Programmer Manual

Tektronix

VM700T Video Measurement Set
Option 48 GPIB Interface

070-9657-00

Programmer Manual

Tektronix

VM700T Video Measurement Set
Option 48 GPIB Interface

070-9657-00

Printed in the U.S.A.

Tektronix, Inc., P.O. Box 1000, Wilsonville, OR 97070–1000

WARRANTY

Tektronix warrants that this product will be free from defects in materials and workmanship for a period of one (1) year from the date of shipment. If any such product proves defective during this warranty period, Tektronix, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. Customer shall be responsible for packaging and shipping the defective product to the service center designated by Tektronix, with shipping charges prepaid. Tektronix shall pay for the return of the product to Customer if the shipment is to a location within the country in which the Tektronix service center is located. Customer shall be responsible for paying all shipping charges, duties, taxes, and any other charges for products returned to any other locations.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Tektronix shall not be obligated to furnish service under this warranty a) to repair damage resulting from attempts by personnel other than Tektronix representatives to install, repair or service the product; b) to repair damage resulting from improper use or connection to incompatible equipment; c) to repair any damage or malfunction caused by the use of non-Tektronix supplies; or d) to service a product that has been modified or integrated with other products when the effect of such modification or integration increases the time or difficulty of servicing the product.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THIS PRODUCT IN LIEU OF ANY OTHER WARRANTIES, EXPRESSED OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

List of Figures

List of Tables

Preface

This manual describes the VM700T Video Measurement Set Option 48 programming capabilities for the GPIB interface. The manual is organized as follows:

Getting Started provides the information needed to set up and use the GPIB interface option of the VM700T. Additional information on remote control of the VM700T and programming is found in the *VM700T RS-232 Interface Programmer Manual*. That manual contains the keywords lists for all of the options and samples of the get results files that are sent by the VM700T Video Measurement Set when commanded to do so. That manual also describes how to use the Function Key feature.

GPIB Commands describes the VM700T Video Measurement Set GPIB commands and gives examples of how they may be used to control the operation of the VM700T and work in its internal file system through the GPIB interface option.

Status Events provides a list of the GPIB event codes and GPIB status bytes.

VM700T Programming Procedures provides a general outline of the programming process for the GPIB interface.

Appendix A: Specification provides a listing of the GPIB functions implemented in the Option 48 GPIB Interface hardware, and gives the electrical specifications for the module.

Appendix B: GPIB Diagnostics provides the operation information for running the diagnostic procedures added by Option 48. A list of the GPIB diagnostic tests and the subtest within each test is given for reference.

Getting Started

Getting Started

This section provides the information needed to set up and use the GPIB interface option of the VM700T. Additional information on remote control of the VM700T and programming is found in the *VM700T RS-232 Interface Programmer Manual*. That manual contains the keywords lists for all of the options and samples of the get results files that are sent by the VM700T Video Measurement Set when commanded to do so. That manual also describes how to use the Function Key feature.

Standards

The VM700T GPIB option conforms to the following standard:

- IEEE 488.1 Standard (See Table A-1 for list of interface functions supported.)

The following exceptions or modifications to the standard have been made:

- Added <definite length arbitrary data> data type.

  This format is useful for transmitting either binary or text data. The format for this data type is:

  ```
  #<header length digit><data length digits><8-bit data>...
  ```

  Example:

  The string “abc” is represented by either of the following:

  #6000003abc

  #13abc

- Talk terminator: LF(EOI).
- Listen terminator: LF, LF(EOI), or (EOI).
- SET?, and TEST? not implemented.
- Every received command outputs a result (either data, or 0xFF if no data).

To keep the command syntax similar to the existing VM700T remote commands:

- Arguments are separated either by a ‘,’ (comma) or a ‘ ’ (space).
- Arguments are entered as either CHAR data type or STRING data type. If the text contains an embedded ‘;’ (semicolon), ‘,’ (comma), or ‘LF’, use STRING data type.

Setup

The setup procedure is given in the following steps:

1. Verify that the GPIB port on the PC or other controller and the GPIB port on the VM700T are connected via a standard GPIB cable.
2. Verify that the PC has the appropriate GPIB controller hardware and software drivers installed. (Refer to the appropriate controller manuals for setup of the GPIB controller hardware and system file changes needed.)
3. Check the settings contained in the Communication Setup file, shown in Figure 1–1, in the /nvram0/ConfigFiles directory. To do so, do the following:
   a. Display the Communication Setup file on the VM700T screen (press the Configure button, touch the Configure Files soft key, touch the icon for the Communication Setup file).
   b. Set the Remote Control Port to GPIB and touch Accept Input. If you want to send reports and copy to the GPIB bus, set those ports to GPIB also.

***NOTE**. The GPIB address is factory set to address 1. You can change the VM700T GPIB card to any address in the range of 0–30, offline, and talk only. Offline takes the GPIB card off the GPIB bus. Talk Only enables the VM700T to output directly to a GPIB printer without an intervening controller.*

   c. Set the GPIB device address to the system address you will use to communicate with the VM700T. (This selection is the last line in the Communications Setup file.) Touch Accept Input and Update & Exit to complete the file edit.
4. With GPIB selected as the Remote Control port, the VM700T powers up ready to begin GPIB communication. It does not go into remote mode until addressed. After being addressed, the VM700T will remain in remote mode until a Go to Local command is sent to the GPIB board. An SRQ is asserted on the GPIB bus, and a power-up status byte (65) returns from the VM700T at the first serial poll.

***NOTE**. The power-up status byte will also be returned after doing the GPIB diagnostics. After the diagnostics are finished and exited, the GPIB board is reinitialized.*

```
Communication setup

Copy
Port:     GPIB
Format:   Epson LQ

Report
Port:     GPIB
Format:   Epson LQ

Log
Port:     None
Format:   Epson LQ

Control Port:   Serial Port 1

Remote Control
Port:     GPIB
Prompt:   VM700>
Message Display: Remote
Non-SLIP Interfacing Mode: Computer

Port 0
Protocol:         None
Baud Rate:        19200
Flow Control:     CTS/RTS
Character Size:   8
Parity:           None
Reset Character:  None
Carrier Detect:   disabled

Port 1
Protocol:         None
Baud Rate:        9600
Flow Control:     CTS/RTS
Character Size:   8
Parity:           None
Reset Character:  None
Carrier Detect:   disabled

GPIB Device Address: 1
```

Figure 1–1: Communication setup file

Operation

This section describes the various operating modes of the GPIB interface and shows the meaning of the indicator lights on the GPIB module.

Power-up

At power-up, the VM700T GPIB interface card is not active on the GPIB bus until it receives a GPIB device address message from the VM700T. The device address message is sent to the GPIB card by the VM700T at power-up if GPIB is selected as the Remote Port. Otherwise, it is sent when the user goes into the Communication Setup file and selects GPIB for the Remote Port.

When the GPIB card receives the VM700T GPIB active message, the GPIB hardware is initialized using the GPIB Device Address in the Communication Setup file.

The VM700T will remain in local mode until the GPIB interface card is addressed with REN (remote enable) asserted. If your controller automatically asserts REN, then the VM700T goes into remote mode when it is addressed. If remote enable is not asserted automatically, a specific remote enable must be asserted to get the VM700T ready to accept commands from the GPIB interface card.

***NOTE**. Talk Only mode should not be selected when the VM700T is used with an external GPIB controller.*

Talk Only Mode

Talk Only mode is selectable in the VM700T configure file as an address choice. In talk only mode, the VM700T may be connected directly to a GPIB printer for hardcopy output without using an external GPIB controller. The VM700T takes control of the bus to output text or graphics to the attached printer. To use this mode for hardcopy output, the desired output port or ports (Copy, Log, or Report) in the Communication setup file must be set to GPIB. The Format choice must be for the attached GPIB printer. You must select a graphics format to output graphics displays.

***NOTE**. Hardcopy is fully supported with the exception of Picture mode. Picture mode hardcopy will not print to a GPIB printer. A printer alarm (two beeps of the internal alarm) is sounded if the user requests a hardcopy in Picture mode.*

GPIB Remote Mode

When GPIB is selected as the Remote Port, the VM700T GPIB interface card is ready to go into GPIB remote mode and accepts GPIB, Remote, or File Transfer type commands. As soon as the GPIB interface card is addressed, the VM700T goes into remote mode. VM700T commands will be passed to the VM700T after entering or returning to remote mode.

GPIB Only Mode

When GPIB is deselected as the Remote Port in the Communication Setup file, the GPIB card goes into GPIB only mode, and the GPIB card responds only to GPIB-type commands. The is the same as having the VM700T in local mode except the VM700T cannot return to remote mode.

Terminating GPIB Remote Control

Sending a GPIB Go to Local command to the GPIB interface card places the VM700T in local mode, and the Configure button LED stops blinking. In this mode, the GPIB card executes GPIB-type commands and goes back into remote mode when addressed.

While the VM700T is in remote mode, the LED indicator in the Configure button of the VM700T is blinking. To permanently exit GPIB remote mode operation, the Remote Control selection must be changed in the Communications Setup file.

1. Press the Configure button on the front panel of the VM700T to return to local control; then press it again to bring up the Configure menu.
2. Touch the Configure Files soft key to display the choices.
3. Enter the Communication Setup file by touching that soft key.
4. Select the Remote port configuration for editing and choose a selection other than GPIB.
5. Touch Accept Input and Update & Exit. The VM700T is now out of GPIB mode and local control remains in effect.

Difference Between RS-232 Remote and GPIB Remote Operation

The first difference is the omission of several serial remote commands, because they are not used in GPIB. These commands are the following:

```
remote

terminal

computer

quit

exit
```

NOTE. *The talker/listener program you use to handle communications between the controller and the VM700T dictates the way commands and responses are handled in the controller. It also determines the prompt seen on the controller. The prompt line in the CommunicationSetup file is not used as a prompt to the controller as it is with serial communications.*

The next difference is the command syntax. GPIB syntax requires commas as argument delimiters, and the character data type is very restrictive. The VM700T GPIB implementation accepts GPIB syntax, and it also accepts VM700T syntax where possible. The exceptions being that VM700T arguments cannot include a ‘,’ (comma), ‘;’ (semicolon), or ‘LF’.

For example, the filemode command to move a file could be entered as any of these:

```
mv filename1 filename2

mv filename1,filename2

mv "filename1","filename2"
```

A third difference is that GPIB commands may be strung together on a single command line separated by semicolon command delimiters as opposed to a single command per line with serial remote control.

Example:

```
execute Waveform;execute Vector;execute H_Timing;getresults;
show filename
```

Each command in the command line returns a reply of at least 0xFF separated by a ; (semicolon) delimiter.

NOTE. *The VM700T GPIB board does not parse the command line to determine if all the commands are correct and executable prior to starting execution of the commands. This may result in several commands being executed prior to finding a command that is incorrect or cannot be executed. The VM700T remains in the state determined by the last command executed.*

GPIB Indicator LED

Seven LED indicators are mounted on the GPIB board. They are visible at the rear of the VM700T when the board is installed as shown in Figure 1–2. Table 1–1 defines the lamp, the lamp label, the lamp color and the meaning of the lamp being lighted.

Table 1–1: Status Indicators

Circuit Number	Label	Color	Meaning
DS103	LSTN	Green	Addressed to listen (MLA) when on
DS104	TALK	Green	Addressed to talk (MTA) when on (The LED is always on in talk-only mode.)
DS101	SRQ	Yellow	Service request (SRQ) pending when on
DS105	REM	Yellow	Remote control active when on
DS102	LOCK	Red	Front Panel lockout
DS106	CACT	Green	Controller active state when on
DS107	SCTL	Green	System control (never on)

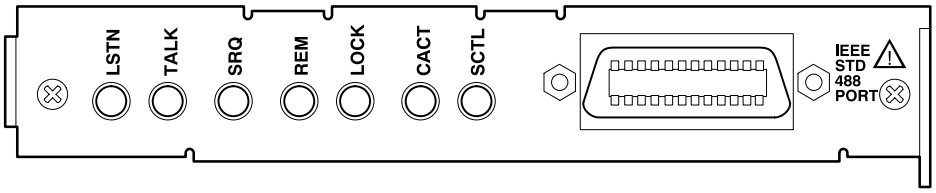

Figure 1–2: GPIB board rear panel

Command Set Summary

This section describes the interface messages and summarizes the VM700T remote control and file transfer commands. Full descriptions of the commands are given in *GPIB Commands* starting on page 2–1.

VM700T GPIB Interface Messages

The following interface messages are implemented.

- `IFC` (Interface Clear)

  This message returns control to the controller by untalking, unlistening, and disabling serial poll in all devices on the bus.

- `DCL` (Device Clear)
- `SDC` (Selected Device Clear)

  These messages are used to restart device communications and have the following actions:

  - Clears any SRQ (except Power On) and all events (except Power On).
  - Clears input and output buffers.
  - Restarts GPIB parser.
  - Will not affect any current settings of the VM700T.

- `rqs on` (turns on RQS)
- `rqs off` (turns off RQS)
- `rqs?` (query state of RQS)

NOTE. *The standard GPIB* `set?` *command was omitted because all settings in VM700T cannot be queried). The standard GPIB* `test?` *command is also omitted from the command set.*

VM700T Remote Commands

Table 1–2 summarizes the three types of GPIB remote control commands.

File Transfer Mode Commands

Sending the VM700T a filemode command puts the VM700T in the File Transfer mode. File Transfer mode enables file transferring to and from the VM700T over the GPIB bus. It also allows creation and deletion of directories.

Users may save and recall VM700T set information such as measurement limits and function definitions using the File Transfer mode. Those files can then be edited and sent back to the VM700T.

Table 1–2: GPIB Command Type

Type	Meaning
remote	Identical to VM700T serial port remote commands
filemode	New commands to support File Transfer Mode
GPIB	GPIB only command

NOTE. *Files edited using the PC editor add a carriage return to the end of each line. When sending these files back to the VM700T, your program must remove the carriage returns from the file.*

Terminating File Transfer Mode. If a remote command (for example, execute Waveform) is sent to the VM700T, File Transfer mode is halted, and the VM700T goes into Display (Waveform) mode. An event is generated if a File Transfer command is given and the VM700T is not in File Transfer mode.

Password. The **passwd** File Transfer command is needed to access the file system if VM700T password file protection is in effect. This command is required each time File Transfer mode is started. Refer to the *VM700T Option 01 (NTSC) and Option 11 (PAL) User Manual* for more information on the password function.

File Transfer Commands. The Filemode transfer commands include the following:

`filemode`	starts filemode operation
`cd`	changes to the designated directory
`dir`	lists the byte size and filenames in the designated directory
`ls`	lists the available directories and files
`pwd`	shows the current working directory
`mkdir`	creates a new directory
`mv`	moves a file name to a new file name
`passwd`	must be used when password mode is in effect
`readfile`	reads a named file
`rm`	deletes a named file
`rmdir`	deletes a named directory
`writefile`	writes a named file to the VM700T

These commands are also listed in Table 1–3 under filemode and are fully described in the *GPIB Commands* section beginning on page 2–1.

GPIB Remote Commands

Table 1–3 is a list of the GPIB Remote Commands.

Table 1–3: VM700T GPIB Remote Commands

Command	Type	Purpose
cancelcopy	remote	clears print spooler
control	remote	sends characters out control port
controlbreak	remote	sends break sequence out control port
event?	GPIB	returns a GPIB event number only
evmsg?	GPIB	return a GPIB and VM700T event number
execute	remote	executes a VM700T application
filemode	remote	starts the GPIB File Transfer Mode
cd	filemode	changes VM700A directory
dir	filemode	lists the byte size and filenames in the designated directory
ls	filemode	lists VM700T directory contents
mkdir	filemode	makes a new VM700T directory
mv	filemode	moves a VM700T file (also rename a file)
passwd	filemode	enters VM700T password
pwd	filemode	returns VM700T directory path
readfile	filemode	reads a file from the VM700T
rm	filemode	removes a VM700T file
rmdir	filemode	removes a VM700T directory
writefile	filemode	writes a file to the VM700T
filesin	remote	returns names of all files in a directory
get	remote	returns configuration values specified by a keyword
getclock	remote	returns date and time from the VM700T system clock
getdisptext	remote	returns text generated by a disptext function key command
getresults	remote	writes results file for current application
getspooltext	remote	returns text spooled by a VM700T application
hardkey	remote	presses and releases a specified front-panel button
hardpress	remote	presses, without releasing, a specified front-panel button
hardrelease	remote	releases a specified front-panel button
help?	GPIB	returns a list of VM700T/GPIB commands

Table 1–3: VM700T GPIB Remote Commands (Cont.)

Command	Type	Purpose
id?	GPIB	returns the VM700T/GPIB identification
init	GPIB	initializes the VM700T/GPIB to a known state
knob	remote	turns the control knob as specified by a numerical argument
mode?	GPIB	returns the current VM700T mode
playback	remote	executes a specified VM700T function key file
print	remote	sends a formatted file to the print spooler
query	remote	returns information about a VM700T keyword
rename	remote	renames a file in the VM700T file system
res	remote	returns coded string results for current measurement
restoreconfig	remote	restores configuration values from VM700T files
resumefunction	remote	resumes a VM700T function which is waiting on disptext
rqs	GPIB	turns the state of RQS on or off: rqs on or rqs off
rqs?	GPIB	queries the state of RQS
send	remote	returns results of current measurement in GPIB format
set	remote	sets configuration values specified by a keyword
setclock	remote	sets date and time on the VM700T system clock
show	remote	returns the contents of a specified file
softkey	remote	presses and releases a specified soft key
softpress	remote	presses, without releasing, a specified soft key
softrelease	remote	releases a specified soft key
spool	remote	sends an ASCII file to the print spooler
stopfunction	remote	stops a VM700T function
touchpress	remote	touches VM700T screen at specified x, y location
touchrelease	remote	releases previous touchpress

GPIB Commands

GPIB Commands

This section describes the VM700T Video Measurement Set GPIB commands and gives examples of how they may be used to control the operation of the VM700T and work in its internal file system through the GPIB interface option.

VM700T Command Set Detailed Description

This section describes the command set, in more detail.

Notation used in this document:

`<arg>`	A required argument. 1 or more arguments.
`[<arg>]`	An optional argument. 0 or more arguments.

Keywords used in this document:

`file`	VM700T style filename (31 chars max, Alpha/Numeric, ...)

`<path>`	Absolute or relative path in VM700T directory structure.
	`<path>` is a UNIX style path. Examples are as follows:
	`.`
	`..`
	`../directory`
	`/dir1/dir2/dir3`

`<filename>`	Valid VM700T file name (31 chars max, Alpha/Numeric, may contain a path).
`<dirname>`	Valid VM700T directory name.
`<passwd>`	Valid VM700T password.
`<result position>`	Position of result returned in res or send command.
`<definite length arbitrary block>`	A data format for sending fixed length 8-bit data.

VM700T Commands

The following is an alphabetical listing of the GPIB commands recognized by the Video Measurement Set.

***NOTE**. The response from the VM700T should be read after every command is sent. This will avoid loading the queue with unread responses. If the VM700T has nothing to say, it still sends FF_{Hex} (ASCII 255).*

cancelcopy

Syntax Form: `cancelcopy`

Type: Remote

See also: `print`

Send `cancelcopy` to clear the print spooler. The command is equivalent to touching the Cancel Copy soft key after pressing the Configure button while the VM700T contains spooled data.

control

Syntax Form: `control string`

Type: Remote (function playback, remote operation)

The `control` command sends a user-specified string out the control port. This string is either any sequence of ASCII characters or any of the following:

^x	a control character, e.g., ^G (bell) or ^M (carriage return)
$nn	a hexadecimal number, where n is 0-9, A-F, or a-f
\$	the '$' character
\^	the '^' character
\\	the '\' character
\<LF>	a trailing '\' on a line says to ignore end-of-line

***NOTE**. GPIB cannot be selected as the control port. Control is meant for use in controlling a serial device connected to one of the RS-232 ports of the VM700T.*

The `control` command is usually used to control a device attached to the VM700T, such as a modem. Note that communication through the `control` command is one-way only. There is no feed-back from the controlled device. This means that if the controlled device is not powered on, or is not even attached, or is sent an incorrect string, the VM700T function or program will continue executing. If a control port is not enabled when a `control` command is sent, a VM event code is generated, and SRQ is asserted (with RQS on).

Note also that the control statement does not add characters to delimit messages. Thus, if the device being controlled requires delimiting messages with a carriage return, line, or carriage-return/line-feed sequence, include those characters in the string accompanying the `control` command.

Examples:

```
control ^G
```

This example sends a bell character out the Control Port.

```
control ATDT123-4567^M
```

This example sends the ASCII string "ATDT123-4567", followed by carriage return, out the Control Port.

controlbreak

Syntax Form: `controlbreak number`

Type: Remote (function playback, remote operation)

Use `controlbreak` to send a break character out the Control Port for the specified number of time units (tenths of seconds). If a Control Port is not enabled, a VM event code is generated, and SRQ is asserted (with RQS on).

Example:

```
controlbreak 10
```

This command sends a break character out the Control Port for one second.

NOTE. *If* `controlbreak` *is executed from remote control or function playback immediately after a* `control` *command has printed characters to the port, the break character may be transmitted for a shorter time than requested. (This happens because of the way the VM700T buffers* `control` *and* `controlbreak` *requests.) To ensure that the break character is sent for the full amount of time requested, allow sufficient time for all characters printed with the* `control` *command to finish printing before issuing the* `controlbreak` *command.*

event?

Syntax Form: `event?`

Type: GPIB

The `event?` query returns the GPIB event code <number>. Use this command after an SRQ is received to find the reason for the SRQ. Compare `evmsg?`

Example:

```
event?
```

```
EVENT 0
```

evmsg?

Syntax Form: `evmsg?`

Type: GPIB

Use the `evmsg?` query to determine the GPIB event code <number> and the VM700T error number (or 0 if there is none). Use this command after an SRQ is received to find the cause of the SRQ. Compare `event?`

Example:

```
evmsg?
EVMSG 0, (VM700 = 0)
```

execute

Syntax Form: `execute string`

Type: Remote (function playback, remote operation)

Send `execute` to start a named VM700T application. An application is one of the executable files (with exceptions noted below) found in the Instrument Operations, VM700T Diagnostics, or Video Measurements directories in the Executable Files directory. Selecting an operational mode application (Waveform, Vector, Picture, or Auto) is equivalent to pressing one of the mode buttons on the front panel; the LED lights up on the corresponding button. Selecting a measurement or diagnostic application is equivalent to touching a soft key from one of the Measure-mode windows.

Example:

```
execute ChromLum~GainDelay
```

This example starts the ChromLum GainDelay application.

When specifying application names in functions or remote operation commands, it is important to match the case (upper or lower) and any special characters used in the application's name. See the *VM700T Programmer Manual* for specifying application names.

Filemode Commands

The following subset of commands are those used to work in the directories of the VM700T. They are used after entering filemode by sending the `filemode` command.

filemode

Syntax Form: `filemode`

Type: Remote (remote operation)

Sending `filemode` puts the VM700T into GPIB File Transfer Mode. Sending a remote command to the VM700T terminates the GPIB File Transfer Mode. A special display is present while filemode is in effect.

The file transfer commands allow navigating through the VM700T file system and writing and reading of files to and from the VM700T through the GPIB interface. The information returned is in <definite length arbitrary block format> (for example, `#6000033[arbitrary text of definite length]`) and terminated by LF - EOI. Each of the commands will generate a response from the VM700T regarding the success or lack of success of employing the command. The `filemode` command responses are all terminated by an appended LF - EOI. These ASCII reply strings are in addition to any error messages generated in the event of a GPIB or VM command error code.

If you send the Go to Local GPIB command to the VM700T while filemode is active, the GPIB filemode display will remain, but filemode is terminated. You must send an executable command to the VM700T to remove the GPIB filemode display. If you wish to return to filemode, send the `filemode` command again to regain remote control.

In filemode the following commands are available.

cd

Syntax Form: `cd "<path>"`

Type: Filemode

See Also: `dir, ls, mkdir, mv, passwd, pwd, rm, rmdir, readfile, writefile`

Use `cd` to change the current directory to a new directory. Sending the `cd` command alone without an appropriate path generates a filemode error.

Example:

```
cd ..
cd ".."
cd ConfigFiles

cd "ConfigFiles"
Change directory successful
```

NOTE. *The command arguments work with and without quotation marks.*

dir

Syntax Form: `dir [ "<path>"]`

Type: Filemode

See Also: `cd, ls, mkdir, mv, passwd, pwd, rm, rmdir, readfile, writefile`

The `dir` command list the byte size and filenames in the designated directory. Sending the `dir` command alone without a path argument generates a list of the current directory files.

Example:

```
dir ..
dir ".."
dir /nvrom0

dir ConfigFiles
#6000349       0   Measurement~Results
     0    Auto_Limit~Files
     0    Camera_Testing
   738    Communication~Setup
   189    Diagnostics~Selection
     0    Measure_Limits~Files
     0    Measurement~Locations
     0    Selected~Measurements
   388    Source_Selection~Video
     0    Timed~Events
     0    Video_Source~Files
   815    Video_Source~Identification
```

ls

Syntax Form: `ls ["<path>"]`

Type: Filemode

See Also: `cd, mkdir, mv, passwd, pwd, rm, rmdir, readfile, writefile`

The `ls` command lists out the VM700T directory contents.

Note that a file called GPIBtmp is created during execution of the `ls` command. This file is deleted when the VM700T leaves File Transfer mode.

Example:

```
ls
#6000102Config Files
FunctionKeys
Executable~Files
Help~Files
IpConfig
Software~Version
```

```
GPIB_ALTERNATE
GPIBtmp
```

mkdir

Syntax Form: `mkdir "<dirname>"`

Type: Filemode

See Also: `cd, ls, mv, passwd, pwd, rm, rmdir, readfile, writefile`

Use `mkdir` to create a new named directory in the VM700T.

Example:

```
mkdir "Site_2~Measurements"
Make directory ok
```

mv

Syntax Form: `mv <filename1> , <filename2>`

Type: Filemode

See Also: `cd, ls, mkdir, passwd, pwd, rm, rmdir, readfile, writefile`

Use `mv` to change a file's name from filename1 to filename2.

Example:

```
mv myfile , oldfile
```

passwd

Syntax Form: `passwd` "<passwd>"

Type: Filemode

See Also: `cd, ls, mkdir, mv, pwd, rm, rmdir, readfile, writefile`

Use the `passwd` command to enter a VM700T password to access protected files. With password in effect, the assigned password is needed each time File Transfer mode is entered. If the password function is not enabled, a password is not needed.

pwd

Syntax Form: `pwd`

Type: Filemode

See Also: `cd, ls, mkdir, mv, passwd, rm, rmdir, readfile, writefile`

Sending pwd returns the complete path name of the current VM700T directory.

Example:

```
pwd
Current directory is /nvram0/Executable~Files
```

readfile

Syntax Form: readfile "<filename>"

Type: Filemode

See Also: cd, ls, mkdir, mv, passwd, pwd, rm, rmdir, writefile

The readfile command returns a VM700T file. The file contents can be ASCII or 8-bit binary data. The file is returned in <definite length arbitrary block> format.

For example, a file that contains “abc” is returned as #6000003abc where the first digit, 6, states how many digits define the block length (000003), and abc is the data block.

Example:

```
readfile "myfile"
#6000120file text of 120 characters . . .
```

If the file requested is not found, an error is generated.

Example:

```
readfile "GPIB_ALT"
Open of "GPIB_ALT" failed, Not found
```

For this error, a serial poll and evmsg? query returns the following:

```
Status byte = 98
EVMSG 253, (VM700 = 550)
```

rm

Syntax Form: rm “<filename>”

Type: Filemode

See Also: cd, ls, mkdir, mv, passwd, pwd, rmdir, readfile, writefile

The rm command removes a named file in the VM700T.

Example:

```
rm oldfile
Delete command okay
```

rmdir

Syntax Form: `rmdir "<path>"`

Type: Filemode

See Also: `cd, ls, mkdir, mv, passwd, pwd, rm, readfile, writefile`

Use `rmdir` to remove a named directory in the VM700T.

Example:

```
rmdir <directory name>
delete command okay
```

writefile

Syntax Form: `writefile` *<filename>,<definite length arbitrary block>*

Type: Filemode

See Also: `cd, ls, mkdir, mv, passwd, pwd, rm, rmdir, readfile`

Use `writefile` to write a file into the VM700T.

Example 1:

The command to write the file `"test"` consisting of the data abc123 in the command line could look like this:

```
writefile test,#3006abc123
```

with 3 meaning the next 3 digits define the block length where 006 is the block length (abc123).

Example 2: (if using IBIC and GURU)

If you want to write a pc file to the VM700T, the following procedure using IBIC commands may prove helpful in following the steps needed.

1. Edit and save the file you want to send to the VM700T. In this example we will name that file `"test"`.
2. DIR the filename to determine how many bytes the file contains.
3. Edit the file again and add the definite length arbitrary block information to the start of the file:

   ```
   #6000100This is a file of 100 bytes .........
   ```

   and resave the file.

4. Create a second file to contain the IBIC commands to send the first file. For this example, call it “sendfile”.

```
ibfind VM700T
ibwrt "filemode"
ibeot 0
ibwrt "writefile test1,"
ibeot 1
ibwrtf test
ibwrt "execute Waveform"
```

The commands in this sendfile do the following tasks:

- finds the VM700T on the bus
- puts the VM700T in filemode
- turns off EOI
- sends the `writefile` command to write to a file called `test1` in the VM700T
- turns EOI back on
- sends the pc file `"test"` to the VM700T
- returns the VM700T back to the Waveform display

Turning off EOI before the `ibwrt` command leaves the command line open so that the second command, `ibwrtf`, will be appended. The `ibeot 1` command turns EOI back on so the end of the pc file will be appended with the EOI.

IBIC has a command used to access a file containing further IBIC commands. Start IBIC and use that command to read the sendfile.

```
$ sendfile
```

filesin

Syntax Form: `filesin string`

Type: Remote (remote operation)

Use `filesin` to get the names of all files in the specified directory in definite length arbitrary block format.

Example:

```
filesin /nvram0/Executable~Files

#6000081Instrument~Operations/

VM700~Diagnostics/
```

```
Video~Measurements/

Video~Options/
```

get Syntax Form: `get string [A|B|C]`

Type: Remote (remote operation)

See also: `set`

Use the `get` command to return the configuration file value specified by `keyword` on the channel specified by `channel_letter`. The keywords for the standard instrument and all available options are given in *Appendix A: Get/Set Keywords* of the *VM700T RS-232 Interface Programmer Manual*.

Example:

```
get VSTA
NTSC
get DHSM A
Meas_Set_1
get SPCF
ASCII Printer
```

The first command in the example returns the current video standard for source A. (In this case, it is NTSC.) The second command in the example returns the name of the current Selected Measurements File for source A when using the NTSC standard. (The name of this file is Meas_Set_1.) The third example returns the present format for the Copy port output.

getclock Syntax Form: `getclock`

Type: Remote (remote operation)

The `getclock` command returns the date and time from the VM700T system clock in the form:

```
mon dd hh:mm:ss yyyy
```

where `mon` is the first three letters of the month, `dd` is the day of the month, `hh` is the current hour, `mm` is the minute, `ss` is the second, and `yyyy` is the current year. The `getclock` command takes no arguments.

Example:

```
getclock
Aug 30 14:54:37 1996
```

getdisptext

Syntax Form: `getdisptext`

Type: Remote

Sending the `getdisptext` command returns disptext (displayed text) issued by a VM700T function in definite length arbitrary block format. An SRQ is generated to indicate that disptext is available. Note that the function is stopped after sending disptext, and the controller must send a `resumefunction` command to continue the function.

If there is no text, a single byte, `0xFF,` is returned.

***NOTE**. This command may take an extended period of time to execute with the VM700T in the Auto Mode and for the Camera Option measurements. You must allow time for it to return with the results file message in your programming to prevent a GPIB time out error.*

getresults

Syntax Form: `getresults [verbose] [string[ string]]`

Type: Remote (function playback, remote operation)

Use the `getresults` command to store Measure or Auto mode measurement results in default files in the Measurement~Results directory. In Measure mode, sending the `getresults` command with no argument(s) stores the measurement results for the current measurement. If no measurement is currently being executed, the message `"Request not supported"` is returned. If a measurement is being executed, a `filename` is returned. Use the `show filename` command to view the results.

Example:

```
getresults
ChromLum~GainDelay
show ChromLum~GainDelay
```

***NOTE**. The* `getresults verbose` *command will generate an error if sent when asking for AUTO mode measurement results.*

Entering `getresults verbose` in Measure mode stores additional information for DGDP, GroupDelay~SinX_X, Luminance~NonLinearity, and Noise~Spectrum measurements. The information is displayed as one or more rows of unlabeled numbers, and is set off from the main file display by a line of plus signs (++++).

In Auto mode, entering `getresults` with no argument(s) executes the current selected measurement list and stores the results in the Measurement~Results Auto file. `Auto` is the message returned.

You can also use `getresults` in Auto mode with one or more keyword arguments. The keywords, listed in Appendix A of the *VM700T Programmer Manual*, in effect specify a temporary selected measurements list that overrides the active Selected Measurements file. The new selected measurements stay in effect until a `restoreconfig` command or a `set` command specifying the Selected Measurements file is issued. The GPIB `init` command will also restore the complete measurements list to the AUTO mode measurements when AUTO mode is again started.

Example:

```
execute H_Timing
getresults
H_Timing
```

The preceeding command sequence executes the H_Timing measurement, then stores the results in file /nvram0/ConfigFiles/Measurement~Results/H_Timing.

```
execute Auto
getresults PBAA PBRT PSTB
Auto
```

The preceeding command sequence creates a temporary selected measurements list of three measurements: Luminance Bar Amplitude, Bar Rise Time, and Sync-to-Burst Start.

getspooltext

Syntax Form: `getspooltext`

Type: Remote

See also: `spool`

The `getspooltext` command returns text sent by the VM700T print spooler in Auto mode or Function mode or any file you spooled using the `spool` command. The text is returned in definite length arbitrary block format. An SRQ is generated when spooled text is available. You may queue up to 10 spooled text messages (less if the spool memory is full).

Note that spooled text is sent by the VM700T, even if GPIB is not selected as the remote port, if the copy, log, or report ports are set to GPIB, and the user initiates an action which causes the print spooler to function.

For example, if the copy port is set to GPIB, the data is sent to the GPIB interface card as spooled text when the user presses the COPY button.

If there is no text to return, the `getspooltext` command returns the single hex byte FF.

hardkey

Syntax Form: `hardkey string`

Type: Remote (function playback, remote operation)

See also: `hardpress`, `hardrelease`

Sending the `hardkey` command is equivalent to pressing and releasing a specified front panel button. The `hardkey` command has the same effect as `hardpress` followed by `hardrelease`. However, in general use `hardkey` instead of these commands.

Example:

```
hardkey Vector
```

This command has the same effect as pressing the Vector button on the front panel. See the *VM700T Programmer Manual* for the button names.

NOTE. *You can not select the Configure, Function, and Measure buttons in functions or remote operation.*

When specifying button names in functions or remote operation commands, match the name given in the *VM700T Programmer Manual*, including upper or lower case.

If you use the `hardkey` command to press the COPY button, the Copy port must be set to GPIB to spool the display to the GPIB card. In addition, the file format must be correct for the type of display your are attempting to send to the Copy port (that is, a waveform display must be PostScript or HPGL and a text display should use ASCII Printer or EPSON LQ). Since pressing the front panel button is usually a local event, the error indicator in the event of an improper format selection is to sound the VM700T alarm bell; a GPIB error or event is not generated.

hardpress

Syntax Form: `hardpress string`

Type: Remote (function playback, remote operation)

See also: `hardkey`, `hardrelease`

Sending the `hardpress` command is equivalent to pressing a specified front panel button without releasing it. The button remains “pressed” until a `hardrelease` command is received. Use the `hardpress` and `hardrelease` commands for buttons you have to hold in while another action takes place. For all other buttons, use `hardkey`. Front-panel button names are listed in the

VM700T Programmer Manual. Note also that if a button toggles (e.g., Freeze, Average) you can use `hardpress` again to turn it off.

When specifying button names in functions or remote operation commands, match the name shown in the *VM700T Programmer Manual*, including upper or lower case.

Example:

```
hardpress Display
knob 50
hardrelease Display
```

The preceeding command sequence presses and holds the Display button, increases the display intensity by turning the knob clockwise fifty clicks, then releases the Display button.

hardrelease

Syntax Form: `hardrelease string`

Type: Remote (function playback, remote operation)

See also: `hardkey`, `hardpress`

Sending the `hardrelease` command indicates that the specified front panel button has been released. Front-panel button names are listed in the *VM700T Programmer Manual*.

Example:

```
hardrelease Display
```

The preceeding example releases the Display button on the front panel. See also the example accompanying the explanation of `hardpress`.

help?

Syntax Form: `help?`

Type: GPIB

The `help?` query returns a list of all available command headers in the VM700T/GPIB option.

id?

Syntax Form: `id?`

Type: GPIB

The `id?` query returns the instrument identification and current GPIB firmware release.

Example:

```
ID TEK/VM700T/GPIB,FV1.0
```

init Syntax Form: `init`

Type: GPIB

Use the `init` command to initialize the VM700T/GPIB to a known state. The initialization actions include:

VM700T goes to Display (Waveform) mode.

`restoreconfig` is issued to the VM700T.

The spooled text queue is cleared.

The disptext buffer cleared.

RQS is set ON.

***NOTE**. Queued error messages are not cleared.*

knob Syntax Form: `knob integer`

Type: Remote (function playback, remote operation)

See also: `cknob`

Use the `knob` command to turn the control knob a specified number of clicks. The effect of a single click depends on the current application and the current screen scaling. A positive number of clicks turns the knob clockwise; a negative number of clicks turns the knob counterclockwise.

Example:

```
knob -50
```

This command turns the knob 50 clicks counterclockwise.

mode? Syntax Form: `mode?`

Type: GPIB

The `mode?` query returns the current mode of the VM700T. The possible modes include:

Display (Waveform, Vector, or Picture)

Application (Video, Audio, or Diagnostic application)

Auto

FTP

File Transfer

Function

Function Waiting

Function Done

GPIB not active

GPIB connected

The GPIB card maintains the mode of the VM700T and will issue extra commands (such as ‘execute Waveform’) to put the VM700T into a known mode. For example, after a VM700T function has been executed, the GPIB card will send an `'execute Waveform'` command to the VM700T.

***NOTE**. If you go to local control mode by sending the GPIB Go to Local command to the VM700T, the GPIB card remains connected to the VM700T to send appropriate commands to the VM700T and put it back in remode control. However, the GPIB card doesn't know the mode that the VM700T was changed to while in local, and even if a command such as* `id?` *is sent that turns on the Configure LED to show that remote control is again in effect, the reply to a* `mode?` *query will remain* `GPIB connected`*. You must send a command that the VM700T executes before the GPIB card knows the VM700T mode.*

---

playback

Syntax Form: `playback "<filename>"`

Type: Remote

See also: `displaytext, resumefunction, stopfunction`

Sending the `playback` command starts executing a VM700T function. The file designated must contain a valid VM700T function. The default path for VM700T function files is `/nvram0/FunctionKeys`.

If the playback function contains a `print` command, The GPIB card will issue an SRQ to signal the controller that there is spooled text. The controller can then get the text by sending a `getspooltext` command. If the Copy port is set to GPIB, the spooled text will be sent to the GPIB card.

If the playback function contains a `disptext` command, the GPIB card issues an SRQ to signal the controller that a disptext message is available. The controller can get the disptext message by sending the `getdisptext` command. The controller must send a `resumefunction` command to continue the executing function.

While a playback is executing, the controller can send GPIB commands, but it can only send the `getdisptext`, `resumefunction`, or `stopfunction` remote commands. All other remote commands generate an error event.

print

Syntax Form: `print string`

Type: Remote (function playback, remote operation)

See also: `spool`

The `print` command sends a formatted file to the print spooler. A file header is prepended showing the file name, the date and time, and the page number.

NOTE. *Waveform displays can not be printed with ASCII format selected. You must properly format the output file (for example, PostScript) for a graphics device capable of handling a waveform display.*

The file's format is specified by the Format item for the Copy port in the Communications Setup file. You may determine the format by using the `get` command with the keyword for the format setting (SPCF). You may set the format by using the `set` command with the keyword and the correct argument for the format you want.

Example:

```
get SPCF
ASCII Printer
set SPCF PostScript
```

The default path for the file to print is the Measurement~Results directory. Files in other directories are specified with a full pathname or a path relative to the Measurement~Results directory. Compare `spool`. To get waveform displays via the GPIB card into the GPIB controller, use the `hardkey Copy` command; then, upon SRQ, use the `getspooltext` command to read the data. The printer format must be set to output a graphics file (for example, PostScript), and your program will have to be able to handle the large data files.

Example:

```
print ChromLum~GainDelay
```

This example sends the file ChromLum~GainDelay from the Measurement~Results directory to the print spooler, in the format currently specified by the Format item for the Copy port in the Communications Setup file. It will only send to the GPIB if the Copy port is set to GPIB.

```
print/rom/ConfigFiles/Measurement~Locations/NTSC/System~Default
```

This example prints the default Measurement Locations file for the NTSC standard, in the format currently specified by the Format item for the Copy port in the Communications Setup file. (Note that the path shown only applies to VM700T units equipped with both Option 01, NTSC, and Option 11, PAL.) Again, the file is sent to the GPIB only if the Copy port is set for GPIB.

query

Syntax Form: `query string`

Type: Remote (remote operation)

Sending a `query` command returns information about the VM700T keyword used as its argument. Information returned includes the type and range of permissible values for each field returned by the get command or used as an argument by the set command. If the field specifies a numeric value, the permissible range of numbers is returned. If the field specifies a file name or choice of other discrete values, the useable argument values are listed.

Examples:

```
query VSCA
#6000033F1: file list:
System~Default
query LZCL
#6000019F1: integer 1 625
query SPOA
#6000032F1: string list:
None
SLIP
```

The information is returned in <definite length arbitrary block> format (for example, `#6000033[arbitrary text block of definite length]`) and terminated by LF and EOI.

rename

Syntax Form: `rename string string`

Type: Remote (function playback, remote operation)

Use `rename` to change the name of a file in the Measurement~Results directory. This command is particularly useful when you want to get two or more results files from the same measurement and compare them.

You can get the results from the first measurement, rename the first measurement's results file, then get a second results file from the same measurement. You can then compare the two results files visually, or transfer their contents to a computer for automated comparison.

Example:

```
rename H_Timing H_Timing_1
```

This command changes the file named "H_Timing" in the Measurement~Results directory to "H_Timing_1".

res

Syntax Form: `res[ -v][,<result position>[,<result position>]...]`

Type: Remote

The `res` command returns the result of executing a measurement on the VM700T. If the optional argument "-v" is given, the data returned is an ASCII string without a <definite length arbitrary block> header. Otherwise, the data returned is binary encoded data in <definite length arbitrary block> format.

Example:

```
execute H_Timing

res -V 1
18 1:141343
```

where 18 identifies the H_Timing measurement and 1 is the first measurement result. The res -v command without further position modifiers returns all the measurement results.

restoreconfig

Syntax Form: `restoreconfig`

Type: Remote (function playback, remote operation)

The `restoreconfig` command restores configuration values from the Configure files. Exiting a function or exiting remote operation does not automatically restore these values; use `restoreconfig` to do so. Note that the system line and other global variables are not restored with `restoreconfig`.

resumefunction

Syntax Form: `resumefunction`

Type: Remote

See also: `playback, stopfunction`

The resumefunction command continues a VM700T function that was halted by a disptext command in a function playback. While a function is halted, a mode? query returns "Function Waiting" until the resumefunction command is given to finish the function. When disptext (display text) is available, an SRQ is asserted (if RQS is on).

rqs

Syntax Form: `rqs [on/off]`

Type: GPIB

Sending `rqs on` sets the RQS flag in the VM700T. RQS ON means that an SRQ is generated for each event. With RQS off an SRQ is not generated for each event. Turning RQS off prevents error events from generating an SRQ, but the messages are queued and may be read by doing a serial poll. When RQS is turned on again, any unread event messages queued during the time RQS was off are cleared.

rqs?

Syntax Form: `rqs?`

Type: GPIB

Sending `rqs?` returns the state of the RQS flag.

send

Syntax Form: `send[ <result position>[,<result position>]...]`

Type: Remote

Example:

The `send` command returns the result of executing a measurement on the VM700T file. This command is similar to the res command with the -v option, except that the data is returned in NR3 format.

```
execute H_Timing

send 1
18 1:1.420E-07
```

where 18 again identifies the H_Timing measurement and 1 is the first measurement result. The `send` command without further position modifiers returns all the measurement results. The reply does not use a <definite length arbitrary block> header.

set

Syntax Form: `set string [A|B|C] number|string [number|string...]`

Type: Remote (function playback, remote operation)

See also: `get, query`

The set command defines the configuration values for use during the remote session. The keywords available to use with set are listed in *Appendix A: Get/Set Keywords* of the *VM700T RS-232 Interface Programmer Manual*. Configuration values changed with set remain in effect until they are changed by another set command, restoreconfig is executed, or the instrument is powered off. Note that set can change the system line and other global variables, but they are not restored with restoreconfig.

Example:

```
set GLN5 100
set GLN6 18
set SPCF PostScript
```

This command sequence sets the system line for NTSC to 100, the system line for PAL to 18, and the serial port copy format to PostScript.

setclock

Syntax Form: setclock *string*

Type: Remote (remote operation)

See also: getclock

Use setclock to set the date and time on the system clock, using the following format in the argument string:

```
mon dd hh:mm:ss yyyy
```

where mon is the first three letters of the month, dd is the day of the month, hh is the current hour, mm is the minute, ss is the second, and yyyy is the current year. Hours are specified in 24-hour format (00 for midnight, 23 for 11 p.m.).

Example:

```
setclock Aug 30 21:16:22 1996
```

show

Syntax Form: show string

Type: Remote (remote operation)

The show command returns the contents of a specified file in definite length arbitrary block format. The default path for the file is the Measurement~Results directory. Specify files in other directories by supplying a full pathname or a path relative to the Measurement~Results directory.

Example:

```
show /nvram0/Software~Version
```

This example returns the contents of file Software~Version in the nvram0 directory.

```
#60001330ption 01 NTSC Version 2.09
Option 11 PAL Version 2.09
```

softkey

Syntax Form: `softkey string`

Type: Remote (function playback, remote operation)

See also: `softpress`, `softrelease`

Sending the `softkey` command is equivalent to pressing and releasing a specified softkey. Sending `softkey` has the same effect as `softpress` followed by `softrelease`. However, in general use `softkey` instead of these commands..

When specifying soft key names in functions or remote operation commands, it is important to match the case (upper or lower) and any special characters used in the soft key name. Refer to *Naming Conventions* in the *VM700T RS-232 Interface Programmer Manual*, for information about the rules for specifying soft key names.

Example:

```
softkey ITS_Search
```

This example is equivalent to pressing and releasing a soft key labeled “ITS Search” on the touch screen.

softpress

Syntax Form: `softpress string`

Type: Remote (function playback, remote operation)

See also: `softkey, softrelease`

Sending the `softpress` command is equivalent to pressing a specified soft key without releasing it. The soft key remains “pressed” until a `softrelease` or `touchrelease` command is received. Use `softpress` and `softrelease` with soft keys you have to touch while another action takes place. For all other soft keys, use the `softkey` command. Note also that if a soft key’s function toggles, you can use `softpress` again to turn it off.

When specifying soft key names in functions or remote operation commands, it is important to match the case (upper or lower) and any special characters used in the soft key’s name. Refer to *Naming Conventions* in the *VM700T RS-232 Interface Programmer Manual*, for information about the rules for specifying soft key names.

Example:

```
softpress Rescale
```

The preceding example is equivalent to pressing a soft key labeled “Rescale” without releasing it.

softrelease

Syntax Form: `softrelease`

Type: Remote (function playback, remote operation)

See also: `softkey`, `softpress`

The `softrelease` command indicates that a soft key has been released.

spool

Syntax Form: `spool` <*filename*>

See also: `print`, `getspooltext`

Type: Remote

The `spool` command sends an unformatted, ASCII-text file to the print spooler. The default path for the file is the Measurement~Results directory. Specify files in other directories with a full path name or a path relative to the Measurement~Results directory.

The VM700T GPIB Interface will issue an SRQ when spooled text is available. The controller can then issue a `getspooltext` command to get the text. The spooled files will only go to GPIB if the Copy port is set to GPIB in the Communication~Setup file.

Example:

```
spool ChromLum~GainDelay
```

The preceeding example sends the file ChromLum~GainDelay from directory Measurement~Results to the print spooler, in ASCII-text format.

```
spool/rom/ConfigFiles/Measurement~Locations/NTSC/System~Default
```

The preceeding example prints the default Measurement Locations file for the NTSC standard, in ASCII-text format. (Note that the path shown only applies to VM700T units equipped with both Option 01, NTSC, and Option 11, PAL.)

stopfunction

Syntax Form: `stopfunction`

Type: Remote

Sending the `stopfunction` command stops a currently executing VM700T function. Use the `stopfunction` command to stop a looping function. The

VM700T remains in the state that is executing at the time the `stopfunction` command is received.

touchpress

Syntax Form: `touchpress` *number number*

Type: Remote (function playback, remote operation)

Descriptive Form: `touchpress` *x_coordinate y_coordinate*

See also: `touchrelease`

The `touchpress` command indicates that a specified x,y location on the touchscreen is being "touched". The 0,0 location is the upper left corner of the screen. The X range is 0 to 639; the Y range is 0 to 479. If the x,y location is within a soft key, the soft key is executed.

Example:

```
touchpress 200 330
```

This example "touches" the screen location 200 units to the right and 330 units down from the upper left corner of the screen.

touchrelease

Syntax Form: `touchrelease`

Type: Remote (function playback, remote operation)

Descriptive Form: `touchrelease`

See also: `softpress`, `touchpress`

The `touchrelease` command indicates a soft key or touch screen location has been released.

Status and Events

Status and Events

This section provides two tables. The first describes the GPIB event codes, and the second gives a list of the status bytes with RSQ on and RSQ off.

GPIB Events

The events returned from the VM700T GPIB card can consist of two parts, the GPIB code and the VM700T code. The GPIB code only is returned if the `event?` command is sent. The GPIB code and VM700T code (or 0 if none) is returned if the `evmsg?` command is sent.

Table 3–1: VM700T GPIB Event Codes

Code	Description
GPIB Events	
101	Command Header Error
103	Command Argument Error
150	VM700T can't parse or execute command
203	I/O buffers Full, output dumped
250	Can't execute command in FUNCTION mode
251	Can't execute command in AUTO mode
252	Can't execute File Transfer command
253	Bad File Transfer command
254	Error during execution of FUNCTION
255	Bad VM700T response to command
350	GPIB event queue full
401	Power On
402	Operation Complete
403	VM700T Configuration button pressed
450	Function end
451	disptext data available
452	Spooled data available
453	Connection to VM700T not active
VM700T Command Errors (returned by evmsg? query)	
001	Bad command argument
002	Sub-function not found
003	Playback nesting too deep

Table 3–1: VM700T GPIB Event Codes (Cont.)

Code	Description
VM700T Command Errors (returned by evmsg? query)	
004	Function directory inaccessible
005	Function not found
006	Unknown command
007	Unknown hard key
008	Out of memory
009	Recursive function call
010	Bad command in this context
011	Name too long
012	No filename
013	Line too long
015	Bad time format (use getclock)
019	Bad control format
020	Bad connection for remote graphics
021	Copy port not enabled
022	Bad conditional
023	Bad conditional type
099	Undefined error
101	Request filtered
102	Screen event not key
103	Unknown soft key
104	Invalid soft key
105	Unwanted hard key
106	Unknown input
107	Not found
108	Request not supported
109	No server resources
110	Illegal name
111	Not writable
112	Not readable
113	No permission
114	Bad arguments
115	Connection does not support data output
199	Undefined error

Table 3–1: VM700T GPIB Event Codes (Cont.)

Code	Description
VM700T File Transfer errors	
426	Verification failed
501	Unknown descriptor
530	Password command error
	Password required
	Incorrect password
550	Filemode error
	Cannot open
	Cannot read
	Cannot write
	Failure in **cd** command
	Failure in **pwd** command
	Failure in **rm** command
	Failure in **rmdir** command
	Failure in **mkdir** command
	File not found
	File not deletable
	File not writable

GPIB Status Bytes

A status byte is returned by the VM700T GPIB interface card when it is serial polled by a controller. Table 3–2 shows the status bytes when the RQS bit is enabled (On) and disabled (Off).

Table 3–2: GPIB Status Bytes

Status Byte		
RSQ On	**RSQ Off**	**Description**
0	0	No status to report
65	1	Power on
66	2	Operation complete
67	3	User request
97	33	Command error
98	34	Execution error
99	35	Internal error
101	37	Execution warning
102	38	Internal warning
192	128	System event
193	129	Device Dependent event

VM700T Programming Procedures

VM700T Programming Procedures

If you are not a skilled programmer, you will need the assistance of someone that is to write the initial GPIB controller program. In the DOS environment you can use the GURU II software package to assist in writing a GPIB communications program using either C, BasicA, or DOS QuickBasic. Library files of many of the necessary routines and variable declarations are included in the GURU II package. Using those and the correct compiling and linking commands, you can quickly produce simple drivers for issuing instrument commands and handling simple responses.

The program can be done very simply for sending commands one at a time and handling the responses of a single instrument; or it can be a very detailed programming task, providing for multiple instrument operation. The following steps provide an outline of the sequence of operations performed in a typical instrumentation program. Using the outline as a guide, you can generate complete applications to meet the needs of your application.

These steps are very general in application. Specific information is found in the GPIB driver manuals for your controller and in other resource manuals such as the GURU II manual. The complexity of your application or system determines the real programming requirements and programming skills needed.

1. Initialize DOS

a. Use DOS commands to initialize CONFIG.SYS parameters that must be included to find the GPIB programming files.

Example:

```
buffers=20
device = \guru\gpib.com
```

b. Again, with the DOS commands, make any changes to the AUTOEXEC.BAT commands you need for your application. Some typical things to consider are:

I/O redirection
Special Messages to the user
Setting the system clock
Initializing RAM drives, if used
Setting VERIFY and BREAK to ON, if desired
Automatic entry into the applications program, once written

2. Initialize the Program (these are commands within the application)

- Declare and initialize the program variables. This includes creating the needed libraries and linking those libraries so the program has use of the

previously declared variable and routines that are supplied with many GPIB driver utilities. You will have to use the manuals supplied with the utility to determine the needed files and organization. Information on creating libraries and linking is found in the *DOS Basic Programmer's Manual*.

- Present a message to the user. (This message may provide options for operation of the application or simply an instruction to enter a command.)
- Clear the GPIB interface (GPIB0:) with the IBSIC command.
- Use the IBFIND command to locate the devices on the bus and obtain their id numbers.
- Check for the presence of instruments at expected addresses, or auto configure the GPIB using time outs to detect no instrument present.
- Clear each instrument found using the IBCLR command.
- Identify each instrument, if possible, to verify that the proper instrument is at the proper address. This also confirms that the proper terminating characters have been set for each instrument. The id? query may be used for this purpose.
- Serial poll power-on SRQs until all are processed.
- Lock out the front-panel control of all instruments on the bus.

NOTE. *The VM700T automatically comes up ready for remote operation when it is configured for GPIB operation. However, it remains in local until it is addressed by the GPIB controller. At that time, the VM700T goes into remote. In remote, its front panel is not totally locked out; the CONFIGURE button is still active. Pressing it returns the VM700T to local control. It returns to remote operation when a command is sent to it unless the Communications Configuration file is edited to switch off GPIB remote control or to set the address to offline. The VM700T also returns to local control if the Go-to-Local command is sent to its address.*

- Program each instrument setting to a known condition if it does not power-on to a predetermined state.

NOTE. *The VM700T comes on in the Waveform display at power up and after exiting GPIB diagnostics.*

- Open disk files if needed for data logging or measurement results retrieval.

3. User Interface

   - Prompt the user for operation specific information. This may be as simple as just prompting for a command input to send to the VM700T.

4. Begin the Program

   - Start sending control commands to the instruments on the bus to perform the operations needed for your application.

5. Look for SRQ interrupts.

   - Handle the SRQ interrupts and acquire any data than may be returned by an instrument. This may be just displaying it on the PC screen for the very simple remote control program. The more advanced remote control programs may be storing measurement data to files, doing complicated data checking to determine if some condition needing user interaction or program branching to handle different errors or perform different operations is required.

   - Check for reasonable data from the instrument. This will require a knowledge on the programmer's part about specific variable limits and the expected returns.

6. Data I/O

   - Present measurement results to the user. The very simple programs merely take the ASCII response to a command and displays it for the user. The more complex programs may generate graphs or other visual displays from the received data. Many of the VM700T responses for getting measurement results are tabular ASCII files.

   - Log data to files or hard copies with a date/time stamp for later analysis.

   - Check for violated limits errors and alert the user to those violations.

7. End or Repeat

   - The program should repeat for making continuous automatic measurement.

   - If the program ends, copy any needed files from RAM to a floppy disk or internal hard disk.

   - Close all files to maintain file integrity.

   - Turn off any controlled signal sources such as high voltage sources, generators, and power supplies.

   - Return instrument to local control.

Appendices

Appendix A: Specification

The GPIB interface function set implemented in the VM700T GPIB interface and the capability level is given in Table A-1.

Table A-1: IEEE 488.1 Interface Functions Implemented in the VM700T GPIB Interface Card

Function	Implemented	Notes
Source Handshake	SH1	Complete capability
Acceptor Handshake	AH1	Complete capability
Talker	T6	Basic Talker, Serial Poll, Unaddr if MLA
Talker (extended)	TE0	No capability
Listener	L4B	Basic Listener, Unaddr if MTA
Listener (extended)	LE0	No capability
Service Request	SR1	Complete capability
Remote Local	RL2	No local lock out
Parallel Poll	PP0	No capability
Device Clear	DC1	Complete capability
Device Trigger	DT0	No capability
Controller	C0	No capability
Electrical Interface	E2	Three-state bus drivers

Electrical and Mechanical Specifications

Table A-2: Electrical Characteristics

Characteristic	Performance Requirement	Comments
Output Signal Levels	Logical 0 ≥ +2.0 V Logical 1 3 +0.8 V	IEEE Std 488–1979 para. 3.2
Data Transfer Rate	Greater than 1 kbyte per second	
Connectors		
GPIB	Standard GPIB parallel connector	
VM700T	3 × 50 square pin connector	

Table A-2: Electrical Characteristics (Cont.)

Characteristic	Performance Requirement	Comments
Power Requirements		
+5 V		
Power Consumption	35 W maximum	
Fuse		
+5 Vin	10 A, fast blow	

Table A-3: Mechanical Specifications

Characteristic	Description
Construction	Plug in circuit board. Designed for installation in the VM700T left side. Multilayer ECB; material FR4.
Dimensions	
Length	11 inches (includes board retainer)
Width	7 7/8 inches (includes VM700T connector)
Height	1 plug in unit high (15/16 inch)
Shipping Weight	Approximately 4 lb (includes GPIB board, manual, and packing materials)

Table A-4: Optional Accessories

Item	Description
GPIB Cable	1 m, double-shielded. Tektronix Part No. 012–0991–012 m, double-shielded. Tektronix Part No. 012–0991–003m, double-shielded. Tektronix Part No. 012–0991–02
GURU II+. GPIB User's Resource Utility for the IBM Personal Computer	Software/manual plus National PC2A GPIB controller board; GPIB cable; software on 5 1/4 inch and 3 1/2 inch floppy diskettes. Tektronix Product No. S3FG100

Appendix B: GPIB Diagnostics

The GPIB Diagnostic routine does extensive testing of the interface hardware. Failure of any diagnostic test indicates a hardware problem that requires servicing. The VM700T may function for local operation, but may be faulty for remote operation if a GPIB Diagnostic test fails. Refer to the *VM700T Option 01 (NTSC)/Option 11 (PAL) User Manual* to configure the Diagnostic Test Selection file to either run or skip the GPIB diagnostic on power-up of the VM700T.

Running the Diagnostics

To start the GPIB diagnostics from the VM700T, under the Measure function, select the diagnostics routines by pressing the Diags soft key. Then, select the GPIB~Diagnostics soft key. The initial GPIB Diagnostic Menu is shown in Figure B-1. This diagnostic may be started via the GPIB interface itself (`execute GPIB~Diagnostic`), but this is not recommended. As soon as the diagnostic starts, the interface ceases to talk to the controller. To halt the diagnostic and return to remote control, press one of the front-panel hard keys. Upon exiting the diagnostic, the VM700T reboots and an SRQ is asserted. A serial poll after the reboot returns the power-on status byte (65). There are two run mode choices: Power-up and Interactive.

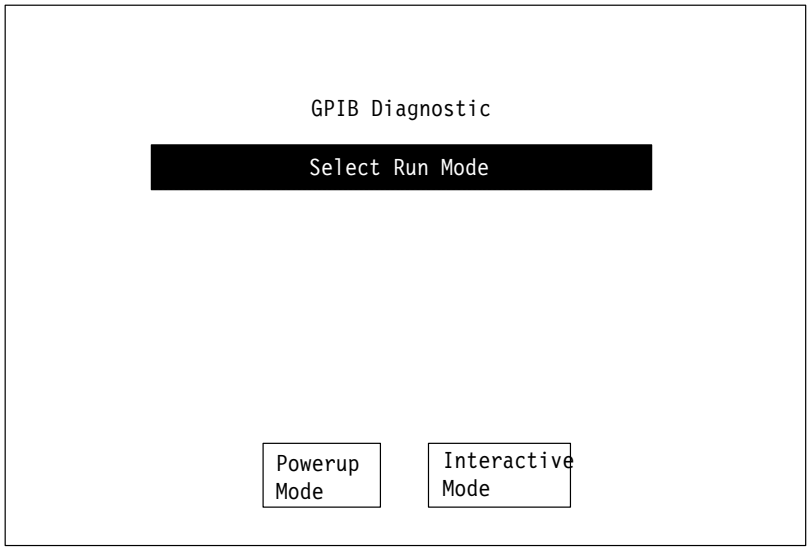

Figure B-1: GPIB diagnostic initial menu

Power-up Mode

GPIB diagnostics may be bypassed or configured to run in normal power-up mode. When configured to run the GPIB diagnostics, all the GPIB tests run completely through, and the display returns to the GPIB Diagnostic menu unless an error is found. In that case, the program halts at the failed main test and shows the test block that failed. All the subtest titles are shown in the test block with any that failed marked with –FAIL–. Passed subtests are marked with –Pass–. You are given the choices of continuing the testing, re-running the failed test, or changing the run mode for the testing program.

***NOTE**. The powerup function key will not run at power on when GPIB is selected as the remote control port.*

Interactive Mode

In interactive mode, each main test block runs and the test halts at the end of each block. The main test title and all the subtest titles are shown with a -Pass- or –FAIL– label. You are given the choices of continuing the testing, looping on the selected test block, re-running the test block, or changing the run mode for the testing program as shown in Figure B-2. Selecting to continue resumes the testing and runs the next main test. This continues until all the test are completed, and then returns to the first menu screen for selecting run mode.

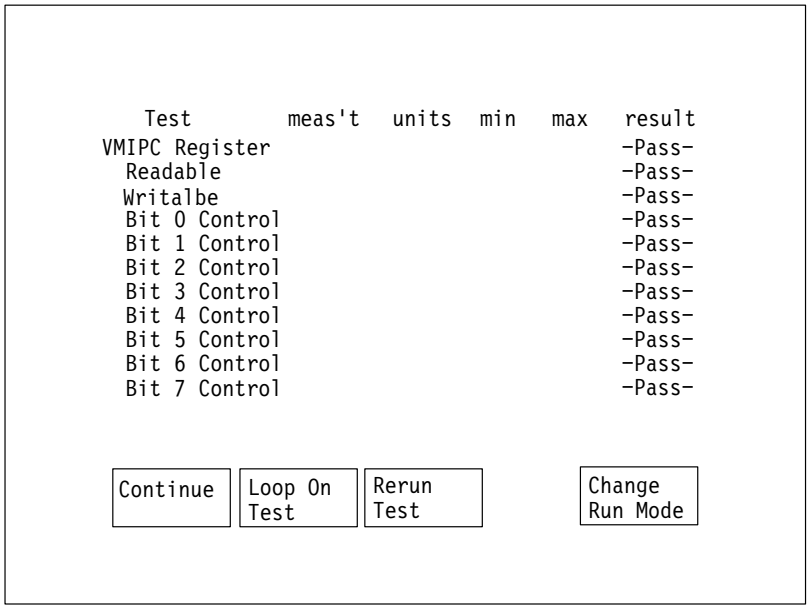

Figure B-2: Interactive run mode menu after running first main test

GPIB Diagnostic Routines

The main test titles are:

- VM IPC Register
- VM Program RAM
- VM Shared RAM
- VM Private RAM
- VM Cen Ctl Reg
- VM Gpib Chip
- GP Program RAM
- GP Shared RAM
- GP Private RAM
- GP Spurious Ints
- GP Int From VM
- GP Clk Tick Int
- GP Get Int

Each of these main tests have a number of subtests as shown in Table B-1. Note that the RAM memory checks look at the installed memory and may be different than that shown depending what your instrument has installed.

Table B-1: GPIB Diagnostic Tests

Test	meas't	units	min	max	result
VM IPC Register ––––––	———	———	———	———	–Pass–
Readable	———	———	———	———	–Pass–
Writable	———	———	———	———	–Pass–
Bit 0 Control	———	———	———	———	–Pass–
Bit 1 Control	———	———	———	———	–Pass–
Bit 2 Control	———	———	———	———	–Pass–
Bit 3 Control	———	———	———	———	–Pass–
Bit 4 Control	———	———	———	———	–Pass–
Bit 5 Control	———	———	———	———	–Pass–
Bit 6 Control	———	———	———	———	–Pass–

Table B-1: GPIB Diagnostic Tests (Cont.)

Test	meas't	units	min	max	result
VM IPC Register ––––––	———	———	———	———	–Pass–
Bit 7 Control	———	———	———	———	–Pass–
VM Program RAM –––––	———	———	———	———	–Pass–
Decoding	2.000	MB	0.500	2.000	–Pass–
Segment Size	2.000	MB	0.500	2.000	–Pass–
Address Lines	———	———	———	———	–Pass–
Data Lines	———	———	———	———	–Pass–
Valid Memory	2.000	MB	0.500	2.000	–Pass–
Write Protect	———	———	———	———	–Pass–
Addr Lines DEBUG	———	———	———	———	–Pass–
Data Lines DEBUG	———	———	———	———	–Pass–
DEBUG jmpr removed	———	———	———	———	–Pass–
VM Shared RAM ––––––	———	———	———	———	–Pass–
Decoding	1.000	MB	0.250	1.000	–Pass–
Segment Size	1.000	MB	0.250	1.000	–Pass–
Address Lines	———	———	———	———	–Pass–
Data Lines	———	———	———	———	–Pass–
Valid Memory	1.000	MB	0.250	1.000	–Pass–
Write Protect	———	———	———	———	–Pass–
Addr Lines DEBUG	———	———	———	———	–Pass–
Data Lines DEBUG	———	———	———	———	–Pass–
VM Private RAM ––––––	———	———	———	———	–Pass–
Decoding	1.000	MB	0.250	1.000	–Pass–
Segment Size	1.000	MB	0.250	1.000	–Pass–
Address Lines	———	———	———	———	–Pass–
Data Lines	———	———	———	———	–Pass–
Valid Memory	1.000	MB	0.250	1.000	–Pass–
VM Cen Ctl Reg ––––––	———	———	———	———	–Pass–
Readable	———	———	———	———	–Pass–
Writable	———	———	———	———	–Pass–
Bit 0 Control	———	———	———	———	–Pass–
Bit 1 Control	———	———	———	———	–Pass–

Table B-1: GPIB Diagnostic Tests (Cont.)

Test	meas't	units	min	max	result
VM Cen Ctl Reg ––––––	———	———	———	———	–Pass–
Bit 2 Control	———	———	———	———	–Pass–
Bit 3 Control	———	———	———	———	–Pass–
Bit 4 Control	———	———	———	———	–Pass–
Bit 5 Control	———	———	———	———	–Pass–
VM Gpib Chip ––––––	———	———	———	———	–Pass–
Rd Int Stat 0 Reg	———	———	———	———	–Pass–
Rd Int Stat 1 Reg	———	———	———	———	–Pass–
Rd Addr Stat Reg	———	———	———	———	–Pass–
Rd Bus Stat Reg	———	———	———	———	–Pass–
Rd Cmd Pass Reg	———	———	———	———	–Pass–
Rd Data In Reg	———	———	———	———	–Pass–
Wr Int Mask 0 Reg	———	———	———	———	–Pass–
Wr Int Mask 1 Reg	———	———	———	———	–Pass–
Wr Aux Cmd Reg	———	———	———	———	–Pass–
Wr Address Reg	———	———	———	———	–Pass–
Wr Ser Poll Reg	———	———	———	———	–Pass–
Wr Par Poll Reg	———	———	———	———	–Pass–
Wr Data Out Reg	———	———	———	———	–Pass–
GP Program RAM ––––	———	———	———	———	–Pass–
Address Lines	———	———	———	———	–Pass–
Data Lines	———	———	———	———	–Pass–
GP Shared RAM ––––––	———	———	———	———	–Pass–
Address Lines	———	———	———	———	–Pass–
Data Lines	———	———	———	———	–Pass–
GP Private RAM ––––––	———	———	———	———	–Pass–
Address Lines	———	———	———	———	–Pass–
Data Lines	———	———	———	———	–Pass–
GP Spurious Ints –––––	———	———	———	———	–Pass–
Level 7 (NMI)	———	———	———	———	–Pass–
Level 6	———	———	———	———	–Pass–
Level 5	———	———	———	———	–Pass–

Table B-1: GPIB Diagnostic Tests (Cont.)

Test	meas't	units	min	max	result
GP Spurious Ints ––––––	———	———	———	———	–Pass–
Level 4	———	———	———	———	–Pass–
Level 3	———	———	———	———	–Pass–
Level 2	———	———	———	———	–Pass–
Level 1	———	———	———	———	–Pass–
GP Int From VM ––––––	———	———	———	———	–Pass–
Int Occurs	———	———	———	———	–Pass–
Int Clears	———	———	———	———	–Pass–
Int Disables	———	———	———	———	–Pass–
GP Clk Tick Int ––––––	———	———	———	———	–Pass–
Int Occurs	———	———	———	———	–Pass–
Int Clears	———	———	———	———	–Pass–
Int Disables	———	———	———	———	–Pass–
GP Get Int ––––––	———	———	———	———	–Pass–
Int Occurs	———	———	———	———	–Pass–
Int Clears	———	———	———	———	–Pass–
Int Disables	———	———	———	———	–Pass–

Index

Index

Symbols

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

V

W