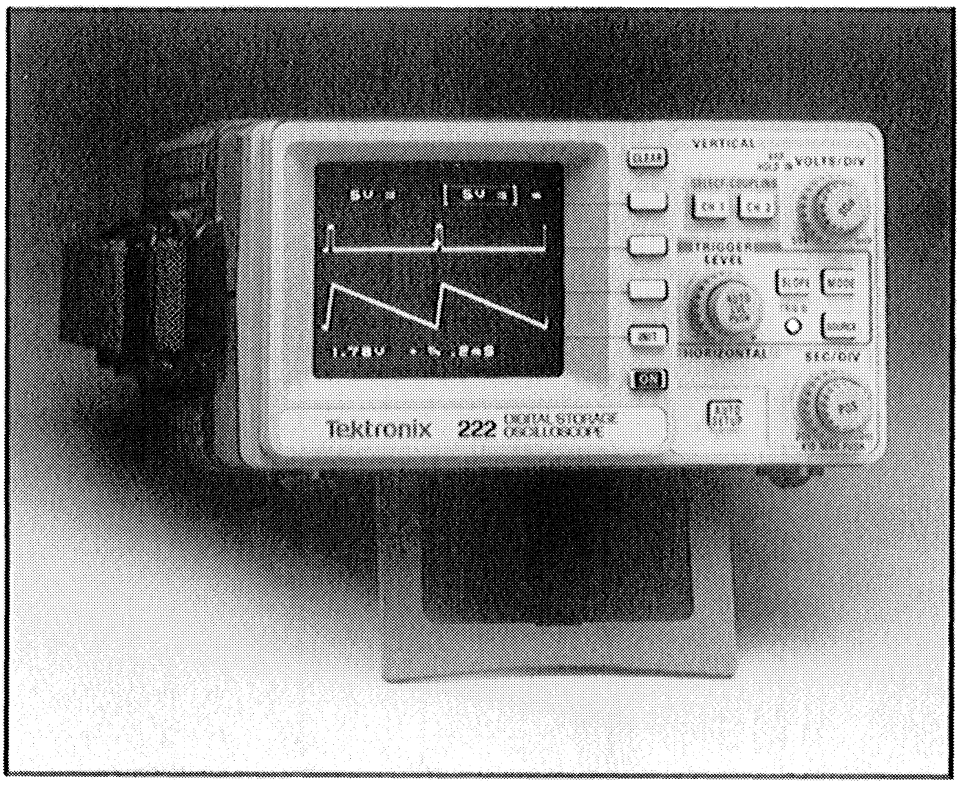
222 RS-232 Interfacing Guide



For additional copies, order:
070-7533-00
First Printing FEB 1989

Tektronix®
COMMITTED TO EXCELLENCE

CONTENTS

RS-232 INTERFACE

HOST COMMUNICATION

Communication between the 222 and an external host computer is provided via an RS-232 communications interface. The actual format of the commands is as close to the Tek Codes and Formats as possible with the limited memory space available. Certain functions are compressed into a more compact form.

Waveforms and front-panel settings may be transferred over the RS-232 serial port in ascii character strings. The optional RS-232 Demonstration Disk accessory (available by request only) supports communications with an IBM® PC AT® or PC compatible computer. Operating details are provided on the disk in a "README" file.

A simple terminal can be used to receive the messages that may be generated at power on or during self calibration if an error occurs. The error codes are defined later in this manual. An RS-232 interconnection cable for use with the 222 is available as an optional accessory. Setting the communication baud rate for the 222 is done using menu selections available in the Auxiliary Functions menu under CONFIG.

RS-232 INTERFACE SPECIFICATIONS

RS-232 Communication Parameters

Start bits	1
Stop bits	1
Data bits	8
Parity	None
Flow Control	XON/XOFF
Signals	RX, TX, and SGND are functional. SGND is connected internally to EXT TRIG COMM. DSR and CTS are always high, and DTR and RTS are ignored.

Baud Rates

300, 1200, 2400, 9600; 0.1% accuracy based on the microprocessor clock.

IBM PC XT and IBM PC AT are registered trademarks of the International Business Machine Corp.

Levels

Compatible with RS-232C.

Maximum Applied Voltage (any pin)

25 V (dc + peak ac).

Plotter/Printer Support

There is no plotter support in the instrument.

Messages

User message displays on the instrument crt are not supported. The controlling PC cannot send messages to be displayed for the operator of the 222.

RS-232 OPTIONAL ACCESSORIES

RS-232 Demo Disk	063-0070-00
RS-232 Interconnection Cable	175-1453-00

CODES AND FORMATS

Commands, front panel setups, and waveforms sent over the RS-232 interface are composed of ascii character strings. The ascii characters represent hexadecimal numbers, with two characters per hexadecimal value. (See "Logical Front Panel Encoding" for the meanings of the numbers.)

The general command format is as follows:

cmd arg:data;

where cmd is the command; arg is an argument to the command, and data is any additional data needed for the command. A single white space (either a space or a tab) is needed between the command and argument. There must be no white space on either side of the colon delimiter between the argument and the data. The semicolon is the command terminator, but a carriage return also terminates the command.

NOTE

The command is executed as soon as the command terminator is received; multiple commands in a string are executed at the command terminator as they are received. A command can be aborted after it has started execution by sending an ESCAPE character.

Commands, Queries, and Responses

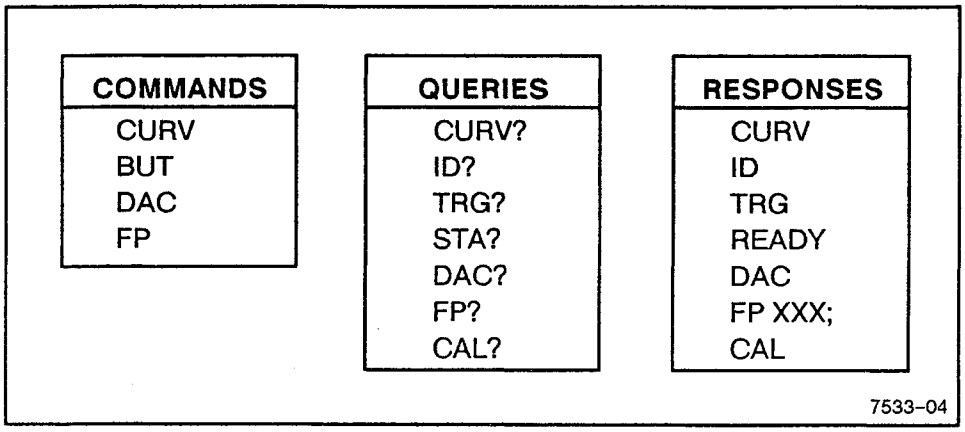


Figure 1. Commands, Queries, and Responses.

The set of commands and queries sent to the instrument and the responses sent by the instrument are shown in Figure 1. Explanation for each is given in the following text.

COMMANDS. The following commands are supported:

CURV <frame>:
<wfrm data> Loads waveform data into the specified 222 reference memory (REF1 – REF4). The wfrm data must be sent as hex-encoded ascii characters.

Note

Waveforms may be written back to Channel 1 and Channel 2, but they will be overwritten by the next acquisitions into those memories.

<frame> is one of the following strings:

CH 1	Channel 1 waveform
CH 2	Channel 2 waveform
REF1	Reference waveform 1
REF2	Reference waveform 2
REF3	Reference waveform 3
REF4	Reference waveform 4

<wfrm data> is composed of <fp data> <frame nr> <byte count> <waveform data> <checksum>, defined as follows:

<fp data> Ten ascii characters representing hexadecimal bytes of the front panel settings for the waveform data. See "Logical Front Panel Encoding" for the meaning of the characters.

<frame nr> Two ascii character representing the frame number (in hexadecimal) the curve data is taken from. The frame numbers range from 00 for the RO (readout) frame to 06 for the REF4 (reference waveform 4 frame).

<byte count> Four ascii characters representing the data byte count for the waveform data (in hexadecimal).

<waveform data> In YT mode, each data point is the hexadecimal value (represented by two ascii characters) of the Y-coordinate.

In XY mode, the X- and Y-coordinates are sent as two hexadecimal values (represented by four ascii characters) with the X-coordinate value first, followed by the Y-coordinate value.

<checksum> Two ascii characters representing the hexadecimal twos complement of the modulo 256 checksum of all data bytes and the byte count bytes.

Note

Programmers take note, additional bytes may be added to the data string after the checksum byte. This space is reserved for future expansion.

BUT <button> Simulates a button press.

<button> is one or two ASCII characters that represent a button code. The button codes are as follows:

1	CLEAR
2	Menu Item 0
3	Menu Item 1
4	Menu Item 2
5	Menu Item 3
6	OFF
9	Trigger SOURCE
A	Trigger MODE
B	Trigger SLOPE
C	CH 2 Select
D	CH 1 Select
E	AUTO SETUP
11	Front-Panel Setup Menu
12	Trigger Position Menu
13	Auxiliary Functions Menu
14	Display Mode Menu
19	Save Waveform Menu
1A	Recall Waveform Menu
1B	STORE/NONSTORE
1C	Acq Mode Menu
20	X10 MAG
21	Variable Gain
22	AUTO LVL: PUSH

The following command string is a simple example of how to use the BUT command to change a front panel setting controlled by button pushes.

BUT 9;BUT 2

calls up the Trigger SOURCE menu then selects menu item 0, the first menu choice (VERT). Notice the semicolon delimiter is not needed at the end of the command string terminated by a carriage return, only between commands in the string.

DAC <dac code>:
<dac value>

Sets a dac value (see Table 1 for the dac code and dac value data).

Table 1

DAC Identification

DAC	DAC CODE	RANGE	SCALE	REFERENCE
Horiz POSITION	00	0–1FFC	± 5 div	Full left 0
CH 1 Trigger LEVEL	01	0–1FFC	±30 div	Center 0FFF
CH 2 Trigger LEVEL	02	0–1FFC	±30 div	Center 0FFF
EXT Trigger LEVEL	03	0–1FFC	± 2.33 V	Center 0FFF
CH 2 VAR Gain	04	0–03FF	-2.5:1 to 1:1	Cal'd 03FF
CH 1 VAR Gain	05	0–03FF	-2.5:1 to 1:1	Cal'd 03FF
CH 2 POSITION	06	0–1FFC	±12 div	Center 0FFF
CH 1 POSITION	07	0–1FFC	±12 div	Center 0FFF

FP <log fp>:
<fp data>

Sends a front panel setup to the <log fp> location. ACQ, REF1–REF4, and STR1–STR4 are logical front panels; front panel data is a string of 10 ascii characters encoding the front panel control settings. (See Tables 2 through 6 for encoding and decoding the fp data.)

<log fp> is one of the following:

ACQ	Acquisition system
REF1	Reference waveform 1
REF2	Reference waveform 2
REF3	Reference waveform 3
REF4	Reference waveform 4
STR1	Front panel setup 1
STR2	Front panel setup 2
STR3	Front panel setup 3
STR4	Front panel setup 4

<fp data> is the 10 ascii characters that represent the five, 2-character hexadecimal bytes of a logical front panel setup.

QUERIES. The following queries are supported:

CURV? <frame>	Request waveform data. The waveform data is sent as hex encoded ascii characters. <frame> is one of six possible sources for curve data (CH1, CH2, or REF-REF4).
ID?	Queries instrument ID and software version.
TRG?	Queries the trigger state.
STA?	Queries the communication task status.
DAC? <dac code>	Queries a dac value. <dac code> is one of eight dacs (digital-to-analog converters) for digitized potentiometer settings. (See Table 1 for dac identification.)
FP? <log fp>	Queries a front panel setup. <log fp> is one of nine possible front-panel data locations.

<log fp> is one of the following:

ACQ	Acquisition system
REF1	Reference waveform 1
REF2	Reference waveform 2
REF3	Reference waveform 3
REF4	Reference waveform 4
STR1	Front panel setup 1
STR2	Front panel setup 2
STR3	Front panel setup 3
STR4	Front panel setup 4

CAL?	Queries the calibration constants.

RESPONSES. The following responses occur as a result of the associated query:

Note

Responses are terminated by just a semicolon if the query or command was terminated by a semicolon. If the query or command was terminated by a carriage return, the response is terminated by a semicolon followed by a carriage return.

CURV <frame>:
<wfrm data>

A frame is one of the six possible source of waveform data in the 222: CH1, CH2, or REF1 through REF4 asked for in the CURV? query. The wfrm data includes the front panel setting and the waveform data point values as hex-encoded ascii characters, encoded the same as the CURVe command.

ID TEK-222
VER:X.XX

This response is in reply to an ID? query. X.XX is the firmware version installed in instrument.

TRG YES or NO

This response indicates if the TRIG'D LED is on (YES—triggered) or off (NO—not triggered) in response to the TRG? query.

READY

This is the reply to a STA? query when the instrument is ready to communicate. The same response is sent as the result of a carriage return. If the instrument is not ready, the reply is delayed until the instrument is ready.

DAC <dac code>:
<dac value>

The digital to analog converters (dac) determine the variable control settings for the POSITION controls (horizontal and vertical), the trigger

LEVEL settings (CH1, CH2, and EXT), and the variable vertical gain (CH1 and CH2). In response to the DAC? <dac code> query, the setting of the queried dac is returned in the same form as the command use to set a new dac value (see Table 1 for dac identification data).

FP <log fp>: <fp data>

This response is also in the same form as the command for setting the controls for a new setup. The <log fp> choices are:

ACQ (the current front panel settings),

REF1 – REF4 (the front panel setting for the stored reference waveforms), and

STR1 – STR4 (the four stored front panel setups).

The front panel data (fp data) is a 10-ascii character string encoded with the front panel control settings.

The response to FP? ACQ returns data in the following format:

FP ACQ:24240C2112;

The byte decoding (by bit) for the front panel settings is given in Tables 2 through 6, with each table showing the decoding for one of the hexadecimal bytes.

CAL <calibration constant data>

The response to a CAL? query is a string of ASCII characters representing the calibration constants stored in the instrument.

Logical Front Panel Encoding

Tables 2 through 6 show how the 10 ascii characters of the logical front panel hexadecimal bytes are encoded. The tables divide the coded number of the front panel setup into five bytes of two characters each. Tables 2 and 3 are very similar in that the same data is encoded in each, with Table 2 for CH 1 and Table 3 for CH 2.

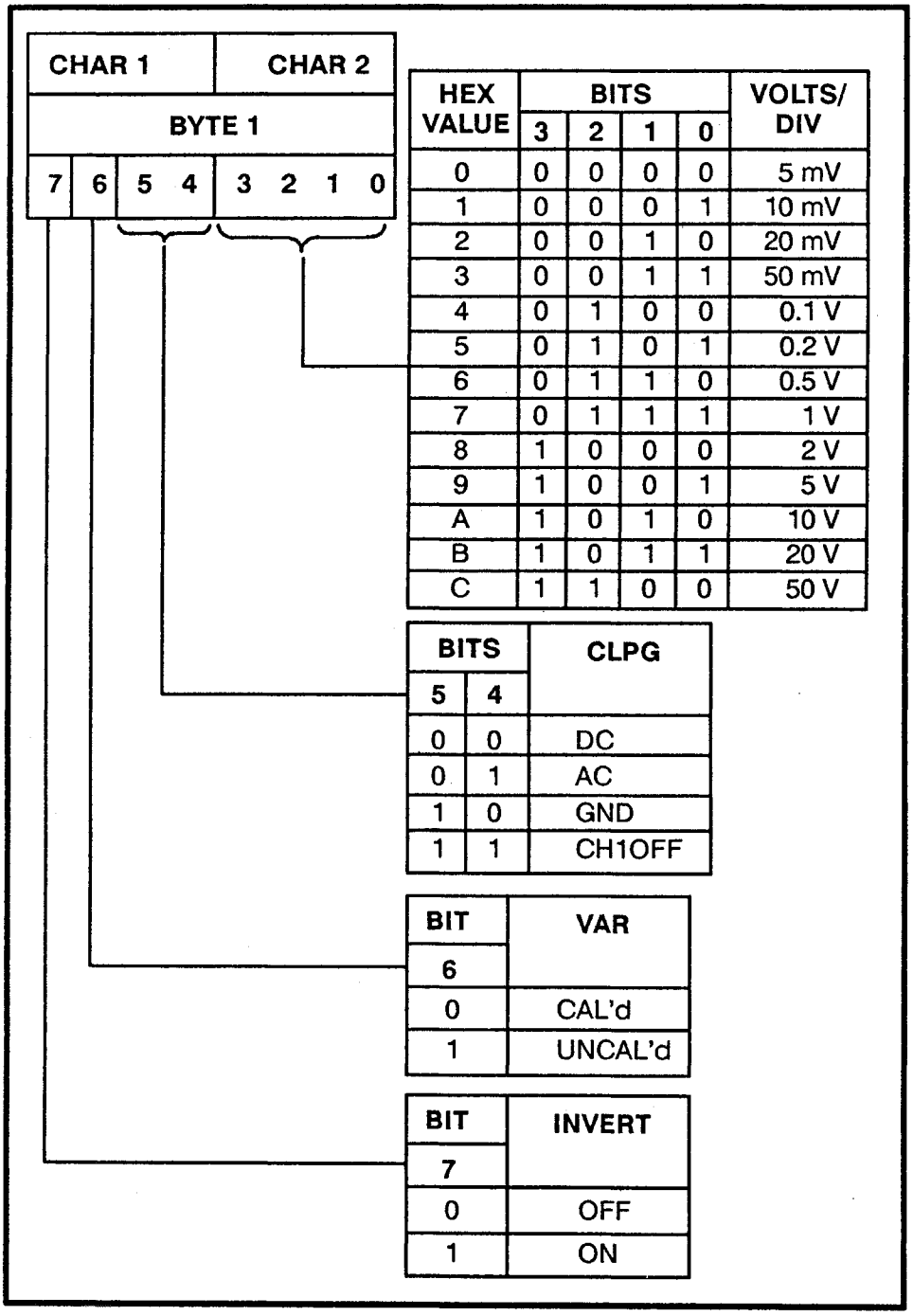
VERTICAL SETTINGS. From Tables 2 and 3 you can decode the ascii characters for the CH 1 and CH 2 vertical settings. In the example setup data string, the first four characters are: 2424. Breaking these numbers into binary bits, a 2 gives 0010, and a 4 gives 0100. Looking at the bit information for the first character tells us that INVERT is OFF, VAR is OFF, and the input coupling is GND. For the second character, its bit values are given, but looking at the HEX value column tells us that the VOLTS/DIV setting is 0.1 V per division. Exactly the same values are given for characters 3 and 4 as 1 and 2 respectively; therefore, the CH 2 settings are the same as the CH 1 settings.

To change the VOLTS/DIV setting for CH 1 to 1 V, change the value of character 2 from 4 to 7 in the front panel setup string when it is sent back to the oscilloscope. The string sent back then is:

FP ACQ:27240C2112

Notice that a semicolon delimiter is needed only if further commands are included in the string. The command is executed as soon as the command terminator (either a semicolon or a carriage return) is received; multiple commands in a string are executed as they are recognized at the command terminator.

Table 2
Channel 1 Settings



CHAR 1	CHAR 2
BYTE 1	
7 6 5 4	3 2 1 0

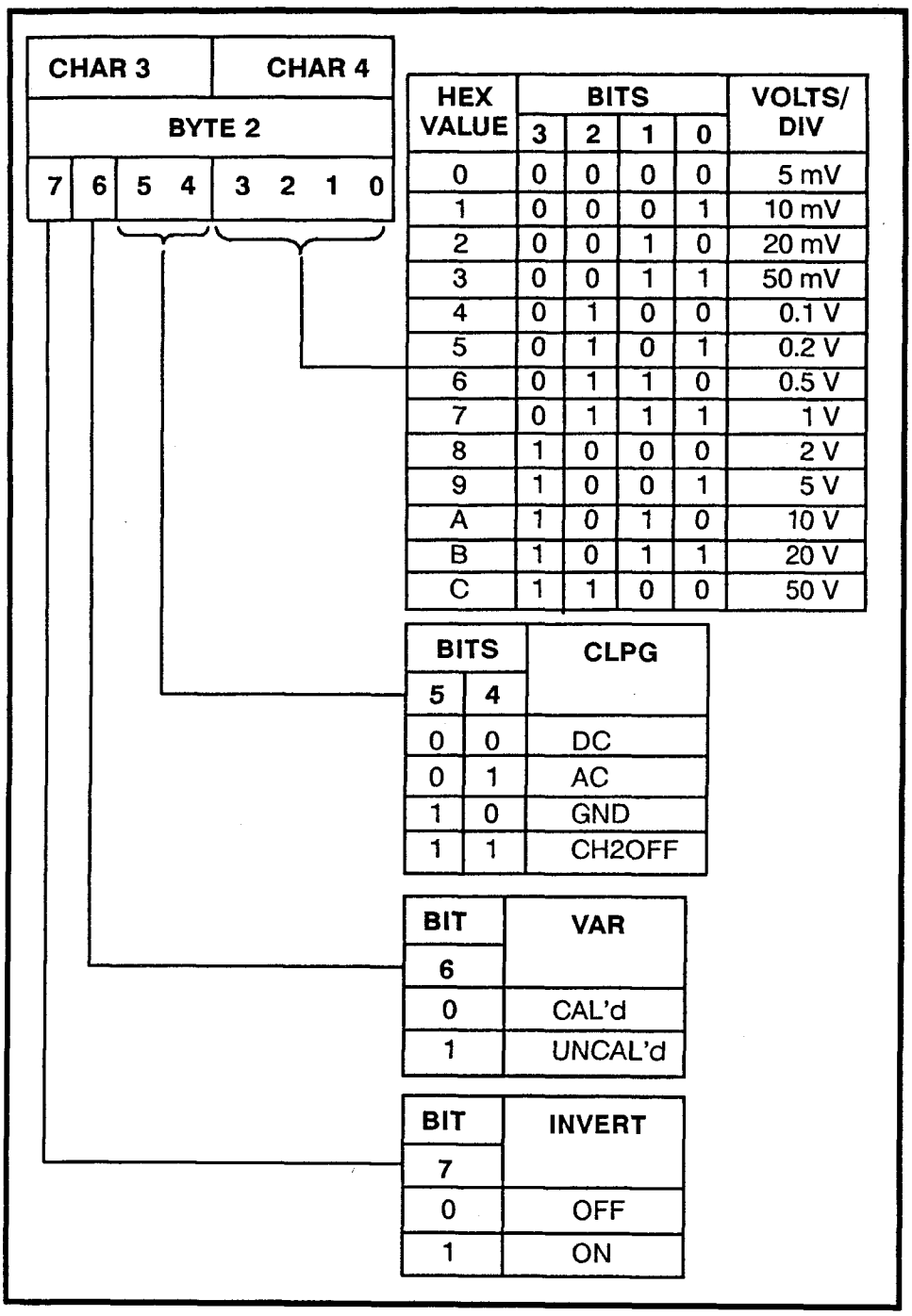
HEX VALUE	BITS 3	2	1	0	VOLTS/ DIV
0	0	0	0	0	5 mV
1	0	0	0	1	10 mV
2	0	0	1	0	20 mV
3	0	0	1	1	50 mV
4	0	1	0	0	0.1 V
5	0	1	0	1	0.2 V
6	0	1	1	0	0.5 V
7	0	1	1	1	1 V
8	1	0	0	0	2 V
9	1	0	0	1	5 V
A	1	0	1	0	10 V
B	1	0	1	1	20 V
C	1	1	0	0	50 V

BITS 5	4	CLPG
0	0	DC
0	1	AC
1	0	GND
1	1	CH1OFF

BIT 6	VAR
0	CAL'd
1	UNCAL'd

BIT 7	INVERT
0	OFF
1	ON

Table 3
Channel 2 Settings



CHAR 3	CHAR 4
BYTE 2	
7 6 5 4	3 2 1 0

HEX VALUE	BITS 3	2	1	0	VOLTS/ DIV
0	0	0	0	0	5 mV
1	0	0	0	1	10 mV
2	0	0	1	0	20 mV
3	0	0	1	1	50 mV
4	0	1	0	0	0.1 V
5	0	1	0	1	0.2 V
6	0	1	1	0	0.5 V
7	0	1	1	1	1 V
8	1	0	0	0	2 V
9	1	0	0	1	5 V
A	1	0	1	0	10 V
B	1	0	1	1	20 V
C	1	1	0	0	50 V

BITS 5	4	CLPG
0	0	DC
0	1	AC
1	0	GND
1	1	CH2OFF

BIT 6	VAR
0	CAL'd
1	UNCAL'd

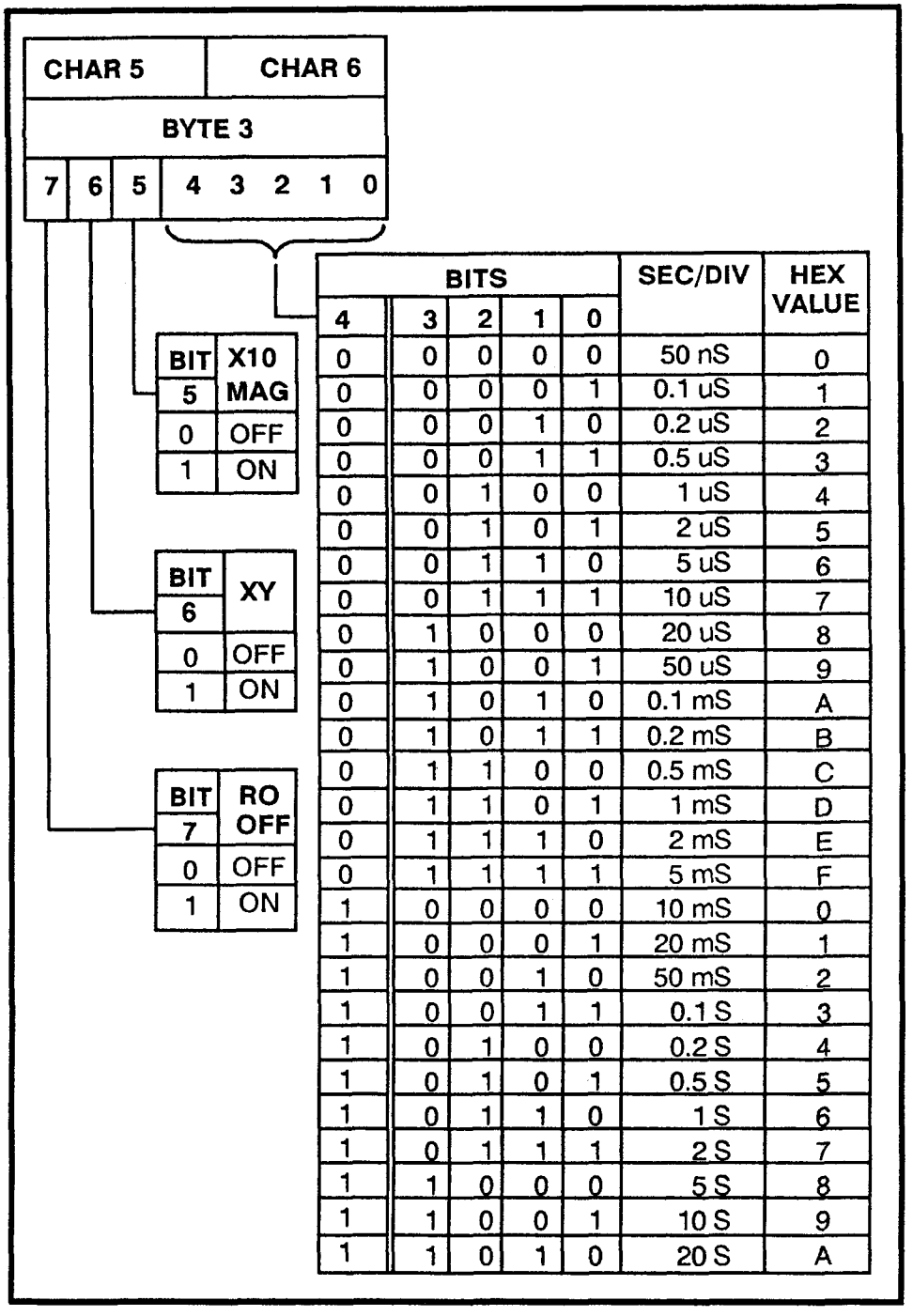
BIT 7	INVERT
0	OFF
1	ON

SEC/DIV SETTING. Characters 5 and 6 (Table 4) from the example string are 0 and C. In binary bits, these characters are 0000 and 1100 respectively. You can see from Table 4 that the bits of character 5 define several settings of the front panel. The example bit values of 0000 decode to show that the READOUT OFF menu choice is OFF (not selected, so the readout is on), XY display mode is OFF, and X10 MAG is OFF.

The fourth bit of character 5 is assigned to the SEC/DIV settings along with all four bits of character 6. A bit value of 0 for this bit defines SEC/DIV settings of 5 ms and faster. A bit value of 1 defines SEC/DIV settings of 10 ms and slower. The four bits of character 6 are all used to define the SEC/DIV setting. For the example then, the SEC/DIV setting is below 10 ms (0 value of bit 4 of character 5), and the 1100 (hex C) value of character 6 gives a SEC/DIV setting of 0.5 ms per division.

Then, to change the SEC/DIV setting to 0.1 ms for example, just change character 6 from C to D. However to change to 20 ms, the value of character 5 will also have to change, with it going from 0 to 1 in the setup example string while character 6 changes to a 1.

Table 4
SEC/DIV and Misc. Settings



BIT 5	X10 MAG
0	OFF
1	ON

BIT 6	XY
0	OFF
1	ON

BIT 7	RO OFF
0	OFF
1	ON

BITS					SEC/DIV	HEX VALUE
4	3	2	1	0		
0	0	0	0	0	50 nS	0
0	0	0	0	1	0.1 uS	1
0	0	0	1	0	0.2 uS	2
0	0	0	1	1	0.5 uS	3
0	0	1	0	0	1 uS	4
0	0	1	0	1	2 uS	5
0	0	1	1	0	5 uS	6
0	0	1	1	1	10 uS	7
0	1	0	0	0	20 uS	8
0	1	0	0	1	50 uS	9
0	1	0	1	0	0.1 mS	A
0	1	0	1	1	0.2 mS	B
0	1	1	0	0	0.5 mS	C
0	1	1	0	1	1 mS	D
0	1	1	1	0	2 mS	E
0	1	1	1	1	5 mS	F
1	0	0	0	0	10 mS	0
1	0	0	0	1	20 mS	1
1	0	0	1	0	50 mS	2
1	0	0	1	1	0.1 S	3
1	0	1	0	0	0.2 S	4
1	0	1	0	1	0.5 S	5
1	0	1	1	0	1 S	6
1	0	1	1	1	2 S	7
1	1	0	0	0	5 S	8
1	1	0	0	1	10 S	9
1	1	0	1	0	20 S	A

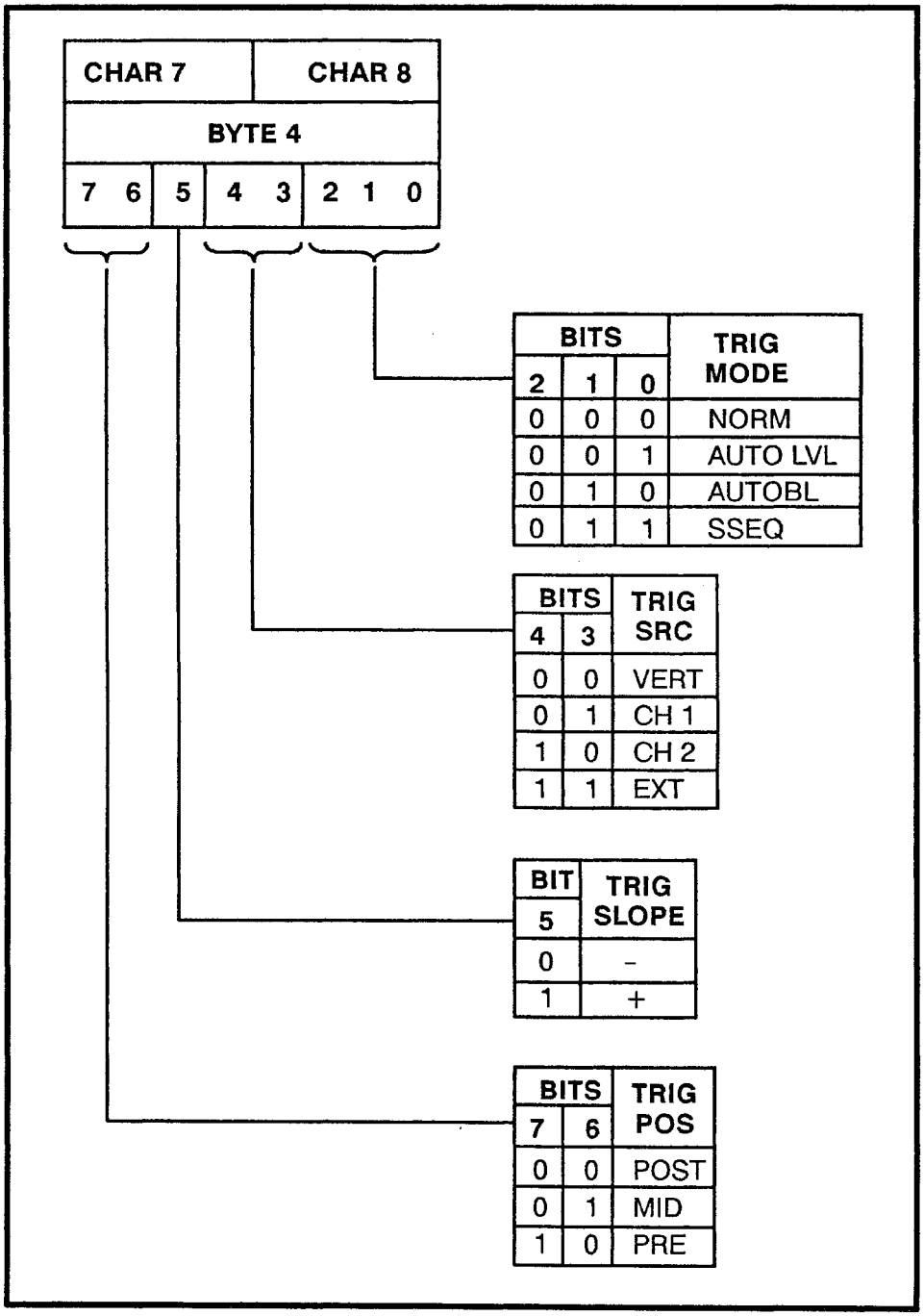
TRIG POS, SLOPE, SOURCE, AND MODE SETTINGS. Characters 7 and 8 (Table 5) of the front panel setup string define several of the trigger settings of the instrument. Two bit values of the two characters are used to define the Trigger SOURCE setting (bit 4 of character 7 and bit 1 of character 8). Therefore, both characters must be set correctly to control the Trigger SOURCE setting.

In our example front panel setup string, characters 7 and 8 are 2 and 1 respectively. The binary bits are therefore 0010 and 0001 for these two characters. The first two bits of character 7 define the TRIG POS setting; the bit values of 0 0 in these positions decode to a TRIG POS of POST. The third bit is a 1 and decodes to a + SLOPE setting.

The last bit of character 7 and the first bit of character 8 are 0 and 0 respectively and decode to VERT Trigger SOURCE. The last three bits of character 8 are 001 and decode to AUTO LVL Trigger MODE.

Table 5

Trigger Position, Slope, Source and Mode Settings



BITS			TRIG MODE
2	1	0	
0	0	0	NORM
0	0	1	AUTO LVL
0	1	0	AUTOBL
0	1	1	SSEQ

BITS		TRIG SRC
4	3	
0	0	VERT
0	1	CH 1
1	0	CH 2
1	1	EXT

BIT	TRIG SLOPE
5	
0	-
1	+

BITS		TRIG POS
7	6	
0	0	POST
0	1	MID
1	0	PRE

ACQUISITION MODE AND MISC SETTINGS. Characters 9 and 10 (Table 6) of the front panel setup string define the remaining settings not defined by the other 8 characters. Table 6 shows setup state controlled by each bit. The only 2-bit setting is for Acquisition MODE. From our example setup string, the last two characters are 1 and 2. The bit values for these are 0001 and 0010 respectively. The decoding for the bit values of character 9 is:

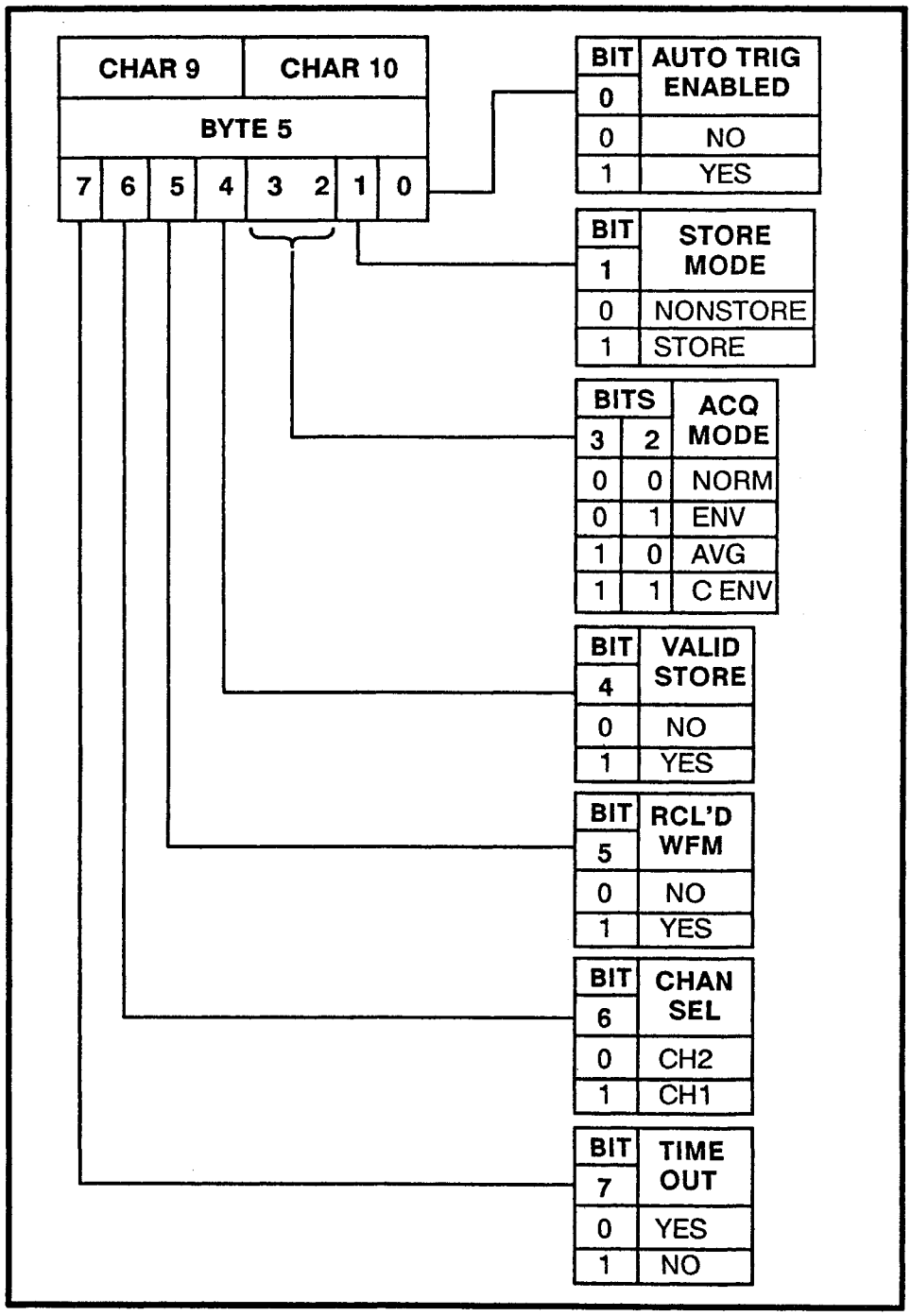
TIME OUT is enabled,
CH2 is selected to respond to the controls settings,
It is not a recalled waveform, and
It is a valid store.

Character 10 (0010) decodes as follows:

The Acquisition MODE (2 bits) is NORM,
It is in STORE mode, and
AUTO TRIGGER is not enabled.

Table 6

Acquisition Mode and Misc. Settings



RS-232 ERROR CODES

When the 222 is connected either to a terminal or to a PC in terminal mode, errors in commands or operation that are output via the RS-232 interface may be read. There are two types of error structure. Status messages are output whenever a command error is detected; diagnostic error codes are output if errors occur during normal operation or when running the calibration routine. If there is no errors, the 222 responds with "READY;" when it is ready to respond to commands.

Communication Error Codes

Status messages are returned when an error is detected in a command.

STAtus 0001	Unrecognized command
STAtus 0002	Unrecognized character
STAtus 0003	Command is query only
STAtus 0004	Command has no query
STAtus 0005	Bad command argument
STAtus 0006	Bad data
STAtus 0007	Data is required
STAtus 0008	Argument is required
STAtus 0009	Communication task is busy
STAtus 000A	CURV command had bad checksum
STAtus 000B	Bad task name for message
STAtus FFFF	User pressed escape

Diagnostic Error Codes

If an error is detected in the oscilloscope during normal operation or calibration, an error message is output to an external terminal via the RS-232 serial port. These error codes are formatted as follows:

ERROR wxyy zzzz	where wxyy and zzzz are 16-bit hexadecimal numbers representing the error message.

These code key is as follows:

w = the error type:

0 = error during normal calibration
2 = EEPROM programming error
4 = EEPROM calibration constant area error
8 = Calibration error
F = Fatal system error

x = the channel affected by the error:

1 = Channel 1
2 = Channel 2
0 = Channel not specified

yy = the error code. The value seen here depends on the type of error (0, 2, 4, 8, or F) at the w position in the portion of the first code group (wxyy) as follows:

Error type 0:

Error code 09 = Trigger search error (auto level mode).

Error type 2:

Error code XX = The data that failed to program. The value of the second code group (zzzz) is the address that failed to program.

Error type 4:

Error code 01 = Bad EEPROM checksum detected.

Error code 02 = Calibration needed. The following zzzz codes indicate which calibration routine needs to be done:

0001 = Channel 1 offset/gain calibration
0002 = Channel 2 offset/gain calibration
0004 = Channel 1 offset dac calibration
0008 = Channel 2 offset dac calibration
0010 = Channel 1 trigger calibration
0020 = Channel 2 trigger calibration
0040 = External trigger calibration
0080 = Clock delay calibration

If the zzzz error code is FFFF, no calibration routines have been done since all the default values were loaded into the EEPROM (this seen only at the first factory calibration).

Error type 8:

01 = Acquisition timeout error
02 = Mid position search error
03 = Mid position range error
04 = Offset search error
05 = Offset range error
06 = Offset gain error
07 = Gain range error
08 = Gain search error
09 = Trigger search error
10 = Trigger offset range error
11 = Trigger gain error
12 = Trigger hysteresis error
13 = External trigger offset range error
14 = External trigger hysteresis error
15 = Clock delay error
16 = Acquisition delay error

Error type F:

00 = COP timeout error
01 = Illegal opcode execution
02 = Interrupt exception
03 = Task exception
04 = CME error (not implemented)

zzzz = an additional 16-bit value the meaning of which depends on the first error word. Except for error type 2 and error type 4 where the zzzz codes are given, the zzzz values (though sent) have no useful meaning to the user.

Error Code Decoding Examples

Following are two error code examples of how wxyy and zzzz are decoded.

ERROR 8105 03FF (seen as the result of running the CH 1 GAIN self calibration via the serial interface)

w = 8 = Calibration error
x = 1 = Channel 1
yy = 05 = Offset range error
zzzz = 03FF = The offset DAC value (out of range limit)

ERROR 4002 0040 (seen on power up if the serial interface is connected)

w = 4 = EEPROM calibration constant area error
x = 0 = No specified channel
yy = 02 = Calibration needed
zzzz = 0040 = External trigger calibration

RS-232 COMMUNICATIONS

Waveform Transfers

Software Tools

There are several PC XT® and PC AT® compatible software utilities available to assist you in using the communications interface to transfer waveforms and control the instrument. A user may also develop programs written in a program language supported by their installed software (C is a popular language that is well documented). The commercial software packages provide different levels of capability and control. A brief summary is included here. Full documentation of the software is provided with the packages.

Available Software

CAT200©. This is a Tektonix, Inc. purchased software product. It provides a virtual 222 front panel (a graphical interface with mouse-input facilities) on the PC screen. Waveforms may be transferred either to or from a local 222 connected to the serial communications port of the PC or a remote 222 via a telephone modem. There are no capabilities for further processing of the captured waveform data or for automated control of the front panel under CAT200 programming.

LabWindows®. This a National Instruments, Inc. software product. LabWindows is similar to the CAT200 package, but it does not provide a virtual 222 front panel on the PC screen. Waveform data manipulation after transfer to the PC is part of this package plus automated operation of the front panel controls. There is no modem control in LabWindows so a user must connect to a remote 222 with a terminal communications program before using LabWindows® to obtain waveforms from and control the remote oscilloscope.

Demo Disk. This demonstration software disk permits control of the 222 front panel and capture and display of waveforms. The PC compatible program is provided free as a demonstration of the capabilities of the RS-232 communications interface. It is not a supported software product.

PC XT and PC AT are registered trademarks of International Business Machines Corporation.
LabWindows is a registered trademark of National Instruments, Inc.
CAT200 is a registered copyright of Tektronix, Inc.

Transfer to a Local PC

Transferring waveform data and controlling the front panel settings using a local PC is one type of system setup (see Figure 2). With a PC XT as the host computer, the optional RS-232 interconnection cable provides the required match from the 25 pin connector on the PC to the nine pin connector on the 222. The optional cable is also compatible with a PC AT that has a 25 pin communications port.

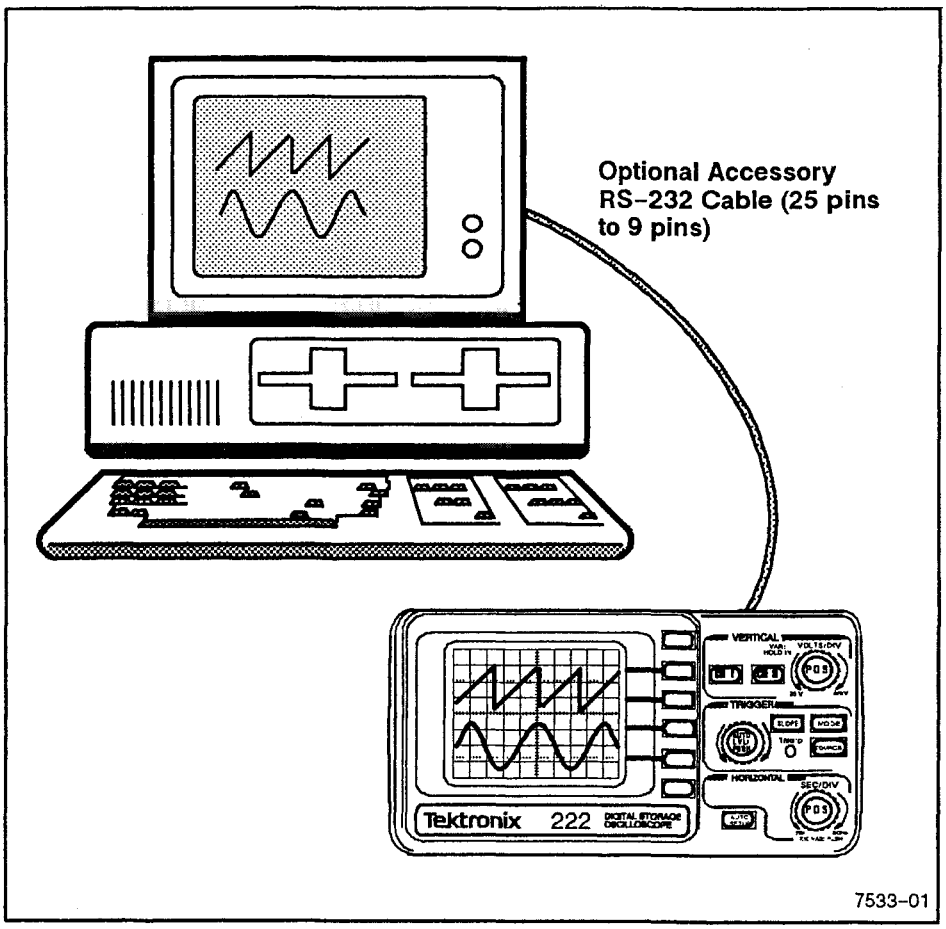


Figure 2. Local Control of a 222 from a PC.

Transfer via a Modem

Telephone lines may be used to control a remote 222 and transfer waveforms between the 222 and a controlling PC (see Figure 3). The software utility program you have will determine the operation needed for controlling the modem. If the software does not have a modem control routine, a terminal communications utility must first be used to make the modem connection. That utility must then be exited

to permit the 222 control program to function. The CAT200© software does have modem control built in. Once the software is installed on the PC and the actions needed at both ends of the link have been decided, the procedure to capture a waveform is, in general, as follows:

1. At the remote site, the operator must hook up the 222 to a modem and acquire the test waveforms to be transferred to the PC. In the operation described here, the modem connected to the remote site 222 must be set to the auto answer mode (see your modem manual for operating the modem). Waveforms may be stored in the 4 reference memories and acquired in both CH 1 and CH 2 vertical channels for transfer.

2. The PC operator now must call the remote site and establish the modem link. If the 222 control utility does not have modem control capabilities, the PC operator must first use a terminal communications utility (an example is Kermit) to establish the modem link. When the remote modem answers, the PC operator can exit the communications utility and start the 222 control utility (either by command or exiting the terminal communications utility without hanging up the modem).

3. At the PC, the operator in now in control of the 222 and may request waveforms from the 222. Waveforms may be transferred from the reference memories and from the two vertical channels. If further waveforms are needed, the remote site operator must move the probes to new test points or make other adjustments to gather new waveforms. The operator at the PC may store new waveforms into the reference memories or, if wanted, continually ask for new waveforms from CH 1 and CH 2 as new waveforms are acquired. The waveforms received can be filed for future study.

4. The 222 does not support user messages on-screen, so a second voice telephone connection between the two operators may be useful when a series of different waveforms needs to be transferred for use at the PC site. If the modems in use permit, VOICE/DATA switching may be done using the same phone line for both modes. The CAT200 software provides this utility. You may need to consult your modem/communications software manual for operating details.

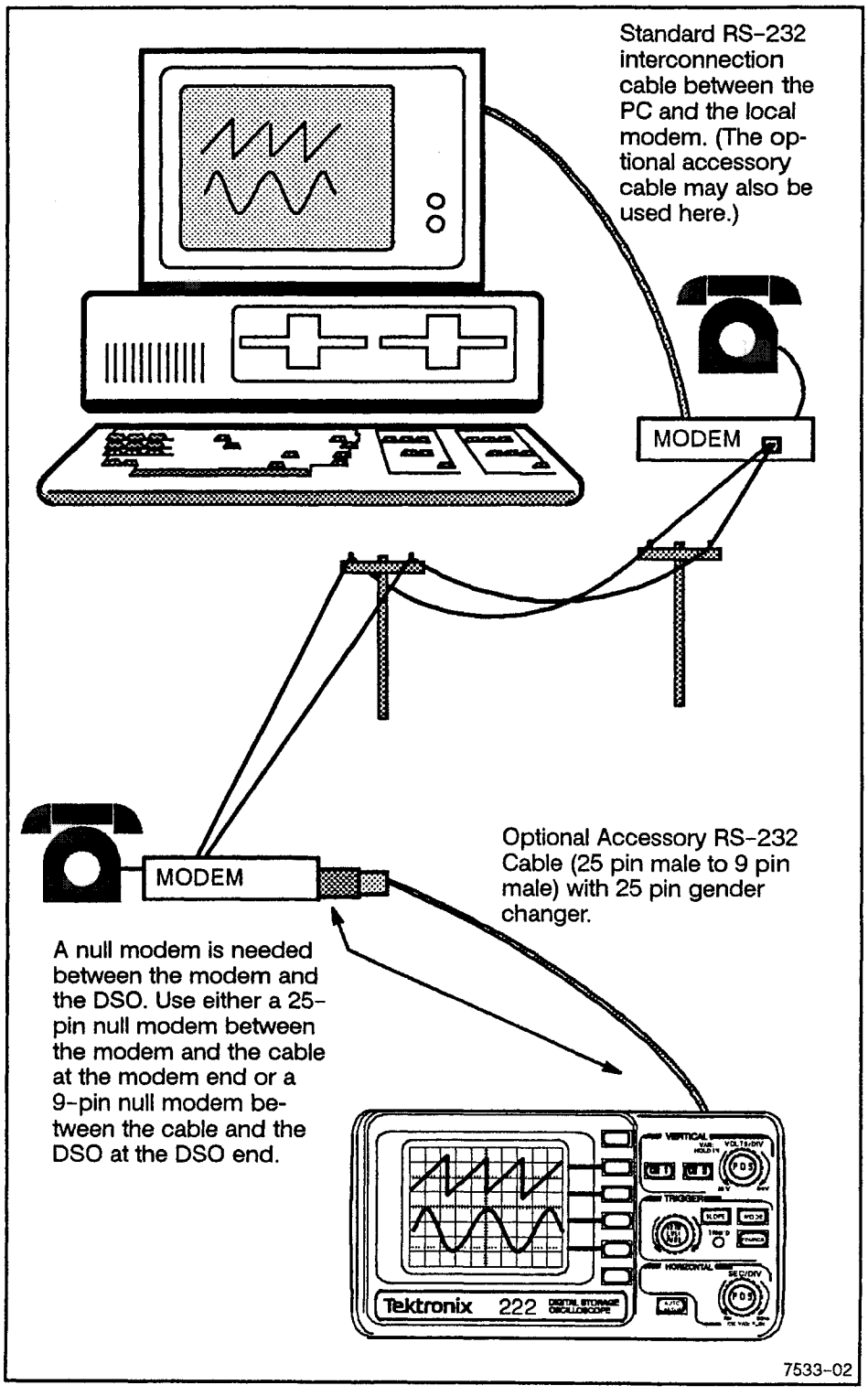


Figure 3. Remote control of a 222 from a PC via a modem.

RS-232 Interconnection Cables

The optional accessory RS-232 interconnection cable supports hook up of the 222 with its DB-9 connector to a PC XT® or compatible with a DB-25 connector (see Figure 4). For connection to other types of equipment (PC AT®, Macintosh®, or modem) a user must provide the correct cabling (see Table 7 for typical pin connections).

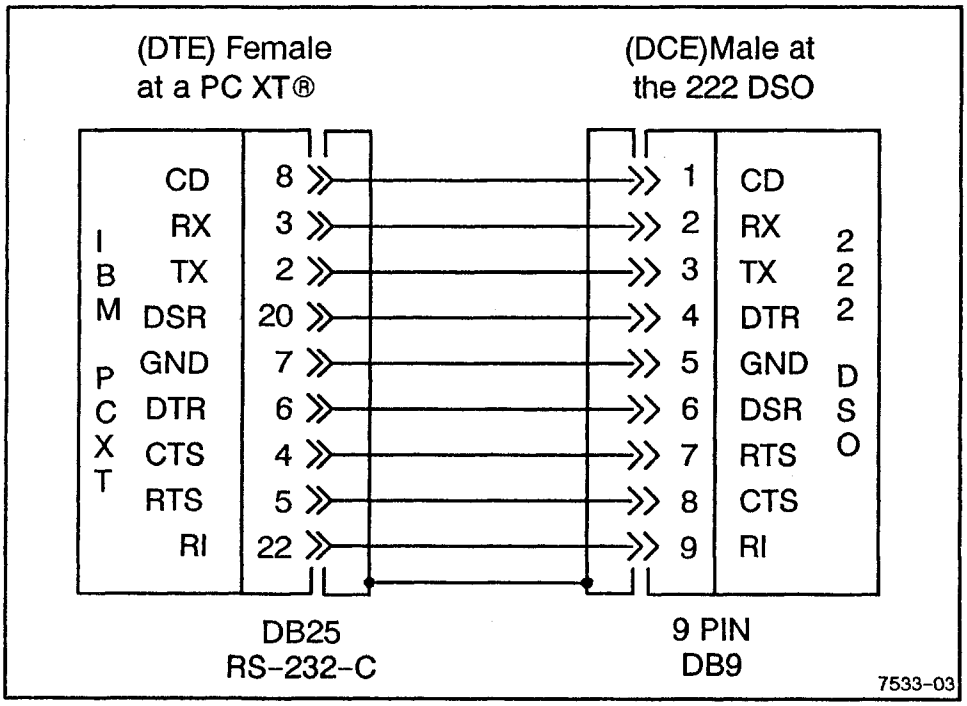


Figure 4. RS-232 interconnection cable wiring between a PC XT® or compatible and the 222. This is the optional accessory cable part numbered in this manual.

PC XT, PC AT, and IBM are registered trademarks of International Business Machines Corp. Macintosh is a registered trademark of Apple Computer, Inc.

Table 7
222 Interconnect Pin Assignments

	(DCE) Male at the 222 DSO		TO					
			(DCE) Modem		(DTE) PC XT		(DTE) PC AT	
*	1	CD	8	CD	8	CD	1	CD
	2	RX	2	TX	3	RX	2	RX
	3	TX	3	RX	2	TX	3	TX
	4	DTR	6	DSR	20	DTR	4	DTR
*	5	SGND	7	GND	7	GND	5	SGND
	6	DSR	20	DTR	6	DSR	6	DSR
*	7	RTS	5	CTS	4	RTS	7	RTS
	8	CTS	4	RTS	5	CTS	8	CTS
*	9	RI	22	RI	22	RI	9	RI

* Connection optional

INDEX

INDEX