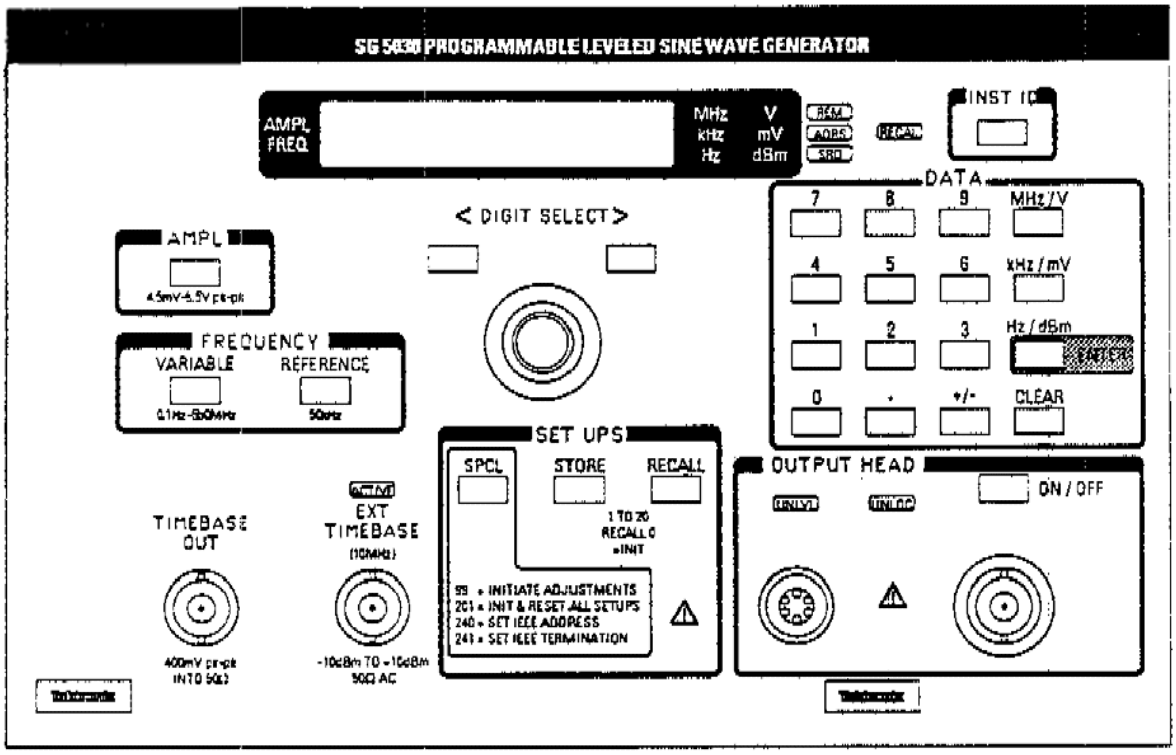
SG 5030 Instrument Interfacing Guide



This interfacing guide is designed to help you get started, as quickly as possible, using the SG 5030 Programmable Leveled Sine Wave Generator with a GPIB controller. This guide tells you how to program IBM PC AT compatible controllers, with GURU II installed, to communicate with the SG 5030. Program samples are included.

This guide does not take the place of the operators manual or other documentation supplied with the SG 5030 and your controller. More complete information in these other documents will assist you in gaining the full benefit from the programmable and manual capabilities of your sine-wave generator.

Setting Up the SG 5030 for GPIB Operation

Connecting the Controller

Connect a GPIB cable between the sine wave generator and the controller (all TM 5000 programmable instruments connect to the GPIB through a TM 5000 Power Module). Refer to the Operating Instructions section in the Instruction manual for information on installing the SG 5030 in the power module. Also review this section for instrument caution and warning statements and to become familiar with front-panel instrument functions. The program examples in this guide assume that the SG 5030 and controller are the only instruments on the bus.

For additional copies, order
070-7704-00
First Printing JUL 1990

Displaying the GPIB Address and Terminator

The SG 5030 primary address and the currently selected message terminator are both displayed in the front-panel readout when you press the INST ID button (e.g., 25 EOI, or 25 LFEOI). The message terminator switch can be set for either EOI, or LF with EOI. See your controller documentation for its requirements.

The SG 5030 is shipped from the factory set to an address of 25, and the message terminator is set to EOI-only.

Changing the GPIB Address and Terminator

To change the GPIB address:

Press SPCL button.
Enter 240 on DATA keypad.
Press ENTER button.

The current address is shown in the display window. Enter the new address on the DATA keypad and press the ENTER key. The SG 5030 primary GPIB address can be set to any decimal number in the range 0 to 31 (address 31 effectively disables the SG 5030 from communicating on the GPIB).

To change the message terminator:

Press SPCL button.
Enter 241 on DATA keypad.
Press ENTER button.

The current message terminator is shown in the display window. Rotate the front-panel knob to select either LFEOI or EOI only, then press the ENTER key.

EOI-only is recommended as the message terminator for use with Tektronix controllers. EOI-or-LFEOI is recommended for use with Hewlett-Packard controllers.

Programming the SG 5030

SG 5030 Power-On

SELF-TEST. When powered on, the SG 5030 microprocessor performs a diagnostic routine (self test) to check the functionality of the ROM and RAM. If no ROM or RAM error is found, the microprocessor goes on to check the functionality of the other instrument hardware.

If a functional failure is detected, an error code will appear in the display window. The error will remain displayed until cleared by pressing any front-panel button. Functional error codes (360 through 377) that occur during power up cannot be cleared through your controller. These errors can only be cleared by pressing a front-panel button. Refer to Table 2 for a brief description of the error codes displayed in the SG 5030 readout window.

Upon successful completion of the self-tests, the SG 5030 restores the instrument settings that were in use when the instrument was last turned off, except that the sine-wave signal to the output head will be turned off.

SG 5030 Command Messages

Commands are provided to control SG 5030 settings, cause actions, or request information. These commands are listed in Table 1. SG 5030 commands begin with a header – a word or abbreviation that describes the function implemented. SG 5030 commands can be combined in a message by separating the commands with the message unit delimiter (semicolon). Either upper or lower-case ASCII characters are accepted (refer to the ASCII Code Chart in Fig. 2).

Some headers are listed in two forms, a full-length version and an abbreviated version. The SG 5030 accepts any header containing at least the characters listed in the abbreviated form (all upper case). Any characters added to the abbreviated version must be those given in the full-length version.

A vertical line is used in a series of two or more units, to separate the units (e.g., <ON | OFF>); any one unit must be selected and sent as part of the command message. Do not include the line in the message.

Square brackets enclose an optional message part (e.g., <num> [:dBm]). Do not include the square brackets in the message.

SPCL Function Codes

SPCL (special) function codes are selected from the front panel of the SG 5030. To invoke a SPCL function, press the SPCL button and enter a two or three-digit SPCL code (shown below) on the DATA keypad, and press ENTER. Press the CLEAR button to exit the SPCL function mode and change the display.

SPCL Code	Description
99	Selects adjustment mode (internal switch toggles must be set to the Adjustment Mode position). This special function is intended for service personnel only.
201	Sets all twenty stored front-panel setups to the initialized default settings.
240	Enables the setting of the GPIB address. You can then press the INST ID button to display the current GPIB address.
241	Enables the setting of the message terminator. You can then press the INST ID button to display the current message terminator.

Table 1
SG 5030 PROGRAMMING COMMANDS

Header	Argument	Description
ABStouch	<n/1>	Causes one or more front-panel buttons or controls to be remotely activated (touched) through the GPIB interface. This feature is useful for evaluation and applications procedures that simulate operator interaction with the front panel. **Switch Name** — **Decimal Number** Knob Increment — 0 Knob Decrement — 1 Output ON — 2 CLEAR — 3 0 — 4 . — 5 +/- — 6 Hz/dBm/ENTER — 7 1 — 8 2 — 9 3 — 10 kHz/mV — 11 4 — 12 5 — 13 6 — 14 MHz/V — 15 7 — 16 8 — 17 9 — 18 INST ID — 19 AMPLITUDE — 20 VARIABLE — 21 STORE — 22 SPCL — 23 < — 24 > — 25 REF — 26 RECALL — 27
AMPlitude	<num> [:dBm]	Sets the output signal amplitude in volts peak-to-peak or dBm. If dBm units are not specified, volts are assumed. The dBm range is −42.95 dBm to +18.75 dBm, with a resolution of 0.05 dBm. The voltage peak-to-peak range of the instrument is 4.50 mV to 5.500 V. This range is divided into three subranges, and the amplitude resolution is different in each of these subranges as follows: **Range** — **Resolution** 4.50 mV to 55.00 mV — 0.02 mV 55.2 mV to 550.0 mV — 0.2 mV 0.552 V to 5.500 v — 2 mV −42.95 dBm to +18.75 dBm — 0.05 dBm If the amplitude specified in the command has a greater resolution than the subrange it falls in, the amplitude is rounded off to the closest allowable resolution.
AMPlitude?		Returns the amplitude setting.

Table 1 (cont)

Header	Argument	Description
ERRor? or EVEnt?		Returns information about the event reported in the last serial poll. If RQS is ON, the <num> response for both commands is the event code for the last reported status byte. If RQS is off, <num> is the event code for the highest priority event that has occurred.
EXTtb?		Returns the current status (ACTIVE or INACTIVE) of the external reference input (EXT TIMEBASE connector).
FREquency	<num>	Sets the frequency of the output signal. The frequency range of the instrument is 0.1 Hz to 550.00000 MHz. This range is divided into three subranges, and the frequency resolution is different in each of these subranges as follows: **Range** — **Resolution** 0.1 Hz to 4.9999 kHz — 0.1 Hz 5.000 kHz to 49.999 kHz — 1 Hz 50.00 kHz to 500.00000 MHz — 10 Hz If the frequency specified in the command has a greater resolution than the subrange it falls in, the frequency is rounded off to the nearest allowable resolution.
FREquency?		Returns the frequency setting.
HELp?		Returns the word HELP followed by a list of all the command headers the SG 5030 accepts.
ID?		Returns the instrument identification and the firmware version as firmware version as follows: TEK/SG 5030—identifies the instrument type. V81.1—identifies the version of the Tektronix Codes and Formats used in the SG 5030 firmware design. Fx.x—identifies the instrument's firmware version. For example, F1.0 indicates the firmware version 1.0.
INIt		Initializes the SG 5030 to the following predefined instrument and GPIB settings: OUTPUT — OFF AMPLITUDE — 1.000 FREQUENCY — 10.00000E+6 REFREQ — OFF RQS — ON USEREQ — OFF After the SG 5030 has executed the INIt command, the display shows FREQ 10.00000 MHz, and the switch button VARIABLE is lighted. (No other switches are lighted.)
LEVeled?		Returns the leveled-signal status of the output signal (YES or NO).
OUTput	<ON|OFF>	Toggles the signal output on or off at the output head connectors, depending on the argument specified. The source impedance is maintained when the output is off. At power-up, the output is set to off.
OUTput?		Returns the current status of the signal output frequency (ON or OFF).

Table 1 (cont)

Header	Argument	Description
RECall	<num>	Recalls the group of instrument settings from the nonvolatile RAM location specified in the <num> argument, and sets the instrument to those settings. The <num> argument can be from 0 to 20 (decimal). Numbers 1 through 20 are used for user-selectable setups; number 0 references the initialization setup (that is, the RECall 0 command is equivalent to the INIt command). If no settings are stored in the specified location, the instrument assumes the INIt command settings.
REFreq	<ON|OFF>	Toggles the 50 kHz reference frequency on and off. When the reference frequency is on, the variable frequency controls are disabled and the reference frequency is then sent to the output. When the reference frequency is toggled off, the variable frequency controls are enabled and the output frequency returns to the previously set variable frequency.
REFreq?		Returns the current status of the reference frequency (ON or OFF).
RQS	ON	Enables the instruments ability to generate service requests (SRQs). When RQS (request for service) is ON, the instrument asserts an SRQ on the GPIB whenever an event occurs that requires a service request. The events that normally cause service requests to be asserted include the power-up sequence, self-test errors, front-panel operation errors, and programming errors.
	OFF	SRQs are saved in a stack in the instrument and the SRQ annunciator light on the instrument front-panel is lit. An ERRor? or EVEnt? command query can then be used to determine which SRQs have been generated.
RQS?		Returns the current status of the SRQ function (ON or OFF).
SET?		Returns the current instrument setting for OUTPUT (ON or OFF), AMPLITUDE, FREQUENCY, REFREQ (ON or OFF), RQS (ON or OFF), and USEREQ (ON or OFF). The maximum length of the settings query data string is 84 bytes.
STOre	<num>	Stores the current instrument front-panel control settings in NVRAM at a location specified with the <num> argument. The <num> argument can be from 1 to 20 (decimal). Twenty front-panel control settings are saved for each instrument setup. The NVRAMs containing an internal power source to save the settings when the external instrument is removed.
TESt		Initiates a self test on the instrument. When the test is complete, the instrument returns to the settings last entered. If the self test failed, the SG 5030 returns an SRQ to the controller.
USEreq	<ON|OFF>	Enables or disables the SRQ interrupt that is generated when the front-panel INST ID button is pressed. The OFF state inhibits the instrument from returning the instrument ID SRQ when the INST ID button is pressed.
USEreq?		Returns the current status of the instrument ID SRQ (ON or OFF).

Sending Messages to the SG 5030

Controllers. A Tektronix 4041, or an IBM PC AT class compatible GPIB controller is needed to send messages to the SG 5030.

IBM Personal Computers (PC) and some PC compatible computers can be used as GPIB controllers by installing the Tektronix GPIB User's Resource Utility (GURU II) software/hardware package.

For the IBM programming examples the GURU II software must be initialized. For an example of the exact program lines to use for the GURU initialization, see the Quick Basic Sample Utility Program later in this guide. Refer to your GURU II documentation for more specific information.

```
IBM PC:
200 WRT$="FREQ 6.7E+6,OUT ON"
210 CALL IBWRT (BD%,WRT$)

4041:
180 PRINT #25:"FREQ 6.7E+6;OUT ON"
```

A useful variation assigns the SG 5030 address to a variable and inserts that variable in the print statement in place of the number for the address. This works with either the IBM or 4041 and allows you to change the program to work with the SG 5030 set to other addresses by changing only the statement that assigns the variable.

```
IBM PC:
200 DEVICE$="TEKDEV1"
210 CALL IBFIND (DEVICE$,BD$)
220 V%=25
230 CALL IBPAD (BD%,V%)
240 WRT$="FREQ 6.7E+6,OUT ON"
250 CALL IBWRT (BD%,WRT$)

4041:
200 SG=25
210 PRINT #SG:"FREQ 6.7E+6;OUT ON"
```

The SG 5030 command message inside the quote marks is the same for all of the above examples. The rest of each example varies to match the statement syntax designed into each controller as shown in Table 2. This suggests that once you understand your controller's input/output statements, it's just a matter of plugging in the SG 5030 commands you need.

Table 2
Controller Print/Output Statements

CONTROLLER	STATEMENT
IBM BASIC	WRT$=RQS CALL IBWRT (BD%,WRT$)
4041 BASIC	PRINT #25:"RQS ON"
HP-85 BASIC	OUTPUT 725;"RQS ON"
FLUKE 1720A BASIC	PRINT @25%,"RQS ON"
HP 9826 BASIC	OUTPUT 725,"RQS ON"

A message to a GPIB device is contained within the controller's GPIB statement. The statement is composed of three parts: the keyword, the address or logical unit number, and the device-dependent message. All the statements shown in the table send the same standard Tektronix Codes & Formats message (RQS ON) that enables SRQ interrupts. All send the message to an instrument with primary address 25. The difference lies in the syntax of the statement for a particular controller.

Getting SG 5030 Settings Information

SG 5030 queries include a ? as the last character of the command (such as SET?, FREq?, or ID?) and prepare the instrument for data output, but do not start such output. The SG 5030 waits until it sees it's talk address to begin sending the requested data. This is accomplished by the INPUT statement.

```
IBM PC:
290 WRT$="FREQ?"
300 CALL IBWRT (BD%,WRT$)
310 RD$=SPACE$ (100)
320 CALL IBRD (BD%,FREQUENC$)

4041:
290 INPUT #25 PROMPT "FREQ?":FREQUENC$
```

All instrument settings can be obtained in one message. Just dimension a string large enough (100 characters) and input the settings string.

```
IBM PC:
330 SETTIN$=SPACE$ (100)
340 WRT$=SET?"
350 CALL IBWRT (BD%,WRT$)
360 CALL IBRD (BD%,SETTINGS$)
370 PRINT SETTINGS$

4041:
330 DIM SETTINGS$ TO 100
340 INPUT #25 PROMPT "SET?":SETTINGS$
```

You can restore the settings you input from the SG 5030 by sending back the settings string.

```
IBM PC:
380 CALL IBWRT (BD%,SETTING$)

4041:
340 PRINT #25:SETTING$
```

Using SG 5030 Interrupts

Programmable interrupts are provided in the SG 5030 to inform the controller of asynchronous events such as command errors, syntax errors, or instrument events. The status bytes returned in response to an error query (ERR? or EVENT?) are shown in Table 3. The error query obtains more detail in the case of abnormal events. For instance, in the case of a command error, was it a problem with a header, argument, or delimiter? You can find out from the error code.

The error messages defined in Table 3 are returned as arguments to the ERRor? or EVEnt? query.

The Error Codes displayed in the SG 5030 readout window are listed in Table 4.

Table 3
BUS ERROR CODES

Description	Error/Event Query Response	Serial Poll Response[1] (decimal)
Command Errors/Events		
System normal	0	NSB[2]
Command header error	101	97 or 113
Header delimiter error	102	97 or 113
Command argument error	103	97 or 113
Argument delimiter error	104	97 or 113
Non-numeric argument	105	97 or 113
Missing argument	106	97 or 113
Invalid message unit delimiter	107	97 or 113
Bad symbol	150	97 or 113
Syntax error	151	97 or 113
Symbol number too long	153	97 or 113
Invalid input character	154	97 or 113
Invalid string input	155	97 or 113
Numerical underflow	156	97 or 113
Execution Errors/Events		
Argument out of range	205	98 or 114
Not in adjustment mode	250	98 or 114
I/O buffers full, output flushed	251	98 or 114
Settings buffer empty	252	98 or 114
Illegal settings number specified	253	98 or 114
Beyond adjustment limit	254	98 or 114

Table 3 (cont)

Description	Error/Event Query Response	Serial Poll Response[1] (decimal)
Internal Errors/Events		
Interrupt fault	301	99 or 115
System error	302	99 or 115
HF unleveled	350	99 or 115
Reference loop unlocked	351	99 or 115
Wide loop unlocked	352	99 or 115
Narrow loop unlocked	353	99 or 115
Offset loop unlocked	354	99 or 115
DDS loop unlocked	355	99 or 115
Unplugged error	356	99 or 115
CPU Self Test Errors		
EPROM checksum failure	360	99 or 115
NVRAM test failure	361	99 or 115
RAM test failure	362	99 or 115
NVRAM lost	363	99 or 115
CAL Constant Checksum failure	364	99 or 115
Hardware Self Test Errors		
OUTPUT OFF test failure	370	99 or 115
REF FREQ test failure	371	99 or 115
DDS OFF test failure	372	99 or 115
10.00000 MHz test failure	373	99 or 115
10.00001 MHz test failure	374	99 or 115
500.00000 MHz test failure	375	99 or 115
WIDE LOOP DIVIDER test failure	376	99 or 115
OUTPUT AMP powered test failure	377	99 or 115
System Events/Errors		
No errors or events	0	NSB[2]
Power on	401	65 or 81
ID User request	403	67 or 83
SRQ pending	455	NSB[2]

[1] If the message processor is busy, the instrument returns a number 16 (decimal) higher than the serial poll response.
[2] No status byte.

Table 4
FRONT PANEL ERROR CODES

Error	Description
E101	Command header error
E102	Header delimiter error
E103	Command argument error
E104	Argument delimiter error
E105	Non-numeric argument
E106	Missing argument
E107	Invalid message unit delimiter
E150	Bad symbol
E151	Syntax error
E153	Symbol number too long
E154	Invalid input character
E155	Invalid string input
E156	Numerical underflow
E205	Argument out of range
E250	Not in adjustment mode
E251	I/O buffers full, output flushed
E252	Settings buffer empty
E253	Illegal settings number specified
E254	Beyond calibration limit
E301	Interrupt fault
E302	System error
E350	HF unleveled
E351	Reference loop unlocked
E352	Wide loop unlocked
E353	Narrow loop unlocked
E354	Offset loop unlocked
E355	DDS loop unlocked
E356	Unplugged error
E360	EPROM checksum failure
E361	NVRAM test failure
E362	RAM test failure
E363	NVRAM lost
E364	CAL Constant Checksum failure
E370	OUTPUT OFF test failure
E371	REF FREQ test failure
E372	DDS OFF test failure
E373	10.00000 MHz test failure
E374	10.00001 MHz test failure
E375	500.00000 MHz test failure
E376	WIDE LOOP DIVIDER test failure
E377	OUTPUT AMP powered test failure

SRQ Handlers

Here are typical SRQ handlers that alert you to a reporting instrument's address, status, and error code with a message on your console. The error code is helpful during debugging because it identifies the specific command or execution problem should one occur.

IBM PC:

(Using National Instrument BASICA drivers for PC2 or PC2A GPIB card.)

```
1540 ' ***************************ERROR ROUTINE
     ******************************
1550 IF IBSTA% >= 0 AND BD% >= 0 AND IBSTA% <
     &H4000 AND IBERR% <> 6 THEN RETURN
1560 IF BD% < 0 THEN PRINT "device not installed -
     use IBCONF then reboot"
1570 IF IBSTA% > 0 AND IBSTA% >= &H4000 THEN PRINT
     "timeout"
1580 IF IBERR% = 6 THEN PRINT "timeout"
1590 PRINT "gpib error " ; IBERR%
1600 IF IBERR% = 0 THEN PRINT "DOS error device not
     installed
1610 IF IBERR% = 1 THEN PRINT "function requires
     GPIB-PC to be CIC
1620 IF IBERR% = 2 THEN PRINT "no listener on write
     function
1630 IF IBERR% = 3 THEN PRINT "GPIB-PC not
     addressed correctly
1640 IF IBERR% = 4 THEN PRINT "invalid argument to
     function call
1650 IF IBERR% = 5 THEN PRINT "GPIB-PC not system
     controller as required
1660 IF  IBERR%  =  6  THEN  PRINT  "I/O  operation
     aborted
1670 IF IBERR% = 7 THEN PRINT "non-existent GPIB-PC
     board
1680 IF IBERR% = 10 THEN PRINT "I/O started before
     previous operation completed
1690 IF IBERR% = 11 THEN PRINT "no capability for
     operation
1700 IF IBERR% = 12 THEN PRINT "file system error
1710 IF IBERR% = 14 THEN PRINT "command error
     during device call
1720 IF IBERR% = 15 THEN PRINT "serial poll status
     byte lost
1730 IF IBERR% = 16 THEN PRINT "SRQ stuck in on
     position
1740 INPUT "[ENTER] TO CONTINUE" ; A$
1750 RETURN
```

4041:

```
120 On srq then call dopoll
130 Enable srq
140 Main program
150 !
970 !
980 !
990 End
1000 Sub dopoll
1010 Poll stabyt addr
1020 Input #addr prompt "ERR?";errnum
1030 Print "STATUS=";stabyt,"ADDRESS=";addr,
     "ERROR=";errnum
1040 Resume
```

SG 5030 Response to Interface Messages

The following program sequences show various interface messages transmitted to the SG 5030.

IBM PC:

(Using National Instrument BASICA drivers for PC2 or PC2A GPIB card.)

```
20   ' SG5030 PRIMARY ADDRESS 25
130  '
140  BD$="GPIB0"
150  CALL IBFIND (BD$,BD%)
160  ID$="TEKDEV1"
170  CALL IBFIND (ID$,DEV%)
180  V%=25
190  CALL IBPAD (DEV%,V%) 'CHANGE ADDRESS TO 25
200  RETURN
210  CMD$=CHR$ (V%+32)     ' SEND LISTEN ADDRESS,
     MLA
220  CALL IBCMD (BD%,CMD$)
230  RETURN
240  CMD$=CHR$ (UNL)     ' SEND UNLISTEN, UNL
250  CALL IBCMD BD%,CMD$)
260  RETURN
270  CMD$=CHR$ (V%+64)  'SEND TALK ADDRESS
280  CALL IBCMD (BD%,CMD$)
290  RETURN
300  CMD$=CHR$ (UNT)     ' SEND UNTALK
310  CALL IBCMD (BD%,CMD$)
320  RETURN
330  ' SEND DEVICE CLEAR
340  CMD$=CHR$ (20)
350  CALL IBCMD (BD%,CMD$
360  RETURN
370  ' SEND MLA, SELECTED DEVICE CLEAR,UNL
380  CALL IBCLR (DEV%)
390  RETURN
400  LLO$=CHR$ (17)      'SEND LOCAL LOCKOUT
410  CALL IBCMD (BD%,LLO$)
420  RETURN
430  ' SEND MLA, GO TO LOCAL, UNL
440  CALL IBLOC (BD%)
450  RETURN
460  CMD$="? "           'SEND MLA, GROUP EXECUTABLE
     TRIGGER, UNL
470  CALL IBCMD (BD%,CMD$)
480  CMD$ (8) + "? "
490  CALL IBCMD (BD%,CMD$)
500  RETURN
```

4041:

```
40   Pri addr=25 ! *** SG 5030 Primary Address
     ***
150  !
160  Listen: wbyte atn(pri_addr+32) ! Send Listen
     Address MLA
170  Return
180  Unlisten: wbyte atn(unl) ! Send Unlisten UNL
190  Return
200  Talk: wbyte atn(pri_addr+64) ! Send Talk Ad-
     dress
210  Return
220  Untalk:   wbyte atn(unt) ! Send Untalk
230  Return
240  Devclear:   wbyte dcl ! Send Device Clear
250  Return
260  Selctclr:   wbyte sdc(pri_addr),atn(unl)
270  ! Send MLA, Selected Device Clear, UNL
280  Return
290  Lockout:    wbyte llo ! Send Local Lockout
300  Return
310  Gt local:   wbyte gtl(pri_addr),atn(unl)
320  !  Send MLA, Go to Local, UNL Trigger, UNL
330  Return
340  Loclstat:    wbyte ren(0),ren(1)
350  !  Pulse unassert REN line
360  Return
```

The SG 5030 responds to DCL and SDC by clearing its I/O Buffer and any unexecuted setting commands in its Pending Settings Buffer, along with any errors or events waiting to be reported (except power-on).

LLO locks out the operator from restoring local (front-panel) control when the instrument is under remote control.

GTL restores local control if the instrument receives the message while listen addressed.

See the SG 5030 Operators manual for a full discussion of how the instrument responds to interface messages.

QuickBASIC Sample Utility Program for IBM PC Compatibles

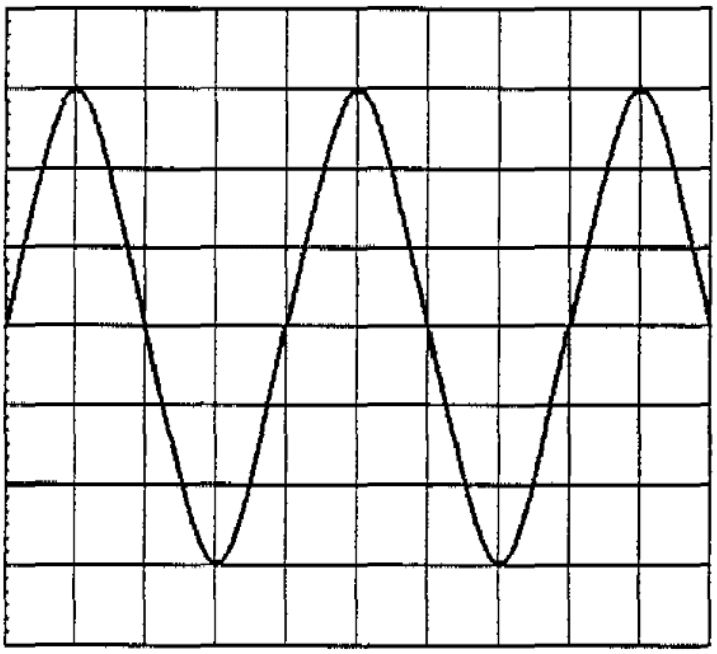
This program illustrates a method to check oscilloscope bandwidth with the SG 5030. Read the comment portions of this program for controller requirements and operation. All or part of this program may be used in your software.

Preliminary information to use this program is detailed earlier in this guide under Sending Messages to the SG 5030.

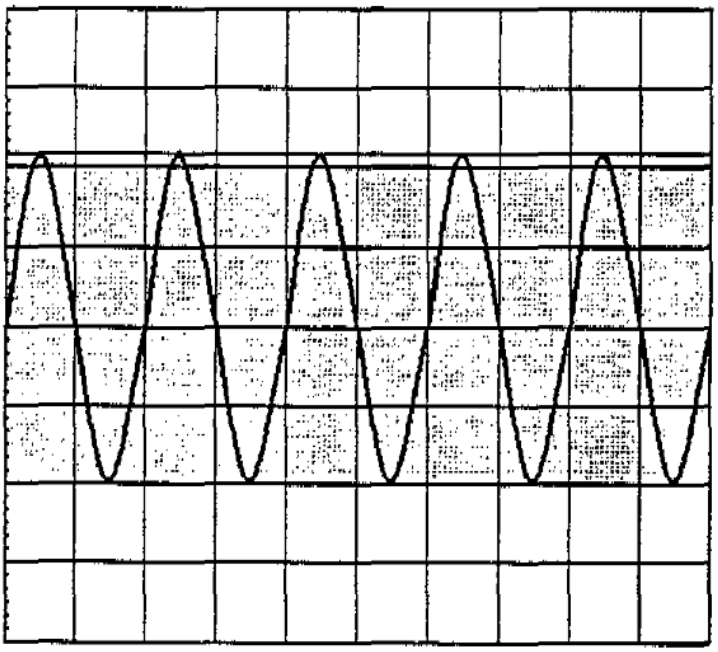
```
' ******************************************************************************
' ********************** SG 5030 SAMPLE MEASUREMENT PROGRAM ************************
' ******************************************************************************
' BY BRUCE VIRELL, MPD MARKETING, 2-2-90
' COPYRIGHT (C) 1988, TEKTRONIX, INC. ALL RIGHTS RESERVED. THIS SOFTWARE
' IS PROVIDED ON AN AS IS BASIS WITHOUT WARRANTY OF ANY KIND, AND IS NOT
' SUPPORTED BY TEKTRONIX.
' THIS SOFTWARE MAY BE REPRODUCED IN WHOLE OR IN PART WITHOUT PRIOR
' PERMISSION. COPIES MUST INCLUDE THE ABOVE COPY RIGHT AND WARRANTY
' NOTICE.
' REQUIRED EQUIPMENT: SG 5030, TM 5000 SERIES MAINFRAME,
' OSCILLOSCOPE, & GPIB CABLING.
'
' THIS PROGRAM REQUIRES THAT THE SG 5030 ADDRESS TO BE SET
' TO THE FACTORY DEFAULT OF 25.
' PC REQUIREMENTS:
' TEKTRONIX PEP 301,IBM PC,XT,AT,PORTABLE PC, OR COMPATIBLE.
' IBM 5153 COLOR DISPLAY,OR EQUIVALENT, OR A COMPOSITE VIDEO MONITOR
' (COLOR OR MONOCHROME).
' S3FG100 GURU II HARDWARE/SOFTWARE & BASICA, WITH IBM PC DOS 2.0 OR
' HIGHER OR MS-DOS 2.02 OR HIGHER.
' PROGRAM PURPOSE:
' TO PROVIDE A SAMPLE OF INSTRUMENT COMMANDS WHICH ILLUSTRATES A
' METHOD OF IMPLIMENTING A PROGRAM TO VERIFY OSCILLOSCOPE AMPLIFIER
' BANDWIDTHS.
' OPERATING PROCEDURE:
' CONNECT THE CONTROLLER VIA GPIB CABLE, TO THE TM 5000 POWER MODULE.
' THE SG 5030 MUST BE SET FOR FACTORY GPIB PRIMARY ADDRESS OF 25.
' SET THE OSCILLOSCOPE CHANNEL 1 FOR 1V/DIVISION AND CONNECT THE SG 5030
' OUTPUT TO CHANNEL 1. THE PROGRAM CONFIGURES THE SG 5030 AND ASKS THE
' OPERATOR TO ADJUST THE SG 5030 AMPLITUDE FOR 8 DIVISIONS ON THE
' OSCILLOSCOPE DISPLAY (REFER TO FIG.1). THE PROGRAM THEN INCREMENTS THE
' SG 5030 OUTPUT FREQUENCY. THE OPERATOR MUST PUSH KEY (F1) WHEN THE
' SG 5030 SIGNAL AMPLITUDE REACHES 4.2 DIVISIONS ON THE OSCILLOSCOPE
' SCREEN.
' VARIABLES:
' IBINIT1 = MEMORY LOCATION (BASICA WORKSPACE - SIZE(BIB728.M)+SIZE(BIB.M))
' IBINIT2 = IBINIT1 + 3
' SG$ = TEKDEV1.IBCONF DRIVER FOR TEKTRONIX INSTRUMENTATION
' WRT$ = COMMAND VARIABLE SENT TO CG VIA IBWRT ROUTINE
' RD$ = VALUE RETURNED FROM INSTRUMENT REQUESTED VIA IBRD ROUTINE
' SPACE$(xx) = DIMMENSIONS STRING VARIABLE SIZE
' V% = INSTRUMENT PRIMARY ADDRESS, 15
' BD% = V%, INSTRUMENT PRIMARY ADDRESS
' SPR% = STATUS BYTE RETURNED VIA IBRSP ROUTINE
' IBSTA% = STATUS WORD RETURNED VIA EACH FUNCTION CALL
' IBERR% = ERROR BIT RETURNED VIA EACH FUNCTION CALL
' GURU II CALL FUNCTIONS:
' IBWRT = SENDS COMMANDS TO INSTRUMENT
```

```
' IBRD = RETURNS PARAMETERS FROM INSTRUMENT
' IBRSP = PERFORMS SERIAL POLL OF INSTRUMENT
' IBPAD = CONVERTS IBCONF ADDRESS TO 25
' IBRDF = RETURNS INSTRUMENT PARAMETERS TO FILE
' REFER TO GURU II DOCUMENTATION FOR A COMPLETE LISTING.
' ******************************************************************
' GURU II/ PEP 301 INCLUDE GURU CALL FUNCTION LINES IN BASICA PROGRAMS
' ******************************************************************
COMMON SHARED IBSTA%, IBERR%, IBCNT%
ID$ = "TEKDEV1"
CALL ibfind(ID$, BD%)                  ' SELECT TEKDEV1 FOR GPIB ACCESS
SG% = 25                               ' SG% = FACTORY DEFAULT ADDRESS OF 25
CALL ibpad(BD%, SG%)                   ' CHANGE TEKDEV1 PRIMARY ADDRESS TO 25
ID$ = "GPIB0"                          ' SET UP GPIB FOR BOARD LEVEL
CALL ibfind(ID$, GP%)                  ' COMMUNICATION
REMOTE% = 1
CALL IBSRE(GP%, REMOTE%)               ' SET REMOTE ENABLE
CLS                                    ' CLEAR SCREEN
REPLY$ = SPACE$(125)                   ' DIMENSION REPLY$ TO 125
SET$ = SPACE$(125)                     ' DIMENSION SET$ TO 125
WRT$ = SPACE$(125)                     ' DIMENSION WRT$ TO 125
RD$ = SPACE$(125)                      ' DIMENSION WRT$ TO 125
BANDW$ = SPACE$(125)                   ' DIMENSION BANDW$ to 125
FREQ$ = SPACE$(18)                     ' DIMENSION FREQ$ TO 18
MAINPROG:
CLS
PRINT "*****************************************************************"
PRINT "******************* SG 5030 SAMPLE PROGRAM ************************"
PRINT "*****************************************************************"
CALL IBWRT(BD%, "INIT")                 ' SEND MESSAGE TO INST
GOSUB ERRORGPIB                         ' CHECK GPIB ERROR
rngval = 50000
ANSFLAG = 0                             ' PRINTER FLAG
WRT$ = "FREQ " + STR$(rngval) + ";OUT ON;"
CALL IBWRT(BD%, WRT$)                   ' SEND MESSAGE TO INST
PRINT : INPUT "WOULD YOU LIKE TO LOG RESULTS TO LINE PRINTER?(Y/N)"; ANS$
IF ANS$ = "y" OR ANS$ = "Y" THEN ANSFLAG = 1
PRINT : INPUT "ENTER YOUR NAME"; REPLY$
LOCATE 9, 43: PRINT "MHZ"
LOCATE 9, 1: INPUT "ENTER THE SPECIFIED SCOPE BANDWIDTH"; BANDW$
PRINT : PRINT "SET SCOPE FOR PRECISELY 6 DIVS OF SINEWAVE AMPLITUDE"
INPUT "       THEN PRESS ENTER TO CONTINUE"; A$
PRINT : PRINT "PRESS KEY F1 WHEN SCOPE DISPLAY REACHES 4.2  DIVISIONS"
KEY(1) OFF: ON KEY(1) GOSUB TERMINATE: KEY(1) ON                 'ENABLE KEY F1
KEY(2) OFF: KEY(3) OFF: KEY(4) OFF: KEY(5) OFF: KEY(6) OFF       'DISABLE OTHER KEYS
       KEY(7) OFF: KEY(8) OFF: KEY(9) OFF: KEY(10) OFF           'DISABLE OTHER KEYS
CHECKCG:
IF rngval < 800000 * VAL(BANDW$) THEN rngval = rngval + 5000000
IF rngval > 800000 AND rngval < 990000 * VAL(BANDW$)
THEN rngval = rngval + 500000
IF rngval > 990000 * VAL(BANDW$) THEN rngval = rngval + 200000
WRT$ = "FREQ " + STR$(rngval) + ";OUT ON;"
CALL IBWRT(BD%, WRT$)                    ' SEND MESSAGE TO INSTRUMENT
IF rngval > 1200000 * VAL(BANDW$) THEN GOTO TERMINATE
FOR T = 1 TO 10000
NEXT T
```

```
GOTO CHECKCG
ERRORGPIB: ' ******************** ERROR ROUTINE ****************************
IF IBSTA% >= 0 AND BD% >= 0 AND IBSTA% < &H4000 AND IBERR% <> 6
THEN RETURN'no error to report
IF BD% < 0 THEN PRINT "device not installed - use IBCONF then reboot"
IF IBSTA% > 0 AND IBSTA% >= &H4000 THEN PRINT "timeout"
IF IBERR% = 6 THEN PRINT "timeout"
PRINT "gpib error "; IBERR%
IF IBERR% = 0 THEN PRINT "DOS error device not installed"
IF IBERR% = 1 THEN PRINT "function requires GPIB-PC to be CIC"
IF IBERR% = 2 THEN PRINT "no listner on write function"
IF IBERR% = 3 THEN PRINT "GPIB-PC not addressed correctly"
IF IBERR% = 4 THEN PRINT "invalid argument to function call"
IF IBERR% = 5 THEN PRINT "GPIB-PC not system controller as required"
IF IBERR% = 6 THEN PRINT "I/O operation aborted"
IF IBERR% = 7 THEN PRINT "non-existant GPIB-PC board"
IF IBERR% = 10 THEN PRINT "I/O started before previous operation completed"
IF IBERR% = 11 THEN PRINT "no capability for operation"
IF IBERR% = 12 THEN PRINT "file system error"
IF IBERR% = 14 THEN PRINT "command error during device call"
IF IBERR% = 15 THEN PRINT "serial poll status byte lost"
IF IBERR% = 16 THEN PRINT "SRQ stuck in on position"
INPUT "[ENTER] TO CONTINUE"; A$' if help$ then
RETURN
TERMINATE: ' ****************** TERMINATE PROGRAM ************************
CALL IBWRT(BD%, "SET?")                    ' SEND MESSAGE TO INST
CALL IBRD(BD%, RD$)
SET$ = RD$
CALL IBWRT(BD%, "FREQ?")                   ' SEND MESSAGE TO INST
CALL IBRD(BD%, RD$)
FREQ$ = LEFT$(RD$, 16)
IF ANSFLAG = 0 THEN GOTO SCREENPRT:       ' TEST FOR LINE PRINT REQUEST
LPRINT
LPRINT : LPRINT "OPERATOR NAME:"          ' OPTIONAL LINE PRINTER OUTPUT
LPRINT REPLY$
LPRINT : LPRINT "SG 5030 SETTINGS:"
LPRINT SET$
LPRINT : LPRINT "SCOPE BANDWIDTH:"
LPRINT FREQ$, "MHZ"
LPRINT
SCREENPRT:                                ' PRINT TO SCREEN INFORMATION
PRINT: PRINT
PRINT : PRINT "SG 5030 SETTINGS"
PRINT SET$
LOCATE 21, 1: PRINT "SCOPE BANDWIDTH"
PRINT FREQ$
LOCATE 22, 19: PRINT "MHZ"
INPUT "                              RUN TEST AGAIN? (Y/n)"; RUNTEST$
IF RUNTEST$ = "Y" OR RUNTEST$ = "y" THEN GOSUB MAINPROG
PRINT "                              PROGRAM TERMINATED."
END
```



A. Set SG 5030 output amplitude for 6 vertical divisions of amplitude at 50 kHz on the oscilloscope display.



B. Increase the SG 5030 frequency until the displayed waveform amplitude drops to 4.2 vertical divisions. This frequency reflects the bandwidth of the oscilloscope, also known as the -3 dB point.

7704-1

Figure 1. Waveform results of the SG 5030 Sample Measurement Program.

QuickBASIC Talker/Listener Utility Program For IBM PC Compatibles

This program allows the user to send any of the instrument commands to the SG 5030. It is useful in learning the command set and in software debugging.

```
' ***********************************************************************
' ****************** SG 5030 TALKER/LISTENER PROGRAM ********************
' ***********************************************************************
' THIS PROGRAM REQUIRES THAT THE SG 5030 ADDRESS BE SET TO THE
' FACTORY DEFAULT OF 25.
COMMON SHARED IBSTA%, IBERR%, IBCNT%
ID$ = "TEKDEV1"
CALL IBFIND(ID$, BD%)                  ' SELECT TEKDEV1 FOR GPIB ACCESS
SG% = 25                               ' SG% = FACTORY DEFAULT ADDRESS OF 25
CALL IBPAD(BD%, SG%)                   ' CHANGE TEKDEV1 PRIMARY ADDRESS TO 25
ID$ = "GPIB0"                          ' SET UP GPIB FOR BOARD LEVEL
CALL IBFIND(ID$, GP%)                  ' COMMUNICATION
REMOTE% = 1
CALL IBSRE(GP%, REMOTE%)               ' SET REMOTE ENABLE
CLS                                    ' CLEAR SCREEN
REPLY$ = SPACE$(125)                   ' DIMENSION RD$ TO 125
PRINT "*******************************************************************"
PRINT "****************** SG 5030 TALKER LISTENER PROGRAM ****************"
PRINT "*******************************************************************"
MAINPROG:
PRINT "RETURN TO EXIT: "
INPUT "ENTER MESSAGE(S)"; WRT$
IF WRT$ = "" THEN GOSUB TERMINATE
CALL IBWRT(BD%, WRT$)                  ' SEND MESSAGE TO SG 5030
GOSUB CHECKGPIB                        ' CHECK FOR GPIB ERROR
' ************************ INPUT FROM DEVICE ******************************
FOR T = 1 TO 1000
NEXT T
CALL IBRD(BD%, REPLY$)                 ' INPUT DATA FROM SG 5030
GOSUB CHECKGPIB                        ' CHECK FOR GPIB ERROR GOSUB
CHECKCG                                ' GET SG 5030 ERROR MESSAGE
PRINT : PRINT "INSTRUMENT REPLY ", REPLY$
PRINT : PRINT "Returned status byte:"; SPR%,
PRINT : PRINT ERRM$
GOTO MAINPROG
' ************************** ERROR ROUTINE ********************************
CHECKCG:
ERRM$ = SPACE$(50)
CALL IBRSP(BD%, SPR%)
CALL IBWRT(BD%, "ERR?")
CALL IBRD(BD%, ERRM$)
RETURN
CHECKGPIB:
IF IBSTA% >= 0 AND BD% >= 0 AND IBSTA% < &H4000 AND IBERR% <> 6 THEN
RETURN'no error to report
IF BD% < 0 THEN PRINT "device not installed - use IBCONF then reboot"
IF IBSTA% > 0 AND IBSTA% >= &H4000 THEN PRINT "timeout"
IF IBERR% = 6 THEN PRINT "timeout"
```

```
PRINT "gpib error "; IBERR%
IF IBERR% = 0 THEN PRINT "DOS error device not installed"
IF IBERR% = 1 THEN PRINT "function requires GPIB-PC to be CIC"
IF IBERR% = 2 THEN PRINT "no listner on write function"
IF IBERR% = 3 THEN PRINT "GPIB-PC not addressed correctly"
IF IBERR% = 4 THEN PRINT "invalid argument to function call"
IF IBERR% = 5 THEN PRINT "GPIB-PC not system controller as required"
IF IBERR% = 6 THEN PRINT "I/O operation aborted"
IF IBERR% = 7 THEN PRINT "non-existant GPIB-PC board"
IF IBERR% = 10 THEN PRINT "I/O started before previous operation completed"
IF IBERR% = 11 THEN PRINT "no capability for operation"
IF IBERR% = 12 THEN PRINT "file system error"
IF IBERR% = 14 THEN PRINT "command error during device call"
IF IBERR% = 15 THEN PRINT "serial poll status byte lost"
IF IBERR% = 16 THEN PRINT "SRQ stuck in on position"
INPUT "[ENTER] TO CONTINUE"; A$' if help$ then
RETURN
' ********************** TERMINATE PROGRAM ****************************
TERMINATE:
REMOTE% = 0
CALL IBSRE(GP%, REMOTE%)              ' CLEAR REMOTE ENABLE
PRINT "PROGRAM TERMINATED."
END
```

ASCII & GPIB CODE CHART

B7 B6 B5 BITS / B4 B3 B2 B1	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	CONTROL		NUMBERS SYMBOLS		UPPER CASE		LOWER CASE	
0 0 0 0	0 NUL 0 0	20 DLE 10 16	40 SP 20 32	60 0 30 48	100 @ 40 64	120 P 50 80	140 ` 60 96	160 p 70 112
0 0 0 1	1 GTL SOH 1 1	21 LLO DC1 11 17	41 ! 21 33	61 1 31 49	101 A 41 65	121 Q 51 81	141 a 61 97	161 q 71 113
0 0 1 0	2 STX 2 2	22 DC2 12 18	42 " 22 34	62 2 32 50	102 B 42 66	122 R 52 82	142 b 62 98	162 r 72 114
0 0 1 1	3 ETX 3 3	23 DC3 13 19	43 # 23 35	63 3 33 51	103 C 43 67	123 S 53 83	143 c 63 99	163 s 73 115
0 1 0 0	4 SDC EOT 4 4	24 DCL DC4 14 20	44 $ 24 36	64 4 34 52	104 D 44 68	124 T 54 84	144 d 64 100	164 t 74 116
0 1 0 1	5 PPC ENQ 5 5	25 PPU NAK 15 21	45 % 25 37	65 5 35 53	105 E 45 69	125 U 55 85	145 e 65 101	165 u 75 117
0 1 1 0	6 ACK 6 6	26 SYN 16 22	46 & 26 38	66 6 36 54	106 F 46 70	126 V 56 86	146 f 66 102	166 v 76 118
0 1 1 1	7 BEL 7 7	27 ETB 17 23	47 ' 27 39	67 7 37 55	107 G 47 71	127 W 57 87	147 g 67 103	167 w 77 119
1 0 0 0	10 GET BS 8 8	30 SPE CAN 18 24	50 (28 40	70 8 38 56	110 H 48 72	130 X 58 88	150 h 68 104	170 x 78 120
1 0 0 1	11 TCT HT 9 9	31 SPD EM 19 25	51) 29 41	71 9 39 57	111 I 49 73	131 Y 59 89	151 i 69 105	171 y 79 121
1 0 1 0	12 LF A 10	32 SUB 1A 26	52 * 2A 42	72 : 3A 58	112 J 4A 74	132 Z 5A 90	152 j 6A 106	172 z 7A 122
1 0 1 1	13 VT B 11	33 ESC 1B 27	53 + 2B 43	73 ; 3B 59	113 K 4B 75	133 [5B 91	153 k 6B 107	173 { 7B 123
1 1 0 0	14 FF C 12	34 FS 1C 28	54 , 2C 44	74 < 3C 60	114 L 4C 76	134 \ 5C 92	154 l 6C 108	174 | 7C 124
1 1 0 1	15 CR D 13	35 GS 1D 29	55 - 2D 45	75 = 3D 61	115 M 4D 77	135] 5D 93	155 m 6D 109	175 } 7D 125
1 1 1 0	16 SO E 14	36 RS 1E 30	56 . 2E 46	76 > 3E 62	116 N 4E 78	136 ^ 5E 94	156 n 6E 110	176 ~ 7E 126
1 1 1 1	17 SI F 15	37 US 1F 31	57 / 2F 47	77 UNL ? 3F 63	117 O 4F 79	137 UNT _ 5F 95	157 o 6F 111	177 DEL (RUBOUT) 7F 127
	ADDRESSED COMMANDS	UNIVERSAL COMMANDS	LISTEN ADDRESSES		TALK ADDRESSES		SECONDARY ADDRESSES OR COMMANDS	

KEY

octal | 25 PPU | GPIB code
NAK | ASCII character
hex | 15 21 | decimal

Tektronix
COMMITTED TO EXCELLENCE

REF: ANSI STD X3. 4-1977
IEEE STD 488-1978
ISO STD 646-1973

7703-8

Figure 2. ASCII and IEEE (GPIB) Code Chart.

TEKTRONIX INSTRUMENTATION SOFTWARE LIBRARY

Tekware Programs Complement Tektronix Instruments and Controllers

With technology more complex and schedules becoming shorter, the need to automate test and measurement environments is greater than ever.

Tek can meet your need for improved test and measurement productivity in the lab, on the factory floor, or at the remote site. Tekware programs are designed to enhance the productivity of Tektronix instruments and controllers.

Tektronix has performed extensive software testing to verify that all standard Tekware measurement, analysis, and utility packages run reliably with Tek instruments and controllers. MS-DOS Tekware programs have been tested for compatibility with 286/386 IBM compatible PC system controllers.

Applications Software

Tekware applications software provides a wide selection of ready-to-use general purpose control, measurement, and display capabilities. Simply load Tekware applications into your controller, and your system is ready to go to work for you. If your software requirements do not call for unique processing or specialized control functions, Tekware applications software could be the shortest path to a complete system solution.

Development Software

Tekware development software tools can be used to create new software or to modify existing software. Some Tekware development software products contain test program generators enabling you to create test procedures without writing a single line of code.

If your testing requirements call for development of a custom program or for customizing an existing program, a Tekware development software tool could be the key to the solution.

Software Support

To keep your software current and up-to-date, our software support is available at no charge during the warranty period. Post-warranty software support is available on a subscription basis.

Technical Assistance Services (TAS)

When you need technical assistance to supplement your own resources, Tektronix can provide the services of an Applications Engineer skilled in meeting your needs. For more information contact your local Tektronix Field office.

Tektronix Test Management System (TEKTMS)

TekTMS provides a highly productive software package for test development and execution in manufacturing, repair/rework, and prototype development testing applications.

EZ-TEST Test Development Software

EZ-TEST is a software productivity tool used to create and run test software for manufacturing, service repair and rework, metrology, and prototype evaluation.

EZ-TEST is for non-programmers or programmers who prefer to concentrate on testing rather than software coding chores.

The EZ-TEST software system provides all the software necessary to develop and run test system programs. The major elements of EZ-Test are:

- Generator program to create, debug, and execute the test procedure.
- Translator program to convert the procedure to Microsoft QuickBASIC code and then compile and link it, resulting in a complete stand-alone test program.
- Test Execution Scheduler to run sequences of compiled tests and gather test data.
- Microsoft QuickBASIC editor, compiler, linker and manuals.
- GPIB interface software for QuickBASIC.

The following Software/Hardware was available when this Instrument Interfacing Guide was printed. Other software may be available; contact your local Tektronix Field office for further information.

Description	Tektronix Part No.
Tek EZ-Test PC 360 K Floppy Disk	S45F030
GURU II hardware/software package for PC	S3FG100
TekTMS	S3FT100
GPIB Inst Lib	S3FT300

Ordering Utility Software U.S. Only

Your local Tektronix Field office has the current prices for software available from the Tektronix Instrumentation Software Library.

Call the toll-free number serving your area and give the Customer Service Representative the Tektronix nine-digit part number and name of the software package you want to order. If you have any questions about the software, call your local Tektronix Field office.