3988 in binary form, not ASCII characters. Read and Write data can be transferred in 8, 16, or 24 bit form (one, two, or three GPIB bytes), as determined by the data in the CSR (Control/Status Register). The various operating modes are described as follows.

Single Transfers

The Single Transfer operating mode is used to execute single CAMAC commands. The information passed to the 3988 and the response from the 3988 are:

1. From host to 3988: NAF (Station number, Subaddress, Function code) and Write data (Write operations only).

2. From 3988 to host: Read data (Read operations only) and status byte response (if enabled) with Q, X, and other status information.

Q-Scan

The Q-Scan operating mode is used to transfer a block of Read or Write data between a group of modules and the GPIB host computer. The standard CAMAC Q-Scan techniqueas defined by IEEE 683 is used. The Function Code, initial Subaddress, and initial Station number are passed to the 3988 in binary form. For Write commands, the Write data block follows the command information. For Read commands, the block of Read data follows. The operation is terminated by a transfer count underflow or by a scan-out-of-crate (N > 23) condition.

Q-Stop/Stop on Transfer Count

The Q-Stop/Stop on Transfer Count mode is similar to Q-Scan, except that the data transfers are to or from a single target NAF (for example, Reading or Writing a block of memory in a module). The CAMAC command is repeated until a Q = 0 response or a transfer count underflow is reached.

Q-Repeat

Q-Repeat operating mode is also used with a single NAF. The number of valid transfers is determined by a transfer count underflow. Data is ignored (and the transfer count is not incremented) for all Dataway operations resulting in Q = 0.

Service Request Capability

The 3988 has full service-request capability. It can issue a GPIB Service Request (SRQ) for the following conditions: Transfer Ended, CAMAC LAM (Look-at-Me), Q = 0, X = 0, ON-Line, Inhibit, and Invalid Transfer. When the GPIB host responds to a Service Request, it performs a serial poll and obtains the information from the Service Request Status register in the 3988. The 24-bit LAM pattern can also be obtained by reading the LAM Request register.

Power Requirements

+6 volts:	3.7 A

Ordering Information

Model 3988-D3A	GPIB Crate Controller with IEC (Europe) connectors
Model 3988-G3A	GPIB Crate Controller with IEEE 488-1978 (USA) connectors

Related Products

Model 3982-Z1x	Auxiliary Crate Controller, autonomous list sequencing, FP Parallel Bus and R/W buffer
Model 5843-Series	Auxiliary Controller Bus Cable Assembly
Model 5864-Series	GPIB Bus Cable Assemblies

PROGRAMMING THE CAMAC SYSTEM

The following provides a brief introduction to the parameters involved in programming a CAMAC system. A more detailed description is available in the IEEE Std. 583-1975, Modular Instrumentation and Digital Interface System.

The CAMAC standard provides guidelines for control of the 25-station CAMAC crate by a crate controller residing in stations 24 and 25.

Communication between the crate controller and the modules is accomplished via Dataway commands. Three parameters common to the Dataway commands are: the Function Code, the Subaddress, and the Station Number. The commands can be separated into three major groups: the Read commands, the Write commands, and the Control commands. The crate controller generates the necessary timing signals to execute the commands.

Status information is conveyed from the module to the crate controller via the Look-At-Me (LAM) line and the Q Response and Command Accepted busses. Each station in the crate has a LAM line. The module may use the LAM to asynchronously inform the crate controller that it requires attention. The Q and X bus are used by the addressed module to convey status during the Dataway command cycle.

There are three common control signals on the CAMAC Dataway. These are Initialize, Clear and Inhibit.

FUNCTION CODE

The Function Code (F), ranging from 0_{10} to 31_{10}, determines which of three command groups a particular command belongs to.

The Read Command group includes commands with F ranging from 0_{10} to 7_{10}. Execution of a read command initiates the transfer of data from the addressed module to the crate controller. Data is transferred on the 24-bit read bus (R1-R24). Modules which do not use the full 24-bit word, such as a 12-bit DAC, will normally use the lower segment of the word, and the unused segment will be read as zero.

The Write Command group consists of those commands with F greater than 15_{10} and less than 24_{10}. Execution of a write command initiates the transfer of data from the crate controller to the addressed module. Data is transferred on the 24-bit write bus (W1-W24).

The Control Command group function code includes: $7_{10} < F < 16_{10}$ and $23 < F < 32_{10}$. No data is transferred on the read or write bus during control command executions, however, one bit of status information can be sent to the crate controller via the Q response line. An example is the F(8) Test Look-At-Me (LAM) command. The addressed module may return the state of its LAM request to the crate controller via the Q response line. Other commands in this group include the F(24) Disable, the F(25) Initiate, and the F(26) Enable command.

THE SUBADDRESS

The Subaddress (A), ranging from 0_{10} to 15_{10}, is used to access different sections of an addressed module. An example is an eight-channel D/A Converter which is read by the F(0)·A(i) command, where i can range from 0_{10} to 7_{10}. This provides access to all eight channels' data registers with the F(0), Read Group 1 Register, function code.

THE STATION NUMBER

The Station Number (N) specifies which slot is intended to receive the command being executed. N can range from 1_{10} to 23_{10} for the modules within the CAMAC crate. Internal features of crate controllers typically use N=30.

STATUS INFORMATION

During a Dataway command execution cycle, the addressed module uses the Q Response (Q) and X Command Accepted (X) bus lines to convey status information to the crate controller. X=1 is generated by the addressed module if it recognizes the command being executed as one that it is equipped to perform. The Q Response may reflect the status of any selected feature in the module.

The Look-At-Me (LAM) signal may be used by a module to inform the crate controller that it needs attention. The LAM signal reflects the state of the LAM request in the module. The LAM request reflects the state of the LAM status and whether or not LAM's are enabled in the module. LAM's are enabled within a module by an Enable command or by setting a bit in a LAM Mask register. LAM requests are disabled at power-up.

COMMON CONTROL SIGNALS

The Initialize (Z) sets registers or control functions in a module to an initial state, particularly when power is turned on.

The Inhibit (I) disables features for duration of the signal. The Inhibit line can be set by the crate controller or a module. An example of this is using a Timing Pulse Generator (Model 3655) to provide a time base for a Counter (Model 3610 or 3615). The 3655 can assert Inhibit at the end of a programmed time, disabling the counters.

The Clear (C) clears the registers or resets flip-flops.

The use of the Z, C and I is dependent on the individual module.

3988 OPERATING MODES

Some general rules apply to all operating modes:

1. System must be on-line for all CAMAC commands to modules in crate.

2. Status byte sent back once when enabled in block transfer modes.

3. EOI sent back with the last byte.

4. All data sent to, or read from, the 3988 is in binary, not ASCII characters.

5. The order in which data bytes are sent to the 3988 is as follows:

    1. High byte (W24 - W17)
    2. Middle byte (W16 - W9)
    3. Low byte (W8 - W1)

6. The order in which data bytes are read from the 3988 is as follows:

    1. High (R24 - R17)
    2. Middle (R16 - R9)
    3. Low (R8 - R1)

7. The order in which the command information is written to the 3988 is as follows:

    1. Station number (N)
    2. Subaddress (A)
    3. Function Code (F)

SINGLE TRANSFERS

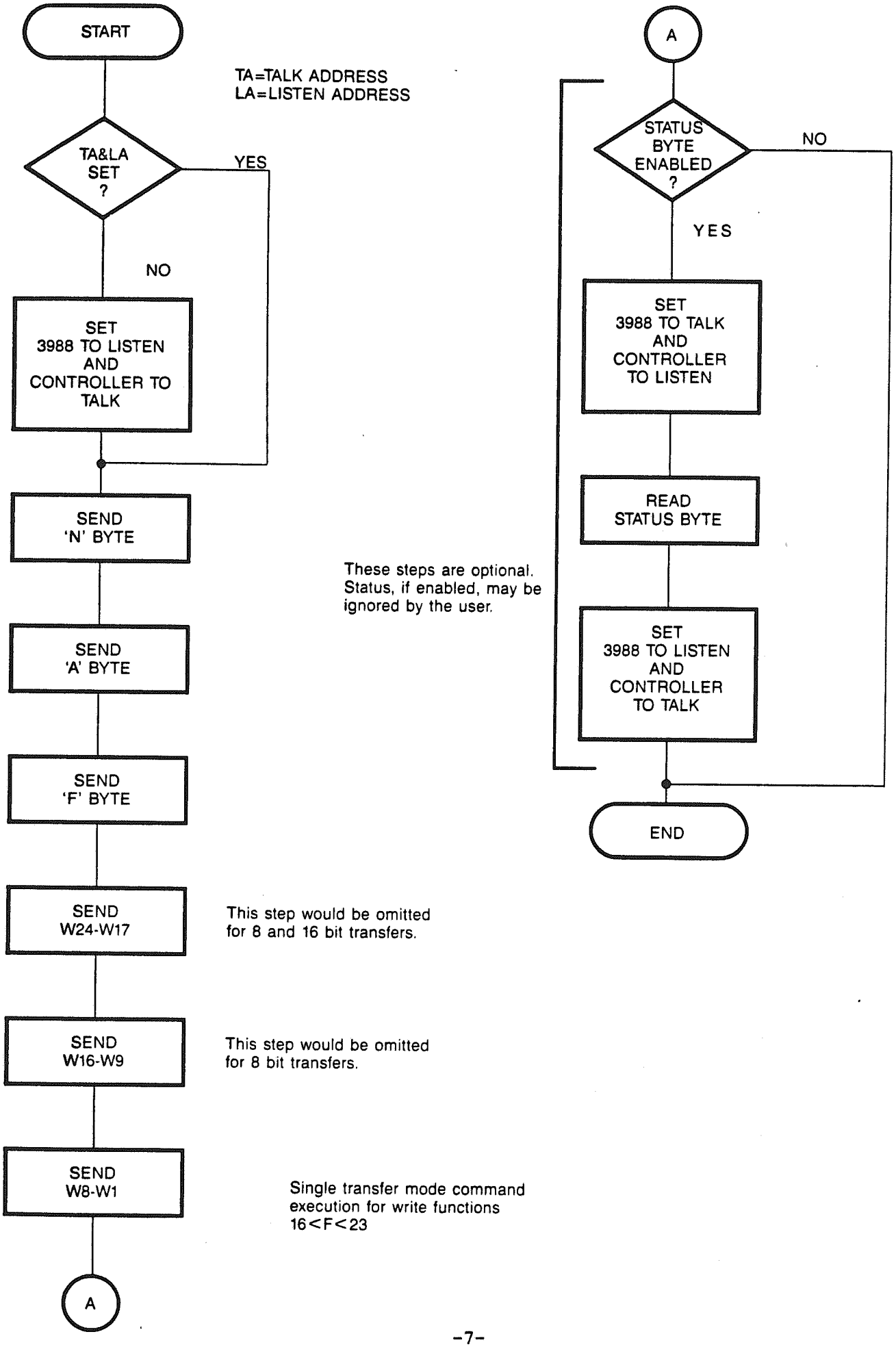
The single transfer mode allows execution of single CAMAC commands.

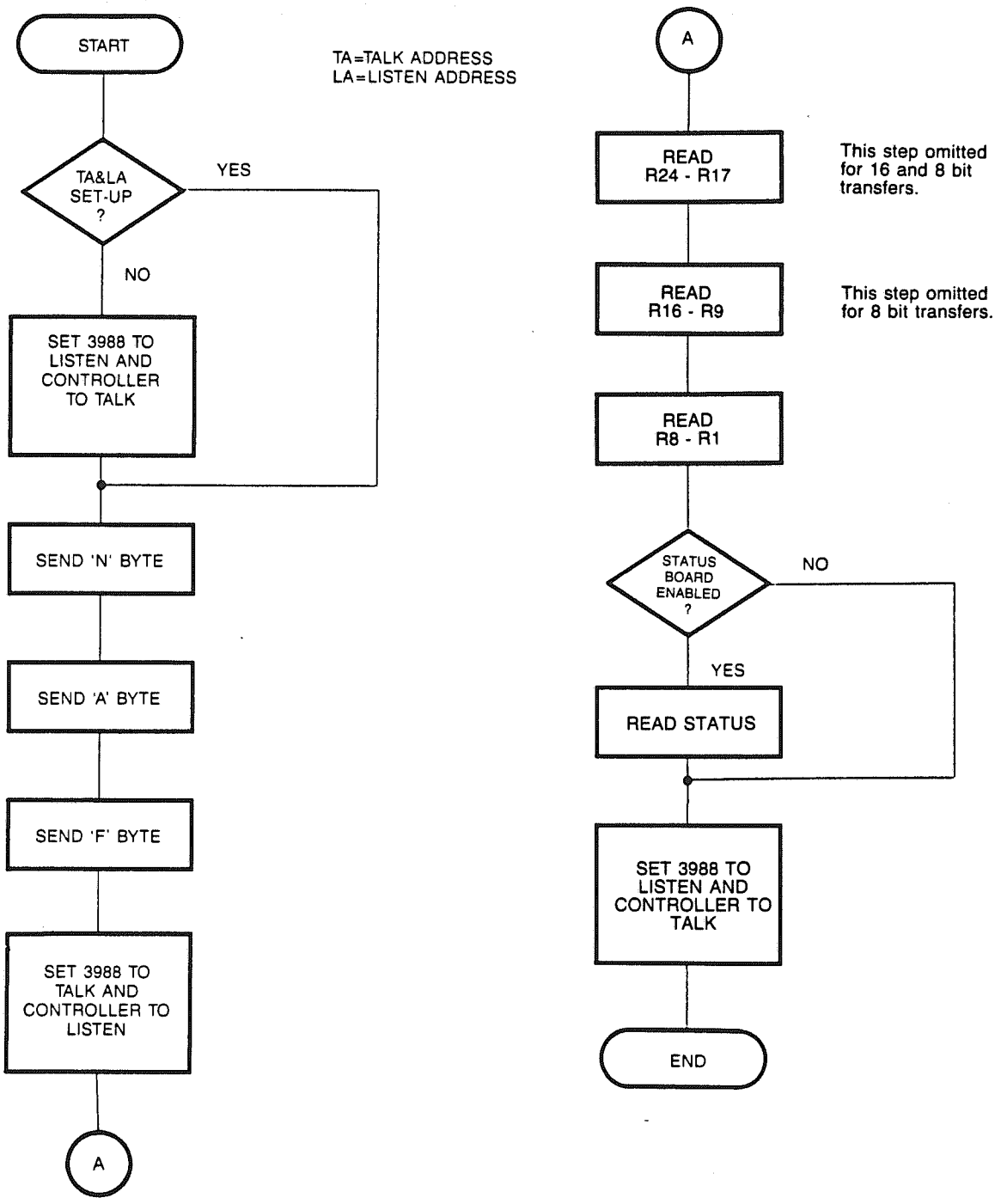
If the function code indicates that a CAMAC "write" is to take place the 3988 will wait for 8, 16, or 24 bits of data, as specified by BT1 and BT2 in the control status register. When the last byte of write data is received the Dataway Cycle will occur. If the status byte is enabled, it will be returned when the 3988 is next made a TALKER.

If a CAMAC "read" is described by the function code a Dataway Cycle will be executed following the acceptance of the function code byte. The number of bits returned is determined by BT1 and BT2 in the control status register. If the status byte is enabled, it will follow the data bytes in the return sequence.

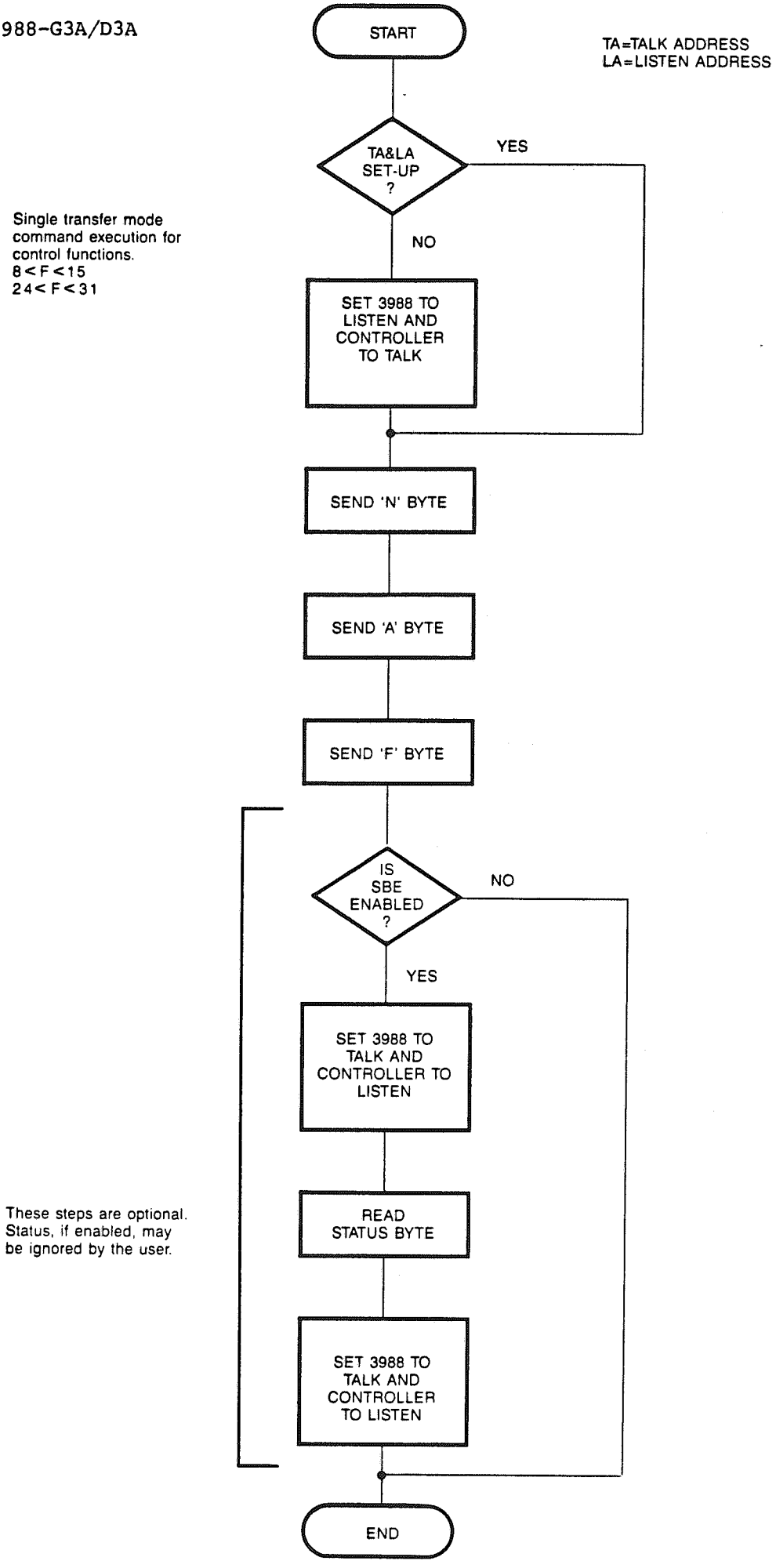
If the function indicated is a control command the 3988 will perform the Dataway Cycle after receiving the 'F' byte. If enabled, a status byte will be returned when the 3988 is next made a TALKER.

Flowcharts on the next three pages show how to perform single transfer write, read, and control function CAMAC commands.

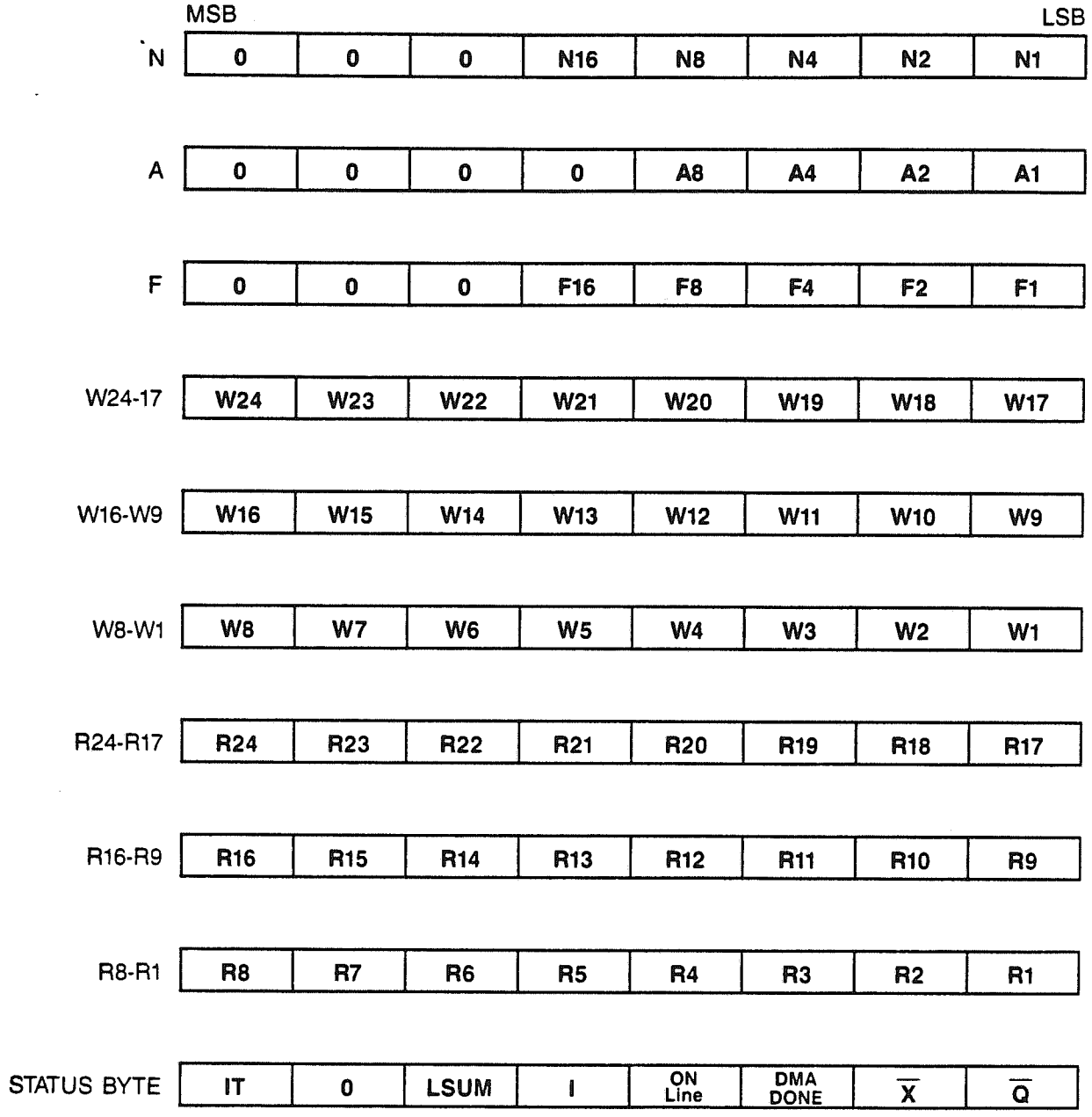

START
TA=TALK ADDRESS
LA=LISTEN ADDRESS
TA&LA SET ?
YES
NO
SET 3988 TO LISTEN AND CONTROLLER TO TALK
SEND 'N' BYTE
SEND 'A' BYTE
SEND 'F' BYTE
SEND W24-W17
This step would be omitted for 8 and 16 bit transfers.
SEND W16-W9
This step would be omitted for 8 bit transfers.
SEND W8-W1
Single transfer mode command execution for write functions 16<F<23
A
A
STATUS BYTE ENABLED ?
NO
YES
SET 3988 TO TALK AND CONTROLLER TO LISTEN
READ STATUS BYTE
SET 3988 TO LISTEN AND CONTROLLER TO TALK
These steps are optional. Status, if enabled, may be ignored by the user.
END



Single transfer mode command execution for Read functions
0<F<7


START
TA=TALK ADDRESS
LA=LISTEN ADDRESS
TA&LA SET-UP ?
YES
NO
Single transfer mode command execution for control functions.
8<F<15
24<F<31
SET 3988 TO LISTEN AND CONTROLLER TO TALK
SEND 'N' BYTE
SEND 'A' BYTE
SEND 'F' BYTE
IS SBE ENABLED ?
NO
YES
SET 3988 TO TALK AND CONTROLLER TO LISTEN
READ STATUS BYTE
These steps are optional. Status, if enabled, may be ignored by the user.
SET 3988 TO TALK AND CONTROLLER TO LISTEN
END

CAMAC COMMAND BYTE STRUCTURES



An example of a 24 bit write to slot number 2 after Power-up. (On Power-up, all the 3988 internal registers are set to zero).

1. Enable GPIB Attention
2. Set 3988 to LISTEN
3. Set GPIB Controller to TALK
4. Disable GPIB Attention
5. Send 6 bytes for Dataway write command and data. In this example, N=2, A=0, F=16, DATA HIGH=3, DATA MIDDLE=7, DATA LOW=15. The sequence sent is:

   2 0 16 3 7 15

   With the reception of DATA LOW, the 3988 will perform the Write Command to slot #2.

An example of a 16 bit write to slot #2 is as follows:

1. Enable GPIB Attention
2. Set 3988 to LISTEN
3. Set GPIB Controller to TALK
4. Disable GPIB Attention
5. Write 3988 CSR to 16 bit transfers. (N=30, A=0, F=17, DATA HIGH=0, DATA MIDDLE=1, DATA LOW=0)
6. Write to slot #2 with 16 bits (N=2, A=0, F=16, DATA MIDDLE=1, DATA LOW=3), where DATA MIDDLE is the upper 8 bits of the 16 bit value and DATA LOW is the lower 8 bits. If slot #2 was a 24-bit register, it would have the following data in it:

STATUS BYTE

The Status Byte reflects the conditions resulting from the last operation, either internal (N=30) or external (CAMAC dataway). This byte can be returned after every operation by setting the Status Byte Enable (SBE) in the Control Status Register (CSR). If enabled, status will be returned at the end of a READ data string as the last byte. When executing WRITE or CONTROL functions, the 3988 will send status when placed in the TALK mode (if Status Byte is enabled). The user is not required to read the Status returned from a WRITE or CONTROL function. (See flow chart examples).

The Status Byte, when sent, will always be accompanied by an EOI signal.

STATUS BYTE REGISTER:

	8	7	6	5	4	3	2	1
STATUS BYTE	IT	RSV	L-SUM	I	ON LINE	TCR=0	NO-X	NO-Q

Bit	Mnemonic	Description
1	NO-Q	A No-Q condition exists
2	NO-X	A NO-X condition exists
3	TCR=0	Transfer Count Register (TCR) = 0
4	On-Line	3988 is on-line
5	I	Inhibit is set in the crate
6	L-Sum	There is an un-masked LAM set in the crate
7	RSV	3988 has enabled a GPIB SERVICE REQUEST
8	IT	Invalid Transfer, the last command was unrecognized, i.e., N>23 (unless N=30), F>31

DMA TRANSFER

The 3988 is capable of performing three different modes of Block Data Transfer. These modes are ADDRESS SCAN, Q-STOP, and Q-REPEAT. The user may set the desired mode by writing to the 3988 Control Status Register (CSR). The next CAMAC dataway operation (N<24) will be executed in this mode. Internal functions (N=30) are not affected by the CSR. The 3988 will remain in Block Transfer mode until the user rewrites the CSR with a new mode setting.

The user must also enter a value to the 3988 Transfer Count Register (TCR). This number will be the number of CAMAC dataway cycles to be executed when performing the Block Transfer. See the flow chart examples for DMA Transfers.

At the end of a Block transfer the Status Byte (if enabled) is returned. The Status Byte is not considered part of the Transfer Count and may be read while in the Block Transfer mode without affecting the Transfer Count.

ADDRESS SCAN

The Address Scan mode of block transfer is used for data transfers between an array of modules that do not necessarily occupy consecutive stations in the GPIB system. The initial station number, function code, and initial subaddress are written to the 3988 in binary form. The state of Q during each dataway cycle is used by the 3988 to determine the station number and subaddress for the next dataway cycle. When Q=1, the subaddress is incremented. If the subaddress was previously equal to fifteen (15), the subaddress will be set to zero and the station number incremented. When Q=0, the subaddress is set to zero and the station number is incremented. This allows unoccupied stations to be within an array of modules in a crate. The block transfer is terminated after a specified word is reached or when the station number reaches 24. The number of bits transferred in the address scan is determined by bits BT1 and BT2 in the 3988's control/status register.

ADDRESS SCAN WRITE OPERATION

The write data block follows immediately after the command information (N, A, F). When operating in the Address Scan Write Mode (sending data to the dataway) and N reaches 24 before the transfer count register (TCR) is exhausted the 3988 will absorb any further data and the TCR will contain the number of transfers that did not occur.

ADDRESS SCAN READ OPERATION

When operating in Address scan read mode (receiving data from the dataway) the 3988 will send EOI true on the word after the TCR reached zero. This last word will contain invalid data and should be discarded. With Status Byte Enabled (SBE) EOI will be sent true with the status byte after N reaches 24 or TCR equals zero. It is recommended that the SBE be set whenever Address scan read mode is used. This allows EOI to be set when N reaches 24 or TCR equals zero. The transfer count register will contain the number of transfers that did not occur as a result of N reaching 24.

STOP ON TRANSFER COUNT MODES

In the Stop on Transfer Count operating modes, Q-stop and Q-repeat, a block of data is transferred to or from a module in the CAMAC crate. N, A and F remain constant and the 3988 will cause the execution of Dataway Cycles until the number of transfers to or from the GPIB controller equals the value written to the Transfer Count register. The number of bits transferred is determined by bits BT1 and BT2 in the Control/Status Register.

Q-STOP MODE (WRITE OPERATION)

The Q-Stop Mode will stop execution of Dataway Cycles if a No-Q condition occurred with the previous cycle. If the No-Q condition occurs before the transfer count is exhausted, the 3988 will absorb the rest of the data sent. The Transfer Count Register (TCR) will contain the number of CAMAC dataway cycles that did not occur, including the cycle that returned NO-Q. (Transfer Count is never decremented unless Q=1).

An example of a Q-STOP WRITE block transfer of 16 bits per transfer to slot #2 is as follows:

1. Enable GPIB Attention
2. Set 3988 to LISTEN
3. Set GPIB Controller to TALK
4. Disable GPIB Attention
5. Write 3988 CSR to Q-STOP mode, 16 bits per transfer.
   (N=30, A=0, F=17, DH=0, DM=17, DL=0)
6. Write 3988 TCR to number of transfers desired (in this case 255).
   (N=30, A=0, F=16, DH=0, DM=0, DL=255).
7. Start the Q-STOP write block transfer by sending N, A, F:
   (N=2, A=0, F=16)
8. Send the Block of Data. The transfer is terminated by the 3988 when the TCR is exhausted, or when a NO-Q is returned (in Q-STOP mode). In this example, the array DATA is sent, where DATA(2,CNT) is the upper 8 bits and DATA(1,CNT) is the lower 8 bits.

Transfer #1	Bytes sent =	DATA (2,1)	DATA (1,1)
Transfer #2		DATA (2,2)	DATA (1,2)
.			
.			
.			
Transfer #255		DATA (2,CNT)	DATA (1,CNT)

9. Remember to write the 3988 CSR to Single Transfer mode when done

Q-STOP MODE (READ OPERATION)

In Q-Stop Read operation data will be read until a NO-Q condition is encountered or until the transfer count is exhausted. EOI will be returned when the TCR equals zero only if the status byte is not enabled. It is recommended that the status byte be enabled during Q-Stop reads. This will guarantee that EOI is sent once the block transfer is terminated. The Transfer Count Register will contain the number of CAMAC Dataway cycles that did not occur.

Q-REPEAT MODE (WRITE OPERATION)

In the Q-Repeat Mode, only Dataway Cycles that return a Q are considered valid. Therefore, the Q-Repeat mode is only terminated after the transfer count is equal to the number of valid transfers that have occurred. The Dataway Cycle is performed after the write data is received, if No-Q occurred it will repeat the cycle again and will continue until Q=1 is received. It will then wait for more write data for the next cycle. This will continue until the transfer count is exhausted.

> **NOTE:** If by some circumstances the 3988 cannot receive a valid Q during a Q-repeat write it will never decrement the TCR, thus not allowing it to exit this mode of operation. The only way to exit is to toggle the GPIB Attention Line or execute a GPIB Interface Clear (IFC).

Q-REPEAT MODE (READ OPERATION)

The Q-Repeat read operation will perform a Dataway cycle and if a valid Q was returned it will send the data back. If NO-Q occurred it will repeat the cycle until Q occurs. EOI will be sent with the status byte (if enabled) at the end of the block transfer (TCR=0). If the status is not enabled, EOI is sent with the byte of data after the last valid byte of information. This byte should be ignored by the user for it does not contain valid data.

> **NOTE:** If by some circumstances the 3988 cannot receive a valid Q during a Q-Repeat read it will never decrement the TCR, thus not allowing it to exit this mode of operation. The only way to exit is to toggle the GPIB Attention line or execute a GPIB Interface Clear (IFC).

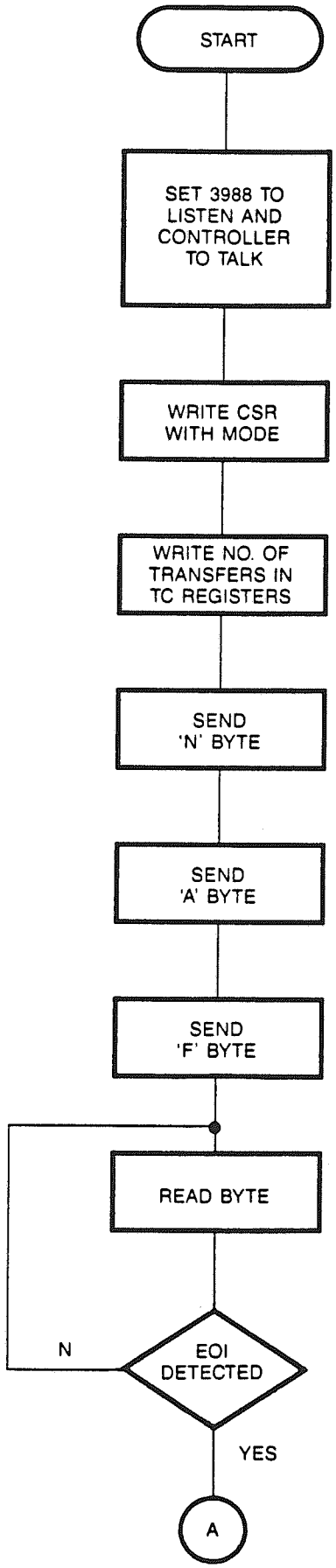
SERIAL POLL

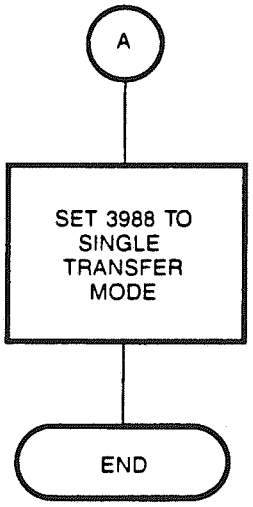
Any device on the GPIB bus requiring action from the controller (the GPIB device that generates the control functions, i.e., ATN, and usually resides inside a computer) can generate the SRQ signal. The controller should request serial poll status from all GPIB devices. Any device whose status word has Bit 7 (1-8) true, generated the SRQ. More than one device could have set the SRQ line true.

SERVICE REQUEST

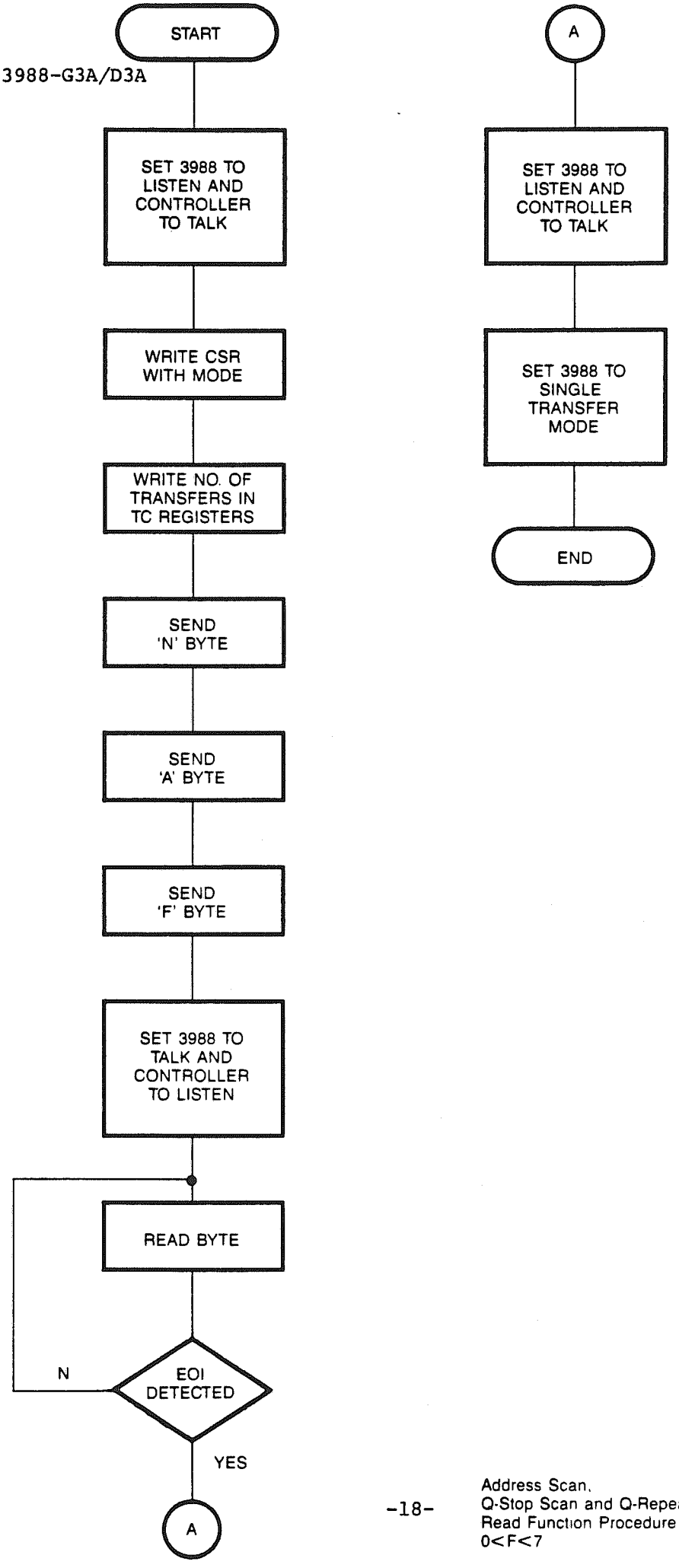
The 3988 is able to issue a GPIB SERVICE REQUEST (SRQ) for several conditions. After setting the SRQ signal, the 3988 will continue to operate in a normal fashion until a SERIAL POLL ENABLE is executed by the GPIB Controller. When a SERIAL POLL ENABLE is executed and the 3988 is placed in the TALK mode, the 3988 will place one byte of status on the GPIB. This byte is of the same format as the Status Byte of the 3988.

There exists a SERVICE REQUEST MASK REGISTER, allowing the user to decide which, if any, of the conditions given can cause a SRQ to be issued. The status byte returned is the full status byte of the 3988. This byte should be "AND"ed in software with the Service Request mask to determine which condition(s) caused the SRQ.





Address Scan,
Q-Stop Scan and Q-Repeat
Write Function Procedure
16<F<23



Address Scan,
Q-Stop Scan and Q-Repeat
Read Function Procedure
0<F<7

An example of a Q-STOP block transfer of 16 bits per transfer to slot #2 is as follows:

1. Enable GPIB Attention

2. Set 3988 to LISTEN

3. Set GPIB Controller to TALK

4. Disable GPIB Attention

5. Write 3988 CSR to Q-STOP mode, 16 bits per transfer

   (N=30, A=0, F=17, DH=0, DM=17, DL=0)

6. Write 3988 TCR to number of transfers desired (in this case 255):

   (N=30, A=0, F=16, DH=0, DM=0, DL=255).

7. Start the Q-STOP write block transfer by sending N, A, F:

   (N=2, A=0, F=16)

8. Send the Block of Data. The transfer is terminated by the 3988 when the TCR is exhausted, or when a NO-Q is returned (in Q-STOP mode). In this example, the array DATA is sent, where DATA(2,CNT) is the upper 8 bits and DATA (1,CNT) is the lower 8 bits.

Transfer #1	Bytes sent =	DATA (2,1)	DATA (1,1)
Transfer #2		DATA (2,2)	DATA (1,2)
.			
.			
.			
Transfer #255		DATA (2,CNT)	DATA (1,CNT)

9. Remember to write the 3988 CSR to Single Transfer mode when done.

SERIAL POLL SEQUENCE

If the 3988 issues an SRQ, this signal will remain true until a GPIB IFC, or the condition that caused the SRQ is removed. The Serial Poll sequence is defined as follows:

1. Enable GPIB Attention
2. Place 3988 to TALK
3. Place GPIB controller to LISTEN
4. Enable GPIB SERIAL POLL (SPE)
5. Disable GPIB Attention
6. Read one byte of SRQ status from the 3988. (Bit 7 will be on if the 3988 caused a SRQ.)
7. Enable GPIB Attention
8. Disable GPIB SERIAL POLL (SPD)
9. Disable GPIB Attention
10. Determine the cause of the 3988 Service Request by removing bits not masked on in the SRQ mask register.
11. Correct the condition that caused the SRQ. (Service the LAM, remove Inhibit, etc.), or rewrite the SRQ Mask to disable the condition that caused the request (i.e., if the module being set to on-line caused the SRQ, rewrite SRQ Mask with bit 4 off. This will make SRQ false.

SRQ STATUS
(Returned during a SERIAL POLL)

8	7	6	5	4	3	2	1
IT	RQS	L-SUM	I	ON LINE	TCR=0	NO-X	NO-Q

Bit	Mnemonic	Description
1,2	NO-Q, NO-X	SRQ was set after a Dataway Cycle because of a NO-Q or NO-X condition.
3	TCR=0	SRQ was set when the transfer count equaled zero. (TRANSFER COMPLETE)
4	ON-LINE	SRQ is set when 3988 is ON-LINE.
5	INHIBIT	SRQ was set when INHIBIT line is detected true
6	L-SUM	SRQ was set when an UN-MASKED LAM was detected in the crate
7	RQS	Indicates that the 3988 set the SRQ line
8	IT	SRQ was set when 3988 received an invalid command, i.e., N>23 (except N=30), N ≤ 31 or an illegal internal NAF function

BLOCK TRANSFER SRQ

A SRQ may be serviced in the middle of a block transfer. The transfer may continue after a Serial Poll. The procedures is as follows:

1. SRQ detected by the user. Place the 3988 in single transfer mode if not in this mode already.

2. Read and save TCR, N=30, F=0, A=0

3. Perform Interrupt Service Routine:
   Serial Poll, Single Transfers, LAM Request Reads, etc.

4. Write TCR with saved value N=30, F=16, A=0

5. Write CSR with Block Transfer Mode. This step not required when CSR remanins unaffected by the Interrupt Service Routines.

6. Continue Transfer by writing NAF (and data if WRITE) of the next Block Transfer command.

INTERNAL REGISTERS

The 3988 provides five internal registers for the purpose of control and communicating status information to the operator. The registers are as follows: The Transfer Count Register (TCR), the Control/Status Register (CSR), the LAM Request register, the disable LAM mask register and the SRQ Mask register. Accessing the internal registers is accomplished in the same manner as single transfers, however, all transfers to these registers must always include 24 bits of data. Each register powers up in a cleared state. In Table 3 are the internal function codes to access the internal registers. It is important to note that station number, N, must be 30 to read or write the registers internal to the 3988.

Table 3:
INTERNAL FUNCTION CODES (N=30)

Command		Description
F(0)·A(0)	RD1	Reads Transfer Count register
F(1)·A(0)	RD2	Reads Control/Status register
F(1)·A(12)	RD2	Reads LAM Request register
F(16)·A(0)	WT1	Writes Transfer Count register
F(16)·A(1)	WT1	Writes SRQ Mask register
F(17)·A(0)	WT2	Writes Status register
F(17)·A(13)	WT2	Writes LAM Mask register

Status Byte will include:
IT = 0 For all valid addressed commands
IT = 1 For invalid commands

Some general rules that apply to internal function codes:

1. Q, X and CAMAC DONE are undefined.
2. The Transfer Count Register is not decremented and remains unchanged.
3. The System does not have to be on-line.
4. Status will be sent anytime SBE is set. Status may be ignored by the user for WRITE or CONTROL functions.
5. Regardless of the Control Status Register setting, these internal functions are 24 bits, single transfer.
6. EOI is sent with the last byte, or with the Status Byte (if enabled).
7. The station number, N, always equals 30 for all internal commands.

CONTROL STATUS REGISTER

24 15	14	13	12	11	10	9
UNUSED	M3	M2	M1	SBE	BT2	BT1
8	7	6	5	4	3	2	1
Z	C	SI	I	ON LINE	DMA DONE	NO-X	NO-Q

The Control/Status register supplies the operator with information concerning the operation of the 3988. It also is used to control the function of the unit.

Given below is a list of the Control/Status register bits and their functions.

CONTROL/STATUS REGISTER BITS

Bit	Mnemonic	Description
1, 2	NO-Q, NO-X	The NO-Q and NO-X bits are read only bits. They indicate the state of Q and X response to the last CAMAC command executed.
3	DMA DONE	DMA Transfer complete (TCR=0).
4	ON-LINE	This bit is read only. It reflects the state of the on-line switch on the front panel. CAMAC commands will execute only if the system is on-line. Internal commands are unaffected by this switch.
5	I	Inhibit - This bit is read only. It signifies the state of Inhibit on the Dataway independent of the source of the Inhibit.
6	SI	Source Inhibit - Setting the SI bit to a logical "1" causes the 3988 to assert the CAMAC I, Dataway Inhibit. Clearing this bit to a logical "0" will cause the 3988 to stop asserting the Inhibit signal.
7	C	Setting this bit to a "1" causes a CAMAC C, clear, to be issued. The bit is cleared immediately following the CAMAC C.

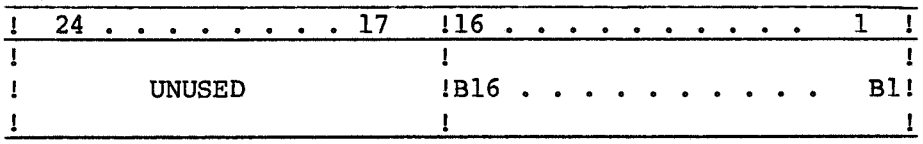
Bit	Mnemonic	Description
8	Z	A CAMAC Z, initialize, will be done on the Dataway following the setting of this bit. After initialization is completed the Z bit will be cleared.
9,10	BT1, BT2	These two bits determine the number of data bits transferred. The 3988 is capable of 8, 16, or 24 bit transfers.
11	SBE	Status Byte Enable - When set to a logical "1" the 3988 will send a byte of data corresponding to the results of the previous command. This byte of data will be sent after each Dataway Cycle but only once during multiple transfers. If a CAMAC read was done the Status Byte will follow the read data.
12,13, 14	M1, M2, M3	These three bits specify which transfer mode the 3988 will use.
15-24		These bits are not used and should be sent as "0"'s.

BT2	BT1	Bits Transferred
0	0	24
0	1	16
1	0	8
1	1	Not defined

M3	M2	M1	Mode
0	0	0	Single Transfer
0	0	1	Address Scan
0	1	0	Q-Stop Scan
0	1	1	Q-Repeat Scan

TRANSFER COUNT REGISTER

The Transfer count register is used in the Stop on Transfer Count mode. The number of transfers desired is written into this register by a N(30)·F(16)·A(0) and can be read with a N(30)·F(0)·A(0). The TCR is decremented with each successful (Q, X returned) transfer between the Dataway and the 3988. The operation is terminated when this register reaches 0. The Transfer count register is configured as shown below. The TCR is not decremented by internal commands (N=30). The TCR is cleared on power-up.



LAM REQUEST REGISTER

24	23 1
	23 1

The LAM Request Register when read will return the LAM pattern generated by the modules in the crate. The LSB, bit 1, corresponds to the LAM for station number 1, bit 2 for LAM 2, and bit 24, the MSB, for LAM 24. The LAM Request Register is not affected by the Disable LAM Mask Register.

DISABLE LAM MASK REGISTER

A Service request can be generated by a LAM through the LSUM Signal. To disable individual LAMs from setting the Service Request, a Disable LAM Mask is provided. On power-up, this mask is cleared enabling all LAMs to set the LSUM bit. Placing a "1" in the mask bit causes the LAMs from the corresponding slot location to be ignored. The contents of the LAM Request Register are not affected by the Disable LAM Mask Register.

SRQ MASK REGISTER

24 9							
UNUSED							
8	7	6	5	4	3	2	1
IT	UNUSED	SLP	I	ON LINE	TCR=0	NO-X	NO-Q

As previously described, certain conditions are capable of asserting a SRQ. The SRQ Mask Register gives the operator control of which of these signals actually will send the SRQ signal. To enable the ability of any of the above conditions, a "1" must be written to its corresponding bit in the SRQ Mask Register.

IEEE 488-1975 Connector
AMP 57-20240

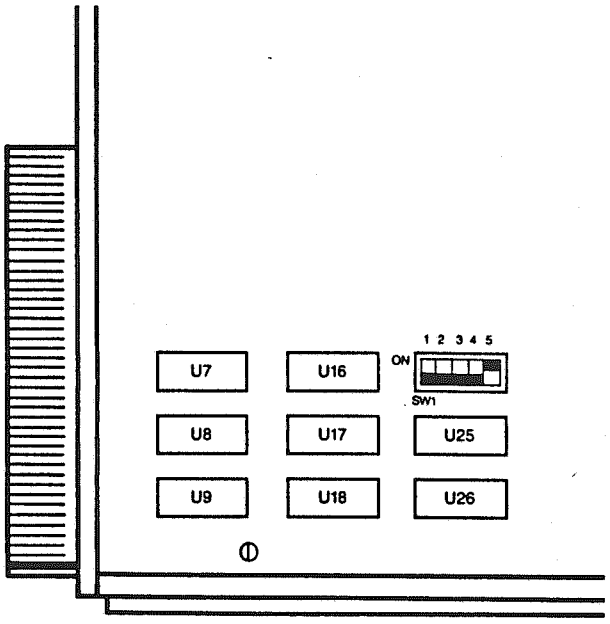
Contact	Signal Line
1	DIO 1
2	DIO 2
3	DIO 3
4	DIO 4
5	EOI
6	DAV
7	NRFD
8	NDAC
9	IFC
10	SRQ
11	ATN
12	SHIELD
13	DIO 5
14	DIO 6
15	DIO 7
16	DIO 8
17	REN
18	GND – DAV
19	GND – NFRD
20	GND – NDAC
21	GND – IFC
22	GND – SRQ
23	GND – ATN
24	LOGICAL GND

IEC Connector
Cannon DB25S

Contact	Signal Line
1	DIO 1
2	DIO 2
3	DIO 3
4	DIO 4
5	REN
6	EOI
7	DAV
8	NRFD
9	NDAC
10	IFC
11	SRQ
12	ATN
13	SHIELD
14	DIO 5
15	DIO 6
16	DIO 7
17	DIO 8
18	GND – DAV
19	GND – NFRD
20	GND – NDAC
21	GND – IFC
22	GND – SRQ
23	GND – ATN
24	SINGLE GND

Model 3988-G3A/D3A

GPIB ADDRESSING



The GPIB address switch (SW1) is located on the component side of the module near the lower left corner

SW1 controls the lower 5 bits of the modules address's for both talk and listen.

	Fixed Value	SW1 5 4 3 2 1	SW1
TALK	0 1 0	X X X X X	ON = 1
LISTEN	0 0 1	X X X X X	OFF = 0

The listen address can start at 32 with SW1 all off and go up/to 63 with SW1 all ON. The talk address can start at 64 with SW1 all OFF and go up to 127 with SW1 all ON. The SW1 setting determines both the talk and listen address.

EXAMPLE:

	8 7 6	SW1 5 4 3 2 1
	FIXED VALUE	0 0 0 0 0
		N F F F F
		F F F F

TALK	0	1	0	1	0	0	0	0	=	64 + 16 = 80
LISTEN	0	0	1	1	0	0	0	0	=	32 + 16 = 48

With SW1-5 on and SW1 (1-4) off the listen address is 48 and the talk address is 80.

Remote Message Coding

Mnemonic		Message Name	Type	Class	Bus Signal Line(s) and Coding That Asserts the True Value of the Message													
					DIO8	7	6	5	4	3	2	DIO1	NDAC	NRFD	ATN	EOI	SRQ	IFC	REN
ATN	attention		U	UC	X	X	X	X	X	X	X	X	XXX	1	X	X	X	X
DAB	data byte	(Notes 1, 5)	M	DD	D8	D7	D6	D5	D4	D3	D2	D1	XXX	Ø	X	X	X	X
DAC	data accepted		U	HS	X	X	X	X	X	X	X	X	XXØ	X	X	X	X	X
DAV	data valid		U	HS	X	X	X	X	X	X	X	X	1XX	X	X	X	X	X
END	end		U	ST	X	X	X	X	X	X	X	X	XXX	Ø	1	X	X	X
IDY	identify		U	UC	X	X	X	X	X	X	X	X	XXX	X	1	X	X	X
IFC	interface clear		U	UC	X	X	X	X	X	X	X	X	XXX	X	X	X	1	X
MLA	my listen address	(Note 2)	M	AD	Y	Ø	1	L5	L4	L3	L2	L1	XXX	1	X	X	X	X
MTA	my talk address	(Note 3)	M	AD	Y	1	Ø	T5	T4	T3	T2	T1	XXX	1	X	X	X	X
RFD	ready for data		U	HS	X	X	X	X	X	X	X	X	XØX	X	X	X	X	X
RQS	request service	(Note 5)	U	ST	X	1	X	X	X	X	X	X	XXX	Ø	X	X	X	X
SPD	serial poll disable		M	UC	Y	Ø	Ø	1	1	Ø	Ø	1	XXX	1	X	X	X	X
SPE	serial poll enable		M	UC	Y	Ø	Ø	1	1	Ø	Ø	Ø	XXX	1	X	X	X	X
SRQ	service request		U	ST	X	X	X	X	X	X	X	X	XXX	X	X	1	X	X
STB	status byte	(Notes 4, 5)	M	ST	S8	X	S6	S5	S4	S3	S2	S1	XXX	Ø	X	X	X	X
UNL	unlisten		M	AD	Y	Ø	1	1	1	1	1	1	XXX	1	X	X	X	X
UNT	untalk	(Note 6)	M	AD	Y	1	Ø	1	1	1	1	1	XXX	1	X	X	X	X

The 1/Ø coding on ATN when sent concurrent with multiline messages has been added to this revision for interpretive convenience.

NOTES:

(1) D1-D8 specify the device dependent data bits.
(2) L1-L5 specify the device dependent bits of the device's listen address.
(3) T1-T5 specify the device dependent bits of the device's talk address.
(4) S1-S6, S8 specify the device dependent status. (DIO7 is used for the RQS message.)
(5) The source of the message on the ATN line is always the C function, whereas the messages on the DIO and EOI lines are enabled by the T function. function.
(6) This code is provided for system use.

AUXILIARY CONTROLLER MODE

An Auxiliary Controller (AC), which occupies one or more normal stations, provides an additional source of control to a CAMAC crate. The main Crate Controller occupies the control station along with one normal station (with slight modifications, the 3988 can be used as either a main or Auxiliary Crate Controller). Since the N and L lines are provided at the control station only, an Auxiliary Controller gains access to these lines from the Auxiliary Controller Bus (ACB). Controllers that can be used as or with an Auxiliary Controller require a rear panel connector for connection to the ACB.

The priority arbitration for control of the Dataway involves three signals. The REQUEST signal is common to all Controllers via the ACB and is extended to the front panel of each controller. The ACB GRANT signal is daisy-chained between front panel connectors of all Controllers. The REQUEST INHIBIT is one of the signals of the ACB.

> The front panel REQUEST output of the highest priority Controller must be connected to ACB GRANT-IN of the same Controller. The front panel signal, ACB GRANT-OUT, of this Controller is connected to the front panel signal, ACB GRANT-IN, of the next highest priority Controller. Similar connections are made until the lowest priority Controller is reached.

To gain control of the Dataway, a Controller asserts the REQUEST line and waits for the GRANT signal. When a Controller receives an ACB GRANT-IN, it generates an ACB GRANT-OUT if it did not make the rquest. The GRANT signal propagates to the other Controllers in the order of priority. When the Controller that initiated the request receives an ACB GRANT-IN, it does not generate an ACB GRANT-OUT and the propagation is halted. Instead, the requesting Controller asserts the REQUEST INHIBIT line, and performs a Dataway operation.

When more than one Controller initiates a request, the Controller with the highest priority receives the ACB GRANT-IN Signal. Once that Controller asserts the RQUEST INHIBIT line, all other Controllers remove their REQUEST signals. On completion of the Dataway operation, the Contreoller removes the REQUEST INHIBIT signal, and control of the Dataway is given to the next Controller requesting it.

Another signal line, Auxiliary Controller Lockout (ACL), of the ACB can be used by one Controller in the crate to gain control of the Dataway. An asserted ACL will abort a Dataway operation in progress if S1 has not occurred. If S1 occurred, the present Dataway operation is completed before control is given to the Controller that asserted ACL. The 3988 is designed to respond to the ACL signal, but it cannot assert it.

The remaining signal lines of the ACB are the N and L lines. There are 5 encoded N signals. An Auxiliary Controller uses these lines (when it is performing an addressed Dataway operation) to transmit a binary-coded N to the main Crate Controller. There are 24 L lines (one for each normal station). When an Auxiliary Controller has control of the Dataway, it must respond to the L signals gernerated by the modules. The main Crate Controller transmits the 24 L lines via the ACB.

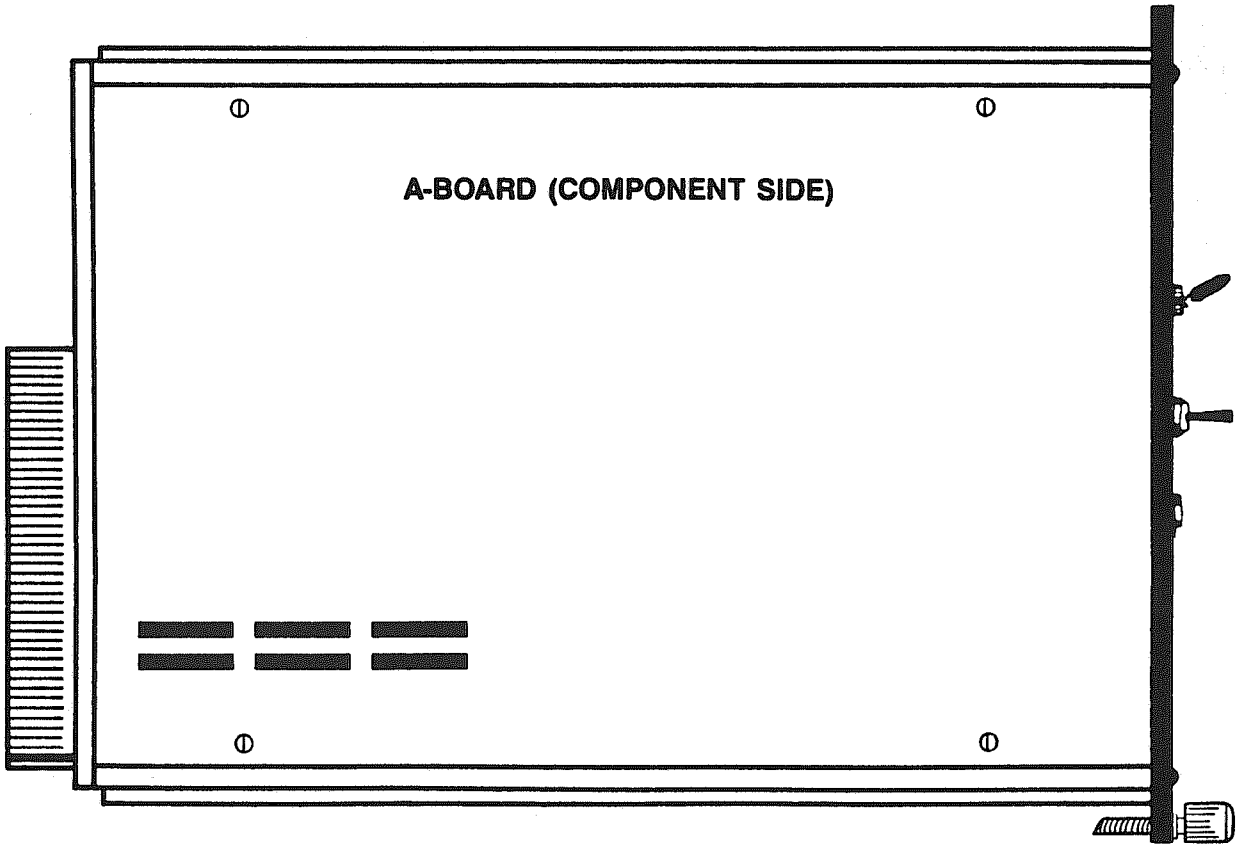
Modifications required to convert the 3988 to an Auxiliary Controller are:

- The front panel connectors labelled ACB Request, GI and GO must be connected as described in the indented text on the previous page.

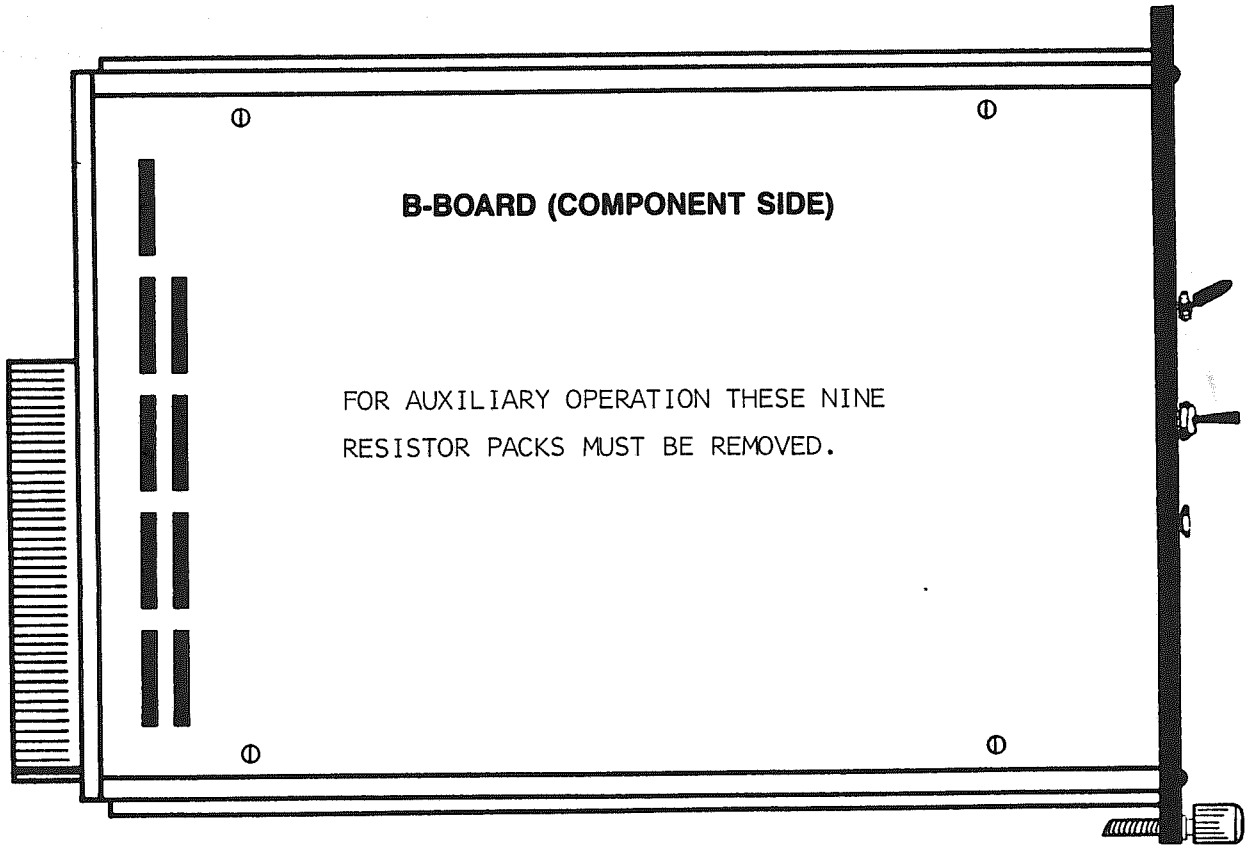
- The ACB cable must be connected to the rear panel connector of all Controllers within the crate

- Fifteen resistor packs must be removed from sockets on the 3988 as described below:

The 3988 consists of two circuit boards. Looking at the front panel, the circuit board on the left is called the A-board. The B-board is the circuit board on the right. With the 3988 out of the crate, the component side of the A-board is accessible. the six (6) resistor packs shown in the diagram below must be removed (desoldering is not necessary since the resistor packs are in sockets.



To gain access to the component side of the B-board, the two screws on the front panel (connected to the A-board) must be removed. Also the four screws on the rear panel and the rear panel itself must be removed. Loosen the two screws that hold the B-board to the front panel. Once the screws and rear panel are removed, the unit should be gently separated . If necessary, disconnect the front panel connector from the A-board. The nine resistor packs must be removed from the B-board. The nine resistor packs must be removed from the B-board. The placement of these resistor packs is shown below:



The A-board, rear panel, and all six screws must be reinserted before the unit is mounted in the CAMAC crate.

NOTE: CARE SHOULD BE TAKEN TO PREVENT ANY OF THE INTERBOARD PINS FROM BEING BENT.

Now the 3988 can be used as an Auxiliary Controller. All resistor packs must be reinserted when the 3988 is used as a Main Crate Controller.

Program example using KSC 3982 List Sequencing Crate Controller and KSC 3988 GPIB Crate Controller is as follows:

```
/* WORD - program to read 3982 and store data 121 cycles at a time. */

/* Link this program with appropriate *cib*.obj.    */


#include "stdio.h"
#include "decl.h"
#include <stdlib.h>

#define ERR      (1<<15)      /* Error detected          */
#define TIMO     (1<<14)      /* Timeout                 */
#define SRQI     (1<<12)      /* SRQ detected by CIC     */

/* Application program variables passed to GPIB functions */
int  dat[6768];         /* read data buffer */
int  stat;
int  bd;                     /* board or device number   */
int  dd;
int  i;
main() {

 /* Assign unique identifier to board 0 and store
    in variable bd.  Check for error.
    (ibfind error = negative value returned.)      */

    if ((bd = ibfind ("GPIB0")) < 0) finderr();
    if ((dd = ibfind ("camac")) < 0) finderr();

 /* Send the Interface Clear (IFC) message to all
    devices.  Check for error on each GPIB call to
    be safe.  Note that GPIB status is available
    both through global variable ibsta and through
    the return values of all GPIB functions except
    IBFIND.                                        */

    if (ibsic (bd) & ERR) error();

 /* reset the data pointers */

    ibwrt (dd,"\x16\x0\x9",3);
    if (ibsta & ERR) error();
```

```
/* set mem pointer for naf to zero */

   ibwrt (dd,"\x16\x2\x10\x0\x0\x0",6);
   if (ibsta & ERR) error();
/* load 16 f2 a0 n 14 commands in list */
   for (i = 0 ; i < 16 ; i++) {
   ibwrt (dd,"\x16\x1\x10\x0\x5C\x02",6);
   if (ibsta & ERR) error();
   }

/* load last naf with eol marker */

   ibwrt (dd,"\x16\x1\x10\x0\xDC\x02",6);
   if (ibsta & ERR) error();
/* load cycle control with 200 hz cycle and 50 khz dw time */

   ibwrt (dd,"\x16\x0\x11\x0\x0\x73",6);
   if (ibsta & ERR) error();

/* enable half full lam in lam mask */

   ibwrt (dd,"\x16\xD\x11\x0\x0\x10",6);
   if (ibsta & ERR) error();

/* enable module for list execution */

   ibwrt (dd,"\x16\x0\x1A",3);
   if (ibsta & ERR) error();

/* start executing list */

   ibwrt (dd,"\x16\x0\x19",3);
   if (ibsta & ERR) error();

/* clear inhibit line for counter to start counting */

   ibwrt (dd,"\xle\x0\x11\x0\x0\x0",6);
   if (ibsta & ERR) error();

restart:
    do {
    ibwrt (dd,"\xle\xc\x1",3);  /* check for lam in 3988 */
    if (ibsta & ERR) error();
    ibrd (dd, rd ,3);
    if (ibsta & ERR) error();
```

```
    ) while ( rd[0] == 0 );
/*      setup for dma operation */
    ibwrt (dd,"\x1e\x0\x10\x0\x08\x09",6);  /* load word count */
    if (ibsta & ERR) error();

    ibwrt (dd,"\x1e\x0\x11\x0\x18\x0",6);   /* wrt csr (qrpt,24b) */
    if (ibsta & ERR) error();
/*  send starting NAF */
    ibwrt (dd,"\x16\x0\x0",3);
    if (ibsta & ERR) error();

 /* get data from gpib */
    ibrd (dd, dat ,6171);
   printf(" ibcnt=0x%x\n",ibcnt);
    if (ibsta & ERR) error();
/*    printf( "%d%d%d\n",dat[0],dat[1],dat[2]); */

 /* clear half full lam in 3982 */

    ibwrt (dd,"\x16\xc\x17\x0\x0\x10",6);
    if (ibsta & ERR) error();

 /* go do it again */

    goto restart;
 /*   ibwrt (dd,"\x16\x0\x18",3);
    if (ibsta & ERR) error();   */

   }

finderr() {
/* This routine would notify you that the ibfind
   call failed, and refer you to the handler
   software configuration procedures.              */

   printf("Ibfind error; does device or board\n");
   printf("name given match configuration name?\n");
   }
```

```
error() {
/* An error checking routine at this location would,
   among other things, check iberr to determine the
   exact cause of the error condition and then take
   action appropriate to the application.  For
   errors during data transfers, ibcnt may be
   examined to determine the actual number of bytes
   transferred.                                    */

   printf("GPIB function call error:\n");
   printf("ibsta=0x%x, iberr=0x%x,",ibsta,iberr);
   printf(" ibcnt=0x%x\n",ibcnt);
   }
```