

000 000 00000000
0000 000 000000000
00000 000 000000000
000000 000 000 000
0000000 000 000 000
000 00000 000000000
000 00000 000000000
000 000 000000000
000 000 000000000

Norsk Data A.S M A N U A L

G P I B O N N D - 1 0 0

P R E L I M I N A R Y

Written: 23 November 1982

T A B L E O F C O N T E N T S

<u>Section</u>	<u>Page</u>
1 T H E G P I B S Y S T E M	2
1.1 IMPLEMENTATION OF THE IEEE-488 INTERFACE FUNCTION ON ND-100	2
1.2 A GPIB-system with ND-100	3
2 U S I N G T H E G P I B - S Y S T E M	4
2.1 A MULTI USER GPIB SYSTEM	4
3 H A R D W A R E	5
3.1 MICROPROCESSOR SOFTWARE	8
4 S O F T W A R E	10
4.1 HOW TO START AND STOP THE GPIB-DRIVER	12
4.2 GPIB-LIBRARY	12
4.2.1 LOG ON AS NONPRIVILDEGED USER	13
4.2.2 LOG OFF GPIB CONTROLLER	13
4.2.3 WRITE ERROR MESSAGE AND STOP	14
4.2.4 WRITE ERROR MESSAGE AND CONTINUE	14
4.2.5 READ DATA FROM DEVICE(S)	14
4.2.6 SEND DATA TO DEVICE(S)	14
4.2.7 TRANSFER DATA BETWEEN DEVICES	14
4.2.8 TRIGGER DEVICE(S)	14
4.2.9 ENABLE SRQ INTERRUPT FROM DEVICE(S)	14
4.2.10 DISABLE SRQ INTERRUPT	15
4.2.11 CHECK/WAIT FOR SRQ-INTERRUPT	15
4.2.12 CLEAR DEVICE(S)	15
4.2.13 SET DEVICE(S) IN LOCAL	15
4.2.14 RESERVE DEVICE(S)	15
4.2.15 RELEASE DEVICE(S)	15
4.2.16 LIST THE USERS DEVICE(S)	16
4.2.17 LIST THE SYSTEMS DEVICE(S)	16
4.2.18 GET DEVICE NUMBER	16
4.2.19 LOG ON AS PRIVILIDGED USER	16
4.2.20 CONFIGURE DEVICE	16

<u>Section</u>	<u>Page</u>
4.2.21 DELETE DEVICE	16
4.2.22 SEND PARALLELL POLL CONFIGURATION BYTE TO DEVICE	16
4.2.23 PASS CONTROLL TO DEVICE	16
4.2.24 SET ND-100 PARALLELL POLL STATUS	17
4.2.25 SET ND-100 SERIAL POLL STATUS	17
4.2.26 SET ND-100 TALKER ADDRESS	17
4.2.27 SEND INTERFACE CLEAR	17
4.2.28 SET LISTEN ONLY MODE	17
4.2.29 SET TALKER ONLY MODE	17
4.2.30 RESET ONLY MODE	17
4.2.31 CLEAR CONTROLLER	17
4.3 GPIB MONITOR	18
4.3.1 LOG-ON	18
4.3.2 LOG-OFF	18
4.3.3 SET-DEFAULT-CONTROLLER	19
4.3.4 RESET-DEFAULT-CONTROLLER	19
4.3.5 CONFIGURE-DEVICE	19
4.3.6 DELETE-DEVICE	20
4.3.7 RECONFIGURATE	21
4.3.8 CONTROLLER-CLEAR	21
4.3.9 SET-CONTROLLER-ADDRESS	21
4.3.10 RESERVE-DEVICE	22
4.3.11 RELEASE-DEVICE	22
4.3.12 TRIGGER-DEVICES	22
4.3.13 CLEAR-DEVICES	22
4.3.14 LOCAL-DEVICES	23
4.3.15 INTERFACE-CLEAR	23
4.3.16 SEND-ASCII-STRING	23
4.3.17 SEND-DATA-BUFFER	23
4.3.18 RECEIV-DATA	24
4.3.19 DUMP-DATABUFFER	24
4.3.20 SAVE-DATABUFFER	25
4.3.21 LOAD-DATABUFFER	26
4.3.22 DATA-BUFFER-CLEAR	27
4.3.23 LIST-SYSTEM-DEVICES	27
4.3.24 LIST-USER-DEVICES	27
4.3.25 HELP	27
4.3.26 GET-ERRORMESSAGE	28
4.3.27 GET-FUNCTION	28
4.3.28 PERMANENT-SRQ-INTERRUPT-ENABLE	28
4.3.29 TEMPORARY-SRQ-INTERRUPT-ENABLE	29
4.3.30 SRQ-INTERRUPT-DISABLE	29
4.3.31 CHECK-FOR-SRQ-INTERRUPT	29
4.3.32 WAIT-FOR-SRQ-INTERRUPT	30
4.3.33 CHANGE-SRQ-WAIT-TIME	30
4.3.34 EXIT	30

A P P E N D I X

A	MONITOR COMMANDS	31
---	----------------------------	----

1 T H E G P I B S Y S T E M

The GPIB is an IEEE standard defined for use as a Digital Interface for Programmable Instrumentation. (ANSI/IEEE Std. 488-1978). It is also an IEC standard (EIC-625). The interface bus is called GPIB, HP-IB, ASCII-bus, IEEE-488-bus or IEC-bus.

The reader of this report should first get some general information about the GPIB-standard. Many documents that describes this standard is published. You can get the standard yourself from:

Standard Department
The Institute of Electrical and
Electronics Engineers
345, East 47th Street
New York, N.Y. 10017

1.1 IMPLEMENTATION OF THE IEEE-488 INTERFACE FUNCTION ON ND-100

There are 10 interface functions implemented on ND-100. We are using the most advanced subsets of almost all the functions:

- SH1 - source handshake function for control of the DAV-line
- AH1 - acceptor handshake function for control of the NRFD- and NDAC-line
- T5 - talker function for sending device dependent data (include serial poll status)
- L3 - listener function for receiving device dependent data (included serial poll status)

These four interface functions are used for sending and receiving multiline message - like device dependent data, commands etc. The next 5 functions are used when ND-100 is not the controller-in-charge. When ND-100 is the controller-in-charge, the last interface function is used

- SR1 - service request function is implemented to give ND-100 the opportunity to ask asynchronously the controller-in-charge for service by activating the SRQ-line. The controller-in-charge have to conduct a polling to find the source of the request.
- RLO - remote local function is not implemented because ND-100 can always be in remote mode.
- PP1 - parallell poll function provides capability for ND-100 to uniquely identify itself if it requires service and the controller-in-charge is requesting a response. The controller-in-charge have periodically to conduct a parallell poll.

DC1 - device clear function is used to allow ND-100 to be initialized to a pre-defined state by the controller-in-charge.

DT1 - device trigger function is implemented to give the controller- in-charge the opportunity to start a predefined action in ND-100.

C1-4,5 - Controller function allows ND-100 to:

- send device addresses
- send universal commands
- send addressed commands
- be the system controller (send IFC and REN)
- receive the control from another controller
- pass the control to another controller
- take control synchronously of the bus without loosing data
- answer a service request (conduct a serial polling)
- conduct a parallell polling

If more than one device on the interface bus has a Controller interface function, then all but one of them should be in CIDS (controller idle state). The one not in CIDS are called the controller-in-charge. The devices with C-function can take turns as the controller-in-charge of the interface. Just one of the devices with C-function can exist in the SACS (system controller active state). It has to remain in this state, and only this system controller can send IFC and REN. In a system we can select ND-100 as the system controller with a switch on the card - SYC.

1.2 A GPIB-system with ND-100

The standard is defined with some limitations:

- max 15 devices on one bus
- max 20 meters total length
- max 500 000 bytes/s data rate.

To avoid this limitation we have made a remote option (see fig. 1). On the ND-100 GPIB-card, we translate the 16 signals in the GPIB to 19 pairs of signals in a differential bus. In an option box containing one card we translate this 19 pairs of signals back to the 16 GPIB signals.

This differential bus can be maximum 100 meters. In this way we can have 13 devices connected to the ND-100 and another 14 devices standing in another room connected to the option box. This makes 28 devices connected to each other and they do not have to stand in the same room.

We have to make two limitations in the use of this option:

- all devices acting as listeners at the same time have to be either remote or local.
- all the devices we want to use as controller-in-charge have to be local (within 20 meters of each other)

There will be a small reduction in the speed when exchanging data between remote and local devices

2 USING THE GPIB-SYSTEM

The GPIB-system is a standards interface system used for programmable instrumentation. It is intended for application like laboratory instrumentation in hospitals, research institutions, and automatic test systems. However, it can also be used to interface peripherals like plotters or printers if these are IEEE- or IEC-compatible.

There are many reasons for using the interface system when we have:

- Measurements requiring high reproducibility and accuracy.
- Measurements requiring simultaneously testing of input and output characteristics.
- Measurements requiring immediate further processing of the measured data to allow, say, decision-making.
- Measurement procedures which are extremely diversified or constantly recurring.
- Procedures measuring many parameters.
- Measurements with only a few interesting results.

2.1 A MULTI USER GPIB SYSTEM

One GPIB-module can be used of maximum 16 users. The microprocessor on the card take care of creating new users, deleting old users, reserving devices for a user, enabling for a new device that has been connected to the bus, deleting old devices etc. It also tests the commands from the user according to device list, device configuration, device status, reserved devices etc.

The microprocessor controls the interface bus system. This makes it possible to avoid too many interrupts to the ND-100-CPU, and also reduce the load on the ND-100-bus as much as possible. This means that ND-100 are free to be used in other timesharing jobs most of the time.

One of the users may be privileged. This user is responsible for the configuration of the whole system and of each device.

3 H A R D W A R E

The hardware structure is the structure commonly used for ND-100-input/output modules with DMA. The device dependent data are usually exchanged with DMA through a 4K x 16 bits buffer. The hardware translate to/from 16 bits words from/to 8 bits-bytes. The devices on the market today have extremely different data rate and therefore it is important to have a data buffering on the module. The microprocessor on this interface module gives a new dimension in use of I/O modules

7 IOX-addresses are used to communicate with GPIB-module, 4 output and 3 input.

DN+0:

- Read memory address (RMAR), the address in ND-100's RAM which will be used in the next DMA-transfer (24 bits). You have to read RMAR two times to get the 24 bits:
 - 1st time - the 16 least significant bits
 - 2nd time - the 8 most significant bits.

DN+1:

- Load memory address (LMAR), write the start address of a DMA-transfer (24 bits). You have to load LMAR two times to give the module 24 bits address:
 - 1st time - the 8 most significant bits
 - 2nd time - the 16 least significant bits

DN+2:

- Read data word (RDAT) are used for receiving data from the module when not using DAM (16 bits):
- parallel poll status:
 - Bit 0-7 parallel poll status
 - Bit 8-15 user no
- Serial poll status:
 - Bit 0-6 serial poll status
 - Bit 7-11 device no
 - Bit 12-15 user no
- error status:
 - Bit 0-7 error no
 - Bit 8-15 device no (if used)
- input device dependent data
 - 1st word is no of bytes to be received
 - The rest is device dependent data from a device
- User or system device list
 - 1st word is no of bytes to be received.

The rest is three and three words:

Word 1:

Bit 0-7 the configuration of this device device
 Bit 8-12 device no
 Bit 14 interrupt on SRQ temporary
 Bit 15 interrupt on SRQ permanent

Word 2:

Bit 0-7 device talker address
 Bit 8-15 device listen address

Word 3

Bit 0-15 is indicating which user(s) using this device (bit 3 set if user 3 is using this device)

- GPIB-controller test status:
 177400 if controller is in function, else error code.

DN+3:

- Load data word (LDAT) used when sending command and data to the microprocessor. These commands are buffered in the command-FIFO (64x8 bits). The ND-100 can just place the commands in this FIFO and the microprocessor on the module will fetch and execute them one after the other. This increase the flexibility of the system. A string of max. 64 commands (or data) may be loaded into the command-FIFO at one time. The handler will see to that no overflow situation will occur.

When the microprocessor has executed all the commands in the FIFO, it will give FIFO-interrupt and a new string of commands may be given. (See more detail in the software description). The commands are checked by the Z80 and an error interrupt is sent if an illegal command or parameter is found.

DN+4:

- Read status register (RSTAT), contain information for ND-100 (the reason for interrupt etc.).

Bit 0 - ready for transfer interrupt enable (INTRFT)
 Bit 1 - ERROR, DEV, SRQ, FIFO, or RQSA interrupt enable (INTEN)
 Bit 2 - interface active - (ACTIVE)
 Bit 3 - ready for transfer (RFT)
 Bit 4 - ERROR
 Bit 5 - device interrupt received (DEV)
 Bit 6 - service request received (SRQ)
 Bit 7 - request service answered (RQSA)
 Bit 8 - command-FIFO is ready (FIFO)
 Bit 9 and 10 - 00 - no device interrupt request
 01 - interface clear received (IFC)
 10 - groupe execute trigger received (GET)
 11 - device clear received (DCL)
 Bit 11 - odd number of bytes have been exchanged (ODD).
 Bit 12 - the last character in DMA-input is the EOS (EOS)
 Bit 13 - the number of bytes received is the expected (ALL)
 Bit 14 - system controller (SYC)
 Bit 15 - listen/talk only mode - ONLY

DN+5:

- Load control word (LCW), for control of the module (enabling of interrupt etc.). (The control of the module is mostly through LDAT).

Bit 0 - enable interrupt on ready for transfer (EIRFT)
 Bit 1 - enable interrupt on error (EIERR)
 Bit 2 - activate a DMA-transfer (START)
 Bit 3 - not used
 Bit 4 - programmable interfacemodule clear (PCL)
 bit 5 - enable interrupt on device interrupt received (EIDDEV)
 Bit 6 - enable interrupt on SRQ received (EISRQ)
 Bit 7 - enable interrupt on RQSA received (EIRQSA)
 Bit 8 - enable interrupt on command-FIFO ready (EIFIFO)
 Bit 9 and 10 - not used
 Bit 11 - DMA transfer read to memory (READ)
 bit 12-15 - not used

DN+7:

- Load word counter (LWC), the number of word to be exchanged in the next DMA-transfer (16 bits).

With the LCW register we can enable for 6 different reasons for interrupt:

- RFT-ready for transfer. This is activated:
 - a) a DMA-transfer is finished.
 - b) a data word is ready in the RDAT register.
 - c) a transfer between other devices is finished.
- ERROR. There are many reasons for errors. The microprocessor will place an error code in RDAT-register.
- DEV - device interrupt. The ND-100 has received an IFC, DEC or GET over the interface system.
- SRQ - service request. The ND-100 is the controller-in-charge and has received a request for service from a device
- RQSA - requested service answered. ND-100 has requested service from the controller-in-charge and are answered.
- FIFO. The command-FIFO are ready to take care of more commands.

We are using Z80-CPU, Z80-CTC as interrupt controller and timer, Am9517 DMA-controller and 6k x 8 bits PROM and 1k x 8 bits RAM. The program in PROM will be general for all the customers. (If a customer want to make his own special solution, he is free to do so, but it will not be supported by Norsk Data). Texas Instrument's TMS9914 are chosen as GPIB-controllers with SN75160/161/162 as tranceivers. The DMA pass through a data-FIFO (4k x 16 bits) and the hardware will split words to bytes in output and make words of bytes in input

3.1 MICROPROCESSOR SOFTWARE

The microprocessor on the module give us an opportunity to handle more on the module itself and give interrupt to ND-100 CPU only when necessary. The data exchanging between the microprocessor and user are through RDAT for input to ND-100-CPU and LDAT for output from ND-100-CPU.

A preliminary list of the commands which can be sent through the command-FIFO (LDAT). The codes are the octal (hexamal) values.

The general commands with parameters:

- DMAIN - 141(61) will initiated a DMA-input to ND-100. 1st par. is 8 least significant bits in byte counter, 2nd par. is 8 most bits. The rest of the parameters are device no. If more than ND-100 shall be listening all parameters with bit 7=0 are listen devices. The last paremeter should have bit 7=1 and is the talker device. (Note 1)
- DMAOUT - 142(62) will initiate a DMA-output from ND-100, 1st parameter is 8 least significant bits of byte counter. 2nd par is 8 most bits. The rest of the parameters are device no. The last parameter should have bit 7=1 indicating the last listener. (Note 1)
- TRANS - 143(63) will initiate a transfer of data between other devices. 1st par. is 8 least significant bits in byte counter, 2nd par. is 8 most. The rest of the parameter(s) are device no. All parameters with bit 7=0 are listen devices. The last parameter should have bit 7=1 and is the talker. (Note 1)
- UDDIN - 144(64) will initiate an input of unit depended data without DMA. 1st par. is number of bytes to be received. The rest of the parameters are device no. If more than ND-100 shall be listening all parameters with bit 7=0 are listen devices. The last paremeter should have bit 7=1 and is the talker device. (Note 1)
- UDDOUT - 145(65) will initiate an output of unit depended data wihtout DMA. 1st par. is numebr of bytes to be exchanged. The rest of the parameters are device no. The last parameter should have bit 7=1 indicating the last listener. (Note 1)
- DEVTRG - 146(66) will initiate a device trigger. The parameter(s) is(are) the device no. The last parameter should have bit 7=1.
- DEVCLR - 147(67) will initiate a device clear. The parameter(s) is(are) the device no. The last parameter should have bit 7=1.
- DEVLOC - 150(68) will initiate device go to local. The parameter(s) is (are) the device no. The last parameter should have bit 7=1

- PPCONF - 151(69) will initiate a parallel poll configure. The 1st par. is the device no. The 2nd par. is the configuration byte.
- TCONT - 152(6A) will pass control to a device. The parameter is device no.
- SPSTAT - 153(6B) will give the modul our ND-100 serial poll status. The parameter is this status.
- PPSTAT - 154(6C) will give the modul our ND-100 parallel poll status. The parameter is this status.
- RESERV - 155(6D) will reserve device(s) for the user. The parameter(s) is(are) the device no. The last parameter has bit 7=1.
- RELEASE - 156(6E) will release reserved devices from the user. The parameter(s) is(are) the device no. The last parameter has bit 7=1.
- SRQP - 157(6F) will enable interrupt on SRQ permanent until SRQD or release of the device. The parameter(s) is (are) the device no. The last parameter has bit 7=1.
- SRQT - 160(70) will enable interrupt on SRQ temporary until SRQD or release of the device or executed serial poll on the device. The parameter(s) is (are) the device no. The last parameter has bit 7=1.
- SRQD - 161(71) will diable interrupt on SRQ. The parameter(s) is (are) the device no. The last parameter has bit 7=1.

(Note 1: If a device have secondary address, the device no bit 6=1, and the secondary address byte is sent before next device no.)

The general commands without parameters.

- CREATE - 101(41) will create a new user of the module.
- EXSP - 102(42) the module will conduct a serial poll. The device (which has enabled for SRQ) with least device no. asking for service will be served first (see RDAT)
- EXPP - 103(43) the module will conduct a parallel poll (see RDAT)
- LUSDEV - 104(44) will give a list of devices reserved by an user. (see RDAT).
- DELETE - 105(45) will delete an user from the module
- LSYSDEV - 106(46) will give a list of the device in the GPIB system.

Privileged user special commands with parameters.

- DEVLIST - 41(21) will create a new device in the device list of the system. The 1st parameter is device no. 2nd is LADDR and 3rd is TADDR. The 4th is the configuration of the device. The 5th is EOS (End of string) character for this device.
- DELODEV - 42(22) will delete a device from the list of devices for the system. The parameter is device no.
- ADDR - 43(23) our ND-100 GPIB address - The parameter is 5 bits code.

Privileged user special commands without parameters:

- SRES - 1(1) Send a reset command and will reset the module and the interface bus.
- SIFR - 2(2) Send the IFC (interface clear) command.
- SREN - 3(3) will set the REN line (remote enable)
- SLOC - 4(4) will reset the REN line
- STON - 5(5) will set talk only mode
- SLON - 6(6) will set listen only mode
- RONLY - 7(7) will reset listen/talk only mode

Special test command:

- MTEST - 167(77) a command to test if Z80's enwirement is in function, (see RDATA)

4 S O F T W A R E

The GPIB-module is an input/output unit - with or without direct memory access (DMA). To help the user in using this I/O-module, the GPIB-system will be delivered with a GPIB-software packet consisting of a set of FORTRAN callable subroutines, and a GPIB-MONITOR. The packet will be general. Norsk Data will not make device dependent software.

The IEEE-488 specify the signal level, the interface functions etc., but there are no software specifications at all. (IEEE are working to standardize parts of the software too). On the market today, there are many different devices, so we have to give the ND-100 system a configuration word for every device we connect to the GPIB-bus. Information like this are placed in RAM on the GPIB-module. In this way the microprocessor on cards can check everything the user of the module wants to do, avoiding all this checking in the ND-100-CPU. To make the use of the system simple we have two types of users:

- Priviledged I/O user - just one
- Nonpriviledged I/O user - 15 for each module.

Only the priviledged I/O user is allowed to configure the GPIB system. The priviledged I/O user has to know a device to be connected to the system in detail when configuring it. The microprocessor on the card make a list of these devices. For each device the list contains listener address, talker address configuration of the device, status of the device, end of string character, who are the user(s). The microprocessor also make a list of the user of the system which contain the user's configuration of ND-100, the user's ND-100 status. Obviously the users of a ND-100 either

- all have to use ND-100 as a system controller (SYC=1)
- or all have to use ND-100 as a non-system controller (SYC=0).

This can be chosen with a switch on the card. Only the Priviledged user is allowed to send universal message such as IFC, GTL, LLO, REN etc. on the GPIB interface.

If ND-100 is the system controller, it almost always will be the controller-in-charge. When a user give up the control of the system no new user can use the module before the ND-100 has received the control again.

When an nonpriviledged I/O user want to run a program on the GPIB module, he has to:

1. Ask to be logged on as a user.
2. Reserve the device he want to use (two types of devices - single user or multi user). If the device is reserved by an other user (and single), the user will be given a message.
3. Now the program can run and all the operations we want to do are checked by the microprocessor on the module.
4. When an user doesn't need a device any more, it should be released.
5. When the program is finished the rest of the user devices should be released and the user logged off

The GPIB-module is, as mentioned, an I/O device. The contact between the programmer and the module connected to the GPIB devices are through:

- Sending device dependent data with or without DMA.
- Receiving device dependent data with or without DMA.
- Receiving interface status like parallell poll, serial poll, error status through the microprocessor.
- Sending special messages to the module like word counter, memory address and control word.

- Sending commands to the microprocessor on the module.
- Receiving status from the module like interrupt status, state status, memory address.

4.1 HOW TO START AND STOP THE GPIB-DRIVER

To start the GPIB system, logg on as user system, then enter the sintran service program and execute the command START-GPIB. Note that if devices has been configured before, enter the GPIB-MONITOR and execute the command RECONFIGURATE. To stop the GPIB driver execute the command STOP-GPIB in the sintran service program.

4.2 GPIB-LIBRARY

The routines will have one or more parameters:

* NAME *	* TYPE *	* EXPLANATION *
STAT	INTEGER	- when returning from most library routines, stat will have the value: 0 - no error error code - if wrong parameter etc.
MVERS	INTEGER	- Microprogram version .
CNTNO	INTEGER	- Controller number (0-3) *** PARAMETER IGNORED IN FIRST VERSION OF SOFTWARE
DEVAR	INTEGER	- Array containing the devicenum(s). In all routines where a device is to be addressed as a talker, the last device in the array will be the talker
NODEV	INTEGER	- number of devices in DEVAR
DEVNO	INTEGER	- device number
DEVNAM	STRING	- device name
BYTCO	INTEGER	- number of bytes to be exchanged
RBYTCO	INTEGER	- number of bytes read on input
DATAD	INTEGER	- integer array or integer containing data to be transfered
LADDR	INTEGER	- listen address
TADDR	INTEGER	- talker address
CONF	INTEGER	- integer containing configuration byte in right byte

USNO	INTEGER	- user number
SECAD	LOGICAL	- set to true if device has secondary address
TIMEOUT	LOGICAL	- set to true if 16ms timeout should be enabled
REMOTE	LOGICAL	- set to true if device is on remote bus
SERPO	LOGICAL	- set to true if serial poll is allowed
SIUSER	LOGICAL	- set to true if device is a singel user type
EOIAFT	LOGICAL	- set to true if device need EOI after last byte
EOIWIT	LOGICAL	- set to true if device need EOI with last byte
EOS	LOGICAL	- set to true if device need EOS
EOSI	LOGICAL	- set to true if EOS shall terminate transfer on input
EOSCH	INTEGER	- integer containing ASCII-value of EOS character for device in right byte
TIMEUNIT	INTEGER	- timeunit for SRQ routine 1=basic timeunit (20mS), 2=second, 3=minute, 4=hour
TIMENO	INTEGER	- number of timeunits

All parameters written in capital letters are returned by the library.

There are two kinds of routines:

For all users:

4.2.1 LOG ON AS NONPRIVILDEGED USER

GENT (STAT,MVERS,cntno,USNO).

Will log the user on the GPIB-system and return the user no.

4.2.2 LOG OFF GPIB CONTROLLER

GEXIT (STAT,cntno).

This will log user off the specified controller and release all instruments reserved by user.

4.2.3 WRITE ERROR MESSAGE AND STOP

GSERM (stat).

This will write a short error message on terminal and the program terminates. Stat is error code. If this routine is called in a RT program the error message is written in the format:

HH.MM.SS ERROR 50 IN rname AT adr; USER ERROR.

SUBERROR: stat

where rname is name of RT program. Note that "adr" is always pointing inside the GSERM routine and not to the address where the routine was called, also note that stat is output as a decimal number.

4.2.4 WRITE ERROR MESSAGE AND CONTINUE

GCERM (stat).

This routine will do the same as the previous routine except that it will return to the calling program.

4.2.5 READ DATA FROM DEVICE(S)

GREAD (STAT,cntno,nodev,devar,bytco,datad,RBYTCO).

This will read maximum no bytes from last device in DEVAR and place the data in DATAD. There can be other device(s) listening to the transfer (Note 2)

4.2.6 SEND DATA TO DEVICE(S)

GWRIT (STAT,cntno,nodev,devar,bytco,datad).

This will write no bytes to the device(s) in DEVAR (Note 2)

4.2.7 TRANSFER DATA BETWEEN DEVICES

GTRNS (STAT,cntno,nodev,devar,bytco).

This will initiate a data transfer of no bytes from last device in DEVAR to the rest of the devices in DEVAR (Note 2)

4.2.8 TRIGGER DEVICE(S)

GTRG (STAT,cntno,nodev,devar).

This will send a device trigger to the device(s) in DEVAR

4.2.9 ENABLE SRQ INTERRUPT FROM DEVICE(S)

GPESR (STAT,cntno,nodev,devar).

This routine will permanently enable the service request interrupt on the device(s) specified. That means that when a device sends a SRQ to the controller the device is polled, and the pollstatus send to the user. The interrupt is not disabled until the GDISR routine is called or the device is released.

GTESR(STAT,cntno,nodev,devar).

This routine will temporary enable the service request interrupt on the device(s) specified. The interrupt is disabled when the first SRQ has been received or the GDISR routine is called or the device is released.

4.2.10 DISABLE SRQ INTERRUPTGISR (STAT,cntno,nodev,devar).

This routine will disable SRQ interrupt from the device(s) in DEVAR

4.2.11 CHECK/WAIT FOR SRQ-INTERRUPTGCSRQ (STAT,CNTNO,DEVNO)

This routine will check if a service request interrupt has been received since last time the routine was called. If no SRQ-interrupt was received ,STAT will be zero. If a SRQ-interrupt was received STAT will contain the pollstatus, CNTNO will contain the controller no. and DEVNO will contain the device number of the device that asked for service. NOTE THAT THE VARIABLE "CNTNO" IN THIS ROUTINE IS A INPUT PARAMETER TO YOUR PROGRAM.

GWSRQ (STAT,CNTNO,DEVNO,TIMEUNIT,TIMENO)

This routine will check if a service request interrupt has been received since last time this or the previous routine was called, and if no SRQ interrupt was received wait a specified time for a SRQ interrupt. The time it will wait is specified with the parameter TIMEUNIT, where 1 is basic timeunit (20MS), 2 is second, 3 is minute, 4 is hour, and the parameter TIMENO wich is the number of timeunits.

4.2.12 CLEAR DEVICE(S)GCLR (STAT,cntno,nodev,devar).

This will send a device clear to the device(s)

4.2.13 SET DEVICE(S) IN LOCALGLOCD (STAT,cntno,nodev,devar).

This will set the devices in DEVAR in local stat

4.2.14 RESERVE DEVICE(S)GRES (STAT,cntno,nodev,devar).

This routine will reserve device(s) for the user

4.2.15 RELEASE DEVICE(S)GREL (STAT,cntno,nodev,devar).

This routine will release device(s) from the user

4.2.16 LIST THE USERS DEVICE(S)GUSED (STAT,cntno,RBYTCO,datad).

This will return the device-list of the devices reserved by the user,RBYTCO will indicate how many bytes of valid data was returned

4.2.17 LIST THE SYSTEMS DEVICE(S)GSYSD (STAT,cntno,RBYTCO,datad).

The same as GUSED except that all the devices of the system is listed

4.2.18 GET DEVICE NUMBERGGTNO (STAT,cntno,devnam,DEVNO).

Will return the deviceno. of the device with name=DEVNAM

For the privileged I/O-user.

4.2.19 LOG ON AS PRIVILEGED USERGPENT (STAT,MVERS,cntno)

This routine will log the user on as privileged user.

4.2.20 CONFIGURE DEVICEGCDEV(STAT,cntno,DEVNO,devnam,laddr,taddr,secad,timeout,remote,serpo
,siuser,eoiaft,eoiwit,eos,eosi,eosch).

This routine will configure a new device in the system. dev will be given the listen address laddr, talker address taddr, and the configuration according to the parameters in the call.

4.2.21 DELETE DEVICEGDEL (STAT,cntno,devno).

This routine will delete a device from configuration

4.2.22 SEND PARALLEL POLL CONFIGURATION BYTE TO DEVICEGPPCD (STAT,cntno,devno,conf).

The routine will send conf as parallel poll configuration byte to dev (Note 3)

4.2.23 PASS CONTROL TO DEVICEGCONT (STAT,cntno,devno).

The routine will pass control to dev (Note 3)

4.2.24 SET ND-100 PARALLELL POLL STATUSGPPS (STAT,cntno,conf).

Will configurate ND-100's parallell poll status conf (Note 3)

4.2.25 SET ND-100 SERIAL POLL STATUSGSDPS (STAT,cntno,conf).

Will configurate ND-100's serial poll status conf (Note 3)

4.2.26 SET ND-100 TALKER ADDRESSGDDEV (STAT,cntno,taddr).

This routine will give ND-100 the talker address taddr and the listen address will be the corresponding (bit 7=0, bit 6=1)

4.2.27 SEND INTERFACE CLEARGIFC (STAT,cntno).

This routine will send interface clear on the GPIB.

4.2.28 SET LISTEN ONLY MODEGSLOM (STAT,cntno).

Set listen only mode (Note 3)

4.2.29 SET TALKER ONLY MODEGSTOM (STAT,cntno).

Set talker only mode (Note 3)

4.2.30 RESET ONLY MODEGROM (STAT,cntno).

Reset only mode (Note 3)

4.2.31 CLEAR CONTROLLERGCLR (STAT,cntno).

Will send device clear to the GPIB-controller

NOTES

- 1) If a device have secondary address, the secodnary address must be given before the next device number
- 2) Will be implemented later

4.3 GPIB MONITOR

The GPIB MONITOR is a subsystem where the different calls to the GPIB library is implemented as commands. This will make the configuration and test of new instruments easier.

The command and the parameters can be entered on the same line, but the monitor will prompt the user if any parameters are missing. If a "a" is entered as a parameter in a command the monitor will ignore the command, and ask for next command.

When the monitor is started controller number 0 is set as default controller, and the routines will use this number without asking for a controller number. An other controller may be set as default with the command "SET-DEFAULT-CONTROLLER", or by using the command "RESET-DEFAULT-CONTROLLER" the monitor will ask for the controller number on all commands where it is needed.

COMMANDS

4.3.1 LOG-ON

This command will log you on as a user to a GPIB-CONTROLLER. If this command is executed from user SYSTEM, it will ask if the user want to be priviledged. If yes, it will try to log on as user 0. If the command is executed from any other user it will be the same as for user SYSTEM except that it will try to log on as nonpriviledged user. If the controller is not active a errormessage is written.

PARAMETERS

1) CONTROLLER NUMBER

A number from 0 to 15, if a default controller is set this will be used.

2) PRIVILDGED USER ? (Only if user system)

Y or N

4.3.2 LOG-OFF

This command will log you off a GPIB-CONTROLLER.

PARAMETERS

1) CONTROLLER NUMBER

4.3.3 SET-DEFAULT-CONTROLLER

This command will set the default controller number. This number will be used as controller number instead of asking user for controller number.

PARAMETERS

- 1) CONTROLLER NUMBER

4.3.4 RESET-DEFAULT-CONTROLLER

This command will reset the default controller number. That means that the monitor will always ask user for controller number where needed.

PARAMETERS

NONE

4.3.5 CONFIGURE-DEVICE

This command will configure a new device on the system. The configuration of the device will be written on the GPIB-CONFIGURATION file, and to the controller.

PARAMETERS

- 1) CONTROLLER NUMBER

- 2) DEVICE NAME

A string of max. 10 (dec) characters. Since the name has to be spelled out in all commands in the monitor, it is recommended to have the name as short as possible

- 3) LISTEN ADDRESS

Device listen address. The complete address has to be specified as an octal number. The listen address has to be in the range 40 to 76 (77 is the GPIB UNL command)

- 4) TALKER ADDRESS

Device talker address. The address has to be in the range 100 to 136 (137 is the GPIB UNT command)

- 5) DOES DEVICE HAVE SECONDARY ADDRESS

If the answer is yes, the secondary address always has to follow the device number when we address the device

6) TIMEOUT ENABLED

If a 16ms timeout should be enabled. If the device never uses more than 16ms to send or receive the first byte after it has been addressed the answer should be Y.

7) LOCAL OR REMOTE

The answer is L if the device is connected to the local bus, or R if the device is connected to the remote bus

8) SERIAL POLL ALLOWED

If the device has the service request (and serial poll) option (SR1) the answer should be yes. If the answer is no the controller will not poll the device if a SRQ is received and this may jam the controller

9) SINGLE OR MULTI USER

If the device can be reserved by several users simultaneously the answer is M, if only one user can reserve the device at a time the answer is S

10) EOI WITH, AFTER OR NONE EOI

If the controller should set the EOI line "TRUE" with or after the last byte in a transfer, or not use the line at all

11) DOES THE DEVICE NEED EOS CHARACTER

If E is specified the controller will add the specified end of string character at the end of each transfer. If E is specified the monitor will ask if the controller should terminate a input transfer if an EOS character is detected, and also what the octal value of the EOS character is (i.e LF=12)

4.3.6 DELETE-DEVICE

This command will delete the device from the controller device list, and also remove the device from the configuration file.

PARAMETERS

1) CONTROLLER NUMBER

2) DEVICE NAME

4.3.7 RECONFIGURATE

This command will read the file GPIB-CONFIG-XX ,wich contains the information about all the devices that has been configured for this controller, and send the information to the GPIB controller.

NOTE 1. No other users can be logged on when this command is executed since the user tables on the controller is set to zero.

NOTE 2. Command has to be spelled completely out.

PARAMETERS

- 1) CONTROLLER NUMBER

4.3.8 CONTROLLER-CLEAR

This command will clear the GPIB controller, but all user and device information is not changed.

PARAMETERS

- 1) CONTROLLER NUMBER

4.3.9 SET-CONTROLLER-ADDRESS

This command will change the GPIB controllers talk and listen address. Note that when the controller is powered up the listen address is set to 65 (octal) and the talk address is set to 125 (octal).

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) CONTROLLER ADDRESS

Give the talker address you want the controller to have as an octal number. The controller will set the coresponding listen address. I.e. if 115 is given as talker address the listen address will be 55

4.3.10 RESERVE-DEVICE

This command will reserve the device(s) specified.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

4.3.11 RELEASE-DEVICE

This command will release the device(s) specified

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

4.3.12 TRIGGER-DEVICES

This command will send a trigger command to the device(s) specified.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

4.3.13 CLEAR-DEVICES

This command will send a clear command to the device(s) specified.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

4.3.14 LOCAL-DEVICES

This command will set the device(s) specified in local mode.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

4.3.15 INTERFACE-CLEAR

This command will pulse the IFC line on the bus.

PARAMETERS

- 1) CONTROLLER NUMBER

4.3.16 SEND-ASCII-STRING

This command will send a ASCII string to the device(s) specified.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)
- 3) ASCII STRING

String of ASCII characters to be sendt to device(s). If all parameters are to be entered on the same line the string has to be preceded by a "Q"

EXAMPLE : SE-ASC HP1610 QMN,5

This will send the string "MN,5" to the device with the name "HP1610". Note that the command string max. can contain 24 characters (DEC). That means that the command + the device name(s) + the ASCII string is limited to 24 characters. If more characters are needed in the string, give CR after the device name(s) have been entered, and the monitor will respond with "ENTER STRING : Now a string of max. 80 characters may be entered.

4.3.17 SEND-DATA-BUFFER

This command will send the data in the data buffer to the device(s) specified.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)
- 3) NUMBER OF BYTES

The number of bytes to be transfered.

4.3.18 RECEIV-DATA

This command will address the device as talker. The data will be placed in the monitors data buffer, and may be inspected with the command "DUMP-DATA-BUFFER"

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

Name of the device. If more than one device is specified the last device entered will be addressed as talker, and the rest of the devices as listners

- 3) NUMBER OF BYTES

Max. number of bytes to be received. If the EOI line is set "TRUE" or a EOS character is detected (if device is configured for EOS detection on input) before the max. number of bytes has been transfered, the transfer is terminated, and the monitor will print the number of bytes received.

4.3.19 DUMP-DATABUFFER

This command will dump the databuffer on the terminal. The print is divided in tree parts, the left part is the databuffer dumped as 16 bit words (octal), the center part is the databuffer dumped as bytes, and the right part is the buffer dumped as ASCII characters. All nonprintable characters is replaced with a period.

PARAMETERS

1) FIRST BYTE

The first byte of the buffer one want to dump. Note that the monitor will allways dump complete lines contaning 8 bytes , and it will start the dump with the line containing the specified byte.

2) NUMBER OF BYTES

The number of bytes to be dumped. Since the monitor will dump complete lines it will dump the number of lines needed to display the first byte, plus the number of bytes specified.

4.3.20 SAVE-DATABUFFER

This command will save the databuffer to a file.

PARAMETERS

1) FILE NAME

The file name of the save file. If a non-existing file is specified the file will be created.

4.3.21 LOAD-DATABUFFER

This command will save the databuffer to a file.

PARAMETERS

1) FILE NAME

The file name of the load file.

4.3.22 DATA-BUFFER-CLEAR

This command will zero the monitors databuffer.

PARAMETERS

NONE

4.3.23 LIST-SYSTEM-DEVICES

This command will list all the devices that has been configured.

PARAMETERS

1) CONTROLLER NUMBER

2) OUTPUT FILE

If no output file is spcified the list is printed on the terminal.

4.3.24 LIST-USER-DEVICES

This command will produce a simular list as the previous command, but only devices reserved by the user is listed.

PARAMETERS

1) CONTROLLER NUMBER

2) OUTPUT FILE

Se previous command.

4.3.25 HELP

This command will list all commands available in the monitor.

PARAMETERS

1) COMMAND

by giving a string as a parameter, the monitor will list all commands that will give a match in the command table. If no parameter is given all commands are listed.

4.3.26 GET-ERRORMESSAGE

This command will print the errormessage corresponding to the errorcode given.

PARAMETERS

1) ERRORCODE

The errorcode is a octal number between 1 and 477

4.3.27 GET-FUNCTION

This command will print the driver function corresponding to the functioncode given

PARAMETERS

1) FUNCTIONCODE

The function code is a octal number between 0 and 16

4.3.28 PERMANENT-SRQ-INTERRUPT-ENABLE

This command will permanently enable the service request interrupt on the device(s) specified. That means that when a device sends a SRQ to the controller the device is polled, and the pollstatus send to the user. The interrupt is not disabled until the SRQ-INTERRUPT-DISABLE command is executed or the device is released.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

4.3.29 TEMPORARY-SRQ-INTERRUPT-ENABLE

This command will temporary enable the service request interrupt on the device(s) specified. The interrupt is disabled when the first SRQ has been received or the SRQ-INTERRUPT-DISABLE command is executed or the device is released.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

4.3.30 SRQ-INTERRUPT-DISABLE

This command will disable the service request interrupt on the device(s) specified.

PARAMETERS

- 1) CONTROLLER NUMBER
- 2) DEVICE NAME(S)

4.3.31 CHECK-FOR-SRQ-INTERRUPT

This command will check if a service request interrupt has been received since last time the command was executed.

PARAMETERS

NONE

4.3.32 WAIT-FOR-SRQ-INTERRUPT

This command will check if a service request interrupt has been received since last time this or the previous command was executed, and if no SRQ interrupt was received wait a specified time for a SRQ interrupt. The time is default 10 sec. , but can be changed with the command CHANGE-SRQ-WAIT-TIME.

PARAMETERS

NONE

4.3.33 CHANGE-SRQ-WAIT-TIME

This command will change the time the monitor will wait for a SRQ interrupt. The current value is printed out.

PARAMETERS

1) TIME UNIT

Type of timeunit to use. Enter B for basic time unit, S for seconds, M for minutes, or H for hours. If a CR is given the current value is kept.

2) NUMBER OF TIMEUNITS

Enter the number of timeunits to wait.

4.3.34 EXIT

This command will exit from the monitor, and all devices reserved will be released.

PARAMETERS

NONE

A P P E N D I X A

MONITOR COMMANDS

** COMMAND **	** PARAMETER(S) **
LOG-ON	<CONTROLLERNO.>(,PRIVILEDGED USER)
LOG-OFF	<CONTROLLERNO.>
SET-DEFAULT-CONTROLLER	CONTROLLERNO.
RESET-DEFAULT-CONTROLLER	
CONFIGURE-DEVICE	<CONTROLLERNO.>,DEVNAME,LADDR,TADDR, SEC. ADDR.,TIMEOUT,LOCAL/REMOTE, SERIAL POLL,SINGEL/MULTI USER,EOI,EOS
DELETE-DEVICE	<CONTROLLERNO.>;DEVICE
RESERVE-DEVICE	<CONTROLLERNO.>,DEVICE(S)
RELEASE-DEVICE	<CONTROLLERNO.>,DEVICE(S)
LIST-SYSTEM-DEVICES	<CONTROLLERNO.>,(OUTPUT-FILE)
LIST-USER-DEVICES	<CONTROLLERNO.>,(OUTPUT-FILE)
RECONFIGURATE	<CONTROLLERNO.>
SEND-ASCII-STRING	<CONTROLLERNO.>,DEVICE(S),STRING
HELP	
EXIT	
TRIGGER-DEVICES	<CONTROLLERNO.>,DEVICE(S)
CLEAR-DEVICES	<CONTROLLERNO.>,DEVICE(S)
LOCAL-DEVICES	<CONTROLLERNO.>,DEVICE(S)
PERMANENT-SRQ-INTERRUPT-ENABLE	<CONTROLLERNO.>,DEVICE(S)
TEMPORARY-SRQ-INTERRUPT-ENABLE	<CONTROLLERNO.>,DEVICE(S)
SRQ-INTERRUPT-DISABLE	<CONTROLLERNO.>,DEVICE(S)
GET-ERROR-MESSAGE	ERRORCODE
CHECK-FOR-SRQ-INTERRUPT	
WAIT-FOR-SRQ-INTERRUPT	
RECEIV-DATA	<CONTROLLERNO.>,DEVICE(S),NO. OF BYTES
DUMP-INPUT-DATABUFFER	FIRST WORD,NO. OF WORDS
DATA-BUFFER-CLEAR	
CONTROLLER-CLEAR	<CONTROLLERNO.>
INTERFACE-CLEAR	<CONTROLLERNO.>
CHANGE-SRQ-WAIT-TIME	TIMEUNIT,NO. OF TIMEUNITS
SET-CONTROLLER-ADDRESS	<CONTROLLERNO.>,TALKER ADDRESS
SEND-DATA-BUFFER	<CONTROLLERNO.>,DEVICE(S),NO. OF BYTES
GET-FUNCTION-CODE	FUNCTION-CODE

NOTE : CONTROLLERNO. IS ONLY NEEDED IF NO DEFAULT CONTROLLER IS SET

