EG&G

EGG-1183-1733
1 November 1978

LASL/EG&G
GENERAL PURPOSE IEEE-488
BUS DEVICE DRIVER
—GPDRV—

By

Ivan J. Brower

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

Approved for Publication

F.R. Bentley
Engineering Department Manager

This Document is
UNCLASSIFIED

A.J. Tarr
Assistant Operations Manager

Prepared for the U.S. Department of Energy Under

Contract No. EY-76-C-08-1183.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

CONTENTS

CONTENTS (Cont)

ILLUSTRATIONS

1. INTRODUCTION

GPDRV is a Device Driver for the RSX-11M Operating System designed as a software interface between RSX-11M and the IEEE-488 Bus via a Tektronix CP1100 Interface. GPDRV allows communication with the IEEE-488 Bus by use of the RSX-11M standard FORTRAN READ, WRITE, and QIO Statements.

The Tektronix CP1100 Interface must be modified so all 8 interrupts will interrupt through the same vector and bit 14 of the Bus Address Register will indicate if the CP1100 Interface is System Controller. The CP1100 Interface may be further modified if it is System Controller (bit 14 of the Bus Address Register equals a zero) so that Interface Clear and Remote Enable cannot be sent by the Interface. A more detailed explanation of the CP1100 may be found in the "TEKTRONIX CP1100/IEEE 488 INTERFACE" Instruction Manual.

2. GPDRV OPERATING INSTRUCTIONS

GPDRV is a Device Driver for the RSX-11M Operating System designed to allow communications with the IEEE-488 Bus by use of the RSX-11M standard FORTRAN READs, WRITEs, and QIOs.

In order to use GPDRV, a program must first assign a logical unit number to the IEEE-488 Bus unit to be used. This is done using the subroutine ASNLUN. (See "RSX-11M EXECUTIVE REFERENCE MANUAL", p. 4-9.)

```
IUNT=1
CALL ASNLUN (1,'GP',IUNT,IOSB)
```

1	=	Logical Unit Number
'GP'	=	2-Character Device Mnemonic
IUNT	=	IEEE-488 Bus Unit
IOSB	=	Directive Status

After a logical unit number is assigned and before I/O can be processed, the program must be attached to the logical unit number. The Attach sets the proper conditions in GPDRV to allow use of the IEEE-488 Bus. A program can then talk and listen to the IEEE-488 Bus by using WRITEs, READs, and QIOs. The IEEE-488 Bus status can be changed and read using the Alter Device Status function. When GPDRV is not Controller-in-Charge, timeouts are disabled. If GPDRV is Controller-in-Charge, the default System Timeout for the I/O is 2 seconds. This can be changed using the Change Timeout function. Errors are handled in the standard RSX-11M manner.

2.1 ATTACH TO GPDRV

In order to function properly, a program must attach to GPDRV before I/O to the IEEE-488 Bus is attempted. This is done by using a QIO to send the function IO.ATT. (See "RSX-11M EXECUTIVE REFERENCE MANUAL", p. 4-68.)

FORTRAN CALL:

```
CALL WTQIO (IOATT,1,2,,IOSB)
```

MACRO CALL:

```
QIOW$ IO.ATT,1,2,,IOSB
```

IOATT = IO. ATT Function Code (Octal 1400 for RSX-11M Version 3.1)

1 = Logical Unit Number

2 = Event Flag Number

IOSB = 2-Word Integer Array to Receive Final I/O Status

An Attach function clears the CP1100 Interface status and sets GPDRV as Controller-in-Charge if the CP1100 Interface is set as System Controller.

2.2 DETACH FROM GPDRV

When a program completes I/O to the IEEE-488 Bus, it should detach from GPDRV. This is done by using a QIO to send the function IO. DET. (See "RSX-11M EXECUTIVE REFERENCE MANUAL", p 4-68.) A Detach is automatically performed when a program is terminated.

FORTRAN CALL:

```
CALL WTQIO (IODET,1,2,,IOSB)
```

MACRO CALL:

```
QIOW$ IO.DET,1,2,,IOSB
```

IODET = IO. DET Function Code (Octal 2000 for RSX-11M Version 3.1)

The other variables used in the FORTRAN and MACRO calls are the same as those used for the Attach function. A Detach function clears the CP1100 Interface status.

2.3 TALK

Talking to the IEEE-488 Bus can be done by using FORTRAN WRITE or QIO statements. If the first character sent by a WRITE or QIO statement is a colon, all data following the colon will be transferred without further checks. If the first character is not a colon, the data is interrupted as Bus and Device Commands.

Commands

To send Bus and Device Commands, the first character sent by the WRITE or QIO statement should not be a colon; however, Bus and Device Commands can be sent in the same WRITE or QIO statement if separated by a colon. Bus Commands are separated by a comma. BUS COMMANDS ARE IN OCTAL. If a program is to listen to a talker, it must address itself as a listener at the same time it addresses the talker. This is done to prevent some or all of the data from being lost.

```
        WRITE (1,100)
100     FORMAT ('137,77')                       Bus Command
```

In this case, Logical Unit 1 is assigned to a GPDRV Unit and the untalk (137) and unlisten (77) are sent. Other possible FORMAT statements might be:

```
100     FORMAT ('MODE DIG')                     Device Command
100     FORMAT ('40/140')                       Bus Address
100     FORMAT ('77,40/140:MODE DIG')           Bus Command, Bus
                                                Address, and Device
                                                Command
```

For convenience, the primary and secondary addresses are separated by a slash or a comma.

Data

To send data, the first character sent by the WRITE or QIO statement must be a colon. Everything following the colon is then sent to the IEEE-488 Bus just as it is received by GPDRV.

```
        WRITE (1,100,ERR=50) (IDATA(I),I=1,40)
100     FORMAT (':',/40A2)
```

If more than 80 bytes of data are transferred, a QIO should be used.

```
        DIMENSION IOSB (2),IDATA(500),IPARAM(6)
        DATA IWRITE/"400/

        IPARAM (2)=500
        CALL GETADR (IPARAM(1),IDATA(1))
        CALL WTQIO (IWRITE,1,2,,IOSB,IPARAM)
```

IWRITE = IO.WLB Function Code (Octal 400 for RSX-11M Version 3.1)

IPARAM = 6-Word Integer Array Containing Device Dependent Parameters

1st Word Is Buffer Address
2nd Word Is Byte Count

The other parameters in the QIO are the same as those used with the Attach QIO.

2.4 LISTEN

Listening to the IEEE-488 Bus can be done by using FORTRAN READ or QIO statements. If a program is to listen to a talker, it must address itself as a listener at the same time it addresses the talker. If this is not done, some or all of the data sent by the talker will be lost. If more than 80 bytes will be read or no EOI will be sent with the data, a QIO should be used. When no EOI is sent, the QIO byte should be the exact number of bytes to be transferred. If the byte count is greater than the number of bytes to be transferred and no EOI is sent, a Timeout Error will occur while GPDRV is waiting for the transfer to complete.

```
          READ (1,200,END=20,ERR=50) (IDATA(I),I=1,40)
200       FORMAT (40A1)
```

In this case, if an EOI is received, and End-of-File is returned to the READ statement.

```
          DATA IREAD/"1000/

          IPARAM (2)=1
          CALL GETADR (IPARAM(1),IDATA)

          CALL WTQIO (IREAD,1,2,,IOSB,IPARAM)
```

IREAD = IO.RLB Function Code (Octal 1000 for RSX-11M Version 3.1)

IPARAM = 6-Word Integer Array Containing Device Dependent Parameters

1st Word is Address of Buffer
2nd Word is Byte Count

The other parameters in the QIO are the same as those used with the Attach QIO. This is an example of a QIO that would be used to read a response from a serial poll.

2.5 INTERCHANGE DATA

A program can cause 2 devices on the IEEE-488 Bus to Interchange Data by addressing one as a listener and the other as a talker. GPDRV, if not addressed to listen, does not listen to the interchange but waits for EOI to be set by the talker or a timeout to occur.

```
        WRITE (1,100,ERR=50)
100     FORMAT ('41/140:READ PTR;')
        WRITE (1,101,ERR=50)
101     FORMAT ('60,101/140')
```

This tells the 7912AD (Unit 1) to send its Pointers Array, addresses device 60 to listen, and the 7912AD to talk. The Pointers Array will be sent to device 60 without intervention from GPDRV if it completes before a timeout occurs. When EOI is sent by the 7912AD, GPDRV will return a successful I/O completion to the WRITE statement.

2.6 ALTER DEVICE STATUS

GPDRV status for the IEEE-488 Bus can be altered and read using the function IO.ADS with the subfunction (the lower byte) used to indicate the desired action.

```
DIMENSION IOSB (2)
DATA IOADS/"14nnn/
CALL WTQIO (IOADS,1,2,,IOSB)
```

IOADS = IO.ADS Function Code (Octal 14nnn)
nnn = Subfunction Code

When subfunction codes are sent:

000 = Clear Status
001 = Want Control
002 = Waiting to Receive Data
004 = Waiting to Send Data
010 = Busy
020 = Set SRQ
040 = Wait for SRQ
100 = Send IFC
200 = Read Status

When the Status is read, all bits (except for 200) have the same meaning as when sent. The 200 means GPDRV is Controller-in-Charge.

2.7 CHANGE TIMEOUT

To change the system timeout for GPDRV, the IO.CTI function can be used with the subfunction (the lower byte) equal to the timeout desired in seconds (expressed in OCTAL).

```
DATA IOCTI/"154nn/
      IOCTI  =  IO.CTI Function Code (Octal 154nn)
                nn = Timeout in Seconds (0-377 Octal)
```

If the timeout is set to 0, it disables timeouts.

2.8 ERRORS

To handle errors from READs or WRITEs, the subroutines ERRSET and ERRSNS are used. (See "IAS/RSX-11 FORTRAN IV USER'S GUIDE", pp. B-4 and B-5, for explanations of these subroutines.)

```
      CALL ERRSET (39,,.FALSE.,,.FALSE.,)
            39  =  Read Error, 38 =  Write Error
                   The Two .FALSE. 's tell the system not to count
                   this error against the Task's maximum error count
                   and not to produce an error message for this error.
      READ (1,200,END=20,ERR=50) (IDATA(I),I=1,40)
50    CALL ERRSNS (NUM,IFCER,IFCER1,IUNIT)
                NUM  =  Error Number (In this case, 39.)
              IFCER  =  Error Code From 1st Status Word
             IFCER1  =  2nd Status Word
              IUNIT  =  Logical Unit Number
```

To handle errors from QIOs, the I/O Status Block is checked to see if an error occurred.

```
      CALL WTQIO (IREAD,1,2,,IOSB,IPARAM)
               IOSB  =  2-Word Array
                        1st Word is Status of I/O (1=successful completion)
                        2nd Word is Byte Count When No Error Occurred
      IF (IOSB(1).NE.1) GO TO 50
50    IF (IOSB(1).GE.128)IOSB(1)=IOSB(1)-256
            This converts a negative byte into a negative word.
```

The error codes returned by the driver are listed below. In addition to these errors, the system returns the standard errors described in Appendix 1 of "IAS/RSX-11 I/O OPERATIONS REFERENCE MANUAL".

	0	=	Device Not Loaded (Returned by Attach)
IE.IFC	-2	=	Illegal Function
IE.DNR	-3	=	Device Not Ready
IE.EOF	-10	=	End-of-File
IE.ABO	-15	=	I/O Aborted (Handled by the system.)
	-68	=	IFC Received Or Sending IFC When Not System Controller
	-69	=	DCL Or SDC Received
	-70	=	REN False Occurred
	-71	=	ATN Occurred While Controller-in-Charge or Setting ATN When Not Controller-in-Charge (IOSB (2) = IEEE-488 Status to check for Controller-in-Charge)
	-72	=	Waiting for SRQ When Not Controller-in-Charge Or Setting SRQ When Controller-in-Charge (IOSB (2) = IEEE-488 Status to check for Controller-in-Charge)
	-73	=	Changing Timeout When Not Controller-in-Charge
IE.TMO	-74	=	Timeout

3. GPDRV INTERNALS

GPDRV Internals is a detailed description of the Driver GPDRV. GPDRV, which is similar to all RSX-11M Drivers, is comprised of four main parts.

1. Device Initiator initializes the I/O operation to GPDRV.
2. Device Timeout cancels I/O when a timeout occurs.
3. Interrupt routine does a poll to determine which interrupt occurred. Once the interrupt is determined, the appropriate action is taken.
4. Cancel I/O Operation cancels I/O being done by GPDRV when a program is aborted.

GPDRV will respond to six I/O Function Codes: Attach, Detach, Read Logical Block, Write Logical Block, Change Timeout, and Alter Device Status. These functions are used to control GPDRV I/O.

GPDRV responds to I/O with the following Return Codes: Successful I/O Completion, Illegal Function, Device Not Ready, End-of-File, Abort, Interface Clear Error, Device Clear Error, Remote Enable False Occurred, Attention Error, Service Request Error, Change Timeout Error, and Timeout.

The CP1100 Interface has eight control registers. The symbols equated to each of these registers are:

Talker Data Buffer	=	TDB
Listener Data Buffer	=	LDB
Interrupt Control Register	=	ICR
Bus Status Register	=	BSR
Bus Control Register	=	BCR
Interface Status Register	=	ISR
Byte Counter Register	=	CTR
Bus Address Register	=	BAR

The Bus Address set for GPDRV is 8. This means the Listen Address is 50 Octal and the Talker Address is 110 Octal.

Internal flags, status, and counters are kept in the UCB. These indicate the conditions of GPDRV and the CP1100 Interface.

Many of the Bus Commands are referred to by their symbolic name. The valid ones for GPDRV are listed below with their Octal value.

Device Clear	=	DCL	=	024
Selected Device Clear	=	SDC	=	004
Serial Poll Disable	=	SPD	=	031
Serial Poll Enable	=	SPE	=	030
Take Control	=	TCT	=	011
Unlisten	=	UNL	=	077
Untalk	=	UNT	=	137
GPDRV Listen Address	=	LADD	=	050
GPDRV Talk Address	=	TADD	=	110

The Bus Control Lines have symbolic names and are listed below.

Interface Clear	=	IFC
Attention	=	ATN
Service Request	=	SRQ
Remote Enable	=	REN
End or Identify	=	EOI
Data Valid	=	DAV
Not Ready for Data	=	NRFD
No Data Accepted	=	NDAC

3.1 DEVICE INITIATOR

This section of the Driver is entered when an I/O request is queued and again at the end of the I/O operation to dequeue the next I/O request. (See Figure 1.) The type of I/O operation is determined, the appropriate interrupts are enabled, and conditions are set on the CP1100 Interface.

3.1.1 Driver Initialization

The I/O request is dequeued by calling $GTPKT to get an I/O Packet to process. If the Carry Bit in the Processor Status Word is cleared after $GTPKT is called, an I/O request was successfully dequeued. The addresses of the I/O Packet, Status Control Block (SCB), and Unit Control Block (UCB), along with the IEEE-488 Bus unit number and the controller index, are returned in registers.

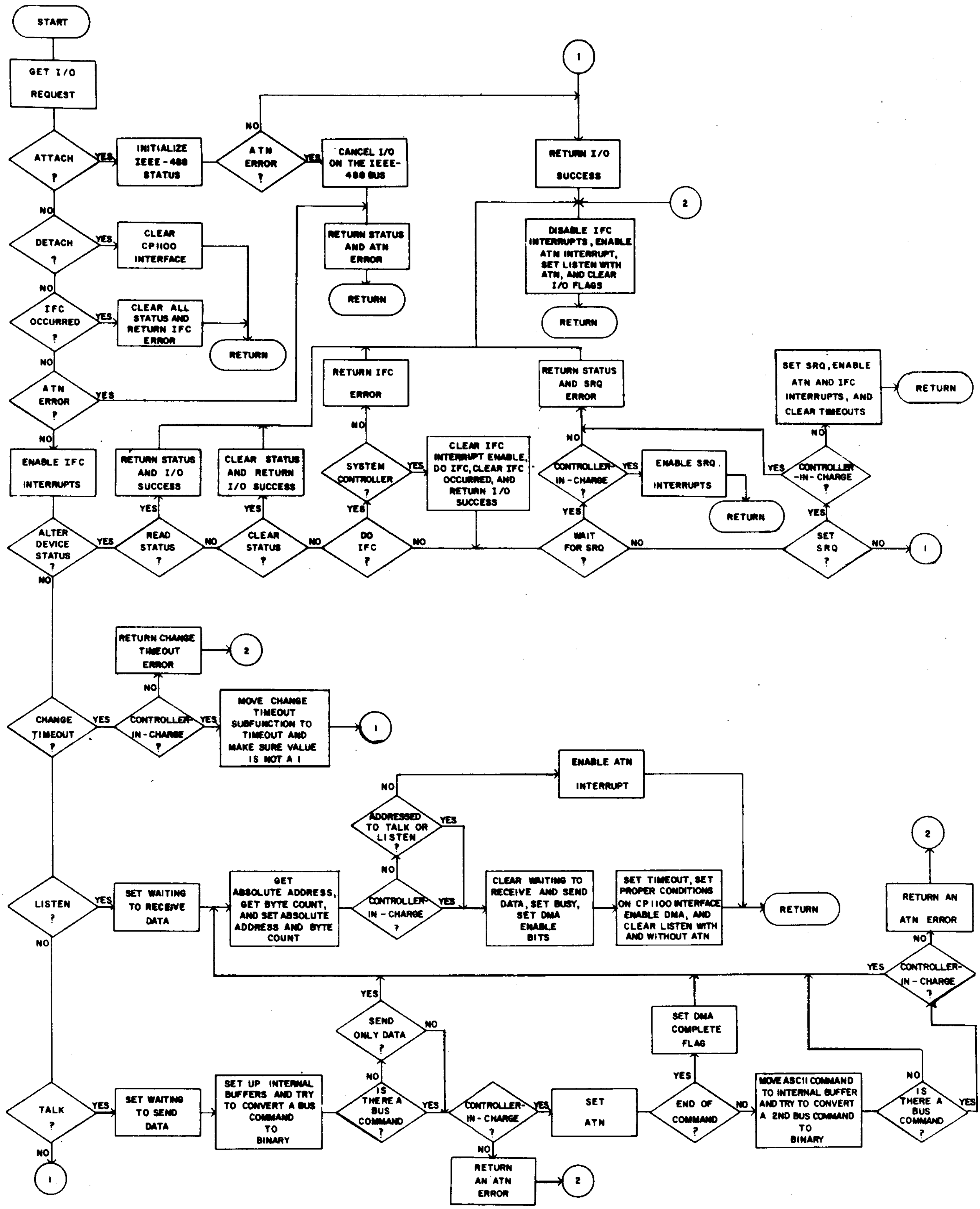

Figure 1. Device initiator (GPINI).

The UCB address is saved in the Impure Data Table CNTBL:, the Abort Flag is cleared, the Function Code is saved in the UCB, the Timeout is initialized, the CSR address is placed in R4, the (IFC) Occurred is checked, and the IFC interrupt is enabled. If an IFC did occur, I/O is terminated and a IFC error is returned to the program. If an IFC did not occur, the driver is now ready to check for the I/O operation that has been requested.

3.1.2 I/O Initialization

Comparisons of the I/O Function Code sent are made with the valid I/O Function Codes. When the function is identified, a jump is made to the appropriate routine to initialize the I/O operation.

Attach

The Attach function initializes the internal flags for the IEEE-488 Bus. If the CP1100 Interface is set as System Controller, GPDRV is set to be Controller-in-Charge and REN is set true. All other internal flags are cleared. The ATN interrupt is enabled and Listen with ATN is set. This allows ATN to be handled according to the IEEE-488 Spec. The Timeout is set at 2 seconds if GPDRV is Controller-in-Charge or disabled if it is not Controller-in-Charge.

Detach

The Detach function clears conditions on the CP1100 Interface by doing a Local Reset and returning successful I/O completion to the program.

Alter Device Status

The Alter Device Status function affects GPDRV status for the CP1100 Interface. This status is a byte in the UCB.

Read Status

If bit 7 of the Subfunction Code is set, the Status is returned in the Low-order byte of the 2nd status word.

Clear Status

If all bits are cleared in the Subfunction Code, the status is cleared.

Do IFC

If bit 6 of the Subfunction Code is set, an IFC is performed when the CP1100 Interface is set as System Controller. If the CP1100 Interface is not set as System Controller, an IFC error is returned.

Wait for SRQ

If bit 5 of the Subfunction Code is set and GPDRV is Controller-in-Charge, (bit 7 of the Status is set) GPDRV will enable the SRQ interrupt and wait for SRQ to be set or a timeout to occur. If GPDRV is not Controller-in-Charge, an SRQ error is returned with GPDRV Status in the 2nd I/O Status Word.

Set SRQ

If bit 4 of the Subfunction Code is set and GPDRV is not Controller-in-Charge, the SRQ is set and GPDRV will wait for a serial pole to be performed. If GPDRV is Controller-in-Charge, an SRQ error is returned with GPDRV Status in the 2nd I/O Status Word.

Change Timeout

The Change Timeout function will change the device timeout used by RSX-11M for GPDRV. The default is 2 seconds and is stored in the SCB. The GPDRV timeout is stored in a byte in the UCB. The Subfunction Code is moved into the UCB. If GPDRV is not Controller-in-Charge, a Change Timeout error is returned.

Listen

The Read Logical Block function sets GPDRV Status to waiting to receive data, gets the Absolute Address in memory where data will be placed, gets the byte count, and enables DMA.

To set up DMA, the extended address bits in R3 are set in the BCR, the address in R2 is set in the BAR, the internal flag (indicating that DMA is in progress) is set, and the 2's complement of R0 (the byte count) is set in the CTR, the other CP1100 Interface registers are then initialized for DMA. If GPDRV is Controller-in-Charge, or GPDRV is not Controller-in-Charge but is addressed to listen; the DMA is enabled by setting the timeout count, clearing listen with and without ATN, clearing the ATN interrupt enable, setting the proper bits in the ISR, and enabling the DMA interrupts. If GPDRV is not Controller-in-Charge and addressed to listen, it waits until it is addressed to listen before enabling DMA.

Talk

The Write Logical Block functions sets GPDRV Status to waiting to send data. It then checks to see if data or commands are to be sent and finally initializes DMA.

Data

If the first byte is a colon, the rest of the data is transferred just as it is received by GPDRV. A flag is set to indicate DMA will be complete at the end of this transfer, the Absolute Address of the data is placed in registers R2 and R3, and the byte count is placed in R0. The DMA is now ready to be initialized.

Commands

If the first byte is not a colon, the data is translated as commands and is placed in an internal buffer in the UCB. There are 2 types of commands: Bus Commands and Device Commands, which can be sent across the IEEE-488 Bus. Bus Commands can be sent by themselves or they can be sent before and/or after a Device Command. A Device Command can also be sent by itself.

Bus Commands are received by the Driver as ASCII numbers and are translated into binary to be sent to the IEEE-488 Bus. Device Commands are sent just as they are received. A Bus Command is separated from a Device Command by a colon.

The Absolute Address of the UCB buffer is placed in R2 and R3 and the Byte Count is placed in R0. The DMA is then ready to be Initialized.

Commands sent to the IEEE-488 Bus can be sent in 1, 2, or 3 segments according to the arrangement of the Bus and Device Commands.

Initialize DMA

To set up DMA, the extended address bits in R3 are set in the BCR, the address in R2 is set in the BAR, the internal flag indicating DMA is in progress is set, and the 2's complement of R0 is set in the CTR. The other CP1100 Interface registers are then initialized for DMA. If GPDRV is Controller-in-Charge, or GPDRV is not

Controller-in-Charge but is addressed to listen; the DMA is enabled by setting the timeout count, clearing listen with and without ATN, clearing the ATN interrupt enable, setting the proper bits in the ISR, and enabling the DMA interrupts. If GPDRV is not Controller-in-Charge and addressed to listen, it waits until it is addressed to listen before enabling DMA.

3.2 DEVICE TIMEOUT

The RSX-11M System maintains a timer. If the timer decrements to 0 before the I/O completes, RSX-11M calls the Driver timeout routine. (See Figure 2.) The timeout routine clears all interrupts and DMA is halted. If an abort was requested, an abort error is returned. If an abort was requested, a device not ready message is printed on the console terminal. A check is made to see if DMA is in progress, if not, a Timeout error is returned. If DMA is in progress and the byte count is 0, it has completed and I/O success is returned. If DMA was still in progress and DAV is false, a timeout error is returned. If DAV is true and goes false in less than 1/2 second, the timeout is reset and DMA is resumed; otherwise, a timeout error is returned.

When a timeout error is returned, the number of bytes transferred is returned in the 2nd I/O Status Word and an attempt to clear the IEEE-488 Bus is made by sending a Device Clear, doing a Local Reset on the CP1100 Interface, and doing an IFC if the CP1100 Interface is a System Controller.

3.3 DEVICE INTERRUPTS

When an interrupt occurs, all interrupts are disabled on the CP1100 Interface and the Processor Status is dropped to 0. (See Figure 3.) A poll is then done to find which interrupt occurred.

3.3.1 ATN Interrupt

When the ATN Interrupt occurs, Listen with ATN is set and a check is made to see if previous ATN Interrupt handling is complete. This prevents an attempt to handle 2 ATN Interrupts simultaneously causing erroneous results. A check is then made to see if GPDRV is Controller-in-Charge.

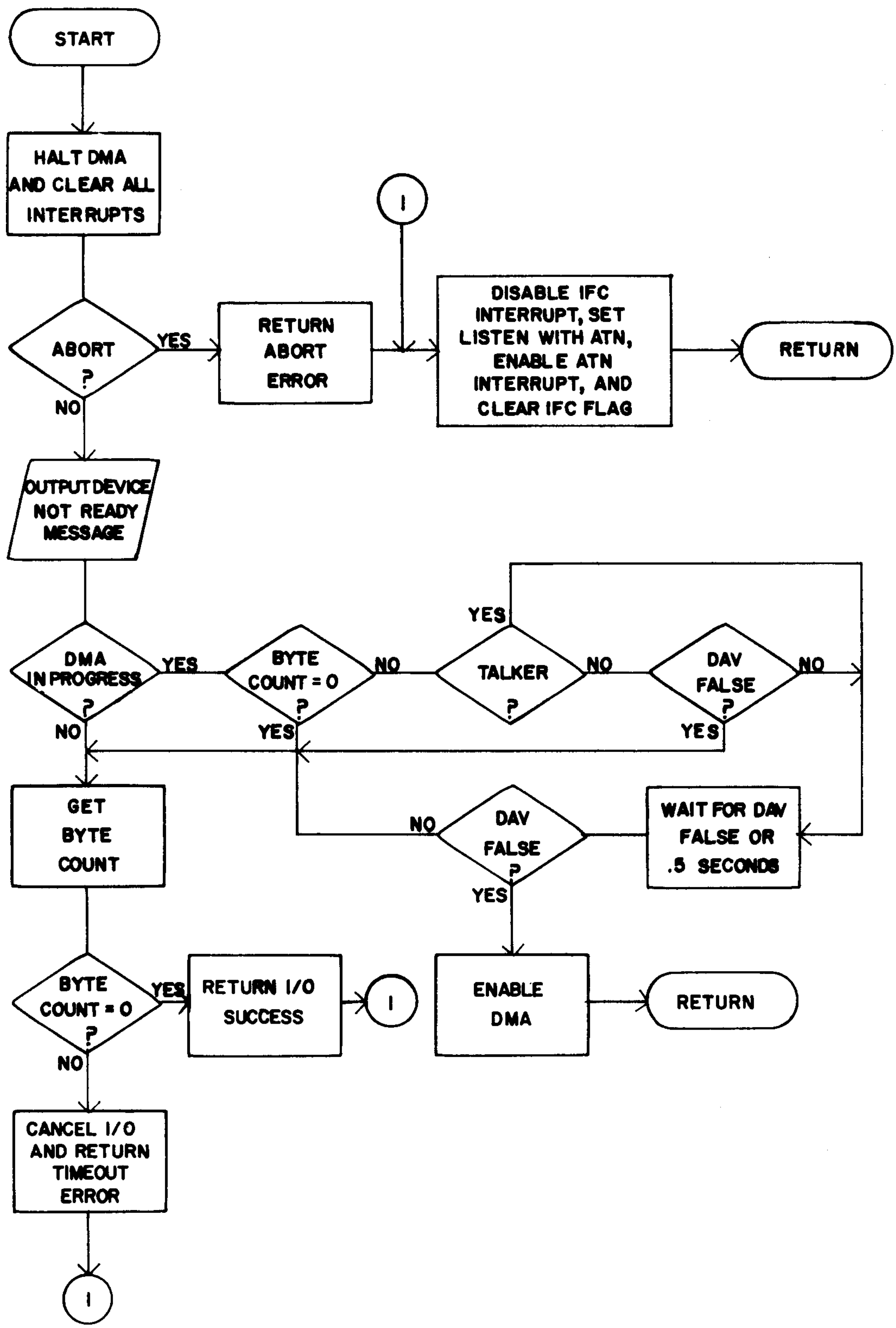

Figure 2. Device timeout (GPOUT).

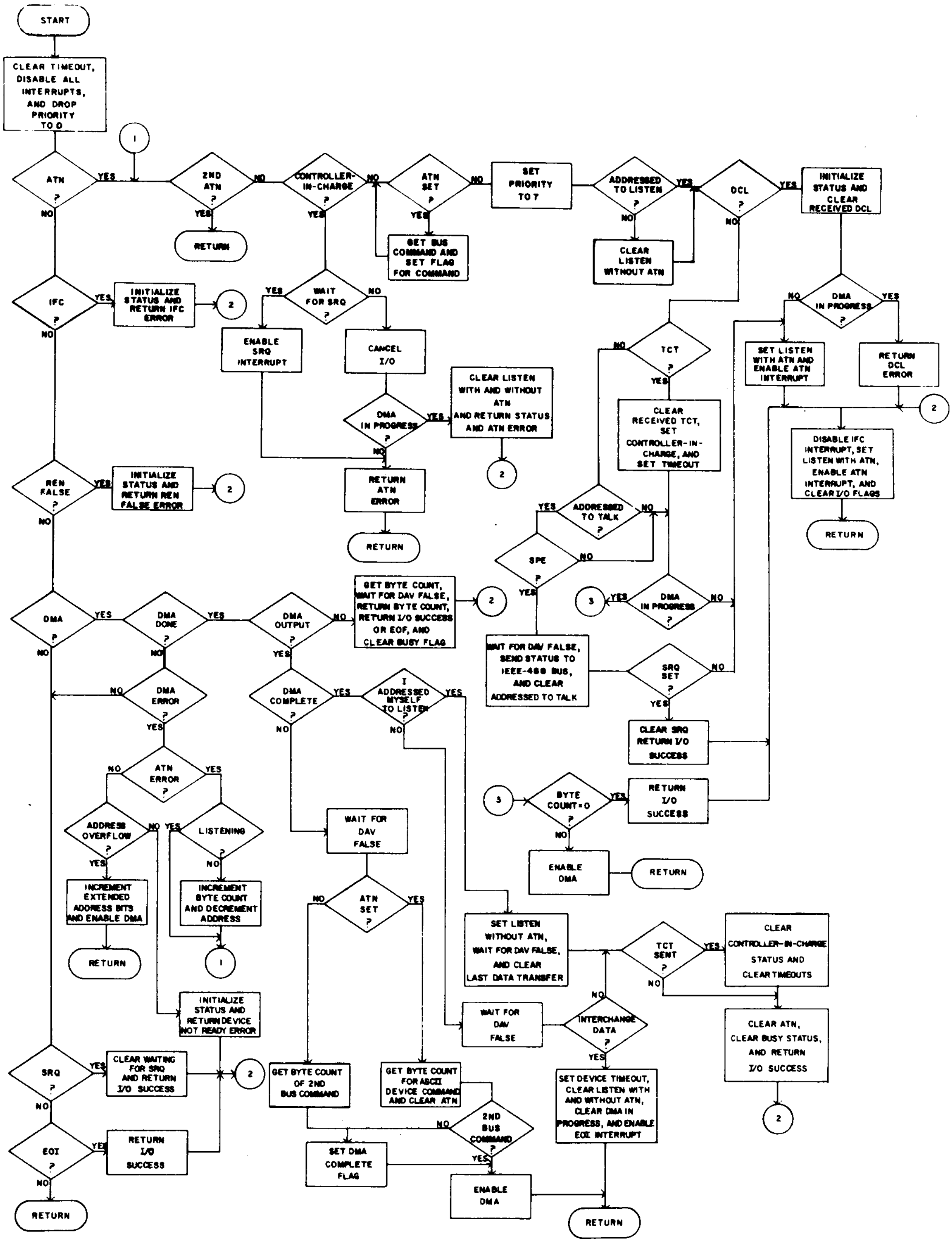

Figure 3. Device interrupts ($GPINT).

If GPDRV is Controller-in-Charge, it is an error and I/O is canceled. If no I/O is in progress and ATN goes false before I/O is attempted, an error is not reported and the I/O will continue normally.

If GPDRV is not Controller-in-Charge and ATN is set true, all Bus Commands that come across the IEEE-488 Bus are checked to see if they are valid for GPDRV. If they are valid, the appropriate flags are set. When ATN goes false, the Processor Status priority is set to 7 to prevent all interrupts and action is then taken on the flags set while ATN was true.

If a DCL was received and I/O was in progress, a DCL error is returned; otherwise, the ATN Interrupt is reenabled and GPDRV returns control to the System.

If GPDRV received a TCT command, it takes control of the IEEE-488 Bus by enabling timeouts and setting GPDRV Status to Controller-in-Charge. I/O is restarted if it was in progress when ATN was set true, the ATN Interrupt is enabled, and control is returned to the System.

If GPDRV was addressed to talk, a check is made to see if SPE was received. If SPE was received, GPDRV Status is sent on the IEEE-488 Bus, clears the Addressed to Talk flag, and clears SRQ if set true. If I/O was in progress it is restarted, the ATN Interrupt is enabled, and control is returned to the System.

All other flags set while ATN was true require no further action; therefore, the ATN Interrupt is enabled. If I/O was in progress, it is restarted and control is returned to the System.

3.3.2 IFC Interrupt

When an IFC Interrupt occurs, GPDRV and the CP1100 Interface are set to their power-up state and an IFC occurred error is returned.

3.3.3 REN False Interrupt

When REN False Interrupt occurs, the CP1100 Interface is cleared and a REN False occurred error is returned. This interrupt occurs only during DMA.

3.3.4 DMA Done Interrupt

When the DMA Done Interrupt occurs, the DMA conditions on the CP1100 Interface are cleared and a check is made to see if the DMA was input or output.

If DMA input completed, the number of bytes received is placed in the 2nd I/O Status Word. Either I/O success or End-of-File is returned. An End-of-File is returned if an EOI was received.

If DMA output completed, a check is made to see if all output from the QIO is complete. If not, the next part of the I/O from the QIO is started. If output from QIO is complete, a test to see if GPDRV addressed itself to listen is made. If GPDRV addressed itself to listen, listen without ATN is set to prevent the loss of data. If GPDRV did not address itself to listen, a check is made to see if an Interchange of Data was set up. If an Interchange of Data was set up, listen with and without ATN was cleared and the EIO interrupt is enabled to let GPDRV know when the Interchange is complete. GPDRV then returns to the System. If an Interchange of Data was not set up, a check is then made to see if TCT was sent. If TCT was sent, GPDRV gives up control by disabling timeouts, clearing Controller-in-Charge Status, and returning successful I/O complete. If TCT was not sent, successful I/O completion is returned.

3.3.5 DMA Error Interrupt

When the DMA Error Interrupt occurs, it can be caused by ATN being set, an overflow of the BAR, or a hardware error on the CP110 Interface or the IEEE-488 Bus.

If ATN is set and GPDRV was listening, a jump is made to the ATN Interrupt routine. If ATN is set and GPDRV was talking, a check is made to see if GPDRV has ATN set. If GPDRV has ATN set, the error was either a BAR overflow or a hardware error. If GPDRV does not have ATN set, the byte count is incremented and the bus address is decremented so the last byte will be sent again. A jump is then made to the ATN Interrupt routine.

If an overflow of the BAR caused the error, the extended address bits are incremented and the DMA is continued.

If a hardware error occurred, the CP1100 Interface is cleared and a Device not Ready error is returned.

3.3.6 SRQ Interrupt

When an SRQ Interrupt occurs, the waiting for SRQ status is cleared and I/O success is returned.

3.3.7 EOI Interrupt

When the EOI Interrupt occurs, successful I/O completion is returned to the program.

3.4 CANCEL I/O OPERATION

The Cancel I/O Operation (GPCAN) is called when a program is aborted. (See Figure 4.) A check is made to see if the request is for the Current Task. If not, a return is made to the System. If it is for the Current Task, any I/O in progress is canceled by sending a DCL, Local Reset, and if the CP1100 Interface is a System Controller, doing an IFC. All status is set to the power-on state and an abort error is returned.

3.5 COMMON ROUTINES

The Common Routines are those used to accomplish a specific task by several segments of GPDRV.

3.5.1 Cancel I/O on IEEE-488 Bus

Cancel I/O on the IEEE-488 Bus (CANIO) disables all interrupts and cancels I/O by doing a DCL, a Local Reset; and if the CP1100 Interface is a System Controller, an IFC. (See Figure 5.) CANIO waits for DAV to go false before each operation is attempted. The wait is never more than 0.5 seconds. After canceling the I/O, all status is reset to the power-up state by using the Subroutine Initialize Status (INIS).

3.5.2 Get Absolute Address

Get Absolute Address (GABSA) converts a virtual address in R2 and an APR value in R3 to an absolute address in R2 with the extended address bits in R3. (See

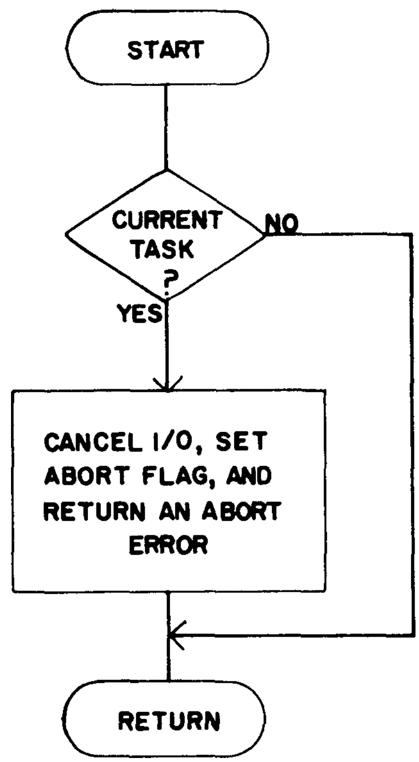

Figure 4. Cancel I/O operation (GPCAN).

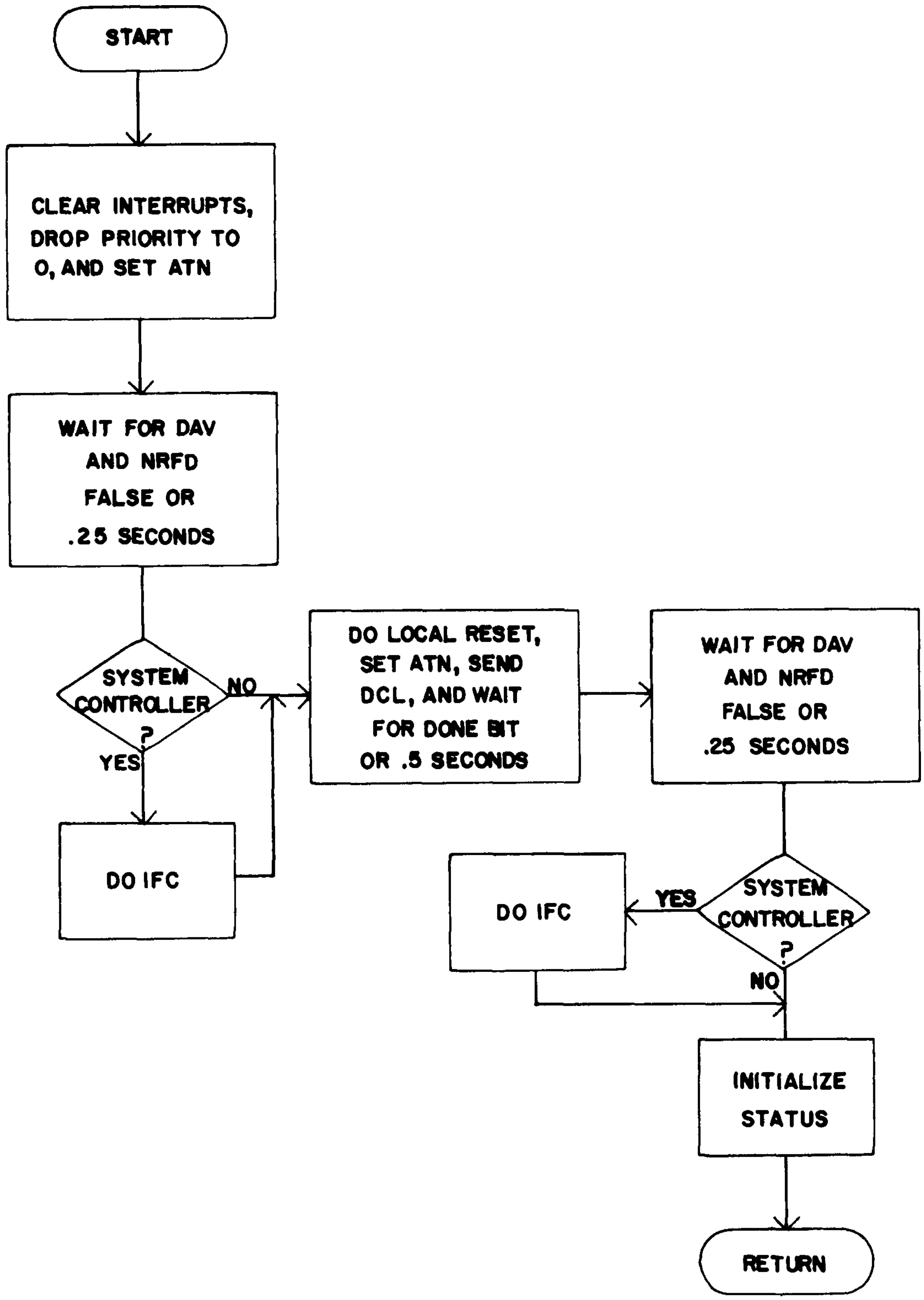

Figure 5. Cancel I/O on IEEE-488 bus (CANCIO).

Figure 6.) This routine sets up the absolute address to be used in the routine Start DMA (SDMA) where it loads the absolute address in the CP1100 registers.

3.5.3 Initialize Status

Initialize Status (INIS) sets the initial IEEE-488 status in GPDRV and on the CP1100 Interface. (See Figure 7.) A Local Reset is done to clear the CP1100 Interface and then Listen with ATN is set and the ATN interrupt is enabled. GPDRV status is cleared and timeouts are disabled. A check is then made to see if the CP1100 Interface is a System Controller; if not, a return is executed. If the CP1100 is System Controller, timeouts are set to the default timeout and the Status is set to Controller-in-Charge.

3.5.4 Enable DMA

Enable DMA (SDMA) has four entry points: SDMA, CDMA, ADMA, and TDMA. When called at SDMA, a complete initialization is done for DMA. (See Figure 8.) The last 3 entry points (CDMA, ADMA, and TDMA) are to restart DMA after it has been stopped.

SDMA, loads the BAR with the absolute address and sets the extended address bits in the BCR for the buffer.

CDMA, sets the DMA in progress flag and sets the 2's-complement of R0 (the byte count) in the CTR.

ADMA, initializes the CP1100 Interface; and if GPDRV is not Controller-in-Charge, checks to see if it is addressed to talk or listen (whichever operation is to be performed). If GPDRV is Controller-in-Charge, or is addressed to talk or listen; it will continue to initialize the DMA. If not, it will wait until GPDRV is addressed to talk or listen.

TDMA clears listen with and without ATN, clears the ATN interrupt enable, sets ATN if needed, enables DMA, and returns to the System.

3.5.5 Wait for DAV False

Wait for DAV False (DAVLOW) will wait until DAV is unasserted or 0.7 seconds, whichever occurs first.

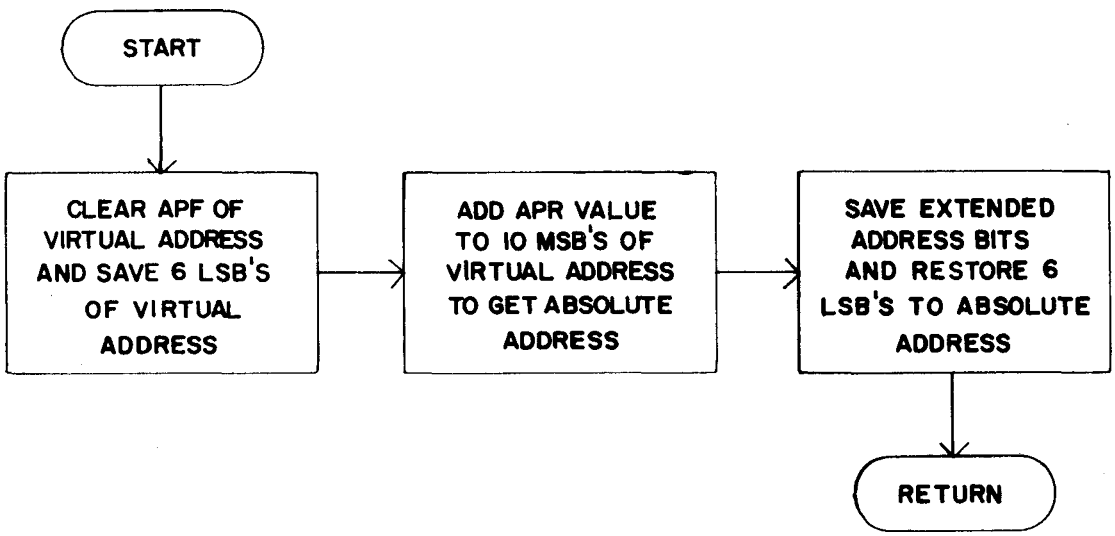

Figure 6. Get absolute address (GABSA).

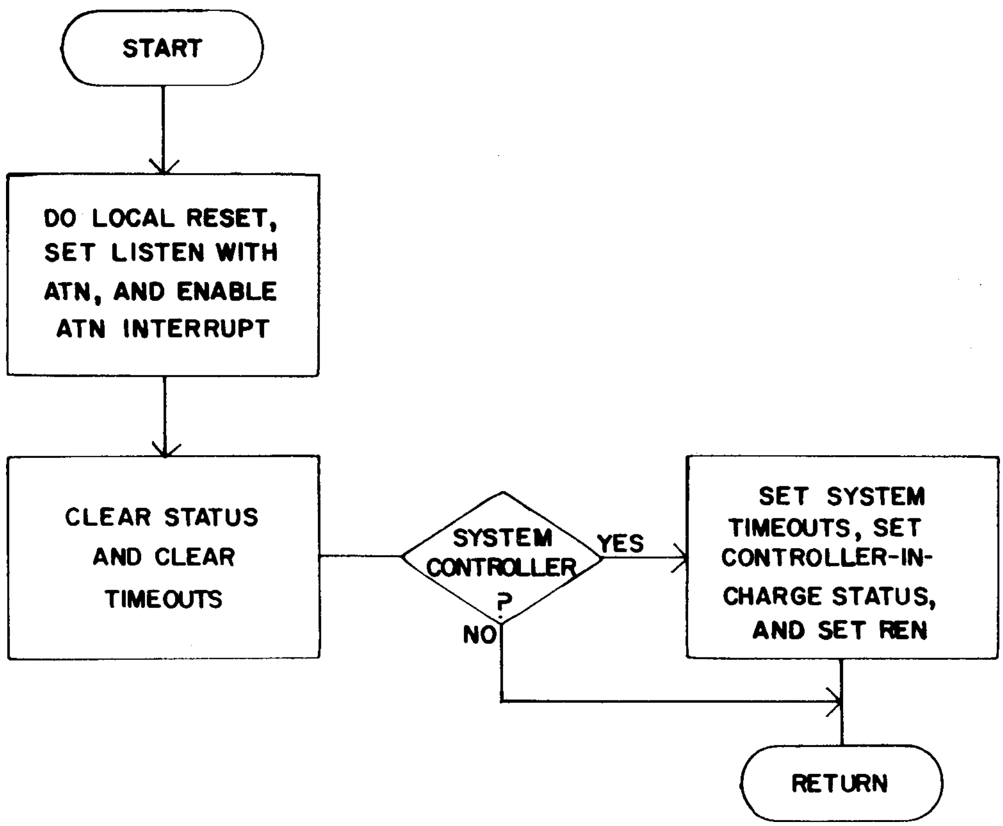

Figure 7. Initialize status (INIS).

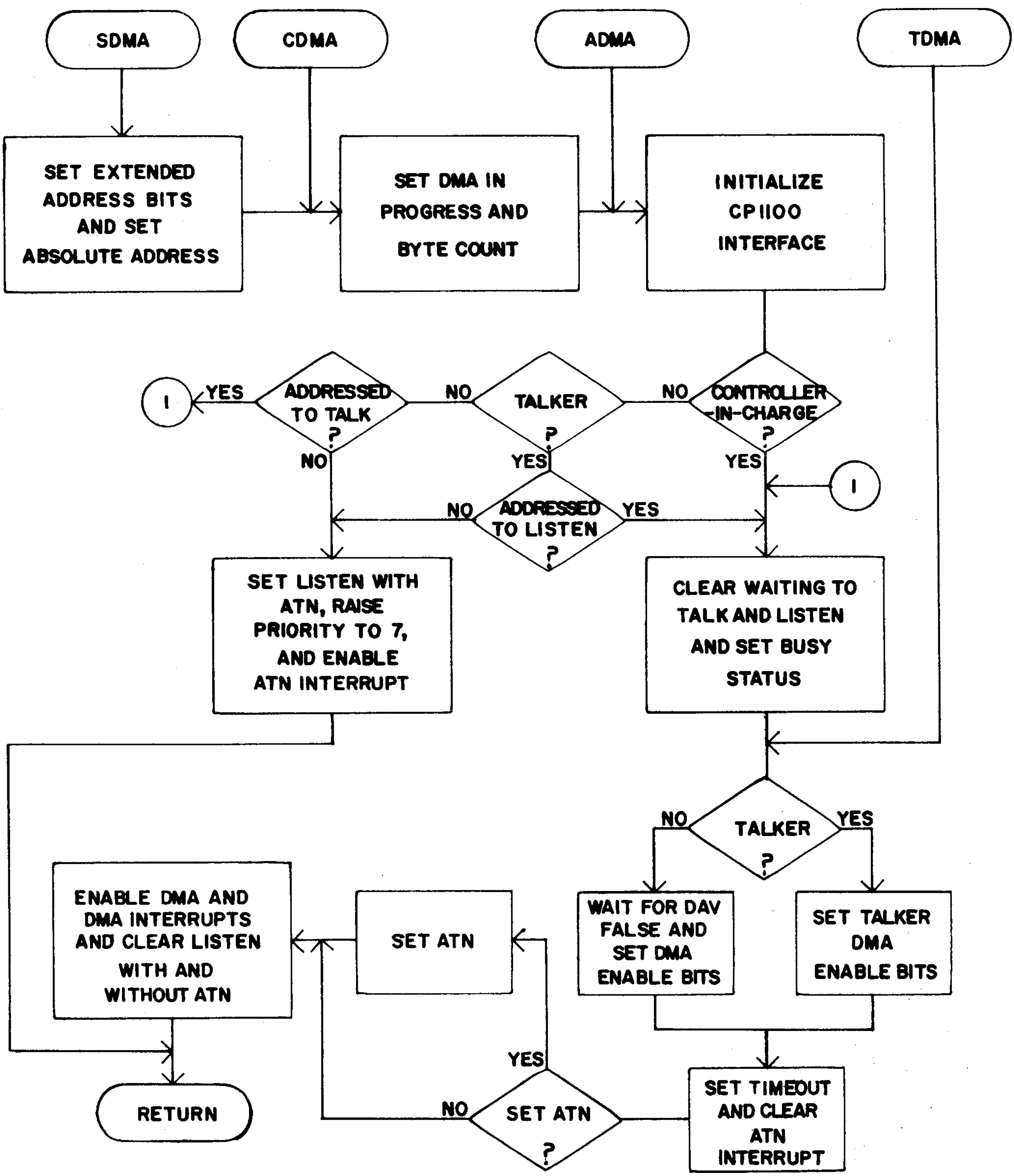

Figure 8. Enable DMA (SDMA, CDMA, ADMA, and TDMA).

3.5.6 Convert Bus Command to Binary

Convert Bus Commands to Binary (CNVB) gets a byte sent to GPDRV by using the $GTBYT RSX-11M routine, compares it to a number; and if the byte is a number, converts it to binary. (See Figure 9.) If the byte is not a number, the subroutine returns. After a Bus Command or Bus Address is converted, the byte is saved by the subroutine Save Byte (SVBYTE). When CNVB has converted all the Bus Commands and Bus Addresses, it checks to see if both a talker and a listener were addressed; and if they were, sets the Interchange Data Flag.

3.5.7 Save Byte of Bus Command

Save Byte of Bus Command (SVBYTE) saves a byte that is a Bus Command or a Bus Address in an internal buffer in the UCB. (See Figure 10.) SVBYTE then checks the byte and sets the appropriate flag if a talker was addressed, a listener was addressed, or TCT was sent after a talker was addressed. The byte count is kept in R0.

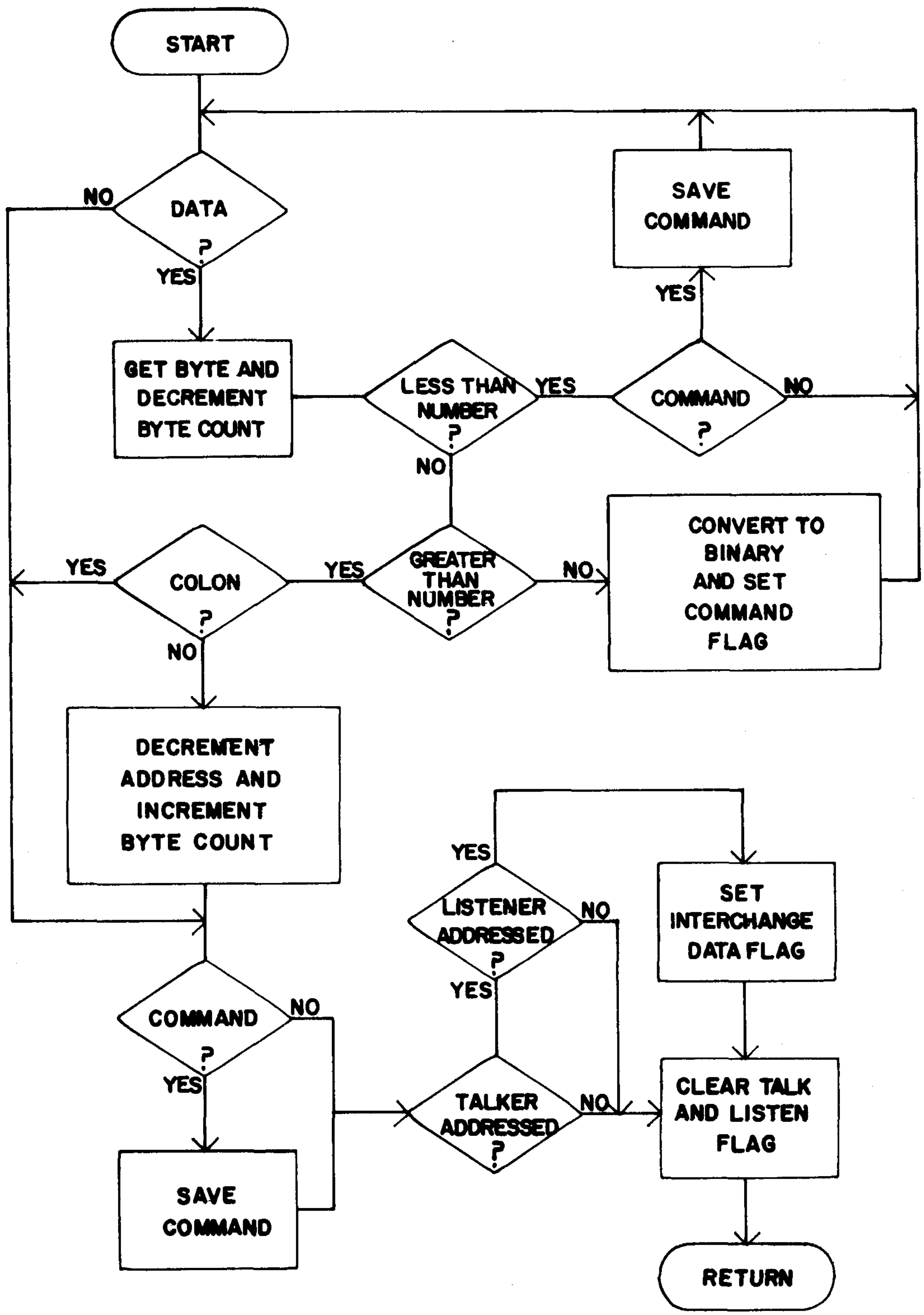

Figure 9. Convert bus command to binary (CNVB).

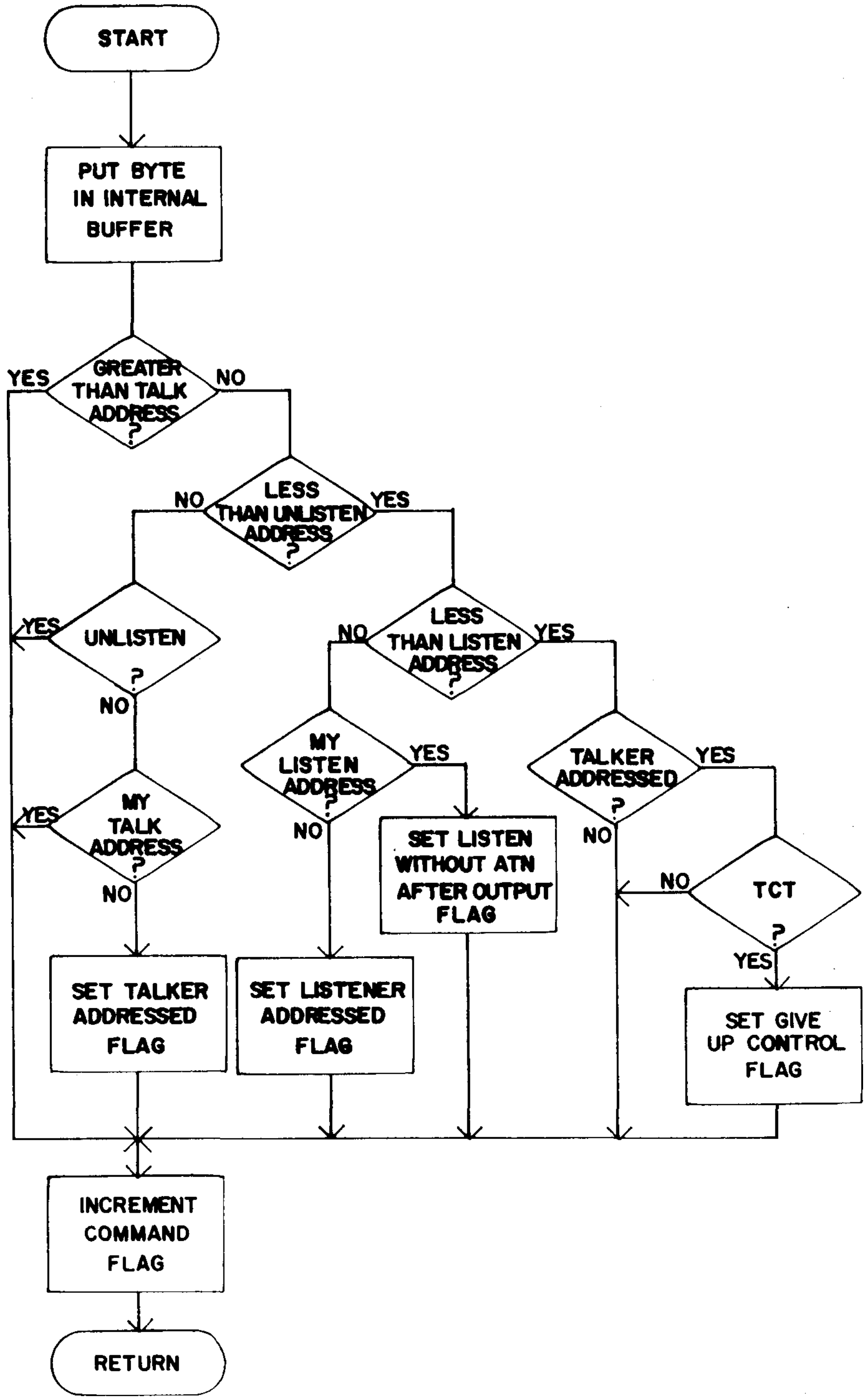

Figure 10. Save byte of bus command (SVBYTE).

APPENDIX A

BUILDING GPDRV

To build GPDRV for a mapped RSX-11M System (Version 3.1), the files RSXMC, MAC, EXEMC.MLB, and EXELIB.OLB must be on the disc LB0: under the UIC [1,1] and the file RSX11M.STB must be on disc LB0: under the UIC [1,54]. Place the floppy for building GPDRV in DX0: and use the following commands:

```
MOU DX0:GPDRV
SET /UIC=[1,54]
```

Edit GPDAT. MAC on DX0: and set the variable G$$P11 equal to the number of units desired. Repeat the Macro DEV once for each unit.

```
G$$P11=2
DEV G,P,0,120,164000,340,1

         0 = Unit Number
       120 = Vector Address
    164000 = Device Address
       340 = Interrupt Priority
         1 = Timeout In Seconds

DEV G,P,1,124,164020,340,1
```

Then edit GPDRV. MAC on DX0: and set G$$P11 equal to the number of units desired.

Execute the command file GPDRV.CMD to complete the building of GPDRV.

```
@DX0:GPDRV
```

To produce listings of GPDAT. MAC and GPDRV. MAC, use the Command File GPLST. CMD.

```
@DX0:GPLST
```

APPENDIX B

I/O FUNCTIONS

Function = Octal Value

IO. ATT	=	1400	Attach Device
IO. DET	=	2000	Detach Device
IO. RLB	=	1000	Read Logical Block
IO. WLB	=	400	Write Logical Block
IO. CTI	=	15400	Change Timeout Low Order Byte = Timeout In Seconds A 0 Disables Timeouts
IO. ADS	=	14000	Alter Device Status Low Order Byte = Status

Alter Status = Status

14000 = 000	Clear Status
14001 = 001	Want Control
14002 = 002	Waiting To Receive Data
14004 = 004	Waiting To Send Data
14010 = 010	Busy
14020 = 020	Set SRQ
14040 = 040	Wait For SRQ
14100 = 100	Send IFC
14200 = 200	Read Status

When the Status is read, all bits (except for 200) have the same meaning as when sent. The 200 means GPDRV is Controller - In - Charge.

APPENDIX C

ERROR CODES

Name	Value	=	Meaning
	0	=	Device Not Loaded (Returned By Attach)
IE.IFC	-2	=	Illegal Function
IE.DNR	-3	=	Device Not Ready
IE.EOF	-10	=	End-Of-File
IE.ABO	-15	=	I/O Aborted (Handled By The System)
	-68	=	IFC Received Or Sending IFC When Not System Controller
	-69	=	DCL Or SDC Received
	-70	=	REN False Occurred
	-71	=	ATN Occurred While Controller-In-Charge Or Setting ATN When Not Controller-In-Charge (IOSB(2) = IEEE-488 Status To Check For Controller-In-Charge)
	-72	=	Waiting For SRQ When Not Controller-In-Charge Or Setting SRQ When Controller-In-Charge (IOSB(2) = IEEE-488 Status)
	-73	=	Changing Timeout When Not Controller-In-Charge
IE.TMO	-74	=	Timeout

APPENDIX D

CP1100 INTERFACE MODIFICATIONS

In order to modify the CP1100 Interface so that all eight interrupts will interrupt through the same vector, pin 4 of IC's U1350, U1220, and U1230 must be lifted and tied to either ground or +5V. To set the desired Vector Address for bits 2-4, use the following:

BIT	PIN and IC
2 = 0	Tie pin 4 of U1230 to GND.
2 = 1	Tie pin 4 of U1230 to +5V.
3 = 0	Tie pin 4 of U1220 to GND.
3 = 1	Tie pin 4 of U1220 to +5V.
4 = 0	Tie pin 4 of U1350 to GND.
4 = 1	Tie pin 4 of U1350 to +5V.

To set the CP110 Interface as System Controller, lift pin 2 of U950 and tie it to +5V.

To set the CP1100 Interface as Not System Controller, leave pin 2 of U950 inserted in the socket. To prevent IFC and REN from being asserted, lift pin 4 and pin 13 of U1520 and tie them to GND.

APPENDIX E

SAMPLE PROGRAM

```
        DIMENSION IOSB(2)
        DATA IS/'S'/,IT/'T'/,IQ/'Q'/,IR/'R'/,IW/'W'/,IBL/' '/,IU/'U'/
        1,IE/'E'/,IOATT/"1400/,IODET/"2000/
        WRITE(5,101)
        READ(5,102)IUNT
101     FORMAT(' ENTER UNIT NUMBER # ',$)
102     FORMAT(1I4)
        CALL ASNLUN(1,'GP',IUNT,4)
        CALL WTQIO(IOATT,1,2,,IOSB)
        IF(IOSB(1).NE.1)GO TO 40
4       WRITE(5,110)
110     FORMAT(' S = CHANGE STATUS',/' T = CHANGE TIMEOUT',
        1/' W = WRITE',/' R = READ',/' Q = QUIT',
        2/' RS# = READ STATUS (# = NUMBER)',/' RU# = READ UNIT',
        1/' RE# = REP 1 UNIT')
5       WRITE(5,100)
100     FORMAT(' ENTER COMMAND CHARACTER * ',$)
        READ(5,200)IDAT,IDAT1,NUM
        IF(IDAT.EQ.IS)GO TO 6
        IF(IDAT.EQ.IT)GO TO 7
        IF(IDAT.EQ.IW)GO TO 8
        IF(IDAT.EQ.IR)GO TO 9
        IF(IDAT.EQ.IQ)GO TO 10
        GO TO 4
6       CALL CST
        GO TO 5
7       CALL CTI
        GO TO 5
8       CALL W
        GO TO 5
9       IF(IDAT1.EQ.IBL)CALL R
        IF(IDAT1.EQ.IS)CALL RS(NUM)
        IF(IDAT1.EQ.IU)CALL RU(NUM)
        IF(IDAT1.EQ.IE)CALL RE(NUM)
        GO TO 5
10      CALL WTQIO(IODET,1,2,,IOSB)
        IF(IOSB(1).EQ.1)GO TO 50
40      WRITE(5,206)IOSB(1),IOSB(2)
50      CALL CLOSE (1)
        STOP
200     FORMAT(2A1,I5)
206     FORMAT(2O8)
        END
```

```
C  WRITE TO IEEE-488 BUS
        SUBROUTINE W
        DIMENSION IDAT(40),IOSB(2)
        DATA ID/' '/,IP/';'/,IREAD/"1000/,IOATT/"1400/,IODET/"2000/
        IFLG=0
        CALL ERRSET(38,,.FALSE.,,.FALSE.,)
5       WRITE(5,100)
100     FORMAT(' ENTER 488 COMMAND STRING * ',$)
        READ(5,200)(IDAT(I),I=1,40)
        IF(IDAT(1).EQ.ID)RETURN
200     FORMAT(40A1)
        N=40
        DO 10 I=1,40
        IF(IDAT(N).NE.ID)GO TO 11
10      N=N-1
11      WRITE(5,102)(IDAT(I),I=1,N)
        WRITE(1,101,ERR=50)(IDAT(I),I=1,N)
101     FORMAT(40A1)
102     FORMAT(' COMMAND = ',40A1)
        RETURN
50      CALL ERRSNS(NUM,IFCSE,IFCSE1,IUNIT)
        WRITE(5,300)IFCSE,IFCSE1
300     FORMAT(' ',2I6)
51      IF(IFCSE.EQ.-87)WRITE(5,110)
        IF(IFCSE.EQ.-3)WRITE(5,111)
110     FORMAT(' IFC RECIEVED')
111     FORMAT(' DEVICE NOT READY')
        RETURN
        END


C  READ FROM THE IEEE-488 BUS
        SUBROUTINE R
        DIMENSION IDAT(40),IOSB(2)
        DATA ID/' '/,IP/';'/,IREAD/"1000/,IOATT/"1400/,IODET/"2000/
        CALL ERRSET(39,,.FALSE.,,.FALSE.,)
20      READ(1,101,ERR=50)(IDAT(I),I=1,40)
        WRITE(5,102)(IDAT(I),I=1,40)
        WRITE(5,103)(IDAT(I),I=1,10)
102     FORMAT(' COMMAND = ',40A1)
103     FORMAT(' OCTAL = ',10O6)
        RETURN
50      CALL ERRSNS(NUM,IFCSE,IFCSE1,IUNIT)
        WRITE(5,300)IFCSE,IFCSE1
300     FORMAT(' ',2I6)
51      IF(IFCSE.EQ.-87)WRITE(5,110)
        IF(IFCSE.EQ.-3)WRITE(5,111)
110     FORMAT(' IFC RECIEVED')
111     FORMAT(' DEVICE NOT READY')
        RETURN
101     FORMAT(40A1)
        END
```

```
C  ALTER DEVICE STATUS
         SUBROUTINE CST
         DIMENSION IOSB(2)
         DATA ID/' '/,IP/';'/,IREAD/"1000/,IOATT/"1400/,IODET/"2000/
         1,IOADS/"14000/,SRQ/"20/,IOADSR/"14200/
10       WRITE(5,200)
200      FORMAT(' ENTER STATUS (OCT) # ',$)
         READ(5,100)IST
100      FORMAT(107)
         IF(IST.GE.256)GO TO 70
         IOADS1=IOADS+IST
         CALL WTQIO(IOADS1,1,2,,IOSB)
60       WRITE(5,206)IOSB(1),IOSB(2)
206      FORMAT(208)
70       RETURN
         END

C  CHANGE TIMEOUT
         SUBROUTINE CTI
         DIMENSION IOSB(2)
         DATA ID/' '/,IP/';'/,IREAD/"1000/,IOATT/"1400/,IODET/"2000/
         1,IOCTI/"15400/,SRQ/"20/,IOADSR/"14200/
10       WRITE(5,200)
200      FORMAT(' ENTER TIMEOUT (OCT) # ',$)
         READ(5,100)IST
100      FORMAT(107)
         IF(IST.GE.256)GO TO 70
         IOCTI1=IOCTI+IST
         CALL WTQIO(IOCTI1,1,2,,IOSB)
         IF(IOSB(1).EQ.1)RETURN
         WRITE(5,206)IOSB(1),IOSB(2)
206      FORMAT(208)
70       RETURN
         END

C  READ STATUS FROM 7912AD
         SUBROUTINE RS(NUM)
         DIMENSION IOSB(2),IPARAM(6)
         DATA ID/' '/,IP/';'/,IREAD/"1000/,IOATT/"1400/,IODET/"2000/
         IPARAM(2)=1
         CALL GETADR(IPARAM(1),IDAT)
         IADD=100+NUM
         IDAT=0
         CALL ERRSET(38,,.FALSE.,,.FALSE.,)
         WRITE(1,104,ERR=50)IADD
104      FORMAT('77,137,50,30,',I13,'/140')
         CALL WTQIO(IREAD,1,2,,IOSB,IPARAM)
         WRITE(1,105,ERR=50)
105      FORMAT('77,137,31')
40       WRITE(5,103)IDAT
103      FORMAT(' OCTAL = ',106)
         RETURN
50       CALL ERRSNS(NUM,IFCSE,IFCSE1,IUNIT)
         WRITE(5,110)IFCSE,IFCSE1
110      FORMAT(2I5)
         RETURN
         END
```

```
C  READ DATA FROM 7912AD
        SUBROUTINE RU(NUM)
        DIMENSION IDAT(2048),IOSB(2),IPARAM(6)
        DATA ID/' '/,IP/';'/,IREAD/"1000/,IOATT/"1400/,IODET/"2000/
        IADD=100+NUM
        IADD1=40+NUM
        ILUN=1
        IEFLG=2
        CALL GETADR(IPARAM(1),IDAT(1))
        IPARAM(2)=4096
        CALL ERRSET(38,,.FALSE.,,.FALSE.,)
        CALL ERRSET(39,,.FALSE.,,.FALSE.,)
        IFLG=0
        WRITE(1,200,ERR=50)
200     FORMAT('41,140:READ PTR;')
        WRITE(1,201,ERR=50)
201     FORMAT('50,101/140')
101     FORMAT(40A2)
20      CALL WTQIO(IREAD,ILUN,IEFLG,,IOSB,IPARAM)
        IF(IOSB(1).NE.1)GO TO 51
30      N=(IOSB(2)/2)+1
        WRITE(5,302)IDAT(1),IDAT(1)
302     FORMAT(O8,1A2)
        WRITE(5,103)(IDAT(I),I=2,N)
        WRITE(1,202,ERR=50)
202     FORMAT('41,140:READ VER;')
        WRITE(1,201,ERR=50)
        CALL WTQIO(IREAD,ILUN,IEFLG,,IOSB,IPARAM)
        IF(IOSB(1).NE.1)GO TO 51
40      N=(IOSB(2)/2)+1
        WRITE(5,302)IDAT(1),IDAT(1)
        WRITE(5,103)(IDAT(I),I=2,N)
103     FORMAT(14I5)
        RETURN
50      CALL ERRSNS(NUM,IFCSE,IFCSE1,IUNIT)
        WRITE(5,206)IFCSE,IFCSE1
        RETURN
51      WRITE(5,206)IOSB(1),IOSB(2)
206     FORMAT(2I5)
        RETURN
        END
```

```
C  DO A REP 1 TO GET DATA FROM 7912AD
        SUBROUTINE RE(NUM)
        DIMENSION IDAT(4096),IOSB(2),IPARAM(6)
        DATA ID/' '/,IP/';'/,IREAD/"1000/,IOATT/"1400/,IODET/"2000/
        IADD=100+NUM
        IADD1=40+NUM
        ILUN=1
        IEFLG=2
        CALL GETADR(IPARAM(1),IDAT(1))
        IPARAM(2)=8192
        CALL ERRSET(38,,.FALSE.,,.FALSE.,)
        CALL ERRSET(39,,.FALSE.,,.FALSE.,)
        IFLG=0
        WRITE(1,203,ERR=50)
203     FORMAT('41,140:REP 1;')
        WRITE(1,201,ERR=50)
201     FORMAT('50,101/140')
101     FORMAT(40A2)
C       IF IOSUB(1) = 1, THEN IS.SUC. OR IOSUB(1) = 246, THEN IE.EOF
20      CALL WTQIO(IREAD,ILUN,IEFLG,,IOSB,IPARAM)
C IF AN IFC OCCURED IOSB(1)=188
        IF(IOSB(1).EQ.188)IFCSE=-68
        WRITE(5,206)IOSB(1),IOSB(2)
206     FORMAT(2I5)
        IF(IOSB(1).EQ.182)WRITE(5,301)
301     FORMAT(' TIME OUT OCCURED')
        IF((IOSB(1).EQ.246).OR.(IOSB(1).EQ.182))GO TO 30
        IF(IOSB(1).NE.1)GO TO 51
30      N=(IOSB(2)/2)+1
        WRITE(5,302)IDAT(1),IDAT(1)
302     FORMAT(O8,1A2)
        WRITE(5,103)(IDAT(I),I=2,N)
103     FORMAT(14I5)
        RETURN
50      CALL ERRSNS(NUM,IFCSE,IFCSE1,IUNIT)
51      IF(IFCSE.EQ.-68)WRITE(5,110)
        IF(IFCSE.EQ.-3)WRITE(5,111)
110     FORMAT(' IFC RECIEVED')
111     FORMAT(' DEVICE NOT READY')
        RETURN
        END
```

REFERENCES

1. Tektronix CP1100/IEEE 488 Interface Instruction Manual, Tektronix, Inc., Beaverton, Oregon, March 1978.

2. RSX-11M Executive Reference Manual, Digital Equipment Corporation, Maynard, Massachusetts, December 1977.

3. IAS/RSX FORTRAN IV User's Guide, Digital Equipment Corporation, Maynard, Massachusetts, March 1978.

4. IAS/RSX-11 I/O Operations Reference Manual, Digital Equipment Corporation, Maynard, Massachusetts, December 1977.

5. RSX-11M Guide to Writing an I/O Driver, Digital Equipment Corporation, Maynard, Massachusetts, December 1977.

6. RSX-11M I/O Drivers Reference Manual, Digital Equipment Corporation, Maynard, Massachusetts, December 1977.

7. Condensed Description of the Hewlett Packard Interface Bus, Hewlett-Packard Company, Loveland, Colorado, 1975.