Reference Manual

Tektronix

VM700A Video Measurement Set Programmer’s Reference Manual

070-8550-01

For further information, contact: Tektronix, Inc., Corporate Offices, P.O. Box 1000, Wilsonville, OR 97070-1000. Phone: (503) 627-7111; TLX: 192825; TWX: (910) 467-8708; Cable: TEKWSGT.

TABLE OF CONTENTS

Table of Contents

Operator's Safety Summary

Section 1 Introduction

Section 2 Connecting the PC and the VM700A

Section 3 Functions

Section 4 Simple Remote Control

Section 5 Computer-Based Remote Control

Section 6 Miscellaneous Topics

Appendix A VM700A Commands

Appendix B Get/Set Keywords

Appendix C Measurement Results Files

LIST OF ILLUSTRATIONS

Figure **Page**

LIST OF TABLES

Table **Page**

Table Page

OPERATOR'S SAFETY SUMMARY

The general safety information in this part of the summary is for both operating and servicing personnel. Specific warnings and cautions will be found throughout the manual where they apply, but may not appear in this summary.

Terms In This Manual

CAUTION statements identify conditions or practices that could result in damage to the equipment or other property or loss of data.

WARNING statements identify conditions or practices that could result in personal injury or loss of life.

Terms As Marked on Equipment

⚠ **CAUTION** indicates a personal injury hazard not immediately accessible as one reads the marking, or a hazard to property, including the equipment itself. Refer to the manual for information.

⚡ **DANGER** indicates a personal injury hazard immediately accessible as one reads the marking.

⏚ Protective ground (earth) terminal.

SAFETY INFORMATION

Use the Proper Power Source. This product is intended to operate from a power source that will not apply more than 250 V rms between the supply conductors or between either supply conductor and ground. A protective-ground connection by way of the grounding conductor in the power cord is essential for safe operation.

Ground the Product. This product is grounded through the grounding conductor of the power cord. To avoid electrical shock, plug the power cord into a properly wired receptacle before connecting to the product input or output terminals. A protective-ground connection by way of the grounding conductor in the power cord is essential for safe operation.

Danger May Arise From Loss of Ground. Upon loss of the protective-ground connection, all accessible conductive parts (including knobs and controls that may appear to be insulating) can render an electric shock.

Use the Proper Fuse. To avoid fire hazard, use only the fuse of correct type, voltage rating, and current rating as specified in the parts list for your product. Refer fuse replacement to qualified service personnel.

Do Not Operate in Explosive Atmospheres. To avoid explosion, do not operate this product in an explosive atmosphere.

Do Not Remove Covers. To avoid personal injury, do not remove the product covers or panels. Do not operate the product without the covers and panels properly installed.

Section 1
INTRODUCTION

With its various modes and measurements available in a single instrument, the VM700A is a great tool for enhancing measurement productivity. Learning to use the VM700A's programming capability can make it even more productive.

For instance, by defining a function on the VM700A, you can make it execute repetitive sequences of commands, either from the front panel or under computer control.

By connecting the VM700A to a terminal, an IBM PC, or a PC-compatible computer, you can perform any operation available from the front panel, and more. Additional capabilities include logging the commands sent to and the data returned by the VM700A, and branching to different parts of a program based upon earlier results.

This manual describes the VM700A's programming capabilities. The manual is organized as follows:

> Section 2, "Connecting the PC and the VM700A," tells you how to connect a PC and a VM700A to allow them to talk to each other.
>
> Section 3, "Functions," tells you how to define, edit, execute, and delete VM700A functions from the front panel.
>
> Section 4, "Simple Remote Control," tells you how to control the VM700A remotely in an interactive session, typing in one command at a time.
>
> Section 5, "Computer-Based Remote Control," tells you how to write programs to control the VM700A without human intervention. A sample program that reads a file of VM700A commands, executes them, and logs both commands and data to a file is included. (A computer, not just a terminal, is required to perform the operations described in this section.)
>
> Appendix A, "VM700A Commands," describes the VM700A commands available in functions and remote operation and gives examples of their use.
>
> Appendix B, "GET/SET Keywords," lists and describes the keywords that can be used with the VM700A **get** and **set** commands.
>
> Appendix C, "Getting Measurement Results," shows the format of the results files produced by each VM700A measurement.
>
> Appendix D, "Error Messages," lists the error and warning messages produced by function playback and remote operation.

RELATED PROGRAMS

Tektronix sells the following programs for use with the VM700A (all require an IBM PC or compatible computer):

- VMT: The VMT program facilitates communication between PC's/compatibles and VM700/VM700A measurement sets. It features a menu-driven interface for common operations, as well as extra commands to add decision making and conditional branching capabilities.

- VMBKUP: This program package actually consists of three programs, VMFTP, VMTERM, and VMBACKIT, that transfer files between the VM700A and an IBM PC or compatible. VMFTP transfers one file at a time, using the familiar FTP protocol. VMTERM is a terminal emulator program that you can use to control the VM700A remotely. VMBACKIT transfers several files at a time, and is useful primarily for backing up file sets from the VM700A onto a PC.
- VMREMGR: The VMREMGR program displays VM700A graphics on a VGA screen.

See your Tektronix representative for information about any of these programs.

Section 2
CONNECTING THE PC AND THE VM700A

This section discusses the wiring necessary to connect the VM700A to a computer, terminal, or modem. Successful communication also requires correct settings for Communications Setup file parameters. Setting the Communications Setup file parameters is discussed in Section 4, "Simple Remote Control."

The rear panel of the VM700A features three video loop-through inputs, one external sync loop-through, and two RS-232-C DTE ports (Figure 2-1).

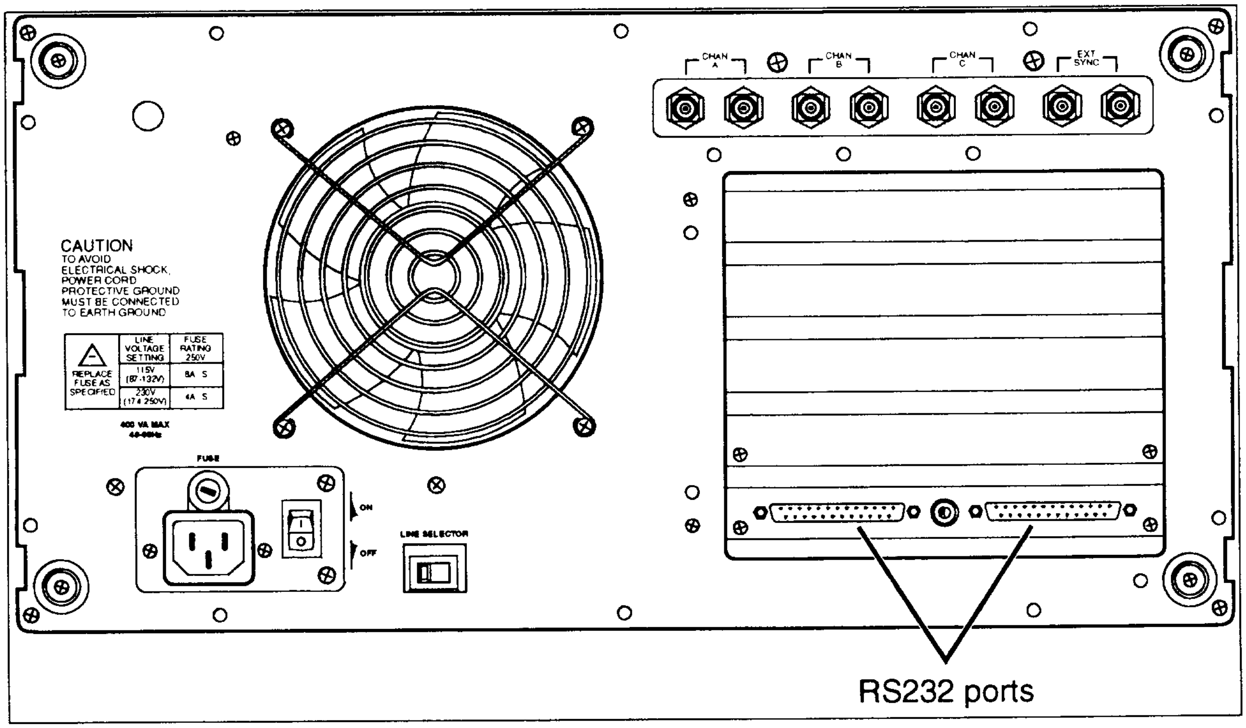


Figure 2-1. VM700A Rear Panel.

The VM700A's two RS-232C DTE ports are used for remote operation via a modem or direct serial connection, as well as to output graphics and reports to a printer. Both connectors are located on the rear panel of the instrument. Both are 25-pin male "D" connectors.

Table 2-1 lists the input and output signals used by the RS-232-C ports.

Table 2-1
Signals Used by VM700A RS-232-C Ports

Pin	Signal	Notes
1	Frame Ground	Required for modem connections only.
2	TD (Transmit Data)	The VM700A transmits data on this pin.
3	RD (Receive Data)	The VM700A receives data on this pin.
4	RTS (Request To Send)	When the Flow Control item in the Communications Setup file is set to CTS/RTS, RTS on the VM700A connects to the CTS line of a terminal, and is active when the VM700A is ready to receive data. When the VM700A's receive buffers approach capacity, RTS goes inactive, telling the sending device to stop transmitting data. When the Flow Control item in the Communications Setup file is set to None or XON/XOFF, RTS is always active.
5	CTS (Clear To Send)	When the Flow Control item in the Communications Setup file is set to CTS/RTS, the VM700A is enabled to transmit data by a high level on CTS, and is disabled by a low level. It is normally connected to the RTS line of a terminal. When the Flow Control item in the Communications Setup file is set to None or XON/XOFF, the VM700A transmits data regardless of the CTS level.
6	DSR (Data Set Ready)	Required for modem connections only. Data Set Ready is the modem's equivalent of DTR. When DSR is asserted, the PC/terminal knows it is connected to a "live" modem.
7	Signal Ground	—
8	CD (Carrier Detect)	Required for modem connections only. This signal typically comes from a modem and indicates that a connection is made.
20	DTR (Data Terminal Ready)	Required for modem connections only. This pin is always high when the VM700A is turned on. When DTR is asserted, the modem knows it is connected to a "live" PC/terminal.

Table 2-2 and Figure 2-2 show the pin configuration for a cable connecting the VM700A's 25-pin male connector to a 9-pin male connector on a PC or terminal. Note that the table and illustration describe the cable ends, not the ports on the devices.

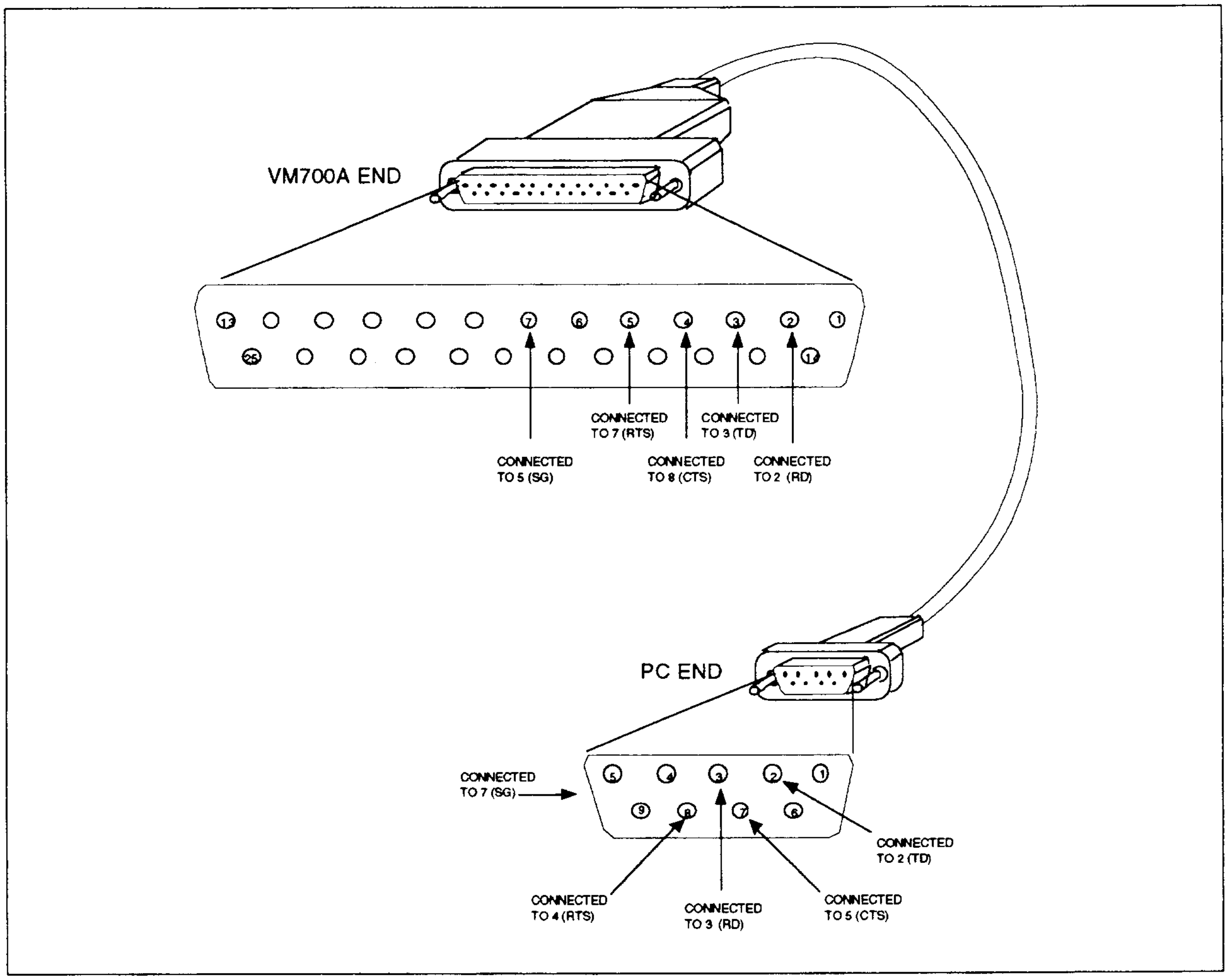


Figure 2-2. Pin Configuration for VM700A 25-Pin to PC 9-Pin Cable.

Table 2-2
Pin Configuration for VM700A 25-Pin to PC 9-Pin Cable

VM700A Cable End (25-Pin Female)	Flow	PC Cable End (9-Pin Female)
2 - Transmit Data (TD)	⇒	2 - Receive Data (RD)
3 - Receive Data (RD)	⇐	3 - Transmit Data (TD)
7 - Signal Ground (SG)		5 - Signal Ground (SG)
5 - Clear to Send (CTS)	⇐	7 - Ready to Send (RTS)
4 - Ready to Send (RTS)	⇒	8 - Clear to Send (CTS)

Table 2-3 and Figure 2-3 show the pin configuration for a cable connecting the VM700A's 25-pin male connector to a 25-pin connector on a PC or terminal. Note that the table and illustration describe the cable ends, not the ports on the devices.

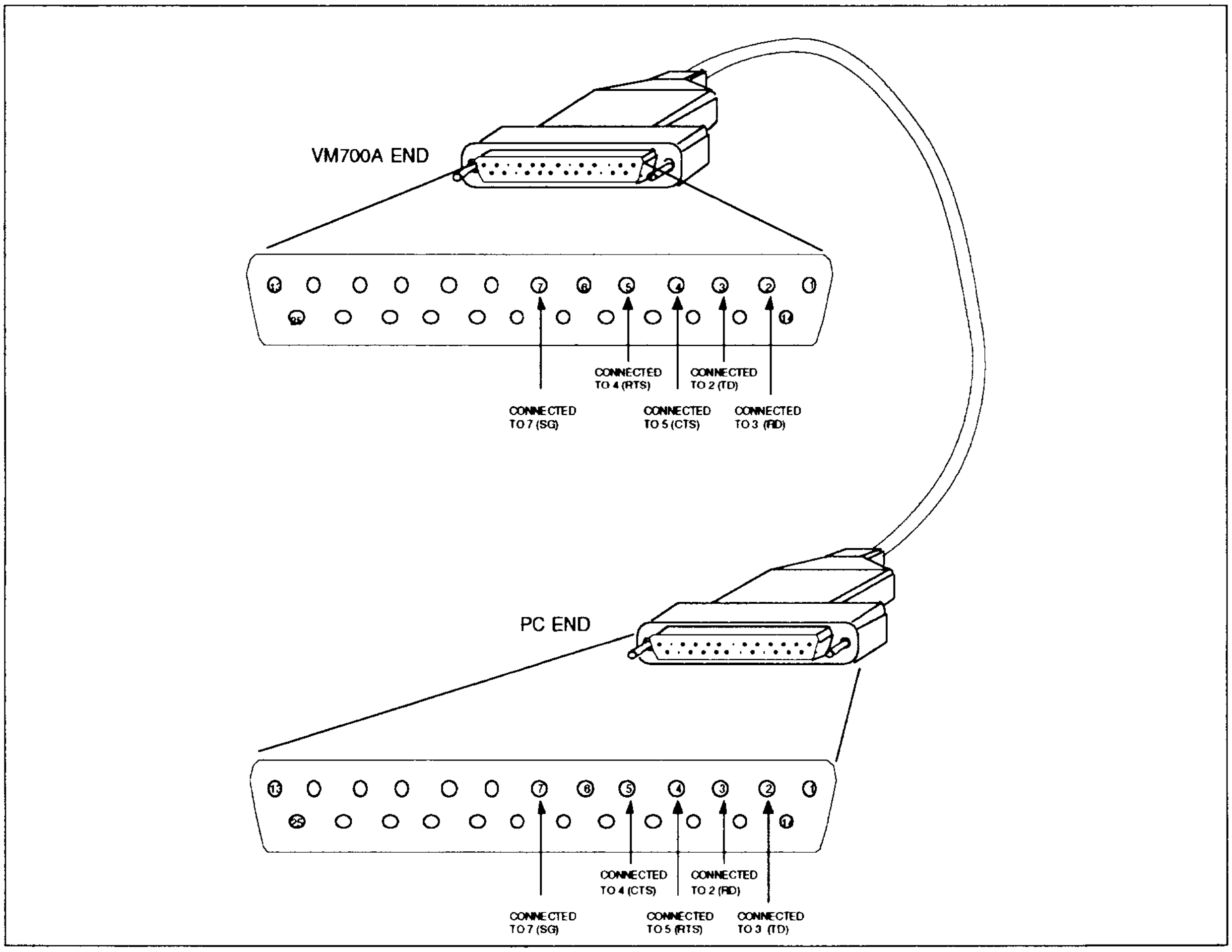


Figure 2-3. Pin Configuration for VM700A 25-Pin to PC 25-Pin Cable.

Table 2-3
Pin Configuration for VM700A 25-Pin to PC 25-Pin Cable

VM700A Cable End (25-Pin Female)	Flow	PC Cable End (25-Pin Female)
2 - Transmit Data (TD)	⇒	3 - Receive Data (RD)
3 - Receive Data (RD)	⇐	2 - Transmit Data (TD)
7 - Signal Ground (SG)		7 - Signal Ground (SG)
5 - Clear to Send (CTS)	⇐	4 - Ready to Send (RTS)
4 - Ready to Send (RTS)	⇒	5 - Clear to Send (CTS)

Table 2-4 and Figure 2-4 show the pin configuration for a cable connecting the VM700A's 25-pin male connector to a 25-pin connector on a modem. Once again, the table and illustration describe the cable ends, not the ports on the devices.

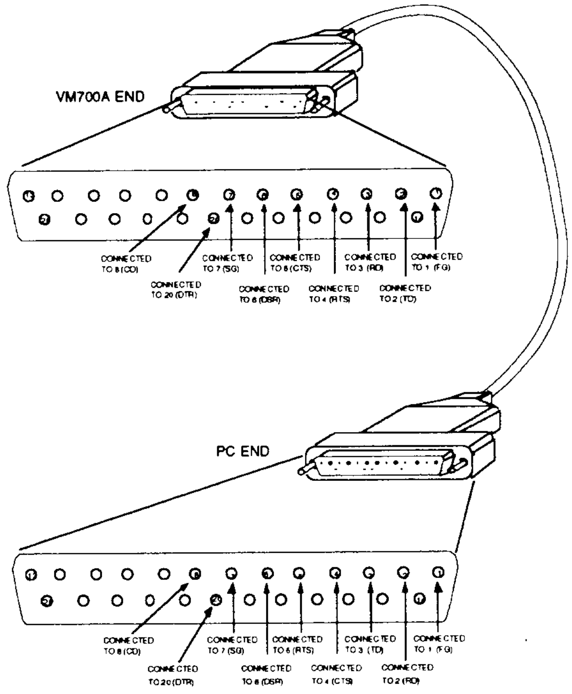


Figure 2-4. Pin Configuration for VM700A 25-Pin to Modem 25-Pin Cable.

Table 2-4
Pin Configuration for VM700A 25-Pin to Modem 25-Pin Cable

VM700A Cable End (25-Pin Female)	Flow	Modem Cable End (25-Pin Female)
1 - Frame Ground (FG)		1 - Frame Ground (FG)
2 - Transmit Data (TD)	⇒	2 - Transmit Data (TD)
3 - Receive Data (RD)	⇐	3 - Receive Data (RD)
4 - Ready to Send (RTS)	⇒	4 - Ready to Send (RTS)
5 - Clear to Send (CTS)	⇐	5 - Clear to Send (CTS)
6 - Data Set Ready (DSR)	⇐	6 - Data Set Ready (DSR)
7 - Signal Ground (SG)		7 - Signal Ground (SG)
8 - Carrier Detect (CD)	⇐	8 - Carrier Detect (CD)
20 - Data Terminal Ready (DTR)	⇒	20 - Data Terminal Ready (DTR)

Section 3
FUNCTIONS

VM700A functions are user-defined sequences of commands. Functions can be recorded as a sequence of front-panel actions, or edited with the VM700A's on-screen function editor.

Once a function is stored in the VM700A, it can be executed from the front panel by pressing the Function button, followed by touching the softkey corresponding to the function. Functions can also be executed from remote operation.

Functions can be used to execute a sequence of commands repeatedly and reliably.

They are also useful as programming aids. Sometimes, the best way to find out a command sequence to send over remote control is to record a function that performs the same action, then examine the function to see how it works.

Lastly, functions can also be used as "building blocks" to be called from within remote programs (or other functions). Programs are much smaller and easier to write if they can simply play back a sequence of pre-recorded functions.

THE FUNCTION KEYS DIRECTORY

The Function Keys directory (/nvram0/FunctionKeys) contains user-defined functions, the Timed Events directory, and user-created directories that can contain other functions. To display the Function Keys directory, do the following:

1. If the VM700A is in Configure mode, press the Configure button to exit it.
2. Press the Configure button.
3. Touch the Function Keys softkey. This displays the FunctionKeys directory and the Function Keys menu (Figure 3-1).

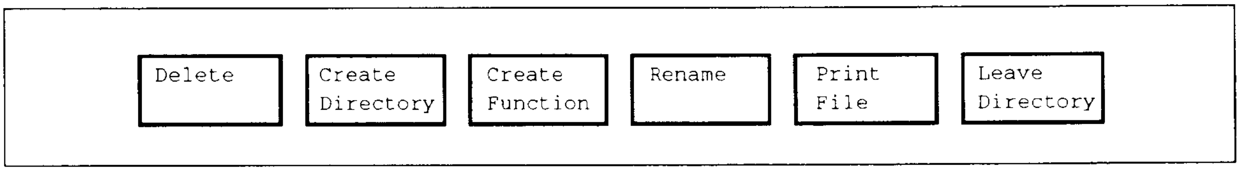
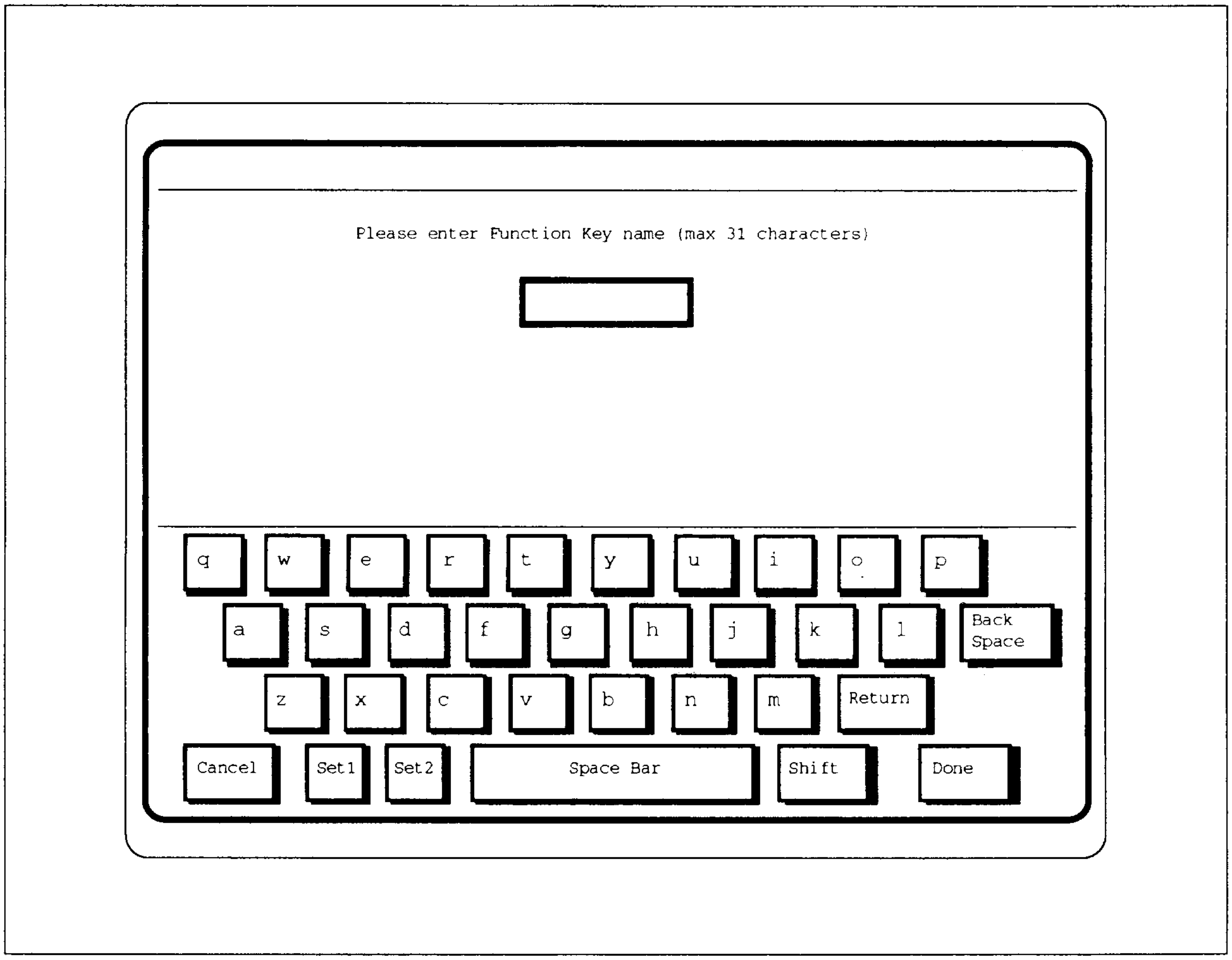


Figure 3-1. Function Keys menu.

You can use the Function Keys directory window and the main Function Keys menu to create functions, view functions, edit functions, print functions, delete functions, create sub-directories, delete sub-directories, and rename functions or sub-directories. The rest of this chapter describes each of these operations.

CREATING FUNCTIONS

To create a new function from the front panel, bring up the Function Keys directory and main Function Keys menu. Then, touch the Create Function softkey. This brings up an on-screen keyboard (Figure 3-2), with which you can type in the new function's name.



Please enter Function Key name (max 31 characters)

Figure 3-2. VM700A on-screen keyboard.

When you name the function to be created, keep the following rules in mind:

1. No spaces are allowed in the name; use an _ (underline) or . (dot) to separate words in a name.
2. Use only upper and lower case letters, numerals, and the following special characters: _ (underline), . (dot), - (minus sign), + (plus sign, : (colon), and ~ (tilde) in names.
3. Use the Return key to place a word on the second line of the name. The '~' key also places a word on the second line of the name. For example, typing "Transmitter~Functions" would place the word "Functions" below "Transmitter" when the function's icon appears in the Function Keys directory.
4. A maximum of 31 characters are allowed in a name.

5. The Special character key and the Shift key "lock" when selected. To unlock them, touch the key a second time.
6. After typing the name, touch Done.
7. Press Cancel to quit the process without creating the function.

If you entered a name for the function, the VM700A goes into "learn" mode. The LED on the Configure button blinks while to indicate this. In learn mode, the VM700A records all user interactions with the front panel. The VM700A displays the Learn Mode menu (Figure 3-3), containing softkeys used in recording functions. The learn mode softkeys are described below.

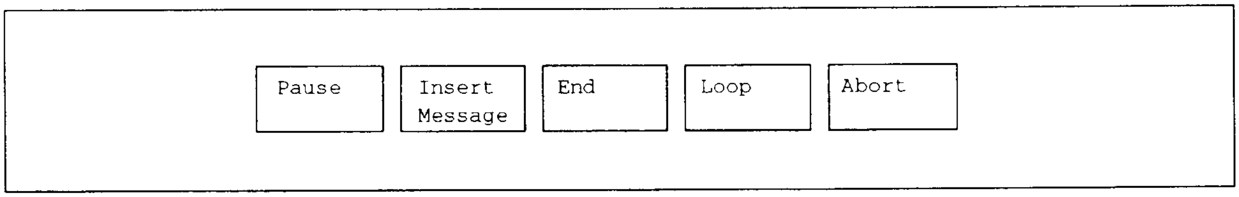


Figure 3-3. Learn Mode menu.

Pause Selecting the Pause softkey suspends function recording. This allows you to press VM700A controls without making those actions part of the recorded function. (The subsequent state of the VM700A, however, IS recorded.) When you want to resume function recording, press the Configure button, touch Pause a second time, and press the Configure button again.

Insert Message Selecting the Insert Message softkey displays the on-screen keyboard, with which you can enter a message of up to four lines of 76 characters each. When the function plays back, the message will be displayed in the lower left corner of the screen at the appropriate point in the playback sequence. When the message appears, function playback pauses and the message **Press screen to continue** is also displayed. Touch any point on the screen to continue function playback.

End Selecting the End softkey terminates function recording. The VM700A remains in the Function Keys directory in Configure mode. When executed, a Function Key terminated with End plays back once and stops.

Loop Selecting the Loop softkey terminates function recording. The VM700A remains in the Function Keys directory in Configure mode. When executed, a Function Key terminated with Loop plays back continuously until halted by pressing the Function button again.

Abort Selecting the Abort softkey during function recording deletes the function you were creating.

EDITING FUNCTIONS

To edit a function once it has been learned, bring up the Function Keys directory and main Function Keys menu. Then, touch the icon corresponding to the function you wish to edit. This brings up the Function Editing menu (Figure 3-4).

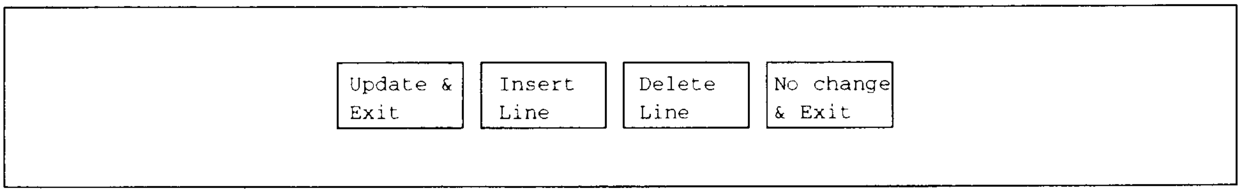


Figure 3-4. Function Editing menu.

To edit a line:

1. Highlight the line to be edited, using the control knob (i.e., rotate the knob until the line you wish to edit appears in the center of the screen, brighter than the surrounding lines).
2. Touch the line. The on-screen keyboard appears.
3. Use the knob, the backspace key, and other keys of the on-screen keyboard to modify the line.
4. Touch "Done" when you have finished editing the line.

To insert a line:

1. Highlight the line following the insertion point, using the control knob (i.e., rotate the knob until the line after the insertion point appears in the center of the screen, brighter than the surrounding lines).
2. Touch the Insert Line softkey. The on-screen keyboard appears.
3. Enter the new line, using the on-screen keyboard.
4. Touch "Done" when the new line is completed.

To delete a line:

1. Highlight the line to be deleted, using the control knob (i.e., rotate the knob until the line to be deleted appears in the center of the screen, brighter than the surrounding lines).
2. Touch the "Delete Line" softkey.

To save changes and exit:

Touch the "Update & Exit" softkey.

To exit without saving changes:

1. Touch the "No Change & Exit" softkey.
2. If you have made changes to the function, a message appears on the screen, warning you that the function has been changed.

- If you really don't want to save the changes, touch the "No Change & Exit" softkey again.
- If you want to save the changes instead, touch the "Update & Exit" softkey.

Adding Comments to a Function

You can add comments to a function by preceding the comment with a "#" (pound sign) character. Any characters appearing on a line after a "#" character are ignored.

Annotating a Function

You can "annotate" a function (i.e., add descriptive text explaining what the function does) with up to three lines of text. The annotation text appears when you press the "Function" button on the front panel.

To annotate a function:

1. Bring up the function for editing (i.e., from the Function Keys directory, touch the icon corresponding to the function to be edited).
2. Insert a line at the beginning of the function, by highlighting the first line of the function and touching the "Insert Line" softkey.
3. Type the text of the annotation line, making sure that the first character is a "#" (pound sign) character.
4. Repeat steps 2 and 3, if desired, to insert up to three annotation lines.
5. Touch the "Update & Exit" softkey to save changes to the function.

The annotation text (i.e., the first one, two, or three lines of comments, inserted as defined above) will appear next to the icon for the function when you press the Function button on the front panel.

Speeding Up or Slowing Down a Function

When you record a function from the front panel, it plays back EXACTLY the way you recorded it—including the delay time between actions. This time is recorded in the form of **delay** commands that appear in the function. The argument following each **delay** command is the amount of time, in tenths of seconds, for the function to pause before continuing execution. Editing the argument following each **delay** command, or deleting the **delay** command altogether, is an easy way to speed up or slow down function playback.

Note, however, that some measurements take a certain amount of time to set up before their results become valid (or, in some cases, before they can obtain results at all). Therefore, it is a good idea to reduce, but not eliminate, the delay time following most **execute** or **appstart** commands.

Another useful way to modify the timing of a function is to change **knob** commands to **cknob** commands. While **knob** commands only specify an amount by which to turn the knob, **cknob** commands specify both an amount to turn the knob and a time over which to turn it continuously. Thus, while **knob** commands execute near-instantaneously, **cknob** commands execute over a user-specified period of time (given in tenths of seconds). **cknob** commands are most useful when you want to watch the progress of the function while you are turning the knob. **knob** commands, conversely, are most useful when you simply want to turn the knob, then observe the result or perform another action.

FUNCTION PLAYBACK

Once recorded, functions can be played back from the front panel or from remote operation. The following sections describe both kinds of function playback operation.

Function Playback from the Front Panel

To play back a function from the VM700A Front Panel:

1. Press the Function button.

2. Touch the icon corresponding to the function you wish to play back.

The VM700A performs the sequence of actions you recorded, taking almost exactly the same amount of time as when it was recorded. (See “Speeding Up or Slowing Down a Function,” earlier in this chapter, if you want to modify the function's timing.) The Function button's LED flashes while a function is being played back. Pressing the Function button during function playback cancels the playback, and leaves the VM700A in the state it was in when function playback was cancelled.

If there is a playback error, or if a **disptext** command is encountered, an appropriate message is displayed on the bottom of the VM700A screen, along with a request that the screen be touched to continue playback.

When you execute a function terminated with End, the function plays back once and exits. When you execute a function terminated with Loop, the function loops continuously until the Function button is pressed to cancel playback.

The **restoreconfig** command is added by default to functions create with the “Create Function” softkey. This command restores various VM700A operating parameters to the values contained in the configuration files. If you wish, you may delete this command when editing the function.

When you abort a function, a **restoreconfig** command is always executed automatically.

Function Playback from Remote Operation

To play back a function from remote operation, send the **playback** command:

```
VM700A> playback function-name
```

If the function is in a sub-directory, the directory name must be included in the argument to the **playback** command. For example, if the function is named TransDGDP and is stored in directory Special~Measurements within the Function Keys directory, the **playback** command would be:

```
VM700A> playback Special~Measurements/TransDGDP
```

When a function is playing back remotely, it can be terminated by pressing Ctrl-C on the remote keyboard, or by pressing the Configure button on the front panel (which also terminates the remote control session).

If the “Message Display:” line in the Communication~Setup file is set to “VM700 Screen”, a message appears when the **playback** command is first issued, telling you that function playback is in progress and how to stop it:

```
VM700A> playback Special~Measurements/TransDGDP
Function playback in progress. Enter ^C to stop it.
```

If the function finishes normally, the following message appears:

```
Function playback completed
```

If function playback is terminated with a Ctrl-C, the following message appears:

```
Function playback terminated
```

If there is a playback error, or if a **disptext** command is encountered, an appropriate message is displayed.

If the Message Display line in the Communications~Setup file is set to **VM700 Screen**, error messages or text from **disptext** commands are displayed on the VM700A screen. To continue function playback, touch the screen. If the Message Display line in the Communications~Setup file is set to **Remote**, error messages or text from **disptext** commands are displayed on the terminal screen, followed by the message

```
Hit CR to continue
```

Pressing return on the remote terminal displays the following message:

```
Function playback continues...
```

Function Playback on Power Up

There is a special function file that can be created for use on powering up the VM700A. The "Powerup" function file is looked for by the VM700A as part of the initialization process, and, if found, played back immediately after initialization has finished. The powerup function file is created in exactly the same manner as other function files. If there is a state or measurement mode you want the VM700A to be in or series of measurements that needs to be done after power is either turned on or restored after a power loss, the powerup function key file may be used to perform the required steps.

SUB-DIRECTORIES

If you have many functions defined in the Function Keys directory, it is often convenient to store them in sub-directories. This reduces clutter in the Function Keys directory window, and makes it easier to find the function you want for a specific task.

Creating Sub-Directories

To create a new sub-directory in the Function Keys directory (or in any of its sub-directories), use the following steps:

1. Bring up the Function Keys directory by pressing the Configure button, then touching the Function Keys softkey.

   If you wish to create a sub-directory within an existing sub-directory of the Function Keys directory, touch the icon(s) of the sequence of sub-directories until the VM700A screen displays the sub-directory in which you will create the new sub-directory.

2. Touch the Create Directory softkey. The VM700A displays the on-screen keyboard, with which you can type in the new sub-directory's name. When typing in a sub-directory name, follow the same rules as for typing in function names (listed in "Creating Functions," earlier in this chapter).

Traversing the Directory Hierarchy

You can think of the Function Keys directory and its sub-directories as being organized in a hierarchy, with the Function Keys directory at the top, any sub-directories it contains on the next level below, any sub-directories contained in the sub-directories on the next level below, etc. (See Figure 3-5.)

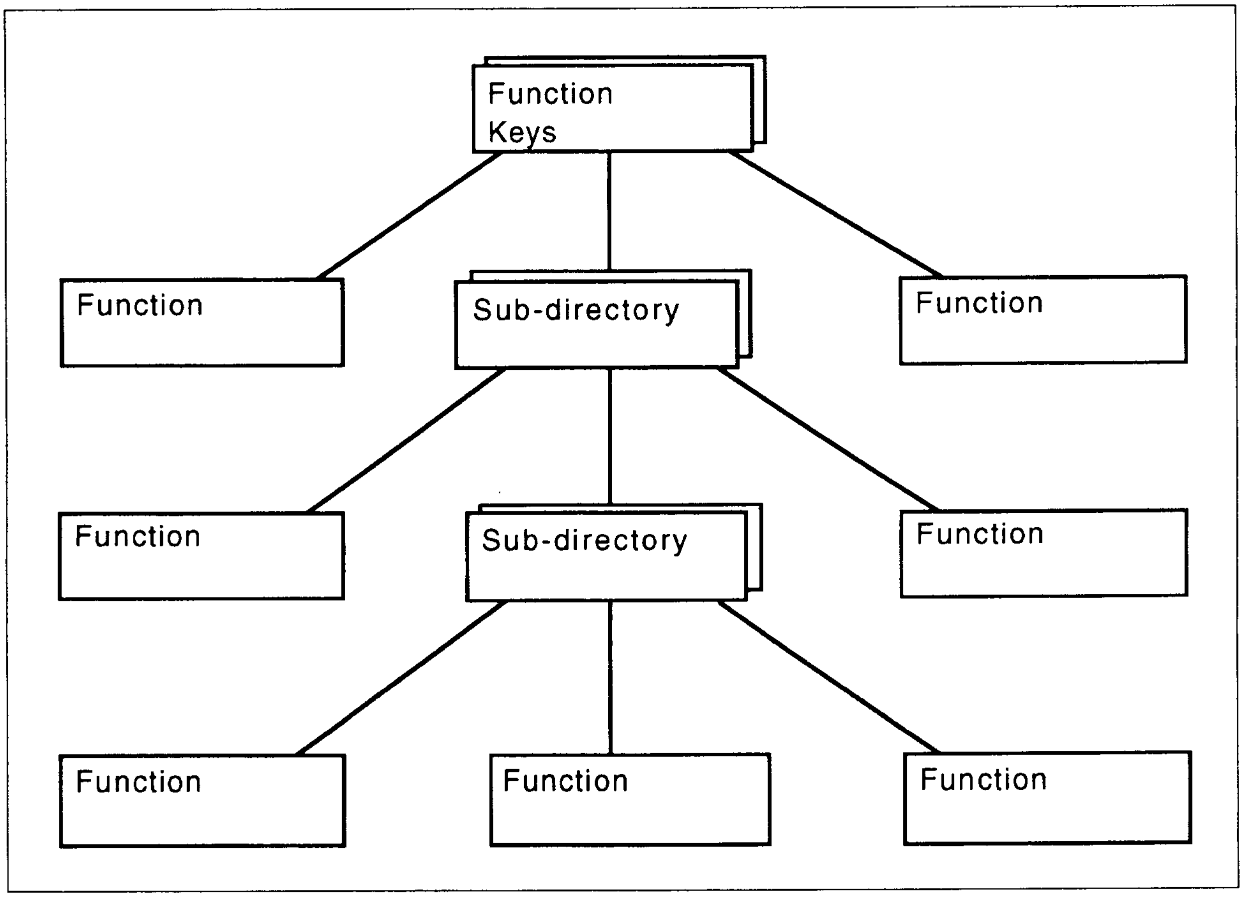


Figure 3-5. Function Keys directory hierarchy.

To travel “down” the hierarchy (i.e., from a directory to one of its sub-directories), touch the icon corresponding to the directory you wish to see.

To travel “up” the hierarchy (i.e., from a sub-directory to the directory the next level above), touch the “Leave Directory” softkey at the bottom right of the screen.

DELETING FUNCTIONS AND SUB-DIRECTORIES

To delete a function or empty sub-directory, bring up the main Function Key menu, then touch the Delete softkey, followed by the icon corresponding to the function or directory. The VM700A beeps at one-second intervals while the selected item “disintegrates” on the screen. Once the disintegration starts, you have six seconds to cancel the Delete command, either by touching the Delete softkey again or by touching the screen anywhere inside the directory window. Canceling the Delete process leaves the function or sub-directory intact (i.e., nothing has been deleted).

A directory that is not empty cannot be deleted. Its icon disintegrates, but reappears after the disintegration process completes.

RENAMING FUNCTIONS AND SUB-DIRECTORIES

To rename a function or sub-directory, bring up the main Function Key menu, then touch the Rename softkey, followed by the icon corresponding to the icon or function to be renamed. The on-screen keyboard appears, allowing you to type the new function or directory name. When typing in a new name, follow the naming rules listed in “Creating Functions,” earlier in this chapter.

PRINTING FUNCTION CONTENTS

To print the contents of a function, bring up the main Function Key menu, then touch the Print File softkey, followed by the icon corresponding to the function to be printed.

You can cancel the print request by entering the main Configure menu (you may have to press the Configure button twice), and then selecting the Cancel Copy softkey.

Section 4
SIMPLE REMOTE CONTROL

This section describes how to control the VM700A remotely, using a terminal emulator program such as Kermit or Telnet. It assumes that you already have a terminal emulator program on your PC, and that you know how to use it.

The section concerns itself with the specific operations required to control the VM700A in a terminal session, where the user types commands into the PC/terminal one at a time. Section 5, "Computer-Based Remote Control," tells how to write programs that control the VM700A without requiring human intervention or supervision.

Remote operation on the VM700A is performed via the serial (RS-232-C) ports on the rear panel. Section 2, "Connecting the PC and the VM700A," discusses the VM700A's RS-232-C port requirements in detail and shows typical cable wiring configurations.

The VM700A has two modes for remote control: "no-protocol" mode and "SLIP" mode. The VM700A is in "no-protocol" mode when the Protocol setting for the Remote Control port in the Communication Setup file has a value of "None". The VM700A is in "SLIP" mode when the "Protocol" setting for the Remote Control port in the Communication Setup file has a value of "SLIP".

In no-protocol mode, the VM700A and the PC/terminal communicate without any special communications protocol. There is simply a cable connection between the VM700A and the PC/terminal, over which commands and data are exchanged. When the VM700A is in no-protocol mode, you can use any commonly available communications program, such as Kermit, ProComm, or Qmodem.

No-protocol mode has two important limitations:

1. In no-protocol mode, there is no control to ensure that information is not corrupted in transit between the VM700A and the PC/terminal.
2. No-protocol mode cannot be used to write files onto the VM700A's non-volatile memory. This limitation ensures that corrupted information will not be written into the VM700A's critical operating files and settings.

Protocol, or SLIP, mode establishes an error-free connection between the VM700A and the PC, through the use of data checking and re-transmission techniques. When the VM700A is in SLIP mode, it lets you write files into its non-volatile memory, by means of programs such as FTP (File Transfer Protocol, for which implementations exist on many platforms, including PC's).

Communicating with the VM700A when it is in SLIP mode requires that you have a program running on the PC that implements SLIP (Serial Line Internet Protocol). Several SLIP implementations are commercially available. The one in use by the VM700A developers at Tektronix is called "PC/TCP," and is produced by FTP Software, Inc.[1] Other packages available for PC's and compatibles include KA9Q/NET,[2] BW?TEL,[3] CMU PC/IP,[4] MIT PC/IP,[5] FUSION,[6] Internet-PC,[7] PC-NFS, [8] and WIN/PC.[9]

If you only need to send commands and receive data from the VM700A and do not need to send files into it, you should consider using Tektronix' VMT program. This program is made to communicate with the VM700A. It incorporates more knowledge of the VM700A than a general-purpose terminal emulator can, and makes the task of remote control of the VM700A considerably easier. To order the VMT program, contact your Tektronix sales representative.

Tektronix also makes available a set of programs called VMBKUP, consisting of the VMFTP program (for one-file-at-a-time transfers) and VMBACKIT (for multiple-file transfers). These programs are highly recommended if your major purpose in transferring files is simply to back up data residing on the VM700A. See your Tektronix sales representative for more details about the VMBKUP software package.

The rest of this section tells you how to perform common VM700A operations using either a no-protocol terminal emulator (e.g., Kermit) or a SLIP-protocol one (e.g., Telnet). Whether the VM700A is in no-protocol mode or SLIP mode, the steps you use to perform most actions during a terminal session are the same. The operations described in this section include:

- how to start a terminal session
- how to read or set configuration parameter values
- how to execute measurements
- how to get measurement results
- how to end a terminal session
- how to transfer files (SLIP mode only)

[1] Contact: FTP Software, Inc., 26 Princess St., Wakefield, MA 01880.
[2] Contact: Phil Karn, KA9Q, 25-B Hillcrest Rd., Warren, NJ 07060.
[3] Contact: Beame & Whiteside Software Ltd., 259 Fiddler's Green Road, Ancaster, Ontario, Canada L9G 1W9.
[4] Contact: Drew D. Perkins, Carnegie Mellon University, 4910 Forbes Ave., Pittsburgh, PA 15213
[5] Contact: M.I.T. Microcomputer Center, Room 11-209, 77 Massachusetts Ave., Cambridge, MA 02139.
[6] Contact: Network Research Corporation, 2380 N. Rose Avenue, Oxnard, CA 93030.
[7] Contact: Sirius Systems, Inc., Box 2202, Petersburg, VA 23804.
[8] Contact: Sun Microsystems, Inc., 2550 Garcia Ave., Mountain View, CA 94043.
[9] Contact: The Wollongong Group, 1129 San Antonio Road, Palo Alto, CA 94303.

STARTING A TERMINAL SESSION

Before you start a terminal session with the VM700A, you should check several things:

1. Verify that a serial port on the PC and a serial port on the VM700A are connected via an RS-232-C cable;
2. Verify that the correct drivers are loaded on the PC, if you intend to use SLIP mode. (Check your CONFIG.SYS file; if you are using PC/TCP, for instance, it should contain two lines that load the SLIP.SYS and IPCUST.SYS drivers, respectively.)
3. Check the settings contained in the Communication Setup file in the /nvram0/ConfigFiles directory. To do so, do the following:
   a. Display the Communications Setup file on the VM700A screen (press the Configure button, touch the Configure Files softkey, touch the icon for the Communications Setup file). Check the following settings:
   b. Remote Control Port should be set to the port that the RS-232-C cable is connected to.
   c. Message Display should be set to **Remote** if you want messages from VM700A functions to be displayed on the PC/terminal, or to **VM700 Screen** if you want messages to be displayed on the VM700A screen.
   d. If you are using no-protocol mode, Non-SLIP Interfacing mode should be set to **Terminal.**
   e. Scroll to the communications parameters (Protocol, Baud Rate, Flow Control, Character Size, Parity, Reset Character, Carrier Detect) for the remote control port (0 or 1).
   f. If you are using no-protocol mode, set Protocol to None, otherwise set it to SLIP.
   g. Set the baud rate as desired.
   h. Set flow control to CTS/RTS for hardware flow control, XON/XOFF for software flow control, whichever your communications program supports. Note that in SLIP mode, **Flow Control** becomes **CTS/RTS** regardless of the setting displayed.
   i. Set Character Size to an appropriate value. (A value of 8 almost always works.) Note that in SLIP mode, **Character Size** becomes **8** regardless of the setting displayed.
   j. Set the Reset Character to "none" on the Remote port.
   k. Set Carrier Detect to "disabled". (This parameter is not currently used, but is reserved for future use.)
4. Start your communications program, and configure it to send commands out the communications port connected to the VM700A.
5. Type the command "remote".

At this point, further presses of the carriage return key should return the VM700A's prompt on a new line. You are now communicating with the VM700A via remote connection.

EXECUTING MEASUREMENTS

The **execute** command starts applications on the VM700A. It takes a single parameter, the name of the application being invoked. When an application name takes two lines to display on the screen, the parameter name is formed using a "~" (tilde) between the two lines. See the description of the **execute** command in Appendix A for a complete list of VM700A applications.

NOTE

New applications may be included as new version of the VM700A are introduced. Press the Measure key to find out what applications are available for your VM700A.

Examples:

```
VM700A> execute Vector
```

starts Vector Mode on the VM700A.

```
VM700A> execute Chrominance~NonLinearity
```

starts the Chrominance~NonLinearity measurement on the VM700A.

GETTING MEASUREMENT RESULTS

To make a measurement and retrieve the results remotely, do the following:

1. start the measurement, using the **execute** command ("execute *application-name*");
2. send the **getresults** command;
3. send the command **show** *application-name*.

execute starts the application. **getresults** takes a "snapshot" of the application's measurements and stores them in a file in the Measurement Results directory. When the results are written in the file, the VM700A sends a message out the control port, consisting of the filename in which the results were stored. This is always the same as the application name.

Each invocation of **getresults** writes a new set of measurement values into the Measurement Results directory. This new set overwrites the previous contents of the Measurement Results file for the current application name. For example, if the H_Timing measurement is running, issuing the **getresults** command writes the current measurement results into a file called H_Timing in the Measurement Results directory. Issuing **getresults** again writes a new set of results into a file with the same name (H_Timing) in the same directory (Measurement Results). The results produced by the first **getresults** command are lost.

You can save results from successive getresults commands by using the **rename** command, or by using logging or file-saving features available with your telecommunications program.

When a measure-mode application is running, getresults takes a single, optional argument. That argument is the keyword **verbose**, which for certain measurements provides a complete list of measured points. The measurements that accept the **verbose** argument are DGDP, Luminance~NonLinearity, Noise~Spectrum, and GroupDelay~SinX_X .

Measure-mode video measurements also write out their results (just as if a **getresults** command was issued) when exiting, i.e., when another measurement or mode is started. Note, however, that while the getresults command always ensures that completed measurement results are written, simply exiting a measurement may not.

In Auto mode, **getresults** can take a list of arguments, each consisting of the four-letter keyword of an Auto-mode measurement. If **getresults** is issued without any arguments, the results of measurements specified by the current selected measurements file are written to the results file.

When a list of measurements is specified as arguments to **getresults**, Auto mode continues to make only the measurements specified in the list until another **getresults** command is sent with a new list, or until the current selected measurements file is reset, or until a **restoreconfig** command is sent. Thus, if you want results from multiple passes of the same measurements list, only the first **getresults** command need supply the measurement list. Subsequent **getresults** commands without arguments will return results from the same list.

show sends the contents of a specified file out the control port. The **show** command takes a single argument, consisting of the name of the file to be sent. If no pathname is supplied in the **show** command's argument, the file is assumed to be a results file, residing in the Measurement Results directory.

Most often, the **show** command is used along with **getresults** to retrieve or view measurement results just written. However, any file can be viewed with the **show** command, by specifying either a relative path from the Measure~Results directory, or an absolute system path.

Examples:

```
VM700A> execute K_Factor
VM700A> getresults
Results in file: K_Factor
VM700A> show K_Factor
Measurement Results Channel A Thu Sep 28 14:28:42
2T Pulse K Factor Waveform->Pulse & Bar
Line = 17
Graticule CCIR-2T
Average Off
---------------------------------------------------------------------
K-2T 0.6 % KF
K-PB -0.2 % KF
PB Ratio 99.0 %
HAD 201.0 n sec
---------------------------------------------------------------------
VM700A>
```

This example executes the K_Factor measurement, stores the results in a file called K_Factor in the Measurement Results directory, and displays the results on the PC/terminal screen.

```
VM700A> execute Auto
VM700A> getresults PBAM PTTK PSYA
Results in file: Auto
VM700A>
```

This example puts the VM700A into Auto mode, measures the PAL Bar Amplitude Error, 2T Pulse K-Factor, and Sync Amplitude Error for the current channel, then writes the results into a file named Auto in the Measurement Results directory.

```
VM700A> show /nvram0/ConfigFiles/Auto_Limit~Files/PAL/Studio
```

This example displays a file called Studio from the directory of PAL Auto_Limit Files.

PRINTING FILES

The **print** command sends VM700A files to the Copy port (which is usually attached to a printer). The file is printed in two steps. First, the file is copied into the print spooler directory. Then, the print spooler sends it out to the Copy port. The VM700A prompts for a new command as soon as the file is submitted to the spooler (i.e., the file need not have actually printed out yet).

print formats files in accordance with the Copy Format entry in the Communications Setup file. "Formatting" a file means generating page headers, page ejects, and special printer commands, depending on the printer type.

The **spool** command is similar to **print**, except that it does not format the file. **spool** is used when the page formatting implemented by the **print** command is not wanted. **spool** differs from **show** in that **spool** is available from remote operation or from function playback, while **show** is only available from remote operation. In addition, **spool** and **print** spool the output file to the Copy port, while **show** simply outputs the file to the Remote Control port.

FUNCTION PLAYBACK

Function playback can be started from remote control with the **playback** command. **playback** takes a single argument, the name of the function to play back. If the function is in a subdirectory, the subdirectory name must be included in the argument.

Example:

```
VM700A> playback Special~Measurements/TransDGDP
```

This example plays back a function called TransDGDP, stored in a directory called Special~Measurements in the /nvram0/FunctionKeys directory.

When a function is being played back remotely, it stops before finishing if issued a Ctrl-C, or upon exit from remote control. If remote control is terminated locally (by pressing the Configure button) while remote function playback is in progress, the function playback stops also.

Informational messages are displayed when functions are played back remotely. The location of the message display is controlled by the Remote Control/Message Display entry in the Communications Setup file. If this entry is set to "VM700A Screen", messages appear on the VM700A screen, and you have to touch the screen to continue function playback. If this entry is set to "Remote", messages appear on the PC/terminal screen, and you have to type Return to continue function playback.

Note that when certain commands that normally provide a response (especially **getresults**) are issued from function playback, the response is not seen on the PC/terminal screen. This is true even if the function playback command was issued during remote operation. In such cases, it is important to know EXACTLY what the function is doing, and to store results away periodically for later retrieval.

Example:

Consider the following function, which makes the same Differential Gain video measurement five times, saving the results for retrieval later:

```
execute DGDP
hardkey Menu
softkey ITS_Search # let it find the right line
hardkey Menu
# make 5 passes for retrieval later
getresults
rename DGDP DGDP1
delay 36000 #delay one hour
getresults
rename DGDP DGDP2
delay 36000 #delay one hour
getresults
rename DGDP DGDP3
delay 36000 #delay one hour
getresults
rename DGDP DGDP4
delay 36000 #delay one hour
getresults
rename DGDP DGDP5
```

Playing back this function from remote operation would produce no output on the screen. If the individual commands were issued directly, each **getresults** command would return the string "Results in file: DGDP." By using the **rename** command, however, it is easy to store away results in the Measurement Results directory for later retrieval with the **show** command.

READING/SETTING CONFIGURATION PARAMETERS

Configuration Files

The VM700A reads several files when it is powered up. The Source_Selection Video file specifies a file from the Video_Source Files directory, one for each channel. VM700A units equipped with both Option 01 and Option 11 specify a file from the Video_Source Files directory for each channel and for each standard (NTSC and PAL).

Files from the Video_Source Files directory tell the VM700A what files to read in other directories to get configuration information. These files and their corresponding directories include:

- Auto Limits File: tells what file to read from the Auto_Limit Files directory to get the caution and alarm limits on Auto-mode measurements.
- Measure Limits File: tells what file to read from the Measure_Limit Files directory to get the alarm limits on Measure-mode measurements.
- Measurement Locations File: tells what file to read from the Measurement Locations directory to get the line/position locations of Measure-mode measurements.
- Selected Measurements File: tells what file to read from the Selected_Measurements directory to get the list of selected Auto-mode measurements.
- "Option" directory files (e.g., Component, Teletext, etc.): tells what file to read from the directory given by the option name to get the configuration parameters for that option.

All these files are stored in the VM700A's non-volatile memory. At start-up time, the files are read into random-access memory, where they are referenced as needed when the instrument makes measurements or performs other operations.

Configuration Keywords

Each configuration parameter has a unique keyword, consisting of four characters (capital letters or digits). Some keywords also require that a channel specification be included. For example, to find out the PAL Bar Amplitude Error limits for channel A in Auto mode, you would use the keyword and channel specification PBAM A.

Some keywords do not reflect a value from the configuration files, but instead record a specific part of the instrument state. Examples include:

- GSRC, which specifies the current video source (channel A, B, or C, or some combination thereof),
- GSSR, which specifies whether or not the signal is locked to source, and
- GLN6, which sets or returns the global line number when using the PAL standard. The corresponding keyword for NTSC is GLN5.

You can find out the current setting of any configuration parameter with the **get** command. You can set most configuration parameters with the **set** command. Note that these commands modify only the configuration values kept in the memory of the instrument; the files stored in non-volatile memory are not touched. Therefore, after power to the instrument is cycled, or after the **restoreconfig** command is executed, or after a function key is aborted, the actual configuration of the instrument again reflects the values in the files.

(Some configuration parameters, especially those relating to communications, such as baud rate, character size, etc., cannot be set remotely. Parameters that can only be read but not set are listed as "read-only" parameters in Appendix B.)

Getting Configuration Parameters

You find out the current value of any VM700A configuration parameter by using the **get** command and the parameter's keyword. The syntax of the **get** command is:

get *keyword* [*channel*]

Not all keywords require a channel specification. See Appendix B for a complete list of VM700A keywords, their syntaxes, and their meanings.

Example:

```
VM700A> get PBAM A
-5.00 5.00 -10.00 10.00
VM700A>
```

The command "get PBAM A"returns the lower caution, upper caution, lower alarm, and upper alarm limits in effect for the PAL Bar Amplitude Error Auto measurement on channel A.

Setting Configuration Parameters

You can specify the value of any settable VM700A configuration parameter by using the **set** command, followed by the parameter's keyword, a channel specification (if applicable), and the correct number of data values for that parameter. The syntax of the set command is:

set *keyword* [*channel*] *value* [*value*[*value*...]]

Not all keywords require a channel specification. See Appendix B for a complete list of VM700A keywords, their syntaxes, and their meanings.

Some of the numeric values associated with a keyword might be undefined (represented in the configuration files by a ----- field). To set a value to undefined, use the string "undef". (See examples, below.)

When a keyword has several values associated with it (for example the set of four floating point numbers associated with Auto measurements limits), you can keep a value unchanged by using the string "same". (See examples, below.)

Reading in a new configuration file is like re-setting every one of the parameters named in that file. For example, if you set new values for a PAL Auto measurement limit, and then reset the Auto Limits File name for that channel (keyword: CHAF), you effectively over-write the parameter values you just set. This leads to an important rule: read in con figuration files first, then set individual parameters.

Examples:

```
VM700A> set PBAM A -6. 6. -12. 12.
```

changes the caution and alarm values used for the PAL Bar Amplitude Error Auto measurement on channel A.

```
VM700A> set SPCF PostScript
```

changes the current copy format to PostScript.

```
VM700A> set PSNP A 42.0 undef 42.0 undef
```

sets the lower caution limit and the lower alarm limit for the Signal-to-Noise Periodic measurement to 42.0 db, and sets the upper limits to "undefined."

```
VM700A> set PBAE A same same same 12.0
```

sets the upper alarm limit for Burst Amplitude Error in Auto mode on Channel A to 12 percent, and leaves the others unchanged.

Restoring Configuration Parameters

The **restoreconfig** command re-reads all configuration values from the configuration files.

Note that when you record a function, the VM700A puts many **set** commands at the start of the function file, and puts a **restoreconfig** command at the end. These commands make it possible to reproduce the exact state in which a VM700A function was recorded when the function is played back. Upon function completion, the restore config command returns the VM700A to its power-up state. (Be warned, however, that any user-specified parameter settings not stored in a file when the **restoreconfig** command is executed are lost.)

Most of the keywords set at the beginning of a function are not associated with the configuration files, but with the current global state of the instrument: values like GLN5/GLN6 (global line number), GACP (clamp position in microseconds for channel A), and GCCC (clamp coupling value for channel C). These keywords are not included in any configuration file, thus there is no concept of restoring these values.

There are three configuration values recorded at the beginning of a function, though, that are normally specified from configuration files. These are the names of the video source files for each channel (VNCA, VNCB, and VNCC for NTSC; VSCA, VSCB, and VSCC for PAL). The trailing **restoreconfig** command undoes the effect of these final three **set** commands when the function is finished playing back. If the **restoreconfig** command is deleted from the function, the state of the VM700A remains as it was when the function key was finished being recorded, which might be different than what is currently in the NVRAM files.

Getting the System Clock Time

The **getclock** command returns the VM700A's system date and time. **getclock** does not take any parameters.

Example:

```
VM700A> getclock
Sep 20 11:03:04 1989
```

Setting the System Clock Time

The **setclock** command sets the VM700A's system date and time. **setclock** takes a single parameter in the same format the instrument returns it as a result of a **getclock** command.

Example:

```
VM700A> setclock Sep 20 20:04:00 1991
```

ENDING A TERMINAL SESSION

To end a terminal session, type **quit** or **exit** to break the remote connection with the VM700A. Then, exit your telecommunications program.

TRANSFERRING FILES (SLIP MODE ONLY)

The VM700A must be in SLIP mode to allow you to transfer files into it. This section tells you how to put the VM700A into SLIP mode, and how to use the **telnet** and **ftp** programs to transfer files to the VM700A.

If your major purpose in transferring files is simply to back up data residing on the VM700A, or to restore files originally backed up from the VM700A, you may wish to consider a set of programs that Tektronix makes available called VMBKUP. VMBKUP consists of the VMFTP program used for one-file-at-a-time transfers, and VMBACKIT used for backing up multiple files. See your Tektronix sales representative for more details about the VMBKUP software package.

Putting the VM700A into SLIP Mode

Use the following steps to put the VM700A into SLIP mode from the front panel:

1. Press the Configure button
2. Touch the Configure Files softkey
3. Turn the knob until the icon for the Communication Setup file appears in the directory window
4. Touch the icon for the Communication Setup file
5. Turn the knob until you can read the Port entry for the Remote Control parameter group. It will read None, Serial Port 0, or Serial Port 1. If it is already set to the serial port that the PC is connected to, skip to step 10. Otherwise, continue with the next step.
6. Turn the knob until the "Port" line in the Remote Control parameter group is highlighted.
7. Touch the line on the screen. A box appears around the highlighted line.
8. Turn the knob until the desired value appears.
9. Touch the line on the screen again, or touch the "Accept Input" softkey.
10. Turn the knob until you can read the Protocol entry in the "Port" parameter group for the Remote Control port. If it is already set to SLIP, you're done. Touch the "Update & Exit" softkey to exit the editing session. Otherwise, continue with the next step.
11. Turn the knob until the "Protocol" line in the "Port" parameter group for the Remote Control port is highlighted.
12. Touch the line on the screen. A box appears around the highlighted line.
13. Turn the knob until the value "SLIP" appears.
14. Touch the line on the screen again, or touch the "Accept Input" softkey.
15. Touch the Update & Exit softkey to exit the editing session.

Starting ftp

FTP (File Transfer Protocol) transfers files from one computer to another, using the TCP/IP protocol.

FTP connections take place between a *server* and a *client*. The server listens on a TCP/IP logical port for connection requests. The client issues a connection request to a waiting server, and a connection is made during which files can be transferred.

File transfers are managed between FTP participants through commands from the client and responses from the server. There are commands available to send and receive files, to change the server's working directory, to get directory listings, to create new directories, and to remove directories and files.

The VM700A implements the FTP protocol in such a way that it can act as an FTP server, while a computer connected to it can act as an FTP client. In order to use FTP to access the file system of the VM700A, you must have an FTP client program running on the computer connected to the VM700A. In addition, the FTP program must be running on the VM700A. You can use FTP on any serial port configured for SLIP protocol.

To start the ftp program on the VM700A from the front panel:

1. press the Measure button.
2. go to the /nvram0/Executable~Files/Instrument~Operations directory.
3. touch the FTP softkey.

The VM700A is ready to be connected to a remote host.

To start the ftp program on the VM700A from remote control:

1. start telnet on your PC, and connect to the VM700A. You can use the vm700a's ipaddress directly (e.g., "telnet 254.254.254.2", as configured in the /nvram0/IpConfig file), or you can use a pseudonym from a hosts table entry (e.g., "telnet vm700a").
2. type the command "execute FTP"
3. type the command "quit"
4. start ftp on your PC, using either the VM700A's ip address (e.g., "ftp 254.254.254.2", as configured in the /nvram0/IpConfig file) or a pseudonym from a hosts table entry (e.g., "ftp vm700a").

If password checking is enabled on the VM700A, you must issue the PASS command with the correct password before file access is granted.

Using ftp

Once you have established an ftp connection between the VM700A and your PC, you can transfer files between the PC and the VM700A, find out or set the current VM700A directory, list the files in the current VM700A directory, and quit from the ftp session (among other operations). See your ftp program documentation for a complete list of program capabilities.

put transfers files from the PC to the VM700A.

Example:

The following command transfers a file named "test.fnc" from the PC to the VM700A's current directory:

```
ftp:VM700A> put test.fnc
foreign file (default test.tnc):
Transferred 91 bytes in 1 seconds (728 bits/sec, 91 bytes/sec)
250 File transfer complete
ftp:VM700A>
```

Certain VM700A files are not writable via FTP -- these generally being those which control communication parameters of the VM700A.

When you try to write a configuration file, the file is compared against an internal template to ensure that it will be readable when needed by the VM700A. If this check fails, an error is returned, and the file is not written. To ensure the proper file format, it is a good idea to first read a valid matching configuration file from the VM700A, modify it locally, then write it back.

NOTE

*Configuration files, when successfully written, will not take effect in the VM700A until the **restoreconfig** command is issued from a remote connection or function playback, or until you cycle power on the VM700A.*

The **get** command transfers files from the VM700A to the PC.

Example:

The following command transfers a file named "Func1" from the VM700A's current directory to the PC.

```
ftp:VM700A> get Func1
local file (default Func1): func1.txt
Transferred 91 bytes in 0 seconds
250 File transfer complete
ftp:VM700A>
```

The **pwd** command returns the current VM700A directory.

Example:

```
ftp:VM700A> pwd
257 Current directory is "/nvram0/FunctionKeys".
ftp:VM700A>
```

The **cd** command sets the current VM700A directory.

Example:

```
ftp:VM700A> cd /nvram0/ConfigFiles
250 Change of directory successful.
ftp:VM700A>
```

The **ls** command lists the files in the current VM700A directory.

Example:

```
ftp:VM700A> ls
Timed-Functions
Func1
Transferred 24 bytes in 1 seconds (192 bits/sec, 24 bytes/sec)
250 File listing complete.
ftp:VM700A>
```

The **dir** command is similar to **ls**, but also returns the size (in bytes) of each directory entry. Entries with a size of 0 are most likely (but are not necessarily) directories.

Example:

```
ftp:VM700A> dir
        0  Timed~Functions
       85  Func1
Transferred 42 bytes in 0 seconds
250 File listing complete.
ftp:VM700A>
```

quit and **exit** terminate the ftp connection.

Example:

```
ftp:VM700A> quit
C:\>
```

Section 5
COMPUTER-BASED REMOTE CONTROL

This section describes how to write programs to control the VM700A remotely, without operator intervention.

There are several reasons you may wish to write a program to control the VM700A instead of controlling it from a terminal or using its front panel:

- **a greater degree of automatic operation.** Repetitive operations are more efficient and less prone to error when executed from a program than manually.
- **faster execution.** A set of operations written into a program will generally execute faster than the same operations performed manually.
- **ability to branch.** You may wish to make some measurement, then perform different action depending on the results of the measurement. Since the VM700A command language contains only limited branching constructs, any alternative courses of action you may wish to take can't be written into a VM700A function. They must be written into a program.

A "shell" program that you can use as a template for your own programs appears later in this chapter. Before presenting it, however, this chapter discusses several other matters you must consider when writing programs to control the VM700A, specifically:

- what programming language you are going to use
- whether or not you need a communications library for the language you choose
- how the VM700A communicates with a computer program, including how the VM700A signals:
  - acceptance of a command
  - error conditions
  - completion of a transmission and readiness to accept another command

PROGRAMMING LANGUAGES

Before you start to write programs to control the VM700A, you have to decide what language to write the program in. You can write programs to control the VM700A in any programming language you like. However, you will probably want to use a language for which libraries of routines are available to handle data communications. BASIC, C, and Pascal are good choices. The sample program presented in this chapter is written in Microsoft QuickBASIC. (C and Pascal users should have no problem understanding the BASIC code; the reverse is probably not true, which is why the program appears in BASIC.)

Communications Libraries

You will probably want to purchase a commercial library of communications routines to use with your program if:

- your program uses SLIP mode to transfer files between the PC and the VM700A, or
- you are writing in any language other than BASIC and using only no-protocol data transfers, or
- you are using CTS/RTS flow control.

Communications packages currently available for BASIC, C, and Pascal (and the companies that produce them) include:

BASIC: QuickComm (Software Interphase)

C: C Asynch Manager (Blaise Computing), Comm Library (Greenleaf Software), C Communications Toolkit (Magna Carta), Silverware C Asynch Library (SilverWare), Essential Communications (South Mountain Software)

Pascal: Pascal Asynch Manager (Blaise Computing)

In addition, Chapter 4, "Simple Remote Control," lists sources for commercially available SLIP implementations.

COMMUNICATING WITH THE VM700A

Before you can write programs to control the VM700A, you must understand how the VM700A communicates with a computer, including the following:

- "terminal" vs. "computer" mode (i.e., local vs. remote echo)
- VM700A responses that indicate: acceptance of a command; error conditions; completion of a transmission
- Exceptions to the previous item: the **playback** command.

Terminal vs. Computer Mode

When remotely controlled, the VM700A operates in either *terminal* mode or *computer* mode.

Terminal mode is intended for use when the VM700A is being controlled interactively from a "dumb" terminal. In terminal mode:

- the VM700A echoes characters it receives;
- the VM700A adds a space after sending its prompt; and
- the VM700A sends error messages as alphabetic text.

Computer mode is intended for use when the VM700A is being controlled by a computer program, without human intervention. In computer mode:

- the VM700A does not echo characters it receives;
- the VM700A sends only its prompt when execution of a command is completed (no additional space); and
- the VM700A sends error messages as numerical encodings, not alphabetic text.

When you write a computer program to control the VM700A, the VM700A is best used in computer mode.

VM700A Responses

In computer mode, the VM700A responds to commands in one of two ways:

- the VM700A sends an '@' character if the command was successful;
- the VM700A sends a '?' character or '!' character followed by a three-digit error encoding if the command was not successful.

After sending the '@' character, the VM700A sends any data transmission initiated by the command (e.g., "show *filename*"). When this data transmission is complete, the VM700A sends the remote prompt to indicate readiness to accept another command.

The exception to this rule is the **playback** command. After it receives a playback command, the VM700A sends no '@'. Once function playback begins, the next character the computer program receives from the VM700A will be either a '?' or '!' character (if an error occurs during function playback, or if the VM700A sends an informational message) or the characters that make up the remote prompt.

After sending the '?' or '!' character and a three-digit error identifier, the VM700A sends the remote prompt, unless the error is returned because remote operation has been terminated.

STRUCTURE OF A VM700A PROGRAM

Once you know the "rules" that the VM700A uses to respond to commands, you can determine what the structure of a "generic" VM700A remote control program should be. Such a program can be divided into three parts:

1. initialize variables, then establish communication and remote operation;
2. execute and respond to commands; and
3. clean up.

The first part of the program sets up any variables that will be required, opens any files that will be required, opens a communications channel to the VM700A, then (in no-protocol mode) sends the command "remote". If this operation is successful, the LED on the VM700A's Configure button starts blinking. This indicates that the VM700A is now in remote mode, being controlled by the computer program. (You can terminate remote operation from the front panel by pressing the Configure button. If remote operation is terminated in this way in no-protocol mode, it can be re-started by sending the command "remote".)

The main loop of the program sends a command to the VM700A, reads the VM700A's response, performs an appropriate action depending on the VM700A's response (such as branching to another part of the program, displaying an error message, logging data into a file, etc.), reads the VM700A's prompt, then loops back to send another command. This loop may also need to handle flow control, i.e., telling the VM700A to stop sending characters until the ones already sent can be processed, and re-starting the flow of characters from the VM700A once they are processed). The main loop of the program should execute until the setting of some variable indicates that program execution is complete.

The final part of the program performs final cleanup functions (writing out and closing files, closing communications channels, restoring the VM700A's initial configuration, etc.).

SAMPLE PROGRAM: SHELL.BAS

The following pages list and explain a sample program called SHELL.BAS to control the VM700A. You can use this sample program as a starting point from which to write your own .VM700A programs.

Before running the program, make sure that the VM700A is properly set up for communicating with the PC: the proper cable is connected to the proper port on both the PC and the VM700A, the remote port is set correctly in the Communication~Setup file, the baud rate is set correctly, etc. When running SHELL.BAS, make sure that Flow Control for the remote port in the VM700A is set to XON/XOFF.

A listing of the SHELL.BAS program follows, along with comments about what each section of the program does.

Program Overview

SHELL.BAS is written in Microsoft QuickBASIC. SHELL.BAS accepts as input a user-specified text file containing VM700A commands, written one command to a line. The program reads and executes each command sequentially. SHELL.BAS logs all commands and data in another user-specified file, in addition to displaying them on the PC screen.

SHELL.BAS demonstrates many items of interest in writing programs for the VM700A, including:

- how to acquire the VM700A prompt the first time, so that it can be monitored to indicate the completion of a transmission from the VM700A;
- how to handle VM700A responses to commands;
- how to handle XON/XOFF flow control;
- how to decode VM700A error codes.

Preliminaries

Lines 1 through 35 perform some preliminary tasks before the main part of the program starts.

Lines 5 through 10 declare several functions and subroutines that will be used later.

Lines 12 through 18 set up some global variables that will be needed later.

Lines 19 through 35 get the names of the input and output files from the keyboard, open the files, then start the output file by writing the date, the time and the names of the input and output files.

Establishing Communication

Lines 37 through 42 establish communication with the VM700A.

Line 38 sets up output port COM1 on the PC to communicate at 9600 baud, no parity, eight data bits, one stop bit, send a line feed with each RETURN, and initialize the DSR (Data Set Ready) line to 0.

Line 39 sends the command "remote" out port COM1.

Line 40 calls the GetPrompt function, which waits for the VM700A's response and records the VM700A prompt for later use in recognizing end-of-transmission.

Line 41 displays the prompt character sequence on the screen. Line 42 writes the prompt character sequence to the output file.

Main Program Loop

Lines 44 through 94 form the main program loop.

Lines 46 through 79 form a large WHILE loop that executes as long as there are lines to be read from the input file and no fatal error has been encountered.

Line 47 reads a line from the input file. If the line read is not blank, line 49 prints it on the screen, line 50 writes it to the output file, and line 51 sends it to the VM700A by calling PutCmd.

Lines 52 through 67 handle the case where PutCmd! returns a non-zero value, i.e., an error or warning message happened during execution of the last command.

Table 5-1
Sample Program for PC Communications With VM700A

```
' sample program for controlling VM700A
' reads text file containing VM700A commands

' declarations
DECLARE FUNCTION ReadError! (return$)
DECLARE FUNCTION ReadString (text$, start$, p$)
DECLARE FUNCTION PutCmd! (Cmd$, return$)
DECLARE FUNCTION getPrompt$ (i%)
DECLARE SUB gettimeout (i%)
DECLARE SUB VMError (c!, message$)

' preliminaries
holdoff = 0             'used for XON/XOFF flow control
xon$ = CHR$(17)         'used for XON/XOFF flow control
xoff$ = CHR$(19)        'used for XON/XOFF flow control
return$ = ""
fatalerror = 0
CLS
' get name of file to read commands from
infile$ = ""
WHILE infile$ = ""
  INPUT "Name of command file (input):", infile$
  WEND
OPEN infile$ FOR INPUT AS #2

' get name of file to log results to
outfile$ = ""
WHILE outfile$ = ""
  INPUT "Name of log file (output):", outfile$
  WEND
OPEN outfile$ FOR OUTPUT AS #3
PRINT #3, DATE$
PRINT #3, TIME$
PRINT #3, "Input file:"; infile$
PRINT #3, "Output file:"; outfile$

 ' setup
 OPEN "com1:9600,N,8,1,LF,DS0" FOR RANDOM AS #1
 PRINT #1, "remote"
 prompt$ = getPrompt$(1)
 PRINT "Prompt: "; prompt$
 PRINT #3, "Prompt: "; prompt$
```

Table 5-1 (cont)

```
'main loop
'read lines from input file
WHILE NOT (EOF(2)) AND fatalerror = 0
  LINE INPUT #2, inputline$
  IF inputline$ <> "" THEN
    PRINT inputline$
    PRINT #3, inputline$
    c = PutCmd!(inputline$, return$)
    IF c <> 0 THEN
      fatalerror = 1
      IF c < 0 THEN
        PRINT "Error encountered:"; -c
        PRINT #3, "Error encountered:"; -c
      ELSE
        PRINT "Warning message:"; c
        PRINT #3, "Warning message:"; c
        END IF
      CALL VMError(c, message$)
      PRINT message$
      PRINT #3, message$
      IF c <> 7 AND c <> 8 AND c <> -17 THEN
        c = PutCmd!("quit", return$)
        END IF
      END IF
    IF fatalerror = 0 THEN
      d = 0
      DO UNTIL d = 1
        d = ReadString(c$, return$, prompt$)
        PRINT c$
        PRINT #3, c$
        LOOP
      PRINT prompt$;
      PRINT #3, prompt$;
      END IF
    END IF
  WEND
IF fatalerror = 0 THEN
  PRINT:PRINT "End of input file reached"
  PRINT #3,:PRINT #3, "End of input file reached"
  PRINT "quit"
  PRINT #3, "quit"
  c = PutCmd!("quit", return$)
  IF c <> 0 THEN CALL VMError(c, message$)
ELSE 'fatalerror=1
  PRINT "Fatal error encountered"
  PRINT "Program terminated"
  PRINT #3, "Fatal error encountered"
  PRINT #3, "Program terminated"
  END IF
CLOSE ALL
END
```

Table 5-1 (cont)

```
Line
  95   FUNCTION getPrompt$ (i%)
       SHARED holdoff
       SHARED xon$, xoff$
        DO
        LOOP UNTIL NOT EOF(i%)
 100   loopal:
       '
       ' Check if the input buffer is up to 128 bytes.
       ' If it is, then send xoff to stop transmission
       '
 105      IF LOC(i%) > 128 THEN
            holdoff = 1
            PRINT #i%, xoff$;
          END IF
       '
 110   ' If there are characters to read in, then do so,
       '
          IF LOC(i%) > 0 THEN
            c$ = INPUT$(LOC(i%), #1)
            text$ = text$ + c$
 115      END IF
       '
       ' If there are no more characters left, and tx has been held off,
       ' then turn it back on with xon.
       '
 120      IF holdoff AND LOC(1) = 0 THEN
            holdoff = 0
            PRINT #i%, xon$;
          END IF
       '
 125   ' Check for eof, and loop for a while in case it was only momentary.
       '
        IF NOT EOF(i%) THEN GOTO loopal
         FOR j = 1 TO 4000
           IF NOT EOF(i%) THEN EXIT FOR
 130     NEXT
       '
       ' If the loop was exited early, then continue with reading.
       '
         IF j < 4000 THEN GOTO loopal
 135   getPrompt$ = text$
       END FUNCTION

       FUNCTION PutCmd (Cmd$, return$)
         PRINT #1, Cmd$
 140     PutCmd = ReadError(return$)
       END FUNCTION
```

Table 5-1 (cont)

```
FUNCTION ReadError (return$)
  return$ = ""
  ErrCode = 0
' Read in VM700A Computer response
  DO UNTIL LOC(1) > 0
  LOOP
  c$ = INPUT$(1, #1)
  IF c$ = "?" OR c$ = "!" THEN
    DO UNTIL LOC(1) >= 3
    LOOP
    num$ = INPUT$(3, #1)
    ErrCode = VAL(num$)
    IF c$ = "?" THEN
      ErrCode = -ErrCode
      END IF
  ELSEIF c$ <> "@" THEN
    return$ = c$
  END IF
ReadError = ErrCode
END FUNCTION

FUNCTION ReadString (text$, start$, p$)
SHARED holdoff
SHARED xon$, xoff$
 DO                                     ' Wait until there are
   LOOP UNTIL NOT EOF(1)                ' Characters to read.
 temp$ = start$                         ' Initialize temp$
 last = 0                               ' Flag for prompt found

 p = 1                                  ' Initialize prompt count
 FOR i = 1 TO LEN(start$)
 temp2$ = MID$(start$, i, 1)
 temp3$ = MID$(p$, i, 1)
 IF temp2$ = temp3$ THEN p = p + 1 ELSE p = 1
 NEXT

 done = 0                               ' Set done flag to zero
```

Table 5-1 (cont)

Line

```
 DO WHILE done = 0
'
' Check if the input buffer is up to 128 bytes.
' If it is, then send xoff to stop transmission
'
   IF LOC(1) > 128 THEN
     holdoff = 1
     PRINT #1, xoff$;
   END IF
'
' If there are characters to read in, then read one in.
'
   IF LOC(1) > 0 THEN
     c$ = INPUT$(1, #1)
     IF ASC(c$) = 10 THEN                ' If linefeed, then done
       done = 1
     ELSEIF ASC(c$) <> 13 THEN           ' Skip carriage returns
       temp$ = temp$ + c$                ' Add it to string
       d$ = MID$(p$, p, 1)
       IF d$ <> c$ THEN                  ' Is it part of prompt?
         p = 1                           ' No
       ELSE
         IF p = LEN(p$) THEN             ' Was it the whole prompt?
           done = 1                      ' Yes
           found = 1                     ' Prompt found
           text$ = MID$(temp$, 1, LEN(temp$) - p)   ' Remove prompt
         ELSE
           p = p + 1
         END IF
       END IF
     END IF
   END IF
'
' If there are no more characters left, and tx has been held off,
' then turn it back on with xon.
   IF holdoff AND LOC(1) = 0 THEN
     holdoff = 0
     PRINT #1, xon$;
   END IF
'
' Check for eof, and loop for a while in case it was only momentary.
  IF EOF(1) AND NOT done THEN
    FOR j = 1 TO 4000
      IF NOT EOF(1) THEN EXIT FOR
    NEXT
' If the loop was exited early, then continue reading.
    IF j > 4000 THEN EXIT DO
  END IF
 LOOP
IF found = 0 THEN text$ = temp$   ' Line if data without prompt
ReadString = found
END FUNCTION
```

Table 5-1 (cont)

```
Line
SUB VMError (c!, message$)
  SELECT CASE c!
  CASE -1
    message$ = "Bad command argument"
  CASE -2
    message$ = "Sub-function not found"
  CASE -3
    message$ = "Playback nesting too deep"
  CASE -4
    message$ = "Function directory inaccessible"
  CASE -5
    message$ = "Function not found"
  CASE -6
    message$ = "Unknown command"
  CASE -7
    message$ = "Unknown hardkey"
  CASE -8
    message$ = "Out of memory"
  CASE -9
    message$ = "Recursive function call"
  CASE -10
    message$ = "Bad command in this context"
  CASE -11
    message$ = "Name too long"
  CASE -12
    message$ = "No filename"
  CASE -13
    message$ = "Line too long"
  CASE -14
    message$ = "Command only meaningful for non-IP connections"
  CASE -15
    message$ = "Bad time format (use getclock)"
  CASE -16
    message$ = "Function playback in progress.  Type ^C to stop it."
  CASE -17
    message$ = "Remote not enabled"
  CASE -101
    message$ = "Request filtered"
  CASE -102
    message$ = "Screen event not key"
  CASE -103
    message$ = "Unknown softkey"
  CASE -104
    message$ = "Invalid softkey"
  CASE -105
    message$ = "Unwanted hardkey"
  CASE -106
    message$ = "Unknown input"
  CASE -107
    message$ = "Not found"
  CASE -108
    message$ = "Request not supported"
```

Table 5-1 (cont)

```
  CASE -109
    message$ = "No server resources"
  CASE -110
    message$ = "Illegal name"
  CASE -111
    message$ = "Not writable"
  CASE -112
    message$ = "Not readable"
  CASE -113
    message$ = "No permission"
  CASE -114
    message$ = "Bad argument(s)"
  CASE 5
    message$ = "Error occurred in function playback"
  CASE 6
    message$ = "Type carriage return to continue"
  CASE 7
    message$ = "Remote terminated"
  CASE 8
    message$ = "Remote has been terminated locally"
  CASE 10
    message$ = "Application returned error status"
  END SELECT
END SUB
```

Lines 68 through 78 handle the case where PutCmd returns a zero value, i.e., the last command executed successfully. Line 71 calls function ReadString to read characters from the VM700A and print them, both on the screen and into the output file, until it sees the VM700A prompt.

Lines 80 through 86 print appropriate messages, both on the screen and in the output log file, when the end of the input file is reached.

Lines 88 through 92 print appropriate messages, both on the screen and in the output log file, when program execution is halted because of a VM700A error condition.

Function GetPrompt

Function GetPrompt reads characters from the VM700A until the VM700A stops sending characters. The characters it reads in are assumed to be the VM700A prompt, which the VM700A uses to signal end-of-transmission.

Lines 96 through 98 declare variables holdoff, xon$ and xoff$ to be "shared" variables. Shared variables share their values with other procedures and functions and with the main program. These are similar to "static" variables in other programming languages.

Lines 100 through 108 check to see if the input buffer is past the "critical" point, defined as 128 bytes. (The LOC function returns the number of characters in the input buffer for a specified communications channel.) If the input buffer contains 128 bytes or more, the program sets the value of variable holdoff to 1 ("true"), then sends the XOFF character to the VM700A.

Lines 110 through 115 read a line of input from the VM700A into variable c$, then appends the contents of c$ to variable text$.

Lines 117 through 123 check to see if the input buffer is empty and the variable holdoff is set to 1. If both of these conditions hold, the programs sets holdoff to 0 and sends the XON character to the VM700A.

Lines 127 loops back to line 100 if "end-of-file" has not been seen from the VM700A.

Lines 128 through 130 simply delay to give the VM700A a chance to transmit another character, in order to avoid spurious end-of-transmission readings.

Line 134 tests to see if the FOR...NEXT loop in lines 128 through 130 was exited early. If it was, the program loops back to line 100. Otherwise, line 135 assigns the contents of variable text$ to function GetPrompt, and line 136 returns from the function call.

Function PutCmd

Function PutCmd sends a string out the COM1 port to the VM700A, then calls function ReadError to see if the VM700A accepts the string as a command.

Function ReadError

Function ReadError reads characters from the VM700A to determine if a command sent by PutCmd caused a VM700A error condition.

If the first character that ReadError reads is an "@" character, the command was successful.

If the first character that ReadError reads is a "?" or "!" character, the command caused an error condition or returned a warning. ReadError reads the next three characters and returns them in ErrCode.

If the first character that ReadError reads is not an "@", "?", or "!", then a playback command is executing and the first character is probably the first character of the VM700A prompt string. It is returned in variable return$ for special handling by the calling routine in this instance.

Function ReadString

Function Readstring reads characters from the VM700A until it reads the VM700A prompt. It then returns all characters read up to the start of the VM700A prompt.

Parameter text$ contains the string that ReadString returns.

Parameter start$ contains the string that ReadString starts out with when it is called. This is usually the null string, but may also contain characters returned by a previous call to ReadError.

Parameter p$ contains the VM700A prompt string. Characters read in from the VM700A are compared to this string to see if the end of the VM700A transmission has been reached.

If ReadString finds the VM700A prompt, it returns a value of 1. If it times out without finding the VM700A prompt, it returns a value of 0.

Sub VMError

Subroutine VMError takes a number as input and returns an error or warning message corresponding to that number.

Section 6
MISCELLANEOUS TOPICS

MONITORING AUTO-MODE OPERATION

A common use of the VM700A's remote capabilities is to monitor Auto-mode measurements, and print a warning message when test results are out of limits.

When doing this, it is important to remember that remote monitoring is different from remote control. The VM700A doesn't have to be in remote mode to allow you to monitor tests. You can simply attach a printer or terminal to the VM700A's log port to monitor test results.

If you want to monitor Auto mode measurements while the VM700A is under remote control, you must avoid sending commands to the VM700A at the same time that the VM700A is sending log data to the computer. Doing so can cause the two data streams (log data and commands) to interfere with each other, with unpredictable results.

The following sections explain how to monitor Auto mode operation, in remote mode or otherwise.

Monitoring Auto-Mode Operation Without Remote Control

To monitor Auto mode operation without putting the VM700A under remote control, use the following procedure:

1. Attach the monitoring device (printer, terminal, data logger, etc.) to the VM700A's Log port. To find out which port is the Log port, check the Log Port entry in the Communication Setup file.
2. In the Communication Setup file, set the format of the Log port to an appropriate setting for the monitoring device (ASCII Printer, PostScript, Epson LQ, etc.).
3. Press the Auto button.

A message appears on the monitoring device when a measurement falls outside its alarm limits, or when a measurement transitions from outside its alarm limits to within its alarm limits. See the section entitled "Consecutive Errors Parameter" for more information about when a measurement is logged.

Monitoring Auto-Mode Operation With Remote Control

To monitor Auto mode operation when the VM700A is under remote control, use the following procedure:

1. Set the Remote Control port and the Log port to the same serial port in the Communication Setup file.
2. Establish a connection between the PC and the VM700A.

3. Execute the Auto measurement.
4. Read the VM700A's prompt, indicating that the command to execute Auto mode has been received.
5. Wait for further response from the VM700A. This will be a message indicating that an alarm limit has been violated.
6. Take appropriate action with the alarm message (e.g., write it into a file, branch to another section of program, etc.).

See the section entitled "Consecutive Errors Parameter" for more information about when a measurement is logged.

Consecutive Errors Parameter

The VM700A sends a message out the Log port when:

- a measurement transitions from within its alarm limits to outside its alarm limits; or
- a measurement transitions from outside its alarm limits to within its alarm limits.

The "consecutive errors before reporting" parameter specified in the current Auto Limits file controls the frequency with which log messages are sent. For a log message to be sent concerning a measurement, that measurement must fall outside its alarm limits for the number of Auto mode cycles specified by the consecutive errors parameter. (An Auto-mode cycle is a single pass through the currently selected Auto-mode measurements.)

Once a log message is sent warning of an out-of-limits measurement, the next message concerning that measurement will be sent only if the measurement falls within its alarm limits for the number of Auto mode cycles specified by the consecutive errors parameter.

Successive messages about a measurement continue in this fashion, one indicating that an alarm limit has been violated, the next indicating that the measurement has returned to its alarm limit range. This method of operation keeps the data monitoring device from being bombarded with a stream of repetitive alarm-violation messages.

MODEM CONTROL

If you connect the VM700A's remote port to a modem, you can set up functions that dial the modem, acquire a carrier signal, transmit data over a telephone line, and hang up the telephone. In order to use the VM700A in this way, the modem must be set up to meet certain requirements:

1. The modem must be silent. It must not issue characters to be read by the VM700A, or echo back characters sent to it by the VM700A. This means, for instance, that Hayes-compatible modems should be set to not return results codes, and to not echo received characters.
2. The modem should use hardware flow control to signal readiness to receive data, or to halt data transmission from the VM700A.

The Control Command

The **control** command sends a specified string of characters out the VM700A's Control port. (Note: The Control port referred to in the previous sentence is not the same as the Remote Control port. It is a distinct entry in the Communication Setup file.) You typically use **control** to send commands to a modem or switcher.

The string can be 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 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 VM700A function or program will continue executing.

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

Examples:

```
VM700A> control ^G
```

This example sends a bell character out the Control Port.

```
VM700A> control ATDT123-4567^M
```

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

The ? and ?! Constructs

The **?** construct is followed by any configuration keyword and a single command. During function playback, if the keyword has non-zero value, the rest of the line following the **?** construct is executed. Otherwise,function execution continues with the next statement in the function.

The **?!** construct is similar, except that the rest of the line is executed if the keyword has a value of zero. Otherwise,function execution continues with the next statement in the function.

The **?** and **?!** constructs are available only during function playback. They are typically used to test the value of a Carrier Detect line in order to control communications with a modem, or to test the global out-of-limits (GOOL) flag to see if an application found one or more measurements that were out-of-limits.

Example:

```
?GCD0 return
delay 5
loop
```

This first line of this example tests the status of the Carrier Detect line on serial port 0. If the test returns true, the function returns to its caller. If it returns false, the function waits one half-second, then loops back to the first command.

A "Generic" Modem Control Function

Functions that dial up a telephone line, send data, then hang up the telephone take a standard form, composed of four parts:

1. send the modem the command to dial the telephone
2. wait until the telephone connection is established
3. send the data
4. hang up the telephone

The following discussion demonstrates and explains the use of each of these parts.

THE MAIN FUNCTION

The following function sends a command to a modem to dial a telephone number, calls a function that waits until the telephone connection is established, runs the H_Timing measurement, obtains its results, sends the results over the telephone line, then hangs up the phone:

```
control ATDT123-4567^M
delay 10
playback waitforCD
execute H_Timing
getresults
spool H_Timing
playback hangup
```

Note that the **control** command does not delimit the sequence of characters it sends by a carriage return or line feed. Therefore, a carriage return (^M) had to be appended to the string sent by the **control** command.

Note also that the **spool** command sends the contents of the H_Timing measurement results file out the Copy port. Thus, the Copy port and the Control port must be configured to the same physical port if this function is to work.

THE WAITFORCD FUNCTION

The main function shown above called a function called "waitforCD" immediately after sending the command to the modem to dial the telephone. The waitforCD function simply consists of a test to see if Carrier Detect is high, followed by an immediate return from the function if the test returns true, or a short delay and a loop back to the start if the test returns false:

```
?GCD0 return
delay 5
loop
```

Note that the **?GCD0** and **?GCD1** commands refer to physical and not logical ports. Thus, if the Copy port is ever changed from 0 to 1 or 1 to 0, any functions using the **?GCD0** or **?GCD1** commands will have to be changed as well.

THE HANGUP FUNCTION

To disconnect a telephone connection, Hayes-compatible modems must send three "attention" characters (by default, these are plus signs), followed by the command "ath", followed by a carriage return. A brief delay between each attention character makes sure it is received properly and the modem takes appropriate action:

```
control +
delay 4
control +
delay 4
control +
delay 20
control ath^M
```

Appendix A
VM700A COMMANDS

This appendix describes the VM700A commands for use in functions and remote operation. For each command, the following information is included:

- the syntax form for the command
- the descriptive form for the command
- the scope of the command, i.e., whether the command can be used in function playback, remote operation, or both
- the VM700A software version the command runs on
- an explanation of what the command does
- one or more examples of the command's use.

Table A-1 summarizes the VM700A commands and their uses.

Table A-1
VM700A Commands

Command	Scope	Purpose
appset	Function playback	Records state information about an application. Not user-editable.
appstart	Function playback	Starts an application's execution within a function. Not user-editable.
cancelcopy	Remote operation	Clears print spooler.
cknob	Function playback	Turns control knob continuously, a specified number of clicks over a specified amount of time.
computer	Remote operation	Sets VM700A to be controlled by a computer.
control	Both	Sends characters out control port.
controlbreak	Both	Sends break sequence out control port.
delay	Function playback	Pauses function execution for a user-specified time.
disptext	Function playback	Displays text messages, pauses function execution.
execute	Both	Executes a VM700A application.
exit	Remote operation	Terminates remote operation, closes PC connection
filesin	Remote operation	Returns names of all files in a directory.
get	Remote operation	Returns configuration values specified by keyword.
getclock	Remote operation	Returns date and time from VM700A system clock.
getresults	Both	Writes a results file for current application.
hardkey	Both	Presses and releases a front-panel button.
hardpress	Both	Presses, but does not release, a front-panel button.
hardrelease	Both	Releases a front-panel button.
knob	Both	Rurns the control knob.
loop	Function playback	Re-starts current function from beginning
playback	Both	Executes a function.
print	Both	Sends formatted file to print spooler.
query	Remote operation	Returns information about a VM700A keyword.
quit	Remote operation	Terminates remote operation, closes PC connection.
remote	Remote operation	Initiates remote control in non-protocol mode.
rename	Both	Renames file on the VM700A.
res	Remote operation	Sends ASCII string containing result of current measurement out remote port.
resoff	Remote operation	Turns off previous reson command.
reson	Remote operation	Similar to res, but sends continuous stream of measurement results out remote port.
restoreconfig	Both	Restores configuration values from VM700A files.
return	Function playback	Returns a function to the calling function.

Table A-1 (cont)

Command	Scope	Purpose
set	Both	Sets configuration values specified by keyword.
setclock	Remote operation	Sets date and time on VM700A system clock.
show	Remote operation	Returns contents of specified file.
softkey	Both	Presses and releases a specified softkey.
softpress	Both	Presses, but does not release, a specified softkey.
softrelease	Both	Releases a specified softkey.
spool	Both	Sends ASCII file to print spooler.
stop	Function playback	Exits from all functions currently executing.
terminal	Remote operation	Sets VM700A to be controlled from a terminal.
touchpress	Both	Touches VM700A screen at specified X, Y location.
touchrelease	Both	Releases previous touchpress.
?	Function playback	Tests keyword, executes command if keyword returns true.
?!	Function playback	Tests keyword, executes command if keyword returns false.

COMMANDS

The remote control and function playback commands are presented in this manual as follows:

Syntax Form: **command** *argument(s)*

Descriptive Form: **command** *argument(s)*

The syntax form gives the name of the command and the types of arguments it takes, if any. The descriptive form gives the name of the command and the meanings of the arguments it takes. **command** is the actual command name. Literal command input is shown in **boldface** type. Arguments and their meanings are shown in *italic*. Optional arguments are enclosed with [ ]. Examples are included for most commands with arguments. Examples are shown indented, in monospace font.

APPSET

Syntax Form: **appset** *string number*

Descriptive Form: **appset** *variable_name value*

Scope: function playback

appset encodes the state of an application within a function. This helps ensure that the function plays back correctly.

Appset commands only appear in functions learned from the front panel. They are not meant to be edited by a user. Editing an appset command could lead to unpredictable results.

APPSTART

Syntax Form: `appstart`

Descriptive Form: `appstart`

Scope: function playback

appstart ends a list of appset commands and starts the application. This command is required for correct function playback. Do not delete it.

CANCELCOPY

Syntax Form: `cancelcopy`

Descriptive Form: `cancelcopy`

Scope: remote operation

cancelcopy clears the print spooler. The command is equivalent to touching the Cancel Copy softkey after pressing the Configure button while the VM700A contains spooled data.

CKNOB

Syntax Form: **cknob** *integer integer*

Descriptive Form: **cknob** *number-of-clicks tenths-of-seconds*

Scope: function playback

See also: `knob`

cknob turns the control knob a specified number of clicks over a specified period of time. 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. Time units are specified in tenths of seconds.

Example:

```
    VM700A> cknob 50 30
```

This command turns the control knob clockwise 50 clicks over a period of three seconds.

COMPUTER

Syntax Form: `computer`

Descriptive Form: `computer`

Scope: remote operation

See also: `remote, terminal`

computer sets the VM700A to be controlled by a computer. This command is used only in no-protocol mode. After this command is executed, characters are not echoed, there is no space after the prompt, and any errors or messages are returned encoded. See Appendix D for information about the encoded error messages returned by the VM700A. See also **remote** and **terminal**.

CONTROL

Syntax Form: `control` *string*

Descriptive Form: `control` *sequence_of_characters_to_send*

Scope: function playback, remote operation

control sends a user-specified string out the control port. This string can be 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

control is usually used to control a device attached to the VM700A, 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 VM700A function or program will continue executing.

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

Examples:

```
VM700A> control ^G
```

This example sends a bell character out the Control Port.

```
VM700A> 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*

Descriptive Form: `controlbreak` *tenths-of-seconds*

Scope: function playback, remote operation

controlbreak sends a break character out the control port for the specified number of time units (tenths of seconds).

Example:

```
VM700A> 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 actually transmitted for a shorter time than requested. (This happens because of the way the VM700A 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.*

DELAY

Syntax Form: **delay** *number*

Descriptive Form: **delay** *tenths-of-seconds*

Scope: function playback

delay pauses function execution for a specified amount of time, given in tenths of seconds.

Example:

```
VM700A> delay 10
```

This command pauses function execution for a full second.

DISPTEXT

Syntax Form: **disptext** *string*[*newline string*...]

Descriptive Form: **disptext** *string*[*newline string*...]

Scope: function playback

disptext displays a user-specified message on the VM700A front panel. When the message appears on the VM700A front panel, function execution stops until you touch the display screen.

Up to 79 characters can appear on a line. You can create a multi-line message by ending a line with a backslash character ("\").

If the **disptext** command is contained in a function being played back from remote control, the message appears on the remote terminal if the Remote Control/Message Display item in the Communication Setup file is set to Remote. In this case, function execution stops until you type a carriage return.

If the Remote Control/Message Display item in the Communication Setup file is set to VM700 Screen, the message appears on the instrument's display screen even when the function is being played back from remote control. Function execution stops until you touch the display screen.

Example:

```
disptext Hello, world!
```

When this line is executed in a function, it displays the message "Hello, world!" on the VM700A display. Function execution stops until you touch the screen.

```
disptext \
Roses are red,\
Violets are blue,\
Sugar is sweet,\
And so's a million dollars tax-free.
```

When this line is executed in a function, it displays four lines of doggerel on the VM700A display. Function execution resumes when you touch the screen.

EXECUTE

Syntax Form: **execute** *string*

Descriptive Form: **execute** *application-name*

Scope: function playback, remote operation

execute starts aVM700A application. An application is one of the executable files (with exceptions noted below) found in the Instrument Operations, VM700 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 softkey from one of the Measure-mode windows.

Example:

```
VM700A> 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. Table A-2 lists the applications you can execute under remote control and gives their exact names in functions and remote operation commands. See "Naming Conventions," near the end of this chapter, for information about the rules for specifying application names.

Table A-2
Applications Available in Functions and Remote Operation

Instrument Operations				
Configure[a]	FTP	Function[a]	Measure[a]	Picture
SystemInit	Vector	Waveform		
VM700A Diagnostics[b]				
Acquisition~Diagnostic	AdcGain~Adjustment	AnalogInput~Diagnostic	AudioAnalog~Diagnostic	AudioProcessor~Diagnostic
CalDac~Adjustment	Controller~Diagnostic	DiagsLoop	FilterBoard~Diagnostic	Genlock~Diagnostic
Measure~Sinewave	Measure~Squarewave	Measure~Temperature		
Video Measurements[c]				
Bar~LineTime	Bounce	Burst~Frequency	ChromLum~GainDelay	Chrominance~AMPM
Chrominance~FreqResp (NTSC only)	Chrominance~NonLinearity	ColorBar	DGDP	GroupDelay~SinX_X
H_Blank	H_Timing	ICPM	Jitter	Jitter~Long_Time
K_Factor	Line~Frequency	Luminance~NonLinearity	MultiBurst	Noise~Spectrum
SCH_Phase	TwoField	VITS~ID (NTSC only)	V_Blank	Video~Standard
Echo/Rounding (Option 1G)[c]				
Echo	Rounding~Errors			
Teletext (Option 20)[c]				
SoundInSync	Teletext			
Camera Testing (Option 21)[c]				
Colorimetry	Defects	Detail	Fixed_Pattern~Noise	Frequency~Response
Gamma	Geometry~Registration	Shading	Vertical~Smear	

Table A-2 (cont)

Component (Option 30) [c]				
Bowtie	Component~ Channel_Delay	Component~ColorBar	Component~ K_Factor	Component~ LevelMeter
Component~ Multiburst	Component~Noise	Component~ NonLinearity	Component~Vector	Lightning
Audio (Options 40 and 41) [c]				
Audio~Analyzer	Audio~Monitor	Audio~Spectrum	Calibrate~ AudioBoard	Identify~ Audio_Hardware
Multitone	View_Audio~ Auto_Test			

[a] *The Configure, Function, and Measure files in the* ***Instrument~Operations*** *directory cannot be selected via remote control.*

[b] *Diagnostic routines can be selected via remote control, but passed test results are not returned. However, errors are logged to the* ***Diagnostic~Errors*** *file which can be read via remote control.*

[c] *New applications may be included as new version of the VM700A are introduced. Press the Measure key to find out what applications are available for your VM700A.*

EXIT

Syntax Form: exit

Descriptive Form: exit

Scope: remote operation

exit terminates the remote session and closes the connection between the VM700A and the computer or terminal. **exit** takes no arguments. **exit** is equivalent to **quit**.

FILESIN

Syntax Form: **filesin** *string*

Descriptive Form: **filesin** *VM700A_directory*

Scope: remote operation

filesin returns the names of all files in the specified directory.

Example:

```
VM700A> filesin /nvram0/FunctionKeys/myfiles
myfile1      myfile2      myfile3
VM700A>
```

GET

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

Descriptive Form: `get keyword [A|B|C]`

Scope: remote operation

See also: **set**

get command returns the configuration file value specified by *keyword* on the channel specified by *channel_letter*. The *keywords* available are listed in Appendix B.

Example:

```
VM700A> get VSTA
NTSC
VM700A> get DHSM A
Meas_Set_1
VM700A>
```

The first command in the example returns the current video standard for source A. (In this case, it's 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 file's name is Meas_Set_1.)

GETCLOCK

Syntax Form: `getclock`

Descriptive Form: `getclock`

Scope: remote operation

getclock command returns the date and time from the VM700A 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. **getclock** takes no arguments.

Example:

```
VM700A> getclock
Jul 27 14:54:37 1989
```

GETRESULTS

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

Descriptive Form: `getresults [verbose] [keyword[ keyword]]`

Scope: function playback, remote operation

getresults command stores Measure or Auto mode measurement results in default files in the Measurement~Results directory. In Measure mode, entering getresults 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, the message **Results in file:** *filename* is returned. Use the **show filename** command to view the results.

Example:

```
VM700A> getresults
Results in file: ChromLum~GainDelay
```

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 (++++).

For DGDP and Luminance~NonLinearity, the additional data are the readouts at each step level in the display.

For GroupDelay~SinX_X, 82 additional measurement results are provided, at various frequencies. The first group of results shows the energy in dB, while the second group shows the delay in ns. Initial frequency is 138.5 kHz, with an increment of 69.3 kHz. Frequency and increment are calculated using the following formulas (based on a sub-carrier frequency, F_{SC}, of 4.43316875 MHz):

freq (GroupDelay) = 2 x $F_{SC}/64$, increment (GroupDelay) = $F_{SC}/64$

For Noise~Spectrum, 198 additional measurements are provided, at various frequencies. The additional data represents noise energy in dB at the frequency. Initial frequency is 34.6 kHz, with frequency increment 34.6 kHz. Frequency and increment are calculated using the following formulas (based on a sub-carrier frequency, F_{SC}, of 4.43316875 MHz):

freq (Noise~Spectrum) = $F_{SC}/128$
increment (Noise~Spectrum) = $F_{SC}/128$

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. The message **Results in file: Auto** is returned.

You can also use **getresults** in Auto mode with one or more keyword arguments. The keywords, listed in Appendix B, 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.

Example:

```
VM700A> execute H_Timing
VM700A> getresults
H_Timing
VM700A>
```

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

```
VM700A> execute Auto
VM700A> getresults PBAA PBRT PSTB
Results in file: Auto
```

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

HARDKEY

Syntax Form: `hardkey` *string*

Descriptive Form: `hardkey` *button_name*

Scope: function playback, remote operation

See also: **hardpress, hardrelease**

hardkey is equivalent to pressing and releasing a specified front panel button. **hardkey** has the same effect as **hardpress** followed by **hardrelease**; however, in general **hardkey** should be used instead of these commands.

Example:

```
VM700A> hardkey Vector
```

This command has the same effect as pressing the Vector button on the front panel.

Front panel button names are listed in Table A-3.

Table A-3
Front Panel Button Names

A	Display	Picture
Auto	Freeze	SelectLine
Average	Graticule	Vector
B	Help	Waveform
C	Menu	XY *(Arrow selector)*
Copy	MoveExpand	

NOTE

The Configure, Function, and Measure buttons cannot be selected in functions or remote operation.

When specifying button names in functions or remote operation commands, it is important to match the name shown in Table A-3 exactly, including upper or lower case.

HARDPRESS

Syntax Form: `hardpress` *string*

Descriptive Form: `hardpress` *button_name*

Scope: function playback, remote operation

See also: **hardkey, hardrelease**

hardpress is equivalent to pressing a specified front panel button without releasing it. The button remains "pressed" until a **hardrelease** command is received. **hardpress** and **hardrelease** should be used with buttons that need to be held while another action takes place. For all other buttons, use **hardkey**. Front-panel button names are listed in Table A-3. 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, it is important to match the name shown in Table A-3 EXACTLY, including upper or lower case.

Example:

```
VM700A> hardpress Display
VM700A> knob 50
VM700A> hardrelease Display
```

This 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*

Descriptive Form: **hardrelease** *button_name*

Scope: function playback, remote operation

See also: **hardkey, hardpress**

hardrelease indicates that the specified front panel button has been released. Front-panel button names are listed in Table A-3. When specifying button names in functions or remote operation commands, it is important to match the name shown in Table A-3 EXACTLY, including upper or lower case.

Example:

```
VM700A> hardrelease Display
```

This example releases the Display button on the front panel. See also the example accompanying the explanation of **hardpress**.

KNOB

Syntax Form: **knob** *integer*

Descriptive Form: **knob** *number_of_clicks*

Scope: function playback, remote operation

See also: **cknob**

knob turns 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:

```
VM700A> knob -50
```

This command turns the knob 50 clicks counterclockwise.

LOOP

Syntax Form: `loop`

Descriptive Form: `loop`

Scope: function playback

loop restarts a function from its first statement. Functions containing a loop statement will only exit if they contain a branch that leads to a **return** statement, or if the Function button is pressed on the front panel, or if the VM700A is powered down.

Example:

```
execute Waveform appstart
   appstart
delay 50
execute Vector appstart
   appstart
delay 50
execute H_Timing appstart
   appstart
delay 50
loop
```

This example function cycles through Waveform, Vector, and H_Timing at five-second intervals. Function execution can be stopped by pressing the Function button on the front panel.

PLAYBACK

Syntax Form: **playback** *string*

Descriptive Form: **playback** *function_name*

Scope: function playback, remote operation

The **playback** command calls and executes a function. The default path for the function is /nvram0/FunctionKeys. If the function is contained in a subdirectory of /nvram0/FunctionKeys, that subdirectory myst be included in the **playback** command's argument.

Example:

```
VM700A> playback my_dir/junk
```

This example plays back a function named "junk" in directory "my_dir", which is a subdirectory of /nvram0/FunctionKeys.

PRINT

Syntax Form: `print` *`string`*

Descriptive Form: `print` *`VM700A_file`*

Scope: function playback, remote operation

See also: `spool`

print sends a formatted file to the print spooler. The file's format is specified by the Format item for the Copy port in the Communications Setup file . The default path for the file is the Measurement~Results directory, but other files can be specified with a full pathname or a path relative to the Measurement~Results directory. Compare **spool**.

Example:

```
VM700A> print ChromLum~GainDelay
```

This examples sends the file ChromLum~GainDelay from directory Measurement~Results to the print spooler, in the format currently specified by the Format item for the Copy port in the Communications Setup file.

```
VM700A> 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 VM700A units equipped with both Option 01, NTSC, and Option 11, PAL.)

QUERY

Syntax Form: `query` *`string`*

Descriptive Form: `query` *`keyword`*

Scope: remote operation

query returns information about the VM700A keyword used as its argument. In computer mode, 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 values that can be used as an argument are listed.

In terminal mode, the query command also tells you whether or not the keyword is channel-specific (i.e., requires a channel-letter when used in a **get** or **set** command)

Examples:

(terminal mode)

```
VM700A> query VSCA
channel independent
field 1 is a file name. Possible names are:
System~Default
VM700A> query LZCL
channel specific
field 1 is an integer value with range 1 to 625
VM700A> query SP0A
channel independent
field 1 is a string. Possible strings are:
none SLIP
```

(computer mode)

```
VM700A> query VSCA
F1: file list:
 System~Default
VM700A> query LZCL
F1: integer 1 625
VM700A> query SP0A
F1: string list:
 None
 SLIP
VM700A>
```

QUIT

Syntax Form: `quit`

Descriptive Form: `quit`

Scope: remote operation

See also: `exit`

quit terminates the remote session and closes the connection between the VM700A and the computer or terminal. **quit** takes no arguments. **quit** is equivalent to **exit**.

REMOTE

Syntax Form: `remote`

Descriptive Form: `remote`

Scope: remote operation

See also: `computer, terminal`

remote initiates remote control in non-protocol mode only. You must enter **remote** before any other remote command is valid. **remote** takes no arguments.

RENAME

Syntax Form: **rename** *string string*

Descriptive Form: **rename** *old_filename new_filename*

Scope: function playback, remote operation

rename changes 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 could then compare the two results files visually, or transfer their contents to a computer for automated comparison.

Example:

```
VM700A> rename H_Timing H_Timing_1
```

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

```
execute H_Timing
delay 50
getresults
rename H_Timing H_Timing_old
delay 18000
getresults
```

This sequence of function commands starts the H_Timing application, delays five seconds, then records measurement results from the application. It then changes the name of the measurement results file from "H_Timing" to "H_Timing_old". It then waits one half-hour, and records measurement results from the application again. The two measurement results files, "H_Timing" and "H_Timing_old", can then be compared for differences.

RES

Syntax Form: **res** [**-v**] [*number*[*number*...]]

Descriptive Form: **res** [**-v**] [*position-of-result*[*position-of-result*...]]

Scope: remote operation

See also: **resoff, reson**

res returns an encoded or human-readable ASCII string containing the result of the measurement executing on the VM700A. **res** can be used whenever a measurement that returns a result is executing on the VM700A. (Most measure-mode measurements return results.)

The -v option tells **res** to produce the "verbose" (human-readable) form of the output. The default is the "encoded" form of the output, intended to be read by computers.

The optional *number* arguments specify the ordinal numbers of measurement results desired. For example, if a measurement produces seven results and you only want results 1, 2, and 5, you would issue the command "res 1 2 5".

Deciphering res Results: the -v Option

Results returned by **res** are deciphered differently, depending on whether or not the -v option is present in the command line.

Consider the following command sequence:

```
execute H_Timing
getresults
show H_Timing
```

This sequence of commands transfers the H_Timing measurement results file out the VM700A's remote port, in the following form:

```
  Measurement Results           Channel B          Tue Jul 30 11:16:12

H Timing
Line =  17
Average Off
-----------------------------------------------------------------------
Sync Rise Time           255.8   n sec
Sync Fall Time           260.0   n sec
Sync Width                 4.69  u sec
Sync Level               301.2   mV
Burst Level              306.0   mV
Sync to Burst Start        5.52  u sec
Burst Width                2.19  u sec
-----------------------------------------------------------------------
```

The equivalent result, generated by the following command sequence using the -v option:

```
execute H_Timing
res -v
```

looks like this:

```
18 1:255841 2:260041 3:46944 4:301249 5:306049 6:55244 7:21944
```

This result is interpreted as follows:

- The first number (18) uniquely identifies the measurement application that generated the result. The H_Timing measurement's identification number is 18.
- The second and succeeding numbers have the form XX:YYYYYY. The digit(s) to the left of the colon signify the data item number. In the H_Timing measurement, 1 is the sync rise time value, 2 is the sync fall time value, etc. The digits to the right of the colon indicate the measurement result, encoded as follows:
  - If D is the number of digits in the measurement result, the leftmost D-2 digits specify the mantissa of the measurement result, normalized to a number between 0 and 10 (preceded by a minus sign if the mantissa is negative). For example, if the measurement result is 301149, the mantissa of the number is 3.011.
  - The rightmost two digits specify the power of ten by which the mantissa is multiplied, plus fifty. Thus, if the multiplier is 10^{-6} ("micro"), the rightmost two digits in the data item are 44. If the multiplier is 10^{3} ("kilo"), the rightmost two digits in the data item are 53.

- If N is the total number of measurement results returned, the total number of data items sent is always equal to 2N +1: one for the measurement's identifying number, followed by an ordinal number and the data value for each measurement result.
- An asterisk character ('*') returned as data indicates that a measurement result is unavailable. This happens, for instance, when the VM700A is unable to find or recognize a signal feature necessary for a particular measurement.

Deciphering "res" Results: No -v Option

When used without the -v option, **res** returns an encoded string containing the same information as the string returned by the -v option, but in compressed form. Each byte of the human-readable message maps to a nibble (i.e., a half-byte; four bits) in the encoded message, as shown in Table A-4.

Table A-4
Encoding Scheme for res
Command Results

ASCII Character	Hexadecimal Encoding
'0'-'9'	0-9
Space	A
-	B
*	C
End-of-Buffer	D

Note that the colon character does not appear in the table. This is because colon characters are only output as visual delimiters for the human-readable form of the measurement result. In the encoded form, only space characters are output as delimiters.

If an odd number of nibbles encode a measurement set, the nibble following the end-of-buffer nibble is filled with another end-of-buffer nibble.

Notes about interpreting res results

res only reports results that are in the application's current "measurement loop". This means that res may not return all results that may be contained in a measurement results file on each pass. Results that are returned, however, are uniquely identified so that a computer program (or a human reader) can always disambiguate them.

Example:

```
VM700A>execute V_Blank
VM700A>hardkey Menu
VM700A>softkey Equalizer_Pulse
VM700A>hardkey Freeze
VM700A>getresults
V_Blank
VM700A>show V_Blank
  Measurement Results           Channel A          Fri Aug 09 12:39:59

V Timing Measurement
Average Off
----------------------------------------------------------------------
Equalizer Pulse           2.29  u sec
Equalizer Pulse 10%       2.42  u sec
Serration Pulse           4.75  u sec
Serration Pulse 10%       4.62  u sec
----------------------------------------------------------------------
VM700A>res -v
28 1:22944 2:24244
VM700A>softkey Serration_Pulse
VM700A>hardkey Freeze
VM700A>hardkey Freeze
VM700A>getresults
V_Blank
VM700A>show V_Blank
  Measurement Results           Channel A          Fri Aug 09 12:41:26

V Timing Measurement
Average Off
----------------------------------------------------------------------
Equalizer Pulse           2.29  u sec
Equalizer Pulse 10%       2.42  u sec
Serration Pulse           4.75  u sec
Serration Pulse 10%       4.62  u sec
----------------------------------------------------------------------
VM700A>res -v
@28 3:47544 4:46244
```

The example shown above runs the V_Blank application and brings up the Equalizer Pulse display. **getresults** returns four distinct data values: two for equalizer pulse data, and two for serration pulse data. **res**, however, returns only the two equalizer pulse values (labeled "1:" and "2:", respectively).

Bringing up the Serration Pulse display and invoking **getresults** returns four distinct data values. Executing **res** at this time, however, returns only the two serration pulse values (labeled "3:" and "4:", respectively).

RESOFF

Syntax Form:	`resoff`
Descriptive Form:	`resoff`
Scope:	remote operation
See also:	`res, reson`

resoff tells the VM700A to stop sending measurement results out its remote port. It is used to "turn off" a previous reson command.

RESON

Syntax Form:	`reson`
Descriptive Form:	`reson`
Scope:	remote operation
See also:	`res, resoff`

reson works similarly to **res**, except that **reson** sends a continuous stream of measurement results out the VM700A's remote port, instead of a single result. The stream of results is sent asynchronously, so that the VM700A can still recognize and respond to incoming commands (such as **resoff**). **reson** can be used with or without the -v option, to produce encoded or human-readable ASCII results.

Reading the VM700A reson Output Stream

When **reson** is issued, the VM700A begins sending results data out the remote port. At this point, there are, in effect, two separate data streams coming from the VM700A. The first is the "normal" data stream for remote control, consisting of prompts, echoed commands, viewed file output, and other output generated by remote control requests. For example:

```
VM700A> execute ColorBar
VM700A> getresults
Results are available in file 'ColorBar'.
VM700A>
```

In this example output, the prompts and the "Results" message were generated by the VM700A in response to a specific input request. By contrast, the results data produced while the VM700A is executing **reson** are sent *asynchronously*, without any specific request after the original **reson** command. For example:

```
VM700A> reson
VM7{results data}00A> {results data}{results data}...
```

Notice that in this example some of the results data arrived in the middle of the VM700A prompt string, making the VM700A output appear garbled. In order to separate the two data streams, you must know how they were combined.

When the two data streams are encoded, all characters in the "normal" output data stream are printable ASCII characters in the range 0 to 127, inclusive. By contrast, the initial character for the measurement results data stream is a non-printable character, with a decimal value of at least 127. To be specific, the initial character of a data packet produced by **reson** has decimal value 182 (hex value B6); the initial character of a data packet produced by **reson -v** has decimal value 183 (hex value B7).

Thus, a computer program reading the output data stream from the VM700A need only compare each received character with the value 127. Any character greater than 127 introduces a measurement results data packet, and a handler function should be called to process the incoming data. The handler function should return when the end-of-buffer nibble (hex value 13) is encountered.

RESTORECONFIG

Syntax Form: `restoreconfig`

Descriptive Form: `restoreconfig`

Scope: function playback, remote operation

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

RETURN

Syntax Form: `return`

Descriptive Form: `return`

Scope: function playback

See also: `stop`

return exits the currently executing function and returns to its calling function. Compare **stop**.

RGOFF

Syntax Form: `rgoff`

Descriptive Form: `rgoff`

Scope: remote operation

See also: `rgon`

rgoff takes the VM700A out of remote graphics mode. This command is only of interest to someone writing remote applications for the VM700A.

RGON

Syntax Form: `rgon`

Descriptive Form: `rgon`

Scope: remote operation

See also: `rgoff`

rgon puts the VM700A into remote graphics mode. This command is only of interest to someone writing remote applications for the VM700A.

SET

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

Descriptive Form: **set** *keyword* [A|B|C] *value* [*value*...]

Scope: function playback, remote operation

See also: `get`

set defines the configuration values to be used during the remote session. The keywords available to use with **set** are listed in Appendix B. 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 the system line and other global variables can be changed with **set** but are not restored with **restoreconfig**.

Example:

```
VM700A> set GLN5 100
VM700A> set GLN6 18
VM700A> 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*

Descriptive Form: setclock *date_and_time_string*

Scope: remote operation

See also: `getclock`

setclock sets the date and time on the system clock, using the following format:

```
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:

```
VM700A> setclock Aug 11 17:07:22 1989
```

SHOW

Syntax Form: **show** *string*

Descriptive Form: **show** *filename*

Scope: remote operation

show returns the contents of a specified file. The default path for the file is the Measurement~Results directory, but other files can be specified by supplying a full pathname or a path relative to the Measurement~Results directory.

Example:

```
VM700A> show /nvram0/ConfigFiles/Video~Source
     The default file for video sources
  Channel A Video Source:     xmiter
  Channel B Video Source:     switcher
  Channel C Video Source:     System~Default
```

This example returns the contents of file Video~Source in directory /nvram0/ConfigFiles.

SOFTKEY

Syntax Form: **softkey** *string*

Descriptive Form: **softkey** *softkey_name*

Scope: function playback, remote operation

See also: **softpress, softrelease**

softkey is equivalent to pressing and releasing a specified softkey. **softkey** has the same effect as **softpress** followed by **softrelease**; however, in general **softkey** should be used instead of these commands.

When specifying softkey names in functions or remote operation commands, it is important to match the case (upper or lower) and any special characters used in the softkey's name. See "Naming Conventions," near the end of this chapter, for information about the rules for specifying softkey names.

Example:

```
VM700A> softkey ITS_Search
```

This example is equivalent to pressing and releasing a softkey labeled "ITS Search" on the touch screen.

SOFTPRESS

Syntax Form: **softpress** *string*

Descriptive Form: **softpress** *softkey_name*

Scope: function playback, remote operation

See also: **softkey, softrelease**

softpress is equivalent to pressing a specified softkey without releasing it. The softkey remains "pressed" until a **softrelease** or **touchrelease** command is received. **softpress** and **softrelease** should be used with softkeys that need to be held while another action takes place. For all other softkeys, use **softkey**. Note also that if a softkey's function toggles, you can use **softpress** again to turn it off.

When specifying softkey names in functions or remote operation commands, it is important to match the case (upper or lower) and any special characters used in the softkey's name. See "Naming Conventions," near the end of this chapter, for information about the rules for specifying softkey names.

Example:

```
VM700A> softpress Rescale
```

This command is equivalent to pressing a softkey labeled "Rescale" without releasing it.

SOFTRELEASE

Syntax Form: **softrelease**

Descriptive Form: **softrelease**

Scope: function playback, remote operation

See also: **softkey, softpress**

softrelease indicates that a softkey has been released.

SPOOL

Syntax Form: **spool** *string*

Descriptive Form: **spool** *filename*

Scope: function playback, remote operation

See also: **print**

spool sends an unformatted, ASCII-text file to the print spooler. The default path for the file is the Measurement~Results directory, but other files can be specified with a full pathname or a path relative to the Measurement~Results directory. Compare **print**.

Example:

```
VM700A> spool ChromLum~GainDelay
```

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

Example:

```
VM700A> spool ChromLum~GainDelay
```

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

```
VM700A> spool /rom/ConfigFiles/Measurement~Locations/NTSC/System~Default
```

This example prints the default Measurement Locations file for the NTSC standard, in ASCII-text format. (Note that the path shown only applies to VM700A units equipped with both Option 01, NTSC, and Option 11, PAL.)

STOP

Syntax Form:	`stop`
Descriptive Form:	`stop`
Scope:	function playback
See also:	**`return`**

stop exits all functions in the current call sequence.

TERMINAL

Syntax Form:	`terminal`
Descriptive Form:	`terminal`
Scope:	remote operation
See also:	**`computer`**, **`remote`**

terminal sets the VM700A to be controlled from a terminal. **terminal** is used only in no-protocol mode. When **terminal** is specified, characters are echoed, there is a space following the prompt, and any errors or messages are returned as text . Compare **computer**.

TOUCHPRESS

Syntax Form:	touchpress *number number*
Descriptive Form:	touchpress *x_coordinate y_coordinate*
Scope:	function playback, remote operation
See also:	**`touchrelease`**

touchpress 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 softkey, the softkey is executed.

Example:

```
VM700A> 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`

Descriptive Form: `touchrelease`

Scope: function playback, remote operation

See also: `softpress, touchpress`

touchrelease indicates a softkey or touchscreen location has been released.

? AND ?! CONSTRUCTS

The ? construct is followed by any configuration keyword and a single command. During function playback, if the keyword has non-zero value, the rest of the line following the ? construct is executed. Otherwise,function execution continues with the next statement in the function.

The ?! construct is similar, except that the rest of the line is executed if the keyword has a value of zero. Otherwise,function execution continues with the next statement in the function.

The ? and ?! constructs are available only during function playback. They are typically used to test the value of a Carrier Detect line in order to control communications with a modem, or to test the global out-of-limits (GOOL) flag to see if an application found one or more measurements that were out-of-limits.

Using ? and ?! With the Carrier Detect Flag

Syntax Form: ?GCD0 *string* | ?GCD1 *string* |
?!GCD0 *string* | ?!GCD1 *string*

Descriptive Form: ?GCD0 *command* | ?GCD1 *command*
?!GCD0 *command* | ?!GCD1 *command*

Scope: function playback

?GCDO tests the status of the Carrier Detect line on Port 0. **?GCD1** tests the status of the Carrier Detect line on Port 1. If the value returned is 1 (true), then *command* is executed. Otherwise, function execution continues with the next statement in the function.

?!GCDO and **?!GCD1** function similarly, except that *command* is executed if the value returned is 0 (false).

?GCDO, ?GCD1, ?!GCD0, and **?!GCD1** are primarily used to control communication with a modem connected to the VM700A.

Example:

```
?GCD0 return
delay 5
loop
```

This first line of this example tests the status of the Carrier Detect line on serial port 0. If the test returns true, the function returns to its caller. If it returns false, the function waits one half-second, then loops back to the first command.

Using ? and ?! With the Global Out-of-Limits Flag

Syntax Form: `?GOOL` *`string`* | `?!GOOL` *`string`*

Descriptive Form: `?GOOL` *`command`* | `?!GOOL` *`command`*

Scope: function playback

?GOOL returns the status of the global out-of-limits flag. If the most recently executed application fould one or more measurements that were out-of-limits, this flag is set to TRUE. If the global out-of-limits flag returns 1 (true) when **?GOOL** is executed, then *command* is executed. Otherwise, function execution continues with the next statement in the function.

?!GOOL functions similarly, except that *command* is executed if the value returned is 0 (false).

The GOOL flag must be reset to 0 explicitly within a function. It is a good programming practice to do this just before executing a measurement application.

Example:

```
set GOOL 0
execute K_Factor
getresults
?GOOL print K_Factor
```

The first line of this sample function sets the global out-of-limits flag to 0. The second and third lines execute a measurement application (in this case, K_Factor) and write a results file from it. The last line tests to see if the global out-of-limits flag was asserted, i.e., if any measurement returned an out-of-limits value since the last time the out-of-limits flag was de-asserted. If ?GOOL returns true, then the command "print K_Factor" is executed to print out the results file.

NAMING CONVENTIONS

This section tells where to find the legal button names, and discusses the rules for forming application, softkey, and function names for use in function and remote operation commands.

Button Names

Table A-3, accompanying the description of the **hardkey** command, lists the legal button names.

Application Names

MAJOR-MODE APPLICATIONS

Major-mode applications are reachable with single-button presses in front-panel operation. These applications are Waveform, Vector, Picture, and Auto. Major-mode applications can be executed in functions or remote operation using the **execute** or **hardkey** commands, followed by the button or application name.

Example:

```
VM700A> execute Waveform
VM700A> hardkey Waveform
VM700A> execute Vector
VM700A> hardkey Vector
VM700A> execute Picture
VM700A> hardkey Picture
VM700A> execute auto
VM700A> hardkey auto
```

All of these example commands cause the major-mode applications named in them to execute.

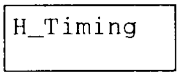
MEASURE-MODE APPLICATIONS

Measure-mode applications are the applications available when you press the Measure front-panel button. These include measure-mode video measurements and diagnostics. They may also include optional video measurements and audio measurements, if your VM700A is equipped with options.

To form the application name for a measure-mode application, use the name exactly as displayed in the directory window. If the name is split over two lines, substitute a tilde character ('~') for the carriage return between the two parts of the name.

Examples:

The H_Timing measurement, whose softkey looks like this in the video-measurements directory window:

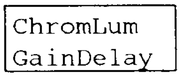


H_Timing

would be executed like this:

```
VM700A> execute H_Timing
```

The ChromLum GainDelay measurement, whose softkey looks like this in the video-measurements directory window:



ChromLum
GainDelay

would be executed like this with a remote command:

```
VM700A> execute ChromLum~GainDelay
```

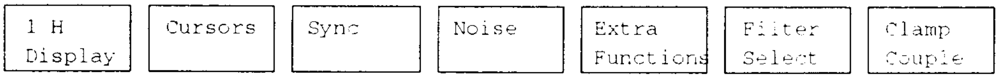
It is important to match upper and lower case and any special characters that appear in the application name *exactly*.

Softkey and Function Names

The general rule for forming a softkey name is to take the spelling and capitalization from the text displayed, omit any variable part, and join all words with an underscore character ('_').

Examples:

The default set of softkeys that appears on the screen when you press the Menu button while the VM700A is in Waveform mode looks like this:



A function that executes Waveform and presses each of these softkeys in turn is as follows:

```
execute Waveform
hardkey Menu
softkey H_Display
delay 20
softkey Cursors
delay 20
hardkey Menu
delay 10
softkey Sync
delay 20
hardkey Menu
delay 10
softkey Noise_dB
delay 10
softkey Noise_dB
delay 10
softkey Extra_Functions
delay 20
hardkey Menu
delay 10
softkey Filter_Select
delay 20
hardkey Menu
delay 10
softkey Clamp_Couple
delay 20
hardkey Menu
delay 10
hardkey Menu
```

The **delay** commands are inserted between the various **softkey** and **hardkey** commands so that you can see the effects of each command when the function is replayed.

Note that a **hardkey Menu** command follows the commands to touch the Cursors, Sync, Extra_Functions, Filter_Select, and Clamp_Couple softkeys. This is because each of these softkeys displays a sub-menu when touched. **hardkey Menu** returns you to the application's top-level menu from any sub-menu.

Note also that the Noise_dB softkey is touched twice in succession. This is because this softkey toggles on and off. The first command highlights the softkey. At this point you could change the value displayed in the softkey by means of a **knob** command. The second command turns the softkey's highlight off.

SELECT LINE SOFTKEYS

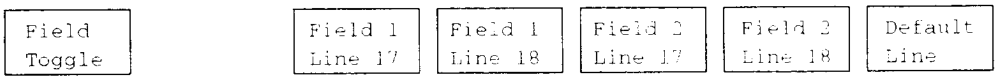
The exceptions to the softkey naming rule are the softkeys that select a system line after you press the Select Line button.

For most Measure-mode applications, the names for these softkeys are Preset*N*, where *N* varies from 1 to the number of line softkeys available.

For the Waveform, Vector, ICPM, Bowtie, and Lightning applications, however, the names for these softkeys are Preset*Nstd*, where *N* again varies from 1 to the number of softkeys available, and *std* can be either NTSC or PAL.

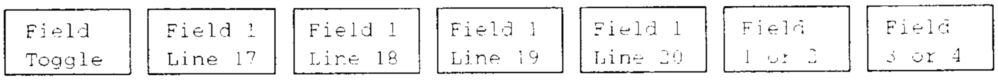
Examples:

The default softkeys that appear when you press the Select Line button in the Bar LineTime application while using the NTSC standard are:



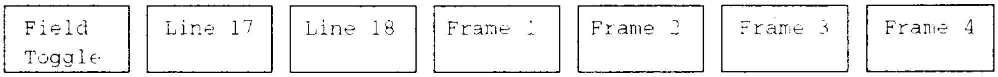
The names of these softkeys, from left to right, are: Field_Toggle, Preset1, Preset2, Preset3, Preset4, Default_Line.

The default softkeys that appear when you press the Select Line button in the Waveform application while using the NTSC standard are:



The names of these softkeys, from left to right, are: Field_Toggle, Preset1NTSC, Preset2NTSC, Preset3NTSC, Preset4NTSC, Field_1_or_2, and Field_3_or_4.

The default softkeys that appear when you press the Select Line button in the Waveform application while using the PAL standard are:



The names of these softkeys, from left to right, are: Field_Toggle, Preset1PAL, Preset2PAL, Frame_1, Frame_2, Frame_3, and Frame_4.

Table A-5 tells whether each video application uses the PresetN form or the PresetNstd form of line selection softkey names. It also lists the video applications that don't use presets in their line selection softkeys, as well as those that don't use the Select Line button.

Audio applications don't use the Select Line button.

For video applications not listed here (e.g., applications that may come in future option packages for the VM700A), the easiest way to ascertain which form of the Select Line softkey name to use is to create a function that uses those softkeys. You can then examine the function for **softpress** commands that correspond to the softkeys you wish to use. The **softpress** commands will show the correct names for the softkeys.

Table A-5
Select Line Softkey Names

Uses PresetN	Bar~LineTime ChromLum~GainDelay Chrominance~FreqResp Chrominance~NonLinearity ColorBar DGDP Echo GroupDelay~SinX_X H_Timing K_Factor Luminance~NonLinearity MultiBurst Noise~Spectrum Rounding~Errors ShortTime~Distortion SoundInSync Teletext TwoField
Uses PresetNstd	Bowtie ICPM Lightning Vector Waveform
Doesn't Use Presets	Chrominance~AMPM
Doesn't Use Select Line	Bounce Burst~Frequency H_Blank Jitter Jitter~LongTime Line~Frequency Picture SCH_Phase VITS~ID V_Blank Video~Standard

Appendix B
GET/SET KEYWORDS

This appendix documents the keywords used with the **get** and **set** commands. For each **get/set** keyword letter-group, it gives the syntax of the **set** command and the **get** result, an explanation of what the keywords in the letter-group do, and one or more examples of the use of keywords from the group. This is followed by an alphabetized table of all the keywords in the group and their meaning.

"A" Group: Audio Configuration

"A" keywords report on or set the values of Audio Option Configuration parameters. These keywords can only be used on a VM700A equipped with Option 40 (Audio).

Get commands used with the "A" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with "A" group keywords take different arguments, depending on the keyword. The form of each Set command used with "A" group keywords is documented in the pages that follow.

Table B-1
"A" Keywords: Audio Configuration

Keyword	Description	Keyword	Description
A33T	0.33 test level	ALEM	Level Meter
ADAA	Dead Air Alarm	ALIL	Lineup level
AERR	Error reporting	ALIM	Audio Limit File
AEXT	External termination	ALIS	Lissajous display
AHPT	Audio Printout Title	ARIA	Report in Auto

A33T

A33T specifies the 0.33 Test Level in dBu. The set command takes one argument after the channel letter. Legal values are integers from -6 to 14.

```
VM700A> get A33T A
0
VM700A> set A33T A 1
VM700A>
```

ADAA

ADAA specifies the amount of time the VM700A waits before it reports an error when it encounters a period of silence. The set command takes one argument after the channel letter. Legal values are: disabled, 15 sec, 30 sec, 1 min, 2 min, 5 min, 30 min, or 60 min. Disabled is the default.

```
VM700A> get ADAA A
disabled
VM700A> set ADAA A 1 min
VM700A>
```

AERR

AERR specifies whether or not audio errors for the specified audio source should be added to the video error report. The set command takes one argument after the channel letter. Legal values are enabled or disabled. Disabled is the default.

```
VM700A> get AERR A
enabled
VM700A> set AERR A disabled
VM700A>
```

AEXT

AEXT informs the VM700A about the value of the VM700A's external termination. The set command takes one argument after the channel letter. Legal values are: 50, 75, 125, 150, 300, 600, 10k.

```
VM700A> get AEXT A
600
VM700A> set AEXT 10k
```

AHPT

AHPT specifies the title that appears on audio report printouts. The set command takes one argument after the channel letter. It consists of an arbitrary string up to 50 characters in length. "#" is a comment character; all characters appearing in the string after a "#" are ignored. To include a "#" in the string, precede it by a "\" character.

```
VM700A> get AHPT A
VM700A Audio Measurements
VM700A> set AHPT A Fred's Audio Measurements
VM700A> get AHPT A
Fred's Audio Measurements
VM700A>
```

ALEM

ALEM specifies the meter ballistics for the bar-graph level meters in the Audio Monitor display. The set command takes one argument after the channel letter. Legal values are: PPM:DIN45406, PPM:NORDIC, PPM:TEK760[1], or VU. The default is PPM:DIN45406.

```
VM700A> get ALEM A
PPM:DIN4506
VM700A> set ALEM A PPM:TEK760
VM700A> get ALEM A
PPM:TEK760
VM700A>
```

[1] Similar to the Tektronix 760 meter, but with its -8 dB tick mark labeled "TEST." The lineup level is at the TEST (-8 dB) tick on this meter. Ballistics are the same as for the DIN 45406 meter.

ALIL

ALIL specifies the meter equivalency in dBu for the VU and dBu scales used in the Audio Monitor display. The set command takes one argument after the channel letter. Legal values are integers from -10 to 10. The default is 0.

```
VM700A> get ALIL A
0
VM700A> set ALIL A 1
VM700A>
```

ALIM

ALIM specifies the Audio Limit File used by the VM700A. The set command takes one argument, the name of the file, after the channel letter. This file is found in directory /nvram0/ConfigFiles/Audio_Limit~Files.

```
VM700A> get ALIM A
System~Default
VM700A> set ALIM A ShortHaul
VM700A> get ALIM A
ShortHaul
VM700A>
```

ALIS

ALIS specifies the type of Lissajous display used in the Audio Monitor. The set command takes one argument after the channel letter. Legal values are x/y or soundstage. The default is soundstage.

```
VM700A> get ALIS A
soundstage
VM700A> set ALIS A x/y
VM700A> get ALIS A
x/y
VM700A>
```

ARIA

ARIA tells the VM700A whether or not to display an audio report in Video Auto mode. The set command takes one argument after the channel letter. Legal values are enabled or disabled. Disabled is the default.

```
VM700A> get ARIA A
disabled
VM700A> set ARIA A enabled
disabled
VM700A> get ARIA A
enabled
VM700A>
```

"B" Group: Audio Limit Files

"B" keywords report on or set the measurement limits for Audio auto Mode measurements. These keywords can only be used on a VM700A equipped with Option 40 (Audio).

Get commands used with the "B" keywords have the form:

```
get <keyword> <channel-letter>
```

Set commands used with "B" group keywords take different arguments, depending on the keyword. The form of each Set command used with "B" group keywords is documented in this section.

Table B-2
"B" Keywords: Audio Limit Files

Keyword	Description	Keyword	Description
BARF	Amplitude response frequency breaks	BHDF	Total harmonic distortion frequency breaks
BARL	Amplitude response lower limit	BHDL	Total harmonic distortion lower limit
BARU	Amplitude response upper limit	BHDU	Total harmonic distortion upper limit
BCEF	Compander error (fall)	BING	Insertion gain
BCER	Compander error (rise)	BSNR	Signal to noise ratio
BCRT	Crosstalk plus noise	BSPF	Stereo phase difference frequency breaks
BGDF	Stereo gain difference frequency breaks	BSPL	Stereo phase difference lower limit
BGDL	Stereo gain difference lower limit	BSPU	Stereo phase difference upper limit
BGDU	Stereo gain difference upper limit		

BARF

BARF sets the frequencies of the break points for the amplitude response vs. frequency measurement. The set command takes eight arguments after the channel letter, each representing the frequency (in Hz) of a break point. Each argument must be an integer between 20 and 20000, inclusive.

```
VM700A> get BARF A
20  50  250  251  1000  10000  10001  20000
VM700A> set BARF A 30 60 300 301 1500 15000 15001 20000
VM700A> get BARF A
30  60  300  301  1500  15000  15001  20000
VM700A>
```

BARL

BARL sets the lower amplitude limits of the break points for the amplitude response vs. frequency measurement. The set command takes eight arguments after the channel letter, each representing the lower limit (in dB) of a break point. Each number must be between -120 and 120, inclusive.

```
VM700A> get BARL A
-1.00  -1.00  -1.00  -0.30  -0.30  -0.30  -1.00  -1.00
VM700A> set BARL A -0.5 -0.5 -0.5 0 0 0 -0.5 -0.5
VM700A> get BARL A
-0.5  -0.5  -0.5  0  0  0  -0.5  -0.5
VM700A>
```

BARU

BARU sets the upper amplitude limits of the break points for the amplitude response vs. frequency measurement. The set command takes eight arguments after the channel letter, each representing the upper limit (in dB) of each break point. Each number must be between -120 and 120, inclusive.

```
VM700A> get BARU A
0.30  0.30  0.30  0.30  0.30  0.30  0.30  0.30
VM700A> set BARU A 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
VM700A> get BARU A
0.50  0.50  0.50  0.50  0.50  0.50  0.50  0.50
VM700A>
```

BCEF

BCEF specifies the lower and upper limits in decibels for the difference between the measured compander response of the signal and the expected response on a high-to-low transition. The set command takes two arguments after the channel letter, representing the lower and upper limits, respectively. Legal values are floating point numbers in the range -120 to 120, inclusive, or "undef", which indicates no limit set.

```
VM700A> get BCEF A
---  0.50
VM700A> set BCEF A undef  0.75
VM700A> get BCEF A
---  0.75
VM700A>
```

BCER

BCER specifies the lower and upper limits in decibels for the difference between the measured compander response of the signal and the expected response on a low-to-high transition. The set command takes two arguments after the channel letter, representing the lower and upper limits, respectively. Legal values are floating point numbers in the range -120 to 120, inclusive, or "undef", which indicates no limit set.

```
VM700A> get BCER A
---   0.50
VM700A> set BCER A undef  0.75
VM700A> get BCER CA
---   0.75
VM700A>
```

BCRT

BCRT specifies the lower and upper limits in decibels for the crosstalk plus noise measurement. The set command takes two arguments after the channel letter, representing the lower and upper limits, respectively. Legal values are floating point numbers in the range -120 to 0, inclusive, or "undef", which indicates no limit set.

```
VM700A> get BCRT A
---   -50.00
VM700A> set BCRT A undef  -75.00
VM700A> get BCRT A
---   -75.00
VM700A>
```

BGDF

BGDF sets the frequencies of the break points for the stereo gain difference measurement. The set command takes eight arguments after the channel letter, each representing the frequency (in Hz) of a break point. Each argument must be an integer between 20 and 20000, inclusive.

```
VM700A> get BGDF A
20  50  250  251  1000  10000  10001  20000
VM700A> set BGDF A 20 50 300 301 1500 15000 15001 20000
VM700A> get BGDF A
20  50  300  301  1500  15000  15001  20000
VM700A>
```

BGDL

BGDL sets the lower amplitude limits of the break points for the stereo gain difference measurement. The set command takes eight arguments after the channel letter, each representing the lower limit (in dB) of a break point. Each number must be between -120 and 120, inclusive.

```
VM700A> get BGDL A
-1.00  -1.00  -1.00  -0.30  -0.30  -0.30  -1.00  -1.00
VM700A> set BGDL A -1 -0.5 -0.5 -0.3 -0.3 -0.3 -0.5 -1
VM700A> get BGDL A
-1.00  -0.5  -0.5  -0.30  -0.30  -0.30  -0.5  -1.00
VM700A>
```

BGDU

BGDU sets the upper amplitude limits of the break points for the stereo gain difference measurement. The set command takes eight arguments after the channel letter, each representing the upper limit (in dB) of a break point. Each number must be between -120 and 120, inclusive.

```
VM700A> get BGDU A
0.30  0.30  0.30  0.30  0.30  0.30  0.30  0.30
VM700A> set BGDU A 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
VM700A> get BGDU A
0.50  0.50  0.50  0.50  0.50  0.50  0.50  0.50
VM700A>
```

BHDF

BHDF sets the frequencies of the break points for the harmonic distortion measurement. The set command takes eight arguments after the channel letter, each representing the frequency (in Hz) of a break point. Each argument must be an integer between 20 and 20000, inclusive.

```
VM700A> get BHDF A
20  50  200  400  1000  2000  4000  6000
VM700A> set BHDF A 20 50 300 500 1200 2400 4800 9600
VM700A> get BHDF A
20  50  300  500  1200  2400  4800  9600
VM700A>
```

BHDL

BHDL sets the lower amplitude limits of the break points for the harmonic distortion measurement. The set command takes eight arguments after the channel letter, each representing the lower limit (in dB) of a break point. Each number must be between -120 and 120, inclusive.

```
VM700A> get BHDL A
0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
VM700A> set BHDL A -5 -5 -5 -5 -5 -5 -5 -5
VM700A> get BHDL A
-5.0  -5.0  -5.0  -5.0  -5.0  -5.0  -5.0  -5.0
VM700A>
```

BHDU

BHDU sets the upper amplitude limits of the break points for the harmonic distortion measurement. The set command takes eight arguments after the channel letter, each representing the upper limit (in dB) of a break point. Each number must be between -120 and 120, inclusive.

```
VM700A> get BHDU A
0.30  0.30  0.30  0.30  0.30  0.30  0.30  0.50
VM700A> set BHDU A 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
VM700A> get BHDU A
0.50  0.50  0.50  0.50  0.50  0.50  0.50  0.50
VM700A>
```

BING

BING specifies the lower and upper limits in decibels for the difference between the actual and the expected audio signal level. Legal values are floating point numbers in the range -120 to 120, inclusive, or "undef", which indicates no limit set.

```
VM700A> get BING A
-0.50  0.50
VM700A> get BING A -0.75 0.75
VM700A> get BING A
-0.75  0.75
VM700A>
```

BSNR

BSNR specifies the lower and upper limits in decibels for both the weighted and unweighted signal-to-noise ratios. Legal values are floating point numbers in the range -120 to 120, inclusive, or "undef", which indicates no limit set.

```
VM700A> get BSNR A
70.00  ---
VM700A> set BSNR A 75.00 undef
VM700A> get BSNR A
75.00  ---
VM700A>
```

BSPF

BSPF sets the frequencies of the break points for the stereo phase measurement. The set command takes eight arguments after the channel letter, each representing the frequency (in Hz) of a break point. Each argument must be an integer between 20 and 20000, inclusive.

```
VM700A> get BSPF A
20  50  200  400  1000  2000  4000  6000
VM700A> set BSPF A 20 50 300 500 1200 2400 4800 9600
VM700A> get BSPF A
20  50  300  500  1200  2400  4800  9600
VM700A>
```

BSPL

BSPL sets the amplitudes of the lower limits at the break points for the stereo phase difference measurement. The set command takes eight arguments after the channel letter, each representing the lower limit (in dB) at a break point. Each number must be between 0 and 180, inclusive.

```
VM700A> get BSPL A
0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
VM700A> set BSPL A 1 1 1 1 1 1 1 1
VM700A> get BSPL A
1.00  1.00  1.00  1.00  1.00  1.00  1.00  1.00
VM700A>
```

BSPU

BSPU sets the amplitudes of the upper limits at the break points for the stereo phase difference measurement. The set command takes eight arguments after the channel letter, each representing the upper limit (in dB) at a break point. Each number must be between 0 and 180, inclusive.

```
VM700A> get BSPU A
6.00  6.00  3.00  3.00  3.00  3.00  6.00  6.00
VM700A> set BSPU A 6 6 4.5 3 3 4.5 6 6
VM700A> get BSPU A
6.00  6.00  4.50  3.00  3.00  4.50  6.00  6.00
VM700A>
```

"C" Group: Configuration File

"C" keywords report on or set the values of Configuration File parameters. These keywords can only be used on VM700A's equipped with Option 11 (PAL standard).

Get commands used with the "C" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with the "C" keywords have the form

```
set <keyword> <channel-letter> <argument>
```

where <argument> is the text to which the setting associated with the keyword is to be set.

Table B-3
"C" Keywords: Configuration File (PAL)

Keyword	Description	Keyword	Description
CHAF	Auto Limits File	CHSN	Source Name
CHMF	Measure Limits File	CHSS	Sync Source
CHML	Measurements Location File	COCP	Component option config file for PAL
CHPT	Printout Title	COER	Echo & Rounding option config file for PAL
CHSM	Selected Measurements File	COTT	Teletext option config file for PAL

CHAF

CHAF specifies the Auto Limits File for PAL operation. This file is found in directory /nvram0/ConfigFiles/Auto_Limit~Files/PAL.

```
VM700A> get CHAF A
System~Default
VM700A> set CHAF A My_Auto_Limits
VM700A> get CHAF A
My_Auto_Limits
VM700A>
```

CHMF

CHMF specifies the Measure Limits File for PAL operation. This file is found in directory /nvram0/ConfigFiles/Measure_Limit~Files/PAL.

```
VM700A> get CHMF A
System~Default
VM700A> set CHMF A My_Meas_Limits
VM700A> get CHMF A
My_Meas_Limits
VM700A>
```

CHML

CHML specifies the Measurements Location File for PAL operation. This file is found in directory /nvram0/ConfigFiles/Measurement~Locations/PAL.

```
VM700A> get CHML A
System~Default
VM700A> set CHML A My_Meas_Locs
VM700A> get CHML A
My_Meas_Locs
VM700A>
```

CHPT

CHPT specifies the title that appears on printouts generated by the VM700A. It consists of an arbitrary string up to 50 characters in length. "#" is a comment character; all characters appearing in the string after a "#" are ignored. To include a "#" in the string, precede it by a "\" character.

```
VM700A> get CHPT A
VM700A Video Measurement Set
VM700A> set CHPT A Yahoo!
VM700A> get CHPT A
Yahoo!
VM700A>
```

CHSM

CHSM specifies the Selected Measurements File for Auto Mode PAL-standard operation. This file is found in directory /nvram0/ConfigFiles/Selected~Measurements/PAL.

```
VM700A> get CHSM A
System~Default
VM700A> set CHSM A My_Sel_Meas
VM700A> get CHSM A
My_Sel_Meas
VM700A>
```

CHSN

CHSN specifies the source name for PAL-standard operation. This name is printed across the top of printouts generated by Auto mode or by pressing the Copy button. It consists of an arbitrary string up to 32 characters in length. "#" is a comment character; all characters appearing in the string after a "#" are ignored. To include a "#" in the string, precede it by a "\" character.

```
VM700A> get CHSN A
System Default
VM700A> set CHSN A Our_Transmitter
VM700A> get CHSN A
Our_Transmitter
VM700A>
```

CHSS

CHSS specifies the synchronization source for PAL-standard operation. This value can be set to one of the following strings (case-sensitive): Channel A, Channel B, Channel C, External, or Locked to Source.

```
VM700A> get CHSS A
Locked to Source
VM700A> set CHSS A Channel A
VM700A> get CHSS A
Locked to Source
VM700A>
```

COCP

COCP specifies the configuration file for the Component option in PAL-standard operation. This file is found in directory /nvram0/ConfigFiles/Component/PAL. This keyword is only valid for PAL-standard VM700A's with Option 30 (Component) installed.

```
VM700A> get COCP A
System Default
VM700A> set COCP A My_Comp_File
VM700A> get COCP A
My_Comp_File
VM700A>
```

COER

COER specifies the configuration file for the Echo & Rounding option in PAL-standard operation. This file is found in directory /nvram0/ConfigFiles/Echo_Rounding/PAL. This keyword is only valid for PAL-standard VM700A's with Option 1G (Echo/Rounding) installed.

```
VM700A> get COER A
System Default
VM700A> set COER A My_ER_File
VM700A> get COER A
My_ER_File
VM700A>
```

COTT

COTT specifies the configuration file for the Teletext option in PAL-standard operation. This file is found in directory /nvram0/ConfigFiles/Teletext/PAL. This keyword is only valid for PAL-standard VM700A's with Option 20 (Teletext) installed.

```
VM700A> get COTT A
System Default
VM700A> set COTT A My_TT_File
VM700A> get COTT A
My_TT_File
VM700A>
```

"D" Group: Configuration File

"D" keywords report on or set the values of Configuration File parameters. These keywords can only be used on VM700A's equipped with Option 01 (NTSC standard).

Get commands used with the "D" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with the "D" keywords have the form

```
set <keyword> <channel-letter> <argument>
```

where <argument> is the text to which the setting associated with the keyword is to be set.

Table B-4
"D" Keywords: Configuration File (NTSC)

Keyword	Description	Keyword	Description
DHAF	Auto Limits File	DHSN	Source Name
DHMF	Measure Limits File	DHSS	Sync Source
DHML	Measurements Location File	DOCP	Component option config file for NTSC
DHPT	Printout Title	DOER	Echo & Rounding option config file for NTSC
DHSM	Selected Measurements File	DOTT	Teletext option config file for NTSC

DHAF

DHAF specifies the Auto Limits File for NTSC operation. This file is found in directory /nvram0/ConfigFiles/Auto_Limit~Files/NTSC.

```
VM700A> get DHAF A
System~Default
VM700A> set DHAF A My_Auto_Limits
VM700A> get DHAF A
My_Auto_Limits
VM700A>
```

DHMF

DHMF specifies the Measure Limits File for NTSC operation. This file is found in directory /nvram0/ConfigFiles/Measure_Limit~Files/NTSC.

```
VM700A> get DHMF A
System~Default
VM700A> set DHMF A My_Meas_Limits
VM700A> get DHMF A
My_Meas_Limits
VM700A>
```

DHML

DHML specifies the Measurements Location File for NTSC operation. This file is found in directory /nvram0/ConfigFiles/Measurement~Locations/NTSC.

```
VM700A> get DHML A
System~Default
VM700A> set DHML A My_Meas_Locs
VM700A> get DHML A
My_Meas_Locs
VM700A>
```

DHPT

DHPT specifies the title that appears on printouts generated by the VM700A. It consists of an arbitrary string up to 50 characters in length. "#" is a comment character; all characters appearing in the string after a "#" are ignored. To include a "#" in the string, precede it by a "\" character.

```
VM700A> get DHPT A
VM700A Video Measurement Set
VM700A> set DHPT A Yahoo!
VM700A> get DHPT A
Yahoo!
VM700A>
```

DHSM

DHSM specifies the Selected Measurements File for Auto Mode NTSC-standard operation. This file is found in directory /nvram0/ConfigFiles/Selected~Measurements/NTSC.

```
VM700A> get DHSM A
System~Default
VM700A> set DHSM A My_Sel_Meas
VM700A> get DHSM A
My_Sel_Meas
VM700A>
```

DHSN

DHSN specifies the source name for NTSC-standard operation. This name is printed across the top of printouts generated by Auto mode or by pressing the Copy button. It consists of an arbitrary string up to 32 characters in length. "#" is a comment character; all characters appearing in the string after a "#" are ignored. To include a "#" in the string, precede it by a "\" character.

```
VM700A> get DHSN A
System Default
VM700A> set DHSN A Our_Transmitter
VM700A> get DHSN A
Our_Transmitter
VM700A>
```

DHSS

DHSS specifies the synchronization source for NTSC-standard operation. This value can be set to one of the following strings (case-sensitive): Channel A, Channel B, Channel C, External, or Locked to Source.

```
VM700A> get DHSS A
Locked to Source
VM700A> set DHSS A Channel A
VM700A> get DHSS A
Locked to Source
VM700A>
```

DOCP

DOCP specifies the configuration file for the Component option in NTSC-standard operation. This file is found in directory /nvram0/ConfigFiles/Component/NTSC. This keyword is only valid for NTSC-standard VM700A's with Option 30 (Component) installed.

```
VM700A> get DOCP A
System Default
VM700A> set DOCP A My_Comp_File
VM700A> get DOCP A
My_Comp_File
VM700A>
```

DOER

DOER specifies the configuration file for the Echo & Rounding option in NTSC-standard operation. This file is found in directory /nvram0/ConfigFiles/Echo_Rounding/NTSC. This keyword is only valid for NTSC-standard VM700A's with Option 1G (Echo/Rounding) installed.

```
VM700A> get DOER A
System Default
VM700A> set DOER A My_ER_File
VM700A> get DOER A
My_ER_File
VM700A>
```

DOTT

DOTT specifies the configuration file for the Teletext option in NTSC-standard operation. This file is found in directory /nvram0/ConfigFiles/Teletext/NTSC. This keyword is only valid for NTSC-standard VM700A's with Option 20 (Teletext) installed.

```
VM700A> get DOTT A
System Default
VM700A> set DOTT A My_TT_File
VM700A> get DOTT A
My_TT_File
VM700A>
```

"E" Group: Component Configuration (NTSC)

"E" keywords report on or set the values of Component Option Configuration parameters. These keywords can only be used on NTSC-standard VM700A's equipped with Option 30 (Component).

Get commands used with the "E" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with "E" group keywords take different arguments, depending on the keyword. The form of each Set command used with "E" group keywords is documented in the pages that follow.

Table B-5
"E" Keywords: Component Configuration (NTSC)

Keyword	Description	Keyword	Description
EBC1	Pb CB color #1 (mV)	ELCP	Lightning color pk-pk ampl error (%)
EBC2	Pb CB color #2 (mV)	ELPW	Lightning pk-white ampl error (%)
EBC3	Pb CB color #3 (mV)	EMKL	Marker field and line
EBC4	Pb CB color #4 (mV)	EPRI	Probe Input
EBC5	Pb CB color #5 (mV)	ERC1	Pr CB color #1 (mV)
EBC6	Pb CB color #6 (mV)	ERC2	Pr CB color #2 (mV)
EBC7	Pb CB color #7 (mV)	ERC3	Pr CB color #3 (mV)
EBC8	Pb CB color #8 (mV)	ERC4	Pr CB color #4 (mV)
EBHA	Pb K-Factor Pulse HAD	ERC5	Pr CB color #5 (mV)
EBKB	Pb K-PB Factor (%)	ERC6	Pr CB color #6 (mV)
EBKF	Pb K Factor (%)	ERC7	Pr CB color #7 (mV)
EBM1	Pb MB Packet #1 (dB)	ERC8	Pr CB color #8 (mV)
EBM2	Pb MB Packet #2 (dB)	ERCI	Stored Reference Channel Independent
EBM3	Pb MB Packet #3 (dB)	ERHA	Pr K-Factor Pulse HAD
EBM4	Pb MB Packet #4 (dB)	ERKB	Pr K-PB Factor (%)
EBM5	Pb MB Packet #5 (dB)	ERKF	Pr K Factor (%)
EBM6	Pb MB Packet #6 (dB)	ERM1	Pr MB Packet #1 (dB)
EBM7	Pb MB Packet #7 (dB)	ERM2	Pr MB Packet #2 (dB)
EBM8	Pb MB Packet #8 (dB)	ERM3	Pr MB Packet #3 (dB)
EBM9	Pb MB Packet #9 (dB)	ERM4	Pr MB Packet #4 (dB)
EBMB	Pb Multiburst Packets	ERM5	Pr MB Packet #5 (dB)
EBMF	Pb Multiburst flag (mV)	ERM6	Pr MB Packet #6 (dB)
EBNL	Pb Non-Linearity (%)	ERM7	Pr MB Packet #7 (dB)
EBNO	Pb Noise Level (dB rms)	ERM8	Pr MB Packet #8 (dB)
EBPP	Pb P-P Amplitude (mV) (3 values)	ERM9	Pr MB Packet #9 (dB)
EBRD	Pb to Pr Delay (nsec) (3 values)	ERMB	Pr Multiburst Packets
EBWL	Bowtie field and line	ERMF	Pr Multiburst flag (mV)
EBWT	Bowtie interchannel delay (ns)	ERNL	Pr Non-Linearity (%)
EBYD	Pb to Y Delay (nsec) (3 values)	ERNO	Pr Noise Level (dB rms)
ECSD	Colorbar standard	ERPP	Pr P-P Amplitude (mV) (3 values)

Table B-5 (cont)

Keyword	Description	Keyword	Description
ERYD	Pr to Y Delay (nsec) (3 values)	EYM2	Y MB Packet #2 (dB)
ETNM	T (nsec)	EYM3	Y MB Packet #3 (dB)
EYC1	Y CB color #1 (mV)	EYM4	Y MB Packet #4 (dB)
EYC2	Y CB color #2 (mV)	EYM5	Y MB Packet #5 (dB)
EYC3	Y CB color #3 (mV)	EYM6	Y MB Packet #6 (dB)
EYC4	Y CB color #4 (mV)	EYM7	Y MB Packet #7 (dB)
EYC5	Y CB color #5 (mV)	EYM8	Y MB Packet #8 (dB)
EYC6	Y CB color #6 (mV)	EYM9	Y MB Packet #9 (dB)
EYC7	Y CB color #7 (mV)	EYMB	Y Multiburst Packets
EYC8	Y CB color #8 (mV)	EYMF	Y Multiburst flag (mV)
EYHA	Y K-Factor Pulse HAD	EYNL	Y Non-Linearity (%)
EYKB	Y K-PB Factor (%)	EYNO	Y Noise Level (dB rms)
EYKF	Y K Factor (%)	EYPA	Y Peak Amplitude (mV) (3 values)
EYM1	Y MB Packet #1 (dB)	EYSA	Y Sync Amplitude (mV) (3 values)

EBC1 - EBC8

Keywords EBC1 through EBC8 return or set the lower and upper alarm limits for Pb CB colors in the Component ColorBar application. The settings are in milliVolts. Legal values are floating point numbers from -500 to 500, inclusive. The color ordering is as follows: 1, gray; 2, yellow; 3, cyan; 4, green; 5, magenta; 6, red; 7, blue; 8, black.

```
VM700 get EBC1 A
-50.00  50.00
VM700>
```

EBHA

EBHA returns or sets the half-amplitude duration for the Pb(B) K-Factor pulse, in T units. Legal values are integers from 2 to 8, inclusive.

```
VM700 get EBHA A
7
VM700>
```

EBKB

EBKB returns or sets the lower and upper alarm limits for the Pb K-PB factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from -50 to 50, inclusive.

```
VM700 get EBKB A
-5.00  1.00
VM700>
```

EBKF

EBKF returns or sets the lower and upper alarm limits for the Pb K-factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from 0 to 99.9, inclusive.

```
VM700 get EBKF A
0.00  5.00
VM700>
```

EBM1 - EBM9

Keywords EBM1 through EBM9 return or set the lower and upper alarm limits, in dB, for Pb(B) packets 1 through 9 in the Component Multiburst application. Legal values are floating point numbers from -40 to 40, inclusive.

```
VM700 get EBM1 A
-1.00  1.00
VM700>
```

EBMB

EBMB returns or sets the number of Pb (B) multiburst packets. Legal values are integers from 3 to 9, inclusive.

```
VM700 get EBMB A
5
VM700>
```

EBMF

EBMF returns or sets the lower and upper alarm limits, in mV, for the Pb (B) multiburst flag in the Component Multiburst application. Legal values are floating point numbers from 0 to 999.9, inclusive.

```
VM700 get EBMF A
300.00  700.00
VM700>
```

EBNL

EBNL returns or sets the lower and upper alarm limits for Pb (B) non-linearity in the Component NonLinearity application. Legal values are floating-point numbers from 0 to 50, inclusive.

```
VM700 get EBNL A
0.00  5.00
VM700>
```

EBNO

EBNO returns or sets the lower and upper alarm limits for the rms Pb (B) noise level, in dB, in the Component Noise application. Legal values are floating point numbers from -100 to 0, inclusive.

```
VM700 get EBNO A
---  -45.00
VM700>
```

EBPP

EBPP returns or sets the lower and upper alarm limits and the arrow setting for the Pb (B) Peak-to-Peak Amplitude, in mV, in the Component LevelMeter application. Legal values are floating point numbers from 500 to 2000, inclusive.

```
VM700 get EBPP A
665.00  735.00  700.00
VM700>
```

EBRD

EBRD returns or sets the lower and upper alarm limits and the arrow setting for the Pb (B) to Pr (R) Delay, in nsec, in the Component Channel Delay application. Legal values are floating point values from -400 to 400, inclusive.

```
VM700 get EBRD A
-10.00  10.00  0.00
VM700>
```

EBWL

EBWL specifies the field and line used for Bowtie measurements. The set command takes two arguments after the channel letter, representing the field number and line number, respectively.

```
VM700A> get EBWL A
1  45
VM700A> set EBWL A 1 50
VM700A> get EBWL A
1  50
VM700A>
```

EBWT

EBWT specifies the lower and upper alarm limits of interchannel delay (in ns) for Bowtie measurements. The set command takes two arguments after the channel letter, representing the lower and upper limit, respectively. Legal values are numbers from -100 to +100, inclusive, or "undef", which indicates no limit set.

```
VM700A> get EBWT A
-10.00  10.00
VM700A> set EBWT A -5 5
VM700A> get EBWT A
-5.00  5.00
VM700A>
```

EBYD

EBYD returns or sets the lower and upper alarm limits and the arrow setting for the Pb (B) to Y (G) Delay, in nsec, in the Component Channel Delay application. Legal values are floating point values from -400 to 400, inclusive.

```
VM700 get EBYD A
-50.00  50.00  0.00
VM700>
```

ECSD

ECSD returns or sets the ColorBar standard used for the Component ColorBar application. Legal values are: GBR 700, GBR 700 Setup, GBR 714, GBR 714 Setup, YPbPr SMPTE/EBU, YPbPr 714 Betacam Setup, YPbPr 714 Betacam, and YPbPr 700 MII Setup.

```
VM700 get ECSD A
YPbPr SMPTE/EBU
VM700>
```

ELCP

ELCP specifies the lower and upper alarm limits of color peak-to-peak amplitude error, in percent. The set command takes two arguments after the channel letter, representing the lower and upper limit, respectively. Legal values are numbers from -10 to +10, inclusive, or "undef", which indicates no limit set.

```
VM700A> get ELCP A
-2.00  2.00
VM700A> set ELCP A -5 5
VM700A> get ELCP A
-5.00  5.00
VM700A>
```

ELPW

ELPW specifies the lower and upper alarm limits of peak-to-white amplitude error, in percent. The set command takes two arguments after the channel letter, representing the lower and upper limit, respectively. Legal values are numbers from -10 to +10, inclusive, or "undef", which indicates no limit set.

```
VM700A> get ELPW A
-2.00  2.00
VM700A> set ELPW A -5 5
VM700A> get ELPW A
-5.00  5.00
VM700A>
```

EMKL

EMKL specifies the field and line location of the marker used in Bowtie measurements. The set command takes two arguments after the channel letter, representing the field number and line number, respectively.

```
VM700A> get EMKL A
1  164
VM700A> get EMKL A 1 140
VM700A> get EMKL A
1  140
VM700A>
```

EPRI

EPRI tells the VM700A whether or not to use special calibration factors for a probe input. Legal values are "yes" and "no". "Yes" tells the VM700A to use the special calibration factors; "no" uses the standard factors.

NOTE

The EPRI setting applies to all measure-mode applications, not just the component applications.

```
VM700 get EPRI A
no
VM700>
```

ERC1 - ERC8

Keywords ERC1 through ERC8 return or set the lower and upper alarm limits for Pr CB colors in the Component ColorBar application. The settings are in milliVolts. Legal values are floating point numbers from -500 to 500, inclusive. The color ordering is as follows: 1, gray; 2, yellow; 3, cyan; 4, green; 5, magenta; 6, red; 7, blue; 8, black.

```
VM700 get ERC1 A
-50.00  50.00
VM700>
```

ERCI

ERCI tells the VM700A whether or not stored reference values are shared by all channels for each standard, or are stored independently from channel to channel. Legal values are "yes" and "no".

ERCI

ERCI tells the VM700A whether or not stored reference values are shared by all channels for each standard, or are stored independently from channel to channel. Legal values are "yes" and "no". "Yes" means reference values are stored independently of each other; "no" means reference values are shared by all channels for each standard.

NOTE

The ERCI setting applies to all measure-mode applications, not just the component applications.

```
VM700 get ERCI A
no
VM700>
```

ERHA

ERHA returns or sets the half-amplitude duration for the Pr(R) K-Factor pulse, in T units. Legal values are integers from 2 to 8, inclusive.

```
VM700 get ERHA A
7
VM700>
```

ERKB

ERKB returns or sets the lower and upper alarm limits for the Pr K-PB factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from -50 to 50, inclusive.

```
VM700 get ERKB A
-5.00  1.00
VM700>
```

ERKF

ERKF returns or sets the lower and upper alarm limits for the Pr K-factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from 0 to 99.9, inclusive.

```
VM700 get ERKF A
0.00  5.00
VM700>
```

ERMB

ERMB returns or sets the number of Pr (R) multiburst packets. Legal values are integers from 3 to 9, inclusive.

```
VM700 get ERMB A
5
VM700>
```

ERMF

ERMF returns or sets the lower and upper alarm limits, in mV, for the Pr (R) multiburst flag in the Component Multiburst application. Legal values are floating point numbers from 0 to 999.9, inclusive.

```
VM700 get ERMF A
300.00  700.00
VM700>
```

ERNL

ERNL returns or sets the lower and upper alarm limits for Pr (R) non-linearity in the Component NonLinearity application. Legal values are floating-point numbers from 0 to 50, inclusive.

```
VM700 get ERNL A
0.00  5.00
VM700>
```

ERNO

ERNO returns or sets the lower and upper alarm limits for the rms Pr (R) noise level, in dB, in the Component Noise application. Legal values are floating point numbers from -100 to 0, inclusive.

```
VM700 get ERNO A
---  -45.00
VM700>
```

ERPP

ERPP returns or sets the lower and upper alarm limits and the arrow setting for the Pr (R) Peak-to-Peak Amplitude, in mV, in the Component LevelMeter application. Legal values are floating point numbers from 500 to 2000, inclusive.

```
VM700 get ERPP A
665.00  735.00  700.00
VM700>
```

ERYD

ERYD returns or sets the lower and upper alarm limits and the arrow setting for the Pr (R) to Y (G) Delay, in nsec, in the Component Channel Delay application. Legal values are floating point values from -400 to 400, inclusive.

```
VM700 get ERYD A
-50.00  50.00  0.00
VM700>
```

ETNM

ETNM returns or sets the T duration, in nsec. Legal values are integers from 50 to 150, inclusive.

```
VM700 get ETNM A
125
VM700>
```

EYC1 - EYC8

Keywords EYC1 through EYC8 return or set the lower and upper alarm limits for Y CB colors in the Component ColorBar application. The settings are in milliVolts. Legal values are floating point numbers from -500 to 500, inclusive. The color ordering is as follows: 1, gray; 2, yellow; 3, cyan; 4, green; 5, magenta; 6, red; 7, blue; 8, black.

```
VM700 get EYC1 A
-50.00  50.00
VM700>
```

EYHA

EYHA returns or sets the half-amplitude duration for the Y(G) K-Factor pulse, in T units. Legal values are integers from 2 to 8, inclusive.

```
VM700 get EYHA A
2
VM700>
```

EYKB

EYKB returns or sets the lower and upper alarm limits for the Y K-PB factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from -50 to 50, inclusive.

```
VM700 get EYKB A
-5.00  1.00
VM700>
```

EYKF

EYKF returns or sets the lower and upper alarm limits for the Y K-factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from 0 to 99.9, inclusive.

```
VM700 get EYKF A
0.00  5.00
VM700>
```

EYM1 - EYM9

Keywords EYM1 through EYM9 return or set the lower and upper alarm limits, in dB, for Y (G) packets 1 through 9 in the Component Multiburst application. Legal values are floating point numbers from -40 to 40, inclusive.

```
VM700 get EYM1 A
-1.00  1.00
VM700>
```

EYMB

EYMB returns or sets the number of Y (G) multiburst packets. Legal values are integers from 3 to 9, inclusive.

```
VM700 get EYMB A
6
VM700>
```

EYMF

EYMF returns or sets the lower and upper alarm limits, in mV, for the Y (G) multiburst flag in the Component Multiburst application. Legal values are floating point numbers from 0 to 999.9, inclusive.

```
VM700 get EYMF A
300.00  700.00
VM700>
```

EYNL

EYNL returns or sets the lower and upper alarm limits for Y (G) non-linearity in the Component NonLinearity application. Legal values are floating-point numbers from 0 to 50, inclusive.

```
VM700 get EYNL A
0.00  5.00
VM700>
```

EYNO

EYNO returns or sets the lower and upper alarm limits for the rms Y (G) noise level, in dB, in the Component Noise application. Legal values are floating point numbers from -100 to 0, inclusive.

```
VM700 get EYNO A
---  -45.00
VM700>
```

EYPA

EYPA returns or sets the lower and upper alarm limits and the arrow setting for the Y (G) Peak Amplitude, in mV, in the Component LevelMeter application. Legal values are floating point numbers from 500 to 2000, inclusive.

```
VM700 get EYPA A
678.60  750.00  714.30
VM700>
```

EYSA

EYSA returns or sets the lower and upper alarm limits and the arrow setting for the Y (G) Sync Amplitude, in mV, in the Component LevelMeter application. Legal values are floating point numbers from 500 to 2000, inclusive.

```
VM700 get EYSA A
271.40  300.00  285.70
VM700>
```

"F" Group: Component Configuration (PAL)

"F" keywords report on or set the values of Component Option Configuration parameters. These keywords can only be used on PAL-standard VM700A's equipped with Option 30 (Component).

Get commands used with the "F" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with "F" group keywords take different arguments, depending on the keyword. The form of each Set command used with "F" group keywords is documented in the pages that follow.

Table B-6
"F" Keywords: Component Configuration (PAL)

Keyword	Description	Keyword	Description
FBC1	Pb CB color #1 (mV)	FLCP	Lightning color pk-pk ampl error (%)
FBC2	Pb CB color #2 (mV)	FLPW	Lightning pk-white ampl error (%)
FBC3	Pb CB color #3 (mV)	FMKL	Marker line
FBC4	Pb CB color #4 (mV)	FPRI	Probe Input
FBC5	Pb CB color #5 (mV)	FRC1	Pr CB color #1 (mV)
FBC6	Pb CB color #6 (mV)	FRC2	Pr CB color #2 (mV)
FBC7	Pb CB color #7 (mV)	FRC3	Pr CB color #3 (mV)
FBC8	Pb CB color #8 (mV)	FRC4	Pr CB color #4 (mV)
FBHA	Pb K-Factor Pulse HAD	FRC5	Pr CB color #5 (mV)
FBKB	Pb K-PB Factor (%)	FRC6	Pr CB color #6 (mV)
FBKF	Pb K Factor (%)	FRC7	Pr CB color #7 (mV)
FBM1	Pb MB Packet #1 (dB)	FRC8	Pr CB color #8 (mV)
FBM2	Pb MB Packet #2 (dB)	FRCI	Stored Reference Channel Independent
FBM3	Pb MB Packet #3 (dB)	FRHA	Pr K-Factor Pulse HAD
FBM4	Pb MB Packet #4 (dB)	FRKB	Pr K-PB Factor (%)
FBM5	Pb MB Packet #5 (dB)	FRKF	Pr K Factor (%)
FBM6	Pb MB Packet #6 (dB)	FRM1	Pr MB Packet #1 (dB)
FBM7	Pb MB Packet #7 (dB)	FRM2	Pr MB Packet #2 (dB)
FBM8	Pb MB Packet #8 (dB)	FRM3	Pr MB Packet #3 (dB)
FBM9	Pb MB Packet #9 (dB)	FRM4	Pr MB Packet #4 (dB)
FBMB	Pb Multiburst Packets	FRM5	Pr MB Packet #5 (dB)
FBMF	Pb Multiburst Flag (mV)	FRM6	Pr MB Packet #6 (dB)
FBNL	Pb Non-Linearity (%)	FRM7	Pr MB Packet #7 (dB)
FBNO	Pb Noise Level (dB rms)	FRM8	Pr MB Packet #8 (dB)
FBPP	Pb P-P Amplitude (mV) (3 values)	FRM9	Pr MB Packet #9 (dB)
FBRD	Pb to Pr Delay (nsec) (3 values)	FRMB	Pr Multiburst Packets
FBWL	Bowtie line	FRMF	Pr Multiburst Flag (mV)
FBWT	Bowtie interchannel delay (ns)	FRNL	Pr Non-Linearity (%)
FBYD	Pb to Y Delay (nsec) (3 values)	FRNO	Pr Noise Level (dB rms)
FCSD	Colourbar standard	FRPP	Pr P-P Amplitude (mV) (3 values)

Table B-6 (cont)

Keyword	Description	Keyword	Description
FRYD	Pr to Y Delay (nsec) (3 values)	FYM2	Y MB Packet #2 (dB)
FTNM	T (nsec)	FYM3	Y MB Packet #3 (dB)
FYC1	Y CB color #1 (mV)	FYM4	Y MB Packet #4 (dB)
FYC2	Y CB color #2 (mV)	FYM5	Y MB Packet #5 (dB)
FYC3	Y CB color #3 (mV)	FYM6	Y MB Packet #6 (dB)
FYC4	Y CB color #4 (mV)	FYM7	Y MB Packet #7 (dB)
FYC5	Y CB color #5 (mV)	FYM8	Y MB Packet #8 (dB)
FYC6	Y CB color #6 (mV)	FYM9	Y MB Packet #9 (dB)
FYC7	Y CB color #7 (mV)	FYMB	Y Multiburst Packets
FYC8	Y CB color #8 (mV)	FYMF	Y Multiburst Flag (mV)
FYHA	Y K-Factor Pulse HAD	FYNL	Y Non-Linearity (%)
FYKB	Y K-PB Factor (%)	FYNO	Y Noise Level (dB rms)
FYKF	Y K Factor (%)	FYPA	Y Peak Amplitude (mV) (3 values)
FYM1	Y MB Packet #1 (dB)	FYSA	Y Sync Amplitude (mV) (3 values)

FBC1 - FBC8

Keywords FBC1 through FBC8 return or set the lower and upper alarm limits for Pb CB colors in the Component ColorBar application. The settings are in milliVolts. Legal values are floating point numbers from -500 to 500, inclusive. The color ordering is as follows: 1, gray; 2, yellow; 3, cyan; 4, green; 5, magenta; 6, red; 7, blue; 8, black.

```
VM700 get FBC1 A
-50.00  50.00
VM700>
```

FBHA

FBHA returns or sets the half-amplitude duration for the Pb(B) K-Factor pulse, in T units. Legal values are integers from 2 to 8, inclusive.

```
VM700 get FBHA A
7
VM700>
```

FBKB

FBKB returns or sets the lower and upper alarm limits for the Pb K-PB factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from -50 to 50, inclusive.

```
VM700 get FBKB A
-5.00  1.00
VM700>
```

FBKF

FBKF returns or sets the lower and upper alarm limits for the Pb K-factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from 0 to 99.9, inclusive.

```
VM700 get FBKF A
0.00  5.00
VM700>
```

FBM1 - FBM9

Keywords FBM1 through FBM9 return or set the lower and upper alarm limits, in dB, for Pb(B) packets 1 through 9 in the Component Multiburst application. Legal values are floating point numbers from -40 to 40, inclusive.

```
VM700 get FBM1 A
-1.00  1.00
VM700>
```

FBMB

FBMB returns or sets the number of Pb (B) multiburst packets. Legal values are integers from 3 to 9, inclusive.

```
VM700 get FBMB A
5
VM700>
```

FBMF

FBMF returns or sets the lower and upper alarm limits, in mV, for the Pb (B) multiburst flag in the Component Multiburst application. Legal values are floating point numbers from 0 to 999.9, inclusive.

```
VM700 get FBMF A
300.00  700.00
VM700>
```

FBNL

FBNL returns or sets the lower and upper alarm limits for Pb (B) non-linearity in the Component NonLinearity application. Legal values are floating-point numbers from 0 to 50, inclusive.

```
VM700 get FBNL A
0.00  5.00
VM700>
```

FBNO

FBNO returns or sets the lower and upper alarm limits for the rms Pb (B) noise level, in dB, in the Component Noise application. Legal values are floating point numbers from -100 to 0, inclusive.

```
VM700 get FBNO A
---  -45.00
VM700>
```

FBPP

FBPP returns or sets the lower and upper alarm limits and the arrow setting for the Pb (B) Peak-to-Peak Amplitude, in mV, in the Component LevelMeter application. Legal values are floating point numbers from 500 to 2000, inclusive.

```
VM700 get FBPP A
665.00  735.00  700.00
VM700>
```

FBRD

FBRD returns or sets the lower and upper alarm limits and the arrow setting for the Pb (B) to Pr (R) Delay, in nsec, in the Component Channel Delay application. Legal values are floating point values from -400 to 400, inclusive.

```
VM700 get FBRD A
-10.00  10.00  0.00
VM700>
```

FBWL

FBWL specifies the line used for Bowtie measurements. The set command takes one argument after the channel letter, representing the line number.

```
VM700A> get FBWL A
45
VM700A> set FBWL A 50
VM700A> get FBWL A
50
VM700A>
```

FBWT

FBWT specifies the lower and upper alarm limits of interchannel delay (in ns) for Bowtie measurements. The set command takes two arguments after the channel letter, representing the lower and upper limit, respectively. Legal values are numbers from -100 to +100, inclusive, or "undef", which indicates no limit set.

```
VM700A> get FBWT A
-10.00  10.00
VM700A> set FBWT A -5 5
VM700A> get FBWT A
-5.00  5.00
VM700A>
```

FBYD

FBYD returns or sets the lower and upper alarm limits and the arrow setting for the Pb (B) to Y (G) Delay, in nsec, in the Component Channel Delay application. Legal values are floating point values from -400 to 400, inclusive.

```
VM700 get FBYD A
-50.00  50.00  0.00
VM700>
```

FCSD

FCSD returns or sets the ColorBar standard used for the Component ColorBar application. Legal values are: GBR 700, GBR 700 Setup, GBR 714, GBR 714 Setup, YPbPr SMPTE/EBU, YPbPr 714 Betacam Setup, YPbPr 714 Betacam, and YPbPr 700 MII Setup.

```
VM700 get FCSD A
YPbPr SMPTE/EBU
VM700>
```

FLCP

FLCP specifies the lower and upper alarm limits of color peak-to-peak amplitude error, in percent. The set command takes two arguments after the channel letter, representing the lower and upper limit, respectively. Legal values are numbers from -10 to +10, inclusive, or "undef", which indicates no limit set.

```
VM700A> get FLCP A
-2.00  2.00
VM700A> set FLCP A -5 5
VM700A> get FLCP A
-5.00  5.00
VM700A>
```

FLPW

FLPW specifies the lower and upper alarm limits of peak-to-white amplitude error, in percent. The set command takes two arguments after the channel letter, representing the lower and upper limit, respectively. Legal values are numbers from -10 to +10, inclusive, or "undef", which indicates no limit set.

```
VM700A> get FLPW A
-2.00  2.00
VM700A> set FLPW A -5 5
VM700A> get FLPW A
-5.00  5.00
VM700A>
```

FMKL

FMKL specifies the line location of the marker used in Bowtie measurements. The set command takes one argument after the channel letter, representing the line number.

```
VM700A> get FMKL A
164
VM700A> get FMKL A 140
VM700A> get FMKL A
140
VM700A>
```

FPRI

FPRI tells the VM700A whether or not to use special calibration factors for a probe input. Legal values are "yes" and "no". "Yes" tells the VM700A to use the special calibration factors; "no" uses the standard factors.

NOTE

The FPRI setting applies to all measure-mode applications, not just the component applications.

```
VM700 get FPRI A
no
VM700>
```

FRC1 - FRC8

Keywords FRC1 through FRC8 return or set the lower and upper alarm limits for Pr CB colors in the Component ColorBar application. The settings are in milliVolts. Legal values are floating point numbers from -500 to 500, inclusive. The color ordering is as follows: 1, gray; 2, yellow; 3, cyan; 4, green; 5, magenta; 6, red; 7, blue; 8, black.

```
VM700 get FRC1 A
-50.00  50.00
VM700>
```

FRCI

FRCI tells the VM700A whether or not stored reference values are shared by all channels for each standard, or are stored independently from channel to channel. Legal values are "yes" and "no". "Yes" means reference values are stored independently of each other; "no" means reference values are shared by all channels for each standard.

NOTE

The FRCI setting applies to all measure-mode applications, not just the component applications.

```
VM700 get FRCI A
no
VM700>
```

FRHA

FRHA returns or sets the half-amplitude duration for the Pr(R) K-Factor pulse, in T units. Legal values are integers from 2 to 8, inclusive.

```
VM700 get FRHA A
7
VM700>
```

FRKB

FRKB returns or sets the lower and upper alarm limits for the Pr K-PB factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from -50 to 50, inclusive.

```
VM700 get FRKB A
-5.00  1.00
VM700>
```

FRKF

FRKF returns or sets the lower and upper alarm limits for the Pr K-factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from 0 to 99.9, inclusive.

```
VM700 get FRKF A
0.00  5.00
VM700>
```

FRM1 - FRM9

Keywords FRM1 through FRM9 return or set the lower and upper alarm limits, in dB, for Pr (R) packets 1 through 9 in the Component Multiburst application. Legal values are floating point numbers from -40 to 40, inclusive.

```
VM700 get FRM1 A
-1.00  1.00
VM700>
```

FRM1 - FRM9

Keywords FRM1 through FRM9 return or set the lower and upper alarm limits, in dB, for Pr (R) packets 1 through 9 in the Component Multiburst application. Legal values are floating point numbers from -40 to 40, inclusive.

```
VM700 get FRM1 A
-1.00  1.00
VM700>
```

FRMB

FRMB returns or sets the number of Pr (R) multiburst packets. Legal values are integers from 3 to 9, inclusive.

```
VM700 get FRMB A
5
VM700>
```

FRMF

FRMF returns or sets the lower and upper alarm limits, in mV, for the Pr (R) multiburst flag in the Component Multiburst application. Legal values are floating point numbers from 0 to 999.9, inclusive.

```
VM700 get FRMF A
300.00  700.00
VM700>
```

FRNL

FRNL returns or sets the lower and upper alarm limits for Pr (R) non-linearity in the Component NonLinearity application. Legal values are floating-point numbers from 0 to 50, inclusive.

```
VM700 get FRNL A
0.00  5.00
VM700>
```

FRNO

FRNO returns or sets the lower and upper alarm limits for the rms Pr (R) noise level, in dB, in the Component Noise application. Legal values are floating point numbers from -100 to 0, inclusive.

```
VM700 get FRNO A
---  -45.00
VM700>
```

FRPP

FRPP returns or sets the lower and upper alarm limits and the arrow setting for the Pr (R) Peak-to-Peak Amplitude, in mV, in the Component LevelMeter application. Legal values are floating point numbers from 500 to 2000, inclusive.

```
VM700 get FRPP A
665.00  735.00  700.00
VM700>
```

FRYD

FRYD returns or sets the lower and upper alarm limits and the arrow setting for the Pr (R) to Y (G) Delay, in nsec, in the Component Channel Delay application. Legal values are floating point values from -400 to 400, inclusive.

```
VM700 get FRYD A
-50.00  50.00  0.00
VM700>
```

FTNM

FTNM returns or sets the T duration, in nsec. Legal values are integers from 50 to 150, inclusive.

```
VM700 get FTNM A
125
VM700>
```

FYC1 - FYC8

Keywords FYC1 through FYC8 return or set the lower and upper alarm limits for Y CB colors in the Component ColorBar application. The settings are in milliVolts. Legal values are floating point numbers from -500 to 500, inclusive. The color ordering is as follows: 1, gray; 2, yellow; 3, cyan; 4, green; 5, magenta; 6, red; 7, blue; 8, black.

```
VM700 get FYC1 A
-50.00  50.00
VM700>
```

FYHA

FYHA returns or sets the half-amplitude duration for the Y(G) K-Factor pulse, in T units. Legal values are integers from 2 to 8, inclusive.

```
VM700 get FYHA A
2
VM700>
```

FYKB

FYKB returns or sets the lower and upper alarm limits for the Y K-PB factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from -50 to 50, inclusive.

```
VM700 get FYKB A
-5.00  1.00
VM700>
```

FYKF

FYKF returns or sets the lower and upper alarm limits for the Y K-factor, as a percentage, in the Component K-Factor application. Legal values are floating point numbers from 0 to 99.9, inclusive.

```
VM700 get FYKF A
0.00  5.00
VM700>
```

FYM1 - FYM9

Keywords FYM1 through FYM9 return or set the lower and upper alarm limits, in dB, for Y (G) packets 1 through 9 in the Component Multiburst application. Legal values are floating point numbers from -40 to 40, inclusive.

```
VM700 get FYM1 A
-1.00  1.00
VM700>
```

FYMB

FYMB returns or sets the number of Y (G) multiburst packets. Legal values are integers from 3 to 9, inclusive.

```
VM700 get FYMB A
6
VM700>
```

FYMF

FYMF returns or sets the lower and upper alarm limits, in mV, for the Y (G) multiburst flag in the Component Multiburst application. Legal values are floating point numbers from 0 to 999.9, inclusive.

```
VM700 get FYMF A
300.00  700.00
VM700>
```

FYNL

FYNL returns or sets the lower and upper alarm limits for Y (G) non-linearity in the Component NonLinearity application. Legal values are floating-point numbers from 0 to 50, inclusive.

```
VM700 get FYNL A
0.00  5.00
VM700>
```

FYNO

FYNO returns or sets the lower and upper alarm limits for the rms Y (G) noise level, in dB, in the Component Noise application. Legal values are floating point numbers from -100 to 0, inclusive.

```
VM700 get FYNO A
---  -45.00
VM700>
```

FYPA

FYPA returns or sets the lower and upper alarm limits and the arrow setting for the Y (G) Peak Amplitude, in mV, in the Component LevelMeter application. Legal values are floating point numbers from 500 to 2000, inclusive.

```
VM700 get FYPA A
678.60  750.00  714.30
VM700>
```

FYSA

FYSA returns or sets the lower and upper alarm limits and the arrow setting for the Y (G) Sync Amplitude, in mV, in the Component LevelMeter application. Legal values are floating point numbers from 500 to 2000, inclusive.

```
VM700 get FYSA A
271.40  300.00  285.70
VM700>
```

"G" Group: Remote Configuration

"G" keywords report on or set the values of parameters that control the way the VM700A communicates with a PC or remote terminal.

Get commands used with the "G" keywords have the form

```
get <keyword>
```

Set commands used with "G" group keywords take different arguments, depending on the keyword. The form of each Set command used with "G" group keywords is documented in the pages that follow.

Table B-7
"G" Keywords: Remote Configuration

Keyword	Description	Range	Meaning or Units
GACC	Clamp Coupling, Source A	0 - 3	0 = DC, 1 = AC, 2 = FAST, 3 = SLOW
GACL	Clamp Level, Source A	-128 - 127	mV
GACP	Clamp Position, Source A		µs
GACW	Clamp Width, Source A	0 - 3	0 = 0.5, 1 = 0.67, 2 = 1.0, 3 = 2.0 µs
GAFA	Frame Algorithm, Source A	0 - 2	0 = normal synchronous frame pulse, 1 = block field mode (for VTRs), 2 = synthesized frame pulse, arbitrary phase
GALM	Lock Mode, Source A	0 - 3	0 = TV line lock (phase locked to video), 1 = phase locked to crystal, 2 = external strobe, 3 = 20.25 MHz strobe
GASS	Sound-In-Sync, Source A	0, 1	0 = Off, 1 = On
GAST	Audio Auto Measurement Start	0, 1	0 = false, 1 = true
GBCC	Clamp Coupling, Source B	0 - 3	0 = DC, 1 = AC, 2 = FAST, 3 = SLOW
GBCL	Clamp Level, Source B	-128 - 127	mV
GBCP	Clamp Position, Source B		µs
GBCW	Clamp Width, Source B	0 - 3	0 = 0.5, 1 = 0.67, 2 = 1.0, 3 = 2.0 µs
GBFA	Frame Algorithm, Source B	0 - 2	see GAFA (above)
GBLM	Lock Mode, Source B	0 - 3	see GALM (above)
GBSS	Sound-In-Sync, Source B	0, 1	0 = Off, 1 = On
GCCC	Clamp Coupling, Source C	0 - 3	0 = DC, 1 = AC, 2 = FAST, 3 = SLOW
GCCL	Clamp Level, Source C	-128 - 127	mV
GCCP	Clamp Position, Source C		µs
GCCW	Clamp Width, Source C	0 - 3	0 = 0.5, 1 = 0.67, 2 = 1.0, 3 = 2.0 µs
GCFA	Frame Algorithm, Source C	0 - 2	see GAFA (above)
GCLM	Lock Mode, Source C	0 - 3	see GALM (above)
GCSS	Sound-In-Sync, Source C	0, 1	0 = Off, 1 = On
GLN5	System Line (NTSC)	1 - 525	current line number.
GLN6	System Line (PAL)	1 - 625	current line number
GOOL	Out-of-limits flag	0, 1	0 = false, 1 = true
GRLY	Alarm Relay Closure	0, 1	0 opens relay, 1 closes relay
GSNC	Sync	0 - 3	0 = A, 1 = B, 2 = C, 3 = External

GSRC	Source	0 - 7	0 = Source A, 1 = B, 2 = C, 3 = A - B, 4 = B - C, 5 = A -C, 6 = -A, 7= -B.
GSSR	Locked to Source	0, 1	0 = Off; 1 = On
GSYI	Sync Inverted	0, 1	0 = normal sync; 1 = inverted sync

GACC, GBCC, GCCC

GACC, GBCC, and GCCC specify the clamp coupling mode for sources A, B, and C, respectively. The set command takes one argument , an integer from 0 to 3, inclusive.

A setting of 0 indicates DC coupling (no clamping) is in effect.

A setting of 1 indicates AC coupling is in effect.

A setting of 2 indicates that fast clamp coupling is in effect. This removes DC offset, hum, and bounce from the signal.

A setting of 3 indicates that slow clamp coupling is in effect. This still allows hum effects to be visible, but is useful in coping with large DC offsets on an input signal.

```
VM700A> get GACC
```

GACL, GBCL, GCCL

GACL, GBCL, and GCCL specify the clamping level for sources A, B, and C, respectively. The set command takes one argument , representing the clamping level in milliVolts.

```
VM700A> get GACL
```

GACP, GBCP, GCCP

GACP, GBCP, and GCCP specify the clamping position for sources A, B, and C, respectively.. The set command takes one argument , representing the clamping position in µsec.

```
VM700A> get GACP
```

GACW, GBCW, GCCW

GACW, GBCW, and GCCW specify the clamp width for sources A, B, and C, respectively. The set command takes one argument , an integer from 0 to 3, inclusive. A value of 0 sets the clamp width to 0.5 µsec. A value of 1 sets the clamp width to 0.67 µsec. A value of 2 sets the clamp width to 1.0 µsec. A value of 3 sets the clamp width to 2.0 µsec.

```
VM700A> get GACW
```

GAFA, GBFA, GCFA

GAFA, GBFA, and GCFA specify the frame algorithm used for sources A, B, and C, respectively. The set command takes one argument , an integer from 0 to 2, inclusive. A value of 0 sets the frame algorithm to a normal synchronous frame pulse. A value of 1 sets the frame algorithm to block field mode. A value of 2 sets the frame algorithm to an arbitrary-phase, synthesized frame pulse.

```
VM700A> get GAFA
```

GALM, GBLM, GCLM

GALM, GBLM, and GCLM specify the lock mode for sources A, B, and C, respectively. The set command takes one argument , an integer from 0 to 3, inclusive. A value of 0 sets the lock mode to TV line lock (phase locked to video). A value of 1 sets the lock mode to "phase-locked to crystal." A value of 2 sets the lock mode to external strobe. A value of 3 sets the lock mode to a 20.25-MHz strobe.

```
VM700A> get GALM
```

GASS, GBSS, GCSS

GASS, GBSS, and GCSS turns Sound-in-Sync off or on for sources A, B, and C, respectively. The set command takes one argument , 0 or 1. A value of 0 turns Sound-In-Sync off. A value of 1 turns Sound-In-Sync on.

```
VM700A> get GASS
```

GAST

The GAST flag is similar to the GOOL flag, but it will be set TRUE if an Audio Auto Measurement starts (in other words, when it detects the first FSK preamble of the audio test sequence). So, a function can be played back at the same time as an Audio Measurement starts. The state of the GAST flag may be found by using the get GAST command, and it may be set using the set GAST command. The arguments are 0 and 1.

The flag is useful for triggering the audio and the video measurements at the same time from a remote site, and when the video has to start a specific timing such as VTR playback testing.

The GAST flag must be reset by the user to the state needed for the conditional testing.

```
set GAST 0
?GAST return
delay 10
loop
```

GLN5

GLN5 specifies the system line number for NTSC. The set command takes one argument , an integer from 1 to 525, representing the line number. Lines in field 1 are numbered from 1 to 263. Lines in field 2 are numbered from 264 to 525.

```
VM700A> get GLN5
```

GLN6

GLN6 specifies the system line number for PAL. The set command takes one argument , an integer from 1 to 625, representing the line number.

```
VM700A> get GLN6
```

GOOL

GOOL specifies the value of the global out-of-limits flag. The set command takes one argument , 0 (false) or 1 (true). This parameter is set to 1 whenever a measurement application finds a value outside of its lower-to-upper limit range. The status of the global out-of-limits flag can be tested with the **?GOOL** command. The global out-of-limits flag must be reset to 0 explicitly within a function; it does not return to 0 when tested. See the description of the **?GOOL** command in Appendix A for an example showing the use of the GOOL keyword within a function.

GRLY

The ALARM connector on the rear panel of the VM700A (located between the two serial ports connectors) is not active. The internal relay that controls the contact closure is accessible via a set command.

set GRLY 0 opens the relay and set GRLY 1 closes the relay.

The set may be used in a Function Key in conjunction with the GOOL (global-out-of-limits) flag in a Function Key to turn on an alarm indicator.

A second application for a program to monitor some condition for the out-of-limits to occur and use set GRLY 1 to close the relay for a period of time to sound an alarm, then open it again with set GRLY 0 to shut the alarm off.

```
VM700A> get GSNC
```

GSNC

GSNC specifies the sync source. The set command takes one argument, an integer from 0 to 3, inclusive. Values of 0, 1, 2, and 3 set the sync source to A, B, C, and external, respectively.

```
VM700A> get GSNC
```

GSRC

GSRC specifies the signal source. The set command takes one argument , an integer from 0 to 7, inclusive. Values of 0, 1, and 2 set the signal source to A, B, or C, respectively. A value of 3 sets the signal source to A-B. A value of 4 sets the signal source to B-C. A value of 5 sets the signal source to A-C. A value of 6 sets the signal source to -A. A value of 7 sets the signal source to -B.

```
VM700A> get GSRC
```

GSSR

GSSR turns Locked to Source off or on. The set command takes one argument, 0 or 1. A value of 0 turns Locked to Source off. A value of 1 turns Locked to Source on.

```
VM700A> get GSSR
```

GSYI

GSYI controls sync inversion. The set command takes one argument, 0 or 1. A value of 0 sets sync to normal. A value of 1 inverts the sync.

```
VM700A> get GSYI
```

"H" Group: Echo/Rounding Configuration (NTSC)

"H" keywords report on or set the values of Echo/Rounding Option Configuration parameters. These keywords can only be used on NTSC-standard VM700A's equipped with Option 1G (Echo/Rounding).

Get commands used with the "H" keywords have the form

```
get <keyword> <channel-letter>
```

The form of each Set command used with "H" group keywords is documented below.

Table B-8
"H" Keywords: Echo/Rounding Configuration (NTSC)

Keyword	Description	Keyword	Description
HROB	Rounding of Black (%)	HROW	Rouding of White(%)

HROB

HROB specifies the lower and upper limits, in percent, for the Rounding of the Black measurement. The set command takes two arguments after the channel letter, representing the values of the lower and upper limits, respectively. Legal values are numbers between -50 and 50, inclusive, or "undef", which indicates no limit set.

```
VM700A> get HROB A
-1.00  1.00
VM700A> set HROB A undef 1
VM700A> get HROB A
---  1.00
VM700A>
```

HROW

HROW specifies the lower and upper limits, in percent, for the Rounding of the White measurement. The set command takes two arguments after the channel letter, representing the values of the lower and upper limits, respectively. Legal values are numbers between -50 and 50, inclusive, or "undef", which indicates no limit set.

```
VM700A> get HROW A
-1.00  1.00
VM700A> set HROW A undef 1
VM700A> get HROW A
---  1.00
VM700A>
```

"I" Group: Echo/Rounding Configuration (PAL)

"I" keywords report on or set the values of Echo/Rounding Option Configuration parameters. These keywords can only be used on PAL-standard VM700A's equipped with Option 1G (Echo/Rounding).

Get commands used with the "I" keywords have the form

```
get <keyword> <channel-letter>
```

The form of each Set command used with "I" group keywords is documented in the pages that follow.

Table B-9
"I" Keywords: Echo/Rounding Configuration (PAL)

Keyword	Description	Keyword	Description
IROB	Rounding of Black (%)	IROW	Rouding of White(%)

IROB

IROB specifies the lower and upper limits, in percent, for the Rounding of the Black measurement. The set command takes two arguments after the channel letter, representing the values of the lower and upper limits, respectively. Legal values are numbers between -50 and 50, inclusive, or "undef", which indicates no limit set.

```
VM700A> get IROB A
-1.00  1.00
VM700A> set IROB A undef 1
VM700A> get IROB A
---  1.00
VM700A>
```

IROW

IROW specifies the lower and upper limits, in percent, for the Rounding of the White measurement. The set command takes two arguments after the channel letter, representing the values of the lower and upper limits, respectively. Legal values are numbers between -50 and 50, inclusive, or "undef", which indicates no limit set.

```
VM700A> get IROW A
-1.00  1.00
VM700A> set IROW A undef 1
VM700A> get IROW A
---  1.00
VM700A>
```

"J" Group: Teletext Configuration (NTSC)

"J" keywords report on or set the values of Teletext Option Configuration parameters. These keywords can only be used on NTSC-standard VM700A's equipped with Option 20 (Teletext).

Get commands used with the "J" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with "J" group keywords take different arguments, depending on the keyword. The form of each Set command used with "J" group keywords is documented in the pages that follow.

Table B-10
"J" Keywords: Teletext Configuration (NTSC)

Keyword	Description	Keyword	Description
J0LV	'0' Level (mV)	JPPM	P-P Amplitude (mV)
J1LV	'1' Level (mV)	JPPP	P-P Amplitude (%)
JDES	Data End to Sync (µsec)	JRIA	Run In Amplitude (mV)
JDLW	Data Line Width (µsec)	JRIB	Run In Bits (bits)
JEHM	Eye Height (mV)	JRIS	Run In Start (µsec)
JEHP	Eye Height (%)	JTTL	Teletext field and line
JEWP	Eye Width (%)	JTUN	Timing Unit

J0LV

J0LV specifies the lower and upper limits of the '0' level in milliVolts for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbersfrom -50 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get J0LV A
-18.00  18.00
VM700A> set J0LV A -20 20
VM700A> get J0LV A
-20.00  20.00
VM700A>
```

J1LV

J1LV specifies the lower and upper limits of the '1' level in milliVolts for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get J1LV A
482.00  518.00
VM700A> set J1LV A 475 525
VM700A> get J0LV A
475.00  525.00
VM700A>
```

JDES

JDES specifies the lower and upper limits for data-end-to-sync in microseconds for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 99.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JDES A
1.00  3.00
VM700A> set JDES A 0.5 5
VM700A> get JDES A
0.5  5.00
```

JDLW

JDLW specifies the lower and upper limits for data line width in microseconds for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 99.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JDLW A
52.00  59.00
VM700A> set JDLW A 50 60
VM700A> get JDLW A
50.00  60.00
VM700A>
```

JEHM

JEHM specifies the lower and upper limits for eye height in milliVolts for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JEHM A
350.00  500.00
VM700A> set JEHM A 300 600
VM700A> get JEHM A
300.00  600.00
VM700A>
```

JEHP

JEHP specifies the lower and upper limits for eye height in percent for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 100, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JEHP A
70.00  100.00
VM700A> set JEHP A 75 undef
VM700A> get JEHP A
75.00  ---
VM700A>
```

JEWP

JEWP specifies the lower and upper limits for eye width in percent for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 100, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JEWP A
70.00  100.00
VM700A> set JEWP A 75 undef
VM700A> get JEWP A
75.00  ---
VM700A>
```

JPPM

JPPM specifies the lower and upper limits for peak-to-peak amplitude in milliVolts for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JPPM A
500.00  650.00
VM700A> set JPPM A 550 undef
VM700A> get JPPM A
550.00  ---
VM700A>
```

JPPP

JPPP specifies the lower and upper limits for peak-to-peak amplitude in percent for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 100 to 200, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JPPP A
100.00  130.00
VM700A> set JPPP A 105 150
VM700A> get JPPP A
105.00  150.00
VM700A>
```

JRIA

JRIA specifies the lower and upper limits for run-in amplitude in milliVolts for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JRIA A
482.00  518.00
VM700A> set JRIA A 480 520
VM700A> get JRIA A
480.00  520.00
VM700A>
```

JRIB

JRIB specifies the lower and upper limits of the number of run-in bits for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 6 to 25, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JRIB A
14.00  18.00
VM700A> set JRIB A 6 25
VM700A> get JRIB A
6.00  25.00
VM700A>
```

JRIS

JRIS specifies the lower and upper limits for run-in start time in microseconds for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 7 to 15, inclusive, and "undef", which indicates no limit set.

```
VM700A> get JRIS A
9.00  11.00
VM700A> get JRIS A 8.5 12
VM700A> get JRIS A
8.50  12.00
VM700A>
```

JTTL

JTTL specifies the field and line location for the Teletext measurement on NTSC-standard VM700A's. The set command takes two arguments after the channel letter, indicating the field and line number, respectively.

```
VM700A> get JTTL A
1  15
VM700A> set JTTL A 1 10
VM700A> get JTTL A
1  10
VM700A>
```

JTUN

JTUN specifies the display timing unit for the Teletext measurement on NTSC-standard VM700A's. The set command takes one argument after the channel letter, indicating the current timing unit. Legal values are usec (for microseconds) and Tc (for "clock periods", equal to 0.1746 μsec).

```
VM700A> get JTUN A
usec
VM700A> set JTUN A Tc
VM700A> get JTUN A
Tc
VM700A>
```

"K" Group: Teletext Configuration (PAL)

"K" keywords report on or set the values of Teletext Option Configuration parameters. These keywords can only be used on PAL-standard VM700A's equipped with Option 20 (Teletext).

Get commands used with the "K" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with "K" group keywords take different arguments, depending on the keyword. The form of each Set command used with "K" group keywords is documented in the pages that follow.

Table B-11
"K" Keywords: Teletext Configuration (PAL)

Keyword	Description	Keyword	Description
K0LV	'0' Level (mV)	KRIS	Run In Start (µsec)
K1LV	'1' Level (mV)	KS0L	SIS '0' Level (mV)
KDES	Data End to Sync (µsec)	KS1L	SIS '1' Level (mV)
KDLW	Data Line Width (µsec)	KS2L	SIS '2' Level (mV)
KDTM	Data Timing (µsec)	KS3L	SIS '3' Level (mV)
KEHM	Eye Height (mV)	KSHM	SIS Eye Height (mV)
KEHP	Eye Height (%)	KSHP	SIS Eye Height (%)
KEWP	Eye Width (%)	KSPM	SIS P-P Amplitude (mV)
KPPM	P-P Amplitude (mV)	KSPP	SIS P-P Amplitude (%)
KPPP	P-P Amplitude (%)	KSWP	SIS Eye Width (%)
KRIA	Run In Amplitude (mV)	KTTL	Teletext line
KRIB	Run In Bits (bits)		

K0LV

K0LV specifies the lower and upper limits of the '0' level in milliVolts for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbersfrom -50 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get K0LV A
-18.00  18.00
VM700A> set K0LV A -20 20
VM700A> get K0LV A
-20.00  20.00
VM700A>
```

K1LV

K1LV specifies the lower and upper limits of the '1' level in milliVolts for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get K1LV A
482.00  518.00
VM700A> set K1LV A 475 525
VM700A> get K1LV A
475.00  525.00
VM700A>
```

KDES

KDES specifies the lower and upper limits for data-end-to-sync in microseconds for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 99.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KDES A
1.00  3.00
VM700A> set KDES A 0.5 5
VM700A> get KDES A
0.5  5.00
```

KDLW

KDLW specifies the lower and upper limits for data line width in microseconds for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 99.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KDLW A
52.00  59.00
VM700A> set KDLW A 50 60
VM700A> get KDLW A
50.00  60.00
VM700A>
```

KDTM

KDTM specifies the lower and upper limits for data timing in microseconds for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 10 to 15, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KDTM A
11.00  13.00
VM700A> set KDTM A 10 15
VM700A> get KDTM A
10.00  15.00
VM700A>
```

KEHM

KEHM specifies the lower and upper limits for eye height in milliVolts for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KEHM A
350.00  500.00
VM700A> set KEHM A 300 600
VM700A> get KEHM A
300.00  600.00
VM700A>
```

KEHP

KEHP specifies the lower and upper limits for eye height in percent for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 100, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KEHP A
70.00  100.00
VM700A> set KEHP A 75 undef
VM700A> get KEHP A
75.00  ---
VM700A>
```

KEWP

KEWP specifies the lower and upper limits for eye width in percent for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 100, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KEWP A
70.00  100.00
VM700A> set KEWP A 75 undef
VM700A> get KEWP A
75.00  ---
VM700A>
```

KPPM

KPPM specifies the lower and upper limits for peak-to-peak amplitude in milliVolts for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KPPM A
500.00  650.00
VM700A> set KPPM A 550 undef
VM700A> get KPPM A
550.00  ---
VM700A>
```

KPPP

KPPP specifies the lower and upper limits for peak-to-peak amplitude in percent for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 100 to 200, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KPPP A
100.00  130.00
VM700A> set KPPP A 105 150
VM700A> get KPPP A
105.00  150.00
VM700A>
```

KRIA

KRIA specifies the lower and upper limits for run-in amplitude in milliVolts for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KRIA A
482.00  518.00
VM700A> set KRIA A 480 520
VM700A> get KRIA A
480.00  520.00
VM700A>
```

KRIB

KRIB specifies the lower and upper limits of the number of run-in bits for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 6 to 25, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KRIB A
14.00  18.00
VM700A> set KRIB A 6 25
VM700A> get KRIB A
6.00  25.00
VM700A>
```

KRIS

KRIS specifies the lower and upper limits for run-in start time in microseconds for the Teletext measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 7 to 15, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KRIS A
9.00  11.00
VM700A> get KRIS A 8.5 12
VM700A> get KRIS A
8.50  12.00
VM700A>
```

KS0L

KS0L specifies the lower and upper limits of the '0' level in milliVolts for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from -999.9 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KS0L A
-330.00  -270.00
VM700A> set KS0L A -350 -250
VM700A> get KS0L A
-350.00  -250.00
VM700A>
```

KS1L

KS1L specifies the lower and upper limits of the '1' level in milliVolts for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from -999.9 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KS1L A
-97.00  -37.00
VM700A> set KS1L A -100 -30
VM700A> get KS1L A
-100.00  -30.00
VM700A>
```

KS2L

KS2L specifies the lower and upper limits of the '2' level in milliVolts for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from -999.9 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KS2L A
137.00  197.00
VM700A> set KS2L A 130 200
VM700A> get KS2L A
130.00  200.00
VM700A>
```

KS3L

KS3L specifies the lower and upper limits of the '3' level in milliVolts for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from -999.9 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KS3L A
370.00  430.00
VM700A> set KS3L A 350 450
VM700A> get KS3L A
350.00  450.00
VM700A>
```

KSHM

KSHM specifies the lower and upper limits of eye height in milliVolts for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KSHM A
163.00  233.00
VM700A> set KSHM A 160 240
VM700A> get KSHM A
160.00  240.00
VM700A>
```

KSHP

KSHP specifies the lower and upper limits of eye height in percent for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 100, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KSHP A
70.00  100.00
VM700A> set KSHP A 75 undef
VM700A> get KSHP A
75.00  undef
VM700A>
```

KSPM

KSPM specifies the lower and upper limits of peak-to-peak amplitude in milliVolts for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 999.9, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KSPM A
700.00  910.00
VM700A> set KSPM A 650 950
VM700A> get KSPM A
650.00  950.00
VM700A>
```

KSPP

KSPP specifies the lower and upper limits of peak-to-peak amplitude in percent for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 100 to 200, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KSPP A
100.00  130.00
VM700A> set KSPP A 110 undef
VM700A> get KSPP A
110.00  undef
VM700A>
```

KSWP

KSWP specifies the lower and upper limits of eye width in percent for the SoundInSync measurement on PAL-standard VM700A's. The set command takes two arguments after the channel letter, indicating the lower and upper limits, respectively. Legal values are numbers from 0 to 100, inclusive, and "undef", which indicates no limit set.

```
VM700A> get KSWP A
70.00  100.00
VM700A> set KSWP A 75 undef
VM700A> get KSWP A
75.00  undef
VM700A>
```

KTTL

KTTL specifies the line location for the Teletext measurement on PAL-standard VM700A's. The set command takes one argument after the channel letter, indicating the line number.

```
VM700A> get KTTL A
17
VM700A> set KTTL A 16
VM700A> get KTTL A
16
VM700A>
```

"L" Group: Measurement Locations (PAL)

"L" keywords report on or set the measurement locations for Measure Mode measurements on PAL-standard VM700A's.

Get commands used with the "L" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with the "L" keywords have the form

```
set <keyword> <channel-letter> <argument>
```

Table B-12 lists the legal and default values for the Set command used with each "L" group keyword.

Table B-12
"L" Keywords: Measurement Locations (PAL)

Keyword	Description	Units	Legal Values	Default
L2TC	2T Sine-Squared Pulse (B1) center location	µsec	number, 0 to 64 (inclusive)	26.0
L2TL	2T Sine-Squared Pulse (B1) line	line	integer, 1 to 625 (inclusive)	17
LBAL	Luminance bar line	line	integer, 1 to 625 (inclusive)	17
LBAR	Black Level Reference (b1)	µsec	number, 0 to 64 (inclusive)	36.0
LBAS	Luminance bar start location	µsec	number, 0 to 64 (inclusive)	12.0
LBAW	Luminance bar width (B2)	µsec	number, 0 to 64 (inclusive)	10.0
LCBL	Color Bar Line	line	integer, 1 to 625 (inclusive)	100
LLBR	Luminance Bar Reference (b2)	µsec	number, 0 to 64 (inclusive)	17.0
LLSL	5-Riser Luminance Staircase (D1) line	line	integer, 1 to 625 (inclusive)	17
LLSS	5-Riser Luminance Staircase (D1) start location	µsec	number, 0 to 64 (inclusive)	40.0
LMBL	Multiburst line	line	integer, 1 to 625 (inclusive)	18
LMC1	3-Level Mod. Pedestal (G2) packet 1 center	µsec	number, 0 to 64 (inclusive)	16.0
LMC2	3-Level Mod. Pedestal (G2) packet 2 center	µsec	number, 0 to 64 (inclusive)	20.0
LMC3	3-Level Mod. Pedestal (G2) packet 3 center	µsec	number, 0 to 64 (inclusive)	25.0
LMCL	3-Level Mod. Pedestal (G2) line	line	integer, 1 to 625 (inclusive)	331
LMFS	Multiburst flag start	µsec	number, 0 to 64 (inclusive)	12.0
LMFW	Multiburst flag width	µsec	number, 0 to 64 (inclusive)	8.0
LMMS	Measure Mode Sampling	---	asynchronous, synchronous	asynchronous

Table B-12 (cont)

Keyword	Description	Units	Legal Values	Default
LMP1	Multiburst packet #1 center location	µsec	number, 0 to 64 (inclusive)	14.5
LMP2	Multiburst packet #2 center location	µsec	number, 0 to 64 (inclusive)	20.5
LMP3	Multiburst packet #3 center location	µsec	number, 0 to 64 (inclusive)	26.5
LMP4	Multiburst packet #4 center location	µsec	number, 0 to 64 (inclusive)	32.5
LMP5	Multiburst packet #5 center location	µsec	number, 0 to 64 (inclusive)	38.5
LMP6	Multiburst packet #6 center location	µsec	number, 0 to 64 (inclusive)	44.5
LMPR	3-Level Mod. Pedestal (G2) reference	µsec	number, 0 to 64 (inclusive)	30.0
LMRE	Modulated Bar (G1) chroma end	µsec	number, 0 to 64 (inclusive)	28.0
LMRL	Modulated Bar (G1) line	line	integer, 1 to 625 (inclusive)	331
LMRR	Modulated Bar Lum-Reference (b6)	µsec	number, 0 to 64 (inclusive)	30.0
LMRS	Modulated Bar (G1) chroma start	µsec	number, 0 to 64 (inclusive)	14.0
LMSL	5-Riser Modulated Staircase (D2) line	line	integer, 1 to 625 (inclusive)	330
LMSS	5-Riser Modulated Staircase (D2) start location	µsec	number, 0 to 64 (inclusive)	40.0
LMUC	Modulated Pulse (F) center location	µsec	number, 0 to 64 (inclusive)	32.0
LMUH	Modulated Pulse (F) HAD	multiples of 100ns	20T, 10T	20T
LMUL	Modulated Pulse (F) line	line	integer, 1 to 625 (inclusive)	17
LQLL	Quiet line	line	integer, 1 to 625 (inclusive)	22
LSCA	5-Riser Chroma Amplitude	% of bar amplitude	40%, 20%	40%

Table B-12 (cont)

Keyword	Description	Units	Legal Values	Default
LSID	Source ID line	line	integer, 1 to 625 (inclusive)	16
LSIL	Source ID start	µsec	number, 0 to 64 (inclusive)	26.0
LSIS	SIS present	---	yes, no	no
LSXX	Sin X/X Line	line	integer, 1 to 625 (inclusive)	100
LTBL	T Bar Start (SD) line	line	integer, 1 to 625 (inclusive)	17
LTBS	T Bar Start (SD) location	µsec	number, 0 to 64 (inclusive)	12.0
LTBW	T Bar Width (SD)	µsec	number, 0 to 64 (inclusive)	10.0
LZCC	Zero Carrier Pulse center location	µsec	number, 0 to 64 (inclusive)	35.0
LZCL	Zero Carrier Pulse line	µsec	integer, 1 to 625 (inclusive)	13

"M" Group: Measurement Locations (NTSC)

"M" keywords report on or set the measurement locations for Measure Mode measurements on NTSC-standard VM700A's.

All Get and Set commands used with the "M" keywords are channel-specific.

Get commands used with the "M" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with the "M" keywords have the form

```
set <keyword> <channel-letter> <argument> [<argument>]
```

set commands used with "M" group keywords have either one or two arguments, depending on the keyword. Table B-13 lists the legal and default values for the Set command used with each "L" group keyword.

Table B-13
"M" Keywords: Measurement Locations (NTSC)

Keyword	Description	Units	Legal Values	Default
MAMU	Amplitude units	IRE or volts	IRE, volts	IRE
MBNC	Bounce	Field and Line	1,2 / 10 to 262	1 30
MCBF	Color Bars	Field	1, 2	2
MCBL	Color Bars	Line	10 to 262	17
MCBT	ColorBar	Field and Line	1,2 / 10 to 262	2 17
MCFT	Chroma Freq Resp	Field and Line	1,2 / 10 to 262	1 15
MCIF	VIRS	Field	1, 2	1
MCIL	VIRS	Line	10 to 262	19
MCLG	ChromLum GainDelay	Field and Line	1,2 / 10 to 262	1 18
MCNT	Chrominance NonLinearity	Field and Line	1,2 / 10 to 262	1 17
MCVF	Composite VITS	Field	1, 2	1
MCVL	Composite VITS	Line	10 to 262	18
MDGT	DGDP	Field and Line	1,2 / 10 to 262	1 18
MEGD	GroupDelay SinX_X	Field and Line	1,2 / 10 to 262	2 18
MFGR	Fix 0 IRE to	- - -	Back Porch, 0 volts	Back Porch
MHTL	H Timing	Field and Line	1,2 / 10 to 262	1 100
MIRR	IRE bar reference	- - -	yes, no	no
MKFL	K Factor	Field and Line	1,2 / 10 to 262	1 18
MLNT	Luminance NonLinearity	Field and Line	1,2 / 10 to 262	1 18
MMBF	Multiburst	Field	1, 2	1
MMBL	Multiburst	Line	10 to 262	17
MMBT	MultiBurst	Field and Line	1,2 / 10 to 262	1 18
MNLF	Noise Line (Quiet)	Field	1, 2	1
MNLL	Noise Line (Quiet)	Line	10 to 262	12
MNSL	Noise Spectrum	Field and Line	1,2 / 10 to 262	1 12
MNVF	NTC-7 Combination	Field	1, 2	1
MNVL	NTC-7 Combination	Line	10 to 262	17
MPBT	Bar LineTime	Field and Line	1,2 / 10 to 262	1 18
MSAM	Sampling	- - -	synchronous, asynchronous	asynchronous
MTBL	T Bar (SD)	Field and Line	1,2 / 10 to 262	1 18

Table B-13 (cont)

Keyword	Description	Units	Legal Values	Default
MZCA	Zero Carrier Pulse Ref	- - -	yes, no	no
MZCC	Zero Carrier Pulse	Center Location (µsec)	0 to 64	25.5
MZCF	Zero Carrier Pulse	Field	1, 2	1
MZCL	Zero Carrier Pulse	Line	10 to 262	16

“N” Group: Video Wander Configuration (NTSC)

“N” keywords report on or set the values of the Drift Rate and Frequency Offset application configuration parameters on an NTSC-standard VM700T.

All `Get` and `Set` commands used with the “N” keywords are channel-specific.

`Get` commands used with the “N” keywords have the following form:

```
get <keyword> <channel-letter>
```

`Set` commands used with the “N” keywords have the following form:

```
set <keyword> <channel-letter> <argument> [<argument>]
```

Set commands used with “N” group keywords have either one or two arguments, depending on the keyword. Table 1–1 lists the legal and default values for the `Set` command used with each “N” group keyword.

Table 1–1: “N” Keywords: Video Wander Limits (NTSC)

Keyword	Description	Units	Legal Values	Default
NDRT	Drift-rate limits	ppb/s	–999.0 to 999.0	–28.0 to 28.0
NFRO	Frequency-offset limits	ppb	–99990.0 to 99990.0	–2800.0 to 2800.0

“O” Group: Video Wander Configuration (PAL)

“O” keywords report on or set the values of the Drift Rate and Frequency Offset application configuration parameters on an PAL-standard VM700T.

All Get and Set commands used with the “O” keywords are channel-specific.

Get commands used with the “O” keywords have the following form:

```
get <keyword> <channel-letter>
```

Set commands used with the “O” keywords have the following form:

```
set <keyword> <channel-letter> <argument> [<argument>]
```

Set commands used with “O” group keywords have either one or two arguments, depending on the keyword. Table 1–2 lists the legal and default values for the Set command used with each “O” group keyword.

Table 1–2: “O” Keywords: Video Wander Limits (PAL)

Keyword	Description	Units	Legal Values	Default
ODRT	Drift-rate limits	ppb/s	–999.0 to 999.0	–22.6 to 22.6
OFRO	Frequency-offset limits	ppb	–9999.0 to 9999.0	–226.0 to 226.0

"P" Group: Auto Mode Limits (PAL)

"P" keywords report on or set the limits for Auto Mode measurements on PAL-standard VM700A's.

Get commands used with the "P" keywords have the form

```
get <keyword> <channel letter>
```

Set commands used with the "P" keywords have the form

```
set <keyword> <channel letter> <arg1> <arg2> <arg3> <arg4>
```

where <arg1> is the lower caution limit, <arg2> is the upper caution limit, <arg3> is the lower alarm limit, and <arg4> is the upper alarm limit.

Table B-14 lists the legal ranges and default values for the Set command used with each "P" group keyword.

Table B-14
"P" Keywords: Auto Mode Limits (PAL)

Keyword	Description	Units	Legal Range	Default
P2CW	S/N.2 Chr-wgtd	dB	0 to 999.9	52.0, - - -, 42.0, - - -
P2LW	S/N.2 Lum-wgtd (567)	dB	0 to 999.9	58.0, - - -, 48.0, - - -
P2U9	S/N.2 Un-wgtd (569)	dB	0 to 999.9	58.0, - - -, 48.0, - - -
P2UN	S/N.2 Un-wgtd (567)	dB	0 to 999.9	60.0, - - -, 50.0, - - -
P2W9	S/N.2 Lum-wgtd (569)	dB	0 to 999.9	58.0, - - -, 48.0, - - -
PAPL	Ave. Picture Level	%	0 to 150	- - -
PBAA	Luminance Bar Amp	mv	0 to 999.9	650,750,600,800
PBAC	Luminance Bar Amp	% Carr	0 to 999.9	60,68,55,73
PBAE	Burst Ampl Error	%	-50 to 50	-3, 3 -10, 10
PBAM	Bar Ampl Error	%	-50 to 50	-5, 5 -10, 10
PBAT	Residual Carrier	% Carr	0 to 100	10, 12.5, 7.5, 15
PBAV	Burst Amplitude	mV	0 to 999.9	285, 315, 270, 330
PBDF	Peak Differential Gain	%	-50 to 50	- - -, 10, - - -, 20
PBLD	Baseline Distortion	% Bar	-50 to 50	-1, 1, -2, 2
PBLL	Blanking Level	% Carr	0 to 100	72, 76, 69, 79
PBPS	Broad Pulse Sep	μsec	0 to 20	4.5, 4.9, 4.4, 5.0
PBQD	Burst Quadrature Error	deg	-90 to 90	-2, 2, -5, 5
PBRT	Bar Rise Time	nsec	0 to 999.9	180, 220, 160, 240

Table B-14 (cont)

Keyword	Description	Units	Legal Range	Default
PBTE	Bar Tilt (Rec 569)	% Bar	-40 to 40	-5, 5, -10, 10
PCBC	Burst Duration	Cycles	0 to 100	9, 11, 7.5, 12.5
PCBD	Burst Duration	µsec	0 to 20	2, 2.5, 1.7, 2.8
PCEP	number of consecutive errors before printing	- - -	integer, 1 to 3	2
PCGP	Chr/Lum Gn (Mod Bar)	% Bar	-90 to 90	-10, 10, -20, 20
PCHW	S/N Chr-wgtd	dB	0 to 999.9	52, - - -, 42, - - -
PCLD	Chr/Lum Delay Ineq	%	-500 to 500	-40, 40, -50, 50
PCLI	Chr/Lum Intermod	% Bar	-50 to 50	-5, 5, -10, 10
PCRA	Chrom Ref Ampl Err	%	-50 to 50	-3, 3, -10, 10
PDGD	Pk-Pk Diff Gain	%	0 to 100	- - -, 10, - - -, 20
PDPD	Pk-Pk Diff Phase	Deg	0 to 100	- - -, 10, - - -, 20
PEPD	Equalizing Pulse	µsec	0 to 20	2.34, 2.36, 2.3, 2.4
PFPD	Front Porch	µsec	0 to 20	1.2, 1.8, 0.5, 3.0
PFTD	Field Time Distortion	%	-50 to 50	-1.0, 1.0, -2.0, 2.0
PGNP	Chr/Lum Gn (Mod Pls)	% Bar	-90 to 90	-10, 10, -20, 20
PICP	ICPM (Absolute)	deg	-50 to 50	-10, 10, -20, 20
PICR	ICPM (Rel Blanking)	deg	-50 to 50	-10, 10, -20, 20
PLBI	Line Blanking	µsec	0 to 20	11.7, 12.3, 9.1, 16.0
PLF1	10-1000 Hz LF Error	% Bar	0 to 50	0.0, 2.0, 0.0, 5.0
PLF5	50-550 Hz LF Error	% Bar	0 to 50	0.0, 2.0, 0.0, 5.0
PLFC	CCIR LF Error	% Bar	0 to 50	0.0, 2.0, 0.0, 5.0
PLND	Lum Nonlin Dist	%	0 to 50	- - -, 5.0, - - -, 10.0
PLSD	Sync Duration	µsec	0 to 20	4.6, 4.8, 4.5, 4.9
PLTD	Line Time Distortion	% Bar	0 to 40	0.0, 5.0, 0.0, 10.0
PLW9	S/N Lum-wgtd (569)	dB	0 to 999.9	58.0, - - -, 48.0, - - -
PMB1	MB Packet #1	% Flag	0 to 999.9	90, 110, 80, 120
PMB2	MB Packet #2	% Flag	0 to 999.9	90, 110, 80, 120
PMB3	MB Packet #3	% Flag	0 to 999.9	90, 110, 80, 120
PMB4	MB Packet #4	% Flag	0 to 999.9	90, 110, 80, 120
PMB5	MB Packet #5	% Flag	0 to 999.9	90, 110, 80, 120

Table B-14 (cont)

Keyword	Description	Units	Legal Range	Default
PMB6	MB Packet #6	% Flag	0 to 999.9	90, 110, 80, 120
PMBB	Multiburst Flag	% Bar	0 to 999.9	55, 65, 50, 70
PMBM	Multiburst Flag	mV	0 to 999.9	385, 455, 350, 490
PPBR	Pulse/Bar Ratio Err	% Bar	-90 to 90	-10, 10, -20, 20
PPDG	Peak Diff Gain	%	-50 to 50	- - -, 10, - - -, 20
PPDP	Peak Diff Phase	Deg	-50 to 50	- - -, 10, - - -, 20
PSAV	Sync Amplitude	mV	0 to 999.9	285, 315, 270, 330
PSBR	Sync/Bar Ratio	%	0 to 999.9	90, 110, 80, 120
PSBT	Sync to Bar Top	mV	0 to 2000	935, 1065, 870, 1130
PSCH	SCH Phase	deg	-90 to 90	-10, 10, -15, 15
PSFT	Sync Fall Time	nsec	0 to 999.9	170, 300, 140, 330
PSNP	S/N Periodic	dB	0 to 999.9	45.0, - - -, 40.0, - - -
PSNU	S/N Unweighted (567)	dB	0 to 999.9	60.0, - - -, 50.0, - - -
PSNW	S/N Lum-weighted (567)	dB	0 to 999.9	58.0, - - -, 48.0, - - -
PSRT	Sync Rise Time	nsec	0 to 999.9	170, 300, 140, 330
PSTB	Sync-to-Burst Start	µsec	0 to 20	5.5, 5.7, 5.4, 5.8
PSYA	Sync Amplitude Error	%	-50 to 100	-5, 5, -10, 10
PTTK	2T Pulse K-factor	% Kf	0 to 40	0, 1, 0, 4
PUN9	S/N Unweighted (569)	dB	0 to 999.9	60.0, - - -, 50.0, - - -

"R" Group: Auto Mode Limits (NTSC)

"R" keywords report on or set the limits for Auto Mode measurements on NTSC-standard VM700A's.

Get commands used with the "R" keywords have the form

```
get <keyword> <channel letter>
```

Set commands used with the "R" keywords have the form

```
set <keyword> <channel letter> <arg1> <arg2> <arg3> <arg4>
```

where <arg1> is the lower caution limit, <arg2> is the upper caution limit, <arg3> is the lower alarm limit, and <arg4> is the upper alarm limit.

The RFCB keyword is the only exception to these rules in the "R" group. get RFCB followed by a channel letter returns eighteen sets of limits, divided into three groups: amplitude error in percent for yellow, cyan, green, magenta, red, and blue; phase error in degrees for yellow, cyan, green, magenta, red, and blue; and chrominance/luminance ratio error in percent for yellow, cyan, green, magenta, red, and blue.

set RFCB is followed by a channel letter and 72 values (separated by spaces or tabs), specifying lower and upper caution limits and lower and upper alarm limits for each measurement listed in the previous paragraph.

Table B-15 lists the legal ranges and default values for each "R" group keyword used with the Set command.

Table B-15
"R" Keywords: Auto Mode Limits (NTSC)

Keyword	Description	Units	Legal Range	Default
R2LW	S/N.2 NTC7 Unwgtd	dB	0 to 100.0	- - -, - - -, 67.0, - - -
R2PR	2T Pulse K-Factor	% Kf	0 to 25	-.-.-, -.-.-, 0, 2.5
R2SU	S/N.2 Unif. Unwgtd	dB	0 to 100.0	- - -, - - -, 67.0, - - -
R2SW	S/N.2 Unif. Lum-wgtd	dB	0 to 100.0	- - -, - - -, 67.0, - - -
R2UW	S/N.2 NTC7 Lum-wgtd	dB	0 to 100.0	- - -, - - -, 67.0, - - -
RAPL	Average Picture Level	%	0 to 150	-.-.-, -.-.-, -.-.-, -.-.-
RBAA	Bar Amplitude	IRE	50 to 150	-.-.-, -.-.-, 96, 104
RBAP	Burst Amplitude	% sync	50 to 200	-.-.-, -.-.-, -.-.-, -.-.-
RBAT	Bar Top	% carr	0 to 50	10, 15 10 15
RBLL	Blanking Level	% carr	50 to 100	74, 76, 72.5, 77.5
RBVI	Blanking Variation	% Bar/IRE	0 to 40	-.-.-, -.-.-, -.-.-, -.-.-
RBVP	Blanking Variation	% carr	0 to 25	-.-.-, -.-.-, -.-.-, -.-.-

Table B-15 (cont)

Keyword	Description	Units	Legal Range	Default
RBWC	(FCC) Burst Width	cycles	5 to 20	-.-.-, -.-.-, 8, 11
RBZU	(FCC) Breezeway	µsec	0 to 5	0.28, - - -, 0.4, - - -
RCEP	number of consecutive error(s) before printing	- - -	1 to 3	2
RCGP	Chroma-Lum Gain	%	50 to 200	95, 105, 93, 107
RCLD	Chroma-Lum Delay	nsec	-200 to 200	-45, 45, -60, 60
RDGD	Differential Gain	%	0 to 25	0, 7, 0, 10
RDPD	Differential Phase	deg	0 to 50	0, 2.2, 0, 3
REEE	IEEE-511 ST Dist	% SD	0 to 25	0, 2, 0, 3
REWP	(FCC) Equalizer Width	% S.W.	20 to 80	46, 54, 45, 55
RFCB	FCC Color Bars Measurement (18 sets of results)	% (6 sets) Deg (6 sets) %(6 sets)	-50 to 100	-15, 15, -20, 20 (6 sets) -7.5, 7.5, -10, 10 (6 sets) -15, 15, -20, 20 (6 sets)
RFM1	FCC MB Packet #1	% flag	20 to 100	- - -, - - -, 57.1, 63
RFM2	FCC MB Packet #2	% flag	20 to 100	- - -, - - -, 56.2, 64.2
RFM3	FCC MB Packet #3	% flag	20 to 100	- - -, - - -, 54.8, 65.6
RFM4	FCC MB Packet #4	% flag	20 to 100	- - -, - - -, 53.5, 67.3
RFM5	FCC MB Packet #5	% flag	20 to 100	- - -, - - -, 56, 64.3
RFM6	FCC MB Packet #6	% flag	20 to 100	- - -, - - -, - - -, - - -
RFMI	FCC Multiburst Flag	% Bar/IRE	50 to 150	92.5, 107.5, 90, 110
RFMP	FCC Multiburst Flag	% carr	0 to 50	10.6, 14.4, 10, 15
RFPU	(FCC) Front Porch	µsec	0 to 5	1.4, - - -, 1.3, - - -
RFTD	Field Time Distortion		-20 to 20	- - -, - - -, -3, 3
RHB4	(FCC) H Blank 4 IRE	µsec	5 to 20	10.85, 11.35, 10.5, 11.5
RICP	ICPM	deg	-50 to 50	-2, 2, -3, 3
RLND	Lum NL Dist DY	%	0 to 25	0, 7, 0, 10
RLTD	Line Time Dist	%	0 to 25	0, 1.4, 0, 2
RN2C	NTC7 20 IRE Chroma	IRE	0 to 50	- - -, - - -, 15, 25
RN8C	NTC7 80 IRE Chroma	IRE	40 to 120	- - -, - - -, 75, 85
RNCI	NTC7 Chr-Lum Intmd	IRE	-50 to 50	- - -, - - -, -4, 4
RNCP	NTC7 Chr NL Phase	deg	0 to 25	- - -, - - -, 0, 5

Table B-15 (cont)

Keyword	Description	Units	Legal Range	Default
RNM1	NTC7 MB Packet #1	% flag	20 to 100	- - -, - - -, 47.6, 52.5
RNM2	NTC7 MB Packet #2	% flag	20 to 100	- - -, - - -, 46.8, 53.5
RNM3	NTC7 MB Packet #3	% flag	20 to 100	- - -, - - -, 45.7, 54.7
RNM4	NTC7 MB Packet #4	% flag	20 to 100	- - -, - - -, 44.6, 56.1
RNM5	NTC7 MB Packet #5	% flag	20 to 100	- - -, - - -, 46.7, 53.6
RNM6	NTC7 MB Packet #6	% flag	20 to 100	- - -, - - -, 43.6, 57.4
RNMI	NTC7 Multiburst Flag	% Bar/IRE	50 to 150	92.5, 107.5, 90, 110
RNMP	NTC7 Multiburst Flag	% carr	0 to 50	10.6, 14.4, 10,15
RPBR	Pulse/Bar Ratio	%	50 to 200	95.5, 104.5, 94, 106
RRBG	Rel Burst Gain	%	-50 to 100	-15, 15, -20, 20
RRBP	Rel Burst Phase	deg	-50 to 50	-7.5, 7.5, -10, 10
RRBW	RS-170A Burst Width	cycles	5 to 20	-.-.-, -.-.-, -.-.-, -.-.-
RREU	RS-170A Equalizer	µsec	0 to 5	2.21, 2.39, 2.18, 2.42
RRFP	RS-170A Front Porch	µsec	0 to 5	1.41, 1.59, 1.38, 1.62
RRHB	RS-170A H Blanking	µsec	5 to 20	10.71, 11.09, 10.65, 11.15
RRSS	RS-170A Sync-Setup	µsec	5 to 20	9.31, 9.49, 9.28, 9.52
RRSU	RS-170A Serration	µsec	2 to 10	4.61, 4.79, 4.58, 4.82
RRSW	RS-170A Sync Width	µsec	2 to 10	4.61, 4.79, 4.58, 4.82
RSBE	(FCC) Sync-to-Burst-End	µsec	5 to 20	5, 7.8, 5, 7.9
RSBS	Sync-to-Burst Start	µsec	0 to 25	5.21, 5.39, 5.18, 5.42
RSCH	SCH Phase	deg	-90 to 90	-45, 45, - - -, - - -
RSFN	(FCC) Sync Fall Time	nsec	0 to 999.9	0, 190, 0, 250
RSNP	S/N Periodic	dB	0 to 100	- - -, - - -, 57, - - -
RSNU	S/N Unified Unweighted	dB	0 to 100	- - -, - - -, 57, - - -
RSNW	S/N Unified Lum-Weighted	dB	0 to 100	- - -, - - -, 54, - - -
RSRN	(FCC) Sync Rise Time	nsec	0 to 999.9	0, 190, 0, 250
RSSU	(FCC) Sync-to-Setup	µsec	5 to 20	9.4, - - -, 9.2, - - -
RSVI	Sync Variation	% Bar/IRE	0 to 40	-.-.-, -.-.-, -.-.-, -.-.-
RSVP	Sync Variation	% carr	0 to 25	-.-.-, -.-.-, 0, 5
RSWU	(FCC) Serration Width	µsec	2 to 10	3.98, 4.92, 3.8, 5.1

Table B-15 (cont)

Keyword	Description	Units	Legal Range	Default
RSYA	Sync Amplitude	% Bar/IRE	20 to 80	37, 43, 36, 44
RSYU	(FCC) Sync Width	µsec	2 to 10	4.5, 5, 4.4, 5.1
RULW	S/N NTC7 Lum-Weighted	dB	0 to 100	- - -, - - -, 54, - - -
RUUW	S/N NTC7 Unweighted	dB	0 to 100	- - -, - - -, 57, - - -
RVAI	Burst Amplitude	% Bar/IRE	20 to 80	37, 43, 36, 44
RVB2	V Blank 20 IRE F1	lines	15 to 30	20.1, 20.9, 19.9, 21.1
RVB3	V Blank 20 IRE F2	lines	15 to 30	20.1, 20.9, 19.9, 21.1
RVB4	(FCC) V Blank 4 IRE F1	lines	15 to 30	18.5, 20.5, 18, 21
RVB5	(FCC) V Blank 4 IRE F2	lines	15 to 30	18.5, 20.5, 18, 21
RVCB	VIRS Chroma Ampl	% burst	50 to 200	95, 105, 90, 110
RVCI	VIRS Chroma Ampl	% Bar/IRE	20 to 80	38, 42, 36, 44
RVCP	VIRS Chroma Phase	deg	-50 to 50	-5, 5, -10, 10
RVLR	VIRS Luminance Ref	% Bar/IRE	20 to 80	47.5, 52.5, 45, 55
RVSU	VIRS Setup	% Bar/IRE	0 to 25	5.7, 9.3, 5, 10

"S" Group: Communication Setup

"S" keywords report on or set the values of VM700A communication parameters.

Get commands used with the "S" keywords have the form

```
get <keyword>
```

Set commands used with "S" group keywords take different arguments, depending on the keyword. Some "S" group keywords are "read-only" (i.e., cannot be used with the set command). The form of each Set command used with "S" group keywords is documented in the pages that follow.

Table B-16
"S" Keywords: Communication Setup

Keyword	Description	Keyword	Description
SMSD	Remote Control Message Display	SP1P	Port 1 Parity
SP0A	Port 0 Protocol	SP1R	Port 1 Reset Character
SP0B	Port 0 Baud Rate	SPCF	Copy Format
SP0C	Port 0 Character Size	SPCO	Control Port
SP0D	Port 0 Carrier Detect	SPCP	Copy Port
SP0F	Port 0 Flow Control	SPLF	Log Format
SP0P	Port 0 Parity	SPLP	Log Port
SP0R	Port 0 Reset Character	SPRC	Remote Control Port
SP1A	Port 1 Protocol	SPRF	Report Format
SP1B	Port 1 Baud Rate	SPRO	Remote Control Prompt
SP1C	Port 1 Character Size	SPRP	Report Port
SP1D	Port 1 Carrier Detect	SRIM	Non SLIP Interfacing Mode
SP1F	Port 1 Flow Control		

SMSD

SMSD specifies or reports on the destination for messages from the VM700A. SMSD is channel-independent. SMSD takes a single argument after the keyword. Legal values are Remote or VM700 Screen.

```
VM700A> get SMSD
Remote
VM700A> set SMSD VM700 Screen
VM700A> get SMSD
VM700 Screen
VM700A>
```

SP0A

SP0A returns the communications protocol for port 0. SP0A is read-only (cannot be used with the set command). SP0A takes a single argument after the keyword. Legal values are None or SLIP.

```
VM700A> get SP0A
SLIP
VM700A>
```

SP0B

SP0B returns the baud rate for port 0. SP0B is read-only (cannot be used with the set command). SP0B takes a single argument after the keyword. Legal values are 300, 600, 1200, 2400, 4800, 9600, or 19200.

```
VM700A> get SP0B
9600
VM700A>
```

SP0C

SP0C returns the character size (number of bits per character) for port 0. SP0C is read-only (cannot be used with the set command). SP0C takes a single argument after the keyword. Legal values are 7 or 8.

```
VM700A> get SP0C
8
VM700A>
```

SP0D

SP0D returns the status of carrier detect for port 0. SP0D is read-only (cannot be used with the set command). SP0D takes a single argument after the keyword. Legal values are disabled or enabled.

```
VM700A> get SP0D
disabled
VM700A>
```

SP0F

SP0F returns the flow control setting for port 0. SP0F is read-only (cannot be used with the set command). SP0F takes a single argument after the keyword. Legal values are CTS/RTS, XON/XOFF, or None.

```
VM700A> get SP0F
CTS/RTS
VM700A>
```

SP0P

SP0P returns the flow control setting for port 0. SP0P is read-only (cannot be used with the set command). SP0P takes a single argument after the keyword. Legal values are None, Odd, Even, Zero, or One.

```
VM700A> get SP0P
None
VM700A>
```

SP0R

SP0R returns the flow control setting for port 0. SP0R is read-only (cannot be used with the set command). SP0R takes a single argument after the keyword. Legal values are None, Ctrl-A, Ctrl-B, Ctrl-C, Ctrl-D, Ctrl-E, Ctrl-F, Ctrl-G, Ctrl-H, Ctrl-I, Ctrl-J, Ctrl-K, Ctrl-L, Ctrl-M, Ctrl-N, Ctrl-O, Ctrl-P, Ctrl-Q, Ctrl-R, Ctrl-S, Ctrl-T, Ctrl-U, Ctrl-V, Ctrl-W, Ctrl-X, Ctrl-Y, or Ctrl-Z.

```
VM700A> get SP0R
None
VM700A>
```

SP1A

SP1A returns the communications protocol for port 0. SP1A is read-only (cannot be used with the set command). SP1A takes a single argument after the keyword. Legal values are None or SLIP.

```
VM700A> get SP1A
SLIP
VM700A>
```

SP1B

SP1B returns the baud rate for port 0. SP1B is read-only (cannot be used with the set command). SP1B takes a single argument after the keyword. Legal values are 300, 600, 1200, 2400, 4800, 9600, or 19200.

```
VM700A> get SP1B
9600
VM700A>
```

SP1C

SP1C returns the character size (number of bits per character) for port 0. SP1C is read-only (cannot be used with the set command). SP1C takes a single argument after the keyword. Legal values are 7 or 8.

```
VM700A> get SP1C
8
VM700A>
```

SP1D

SP1D returns the status of carrier detect for port 0. SP1D is read-only (cannot be used with the set command). SP1D takes a single argument after the keyword. Legal values are disabled or enabled.

```
VM700A> get SP1D
disabled
VM700A>
```

SP1F

SP1F returns the flow control setting for port 0. SP1F is read-only (cannot be used with the set command). SP1F takes a single argument after the keyword. Legal values are CTS/RTS, XON/XOFF, or None.

```
VM700A> get SP1F
CTS/RTS
VM700A>
```

SP1P

SP1P returns the flow control setting for port 0. SP1P is read-only (cannot be used with the set command). SP1P takes a single argument after the keyword. Legal values are None, Odd, Even, Zero, or One.

```
VM700A> get SP1P
None
VM700A>
```

SP1R

SP1R returns the flow control setting for port 0. SP1R is read-only (cannot be used with the set command). SP1R takes a single argument after the keyword. Legal values are None, Ctrl-A, Ctrl-B, Ctrl-C, Ctrl-D, Ctrl-E, Ctrl-F, Ctrl-G, Ctrl-H, Ctrl-I, Ctrl-J, Ctrl-K, Ctrl-L, Ctrl-M, Ctrl-N, Ctrl-O, Ctrl-P, Ctrl-Q, Ctrl-R, Ctrl-S, Ctrl-T, Ctrl-U, Ctrl-V, Ctrl-W, Ctrl-X, Ctrl-Y, or Ctrl-Z.

```
VM700A> get SP1R
None
VM700A>
```

SPCF

SPCF returns or sets the copy format (i.e., the format used to format output when the Copy button is pressed). SPCF takes a single argument after the keyword. Legal values are Epson LQ, PostScript, ASCII Printer, or HP LaserJet.

```
VM700A> get SPCF
None
VM700A> set SPCF PostScript
VM700A> get SPCF
PostScript
VM700A>
```

SPCO

SPCO returns the name of the VM700A control port. The VM700A uses the control port to send characters in response to a **control** command. SPCO is read-only (cannot be used with the set command). SPCO takes a single argument after the keyword. Legal values are None, Serial Port 0, or Serial Port 1.

```
VM700A> get SPCO
None
VM700A>
```

SPCP

SPCP returns or sets the name of the VM700A copy port. SPCP takes a single argument after the keyword. Legal values are None, Serial Port 0, or Serial Port 1.

```
VM700A> get SPCP
None
VM700A> set SPCP Serial Port 0
VM700A> get SPCP
Serial Port 0
VM700A>
```

SPLF

SPLF returns or sets the log format (i.e., the foramt in which data logging information is sent). SPLF takes a single argument after the keyword. Legal values are Epson LQ, PostScript, ASCII Printer, or HP LaserJet.

```
VM700A> get SPLF
None
VM700A> set SPLF Epson LQ
VM700A> get SPLF
Epson LQ
VM700A>
```

SPLP

SPLP returns or sets the name of the VM700A log port. SPLP takes a single argument after the keyword. Legal values are None, Serial Port 0, or Serial Port 1.

```
VM700A> get SPLP
None
VM700A> set SPLP Serial Port 0
VM700A> get SPLP
Serial Port 0
VM700A>
```

SPRC

SPRC returns the name of the VM700A remote control port. SPRC is read-only (cannot be used with the set command). SPRC takes a single argument after the keyword. Legal values are None, Serial Port 0, or Serial Port 1.

```
VM700A> get SPRC
None
VM700A> set SPRC Serial Port 0
VM700A> get SPRC
Serial Port 0
VM700A>
```

SPRF

SPRF returns or sets the report format (i.e., the format used to format timed reports). SPRF takes a single argument after the keyword. Legal values are Epson LQ, PostScript, ASCII Printer, or HP LaserJet.

```
VM700A> get SPRF
None
VM700A> set SPRF Epson LQ
VM700A> get SPRF
Epson LQ
VM700A>
```

SPRO

SPRO returns or sets the VM700A remote control prompt. SPRO takes a single argument after the keyword. It consists of an arbitrary string up to 32 characters in length. "#" is a comment character; all characters appearing in the string after a "#" are ignored. To include a "#" in the string, precede it by a "\" character.

```
VM700A> set SPRO Archimedes>
Archimedes>
```

SPRP

SPRP returns or sets the name of the VM700A report port (i.e., the port used to print timed reports). SPRP takes a single argument after the keyword. Legal values are None, Serial Port 0, or Serial Port 1.

```
VM700A> get SPRP
None
VM700A> set SPRP Serial Port 1
VM700A> get SPRP
Serial Port 1
VM700A>
```

SRIM

SRIM returns the non-SLIP interfacing mode. SRIM is read-only (cannot be used with the set command). SRIM takes a single argument after the keyword. Legal values are Computer or Terminal.

```
VM700A> get SRIM
Terminal
VM700A>
```

"T" Group: Measure Mode Limits (PAL)

"T" keywords report on or set the limits for Measure Mode measurements on PAL-standard VM700A's.

Get commands used with the "T" keywords have the form

```
get <keyword> <channel letter>
```

Set commands used with the "T" keywords have the form

```
set <keyword> <channel letter> <arg1> <arg2>
```

where <arg1> and <arg2> are the lower and upper limits.

The TSX keywords are the only exceptions to these rules. Keywords TSX1 through TSX7 each specify the frequency at which the limits defined by TGN1 through TGN7 and TDL1 through TDL7 apply. Thus, the get command for any of the TSX keywords returns only one value, while the set command takes only one argument after the channel letter.

Table B-17 lists the legal ranges and default values for each "T" group keyword used with the Set command.

Table B-17
"T" Keywords: Measure Mode Limits (PAL)

Keyword	Description	Units	Legal Range	Default
TAMN	Chrominance AM Noise	dB rms	-100 to 0	----- -40.0
TBFE	Burst Freq Error	Hz	-360 to 360	-50.0 50.0
TBLV	Burst Level	mV	100 to 999.9	250.0 310.0
TBTE	Bar Tilt (Rec 569)	% Bar	-40 to 40	-5.0 5.0
TBWD	Burst Width	µsec	0 to 20	2.00 2.50
TCDL	Chrom Delay	nsec	-400 to 400	-40.0 40.0
TCGN	Chrom Gain	%	10 to 200	90.0 110.0
TCLB	Chrom Level Blue	mV	0 to 999.9	423.0 517.0
TCLC	Chrom Level Cyan	mV	0 to 999.9	598.0 730.0
TCLG	Chrom Level White	mV	0 to 999.9	0.0 10.0
TCLI	Chrom Lum Intermod	%	-50 to 50	-5.0 5.0
TCLK	Chrom Level Black	mV	-50 to 999.9	0.0 10.0
TCLM	Chrom Level Magenta	mV	0 to 999.9	558.0 682.0
TCLN	Chrom Level Green	mV	0 to 999.9	558.0 682.0
TCLR	Chrom Level Red	mV	0 to 999.9	598.0 730.0
TCLY	Chrom Level Yellow	mV	0 to 999.9	423.0 517.0
TCNL	Chrom Non Lin	%	-50 to 50	-5.0 5.0
TCPB	Chrom Phase Blue	deg	0 to 360	342.0 352.0
TCPC	Chrom Phase Cyan	deg	0 to 360	278.0 288.0
TCPM	Chrom Phase Magenta	deg	0 to 360	55.0 65.0
TCPN	Chrom Phase Green	deg	0 to 360	235.0 245.0
TCPR	Chrom Phase Red	deg	0 to 360	98.0 108.0
TCPY	Chrom Phase Yellow	deg	0 to 360	162.0 172.0
TDGN	Diff Gain	%	-50 to 50	-5.0 5.0
TDGP	Diff Gain (p-p)	%	0 to 50	0.0 10.0
TDL1	Sin X/X Group Delay	nsec	-500 to 500	-12.0 12.0
TDL2	Sin X/X Group Delay	nsec	-500 to 500	-12.0 12.0
TDL3	Sin X/X Group Delay	nsec	-500 to 500	-12.0 12.0
TDL4	Sin X/X Group Delay	nsec	-500 to 500	-12.0 12.0
TDL5	Sin X/X Group Delay	nsec	-500 to 500	-12.0 12.0

Table B-17 (cont)

Keyword	Description	Units	Legal Range	Default
TDL6	Sin X/X Group Delay	nsec	-500 to 500	-12.0 12.0
TDL7	Sin X/X Group Delay	nsec	-500 to 500	-40.0 40.0
TDPH	Diff Phase	deg	-50 to 50	-5.0 5.0
TDPP	Diff Phase (p-p)	deg	0 to 50	0.0 10.0
TEFS	H Blanking End from Sync	µsec	0 to 20	8.00 12.00
TEP5	Equalizer Pulse Width (50% sync)	µsec	0 to 20	2.10 2.80
TFES	Falling Edge SD	%	0 to 50	----- 1.0
TFTM	Fall Time	nsec	0 to 360	90.0 105.0
TGN1	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
TGN2	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
TGN3	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
TGN4	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
TGN5	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
TGN6	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
TGN7	Sin X/X Gain	dB	-50 to 20	-1.0 0.5
THAD	HAD	nsec	50 to 999.9	150.0 250.0
THBW	H Blanking Width	µsec	0 to 20	9.00 14.00
TICP	ICPM	deg	-20 to 0 (argument 1) 0 to 20 (argument 2)	-5.0 5.0
TJIF	Peak-Peak Jitter in a Frame	µsec	0 to 999.9	0.0 100.0
TK2T	K-2T	%Kf	0 to 99.9	0.0 5.0
TKPB	K-PB	%Kf	-50 to 50	-5.0 3.0
TLFE	Line Freq Error	%	-40 to 40	-1.0 1.0
TLLB	Lum Level Blue	mV	0 to 999.9	54.0 66.0
TLLC	Lum Level Cyan	mV	0 to 999.9	331.0 405.0
TLLG	Lum Level Grey	mV	0 to 999.9	630.0 770.0
TLLK	Lum Level Black	mV	-50 to 999.9	-10.0 10.0
TLLM	Lum Level Magenta	mV	0 to 999.9	195.0 234.0
TLLN	Lum Level Green	mV	0 to 999.9	277.0 339.0

Table B-17 (cont)

Keyword	Description	Units	Legal Range	Default
TLLR	Lum Level Red	mV	0 to 999.9	141.0 173.0
TLLY	Lum Level Yellow	mV	0 to 999.9	419.0 512.0
TLNL	Lum Non Lin	%	0 to 50	0.0 5.0
TLTJ	Peak to Peak Long Time Jitter	µsec	0 to 40	0.0 5.0
TMB1	MB Packet 1	dB	-40 to 40	-1.0 1.0
TMB2	MB Packet 2	dB	-40 to 40	-1.0 1.0
TMB3	MB Packet 3	dB	-40 to 40	-2.0 1.0
TMB4	MB Packet 4	dB	-40 to 40	-3.0 1.0
TMB5	MB Packet 5	dB	-40 to 40	-5.0 1.0
TMB6	MB Packet 6	dB	-40 to 40	----- 1.0
TMBF	Multiburst Flag	mV	0 to 999.9	400.0 750.0
TMBL	Bar Level (Ref. B1)	mV	500 to 999.9	650.0 750.0
TMBT	Line Time Distortion	% Bar	0 to 40	0.0 5.0
TMSL	Sync Level	mV	100 to 999.9	288.0 315.0
TMVL	Bar Level (Ref. Back Porch)	mV	500 to 999.9	650.0 750.0
TMVP	Sync-to-Bar-Top	mV	500 to 2000	950.0 1050.0
TNLV	Noise Level	dB rms	-100 to 0	----- -45.0
TPBR	PB Ratio	%	10 to 200	90.0 110.0
TPMN	Chrominance PM Noise	dB rms	-100 to 0	----- -40.0
TPNL	Chrom Phase Non Lin	deg	-50 to 50	-5.0 5.0
TRES	Rising Edge SD	%	0 to 50	----- 1.0
TRTM	Rise Time	nsec	0 to 360	90.0 105.0
TSBR	Sync/Bar Ratio	%	0 to 999.9	90.0 110.0
TSBS	Sync-Burst Start	µsec	4 to 9.9	5.50 5.70
TSCH	SCH phase	deg	-360 to 360	-45.0 45.0
TSDC	SD at Cursor	%	0 to 50	----- 1.0
TSFT	Sync Fall Time	nsec	10 to 999.9	100.0 300.0
TSK1	Head SW Skew (Field 1)	µsec	-40 to 40	-5.0 5.0
TSK2	Head SW Skew (Field 2)	µsec	-40 to 40	-5.0 5.0

Table B-17 (cont)

Keyword	Description	Units	Legal Range	Default
TSP5	Serration Pulse Width (50% sync)	µsec	0 to 20	4.40 5.00
TSRT	Sync Rise Time	nsec	10 to 999.9	100.0 300.0
TSTS	H Blanking Start-Sync	µsec	-10 to 0	-5.0 -0.5
TSWD	Sync Width	µsec	2 to 9.9	4.5 4.9
TSX1	Sin X/X Freq	MHz	0.1 to 5.8	0.50
TSX2	Sin X/X Freq	MHz	0.1 to 5.8	1.00
TSX3	Sin X/X Freq	MHz	0.1 to 5.8	2.00
TSX4	Sin X/X Freq	MHz	0.1 to 5.8	3.00
TSX5	Sin X/X Freq	MHz	0.1 to 5.8	4.00
TSX6	Sin X/X Freq	MHz	0.1 to 5.8	4.30
TSX7	Sin X/X Freq	MHz	0.1 to 5.8	4.80

"U" Group: Measure Mode Limits (NTSC)

"U" keywords report on or set the limits for Measure Mode measurements on NTSC-standard VM700A's.

Get commands used with the "U" keywords have the form

```
get <keyword> <channel letter>
```

Set commands used with the "U" keywords have the form

```
set <keyword> <channel letter> <arg1> <arg2>
```

where <arg1> and <arg2> are the lower and upper limits.

The USX keywords are the only exceptions to these rules. Keywords USX1 through USX7 each specify the frequency at which the limits defined by UGN1 through UGN7 and UDL1 through UDL7 apply. Thus, the get command for any of the USX keywords returns only one value, while the set command takes only one argument after the channel letter.

Table B-18 lists the legal ranges and default values for each "U" group keyword used with the Set command.

Table B-18
"U" Keywords: Measure Mode Limits (NTSC)

Keyword	Description	Units	Legal Range	Default
UAMN	Chrominance AM Noise	dB rms	-100 to 0	---- -40.0
UBFE	Burst Freq Error	Hz	-360 to 360	-50.0 50.0
UBLV	Burst Level	mV	100 to 999.9	250.0 310.0
UBTE	Bar Tilt (Rec 569)	% Bar	-40 to 40	-5.0 5.0
UBWD	Burst Width	cycles	0 to 20	8.50 9.50
UBZW	Breezeway	µsec	0 to 20	0.30 0.80
UCDL	Chrom Delay	nsec	-400 to 400	-40.0 40.0
UCF1	CF Packet 1	dB	-40 to 40	-3.0 1.0
UCF2	CF Packet 2	dB	-40 to 40	-2.0 1.0
UCF3	CF Packet 3	dB	-40 to 40	-1.0 1.0
UCF4	CF Packet 4	dB	-40 to 40	-2.0 1.0
UCF5	CF Packet 5	dB	-40 to 40	-3.0 1.0
UCFF	Chroma Freq Resp Flag	mV	0 to 999.9	600.0 800.0
UCGN	Chrom Gain	%	10 to 200	90.0 110.0
UCLB	Chrom Level Blue	mV	0 to 999.9	400.0 489.0
UCLC	Chrom Level Cyan	mV	0 to 999.9	567.0 693.0
UCLG	Chrom Level Grey	mV	0 to 999.9	0.0 10.0
UCLI	Chrom Lum Intermod	%	-50 to 50	-5.0 5.0
UCLK	Chrom Level Black	mV	0 to 999.9	0.0 10.0
UCLM	Chrom Level Magenta	mV	0 to 999.9	530.0 647.0
UCLN	Chrom Level Green	mV	0 to 999.9	530.0 647.0
UCLR	Chrom Level Red	mV	0 to 999.9	567.0 693.0
UCLY	Chrom Level Yellow	mV	0 to 999.9	400.0 489.0
UCNL	Chrom Non Lin	%	-50 to 50	-5.0 5.0
UCPB	Chrom Phase Blue	deg	0 to 360	342.0 352.0
UCPC	Chrom Phase Cyan	deg	0 to 360	278.0 288.0
UCPM	Chrom Phase Magenta	deg	0 to 360	55.0 65.0
UCPN	Chrom Phase Green	deg	0 to 360	235.0 245.0
UCPR	Chrom Phase Red	deg	0 to 360	98.0 108.0
UCPY	Chrom Phase Yellow	deg	0 to 360	162.0 172.0

Table B-18 (cont)

Keyword	Description	Units	Legal Range	Default
UDGN	Diff Gain	%	-50 to 50	-5.0 5.0
UDGP	Diff Gain (p-p)	%	0 to 50	0.0 10.0
UDL1	Sin X/X Group Delay	nsec	-500 to 500	-15.0 15.0
UDL2	Sin X/X Group Delay	nsec	-500 to 500	-15.0 15.0
UDL3	Sin X/X Group Delay	nsec	-500 to 500	-15.0 15.0
UDL4	Sin X/X Group Delay	nsec	-500 to 500	-15.0 15.0
UDL5	Sin X/X Group Delay	nsec	-500 to 500	-15.0 15.0
UDL6	Sin X/X Group Delay	nsec	-500 to 500	-15.0 15.0
UDL7	Sin X/X Group Delay	nsec	-500 to 500	-15.0 15.0
UDPH	Diff Phase	deg	-50 to 50	-5.0 5.0
UDPP	Diff Phase (p-p)	deg	0 to 50	0.0 10.0
UEFS	H Blanking End from Sync	µsec	0 to 20	8.00 12.00
UEP1	Equalizer Pulse Width (10% sync)	µsec	0 to 20	2.10 2.80
UEP5	Equalizer Pulse Width (50% sync)	µsec	0 to 20	2.10 2.80
UFES	Falling Edge SD	%	0 to 50	----- 1.0
UFPR	Front Porch	µsec	0 to 20	0.50 2.50
UFTM	Fall Time	nsec	0 to 360	120.0 130.0
UGN1	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
UGN2	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
UGN3	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
UGN4	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
UGN5	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
UGN6	Sin X/X Gain	dB	-50 to 20	-0.5 0.5
UGN7	Sin X/X Gain	dB	-50 to 20	-6.0 0.5
UHAD	HAD	nsec	50 to 999	200.0 300.0
UHBW	H Blanking Width	µsec	0 to 20	9.00 12.00
UICP	ICPM	deg	-20 to 0 (argument 1) 0 to -20 (argument 2)	-5.0 5.0
UJIF	Peak-Peak Jitter in a Frame	µsec	0 to 999.9	0.0 100.0

Table B-18 (cont)

Keyword	Description	Units	Legal Range	Default
UK2T	K-2T	%Kf	0 to 99.9	0.0 5.0
UKPB	K-PB	%Kf	-50 to 50	-5.0 3.0
ULFE	Line Freq Error	%	-40 to 40	-1.0 1.0
ULLB	Lum Level Blue	mV	0 to 999.9	97.0 119.0
ULLC	Lum Level Cyan	mV	0 to 999.9	360.0 440.0
ULLG	Lum Level Grey	mV	0 to 999.9	494.0 786.0
ULLK	Lum Level Black	mV	-50 to 999.9	48.0 59.0
ULLM	Lum Level Magenta	mV	0 to 999.9	231.0 282.0
ULLN	Lum Level Green	mV	0 to 999.9	311.0 380.0
ULLR	Lum Level Red	mV	0 to 999.9	182.0 222.0
ULLY	Lum Level Yellow	mV	0 to 999.9	445.0 544.0
ULNL	Lum Non Lin	%	0 to 50	0.0 5.0
ULTJ	Peak to Peak Long Time Jitter	µsec	0 to 40	0.0 5.0
UMB1	MB Packet 1	dB	-40 to 40	-1.0 1.0
UMB2	MB Packet 2	dB	-40 to 40	-1.0 1.0
UMB3	MB Packet 3	dB	-40 to 40	-2.0 1.0
UMB4	MB Packet 4	dB	-40 to 40	-3.0 1.0
UMB5	MB Packet 5	dB	-40 to 40	-5.0 1.0
UMB6	MB Packet 6	dB	-40 to 40	-10.0 1.0
UMBF	Multiburst Flag	mV	0 to 999.9	400.0 750.0
UMBL	Bar Level	mV	500 to 999.9	679.0 750.0
UMBT	Line Time Distortion	% Bar	0 to 40	0.0 5.0
UMSL	Sync Level	mV	100 to 999.9	271.0 300.0
UMVL	Video Level	mV	500 to 999.9	679.0 750.0
UMVP	Video Pk-Pk	mV	500 to 2000	950.0 1050.0
UNLV	Noise Level	dB rms	-100 to 0	----- -45.0
UPBR	PB Ratio	%	10 to 200	90.0 110.0
UPMN	Chrominance PM Noise	dB rms	-100 to 0	----- -40.0
UPNL	Chrom Phase Non Lin	deg	-50 to 50	-5.0 5.0
URES	Rising Edge SD	%	0 to 50	----- 1.0

Table B-18 (cont)

Keyword	Description	Units	Legal Range	Default
URTM	Rise Time	nsec	0 to 360	120.0 130.0
USBE	Sync-Burst End	µsec	0 to 20	7.50 8.50
USBR	Sync/Bar Ratio	%	0 to 999.9	90.0 110.0
USBS	Sync-Burst Start	µsec	4 to 9.9	5.30 5.32
USCH	SCH phase	deg	-360 to 360	-40.0 40.0
USDC	SD at Cursor	%	0 to 50	----- 1.0
USFT	Sync Fall Time	nsec	10 to 999.9	100.0 300.0
USK1	Head SW Skew Field 1	µsec	-40 to 40	-5.0 5.0
USK2	Head SW Skew Field 2	µsec	-40 to 40	-5.0 5.0
USP1	Serration Pulse Width (10% sync)	µsec	0 to 20	4.40 5.00
USP5	Serration Pulse Width (50% sync)	µsec	0 to 20	4.40 5.00
USRT	Sync Rise Time	nsec	10 to 999.9	100.0 300.0
USSU	Sync to Setup	µsec	0 to 20	8.00 11.00
USTS	H Blanking Start-Sync	µsec	-10 to 0	-5.0 -0.5
USWD	Sync Width	µsec	2 to 9.9	4.5 4.9
USX1	Sin X/X Freq	MHz	0.15 to 5.8	0.50
USX2	Sin X/X Freq	MHz	0.15 to 5.8	1.00
USX3	Sin X/X Freq	MHz	0.15 to 5.8	2.50
USX4	Sin X/X Freq	MHz	0.15 to 5.8	3.00
USX5	Sin X/X Freq	MHz	0.15 to 5.8	3.58
USX6	Sin X/X Freq	MHz	0.15 to 5.8	4.00
USX7	Sin X/X Freq	MHz	0.15 to 5.8	4.18

"V" Group: Video Source Selection

"V" keywords report on or set the active video source selection files.

Get commands used with the "V" keywords have the form

```
get <keyword>
```

Set commands used with the "V" keywords have the form

```
get <keyword> <file-name>
```

Table B-19 lists the "V" group keywords and their meanings.

Table B-19
"V" Keywords: Video Source Selection

Keyword	Description	Keyword	Description
VNCA	Source A NTSC Config File	VSCC	Source C PAL Config File
VNCB	Source B NTSC Config File	VSTA	Source A Video Standard
VNCC	Source C NTSC Config File	VSTB	Source B Video Standard
VSCA	Source A PAL Config File	VSTC	Source C Video Standard
VSCB	Source B PAL Config File	VSTE	Timed Events file

VNCA

VNCA returns or sets the NTSC Video_Source File for Source A. This file is found in directory /nvram0/ConfigFiles/Video_Source~Files/NTSC.

```
VM700A> get VNCA
NewMeas
VM700A>
```

VNCB

VNCB returns or sets the NTSC Video_Source File for Source B. This file is found in directory /nvram0/ConfigFiles/Video_Source~Files/NTSC.

```
VM700A> get VNCB
System~Default
VM700A>
```

VNCC

VNCC returns or sets the NTSC Video_Source File for Source C. This file is found in directory /nvram0/ConfigFiles/Video_Source~Files/NTSC.

```
VM700A> get VNCC
System~Default
VM700A>
```

VSCA

VSCA returns or sets the PAL Video_Source File for Source A. This file is found in directory /nvram0/ConfigFiles/Video_Source~Files/PAL.

```
VM700A> get VSCA
NewMeas
VM700A>
```

VSCB

VSCB returns or sets the PAL Video_Source File for Source B. This file is found in directory /nvram0/ConfigFiles/Video_Source~Files/PAL.

```
VM700A> get VSCB
System~Default
VM700A>
```

VSCC

VSCC returns or sets the PAL Video_Source File for Source C. This file is found in directory /nvram0/ConfigFiles/Video_Source~Files/PAL.

```
VM700A> get VSCC
System~Default
VM700A>
```

VSTA

VSTA returns or sets the video standard for Source A. Legal values are NTSC and PAL.

```
VM700A> get VSTA
NTSC
VM700A> set VSTA PAL
VM700A> get VSTA
PAL
VM700A>
```

VSTB

VSTA returns or sets the video standard for Source B. Legal values are NTSC and PAL.

```
VM700A> get VSTB
NTSC
VM700A> set VSTB PAL
VM700A> get VSTB
PAL
VM700A>
```

VSTC

VSTC returns or sets the video standard for Source C. Legal values are NTSC and PAL.

```
VM700A> get VSTC
NTSC
VM700A> set VSTC PAL
VM700A> get VSTC
PAL
VM700A>
```

VSTE

VSTE returns or sets the current Timed Events File. This file is found in directory /nvram0/ConfigFiles/Timed~Events.

```
VM700A> get VSTE
System~Default
VM700A> set VSTE My_Events
VM700A> get VSTE
My_Events
VM700A>
```

"W" Group: Audio Source Selection

"W" keywords report on or set the active video source selection files.

Get commands used with the "W" keywords have the form

```
get <keyword>
```

Set commands used with the "W" keywords have the form

```
get <keyword> <file-name>
```

Table B-20 lists the "W" group keywords and their meanings.

Table B-20
"W" Keywords: Audio Source Selection

Keyword	Description	Keyword	Description
WACA	Source A Audio Config File	WACC	Source C Audio Config File
WACB	Source B Audio Config File		

WACA

WACA returns or sets the Audio_Source File for Source A. This file is found in directory /nvram0/ConfigFiles/Audio_Source~Files.

```
VM700A> get WACA
0.33:00TEST
VM700A>
```

WACB

WACB returns or sets the Audio_Source File for Source B. This file is found in directory /nvram0/ConfigFiles/Audio_Source~Files.

```
VM700A> get WACB
0.33:00TEST
VM700A>
```

WACC

WACC returns or sets the Audio_Source File for Source C. This file is found in directory /nvram0/ConfigFiles/Audio_Source~Files.

```
VM700A> get WACC
0.33:00TEST
VM700A>
```

"X" Group: Camera Testing (NTSC)

"X" keywords report on or set the values of Camera Testing Option Configuration parameters. These keywords can only be used on NTSC-standard VM700A's equipped with Option 21 (Camera Testing).

Get commands used with the "X" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with "X" group keywords take different arguments, depending on the keyword. The form of each Set command used with "X" group keywords is documented in the pages that follow.

Table B-21 lists the "X" group keywords and their meanings.

Table B-21
"X" Keywords: Camera Testing (NTSC)

Keyword	Description	Keyword	Description
XCOS	Camera output standard	XB04	(B) Packet #4 (dB)
XSWP	Frequency response packets	XB05	(B) Packet #5 (dB)
XB6F	Packets beyond 6 MHz -- from	XB06	(B) Packet #6 (dB)
XB6T	Packets beyond 6 MHz -- to	XB07	(B) Packet #7 (dB)
XSWL	Frequency response field and line	XB08	(B) Packet #8 (dB)
XGDR	Reference (mv)	XB09	(B) Packet #9 (dB)
XG01	Y (G) Packet #1 (dB)	XB10	(B) Packet #10 (dB)
XG02	Y (G) Packet #2 (dB)	XB11	(B) Packet #11 (dB)
XG03	Y (G) Packet #3 (dB)	XB12	(B) Packet #12 (dB)
XG04	Y (G) Packet #4 (dB)	XRDR	Reference (mV)
XG05	Y (G) Packet #5 (dB)	XR01	(R) Packet #1 (dB)
XG06	Y (G) Packet #6 (dB)	XR02	(R) Packet #2 (dB)
XG07	Y (G) Packet #7 (dB)	XR03	(R) Packet #3 (dB)
XG08	Y (G) Packet #8 (dB)	XR04	(R) Packet #4 (dB)
XG09	Y (G) Packet #9 (dB)	XR05	(R) Packet #5 (dB)
XG10	Y (G) Packet #10 (dB)	XR06	(R) Packet #6 (dB)
XG11	Y (G) Packet #11 (dB)	XR07	(R) Packet #7 (dB)
XG12	Y (G) Packet #12 (dB)	XR08	(R) Packet #8 (dB)
XBDR	Reference (mV)	XR09	(R) Packet 9 (dB)
XB01	(B) Packet #1 (dB)	XR10	(R) Packet #10 (dB)
XB02	(B) Packet #2 (dB)	XR11	(R) Packet #11 (dB)
XB03	(B) Packet #3 (dB)	XR12	(R) Packet #12 (dB)

XCOS

XCOS returns or sets the camera output standard. Legal values for NTSC are: GBR 700, GBR 700 Setup, GBR 714, GBR 714 Setup, YPbPr SMPTE/EBU, YPbPr 714 Betacam Setup, YPbPr 714 Betacam, YPbPr 700 MII Setup.

```
VM700> get XCOS A
GBR 714
VM700>
```

XSWP

XSWP returns or sets the number of frequency response packets. Legal values are integers from 6 to 12, inclusive.

```
VM700> get XSWP A
12
VM700>
```

XB6F

XB6F returns or sets the number of the first frequency response packet beyond 6 MHz. Legal values are integers from 1 to 12, inclusive.

```
VM700> get XB6F A
4
VM700>
```

XB6T

XB6T returns or sets returns or sets the number of the last frequency response packet beyond 6 MHz. Legal values are integers from 1 to 12, inclusive.

```
VM700> get XB6T A
9
VM700>
```

XSWL

XSWL returns or sets the frequency response field and line. XSWL takes two arguments, a field number and a line number. Legal values for the first argument are integers from 1 to 2, inclusive. Legal values for the second argument are integers from 1 to 262, inclusive.

```
VM700> get XSWL A
1  120
VM700>
```

XGDR

XGDR returns or sets lower and upper limits in mV for the reference on the Y (green) channel. Legal values for each limit are integer or floating-point numbers from 0 to 999.9, inclusive.

```
VM700> get XGDR A
---  700.00
VM700>
```

XG01 - XG12

Keywords XG01 through XG12 return or set the lower and upper limits in dB for packets 1 through 12 on the Y channel. Legal values for each limit are integer or floating-point numbers from -40 to 40, inclusive.

```
VM700> get XG01 A
-1.00  0.00
VM700>
```

XBDR

XBDR returns or sets lower and upper limits in mV for the reference on the B (blue) channel. Legal values for each limit are integer or floating-point numbers from 0 to 999.9, inclusive.

```
VM700> get XBDR A
---  700.00
VM700>
```

XB01 - XB12

Keywords XB01 through XB12 return or set the lower and upper limits in dB for packets 1 through 12 on the B (blue) channel. Legal values for each limit are integer or floating-point numbers from -40 to 40, inclusive.

```
VM700> get XB01 A
-1.00  0.00
VM700>
```

XRDR

XRDR returns or sets lower and upper limits in mV for the reference on the R (red) channel. Legal values for each limit are integer or floating-point numbers from 0 to 999.9, inclusive.

```
VM700> get XRDR A
---  700.00
VM700>
```

XR01 - XR12

Keywords XR01 through XR12 return or set the lower and upper limits in dB for packets 1 through 12 on the R (red) channel. Legal values for each limit are integer or floating-point numbers from -40 to 40, inclusive.

```
VM700> get XR01 A
-1.00  0.00
VM700>
```

"Y" Group: Camera Testing (PAL)

"Y" keywords report on or set the values of Camera Testing Option Configuration parameters. These keywords can only be used on PAL-standard VM700A's equipped with Option 21 (Camera Testing).

Get commands used with the "Y" keywords have the form

```
get <keyword> <channel-letter>
```

Set commands used with "Y" group keywords take different arguments, depending on the keyword. The form of each Set command used with "Y" group keywords is documented in the pages that follow.

Table B-22 lists the "Y" group keywords and their meanings.

Table B-22
"Y" Keywords: Camera Testing (PAL)

Keyword	Description	Keyword	Description
YCOS	Camera output standard	YB04	(B) Packet #4 (dB)
YSWP	Frequency response packets	YB05	(B) Packet #5 (dB)
YB6F	Packets beyond 6 MHz -- from	YB06	(B) Packet #6 (dB)
YB6T	Packets beyond 6 MHz -- to	YB07	(B) Packet #7 (dB)
YSWL	Frequency response line	YB08	(B) Packet #8 (dB)
YGDR	Reference (mv)	YB09	(B) Packet #9 (dB)
YG01	Y (G) Packet #1 (dB)	YB10	(B) Packet #10 (dB)
YG02	Y (G) Packet #2 (dB)	YB11	(B) Packet #11 (dB)
YG03	Y (G) Packet #3 (dB)	YB12	(B) Packet #12 (dB)
YG04	Y (G) Packet #4 (dB)	YRDR	Reference (mV)
YG05	Y (G) Packet #5 (dB)	YR01	(R) Packet #1 (dB)
YG06	Y (G) Packet #6 (dB)	YR02	(R) Packet #2 (dB)
YG07	Y (G) Packet #7 (dB)	YR03	(R) Packet #3 (dB)
YG08	Y (G) Packet #8 (dB)	YR04	(R) Packet #4 (dB)
YG09	Y (G) Packet #9 (dB)	YR05	(R) Packet #5 (dB)
YG10	Y (G) Packet #10 (dB)	YR06	(R) Packet #6 (dB)
YG11	Y (G) Packet #11 (dB)	YR07	(R) Packet #7 (dB)
YG12	Y (G) Packet #12 (dB)	YR08	(R) Packet #8 (dB)
YBDR	Reference (mV)	YR09	(R) Packet 9 (dB)
YB01	(B) Packet #1 (dB)	YR10	(R) Packet #10 (dB)
YB02	(B) Packet #2 (dB)	YR11	(R) Packet #11 (dB)
YB03	(B) Packet #3 (dB)	YR12	(R) Packet #12 (dB)

YCOS

YCOS returns or sets the camera output standard. Legal values for PAL are: GBR and YPbPr SMPTE/EBU

```
VM700> get YCOS A
GBR 714
VM700>
```

YSWP

YSWP returns or sets the number of frequency response packets. Legal values are integers from 6 to 12, inclusive.

```
VM700> get YSWP A
12
VM700>
```

YB6F

YB6F returns or sets the number of the first frequency response packet beyond 6 MHz. Legal values are integers from 1 to 12, inclusive.

```
VM700> get YB6F A
4
VM700>
```

YB6T

YB6T returns or sets returns or sets the number of the last frequency response packet beyond 6 MHz. Legal values are integers from 1 to 12, inclusive.

```
VM700> get YB6T A
9
VM700>
```

YSWL

YSWL returns or sets the frequency response line. YSWL takes one argument. Legal values are integers from 1 to 625, inclusive.

```
VM700> get YSWL A
1  120
VM700>
```

YGDR

YGDR returns or sets lower and upper limits in mV for the reference on the Y (green) channel. Legal values for each limit are integer or floating-point numbers from 0 to 999.9, inclusive.

```
VM700> get YGDR A
---  700.00
VM700>
```

YG01 - YG12

Keywords YG01 through YG12 return or set the lower and upper limits in dB for packets 1 through 12 on the Y channel. Legal values for each limit are integer or floating-point numbers from -40 to 40, inclusive.

```
VM700> get YG01 A
-1.00  0.00
VM700>
```

YBDR

YBDR returns or sets lower and upper limits in mV for the reference on the B (blue) channel. Legal values for each limit are integer or floating-point numbers from 0 to 999.9, inclusive.

```
VM700> get YBDR A
---  700.00
VM700>
```

YB01 - YB12

Keywords YB01 through YB12 return or set the lower and upper limits in dB for packets 1 through 12 on the B (blue) channel. Legal values for each limit are integer or floating-point numbers from -40 to 40, inclusive.

```
VM700> get YB01 A
-1.00  0.00
VM700>
```

YRDR

YRDR returns or sets lower and upper limits in mV for the reference on the R (red) channel. Legal values for each limit are integer or floating-point numbers from 0 to 999.9, inclusive.

```
VM700> get YRDR A
---  700.00
VM700>
```

YR01 - YR12

Keywords YR01 through YR12 return or set the lower and upper limits in dB for packets 1 through 12 on the R (red) channel. Legal values for each limit are integer or floating-point numbers from -40 to 40, inclusive.

```
VM700> get YR01 A
-1.00  0.00
VM700>
```

Appendix C
MEASUREMENT RESULTS FILES

This chapter shows the results file produced by each VM700A application.

The sample files shown in this chapter are all produced by executing the following sequence of commands:

```
execute application
getresult
show application
```

The VM700A applications are grouped as shown in Tables C-1 through C-7.

Table C-1
NTSC/PAL Measurements

Auto	GroupDelay~SinX_X	MultiBurst
Bar~LineTime	H_Blank	Noise~Spectrum
Bounce	H_Timing	SCH_Phase
Burst~Frequency	ICPM	ShortTime~Distortion
ChromLum~GainDelay	Jitter	TwoField
Chrominance~AMPM	Jitter~Long_Time	VITS~ID (NTSC only)
Chrominance~FreqResp (NTSC only)	K_Factor	V_Blank
Chrominance~NonLinearity	Level~Meter	Video~Standard
ColorBar (NTSC)/ ColourBar (PAL)	Line~Frequency	
DGDP	Luminance~NonLinearity	

Table C-2
Diagnostics

ADC~Diagnostic	AudioProcessor~Diagnostic	Genlock~Diagnostic
Acquisition~Diagnostic	CalDac~Adjustment	Measure~Sinewave
AdcGain~Adjustment	Controller~Diagnostic	Measure~Squarewave
AnalogInput~Diagnostic	DiagsLoop	Measure~Temperature
AudioAnalog~Diagnostic	FilterBoard~Diagnostic	

Table C-3
Echo/Rounding (Option 1G) Measurements

Echo	Rounding~Errors	

Table C-4
Teletext (Option 20) Measurements

Teletext	SoundInSync	

Table C-5
Camera Testing (Option 21) Measurements

Colorimetry	Defects	Fixed_Pattern~Noise
Frequency~Response		

Table C-6
Component (Option 30) Measurements

Bowtie	Component~LevelMeter	Component~Vector
Component~Channel_Delay	Component~Multiburst	Lightning
Component~ColorBar	Component~Noise	
Component~K_Factor	Component~NonLinearity	

Table C-7
Audio (Options 40 and 41) Measurements

Audio~Analyzer	Calibrate~AudioBoard	View_Audio~Auto_Test
Audio~Monitor	Identify~Audio_Hardware	
Audio~Spectrum	Multitone	

NTSC/PAL RESULTS FILES

Note that results files from NTSC measurements frequently contain a line in the header that reads "Field = X Line = Y". The corresponding line in results files from PAL measurements reads simply "Line = Y". Except for this line, NTSC and PAL measurement results files do not differ, unless otherwise noted.

Auto (NTSC)

```
Channel A     Source  System Default     Mon Jan 20 16:42:20 1992
                          VM700A Video Measurement Set
System Default                            Violated Limits
                                            Lower  Upper
Source ID            ----                                  Not Found
Bar Top              ------  % Carr   **     10.0    15.0  ZC Pulse Unselected
Blanking Level       ------  % Carr   **     72.5    77.5  ZC Pulse Unselected
Bar Amplitude         100.5  IRE
Sync Amplitude         40.1  % Bar
Blanking Variation   ------  % Carr                        ZC Pulse Unselected
Blanking Variation      0.2  % Bar
Sync Variation       ------  % Carr   **      0.0     5.0  ZC Pulse Unselected
Sync Variation          0.1  % Bar
Burst Amplitude       100.3  % Sync
Burst Amplitude        40.2  % Bar
FCC H Blanking         11.24 us
FCC Sync Width          4.84 us
FCC Sync-Setup          9.66 us
FCC Front Porch         1.57 us
Sync to Burst End       7.86 us       *       5.00    7.80
Breezeway Width         0.54 us
FCC Burst Width         8.9  Cycles
Sync Risetime         141    ns
Sync Falltime         140    ns
RS-170A H Blanking     11.42 us       **     10.65   11.15
RS-170A Sync Width      4.70 us
RS-170A Sync-Setup      9.59 us       **      9.28    9.52
RS-170A Front Porch     1.65 us       **      1.38    1.62
Sync to Burst Start     5.31 us
RS-170A Burst Width     9.0  Cycles
V Blank 4 IRE F1       21.0  Lines    *      18.5    20.5
V Blank 4 IRE F2       21.0  Lines    *      18.5    20.5
V Blank 20 IRE F1      21.0  Lines    *      20.1    20.9
V Blank 20 IRE F2      21.1  Lines    *      20.1    20.9
FCC Equalizer          51.0  % S.W.
FCC Serration           4.51 us
RS-170A Equalizer       2.33 us
RS-170A Serration       4.66 us
VIRS Setup              7.5  % Bar
VIRS Luminance Ref     50.1  % Bar
VIRS Chroma Ampl       99.9  % Burst
VIRS Chroma Ampl       40.1  % Bar
VIRS Chroma Phase      -0.1  Deg
Line Time Distortion    0.1  %
Pulse/Bar Ratio        99.1  %
2T Pulse K-Factor       0.2  % Kf
IEEE-511 ST Dist     ------  % SD     **      0.0     3.0  No NTC-7 Comp VITS
S/N NTC7 Unweighted    77.2  dB                            RMS
S/N NTC7 Lum-Wghtd     80.6  dB                            RMS
```

```
S/N Unif Unweighted    76.7  dB                            RMS
S/N Unif Lum-Wghtd     80.7  dB                            RMS
S/N Periodic         ------  dB       **      57.0  ------  Random >> Periodic
S/N.2 NTC7 Unwghtd     78.0  dB                            RMS
S/N.2 NTC7 Lum-Wghtd   82.1  dB                            RMS
S/N.2 Unif Unwghtd     77.4  dB                            RMS
S/N.2 Unif Lum-Wghtd   82.3  dB                            RMS
Chroma-Lum Delay        1.7  ns
Chroma-Lum Gain       100.3  %
Differential Gain       0.21 %                             At  48% APL
Differential Phase      0.09 Deg                           At  48% APL
Lum Non-Linearity       0.39 %                             At  48% APL
Relative Burst Gain     0.00 %                             At  48% APL
Relative Burst Phase   -0.02 Deg                           At  48% APL
FCC Multiburst Flag  ------  % Carr   **      10.0    15.0  No FCC Multiburst
FCC Multiburst Flag  ------  % Bar    **      90.0   110.0  No FCC Multiburst
FCC MB Packet #1     ------  % Flag   **      57.1    63.0  No FCC Multiburst
FCC MB Packet #2     ------  % Flag   **      56.2    64.2  No FCC Multiburst
FCC MB Packet #3     ------  % Flag   **      54.8    65.6  No FCC Multiburst
FCC MB Packet #4     ------  % Flag   **      53.5    67.3  No FCC Multiburst
FCC MB Packet #5     ------  % Flag   **      56.0    64.3  No FCC Multiburst
FCC MB Packet #6     ------  % Flag                         No FCC Multiburst
NTC7 Multiburst Flag ------  % Carr   **      10.0    15.0  ZC Pulse Unselected
NTC7 Multiburst Flag  100.0  % Bar
NTC7 MB Packet #1      49.9  % Flag
NTC7 MB Packet #2      49.8  % Flag
NTC7 MB Packet #3      50.0  % Flag
NTC7 MB Packet #4      50.2  % Flag
NTC7 MB Packet #5      50.1  % Flag
NTC7 MB Packet #6      50.1  % Flag
NTC7 20 IRE Chroma     20.0  IRE                           (Ref 40 IRE Chr)
NTC7 80 IRE Chroma     79.9  IRE                           (Ref 40 IRE Chr)
NTC7 Chr NL Phase       0.2  Deg
NTC7 Chr-Lum Intmd      0.1  IRE                           (Ref Lum Pedestal)
ICPM                 ------  Deg      **      -3.0     3.0  ZC Pulse Unselected
SCH Phase              -3.1  Deg
Field Time Dist      ------- % Bar    **      -3.00    3.00 Not Found
FCC Color Bars
            Amplitude Error      Phase Error     Chr/Lum Ratio Error
               ( % )               ( Deg )              ( % )
   Yellow        0.1                 0.0                -0.1
   Cyan          0.2                 0.0                -0.1
   Green         0.2                 0.1                -0.0
   Magenta      -0.1                 0.2                -0.5
   Red           0.1                 0.1                -0.2
   Blue          0.3                 0.2                -0.3
```

Auto (PAL)

```
Channel B     Source  System Default     Mon Jan 20 16:45:12 1992
                           VM700A Video Measurement Set
System Default                              Violated Limits
                                              Lower  Upper
Source ID            ----                                      Not Found
Luminance Bar Ampl   ------  mV        **     600.0   800.0   Bar Not Found
Luminance Bar Ampl   ------  % Carr    **      55.0    73.0   Bar Not Found
Lum Bar Ampl Err     ------  %         **     -10.0    10.0   Bar Not Found
Line Time Distortion ------  % Bar     **       0.0    10.0   Bar Not Found
Bar Tilt (Rec 569)   ------  % Bar     **     -10.0    10.0   Bar Not Found
Bar Rise Time        ------  ns        **     160.0   240.0   Bar Not Found
Baseline Distortion  ------  % Bar     **      -2.0     2.0   Bar Not Found
Blanking Level       ------  % Carr    **      69.0    79.0   Bar Not Found
Sync/Bar (Rel 3/7)   ------  %         **      80.0   120.0   Bar Not Found
Sync to Bar Top      ------  mV        **     870.0  1130.0   Bar Not Found
Pulse/Bar Ratio Err  ------  % Bar     **     -20.0    20.0   Pulse Not Found
2T Pulse K-factor    ------  % Kf      **       0.0     4.0   Pulse Not Found
C/L Gn Err (Mod Bar) ------  % Bar     **     -20.0    20.0   Not Found
Chr/Lum Delay Ineq   ------  ns        **     -50.0    50.0   Pulse Not Found
C/L Gn Err (Mod Pls) ------  % Bar     **     -20.0    20.0   Not Found
Lum. Nonlin. Dist.   ------  %         **    ------    10.0   No Luminance Steps
Chrom Ref Ampl Err   ------  %         **     -10.0    10.0   Not Found
Pk-Pk Diff Gain      ------  %         **    ------    20.0   Not Found
Peak Diff Gain       ------  %         **    ------    20.0   Not Found
Pk-Pk Diff Phase     ------  Deg       **    ------    20.0   Not Found
Peak Diff Phase      ------  Deg       **    ------    20.0   Not Found
Chr/Lum Intermod     ------  % Bar     **     -10.0    10.0   Not Found
Sync Amplitude        302.2  mV
Sync Ampl Error         0.7  %
Residual Carrier     ------  % Carr    **       7.5    15.0   Bar Not Found
Sync-to-Burst Start    5.51  us
Burst Duration         2.18  us
Burst Duration         9.7   Cycles
Burst Amplitude       307.7  mV
Burst Ampl Error        2.6  %
Burst Ampl Diff         0.5  %
Burst Quad Error       -0.2  Deg
SCH Phase               5.8  Deg
Sync Duration          4.68  us
Sync Rise Time        251.3  ns
Sync Fall Time        251.2  ns
Front Porch          ------- us        **      0.50    3.00 No Blanking Edge
Line Blanking        ------- us        **      9.10   16.00 No Blanking Edge
Broad Pulse Sep        4.66  us
Equalizing Pulse       2.34  us
Multiburst Flag      ------  % Bar     **      50.0    70.0   No Multiburst
Multiburst Flag      ------  mV        **     350.0   490.0   No Multiburst
MB Packet #1         ------  % Flag    **      80.0   120.0   No Multiburst
MB Packet #2         ------  % Flag    **      80.0   120.0   No Multiburst
MB Packet #3         ------  % Flag    **      80.0   120.0   No Multiburst
MB Packet #4         ------  % Flag    **      80.0   120.0   No Multiburst
MB Packet #5         ------  % Flag    **      80.0   120.0   No Multiburst
MB Packet #6         ------  % Flag    **      80.0   120.0   No Multiburst
CCIR LF Error           0.1  % Bar                            No Bar - Ref. 700 mV
50-550 Hz LF Error      0.2  % Bar                            No Bar - Ref. 700 mV
10-1000 Hz LF Error     0.1  % Bar                            No Bar - Ref. 700 mV
S/N Unweighted (567)   79.2  dB                               No Bar - Ref. 700 mV
```

```
S/N Chr-wgtd           83.6  dB                            No Bar - Ref. 700 mV
S/N Periodic           72.0  dB                            No Bar - Ref. 700 mV
S/N Unweighted (569)   80.7  dB                            No Bar - Ref. 700 mV
S/N Lum-wgtd (569)     88.2  dB                            No Bar - Ref. 700 mV
S/N.2 Unwgtd (567)     75.0  dB                            No Bar - Ref. 700 mV
S/N.2 Lum-wgtd (567)   78.4  dB                            No Bar - Ref. 700 mV
S/N.2 Chr-wgtd         72.6  dB                            No Bar - Ref. 700 mV
S/N.2 Unwgtd (569)     76.9  dB                            No Bar - Ref. 700 mV
S/N.2 Lum-wgtd (569)   84.0  dB                            No Bar - Ref. 700 mV
ICPM (Absolute)      ------  Deg      **    -20.0    20.0  No Luminance Steps
ICPM (Rel Blanking)  ------  Deg      **    -20.0    20.0  No Luminance Steps
Field Time Dist      ------  %        **     -2.0     2.0  Not Found
```

Bar-LineTime (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:00:05

Bar & LineTime                                  Waveform->FCC   Composite
Field = 1 Line =  18
Average Off
----------------------------------------------------------------------
Bar Level(Ref. b1)        99.5  IRE
Bar Level(Back Porch)     99.6  IRE
Sync Level                40.0  IRE
Sync to Bar Top          139.6  IRE
Sync/Bar Ratio           100.5  %
LineTime Dist (Rec. 567)   0.1  %
Bar Tilt (Rec. 569)        0.1  %
Bar Width                 18.0  u sec
----------------------------------------------------------------------
```

Bounce (NTSC & PAL)

```
  Measurement Results        Channel A        Fri Aug 21 08:52:53

Bounce
Average Off
---------------------------------------------------------------------
     (High APL)
Settling Time              0.0  sec
Blanking Dev. (+)          0.2  %
Blanking Dev. (-)         -0.2  %
Blanking Dev. (P-P)        0.4  %
Sync Amplitude            40.0  IRE
Sync Amp. Dev. (+)         0.5  %
Sync Amp. Dev. (-)        -0.4  %
Sync Amp. Dev. (P-P)       0.9  %
Bounce Amplitude         100.1  IRE
Bounce Amp. Dev. (+)       0.1  %
Bounce Amp. Dev. (-)      -0.1  %
Bounce Amp. Dev. (P-P)     0.3  %
     (Low APL)
Settling Time              0.0  sec
Blanking Dev. (+)          0.1  %
Blanking Dev. (-)         -0.2  %
Blanking Dev. (P-P)        0.3  %
Sync Amplitude            39.9  IRE
Sync Amp. Dev. (+)         0.6  %
Sync Amp. Dev. (-)        -0.6  %
Sync Amp. Dev. (P-P)       1.1  %
     (High,Low APL)
Blank Level Diff.          0.1  %
Sync Amp. Diff            -0.1  %
---------------------------------------------------------------------
```

Burst~Frequency (NTSC & PAL)

```
  Measurement Results        Channel A        Wed Jan 30 16:03:45

Burst Frequency Measurement        (Ref. Ch-B Burst)
Average Off
---------------------------------------------------------------------
Burst Frequency Error     -0.2  Hz
---------------------------------------------------------------------
```

ChromLum~GainDelay (NTSC & PAL)

```
  Measurement Results        Channel A        Wed Jan 30 16:04:22

Chrom/Lum Gain Delay                          Waveform->FCC   Composite
Field = 1 Line =  18
Average Off
---------------------------------------------------------------------
Chroma Gain               92.5  %
Chroma Delay              -3.7  n sec
---------------------------------------------------------------------
```

Chrominance~AMPM (NTSC & PAL)

```
  Measurement Results           Channel A        Wed Jan 30 16:04:59

Chrominance AMPM                                 Waveform->appropriate
Full Field (Both Fields)
Band width  100Hz to 500kHz
Average Off
---------------------------------------------------------------------
AM Noise                 -64.3  dB rms
PM Noise                 -64.2  dB rms
---------------------------------------------------------------------
(0 dB = 714 mV p-p with AGC for 100% Chrominance Level)
```

Chrominance~FreqResp (NTSC only)

```
  Measurement Results           Channel A        Wed Jan 30 16:06:41

Chrominance Freq Resp                            Waveform->FCC Multi Burst
Field = 1 Line =  50
Average Off
---------------------------------------------------------------------
0 dB = Packet #3          56.4  IRE
Packet #1 0.5 Mhz         0.15  dB
Packet #2 1.2 Mhz         0.10  dB
Packet #3 2.0 Mhz         0.00  dB
Packet #4 3.0 Mhz        -5.88  dB         *       -2.00      1.00
Packet #5 4.1 Mhz        -5.14  dB         *       -3.00      1.00
---------------------------------------------------------------------
```

Chrominance~NonLinearity (NTSC & PAL)

```
  Measurement Results           Channel A        Wed Jan 30 16:07:20

Chrominance NonLinearity                         Waveform->Mod 3 Step
Field = 1 Line =  50
Average Off
---------------------------------------------------------------------
Chroma Amp (Packet 1)           1.4  %
Chroma Amp (Packet 2)           0.0  Ref.
Chroma Amp (Packet 3)          -0.6  %
Chroma Phase (Packet 1)        -0.7  deg.
Chroma Phase (Packet 2)         0.0  Ref.
Chroma Phase (Packet 3)        -0.4  deg.
Chroma Intermod. (Packet 1)     0.2  %
Chroma Intermod. (Packet 2)     0.1  %
Chroma Intermod. (Packet 3)    -0.1  %
---------------------------------------------------------------------
```

ColorBar (NTSC & PAL)

Note that the name of this measurement (according to the results file) is "ColorBar" for NTSC and "ColourBar" for PAL.

```
  Measurement Results         Channel A         Wed Jan 30 16:07:45

ColorBar                                        Waveform->FCC Color Bar
Field = 1 Line =  50
Average Off
-----------------------------------------------------------------------
Luma Level (Gray)          95.2  IRE
Luma Level (Yellow)        64.3  IRE
Luma Level (Cyan)          50.7  IRE
Luma Level (Green)         42.8  IRE        *       43.5       53.2
Luma Level (Magenta)       30.0  IRE        *       32.3       39.5
Luma Level (Red)           22.1  IRE        *       25.5       31.1
Luma Level (Blue)           8.2  IRE        *       13.6       16.7
Luma Level (Black)          0.1  IRE        *        6.7        8.3
Chroma Level (Gray)         0.1  IRE
Chroma Level (Yellow)      60.1  IRE
Chroma Level (Cyan)        86.3  IRE
Chroma Level (Green)       81.4  IRE
Chroma Level (Magenta)     81.7  IRE
Chroma Level (Red)         87.4  IRE
Chroma Level (Blue)        61.8  IRE
Chroma Level (Black)        0.0  IRE
Chroma Phase (Yellow)     166.7  deg
Chroma Phase (Cyan)       283.5  deg
Chroma Phase (Green)      240.9  deg
Chroma Phase (Magenta)     60.9  deg
Chroma Phase (Red)        103.5  deg
Chroma Phase (Blue)       347.0  deg
-----------------------------------------------------------------------
```

DGDP (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:08:06

Differential Gain & Phase                       Waveform->NTC-7 Composite
Field = 2 Line =  18
Average Off
-----------------------------------------------------------------------
Differential Gain (min)   -6.10  %          *       -5.00        5.00
Differential Gain (max)    0.79  %
Differential Gain (p-p)    6.83  %
Differential Phase (min)  -1.41  deg
Differential Phase (max)   1.02  deg
Differential Phase (p-p)   2.43  deg
-----------------------------------------------------------------------
```

GroupDelay~SinX_X (NTSC & PAL)

```
  Measurement Results          Channel A          Wed Jan 30 16:08:52

Group Delay & Gain vs Frequency (SinX/X)          Waveform->Sin X/X
Field = 2 Line =  44
Reference Frequency at 0.20 MHz
Average Off
-----------------------------------------------------------------------
Amplitude (0.5 MHz)          -0.0  dB
Amplitude (1.0 MHz)          -0.1  dB
Amplitude (2.5 MHz)          -0.2  dB
Amplitude (3.0 MHz)          -0.2  dB
Amplitude (3.6 MHz)          -0.4  dB
Amplitude (4.0 MHz)          -0.6  dB        *        -0.5         0.5
Amplitude (4.2 MHz)          -0.6  dB
GroupDelay (0.5 MHz)         -5.5  n sec
GroupDelay (1.0 MHz)         15.9  n sec     *       -15.0        15.0
GroupDelay (2.5 MHz)          2.9  n sec
GroupDelay (3.0 MHz)          3.2  n sec
GroupDelay (3.6 MHz)         19.3  n sec     *       -15.0        15.0
GroupDelay (4.0 MHz)         13.8  n sec
GroupDelay (4.2 MHz)         -6.5  n sec
Cursor (3.58 MHz)            -0.4  dB     19.3 n sec
-----------------------------------------------------------------------
```

H_Blank (NTSC & PAL)

```
  Measurement Results          Channel A          Wed Jan 30 16:09:28

H_Blank  4 IRE (Field 1)  (Line  22 to 260)
-----------------------------------------------------------------------
H Blank Start             -1.69  u sec
H Blank End                9.47  u sec
H Blank Width             11.15  u sec
-----------------------------------------------------------------------
```

H_Timing (NTSC & PAL)

```
  Measurement Results          Channel A          Wed Jan 30 16:09:50

H Timing  (RS-170A)
Field = 2 Line =  44
Average Off
-----------------------------------------------------------------------
Sync Rise Time            148.8   n sec
Sync Fall Time            143.0   n sec
Sync Width                  4.71  u sec
Sync Level                 40.9   IRE
Burst Level                39.2   IRE
Sync to Burst Start         5.31  u sec
Burst Width                 9.0   cycles
Front Porch                 1.70  u sec
Sync to SetUp               9.48  u sec
-----------------------------------------------------------------------
```

ICPM (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:10:27

ICPM
Field = 2  Line =  44
Average Off
--------------------------------------------------------------------
Min Angle              = -67.71 degrees      *    -5.0      5.0
Max Angle              =  66.33 degrees      *    -5.0      5.0
Pk-Pk Angle            = 134.04 degrees
(Absolute)
--------------------------------------------------------------------
```

Jitter (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:10:50

Jitter Measurement   (Line  20 to 250)
Average Off
--------------------------------------------------------------------
Peak to Peak Jitter        4.2  n sec
--------------------------------------------------------------------
```

Jitter~Long_Time (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:11:20

Jitter Long Time Measurement   Frequency Lock Speed    1 sec
Average Off
--------------------------------------------------------------------
Peak to Peak Jitter       0.006  u sec
--------------------------------------------------------------------
```

K_Factor (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:11:49

2T Pulse K Factor                                 Waveform->FCC   Composite
Field = 1 Line =  18
Graticule EIA-2T
Average Off
--------------------------------------------------------------------
K-2T                        0.3  % KF
K-PB                       -0.1  % KF
PB Ratio                   99.6  %
HAD                       254.9  n sec
--------------------------------------------------------------------
```

Level~Meter (NTSC & PAL)

```
  Measurement Results         Channel A        Fri Dec 20 15:57:43

Level Meter                                    Waveform->FCC Multi Burst
Field = 1 Line = 100
Average Off
-----------------------------------------------------------------------
Level(b-a)                 288.3  mV
-----------------------------------------------------------------------
a <--  6.5 u sec
b <-- 35.5 u sec
```

Line~Frequency (NTSC & PAL)

```
  Measurement Results         Channel A        Wed Jan 30 16:12:03

Line Frequency Measurement
Average Off
-----------------------------------------------------------------------
Line Frequency Error     -0.007  %
Line Frequency           15.733  kHz
Field Frequency          59.94   Hz
-----------------------------------------------------------------------
```

Luminance~NonLinearity (NTSC & PAL)

```
  Measurement Results         Channel A        Wed Jan 30 16:12:17

Luma Non Linearity                             Waveform->FCC   Composite
Field = 1 Line =  18
Average Off
-----------------------------------------------------------------------
LumaNonLinearity (p-p)      3.3   %
-----------------------------------------------------------------------
```

MultiBurst (NTSC & PAL)

```
  Measurement Results         Channel A        Wed Jan 30 16:12:40

Multi Burst                                    Waveform->FCC Multi Burst
Field = 1 Line =  17
Average Off
-----------------------------------------------------------------------
0 dB = 60 % of Flag       101.9  IRE
Packet #1 0.50 MHz          0.15  dB
Packet #2 1.25 MHz          0.10  dB
Packet #3 2.00 MHz          0.04  dB
Packet #4 3.00 MHz         -0.06  dB
Packet #5 3.58 MHz         -0.24  dB
Packet #6 4.10 MHz         -0.37  dB
-----------------------------------------------------------------------
```

Noise~Spectrum (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:13:50

Noise Spectrum                                  Waveform->Pedestal
Field = 1 Line =  10
Band width 100kHz to 5.0MHz (SC trap) (Tilt Null)
Average Off
----------------------------------------------------------------------
Noise Level              -60.3  dB rms
Cursor 1                 -70.0  dB p-p    at  1.96 MHz
Cursor 2                -107.2  dB p-p    at  3.58 MHz   -37.2 dB Diff
Noise Area in Cursors    -66.3  dB rms
----------------------------------------------------------------------
```

SCH_Phase (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:14:36

SCH Phase
Average off
-------------------------------------------------------------------------
SCH Phase                        -2.3 deg
-------------------------------------------------------------------------
```

ShortTime~Distortion (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:15:07

Short Time Distortion                           Waveform->NTC-7 Composite
Field = 2 Line =  18
Graticule IEEE-511
Average Off
----------------------------------------------------------------------
Rising Edge                0.7  % SD
Rise Time                131.5  nSec      *       120.0     130.0
Falling Edge               0.9  % SD
Fall Time                130.0  nSec
----------------------------------------------------------------------
```

TwoField (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:15:37

TwoField
Average Off
----------------------------------------------------------------------
Field Time Dist           13.1  %
----------------------------------------------------------------------
APL =  51.8 %
Slow Clamp at Back Porch
Luminance at (35.0 usec), Sync & Back Porch are displayed
```

VITS~ID (NTSC only)

```
  Measurement Results          Channel A          Wed Jan 30 16:16:09

Signal ID (System Line)                           Waveform->NTC-7 Composite
Field = 2 Line =  18
-----------------------------------------------------------------------
  Field 1                                Field 2
Line 15 --> GCR 8 Fields Seq.          Line 15 --> GCR 8 Fields Seq.
Line 16 --> VIRS                       Line 16 --> Sin X/X
Line 17 --> FCC Multi Burst            Line 17 --> NTC-7 Combination
Line 18 --> FCC   Composite            Line 18 --> NTC-7 Composite
Line 19 --> Pedestal                   Line 19 --> Red Field
Line 20 --> Luminance Bar              Line 20 --> Mod Pedestal F-line
-----------------------------------------------------------------------
```

V_Blank (NTSC & PAL)

```
  Measurement Results          Channel A          Wed Jan 30 16:16:49

V Timing Measurement
Average Off
-----------------------------------------------------------------------
Equalizer Pulse            2.33  u sec
Equalizer Pulse 10%        2.47  u sec
Serration Pulse            4.66  u sec
Serration Pulse 10%        4.51  u sec
-----------------------------------------------------------------------
```

Video~Standard (NTSC & PAL)

```
    Video Standard                                Wed Jan 30 16:17:56

-----------------------------------------------------------------------
Source A: NTSC
Source B: NTSC
Source C: PAL
-----------------------------------------------------------------------
```

DIAGNOSTICS RESULTS FILES

The following Diagnostics applications produce no results file: AdcGain~Adjustment, Audio_TS_Tool, CalDac~Adjustment, Diagsloop, Measure~Sinewave, Measure~Squarewave, Warm_Reboot.

ADC~Diagnostic

```
                             ADC Diagnostic
        Test             measmt     units       min         max        result
Mode checks ---------   --------    ------   --------    --------      -Pass-
ADC bit patterns ----   --------    ------   --------    --------      -Pass-
```

Acquisition~Diagnostic

```
                         Acquisition Diagnostic
        Test             measmt     units       min         max        result
Identify Board ------   --------    ------   --------    --------      -Pass-
RAM Test ------------   --------    ------   --------    --------      -Pass-
FIFO Test -----------   --------    ------   --------    --------      -Pass-
Load / Looping ------   --------    ------   --------    --------      -Pass-
Optional HW ---------   --------    ------   --------    --------      -Pass-
External Triggers ---   --------    ------   --------    --------      -Pass-
Long Acquisitions ---     25        loops       25       --------      -Pass-
Sample Dropping -----   --------    ------   --------    --------      -Pass-
```

AnalogInput~Diagnostic

```
                        Analog Input Diagnostic
        Test             measmt     units       min         max        result
DVM -----------------   --------    ------   --------    --------      -Pass-
DC Paths ------------   --------    ------   --------    --------      -Pass-
Gain Control --------   --------    ------   --------    --------      -Pass-
Cal DAC -------------   --------    ------   --------    --------      -Pass-
Offset Control ------   --------    ------   --------    --------      -Pass-
Input Selection -----   --------    ------   --------    --------      -Pass-
A Bias Ctl ----------   --------    ------   --------    --------      -Pass-
B Bias Ctl ----------   --------    ------   --------    --------      -Pass-
C Bias Ctl ----------   --------    ------   --------    --------      -Pass-
A Clamp -------------   --------    ------   --------    --------      -Pass-
B Clamp -------------   --------    ------   --------    --------      -Pass-
C Clamp -------------   --------    ------   --------    --------      -Pass-
```

AudioAnalog~Diagnostic

```
                        Audio Analog Diagnostic
        Test             measmt     units       min         max        result
DAC Calibration -----   --------    ------   --------    --------      -Pass-
ADC Calibration -----   --------    ------   --------    --------      -Pass-
ADC BitPatterns -----   --------    ------   --------    --------      -Pass-
Cal-Signal Paths ----   --------    ------   --------    --------      -Pass-
Attenuator Steps ----   --------    ------   --------    --------      -Pass-
Gain Steps ----------   --------    ------   --------    --------      -Pass-
Notch Filters -------   --------    ------   --------    --------      -Pass-
Left Flatness -------      0.101    dB pp    --------       0.300      -Pass-
```

AudioProcessor~Diagnostic

```
                         Audio Processor Diagnostic
        Test              measmt     units       min         max       result
DSP Program RAM -----    --------   ------   --------    --------    -Pass-
DSP X Data RAM ------    --------   ------   --------    --------    -Pass-
DSP Y Data RAM ------    --------   ------   --------    --------    -Pass-
Sign Extend Buf -----    --------   ------   --------    --------    -Pass-
Shared Bus Timing ---    --------   ------   --------    --------    -Pass-
DSP Host Port -------    --------   ------   --------    --------    -Pass-
Audio Data Reg ------    --------   ------   --------    --------    --NA--
Interrupt Reg -------    --------   ------   --------    --------    -Pass-
Sample Dropping -----    --------   ------   --------    --------    --NA--
```

Controller~Diagnostic

```
                           Controller Diagnostic
        Test              measmt     units       min         max       result
Identify Board ------    --------   ------   --------    --------    -Pass-
Control Registers ---    --------   ------   --------    --------    -Pass-
Overrange Detector --    --------   ------   --------    --------    -Pass-
Clock Detector ------    --------   ------   --------    --------    -Pass-
A-Clamp  Counter ----    --------   ------   --------    --------    -Pass-
B-Clamp  Counter ----    --------   ------   --------    --------    -Pass-
C-Clamp  Counter ----    --------   ------   --------    --------    -Pass-
Acq Sig0 Counter ----    --------   ------   --------    --------    -Pass-
Acq Sig1 Counter ----    --------   ------   --------    --------    -Pass-
Acq Sig2 Counter ----    --------   ------   --------    --------    -Pass-
Register File -------    --------   ------   --------    --------    --NA--
SM Settings RAM -----    --------   ------   --------    --------    -Pass-
SM Trigger  RAM -----    --------   ------   --------    --------    -Pass-
SM Sequence RAM -----    --------   ------   --------    --------    -Pass-
```

FilterBoard~Diagnostic

```
                           FilterBoard Diagnostic
        Test              measmt     units       min         max       result
Times 8 Gain --------      7.99     times       7.20        8.80      -Pass-
Filter slot  0 -------   --------   ------   --------    --------    -Pass-
Filter slot  1 -------   --------   ------   --------    --------    -FAIL-
  Filter ID ---------    --------   ------   --------    --------    -FAIL-
  Freq response -----    --------   ------   --------    --------    --
Filter slot  2 -------   --------   ------   --------    --------    -Pass-
Filter slot  3 -------   --------   ------   --------    --------    -FAIL-
  Filter ID ---------    --------   ------   --------    --------    -FAIL-
  Freq response -----    --------   ------   --------    --------    --
Filter slot  4 -------   --------   ------   --------    --------    -Pass-
Filter slot  5 -------   --------   ------   --------    --------    -Pass-
```

Genlock~Diagnostic

```
                             Genlock Diagnostic
        Test              measmt     units       min         max       result
Status checks -------    --------   ------   --------    --------    -Pass-
NTSC VCO Lock -------    --------   ------   --------    --------    -Pass-
PAL VCO Lock --------    --------   ------   --------    --------    -Pass-
```

Measure~Temperature

```
Measure Temperature Results:
   Temperature:   30.8 C
```

OPTION 1G (ECHO/ROUNDING) RESULTS FILES

Echo (NTSC & PAL)

```
 Echo Application Results: Wed Jan 30 16:33:33
Channel A

-------------------------------------------------------------------
Cursor   Time      percent of Peak      dB rel Curve 1
  1     406 nsec        0.14              -36.96
  2     605 nsec        0.00              -129.50
```

Rounding~Errors (NTSC & PAL)

```
  Measurement Results           Channel A          Wed Jan 30 16:33:59

Rounding Errors                                    Waveform->Pulse
Line =  96
Average Off
-------------------------------------------------------------------
(From  0.40 uS to  1.00 uS)
Rounding of White          0.3  %
Rounding of Black          0.0  %
-------------------------------------------------------------------
```

OPTION 20 (TELETEXT) RESULTS FILES

SoundInSync (PAL Only)

```
  Measurement Results         Channel A         Wed Jan 30 16:37:29

SoundInSync
Line =  96 (SIS mode)
Accumulation 200 times
----------------------------------------------------------------------
    (Eye Threshold = 1/250)
Eye Height                 86.2  %
                          201.0  mV
    (At Clock)
Eye Width                  77.5  %
    (At Middle)
'3' Level                 370.8  mV
'2' Level                 147.1  mV
'1' Level                 -86.1  mV
'0' Level                -310.9  mV
P-P Amplitude             106.0  %
                          722.5  mV
----------------------------------------------------------------------
```

Teletext (NTSC & PAL)

```
  Measurement Results         Channel A         Wed Jan 30 16:19:59

Teletext                                        Waveform->Teletext
Field = 1 Line =  15
Accumulation 400 times
Timing Average Off
----------------------------------------------------------------------
    (Eye Threshold = 1/250)
Eye Height                 45.9  %          *       70.0      100.0
                           28.1  IRE        *       49.0       70.0
    (At Clock)
Eye Width                  58.1  %          *       70.0      100.0
    (At Middle)
'1' Level                  66.0  IRE        *       67.5       72.5
'0' Level                   4.7  IRE        *       -2.5        2.5
P-P Amplitude             156.8  %          *      100.0      130.0
                           96.1  IRE        *       70.0       91.0
Run-In Start               9.91  u sec
Run-In Bits                16.0  bits
Data Line Width           50.32  u sec      *      52.00      59.00
Data End to Sync           3.32  u sec      *       1.00       3.00
Run-In Amplitude           52.7  IRE        *       67.5       72.5
----------------------------------------------------------------------
```

OPTION 21 (CAMERA TESTING) RESULTS FILES

NOTE: these files show the types of results obtainable from the measurements. The results shown were not obtained using camera signals.

Detail

```
  Measurement Results    Channel A              Fri Aug 21 08:19:06

Detail

Camera: cam1
Field = 1 Line = 22 (Synchronous)
Average Off
----------------------------------------------------------------------
Detail (H Rising Edge)   >>>>>>   %
Detail (H Falling Edge)  >>>>>>   %
Detail (V Rising Edge)   ------   %
Detail (V Falling Edge)  ------   %
----------------------------------------------------------------------
Field Rate Display Position  36.0 u sec
```

Gamma

```
  Measurement Results    Channels A, B, and C   Fri Aug 21 08:27:08

Gamma

Camera: cam1
----------------------------------------------------------------------
Gamma (A)                ------
Gamma (B)                ------
Gamma (C)                ------
----------------------------------------------------------------------
             (A)        (B)        (C)
step #1      172.3      171.5      171.0  mV
step #2      172.8      172.0      171.2  mV
step #3      172.8      172.8      172.2  mV
step #4      172.4      172.7      171.8  mV
step #5       71.7       72.0       71.6  mV
step #6      554.6      549.2      545.2  mV
step #7      643.1      641.1      639.4  mV
step #8      544.8      542.5      541.7  mV
step #9      544.9      542.5      541.8  mV
step #10     544.0      542.8      542.1  mV
step #11     544.0      542.5      541.8  mV
```

Geometry~Registration

```
Measurement Results      Channels A, B, and C    Fri Aug 21 08:21:06

Geometry/Registration

Camera: cam1
Porta-Pattern Chart
Averaged 0 Frames
----------------------------------------------------------------------
Geometry (Ch A): Results Unavailable

Registration (Ch B): Results Unavailable

Registration (Ch C): Results Unavailable
```

Shading

```
Measurement Results      Channel A               Fri Aug 21 08:20:21

Shading

Camera: cam1

Acquired 1 Frame
----------------------------------------------------------------------
Ave. Video Level:       10.0 IRE

V. Shading:              0.5 %
H Shading (Averaged):   82.0 %

H. Shading @ F1 L 22:   81.8 %
Max. H. Sh @ F1 L 23:   82.2 %
----------------------------------------------------------------------
Measurement Widthg    1.0 uSec
Measurement Position:
Center  35.0 uSec from sync
```

Vertical~Smear

```
Measurement Results      Channel A                Fri Aug 21 08:19:46

Vertical Smear

Camera: cam1

Reference White Level    98 IRE  at f Stop Undefined
Smear Threshold Level     1 IRE  at f Stop Undefined
    Above aperture smear   -21.1 IRE
    Below aperture smear   -20.9 IRE

Eq. :: 20 * log(Ref White / Threshold) + 20 * log((Ref Iris / Smr Iris) ^ 2)
----------------------------------------------------------------------
Smear: ----- dB
----------------------------------------------------------------------
Measurement Width    1.0 uSec
Measurement Positions:
Aperture  Center  35.0 uSec from sync
Reference Center  20.0 uSec from sync

Reference White Level measured on F1 Line 102
Above aperture smear  measured on F1 Line   9
Below aperture smear  measured on F1 Line 189
```

Colorimetry

```
  Measurement Results    Channels A, B, and C    Wed Mar 04 11:51:31

Colorimetry

Camera ID: Camera 1
Macbeth Chart, 24 Color Chips
3100K (Studio)
SMPTE Phosphors
Nominal Gamma
Default Ref.
----------------------------------------------------------------------
                           DE      DH
Dk. Skin             =   46.0,    2.0
Lt. Skin             =   75.4,    1.4
Blue Sky             =   53.0,   16.9
Foliage              =   42.3,   13.0
Bl. Flower           =   54.5,   11.1
Bl. Green            =   82.4,   23.2
Orange               =   99.9,    5.3
Prpl. Blue           =   53.0,   19.2
Mod. Red             =  106.9,    0.4
Purple               =   35.9,    5.8
Yw. Green            =   77.1,   19.6
Or. Yellow           =   91.3,    6.1
Blue                 =   52.0,   17.5
Green                =   75.2,   23.9
Red                  =  122.0,    1.0
Yellow               =   92.3,    9.1
Magenta              =   98.1,    2.6
Cyan                 =   75.0,   25.3
```

```
White               =  92.1,   2.9
Neutrl 8            =  77.0,   1.6
Neutrl 6.5          =  61.6,   2.0
Neutrl 5            =  46.4,   2.2
Neutrl 3.5          =  30.1,   2.5
Black               =  15.1,   0.1
Average Weighted DE =  69.0
Average Weighted DH =   8.9
----------------------------------------------------------------------
Measurement Width: 8 cycles ( 2.2 uSec)
Measurement Positions:
Packet # 1, F1 L 50, Center 13.0 uSec from sync
Packet # 2, F1 L 50, Center 22.0 uSec from sync
Packet # 3, F1 L 50, Center 30.9 uSec from sync
Packet # 4, F1 L 50, Center 39.5 uSec from sync
Packet # 5, F1 L 50, Center 48.0 uSec from sync
Packet # 6, F1 L 50, Center 56.7 uSec from sync
Packet # 7, F1 L105, Center 13.0 uSec from sync
Packet # 8, F1 L105, Center 22.0 uSec from sync
Packet # 9, F1 L105, Center 30.9 uSec from sync
Packet #10, F1 L105, Center 39.5 uSec from sync
Packet #11, F1 L105, Center 48.0 uSec from sync
Packet #12, F1 L105, Center 56.7 uSec from sync
Packet #13, F1 L160, Center 13.0 uSec from sync
Packet #14, F1 L160, Center 22.0 uSec from sync
Packet #15, F1 L160, Center 30.9 uSec from sync
Packet #16, F1 L160, Center 39.5 uSec from sync
Packet #17, F1 L160, Center 48.0 uSec from sync
Packet #18, F1 L160, Center 56.7 uSec from sync
Packet #19, F1 L220, Center 13.0 uSec from sync
Packet #20, F1 L220, Center 22.0 uSec from sync
Packet #21, F1 L220, Center 30.9 uSec from sync
Packet #22, F1 L220, Center 39.5 uSec from sync
Packet #23, F1 L220, Center 48.0 uSec from sync
Packet #24, F1 L220, Center 56.7 uSec from sync
```

Defects

```
    Measurement Results          Channel A          Thu Feb 27 17:43:05

CCD Defects

Camera: Camera 1
Room Temperature:  20.0 Celsius,     68 Fahrenheit

CCD H. Density: 728 Pixels
Threshold: 250 mV
Averaged 32 Frames

Largest Defect: 23 Pixels
Max Dev. Found: 404.9 mV
91 Bad Pixels Found

---------------------------------------------------------------------
There is 1 Defect With 23 Bad Pixel(s).
There are 2 Defects With 18 Bad Pixel(s) Each.
There is 1 Defect With 14 Bad Pixel(s).
There is 1 Defect With 5 Bad Pixel(s).
There is 1 Defect With 3 Bad Pixel(s).
There are 2 Defects With 2 Bad Pixel(s) Each.
There are 6 Defects With 1 Bad Pixel(s) Each.

Field 1, Line 22, 21.92 uSec
Field 2, Line 22, 21.92 uSec
Field 1, Line 23, 21.92 uSec
Field 2, Line 23, 21.92 uSec
Field 1, Line 24, 21.92 uSec
Field 2, Line 24, 21.85 uSec
Field 1, Line 25, 21.85 uSec
Field 2, Line 25, 21.85 uSec, 22.06 uSec
Field 2, Line 244, 45.13 uSec
Field 1, Line 245, 45.13 uSec
Field 2, Line 245, 42.59 uSec, 45.13 uSec
Field 1, Line 247, 39.85 uSec, 57.57 uSec
Field 2, Line 247, 34.21 uSec
Field 1, Line 248, 34.21 uSec
Field 2, Line 250, 27.07 uSec, 49.05 uSec
Field 1, Line 251, 27.07 uSec, 49.05 uSec
Field 2, Line 251, 49.05 uSec
Field 1, Line 252, 49.12 uSec
Field 2, Line 252, 49.12 uSec
Field 1, Line 253, 49.12 uSec
Field 2, Line 255, 23.02 uSec, 49.19 uSec
Field 1, Line 256, 23.02 uSec, 49.12 uSec
Field 2, Line 256, 23.02 uSec, 49.12 uSec
Field 1, Line 257, 23.09 uSec, 49.12 uSec
Field 2, Line 257, 23.09 uSec, 49.05 uSec
Field 1, Line 258, 23.09 uSec, 49.05 uSec
Field 2, Line 258, 23.16 uSec, 49.05 uSec
Field 1, Line 259, 49.05 uSec
Field 1, Line 260, 16.91 uSec
Field 2, Line 261, 23.16 uSec
Field 1, Line 262, 23.16 uSec
Field 2, Line 262, 23.09 uSec
---------------------------------------------------------------------
```

Fixed_Pattern~Noise

```
  Recorded Measurement Results         Channel A         Thu Feb 27 17:04:29

Fixed Pattern Noise

Camera: Camera 1
100kHz ~ 4.2MHz
----------------------------------------------------------------------
Fixed Pattern Noise       -33.1  dB
----------------------------------------------------------------------
Shading Removed
```

Frequency~Response

```
  Measurement Results    Channels A, B, and C    Thu Feb 27 16:07:21

Frequency Response, Depth of Modulation

Camera: Camera 1

Field = 2 Line = 120 (Synchronous)
Average  20

----------------------------------------------------------------------
G (A) Reference Packet # 1  319.3  mV
Packet # 1,                   0.0  dB
Packet # 2,                  -3.1  dB        *        -3.0       0.0
Packet # 3,                 -11.4  dB        *        -5.0       0.0
Packet # 4,                 -26.0  dB
Packet # 5,                 -29.2  dB
Packet # 6,                 -29.4  dB
Packet # 7,                 -29.8  dB
Packet # 8,                 -28.3  dB
Packet # 9,                 -28.0  dB
Packet #10,                 -16.3  dB        *        -5.0       0.0
Packet #11,                  -6.8  dB        *        -3.0       0.0
Packet #12,                  -1.8  dB        *        -1.0       0.0

B (B) Reference Packet # 1  290.1  mV
Packet # 1,                   0.0  dB
Packet # 2,                  -3.6  dB        *        -3.0       0.0
Packet # 3,                 -10.5  dB        *        -5.0       0.0
Packet # 4,                 -23.8  dB
Packet # 5,                 -25.7  dB
Packet # 6,                 -26.9  dB
Packet # 7,                 -25.3  dB
Packet # 8,                 -24.9  dB
Packet # 9,                 -26.2  dB
Packet #10,                 -16.5  dB        *        -5.0       0.0
Packet #11,                  -5.6  dB        *        -3.0       0.0
Packet #12,                  -1.8  dB        *        -1.0       0.0

R (C) Reference Packet # 1  332.2  mV
Packet # 1,                   0.0  dB
Packet # 2,                  -4.1  dB        *        -3.0       0.0
Packet # 3,                 -11.2  dB        *        -5.0       0.0
Packet # 4,                 -23.1  dB
Packet # 5,                 -28.5  dB
Packet # 6,                 -29.2  dB
Packet # 7,                 -30.1  dB
Packet # 8,                 -29.2  dB
Packet # 9,                 -25.5  dB
Packet #10,                 -15.1  dB        *        -5.0       0.0
Packet #11,                  -6.6  dB        *        -3.0       0.0
Packet #12,                  -1.8  dB        *        -1.0       0.0
----------------------------------------------------------------------
Measurement Width    2.0 uSec
Measurement Positions:
Packet # 1 Center   13.9 uSec from sync
Packet # 2 Center   17.5 uSec from sync
Packet # 3 Center   21.7 uSec from sync
Packet # 4 Center   25.8 uSec from sync
Packet # 5 Center   29.5 uSec from sync
Packet # 6 Center   33.5 uSec from sync
Packet # 7 Center   37.4 uSec from sync
Packet # 8 Center   41.3 uSec from sync
Packet # 9 Center   45.0 uSec from sync
Packet #10 Center   49.0 uSec from sync
Packet #11 Center   52.9 uSec from sync
Packet #12 Center   57.0 uSec from sync
```

Option 22 (Video Wander) Results File

```
  Measurement Results        Channel A        Thu Aug 29 17:43:05

NTSC Wander Measurement  Offset Zero Time 10 sec  LP Corner Frequency 1.00 Hz

------------------------------------------------------------------------------

Peak Drift Rate      -21.0 ppb/sec

Peak Frequency Offset    -445.9 ppb

------------------------------------------------------------------------------
```

OPTION 30 (COMPONENT) RESULTS FILES

The Component_Vector application produces no results file.

Bowtie

```
  Measurement Results    Channels A, B, and C    Wed Mar 06 16:26:37

Bowtie
Field = 1  Line =  45
Artificial Reference in use at: 35.50 uSec
Average Off
-----------------------------------------------------------------------
Relative Timing B-Y       -4.7 ns
Relative Timing R-Y       -4.8 ns
Relative Amplitude B-Y    -3.15 mV
Relative Amplitude R-Y    -2.03 mV
-----------------------------------------------------------------------
```

Component~Channel_Delay

```
  Measurement Results    Channels A, B, and C    Wed Mar 04 16:35:19

Component Channel Delay
Field = 1 Line =  50
Average  32
-----------------------------------------------------------------------
Pb to Y Delay Time         -2.7  n sec
Pr to Y Delay Time          6.0  n sec
Pb to Pr Delay Time        -8.8  n sec
-----------------------------------------------------------------------
```

Component-ColorBar

```
  Measurement Results    Channels A, B, and C    Wed Mar 04 17:35:19

Component Colorbar
Field = 1 Line =  50
Average  32
SMPTE/EBU,  75%
----------------------------------------------------------------------
Y  level  ( Gray   )       702.0  mV
Y  level  (Yellow )        466.9  mV
Y  level  ( Cyan   )       370.5  mV
Y  level  ( Green )        310.4  mV
Y  level  (Magenta)        217.5  mV
Y  level  (   Red   )      158.5  mV
Y  level  ( Blue   )        60.5  mV
Y  level  ( Black )          0.2  mV

Pb  level  ( Gray   )        0.1  mV
Pb  level  (Yellow )      -263.5  mV
Pb  level  ( Cyan   )       88.9  mV
Pb  level  ( Green )      -174.4  mV
Pb  level  (Magenta)       174.2  mV
Pb  level  (   Red   )     -89.1  mV
Pb  level  ( Blue   )      263.2  mV
Pb  level  ( Black )         0.0  mV

Pr  level  ( Gray   )        0.1  mV
Pr  level  (Yellow )        42.4  mV
Pr  level  ( Cyan   )     -262.3  mV
Pr  level  ( Green )      -220.0  mV
Pr  level  (Magenta)       219.5  mV
Pr  level  (   Red   )     262.1  mV
Pr  level  ( Blue   )      -42.5  mV
Pr  level  ( Black )         0.1  mV

----------------------------------------------------------------------
```

Component~K_Factor

```
  Measurement Results    Channels A, B, and C    Wed Mar 04 15:35:19

Component K Factor
Field = 1 Line =  50
Graticule   EIA
Average  32
-----------------------------------------------------------------------
K-2T (Y)                  0.8  % KF
K-5T (Pb)                 0.1  % KF
K-5T (Pr)                 0.1  % KF
HAD (Y)                 199.9  n sec
HAD (Pb)                499.6  n sec
HAD (Pr)                499.4  n sec
K-PB (Y)                 -0.0  % KF
K-PB (Pb)                -0.1  % KF
K-PB (Pr)                -0.1  % KF
-----------------------------------------------------------------------
Pulse position     22.9 (Y), 34.0 (Pb), 33.9 (Pr) u sec
Bar position       36.9 (Y), 25.1 (Pb), 25.1 (Pr) u sec
Ref. position      ---- (Y), ---- (Pb), ---- (Pr) u sec  (---- = PulseLobe)
Measurement Area    100 (Y),   100 (Pb),   100 (Pr) %
T = 100 n sec
```

Component~LevelMeter

```
  Measurement Results    Channels A, B, and C    Wed Mar 04 18:35:19

Component LevelMeter
Field = 1 Line =  50
Average  32
-----------------------------------------------------------------------
Sync                       301.5  mV        *        271.4      300.0
Y                          701.9  mV
Pb                         702.3  mV
Pr                         699.3  mV
-----------------------------------------------------------------------
Sync Tip        2.4 u sec
Back Porch      6.9 u sec
 Measurement Cycles( 4 Subcarrier Cycles)
Y Minus Pos     6.9 u sec
Y Plus Pos     11.5 u sec
 Measurement Cycles( 4 Subcarrier Cycles)
Pb Minus Pos   19.8 u sec
Pb Plus Pos    52.5 u sec
 Measurement Cycles( 4 Subcarrier Cycles)
Pr Minus Pos   26.1 u sec
Pr Plus Pos    46.0 u sec
 Measurement Cycles( 4 Subcarrier Cycles)
```

Component~Multiburst

```
  Measurement Results    Channels A, B, and C    Wed Mar 04 19:35:19

Component Multiburst
Field = 1 Line =  50
Average  32
-----------------------------------------------------------------------
Flag (Y)                  421.2  mV
Packet #1 0.50 MHz (Y)    -0.00  dB
Packet #2 1.00 MHz (Y)    -0.02  dB
Packet #3 2.00 MHz (Y)    -0.07  dB
Packet #4 3.00 MHz (Y)    -0.08  dB
Packet #5 4.00 MHz (Y)    -0.13  dB
Packet #6 5.00 MHz (Y)    -0.15  dB
Flag (Pb)                 421.3  mV
Packet #1 0.50 MHz (Pb)  -0.00  dB
Packet #2 1.00 MHz (Pb)  -0.01  dB
Packet #3 1.50 MHz (Pb)  -0.05  dB
Packet #4 2.00 MHz (Pb)  -0.03  dB
Packet #5 2.50 MHz (Pb)   0.01  dB
Flag (Pr)                 419.4  mV
Packet #1 0.50 MHz (Pr)  -0.01  dB
Packet #2 1.00 MHz (Pr)  -0.01  dB
Packet #3 1.50 MHz (Pr)  -0.05  dB
Packet #4 2.00 MHz (Pr)  -0.04  dB
Packet #5 2.50 MHz (Pr)  -0.00  dB
-----------------------------------------------------------------------
Special Positioning:
 Y  Flag Position    9.3 u sec
 Y  Flag Width       5.6 u sec
 Y  Packet #1  Center  10.1 u sec (from flag)
 Y  Packet #2  Center  17.5 u sec (from flag)
 Y  Packet #3  Center  25.1 u sec (from flag)
 Y  Packet #4  Center  32.8 u sec (from flag)
 Y  Packet #5  Center  40.2 u sec (from flag)
 Y  Packet #6  Center  48.0 u sec (from flag)

 Y  Packet #1  Width    3.5 u sec
 Y  Packet #2  Width    3.0 u sec
 Y  Packet #3  Width    2.2 u sec
 Y  Packet #4  Width    2.2 u sec
 Y  Packet #5  Width    2.0 u sec
 Y  Packet #6  Width    1.4 u sec

 Pb Flag Position  10.1 u sec
 Pb Flag Width       5.6 u sec
 Pb Packet #1  Center  11.5 u sec (from flag)
 Pb Packet #2  Center  20.3 u sec (from flag)
 Pb Packet #3  Center  29.1 u sec (from flag)
 Pb Packet #4  Center  37.9 u sec (from flag)
 Pb Packet #5  Center  46.7 u sec (from flag)

 Pb Packet #1  Width    3.5 u sec
 Pb Packet #2  Width    2.5 u sec
 Pb Packet #3  Width    2.6 u sec
 Pb Packet #4  Width    2.2 u sec
 Pb Packet #5  Width    2.0 u sec

 Pr Flag Position  10.1 u sec
 Pr Flag Width      5.6 u sec
 Pr Packet #1  Center  11.5 u sec (from flag)
 Pr Packet #2  Center  20.3 u sec (from flag)
 Pr Packet #3  Center  29.1 u sec (from flag)
 Pr Packet #4  Center  37.9 u sec (from flag)
 Pr Packet #5  Center  46.7 u sec (from flag)

 Pr Packet #1  Width    3.5 u sec
 Pr Packet #2  Width    2.5 u sec
 Pr Packet #3  Width    2.6 u sec
 Pr Packet #4  Width    2.2 u sec
 Pr Packet #5  Width    2.0 u sec
```

Component~Noise

```
  Measurement Results    Channels A, B, and C    Wed Mar 04 14:35:19

Component Noise
Line = 255
(Y)  Band width  10kHz to 5.0MHz
(Pb) Band width  10kHz to 2.5MHz
(Pr) Band width  10kHz to 2.5MHz
Average  32
----------------------------------------------------------------------
Noise Level (Y)          -77.5  dB rms
Noise Level (Pb)         -81.3  dB rms
Noise Level (Pr)         -83.1  dB rms
----------------------------------------------------------------------
```

Component~NonLinearity

```
  Measurement Results    Channels A, B, and C    Wed Mar 04 20:35:19

Component NonLinearity
Field = 1 Line =  50
Average  32
----------------------------------------------------------------------
LumaNonLinearity (Y)       0.0  %
LumaNonLinearity (Pb)      0.1  %
LumaNonLinearity (Pr)      0.1  %
----------------------------------------------------------------------
Positioning (Y):    5 Steps
 1st Luminance Step  19.6 u sec
 End Luminance Step  52.7 u sec
Positioning (Pb):    5 Steps
 1st Luminance Step  19.6 u sec
 End Luminance Step  52.7 u sec
Positioning (Pr):    5 Steps
 1st Luminance Step  19.6 u sec
 End Luminance Step  52.7 u sec
```

Lightning

```
  Measurement Results    Channels A, B, and C    Wed Mar 06 16:20:50

Lightning
Field = 1  Line =  86
Average Off
-----------------------------------------------------------------------
Reference
  Colorbars:  75% SMPTE/EBU (60Hz)
  Pk-white (100%) 700.0 mV  Setup 0.0%  Color Pk-to-Pk 525.0 mV

Measured
  Color Pk-to-Pk  B-Y   701.79 mV         33.67%
  Color Pk-to-Pk  R-Y   700.22 mV         33.37%
  Pk-white              713.47 mV (100%)   1.92%
  Delay  B-Y             -6 ns
  Delay  R-Y             -7 ns
-----------------------------------------------------------------------
```

OPTIONS 40 AND 41 (AUDIO) RESULTS FILES

The following Audio applications produce no results file: Calibrate~AudioBoard, Identify~Audio_Hardware.

Audio~Analyzer

```
 Audio Analyzer Measurement Results: Tue Jan 21 15:15:47
Input Number 1
----------------------------------------------------------------------
Left Channel:
Frequency =    0.0 Hz
Level = -140.00 dBu
THD+N =   0.000 %
Right Channel:
Frequency =    0.0 Hz
Level = -140.00 dBu
THD+N =   0.000 %
Level Difference (L - R):    0.000 dB
Phase Difference (L - R):    0.00 deg
Left Level:
----------
Frequency (Hz)      Level (dBu)         Lower Limit         Upper Limit
Right Level:
-----------
Frequency (Hz)      Level (dBu)         Lower Limit         Upper Limit
Level Difference:
----------------
Frequency (Hz)      Level Diff (dB)     Lower Limit         Upper Limit
Phase Difference:
----------------
Frequency (Hz)      Phase Diff (deg)    Lower Limit         Upper Limit
Left THD+N:
----------
Frequency (Hz)      THD+N (%)           Lower Limit         Upper Limit
Right THD+N:
-----------
Frequency (Hz)      THD+N (%)           Lower Limit         Upper Limit
Channel Separation L -> R:
-------------------------
Frequency (Hz)      Separation (dB)     Lower Limit         Upper Limit
Channel Separation R -> L:
-------------------------
Frequency (Hz)      Separation (dB)     Lower Limit         Upper Limit
----------------------------------------------------------------------
```

Audio~Monitor

```
 Audio Monitor Results: Tue Jan 21 15:15:53
 Left Meter Level  = >>>> dBu
 Right Meter Level = >>>> dBu
 Sum Meter Level   = >>>> dBu
```

Audio-Spectrum

```
 Audio Spectrum Measurement Results: Tue Jan 21 15:15:59
Input Number 1
----------------------------------------------------------------------
View: SPECTRUM
High Resolution Mode: OFF
Active Channel: LEFT
Range:  10 dBu
Weighting Filter: None
Level (188 Hz - 20 kHz): >>>>>>> dBu
Average Off
----------------------------------------------------------------------
```

Multitone

```
Multitone Measurement Results: Tue Jan 21 15:16:08
Input Number 1
----------------------------------------------------------------------
Analyzed signal: MTone1
Left Level:
----------
Ref:    0.00 dBu at      0 Hz
Frequency (Hz)      Level (dB)
Right Level:
----------
Ref:    0.00 dBu at      0 Hz
Frequency (Hz)      Level (dB)
Level Difference:
----------------
Frequency (Hz)      Level Diff (dB)
Phase Difference:
----------------
Frequency (Hz)      Phase Diff (deg)
----------------------------------------------------------------------
```

View_Audio~Auto_Test

```
  Measurement Results                              Tue Jan 21 15:16:12
View Audio Auto Test
---------------------------------------------------------------------
At           Thu Dec 19 13:14:53 1991   Video Source: A   Audio Input: 1
Test Type    0.33 Program 01   Expected TEST level: 0 dBu
Source       TEK1
                                                               Violated Limits
                                    Left        Right           Lower    Upper
Insertion Gain Error (dB)          83.39       86.90     **    -0.50       0.50
Sweep Max. Gain (dB)            <-120.00    <-120.00     **   Limits Exceeded
Sweep Min. Gain (dB)            <-120.00    <-120.00     **   Limits Exceeded
Stereo Channel Assignment         Normal      Normal
THD+N (at 1020Hz) (%)               0.017       0.020
2nd Harmonic (at 1020Hz) (%)        0.002       0.001
THD+N (at 60Hz) (%)                 0.017       0.020
2nd Harmonic (at 60Hz) (%)          0.003       0.003
3rd Harmonic (at 60Hz) (%)          0.001       0.001
Crosstalk (into channel) (dB)     -93.95      -92.99
SNR (unweighted) (dB)              92.04       87.94
SNR (weighted) (dB)                88.15       87.48
Max. Compandor Error (rise) (dB)   -0.00       -0.00
Max. Compandor Error (fall) (dB)   -0.00        0.00

Gain Difference (dB)            <-120.00                 **   Limits Exceeded
Phase Difference (deg.)             0.02
----------------------------------------------------------------------
```

Appendix D
ERROR MESSAGES

This appendix lists the error messages returned by the VM700A, and their meanings.

Table D-1
Command Input Errors

Computer Mode	Terminal Mode	Computer Mode	Terminal Mode
?001	Bad command argument	?017	Remote not enabled
?002	Sub-function not found	?101	Request filtered
?003	Playback nesting too deep	?102	Screen event not key
?004	Function directory inaccessible	?103	Unknown softkey
?005	Function not found	?104	Invalid softkey
?006	Unknown command	?105	Unwanted hardkey
?007	Unknown hardkey	?106	Unknown input
?008	Out of memory	?107	Not found
?009	Recursive function call	?108	Request not supported
?010	Bad command in this context	?109	No server resources
?011	Name too long	?110	Illegal name
?012	No filename	?111	Not writable
?013	Line too long	?112	Not readable
?014	Command only meaningful for non-IP connections	?113	No permission
?015	Bad time format (use getclock)	?114	Bad argument(s)
?016	Function playback in progress. Enter ^C to stop it.		

Table D-2
Messages

Computer Mode	Terminal Mode	Computer Mode	Terminal Mode
[a]	Function playback in progress. Enter ^C to stop it.	!006	Hit CR to continue.
[a]	Function playback continues...	!007	Remote terminated.
[a]	Function playback terminated.	!008	Remote has been terminated locally.
[a]	Function playback completed.	@*name* [c]	Results in file *name* [c]
!005	[b]	!010	*application-specific message*

[a] This message has no computer mode equivalent.

[b] This message indicates that a function is displaying text on the screen or else an error has occurred in function playback. This message has no terminal mode equivalent.

[c] *Name* is the file where measurement results are stored after a **getresults** command.

NOTE

In computer mode, a @ is returned after successful command completion; a ! precedes a message; and a ? precedes an error.

INDEX