MODEL 2411A

2MHz Arbitrary Waveform Generator

Operation Manual

P/N 810007-CD
Rev. B

This owner's manual was as current as possible when this product was manufactured. However, products are constantly being updated and improved. Because of this, some differences may occur between the description in this manual and the product you received.

TEGAM, INC.
TEN TEGAM WAY
GENEVA, OH 44041
TEL: (440) 466-6100
FAX: (440) 466-6110
www.tegam.com
Publication Date: March 2005

Warranty:

TEGAM, Inc. warrants this product to be free from defects in material and workmanship for a period of one year from the date of shipment. During this warranty period, if a product proves to be defective, TEGAM Inc., at its option, will either repair the defective product without charge for parts and labor, or exchange any product that proves to be defective.

TEGAM, Inc. warrants the calibration of this product for a period of 6 months from date of shipment. During this period, TEGAM, Inc. will recalibrate any product, which does not conform to the published accuracy specifications.

In order to exercise this warranty, TEGAM, Inc., must be notified of the defective product before the expiration of the warranty period. The customer shall be responsible for packaging and shipping the product to the designated TEGAM service center with shipping charges prepaid. TEGAM Inc. shall pay for the return of the product to the customer if the shipment is to a location within the country in which the TEGAM service center is located. The customer shall be responsible for paying all shipping, duties, taxes, and additional costs if the product is transported to any other locations. Repaired products are warranted for the remaining balance of the original warranty, or 90 days, whichever period is longer.

Warranty Limitations:

The TEGAM, Inc. warranty does not apply to defects resulting from unauthorized modification or misuse of the product or any part. This warranty does not apply to fuses, batteries, or damage to the instrument caused by battery leakage.

Statement of Calibration:

This instrument has been inspected and tested in accordance with specifications published by TEGAM Inc. The accuracy and calibration of this instrument are traceable to the National Institute of Standards and Technology through equipment, which is calibrated at planned intervals by comparison to certified standards maintained in the laboratories of TEGAM Inc.

Contact Information:

TEGAM INC.
10 TEGAM WAY
GENEVA, OHIO 44041
PH: 440.466.6100
FX: 440466.6110
EMAIL: sales@tegam.com

TABLE OF CONTENTS

SECTION 1 INTRODUCTION

SECTION 2 CONNECTING POWER

SECTION 3 QUICK START

SECTION 4 GENERATOR CONCEPT and CONTROL

SECTION 5 MEMORY ORGANIZATION

SECTION 6 CREATING and EDITING WAVESHAPES

SECTION 7 SEQUENCE GENERATOR

SECTION 8 MULTIPLE UNITS

SECTION 9 OTHER FEATURES

SECTION 10 RS-232C and GPIB (IEEE 488.2)

APPENDIX

SECTION 1
INTRODUCTION

1.1 PRODUCT SUMMARY

The 2411A provides an unlimited variety of signal source waveshapes and sequences. Twenty standard waveshapes are pre-programmed for instant recall. Arbitrary waveshapes can be downloaded from a computer or created locally using line or vertex programming. All standard and custom waveshapes are digitally synthesized, with 16 bits (65,535 points) of amplitude resolution and 64k of waveform memory. An accurate internal clock up to 2 MHz provides a wide range of output frequencies. The instrument is easy to operate, with an intuitive front panel and a menu-driven, easy-to-read 40-character backlit display. Parameter changes may be done conveniently using either a numeric keypad or a rotary edit knob. Two previously stored waveforms can be added, subtracted or multiplied together for special applications such as an amplitude-modulated signal. Diverse waveform sequences can easily be created by using a Sequence Generator option which permits different waveform segments to be repeated and/or linked in any order. All waveforms, as well as instrument setups, are completely non-volatile. An RS-232C remote interface is standard and GPIB (IEEE 488.2) is available as an option. The Model 2411A is housed in an extremely rugged extruded aluminum case sized small enough to fit comfortably on any bench. High-quality, state-of-the-art components assure the utmost reliability and performance.

1.2 KEY FEATURES

- 20 Standard Waveshapes
- 64K Waveform Memory
- 2 MHz Sample Clock
- 16 Bits Amplitude Resolution
- Waveform Creation and Editing (Mouse and Pad Optional)
- Add, Subtract and Multiply Waveforms
- Loop and Link Waveforms with Sequence Generator (Optional)
- Standard RS-232C Interface
- GPIB Interface (Optional)

1.3 SPECIFICATIONS

OUTPUT WAVEFORMS
Up to 100 High-Definition Custom Waveforms, Sine, Square, Triangle, ±Sawtooth, DC, ±Pulse, ±Exponential, AM, SCM, FM, Lin/Log Sweep, Noise, Sin x/x, Gaussian, Haversine, Circle.

SEQUENCE GENERATOR (Optional)
Waveform: Loop and Link
Repetitions: Loop 1,000,000; Link 100
Program: 1000 Steps

WAVEFORM RESOLUTION
Horizontal Resolution: 65,536 points.
Vertical Resolution: 16 bits; 65,535 points (+32,767; -32,768).

WAVEFORM SAMPLING RATE
Range: 0.1Hz to 2MHz (10s to 500ns).
Resolution: 4 digits
Accuracy: ±50ppm

WAVEFORM RISE/FALL TIME
Less than 150ns, tested with square wave, filter off, 10Vp-p, 50Ω termination.

SPECTRAL PURITY
THD + Noise: Typically below 85dB in 80kHz measurement bandwidth. Tested at 2MHz clock; sinewave; 1,000 points (2 kHz); filter on; full amplitude; 50Ω termination.

AMPLITUDE AND OFFSET

Range	Resolution	Accuracy
±1.00 to 10V	10mV	1% of setting +20mV
±100mV to 999mV	1mV	3% of setting +5mV
±10mV to 99.9mV	100μV	5% of setting +1mV

Note: 50Ω source impedance, measured at open circuit, tested with 1kHz sinewave plus DC offset.

ANALOG FILTERS
User-selectable low-pass; 700kHz, seventh-order Butterworth and 40kHz, third-order Chebyshev.

OPERATIONAL MODES
Continuous: Output runs continuously between selected memory address locations.
Triggered: Output at start point until triggered, then runs once.
Gated: As triggered except output is continuous until gate signal ends.
Toggled: Alternate triggers gate the output waveform.
Burst: Each trigger outputs a pre-programmed number of waveforms from 1 to 1,048,575.
Hold: Front-panel button or external signal stops waveform at present memory location while applied.
RTS: Front-panel button or external signal interrupts the output waveform and ramps the output level back to the start level.

OUTPUTS
Output: Front-panel main waveform output. 50Ω impedance.
Sync Output: Front-panel TTL sync output. Programmable address and width. 50Ω impedance.
Clock Out: Rear-panel ARB waveform x2 sample clock output (TTL).
Reference Out: Rear-panel internal 10-MHz reference output (TTL).
Sync Trigger Out: Rear-panel TTL sync for triggering additional units in parallel or series.
Z Axis Out: Rear-panel Z Axis output in edit mode.
Sync 3/Run Out/DAC Out: Rear-panel TTL output. Programmable sync, high when output signal on or direct DAC out.
Sync 4/End Block Out: Rear-panel TTL output. Programmable sync or single pulse at end of each memory block in continuous and triggered modes; single pulse after each group of cycles in gated and burst modes.

INPUTS

Sum In: Front-panel input allows external signal to be added to output. Gain = 2 open circuit and 1 into 50Ω. 50Ω input impedance.
Trigger Input: Rear-panel TTL trigger input for triggered, gated, toggled, and burst modes.
Clock In: Rear-panel ARB waveform sample clock input (TTL ≤4MHz, divide-by-two).
Reference In: Rear-panel 10-MHz reference input. The internal crystal-controlled oscillator will phase-lock to the input.
RTS In: Rear panel TTL input to initiate RTS mode.
Hold In: Rear panel TTL input to stop waveform at present level.

TRIGGER SOURCES

External Trigger Input
Manual Trigger
Internal Trigger Generator

CREATION TOOLS

Waveform Editing: Line Mode, Vertex Mode; Insert Function, Sum Function, Dump Function, Move, Digital Amplitude, Digital Offset
Waveform Math: A+B, A-B, AxB
Stored Settings: Waveforms, Setups

REMOTE INTERFACE

RS-232C: 19.2kBaud, max.
GPIB, (IEEE STD. 488.2-1987) (Optional)

ACCESSORIES (Optional)

Editing Mouse
Mouse Pad

GENERAL

Temperature Range: +23°C ±3°C for specified operation. Operates 0°C to +50°C. Storage -20°C to +60°C.
Dimensions: 11.5cm (4.53 in.) H; 25.8 cm (10.14 in.) W; 30cm (11.81 in.) D.
Weight: 5.0kg (11 lbs)
Power: 45VA; 40W (max) 100/120/220/240 VAC,
+5%, -10%; 48 to 63 Hz.

SECTION 2
CONNECTING POWER

2.1 GROUND CONNECTION

WARNING

To prevent death or injury from electrical shock, be sure the Model 2411A is connected to earth ground through an approv~~ed and inspected three-wire power cord~~.

2.2 SELECTING LINE (MAINS) VOLTAGE

CAUTION

Severe damage to the Model 2411A can occur if the rear-panel power switches are set to incorrect positions. Be sure to check these settings during initial installation.

2.2.1 Line Switch

The Line Switch (see Figure 2-1) selects one of two coarse power ranges. Set the switch to the 120V position for line (mains) voltages from 90 to 126 volts. Set the switch to the 240V position for voltages from 198 to 252 volts.

2.2.2 HI/LO Switch

The HI/LO Switch (see Figure 2-1) selects the fine power ranges. Set the switch to the HI position for line (mains) voltages from 108 to 126 or from 216 to 252 volts. Set the switch to the LO position for voltages from 90 to 105 or from 198 to 231 volts.

2.2.3 Fuses

A line (mains) fuse is located in the power cord connector module (see Figure 2-1). Replacement fuses must be 5 x 20mm, type GDC slow blow. Use a 0.8A fuse for the nominal 120V range or a 0.4A fuse for the nominal 240V range.

CAUTION

To provide proper instrument protection, be sure the correct size fuse is installed.

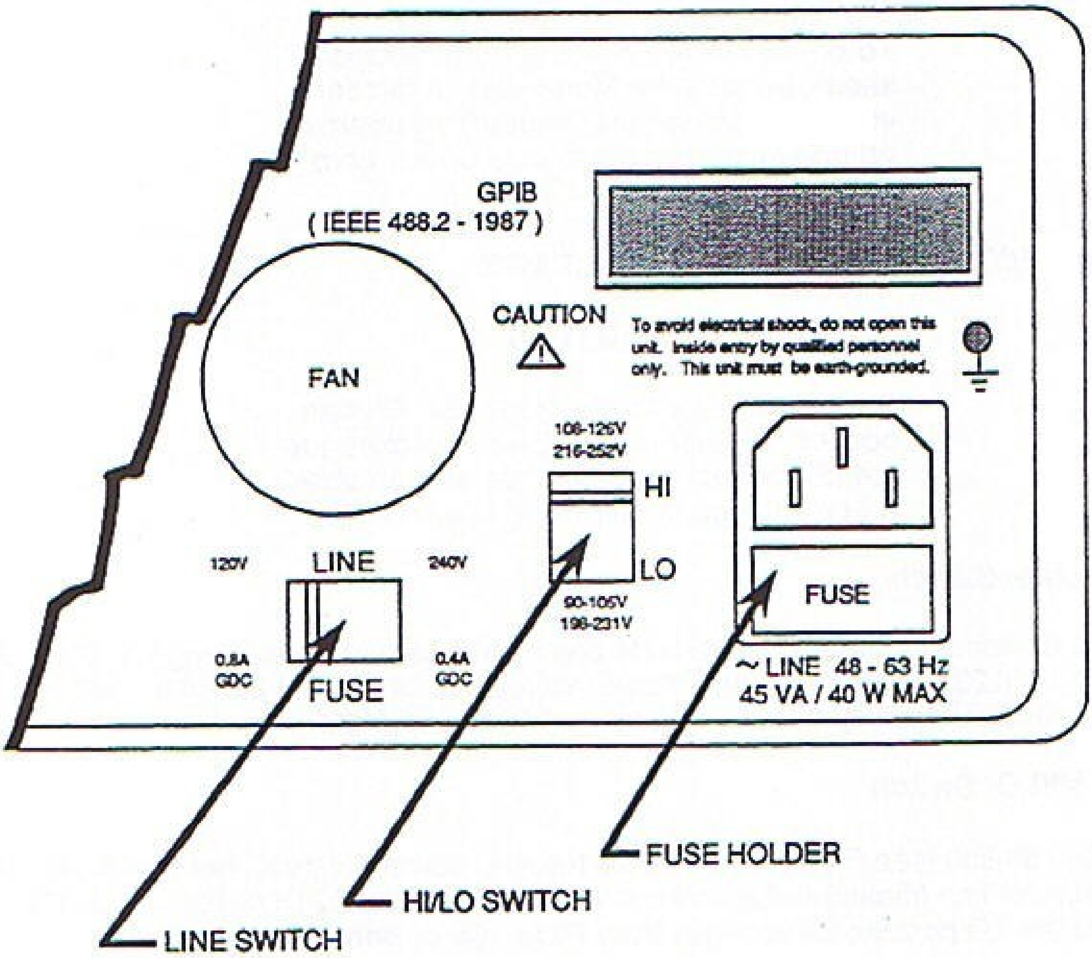


Figure 2-1. Power Switch and Fuse Locations

SECTION 3
QUICK START

3.1 INTRODUCTION

This section explains how to connect the Model 2411A and how to quickly obtain standard waveforms. Also included are instructions on how to set amplitude, offset, and frequency. The six basic operational modes are explained. More detailed operating procedures, including construction of arbitrary waveforms, are contained in the following sections of this manual. See Section 6 for arbitrary waveforms.

3.2 CONNECTIONS

All waveforms are obtained from the OUTPUT BNC connector on the front panel. A TTL sync pulse is available from the SYNC OUT connector on the front panel. Proper operation of the Model 2411A can be verified by connecting an oscilloscope as shown in Figure 3-1. The optional editing mouse should be connected to the serial interface (RS-232C) connector on the rear panel. *(NOTE: Connect the mouse only with the power OFF.)*

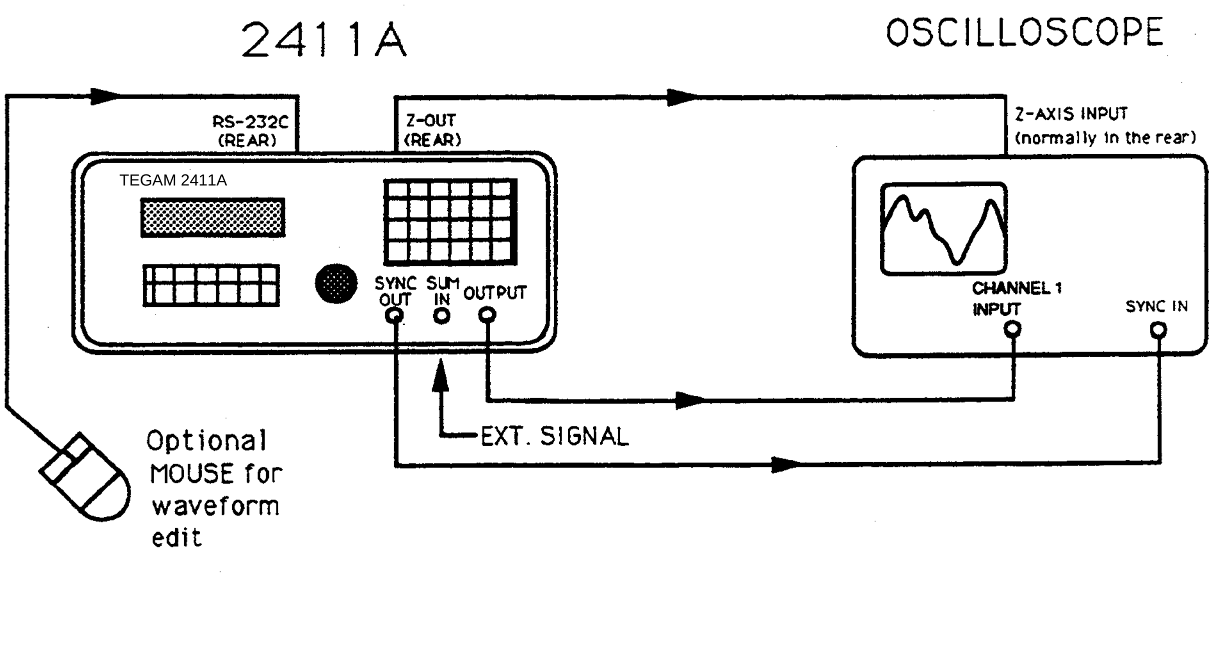


Figure 3-1. Output and Sync Connections

3.3 DEFAULT SETTINGS

The Model 2411A automatically provides a continuous sine wave signal after the instrument is reset. To reset the 2411A:

1) Press the SHIFT key (green LED will come on).
2) Press RESET (9) key.
3) Press CURR or ALL softkey. (CURR resets all the current settings to the values listed below. ALL additionally resets all waveform and sequence programming and stored settings.)
4) Press OK softkey to reset or CANC softkey to cancel.
5) Press SHIFT key again to restore normal key functions (green LED will go out).

Reset Default Values

Function:	Sine
Mode:	Continuous
Amplitude:	5.000 Volts
Offset:	0.000 V
Clock:	1 MHz
Output:	Off

3.4 HOW TO CHANGE DEFAULT PARAMETERS

The following paragraphs explain how to select waveforms and modes and set different values for amplitude, offset and frequency.

3.4.1 Turning Output On

Resetting turns the signal output off. To turn the output on:

1) Press the OUTPUT key.
2) Press the ON softkey.
3) Press the ENTER key. Output LED illuminates.

3.4.2 Selecting Standard Waveforms

All waveforms are digitally synthesized. Twenty standard waveforms can be readily recalled from stored algorithms:

1) Press FUNC key repeatedly until desired waveform is visible.
2) Select desired waveform with softkey (F1 through F4). Selected item is all capital letters.
2) Press ENTER key.

Several standard waveforms have ancillary functions; such as phase, number of cycles, duty cycle, rise and fall times, etc. These are listed in Table 3-1.

	ANCILLARY	RANGE &	DEFAULT
FUNCTION	FUNCTION	RESOLUTION	VALUE
SINE			
	Phase	0.000 to 360.000°	0.000°
	Number	0.01 to 1000.00	1.00
SQUARE			
	Number	1 to 1000	1
	Duty Cycle	1 to 100%	50%
TRIANGLE			
	Number	1 to 1000	1
SAWTOOTH			
	Number	1 to 1000	1
	Duty Cycle	1 to 100%	100%
DC			
	Digital Offset	-32768 to +32767	0
EXPONENTIAL			
	Time Constant	0.01 to 20.00	5.00
AM			
	Carrier Frequency	x1 to x10,000	x20
	Modulation Frequency	x1 to x10,000	x1
	Modulation Index	0 to 200%	100%
	Modulation Phase	0 to 360°	0°
	Carrier Phase	0 to 360°	0°
SCM			
	Carrier Frequency	x1 to x10,000	x20
	Modulation Frequency	x1 to x10,000	x1
	Modulation Phase	0 to 360°	0°
	Carrier Phase	0 to 360°	0°
FM			
	Carrier Frequency	x1 to x10,000	x20
	Modulation Frequency	x1 to x10,000	x1
	Modulation Index	0.01 to 100.00	10.00
	Modulation Phase	0 to 360°	0°
	Carrier Phase	0 to 360°	0°
HAVERSINE			
	Number	0.01 to 1000.00	1.00
LINEAR SWEEP			
	Begin	x1 to x1000	x1
	End	x1 to x1000	x10
LOG SWEEP			
	Begin	x1 to x1000	x1
	End	x1 to x1000	x10
PULSE			
	Delay	0 to 100%	0%
	Rise Time	0 to 100%	10%
	High Time	0 to 100%	30%
	Fall Time	0 to 100%	10%
	Number	1 to 1000	1
GAUSSIAN			
	Exponent Power	0.01 to 20.00	2.00
SINE X/X			
	Number	4.00 to 1000.00	5.50
CIRCLE			
	Number	0.01 to 1000.00	1.00
	Phase	0.01 to 360.00°	0.00

(NOTE: Typically 3 to 10 samples are required to represent the tabulated functions Therefore, the length of the waveform must be taken into consideration when selecting range and resolution values.)

Table 3-1. Model 2411A Ancillary Functions

3.4.3 Amplitude and Offset

Output signal amplitude can be changed from the default value of 5 volts peak (open circuit) to any value between 10mV and 10.20V (within limits of resolution, see Specifications on page 1-2). Either the edit knob or the numeric keypad may be used to obtain the new value. (For maximum resolution, use the keypad.) Press the AMPL/OFST key to display the value.

NOTE

> The edit knob and keypad control only the parameter displayed on the top row of the LCD. Use the double-arrow key to reverse the position of the displayed parameters.

DC offset can be changed from the default value of zero to any value between ±10.2 V.

CAUTION

> To prevent waveform clipping, the combined amplitude and offset must not exceed ±10.2V.

3.4.4 Sample Clock and Output Frequency

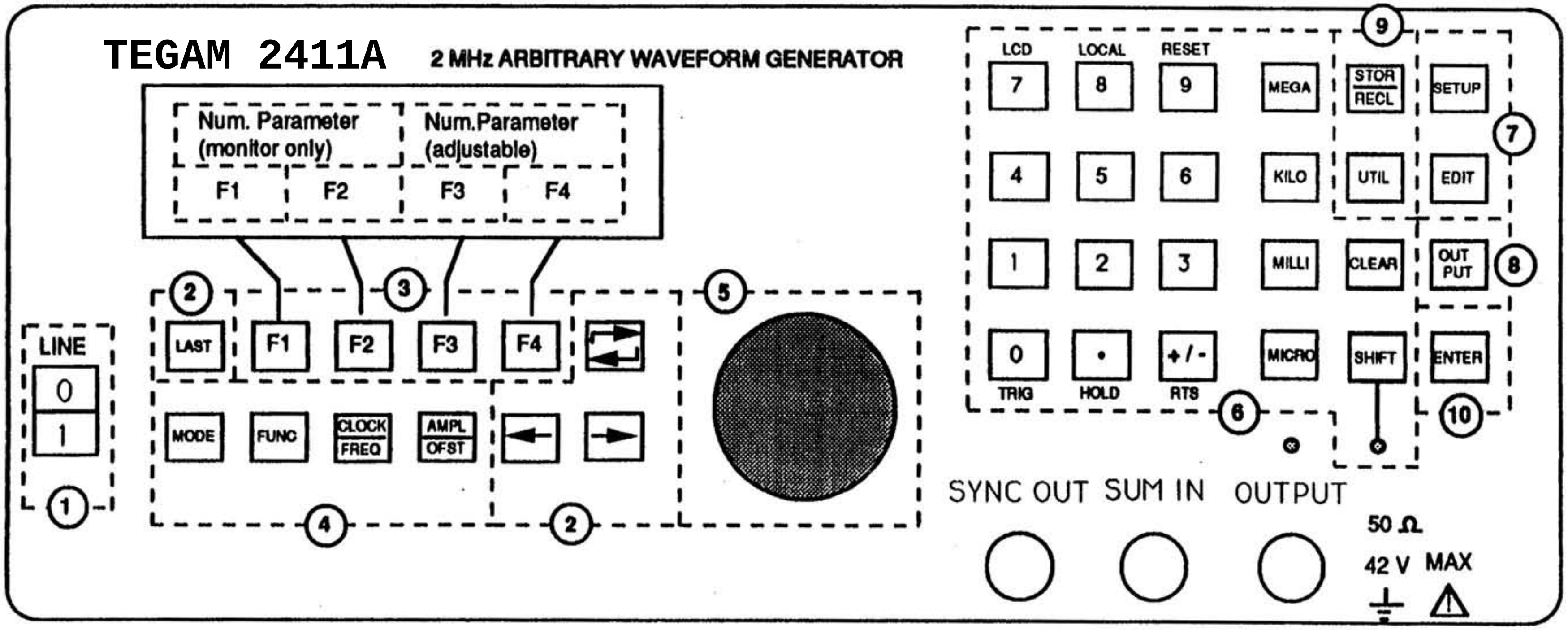
Output waveform frequency is a function of both the clock frequency and the number of samples. (Output f = Clock f/# Samples.) Since the default clock frequency is 1 MHz and the standard waveform default memory allocation is 1000 samples, the default output frequency is 1 MHz/1000 = 1 kHz.

Press the CLOCK/FREQ key. Use the double-arrow key to move SCLK to the top row of the display. Use the edit knob or keypad to set the clock frequency to any value from 0.1 Hz to 2 MHz. Note the new output frequency is computed and displayed automatically. It is not possible to set the output frequency directly. (Any change in the number of waveform samples will also affect the output frequency, as discussed in Section 5.)

3.4.5 Modes

Modes are selected by pressing the MODE key. Press it successively to display all five modes. Use the softkeys (F1 thru F4) to select a mode, then press ENTER. Each mode performs as follows:

Continuous: Output runs continuously between selected memory address locations.
Triggered: Output at start point until triggered, then runs once.
Gated: As triggered except output is continuous until gate signal ends.
Burst: Each trigger outputs a pre-programmed number of waveforms from 1 to 1,048,575.
Toggled: Alternate triggers start and stop the output waveform.



(1) Power Switch

(2) LCD Screen Control Keys

(3) Soft Keys

(4) Waveform Parameter Keys

(5) Edit Knob

(6) Numeric Keypad

(7) Waveform Setup and Edit Keys

(8) Output Control Key

(9) Store and Recall Key

(10) Enter Key

Figure 3-2. Front-Panel Controls and Indicators

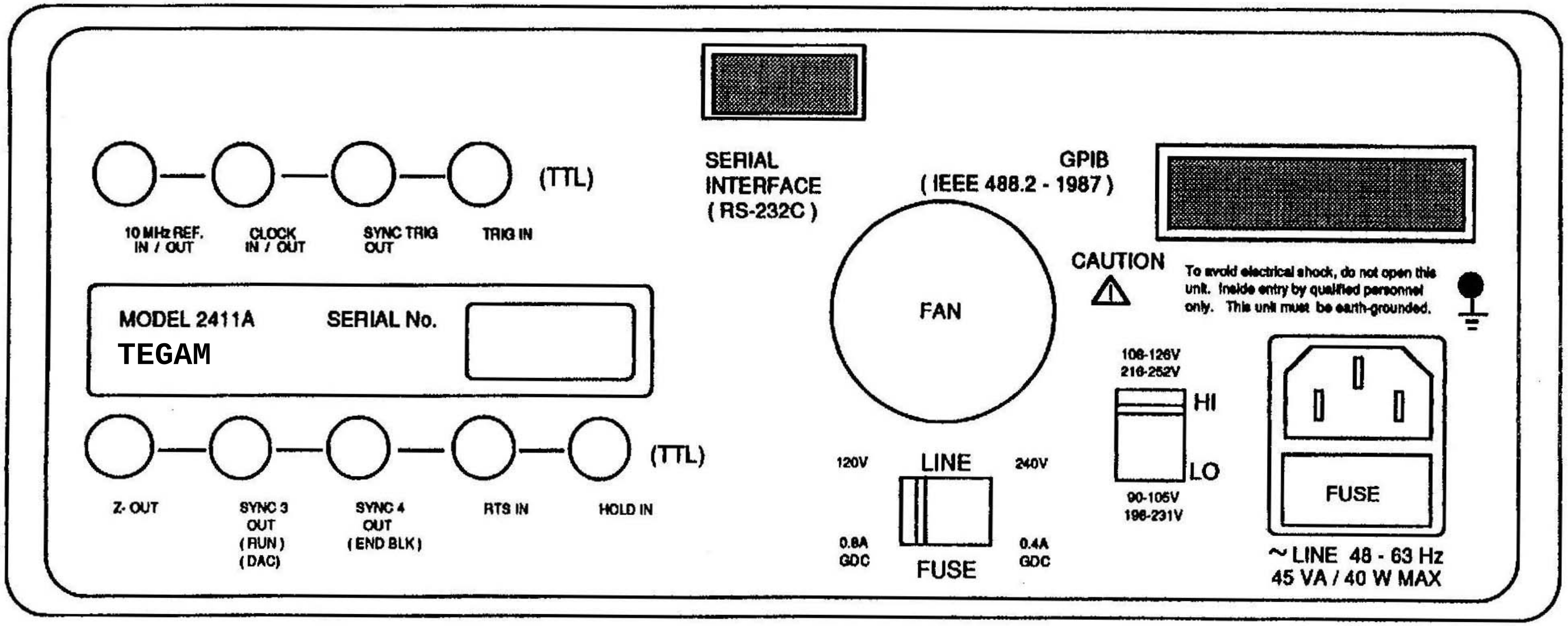


Figure 3-3. Rear-panel Connectors and Switches

SECTION 4
GENERATOR CONCEPT & CONTROL

4.1 INTRODUCTION

This section explains how waveforms are digitally synthesized, how the clocking system works and how waveforms are played back.

4.2 DIGITAL SYNTHESIS

Both standard and arbitrary waveshapes are created by digitally storing X and Y values in random-access memory (RAM). Standard waveshapes have their X and Y values loaded automatically from stored algorithms. Arbitrary waveshapes are created by downloading X and Y values from a computer or by using the line or vertex edit mode. The line and vertex edit modes allow the X and Y values to be entered manually using the edit knob and/or numeric keypad. Waveform creation can also be accomplished with an optional computer mouse connected to the back of the instrument to position and store X and Y intersects.

Figure 4-1 shows how a waveform is described in RAM memory with a series of X-Y values.

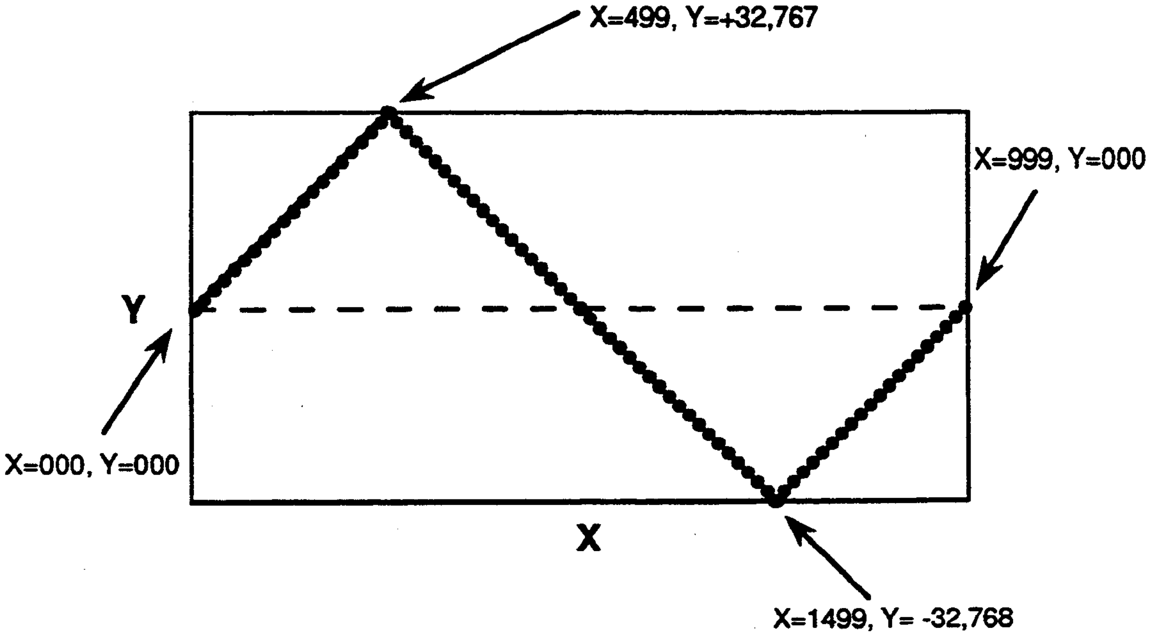


Figure 4-1. Digital Waveform Synthesis

4.3 CLOCKING SYSTEM

Analog waveshapes are created from the digitally-stored values by clocking the contents of the waveform memory through a digital-to-analog converter. The clock is derived from an internal synthesizer with an accuracy of ±50ppm. The clock frequency range is 0.1Hz to 2MHz with a resolution of 4 digits.

Figure 4-2 shows how the clock frequency, together with the number of samples, determines the output waveform frequency. (Output f = Clock f/# Samples). Clock frequency and number of samples should be chosen carefully to ensure the highest practical number of samples and, therefore, the best waveform resolution (detail).

Sample Clock and Waveform Frequency

$$\boxed{f_o = \frac{f_{sc}}{n_s}}$$

f_o : output waveform frequency

f_{sc} : sample clock frequency

n_s : number of samples in the selected waveform

Waveform frequency is dependent on both sample clock rate and number of samples.

Example

$f_{sc} = 2$ MHz

$n_s = 32$ samples

$$f_o = \frac{2\text{MHz}}{32} = 62.5\text{ kHz}$$

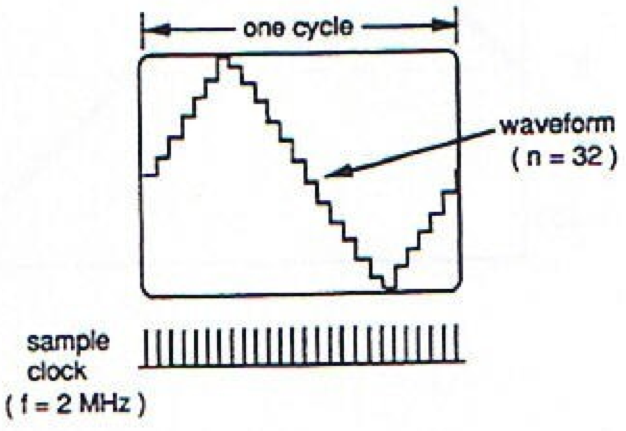


Figure 4-2. Clocking System

4.4 HOW WAVEFORM IS PLAYED BACK

Figure 4-3 shows how the waveform is played back. The digital waveform data stored in the waveform memory is clocked through a D/A converter to create the analog representation of the signal. One of two low-pass filters can be switched in to remove the sampling noise.

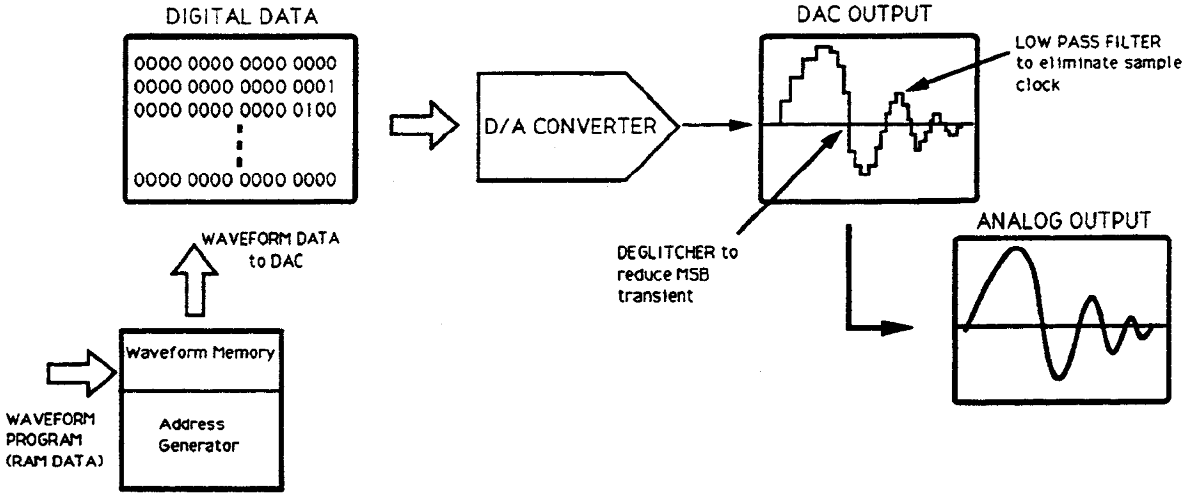


Figure 4-3. Waveform Playback

SECTION 5
MEMORY ORGANIZATION

5.1 INTRODUCTION

This section explains how the waveform memory is organized, the system of waveform numbering, how default memory segments can be changed, and how standard functions can be inserted.

5.2 DEFAULT PARTITIONING

The total available waveform memory is 65,536 points. The memory is initially divided into partitions to provide easy programming of 6 different waveforms. Five blocks of 2000 points each occupy the first 10K of memory. In addition, one of 20 standard waveforms can be readily recalled from the Function menu and downloaded to 1000 points of memory from 10K to 11K. (A sinewave is the default Standard Wave.)

5.3 WAVEFORM NUMBERING

The first 5 blocks of memory are assigned Waveform Numbers. Additional Waveform Numbers can be assigned (up to total of 100, depending on the number of points). Waveform Numbers are used to access the memory blocks for initial waveform programming and to recall the waveforms later. Waveform Numbers are arrayed in the memory in ascending order. Figure 5-1 shows the default waveform numbers and partitioning.

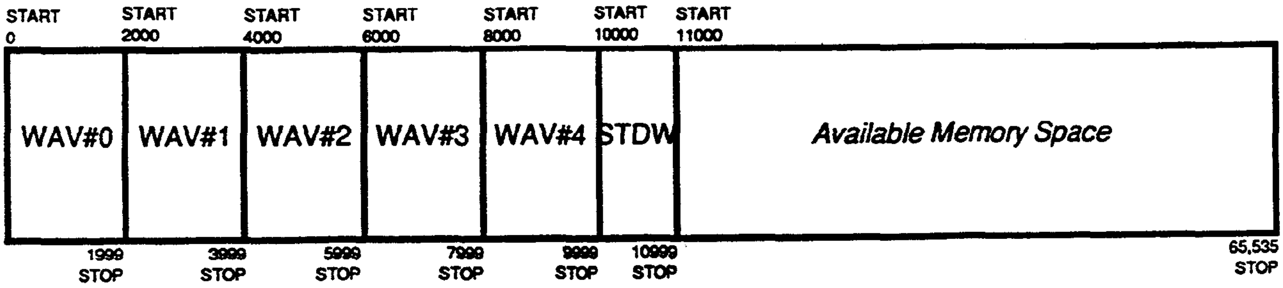


Figure 5-1. Waveform Numbers and Partitions

5.4 CHANGING WAVEFORM BLOCK LENGTHS

5.4.1 Standard Wave

The standard wave memory can be changed from its default value of 1000 points. (Minimum waveform length is 32 points.)

1) Press SETUP key.
2) Press STDW softkey. The display indicates the present length of the standard wave and the address where it begins in memory.
3) Press LEN softkey. The display indicates the available free memory.
4) Select desired standard wave length with edit knob or keypad. (Do not attempt to exceed the amount of memory available.)
5) Press OK to enter change or CANC to cancel.
6) Press ENTER.

5.4.2 Waveform Number Block Lengths

The waveform lengths of the numbered blocks can be changed from their default values of 2000 points. (Minimum waveform length is 32 points.)

1) Press SETUP key.
2) Press WAV# softkey.
3) Select Waveform Number with edit knob. The display indicates the present length and start address of the waveform.
4) Press LEN softkey. The display indicates the available free memory.
5) Select desired waveform block length with edit knob or keypad. (Do not attempt to exceed the amount of memory available.)
6) Press OK to enter change or CANC to cancel.

5.5 DELETING WAVEFORMS

A waveform which is no longer needed can be deleted from memory as follows:

1) Press SETUP key.
2) Press WAV# softkey.
3) Select Waveform Number with edit knob.
4) Press DEL softkey.
5) Press OK to delete or CANC to cancel.

5.6 INSERTING NEW WAVEFORM NUMBERS

Unused Waveform Numbers can be activated as follows:

1) Press SETUP key.
2) Press WAV# softkey.
3) Press NEW softkey.
4) Screen will show all unused Waveform Numbers when edit knob is turned. Select desired number.
5) Move waveform length to right side of display with double-arrow key and set desired length with edit knob or keypad.
   (New waveform cannot be longer than available free memory. See 5.4.2.)
6) Press OK to insert new Waveform Number or CANC to cancel.

SECTION 6
CREATING AND EDITING WAVESHAPES

6.1 INTRODUCTION

This section explains how to create arbitrary, non-standard waveshapes. These custom waveforms can be created a segment at a time using line or vertex edit modes. Each step in the waveform construction may be viewed on an oscilloscope connected to the instrument output. The optional mouse is recommended for line and vertex editing, although it is possible to construct waveforms without it. Arbitrary and standard waveforms can be interspersed. More complex waveforms can be created by adding, subtracting, and multiplying any two standard or arbitrary waveforms that have been previously stored in memory. It is also possible to download waveforms from a computer using the standard RS-232C or optional GPIB interface.

NOTE

> Define Waveform Number block length before creating an arbitrary waveshape (see paragraph 5.4.2).

6.2 LINE MODE

With the line editing mode, waveforms are created a segment at a time from a left-hand start or anchor point. A line is drawn from the start point or anchor to a "vertex" point which is positioned to the right. The vertex becomes a new anchor and the process is interactively repeated until the new arbitrary waveform is completed. This process is illustrated in Figure 6-1. All or any portion of a selected waveform block can be edited. Editing begins at the start point. A start point can be placed at any X and Y position within the selected waveform block. Alternatively, a left anchor can be positioned at any X address, but the Y value follows that of any previously programmed waveform (or baseline if no waveform was programmed).

6.2.1 Editing From Start Point

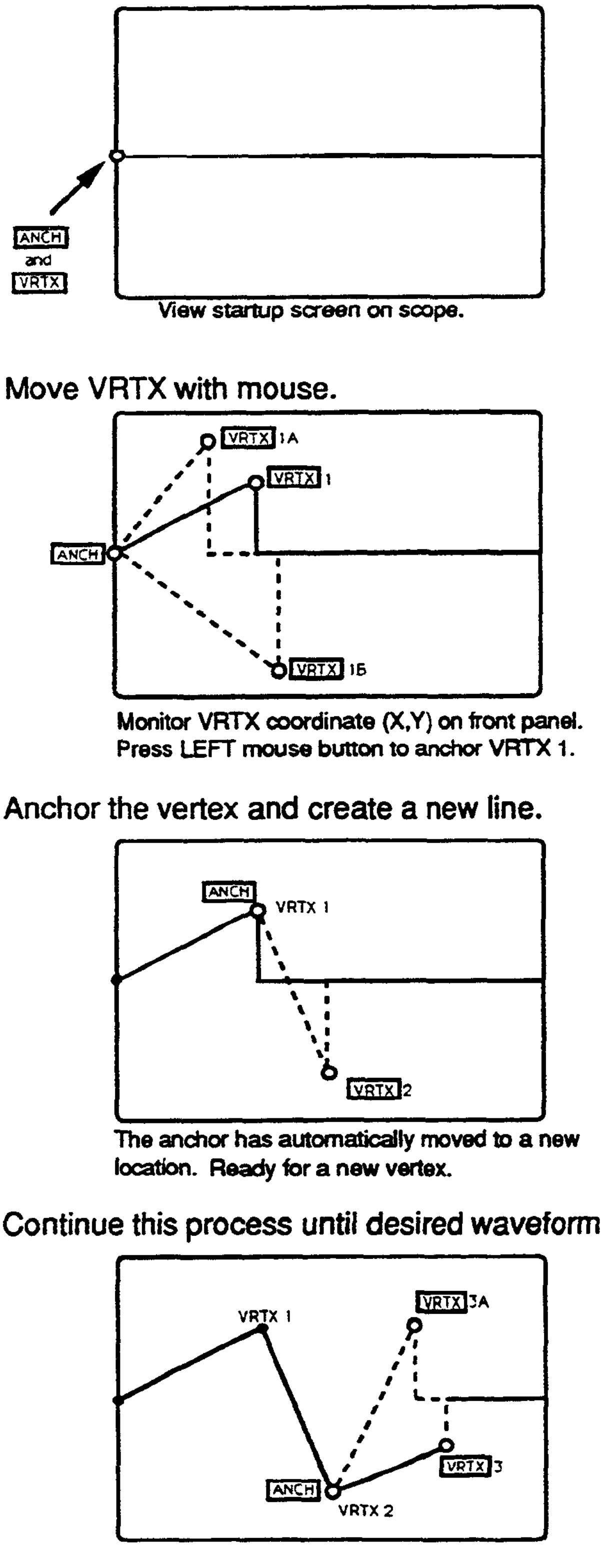
1) Press EDIT key.
2) Press LINE softkey.
3) Select Waveform Number using edit knob or keypad (press ENTER after using keypad).
4) Arbitrary waveform construction can begin at any point within the selected waveform number block. Press STRT softkey.
5) Use edit knob or keypad to select X and Y addresses to start waveform editing within the selected block. Use double-arrow key to move each active parameter to right side of display. (Remember to press ENTER if keypad is used.)
6) Press OK to store start point or CANC to cancel.
7) Press ANCH softkey.
8) Use double-arrow key to move Right Anchor (AR) to right side of LCD. Use edit knob or keypad to select X address for the end of the edited portion of the waveform.
9) Press OK to store right anchor or CANC to cancel.

6.2.2 Editing From Left Anchor

1) Press EDIT key.
2) Press LINE softkey.
3) Select Waveform Number using edit knob or keypad (press ENTER after using keypad).
4) Arbitrary waveform construction can begin at any X address within the selected waveform number block. Press ANCH softkey. *(If the Start Point is to be uniquely defined, see paragraph 6.2.1.)*
5) Use double-arrow key to move Left Anchor (AL) to right side of display. Use edit knob or keypad to select X address. (Remember to press ENTER if keypad is used.)
6) Use double-arrow key to move Right Anchor (AR) to right side of LCD. Use edit knob or keypad to select X address for the end of the edited portion of the waveform.
7) Press OK to store anchors or CANC to cancel.

NOTE

Anchors and vertexes will appear on the oscilloscope screen as intensified points on the edited output waveform. Connect a BNC/BNC cable between the Z-OUT connector on the 2411A rear panel and the Z-AXIS input connector of the oscilloscope. Adjust the oscilloscope intensity for good cursor definition. (If necessary, refer to paragraph 9.3 to adjust the Z-Axis level.)



1. Select LINE mode.

View startup screen on scope.

2. Move VRTX with mouse.

Monitor VRTX coordinate (X,Y) on front panel.
Press LEFT mouse button to anchor VRTX 1.

3. Anchor the vertex and create a new line.

The anchor has automatically moved to a new location. Ready for a new vertex.

4. Continue this process until desired waveform is created.

Figure 6-1. Line Mode Waveform Creation and Editing

6.2.3 Creating Line Segments

1) Press CHRD (chord) softkey. (Set anchors first per paragraph 6.2.2.)
2) Use edit knob, keypad or optional mouse to select X and Y addresses for the destination of the first line segment (chord). If the mouse is used, LCD readouts will continually indicate mouse position.
3) When the desired position is reached, press OK (or the left mouse button) and the line segment will be stored. Press CANC (or the right mouse button) to cancel.
4) Create the next line segment by again using the knob, keypad or optional mouse to set the next X-Y coordinate, as before.
5) Continue adding line segments up to the limit established in 6.2.1 step 5.

6.3 VERTEX MODE

With the vertex editing mode, waveforms are created by establishing two anchor points at selected addresses, positioning a vertex in the active region between the two anchors and then connecting the vertex to the anchors with two line segments. The vertex mode also permits waveform scaling and the insertion of standard functions. Vertex editing is illustrated in Figure 6-2.

1) Press EDIT key.
2) Press VRTX softkey.
3) Select Waveform Number using edit knob or keypad (press ENTER after using keypad).

6.3.1 Selecting Left and Right Anchor Points

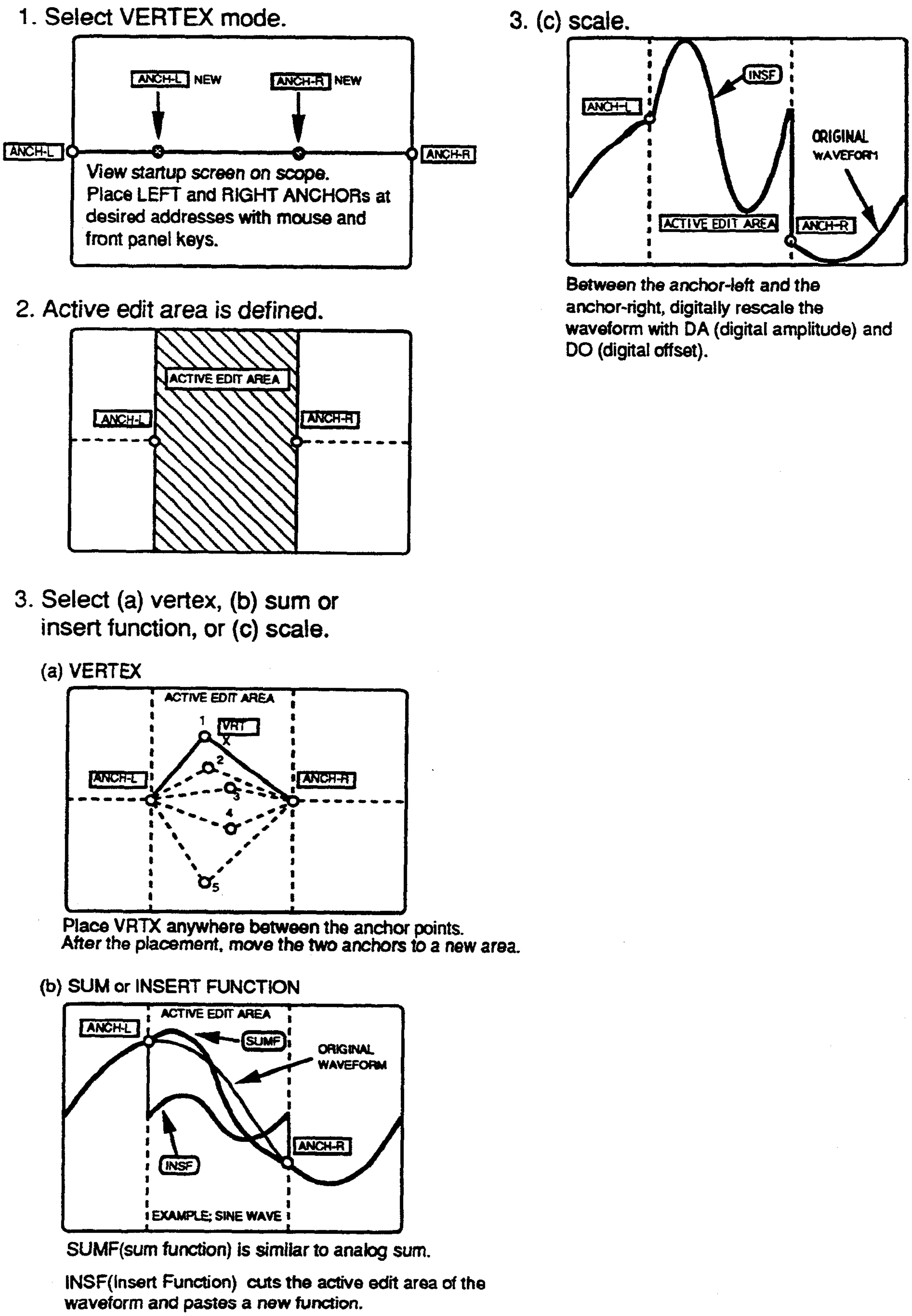
1) Press ANCH softkey.
2) Set left anchor (AL) X value using edit knob, keypad or optional mouse (press ENTER if keypad is used).
3) Use double-arrow key to move right anchor (AR) to right side of LCD. Set right anchor X value using edit knob, keypad or optional mouse.
4) Press OK to store anchors or CANC to cancel.

NOTE

The difference between the left and right anchors is limited to 8000 points or the waveform length, whichever is less.

6.3.2 Selecting Vertex Point

1) Press ADDV softkey.
2) Set vertex X and Y values using edit knob, keypad or optional mouse.
3) When the desired position is reached, press OK (or the left mouse button) and the two line segments will be stored. Press CANC (or the right mouse button) to cancel.
4) Continue adding anchors and vertices until the waveform is completed.



1. Select VERTEX mode.

View startup screen on scope. Place LEFT and RIGHT ANCHORs at desired addresses with mouse and front panel keys.

2. Active edit area is defined.

3. Select (a) vertex, (b) sum or insert function, or (c) scale.

(a) VERTEX

Place VRTX anywhere between the anchor points. After the placement, move the two anchors to a new area.

(b) SUM or INSERT FUNCTION

SUMF(sum function) is similar to analog sum.

INSF(Insert Function) cuts the active edit area of the waveform and pastes a new function.

3. (c) scale.

Between the anchor-left and the anchor-right, digitally rescale the waveform with DA (digital amplitude) and DO (digital offset).

Figure 6-2. Vertex Mode Waveform Creation and Editing

6.3.3 Scaling

Scaling allows any portion of a waveform designated by the left and right anchors to be scaled in amplitude and offset.

1) Select anchors as in 6.3.1.
2) Press SCAL softkey.
3) Set digital amplitude (DA) and digital offset (DO) values for selected portion of waveform using edit knob, keypad or optional mouse. Observe changes on output oscilloscope. Use double-arrow key to select active parameter.
4) Press OK to store the scaled waveform or CANC to cancel. See warning note below.

NOTE

The maximum number of points in the 16-bit memory is +32767 to -32,768. All scaling must be done with these limits in mind to avoid clipping. For example, a waveform which already occupies the full 16 bits of amplitude can only be reduced in amplitude (or inverted by using a negative DA value). Each time the waveform is rescaled, the digital amplitude display readout resets to 32,768. To reduce the waveform amplitude within memory to one-half its former amplitude, set the digital amplitude (DA) to 16,384. To double the amplitude, set the digital amplitude to 65,535. Other scaling factors can be similarly calculated.

CAUTION

Be sure waveform is not inadvertently clipped before pressing OK key. Press CANCel to start over. A clipped waveform cannot be restored to its former shape after being stored.

6.3.4 Smoothing

A smoothing factor may be applied to any part or all of a waveform. It is computed as a moving average over a specified number of samples.

1) Select anchors as in 6.3.1.
2) Press either the left or right arrow key.
3) Press SMOO softkey.
4) Select the number of samples to be averaged (up to 250) using the edit knob or keypad.
5) Press SHOW softkey to preview.
6) Press CANC softkey to cancel.
7) Press OK softkey to store smoothed waveform.

6.3.5 Inserting Standard Functions

Any one of 20 standard functions can be inserted between the left and right anchors.

1) Select anchors as in 6.3.1.
2) Press INSF key.
3) Select desired standard waveshape by pressing softkey. Use left and right arrow keys to view all 20 waveshapes. Access any desired ancillary functions by pressing the --> softkey.
4) Press SHOW softkey to preview selection.
5) Press CANC softkey to cancel selection.
6) Press OK softkev to store selection.

6.3.6 Summing Standard Functions

Any one of the 20 standard functions can be algebraically summed to any part or all of any other standard function.

1) Select the first standard waveshape as in 6.3.5.
2) If the second standard waveshape is to be summed to only a portion of the first standard waveshape, reposition the anchors as described in paragraph 6.3.1.
3) Press either the left or right arrow key.
4) Press SUMF softkey.
5) Select desired standard waveshape by pressing softkey. Use left and right arrow keys to view all 20 waveshapes. Access any desired ancillary functions by pressing the --> softkey. *(Reduce the digital amplitude value as necessary to prevent clipping.)*
6) Press SHOW softkey to preview summed waveforms.
7) Press CANC softkey to cancel.
8) Press OK softkey to store summed waveforms.

6.3.7 Dump Function

Dump Function permits a standard waveform to be conveniently loaded into the entire length of a Waveform Number without specifying left and right anchors. Thus, standard waveforms can be inserted in Waveform Numbers with lengths greater than 8000 points.

1) Select the desired Waveform Number as in 6.3.
2) Press either the left or right arrow key.
3) Press DMPF softkey.
4) Select desired standard waveshape by pressing softkey. Use left and right arrow keys to view all 20 waveshapes. Access any desired ancillary functions by pressing the --> softkey.
5) Press DO softkey to store the waveform in memory and permit further changes or press OK softkey to store waveform and return to previous menu.

CAUTION

It is not possible to preview a function when using Dump Function. Both the DO and OK commands above will cause the new waveform to write over any previous waveforms.

6.3.8 Move

The Move commands allow a section of a waveform as defined by the left and right anchors to be copied and pasted into another section of the same or another Waveform Number.

1) Select the desired Waveform Number as in 6.3.
2) Press either the left or right arrow key.
3) Press MOVE softkey.
4) Set left and right anchors to the waveform section to be copied.
5) Press COPY softkey.
*6) Press LAST key.
*7) Set Waveform Number to receive pasted section.
*8) Press MOVE softkey.
9) Set left and right anchors to the destination waveform section.
10) Press PSTE softkey.

** These steps may be omitted if copy and paste are within the same Waveform Number.*

NOTE

The pasted waveform will be truncated if the destination waveform section has fewer points than the original.

6.4 MATH OPERATIONS

Math operations permit the contents of any two Waveform Numbers of equal size to be algebraically added, subtracted or multiplied together. Complex composite signals can thus be created, such as shaped tone bursts, amplitude modulation, etc. To enter the Math Mode:

1) Press EDIT key.
2) Press MATH softkey.

6.4.1 Selecting Math Function

The three math functions have the following forms:

A*B (Multiply; output amplitude normalized to full-scale waveform memory.)
A+B (Add; output amplitude divided by two.)
A-B (Subtract; output amplitude divided by two.)

To select one of the math functions:

1) Press OP softkey.
2) Press the softkey for the desired math function. The selected function will be capitalized.

6.4.2 Selecting Waveform Numbers

Waveform Numbers to be combined must be selected and a destination assigned for the combined waveform. (To assign a destination Waveform Number, refer to paragraph 5.6.)

1) Press the arrowhead softkey.
2) The displayed equation has the following form:
   Destination Waveform ## = Waveform ## (*,+ or -) Waveform ##.
3) Define each Waveform Number by moving the arrowhead to each location in the equation and selecting a Waveform Number with the edit knob or keypad (followed by ENTER).
4) Press the DO softkey after the three Waveform Numbers have been assigned. The result is visible at the output.

NOTE

Equal-length waveform blocks are recommended although not required when using math functions.

6.5 EXAMPLES

Waveform editing in the Model 2411A is so flexible that often the same complex waveform can be created several different ways. For example, let us construct a waveform described by the equation: $A = \sin \omega t + 1/6 \sin 3\omega t$. (It is presumed that the waveform lengths are at the default values of LEN = 2000.)

6.5.1 Insert and Sum Functions

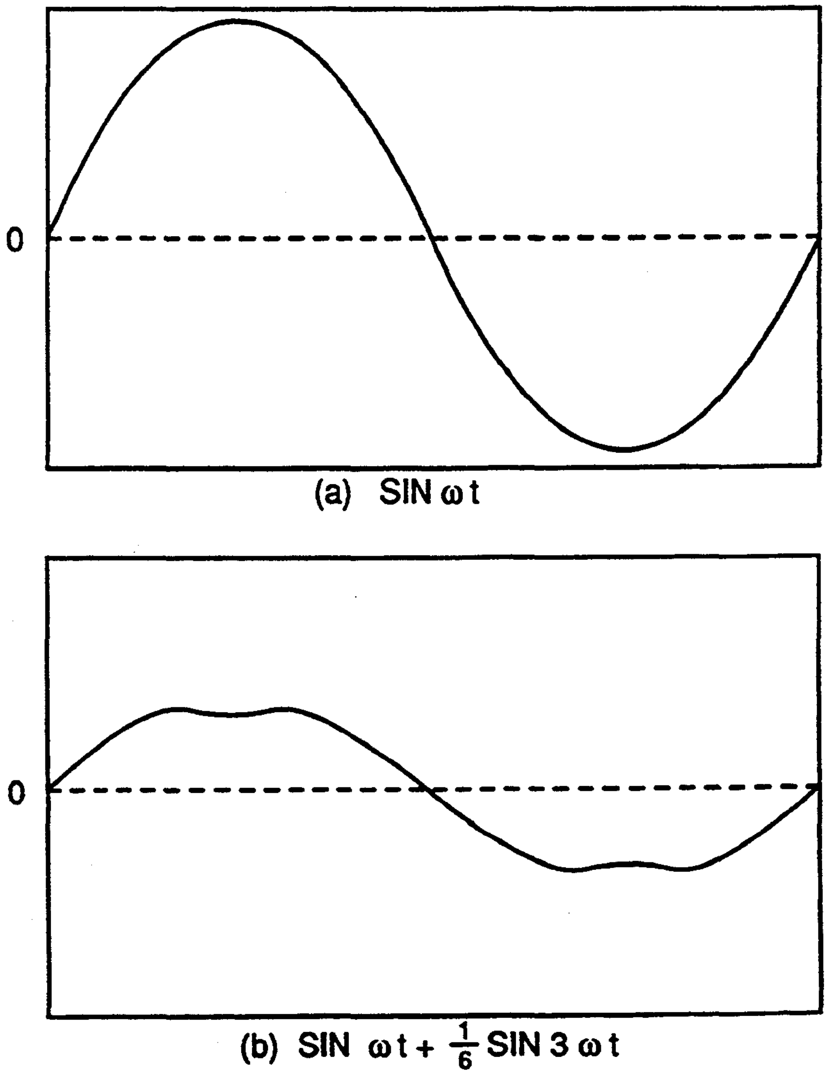
1) Press SETUP key.
2) Press WAV# softkey.
3) Press NEW softkey to select a new Waveform Number.
4) Press OK softkey.
5) Press EDIT key.
6) Press VRTX softkey. Set Waveform Number to that selected in step 3.
7) Press ANCH softkey. Set Left Anchor = 0 and Right Anchor = 1999.
8) Press OK softkey.
9) Press INSF softkey.
10) Press SIN softkey.
11) Set Phase = 0.000 and Number = 1.00.
12) Press --> softkey.
13) Set Digital Offset = 0 and Digital Amplitude = 65,535. (A full-amplitude signal is obtained with DA = 65,535.)
14) Press OK softkey. Observe that the fundamental frequency ($\sin \omega t$) is stored in memory and uses maximum vertical resolution capacity at the waveform peaks, as shown in Figure 6-3a.
15) Press either left or right arrow key.
16) Press SUMF softkey.
17) Press SIN softkey.
18) Set Phase = 0.000 and Number = 3.00.
19) Press --> softkey.

20) Set Digital Offset = 0 and Digital Amplitude = 10,923. (Scaling of the amplitude adjusts for the 1/6 amplitude coefficient.)

21) Press SHOW softkey to preview the results and monitor the output for the desired signal. (Care must be exercised to limit the final waveforms to be within the memory limits.)

22) Press OK softkey.

23) Observe the resulting composite waveform at the output, as shown in Figure 6-3b.



(a) SIN ωt

(b) SIN $\omega t + \frac{1}{6}$ SIN $3\omega t$

Figure 6-3. Insert and Sum Waveform Creation

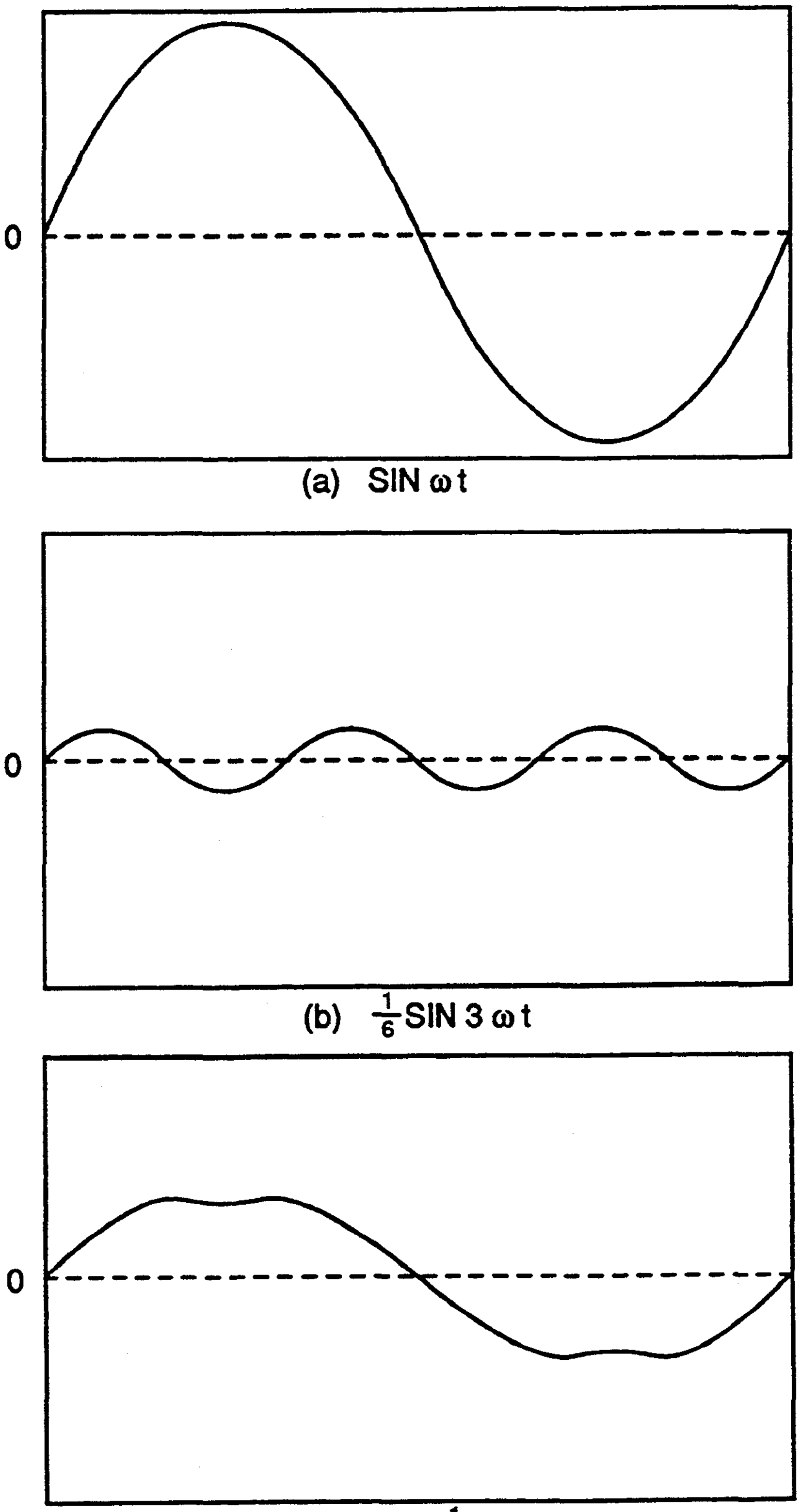
6.5.2 Math Function

1) Press SETUP key.
2) Press WAV# softkey.
3) Use existing Waveform Numbers or press NEW and OK softkeys three times to select three new Waveform Numbers. (As in the previous example, the waveform length is presumed to be LEN = 2000.)
4) Press EDIT key.
5) Press VRTX softkey.
6) Select first Waveform Number from step 3.
7) Press ANCH softkey. Set Left Anchor = 0 and Right Anchor = 1999.
8) Press OK softkey.
9) Press INSF softkey.
10) Press SIN softkey.
11) Set Phase = 0.000 and Number = 1.00.
12) Press --> softkey.
13) Set Digital Offset = 0 and Digital Amplitude = 65,535.
14) Press OK softkey and observe fundamental output as shown in Figure 6-4a.
15) Select second Waveform Number from step 3.
16) Press ANCH softkey. Set Left Anchor = 0 and Right Anchor = 1999.
17) Press OK softkey.
18) Press INSF softkey.
19) Press SIN softkey.
20) Set Phase = 0.000 and Number = 3.00.
21) Press --> softkey.
22) Set Digital Offset = 0 and Digital Amplitude = 10,923 (65,535/6).
23) Press OK softkey and observe third harmonic as shown in Figure 6-4b.
24) Press LAST key.
25) Press MATH softkey.
26) Setup the equation so that the third Waveform Number from step 3 equals the sum of the first waveform (step 6) and the second waveform (step 15).
27) Press DO softkey.
28) Press EDIT key.
29) Press VRTX softkey.
30) Select the third Waveform Number from step 26. Observe resulting half-amplitude composite waveform at the output, as shown in Figure 6-4c.

NOTE

Because the Math Function reduces the composite waveform to one-half its mathematical value in these examples, the Insert and Sum method offers superior resolution.

31) Press SCAL softkey.
32) Set Digital Offset = 0 and Digital Amplitude = 65,535.
33) Press OK softkey.
34) Observe the resulting full-scale composite waveform at the output.



(a) SIN ω t

(b) $\frac{1}{6}$ SIN 3 ω t

(c) SIN ω t + $\frac{1}{6}$ SIN 3 ω t

Figure 6-4. Math Function Waveform Creation

SECTION 7
SEQUENCE GENERATOR

7.1 INTRODUCTION

This section explains how to program and use the optional Sequence Generator. The Sequence Generator permits different waveforms to be repeated and/or linked with each other in any order. The list of programmed instructions for each "loop and link" series is called a Sequence. Up to 100 different Sequences can be programmed. All sequences combined can have up to 1000 steps. Each step defines one Waveform Number (up to 100) and the number of times it repeats (up to 1, 048, 575). Figure 7-1 shows a typical sequence of waveforms. Table 7-1 is a Sequence Programming Worksheet with sample entries corresponding to Figure 7-1. A blank worksheet suitable for reproduction is located in the Appendix.

7.2 PROGRAMMING A SEQUENCE

Before programming a sequence, be sure each desired waveform has been created and stored in a Waveform Number location. In addition, prepare a programming worksheet in a manner similar to Table 7-1. Then, follow this procedure:

1) Press SETUP key.
2) Press SEQ# softkey.
3) Press NEW softkey.
4) Select desired new sequence file number with edit knob or keypad.
5) Press OK to enter number or CANC to cancel.
6) Press OPEN softkey. (A message "STEP=-1" indicates no steps have previously been programmed.)
7) Press ADDS softkey.
8) Select first step number with edit knob or keypad. *(To allow for future changes to the program, it is a good idea to leave room between step numbers, i.e., 10, 20, 30 ....etc.)*
9) Select desired Waveform Number for this step with edit knob or keypad.
10) Press double-arrow key to move burst number to right side of display.
11) Set desired number of waveform repetitions with edit knob or keypad.
12) Press OK to enter numbers or CANC to cancel.
13) Select next step number, repeating steps 7 through 12 above.
14) After all steps have been programmed, view finished results by selecting Sequence Number in Function menu and pressing ENTER. Additional sequences can be programmed and stored by selecting a different sequence file number in the steps above.

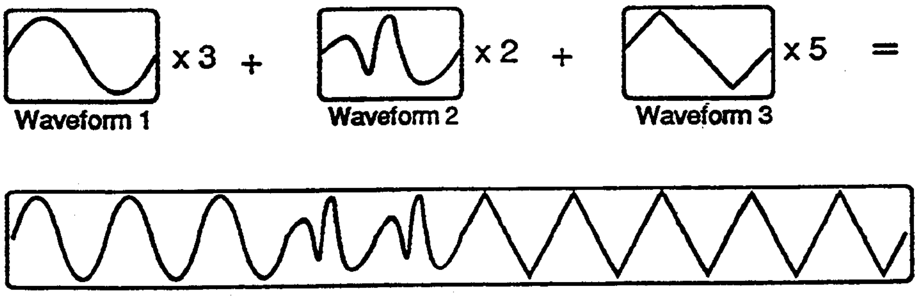


Figure 7-1. .Typical Waveform Sequence

WAVEFORM / SEQUENCE WORK SHEET
TEGAM 2411A

STANDARD WAVEFORM

▨	LENGTH
STDW	1000

SEQUENCE # 10
APPLICATION Demo

CUSTOM WAVEFORMS

WAV #	LENGTH	TYPE
1	2000	Sine
2	2000	Arbitrary
3	2000	Triangle

STEP #	WAV #	NUMBER of BURST
10	1	3
20	2	2
30	3	5

Table 7-1. Sequence Programming Worksheet

7.3 DELETING A SEQUENCE

Sequences no longer required may be deleted from memory by the following procedure:

1) Press SETUP key.
2) Press SEQ# softkey.
3) Select sequence file number to be deleted with edit knob or keypad.
4) Press DELF softkey.
5) Press OK to delete file or CANC to cancel.

7.4 ADDING A STEP TO AN EXISTING SEQUENCE

To add a step to an existing sequence:

1) Press SETUP key.
2) Press SEQ# softkey.
3) Select appropriate sequence file number.
4) Press OPEN softkey.
5) Press ADDS softkey.
6) Select desired new step number with edit knob or keypad..
7) Program new step as in paragraph 7.2, steps 8 to 12.

7.5 DELETING A STEP FROM AN EXISTING SEQUENCE

1) Press SETUP key.
2) Press SEQ# softkey.
3) Select appropriate sequence file number.
4) Press OPEN softkey.
5) Select step number to be deleted with edit knob or keypad.
6) Press DELS softkey.
7) Press OK to delete step or CANC to cancel.

7.6 MODIFYING A STEP WITHIN AN EXISTING SEQUENCE

1) Press SETUP key.
2) Press SEQ# softkey.
3) Select appropriate sequence file number.
4) Press OPEN softkey.
5) Select step number to be modified with edit knob or keypad.
6) Press MODS softkey.
7) Select desired Waveform Number for this step with edit knob or keypad.
8) Press double-arrow key to move burst number to right side of display.
9) Set desired number of waveform repetitions with edit knob or keypad.
10) Press OK to enter numbers or CANC to cancel.

SECTION 8
MULTIPLE UNITS

8.1 INTRODUCTION

Multiple Model 2411A's can be operated synchronously in parallel or series. Synchronous operation of multiple units eliminates triggering jitter and minimizes clock delays. In synchronous operation, the units must share the same sample clock and be programmed for a synchronous trigger interconnect. Refer to Figures 8-1 and 8-2 for diagrams of parallel and series operation.

8.2 PARALLEL OPERATION

Parallel operation is appropriate for applications requiring multi-phase signals, X and Y sweeps, etc.

8.2.1 Clock Connections

To operate multiple units in parallel, designate one unit as the master unit. Connect the rear-panel CLOCK IN/OUT signal from the master unit to the CLOCK IN/OUT connectors of the remaining slave units. Program the slave unit clock inputs to be external as follows: *(The master unit operates with its normal internal clock.)*

1) Press UTIL key.
2) Press SCLK softkey.
3) Press EXT softkey to select external sample clock.
4) Press ENTER.

8.2.2 Trigger Connections

Connect the rear-panel SYNC TRIG OUT signal from the master unit to the TRIG IN connectors of the slave units. Program the master unit sync trigger for parallel operation as follows:

1) Press OUTPUT key.
2) Press >> softkey.
3) Press STRG softkey.
4) Press PAR softkey.
5) Press ENTER.

Program the slave unit trigger inputs for synchronous operation as follows:

1) Press UTIL key.
2) Press TGIN softkey.
3) Press SYNC softkey to select synchronous trigger input.
   (Press ASNC to return to asynchronous mode when returning to single-unit operation.)

When the master unit is operated in the continuous mode, synchronize the units:

1) Press SHIFT key on the master unit.
2) Press TRIG key.

8.3 SERIES OPERATION

Series operation is especially appropriate for complex signal sequences requiring extra-long memory.

8.3.1 Clock Connection

To operate two units in series, designate one unit as the master unit. Connect the rear-panel CLOCK IN/OUT signal from the master unit to the CLOCK IN/OUT connector of the remaining slave unit. Program the slave unit clock input to be external as follows:

1) Press UTIL key.
2) Press SCLK softkey.
3) Press EXT softkey to select external sample clock.
4) Press ENTER.

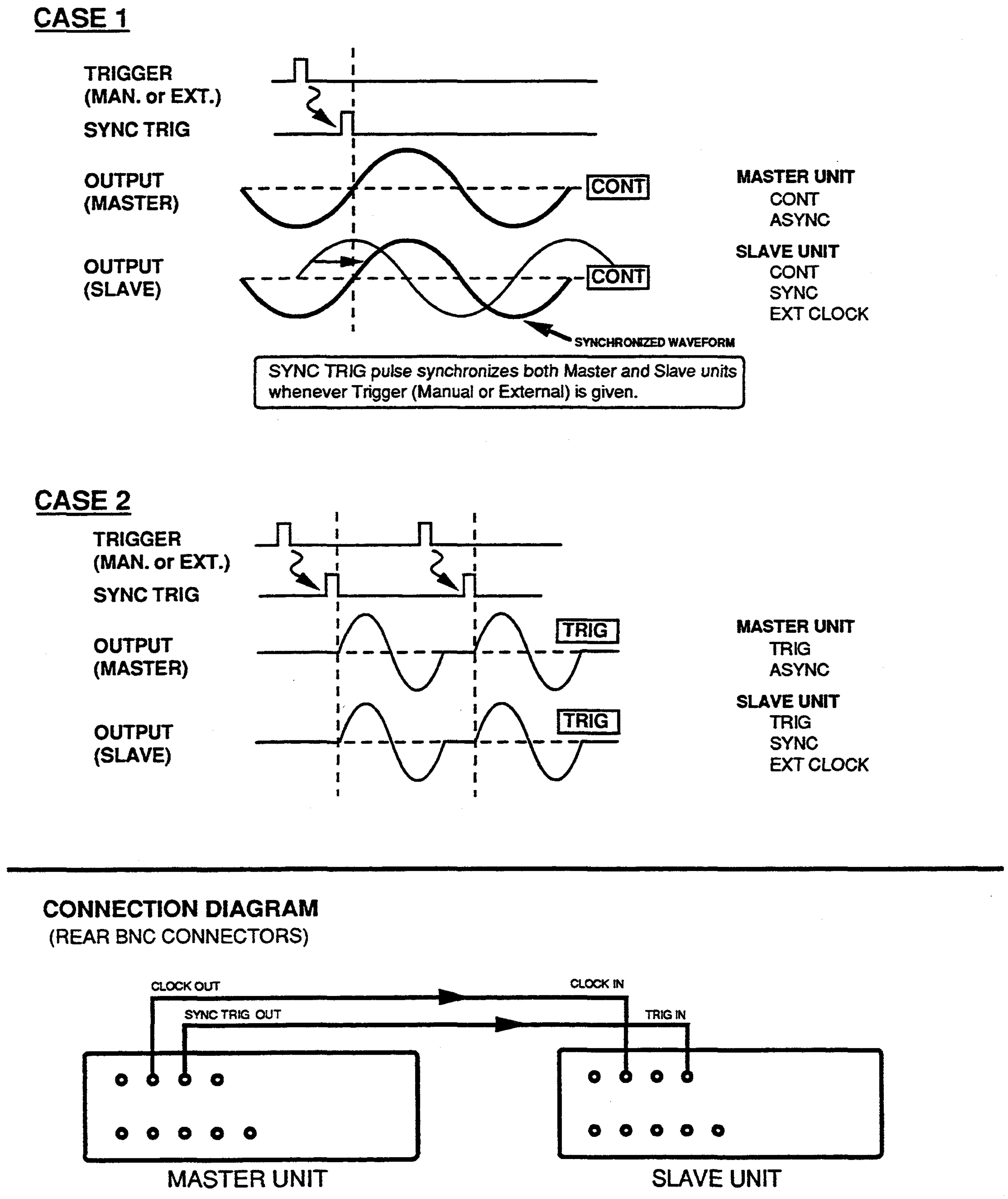
8.3.2 Trigger Connection

Connect the rear-panel SYNC TRIG OUT signal from the master unit to the TRIG IN connector of the slave unit. Program the master unit sync trigger to be serial as follows:

1) Press OUTPUT key.
2) Press >> softkey.
3) Press STRG softkey.
4) Press SER softkey.
5) Press ENTER.

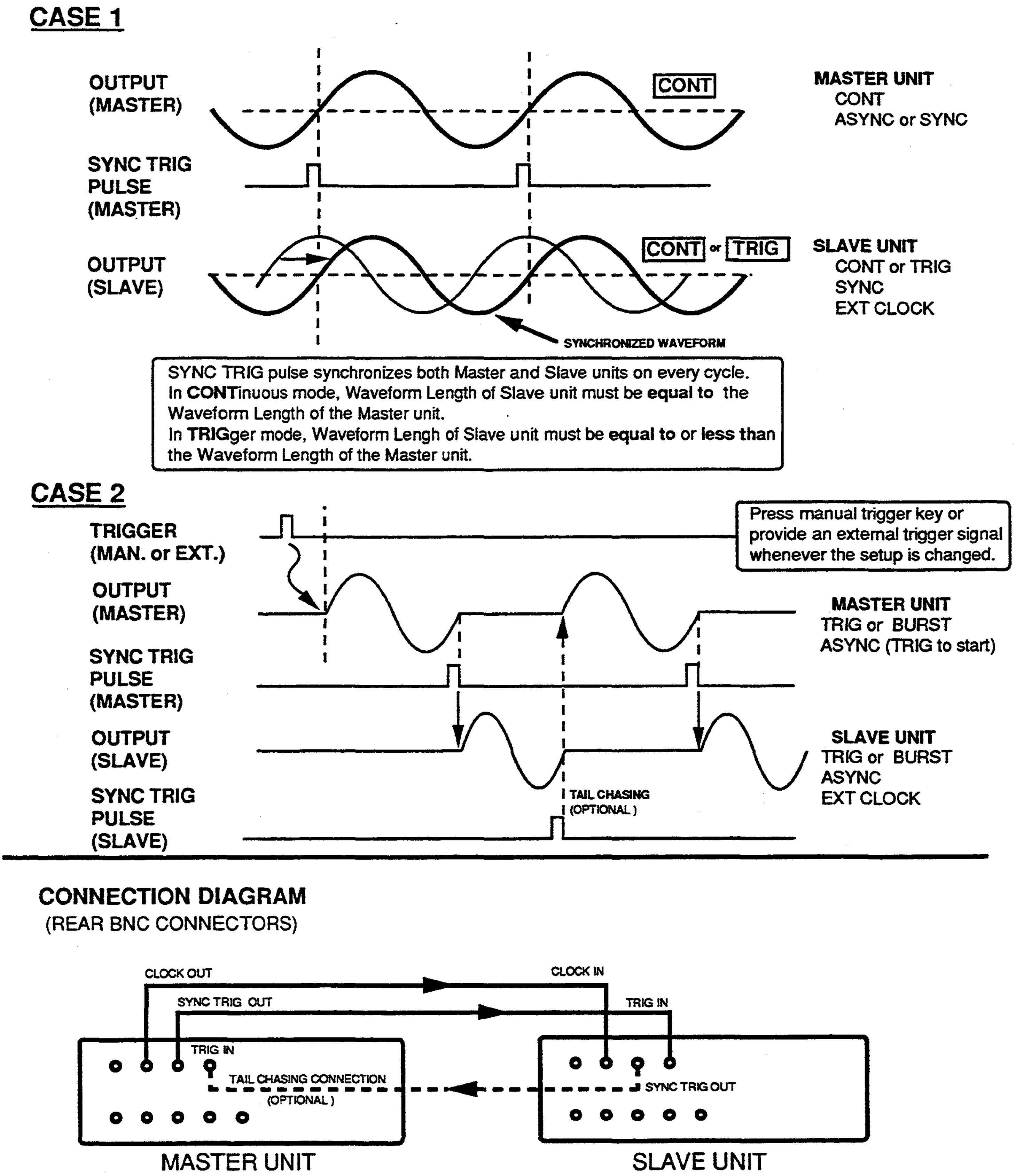
Program the slave unit trigger input to be asynchronous as follows:

1) Press UTIL key.
2) Press TGIN softkey.
3) Press ASNC softkey to select asynchronous trigger input.



SYNC TRIG pulse synchronizes both Master and Slave units whenever Trigger (Manual or External) is given.

Figure 8-1. Parallel Operation



SYNC TRIG pulse synchronizes both Master and Slave units on every cycle.
In **CONT**inuous mode, Waveform Length of Slave unit must be **equal to** the Waveform Length of the Master unit.
In **TRIG**ger mode, Waveform Lengh of Slave unit must be **equal to or less than** the Waveform Length of the Master unit.

Press manual trigger key or provide an external trigger signal whenever the setup is changed.

Figure 8-2. Series Operation

SECTION 9
OTHER FEATURES

9.1 VIEW FUNCTION

The View function allows all or any portion of the 64K waveform memory to be viewed at the output. The amount of memory to be viewed is selected by programming start and stop X addresses, rather than by Waveform Numbers. Thus, several consecutive waveforms occupying different Waveform Numbers can be viewed together. To use the View function:

1) Press FUNC key.
2) Press VIEW softkey.
3) Press STRT softkey.
4) Set View Left Address (VL) with keypad.
5) Press STOP softkey.
6) Set View Right Address (VR) with keypad.
7) Press ENTER key.

The ALL softkey may be used to view the entire 65,536 points in memory.

9.2 SYNC OUTPUTS

The Model 2411A provides three separate sync output signals, as listed in Table 9-1.

SYNC	LOCATION	FUNCTION
SYNC OUT	Front Panel	End Pulse or Programmable Address
SYNC 3 OUT	Rear Panel	Run or Programmable Address or DAC Out
SYNC 4 OUT	Rear Panel	End Block or Programmable Address

Table 9-1. Sync Outputs

Each of the sync outputs has a unique default function, as listed in Table 9-1. In lieu of the default function, each may be programmed to provide a sync output pulse at any address and for any length for each waveform stored in the memory. Selections of the sync functions are made in the Output Menus. Programmable sync address and length settings are made in the Setup Menus. The default sync address is 0 and the default sync pulse length is one clock period.

NOTE

Programmable syncs may need to be re-programmed if they are in a section of memory deleted when the length of a Waveform Number is shortened.

9.2.1 End Pulse

End Pulse is the normal output from the front-panel SYNC OUT connector. It is a TTL level which is high during the last clock interval of the output waveform. In continuous and triggered modes, it is at the end of each cycle. In gated and burst modes, it is at the end of the last cycle. With the optional sequence mode, the End Pulse occurs at the end of the sequence.

9.2.2 Run

Run is the normal output from the rear-panel SYNC 3 OUT (RUN) connector. It is a TTL level which is high whenever an output signal is present.

9.2.3 End Block

End Block is the normal output from the rear-panel SYNC 4 OUT (END BLK) connector. It is used only with the optional Sequence Generator to provide a TTL level which is high during the last clock interval at the end of each step in the sequence.

9.3 Z-AXIS LEVEL

Z-Axis is the output from the rear-panel Z-OUT connector. It provides a variable level pulse at times coincident with the cursor and anchor positions when in the waveform edit modes. When connected to the Z-Axis input of the monitor oscilloscope, it provides intensity modulation of the display to show the cursor and anchor positions. The level setting depends upon the sensitivity of the oscilloscope. To adjust the Z-Axis level:

1) Press EDIT key.
2) Press LINE softkey.
3) Press ZLVL softkey.
4) Use edit knob or keypad to set cursor and anchor intensity to desired level. (Adjust scope trace intensity at the same time for best results.)

9.4 OUTPUT FILTERS

One of two low-pass filters may be switched into the output signal circuit. Either a 40-kHz third-order Chebyshev or a 700-kHz seventh-order Butterworth may be selected. The filters are effective in removing sampling step noise. The default setting for the output filters is OFF. To enable one of the output filters:

1) Press OUTPUT key.
2) Press 40K or 700K softkey (capital "K" = ON; lower-case "k" = OFF).
3) Press ENTER key.

9.5 INTERNAL TRIGGER GENERATOR

The triggered, toggled and burst modes require an external, manual or internal trigger. The internal trigger generator provides a periodic trigger at a variable rate from .02 to 10 seconds. To select the internal trigger:

1) Press UTIL key.
2) Press TGEN softkey.
3) Press ON softkey.
4) Set desired trigger interval with edit knob or keypad. Press ENTER key if keypad is used.

NOTE

The internal trigger generator provides only short pulses which are not useable for gated mode.

CAUTION

Remember to turn the internal trigger generator OFF when it is no longer needed.

9.6 RTS

Return To Start interrupts the output signal and ramps the output signal back to the start level. Ramp time is 15 seconds. RTS may be implemented by applying a TTL level to the rear-panel RTS IN connector or by pressing the SHIFT and RTS keys on the front panel. Apply a TTL level to the rear-panel HOLD IN connector or press the HOLD key to restart the signal at the beginning.

9.7 HOLD

Hold stops the output signal and holds it at its present level. Hold may be implemented by applying a TTL level to the rear-panel HOLD IN connector or by pressing the SHIFT and HOLD keys on the front panel.

9.8 MONITOR BURST COUNT

In the burst mode, the output cycle count can be monitored at any time. This is most appropriate for slow, low-frequency signals. To monitor the burst count:

1) Press UTIL key twice.
2) Press MBST softkey.

Each time the MBST softkey is pressed the counter is updated and displays the burst count at that time.

9.9 DEGLITCHER

When an analog to digital converter makes a major transition, (e.g. from 10000 to 01111) a "glitch" or voltage spike can appear in the analog output waveform. The deglitcher removes this spike. The default deglitcher setting is ON. The deglitcher can, however, cause a slightly higher noise level on constant-amplitude portions of a waveform. The deglitcher can be switched OFF (or ON) as follows:

1) Press OUTPUT key.
2) Press >> softkey.
3) Press DEGL softkey.
4) Press OFF or ON softkey.
5) Press ENTER key.

9.10 DIRECT DAC OUTPUT

An analog signal may be taken directly from the output of the digital-to-analog converter. This signal is prior to the output attenuator, offset, filter and amplifier circuits and has a fixed amplitude of ±5 volts into a high-impedance load. The direct DAC signal may be obtained from the SYNC 3 connector on the rear panel:

1) Press OUTPUT key.
2) Press >> softkey.
3) Press SYNC softkey.
4) Press SYN3 softkey.
5) Press DAC softkey.
6) Press ENTER key.

9.11 STORED SETTINGS

Up to 31 different sets of instrument settings may be stored in non-volatile memory for easy recall later. The settings which may be recalled include the following:

- Mode
- Function
- Amplitude
- Offset
- Sample Clock Frequency
- Trigger & Clock Utilities
- Output On/Off & Filters
- Waveform Number
- Sequence Number

To store a setup:

1) Press STOR/RECL key.
2) Select setting number (0 to 30).
3) Press STOR softkey.
4) Press OK softkey to store setup or CANC to cancel.

To recall a previously stored setup:

1) Press STOR/RECL key.
2) Select setting number.
3) Press RCLL softkey.
4) Press OK softkey to recall setting or CANC to cancel.
5) Press ENTER key.

SECTION 10
RS-232C & GPIB

10.1 INTRODUCTION

This section describes how to remotely control all instrument functions and how to download and upload waveform data using either the standard RS-232C or optional GPIB interface. Included are an overview of both methods of remote control, a complete tabulation and explanation of control commands, and several programming examples.

10.2 QUICK REFERENCE

This command summary is provided as a quick reference and overview of the complete command list for the Model 2411A. For a detailed explanation, including command syntax, parameters and data limits see Section 10.5. For convenience, commands are listed here in the same order as in Section 10.5.

Command Long Form	Short Form	Command Long Form	Short Form
Common Commands		System Commands	
*CLS		EXECUTE	EXEC
*ESE (GPIB only)		HOLD	
*ESE? (GPIB only)		RECALL	RCLL
*ESR?		REF_CLK_ADJ	RADJ
*IDN?		REF_CLK_ADJ?	RADJ?
*OPC (GPIB only)		REF_CLOCK	RCLK
*OPC?		REF_CLOCK?	RCLK?
*RST		RTNTOSTRT	RTST
*SRE (GPIB only)		RESET	
*SRE? (GPIB only)		SAMPLECLOCK	SCLK
*STB? (GPIB only)		SAMPLECLOCK?	SCLK?
*OPT?		STORE	STOR
*TRG		TGENERATOR	TGEN
*TST?		TGENERATOR?	TGEN?
*WAI		TGRRATE	TGRR
		TGRRATE?	TGRR?
Configuration Commands		TRIGGER	TRIG
CONFIGURE	CONF		
HEADERS	HDRS		
HEADERS?	HDRS?		

NOTE:
Immediate execution of the following System Commands does not require the execute command: HOLD, RTNTOSTRT, RESET, STORE, and TRIGGER.

Command Long Form	Short Form
Output Commands	
AMPLITUDE	AMPL
AMPLITUDE?	AMPL?
BURST	
BURST?	
CLOCK_SEL	CLKSEL
CLOCK_SEL?	CLKSEL?
DEGL	
DEGL?	
FILTER	
FILTER?	
FREQUENCY	FREQ
FREQUENCY?	FREQ?
FUNCTION	FUNC
FUNCTION?	FUNC?
MODE	
MODE?	
OFFSET	OFST
OFFSET?	OFST?
OUTPUT_SWTCH	OUTSW
OUTPUT_SWTCH?	OUTSW?
READ_BURST?	RBRS?
SYNCSEL	SYSEL
SYNCSEL?	SYSEL?
TRGINMODE	
TRGINMODE?	
TRGOUTMODE	
TRGOUTMODE?	
Sequence Generator Commands	
WAVEFORM	WVFM
ADDSEQUENCE	ADDSEQ
ADDSEQUENCE?	ADDSEQ?
AUTOSEQUENCE	AUTO
SEQUENCE	SEQ
SEQBURST	SEQB
SEQBURSTNUM	SEQBN
Waveform Editing Commands	
WAVEFORM	WVFM
AM	
CIRCLE	
DC	
EXPONENTIAL	EXP
FM	
GAUSSIAN	GAUSS
HAVERSINE	HSIN
LINE	
LINEARSWEEP	LINS
LOGSWEEP	LOGS
NOISE	
PULSE	
SAWTOOTH	SAW
SCM	
SINE	
SQUARE	
SINE_X_OVR_X	SXX
TRIANGLE	
LENGTH	LEN
LENGTH?	LEN?
MAXY	
MAXY?	
MINY	
MINY?	
POSITION	POSN
POSITION?	POSN?
SIZE	
SIZE?	
SYNC	
SYNC?	
WAVE	
WAVE?	
MEMORY	MEM
MEMORY?	MEM?
MEM_BLOCK?	MBLK?
STATUS	STAT
INSTRUMENT	INST
CLR	
ENABLE	ENBL
ENABLE?	ENBL?

10.3 RS-232C OVERVIEW

10.3.1 Introduction

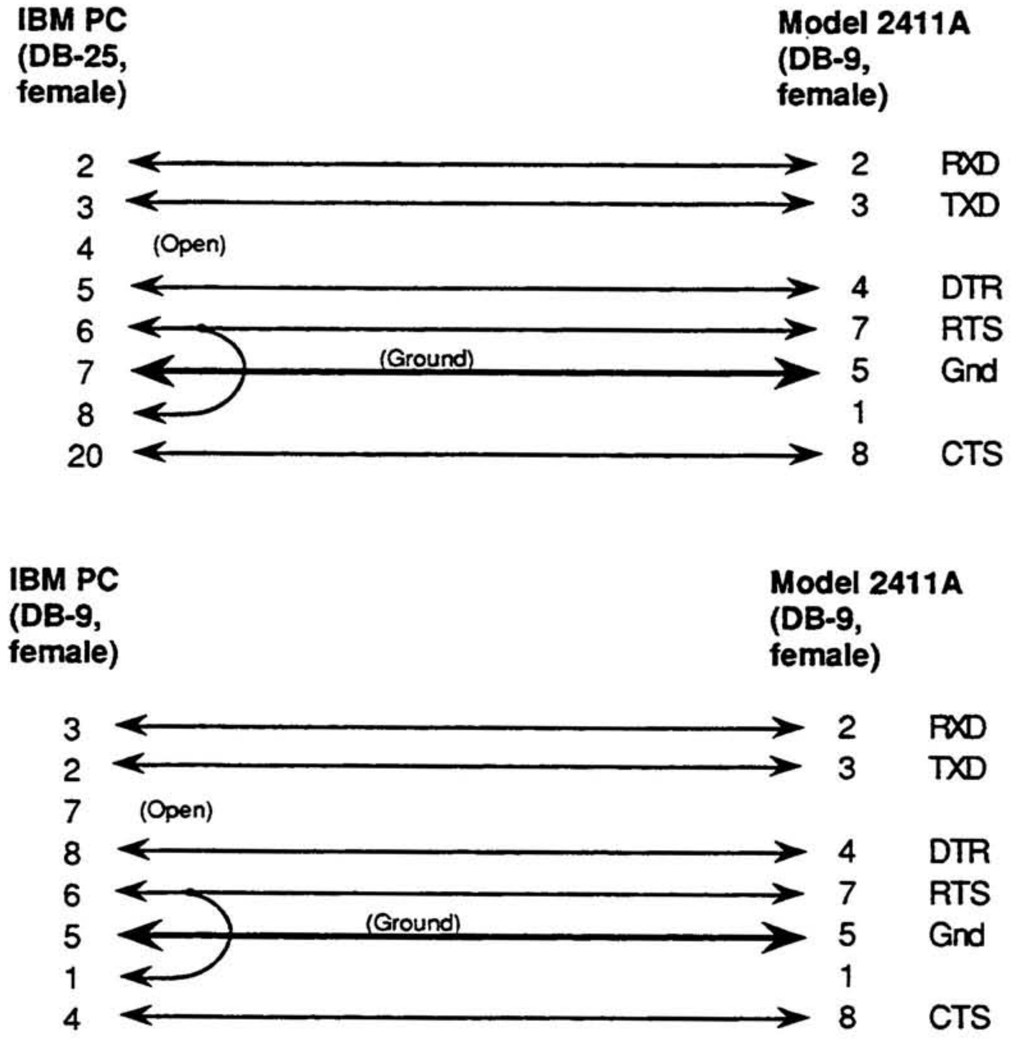
RS-232C is an industry-standard method of sending data back and forth between two pieces of equipment. With the Model 2411A, a computer can remotely control the instrument, download waveform data and upload waveform data. This overview explains the interface requirements, 2411A setup, how to verify communications and the command syntax structure. A sample program is also provided.

10.3.2 Interface Requirements

All IBM (or IBM compatible) personal computers (PCs) should be equipped with at least one serial interface port. It may be either a 9-pin DB-9 or a 25-pin DB-25 connector. An 8-foot 9-pin to 9-pin cable is included with the Model 2411A. If desired, a cable may be constructed per Figure 10-1. Most any software which defines communication protocols may be used. This includes the programming languages Quick Basic, GW Basic, Quick C, Turbo C and Turbo C++. Communications programs such as ProComm, a "shareware" version, are also usually acceptable. A "local echo" feature is helpful to monitor your typing.

RS-232C ADAPTER CABLES

The following wiring diagrams illustrate proper interconnects between the serial port of an IBM (or compatible) PC and the Model 2411A RS-232C connector.



NOTE: 1) The above cables work for both hardware (HW) and software (SW) handshake. However, binary waveform download is not supported in the software handshake setting.

2) Special software is required for the PC to support software handshaking.

Figure 10-1. RS-232C Cable Schematic

10.3.3 2411A Setup

The following communication protocol parameters are recommended:

Baud Rate	19.2k
Parity	None
Bits	8 Data, 1 Stop
Handshake[1]	Hardware

To setup the RS-232C:

1) Press UTIL key.
2) Press UTIL key again.
3) Press R232 softkey.
4) Press BAUD softkey.
5) Press 19k2 softkey.
6) Press LAST key.
7) Press PAR softkey.
8) Press NONE softkey.
9) Press LAST key.
10) Press BITS softkey.
11) Press 8D1S softkey.
12) Press LAST key.
13) Press HAND softkey.
14) Press HW softkey.
15) Press ENTER key.

To setup the GPIB:

1) Press UTIL key.
2) Press UTIL key again.
3) Press GPIB.
4) Set address using the knob or keypad.
5) Press ENTER key.

To setup the LCD display viewing angle:

1) Press SHIFT key.
2) Press 7 key (labelled "LCD" above the key)
3) Set LCD = (0-15).
   Use the knob to maximize the viewing angle
   by selecting a setting between 0 to 15.
4) Press SHIFT key.

[1] With software handshaking, flow control of data to/from the instrument is controlled by the XON/XOFF ASCII characters.
Sending data to 2411A:
  Instrument will send a XOFF (ASCII CTRL-S) when the instrument buffer fills to 200 characters.
  Instrument will send a XON (ASCII CTRL-Q) when instrument buffer emties to 80 characters.
Receiving data from 2411A:
  Instrument will stop sending data when a XOFF (ASCII CTRL-S) is received.
  Instrument will resume sending data when a XON (ASCII CTRL-Q) is received.

With hardware handshaking, flow control of data to/from the instrument is controlled by the DTR/CTS lines of the RS-232C interface.
Sending data to 2411A:
  Instrument will turn the DTR line off (-12V) when the instrument buffer fills to 200 characters.
  Instrument will turn the DTR line on (+12V) when the instrument buffer empties to 80 characters.
Receiving data from 2411A:

Instrument will stop sending data when the CTS line is off and resume sending data when it is on.

10.3.4 Verify Communication

After the PC and the 2411A have been connected together and programmed for compatible interface parameters, the interface should be tested for proper operation. The following program notation conventions will be observed: The symbol ^ represents the computer Control Key. The use of braces { } around two characters means the two keys must be pressed simultaneously.

To test the interface, type the following:

```
*IDN?{^J}{^D}
```

The 2411A should identify itself with the following:

```
Pragmatic Instruments, MODEL 2411A, 0, V1.XX
 (XX represents the current firmware revision number.)
```

10.3.5 Command Syntax

The command syntax of the Model 2411A closely relates to the Institute of Electrical and Electronics Engineers (IEEE) Standard 488.2 - 1987.
Commands can be divided into two major categories; common commands and instrument specific commands.

Overviews of the following topics are provided, as they relate specifically to the RS-232C interface:

Common Commands
Event Register and Status and Error Reporting
Functional Syntax Elements

Instrument specific commands, which are identical to both RS-232C and GPIB interfaces, are explained in Section 10.5.

10.3.5.1 Common Commands

Common commands, recognizable by their leading "*" character, are defined by the IEEE 488.2 standard. Common commands dealing with GPIB protocols or status reporting are not supported in the RS-232C interface. Therefore, RS-232C uses the following subset of the mandated GPIB common commands:

Command	Description
*CLS	Clear Status
*ESR?	Standard Event Status Register Query
*IDN?	Identification Query
*OPC?	Operation Complete Query
*TRG	Trigger Command
*TST?	Self-Test Query
*WAI	Wait-to-Continue
*OPT?	System Option Query
*RST	Reset

See Section 10.5 for further descriptions of command formats, operation and expected responses from queries.

10.3.5.2 Event Register and Status and Error Reporting

The Standard Event Status Register (ESR) may be utilized to indicate the instrument status. Each of six bits within the eight-bit register indicates a different condition within the 2411A. Refer to Page 10-13, section 10.4.3.

ESR - STANDARD EVENT STATUS REGISTER

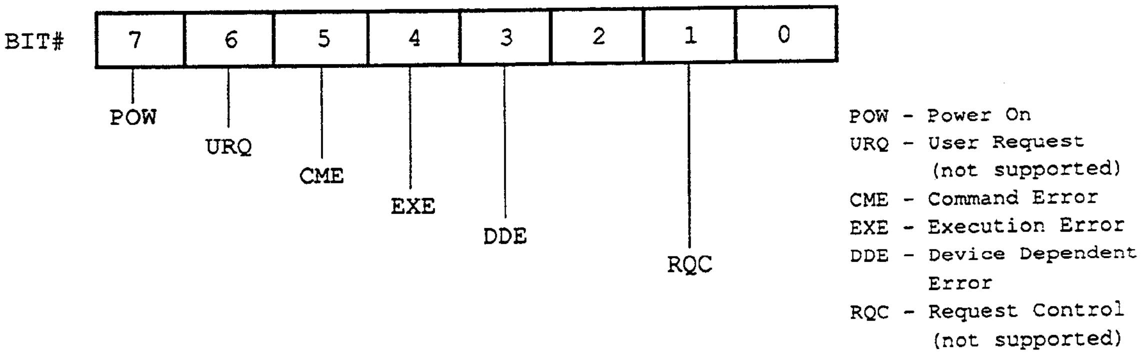

BIT#
7
6
5
4
3
2
1
0
POW
URQ
CME
EXE
DDE
RQC
POW - Power On
URQ - User Request (not supported)
CME - Command Error
EXE - Execution Error
DDE - Device Dependent Error
RQC - Request Control (not supported)

Figure 10-2. ESR Register

10.3.5.3 Functional Syntax Elements

In order to establish programming consistency among different manufacturers' devices, IEEE-488.2 has defined a set of rules governing message headers, mnemonics, separators and data types. The following overview will familiarize the programmer with the fundamentals of these rules. As seen below, the rules for command and query messages are much more flexible than their precise response message counterparts.

<PROGRAM MESSAGE UNIT> - This is the basic message, and represents an operation to be performed by the device. As an example, if you wanted to create a sinewave with 5 cycles in the currently selected waveform, the appropriate <PROGRAM MESSAGE UNIT> would be:

WVFM:SINE 5,0

Notice that a colon (ASCII 3A) is used to separate the <program mnemonic> **WVFM** from **SINE.** For query responses the similarly structured <RESPONSE MESSAGE UNIT> is used. A complete list of <PROGRAM MESSAGE UNITS> and <RESPONSE MESSAGE UNITS> for the 2411A, with definitions, mnemonics, and limitations is given in Section 10.5.

General rules:

1. <program mnemonic> shall have maximum length of 12 characters.
2. ":" is used to separate <program mnemonics>, and when preceding a <PROGRAM MESSAGE UNIT>, it indicates that the following <program mnemonic> is at the root level.

<PROGRAM MESSAGE UNIT>

1. <program mnemonic> can be either upper or lower case alpha.
2. <white space> is only allowed at the end of the header, or next to a "," or a ";". (See note on <white space> below.)

<RESPONSE MESSAGE UNIT>

1. <program mnemonic> shall contain upper case alpha only.
2. No <white space> allowed in message header.

To receive a response to a query command, append "^D" (ASCII 4 or CTRL-D) to the <RESPONSE MESSAGE UNIT>. For example, after sending *IDN?<RMT> followed by "^D", the 2411A will respond with "Pragmatic Instruments, Model 2411A, 0, V1.XX".

Note: <white space> is defined as a length of 1 or more of <white space characters>. A <white space character> is a single ASCII byte in the range of 00-09, 0B-20 Hex.

<PROGRAM MESSAGE> - This is a series of <PROGRAM MESSAGE UNITS> sent to the device in a single string. For example, to set the currently selected waveform to 1) a sinewave output with a single period, 2) burst mode, 3)a burst of 5, and 4) an amplitude of 2.5V, you would send the following <PROGRAM MESSAGE>:

```
WVFM:SINE 1,0;:MODE BURST;:BURST 5;:AMPL 2.5;:EXEC
       { msg 1 }   { msg 2 }          { msg 3 }  { msg 4 }
```

The semicolon (ASCII 3B) is required to separate <PROGRAM MESSAGE UNITS> within a <PROGRAM MESSAGE>. For the 2411A a <PROGRAM MESSAGE> can be virtually any length. The structure for a <RESPONSE MESSAGE> for query responses is similar.

<PROGRAM MESSAGE TERMINATOR> or **<PMT>** - In order for the device to recognize the end of a <PROGRAM MESSAGE>, a special terminator is required. For command messages, the <PMT> can take one of two different formats:

CR LF	CR represents carriage return and is an ASCII 0D. LF represents line feed and is an ASCII 0A.
LF	LF represents line feed and is an ASCII 0A.

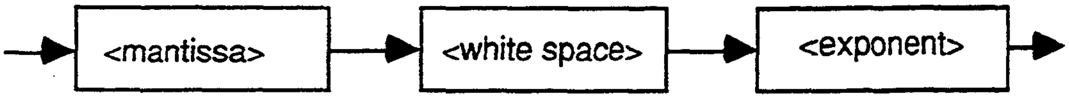
<RESPONSE MESSAGE TERMINATOR> or **<RMT>** - For query messages the required terminator is:

CR LF (as above)

10.3.5.4 Data Formats

Many of the <PROGRAM MESSAGE UNITS> and <RESPONSE MESSAGE UNITS> include numeric data in the message, e.g., the '5' in the message **SINE 5,0**. This section defines acceptable data formats, and Section 10.5 indicates which formats are used with specific commands and responses.

<DECIMAL NUMERIC PROGRAM DATA> or **<NRf>** - This is the most flexible of the numeric representations and takes the following general form:



where <mantissa> = [±][x...x][.x...x] with a maximum length of 255 characters (excluding leading zeros), and the

optional <exponent> = E[±][x...x] or e[±][x...x] , with a maximum value of ±32000.

Note in the above definitions: x represents digits 0-9
[ ] means enclosed characters are optional

The following example demonstrates several acceptable ways to represent the number 1,234,567,890 in <NRf> format:

+1234567890
1.234567890E+9

123456.7890<white space>**e04**
+.1234567890E+10

Three other numeric data formats are used in <RESPONSE MESSAGE UNITS>, and are subsets of the more general <NRf> format.

<NR1 NUMERIC RESPONSE DATA> or **<NR1>** -
[±][x...x] e.g. -98765432

<NR2 NUMERIC RESPONSE DATA> or **<NR2>** -
[±]x[x...x].x[x...x] e.g. +98765.432

<NR3 NUMERIC RESPONSE DATA> or **<NR3>** -
[±]x[x...x].x[x...x]E±x[x..x] e.g. 987.65432E+05

<NON-DECIMAL NUMERIC PROGRAM DATA> - Numeric values may also be represented as a binary, octal, or hex number, as follows:

Binary - **#B**x[x...x] or **#b**x[x...x] where x is a **0** or **1**
Octal - **#Q**x[x...x] or **#q**x[x...x] where x is a **0** thru **7**
Hex - **#H**x[x...x] or **#h**x[x...x] where x is a **0** thru **F**

<ARBITRARY BLOCK PROGRAM DATA> - This data format is used to speed bus transfer in cases where large amounts of data are sent to or from a device (such as waveform or buffer memory data in the 2411A). Only the <DEFINITE LENGTH> block data format is acceptable.

> <DEFINITE LENGTH> - **#zy..yx...x** where z is a number 1-9, and represents the number of y digit elements. The y digits taken together as a decimal integer equal the number of 8-bit bytes that follow.
>
> For example to send 4 data bytes<DAB> using the <DEFINITE LENGTH> format, you could send:
>
> **#14<DAB><DAB><DAB><DAB>**
> or
> **#204<DAB><DAB><DAB><DAB>**

Refer to Section 10.8 for detailed instructions on how to enter data into the waveform memory of the 2411A.

10.3.6 Sample Program

The following sample RS-232C program is written in QuickBasic. It reads the instrument identification and writes a 3000-point sine wave into Waveform Number 00:

```
'*****************************************************************************
'****                                                                     ****
'****       MODEL 2411A : Programming in RS232    (REV 0  2-18-93)        ****
'****                                                                     ****
'*****************************************************************************
'
'   QuickBASIC program using COM1 serial port on IBM PC.
'   Program   a) Reads Instrument Identification  (Query)
'             b) Writes a 3000 point sine wave into WAV#00
'
'   The 2411A communication settings must be programmed to:
'   BAUD: 9.6K     PAR: NONE     BITS: 8D1S     HAND: HW
'
CLS                                         ' Clear Screen
CONST Pi = 3.14159
CrLf$ = CHR$(13) + CHR$(10)                 ' Command terminator
Talk$ = CHR$(4)                             ' Request query response

' OpenCommLink

  OPEN "COM1:9600, N, 8, 1, CS5000, DS5000, BIN" FOR RANDOM AS #1
  PRINT #1, CrLf$ + CrLf$;                  ' Flush 2411A receive buffer


' Read Identification

  PRINT #1, "*IDN?" + CrLf$;                ' Identification query command
  PRINT #1, Talk$;                          ' Request query response
  LINE INPUT #1, QueryResponse$             ' and read it.
  Flush$ = INPUT$(1, 1)                     ' Flush trailing `Line Feed`
  PRINT "Instr. ID : "; QueryResponse$      ' which is left in buffer


' Reset 2411A

  PRINT #1, "*Rst" + CrLf$;

' Size WAV#00 to 3000 points and run it

  PRINT #1, ":Wvfm:Wave 0; Size 0" + CrLf$;     'Erase old WAV#00
  PRINT #1, ":Wvfm:Wave 0; Size 3000" + CrLf$;
  PRINT #1, ":Func Wave,0; :exec" + CrLf$;
  PRINT #1, ":Outsw On; :exec" + CrLf$;


' Draw sine in WAV#00, starting at address 0.

  Slice = 2 * Pi / 3000
  PRINT #1, ":Wvfm:Mem 0";                  ' Command header, followed by
  FOR Addr = 0 TO 2999                      ' 3000 points of sine wave.
     DataPoint = INT(32767.5 * SIN(Addr * Slice))
     PRINT #1, "," + STR$(DataPoint);       ' Data `,` separated
  NEXT Addr
  PRINT #1, CrLf$;                          ' Terminate command now

CLOSE
'~~~~~~~~~~~~~~~~~~~~~
```

10.4 GPIB (IEEE-488.2) OVERVIEW

10.4.1 Introduction

The Model 2411A with the GPIB option, conforms to the Institute of Electrical and Electronics Engineers (IEEE) Standard 488.2-1987. The specific implementation of IEEE-488.1 includes the following functions and subsets:

Interface Function	Subset
Source Handshake	SH1
Acceptor Handshake	AH1
Talker	T6
Listener	L4
Service Request	SR1
Remote Local	RL1
Parallel Poll	PP0
Device Clear	DC1
Device Trigger	DT1
Controller	C0
Electrical Interface	E1

To facilitate programming, a brief overview of the IEEE-488.2 Standard (as it specifically applies to the 2411A) is provided.

This section includes: **Common Commands, Status and Event Registers, Functional Elements** (including syntax and nomenclature), **Data Formats, and Error Reporting.** For a more detailed discussion of these topics, a copy of IEEE Standard 488.2-1987 may be obtained from:

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017

10.4.2 Common Commands

Commands can be divided into two major categories: **common commands and instrument specific commands.** Instrument specific commands are detailed in Section 10.5. Common commands are defined by the standard, and, among other things, are used to manage status registers and synchronization. The following is a list of common commands as implemented in the 2411A:

Command		Description
***CLS**		Clear Status
***ESE**	(GPIB only)	Standard Event Status Enable
***ESE?**	(GPIB only)	Standard Event Status Enable Query
***ESR?**		Standard Event Status Register Query
***IDN?**		Identification Query
***OPC**	(GPIB only)	Operation Complete
***OPC?**		Operation Complete Query
***RST**		Reset
***SRE**	(GPIB only)	Service Request Enable
***SRE?**	(GPIB only)	Service Request Enable Query

Command (Cont.)		Description (Cont.)
***STB?**	(GPIB only)	Status Byte Query
***TRG**		Trigger Command
***TST?**		Self-Test Query
***WAI**		Wait-to-Continue
***OPT?**		System Option Query

See Section 10.5 for further descriptions of command formats, operation, and expected responses from queries.

10.4.3 Status and Event Registers

There are four required status or event registers. They are: 1)**Standard Event Status Enable (ESE) Register**, 2)**Standard Event Status (ESR) Register**, 3)**Service Request Enable (SRE) Register**, and 4)**Status Byte (STB)**. These registers indicate device status, and allow the programmer to specify which device events will enable a service request.

ESR and ESE Registers - Each bit of the 8 bit ESR Register indicates a different condition within the device (see Figure 10-2). The ESE Register provides a bit by bit mask of the ESR register. When a bit in the ESE Register is set TRUE, it enables the corresponding ESR bit to generate a Service Request (SRQ), if the ESB bit (bit 5) in the SRE Register has also been enabled. (See discussion on SRE and STB Registers.)

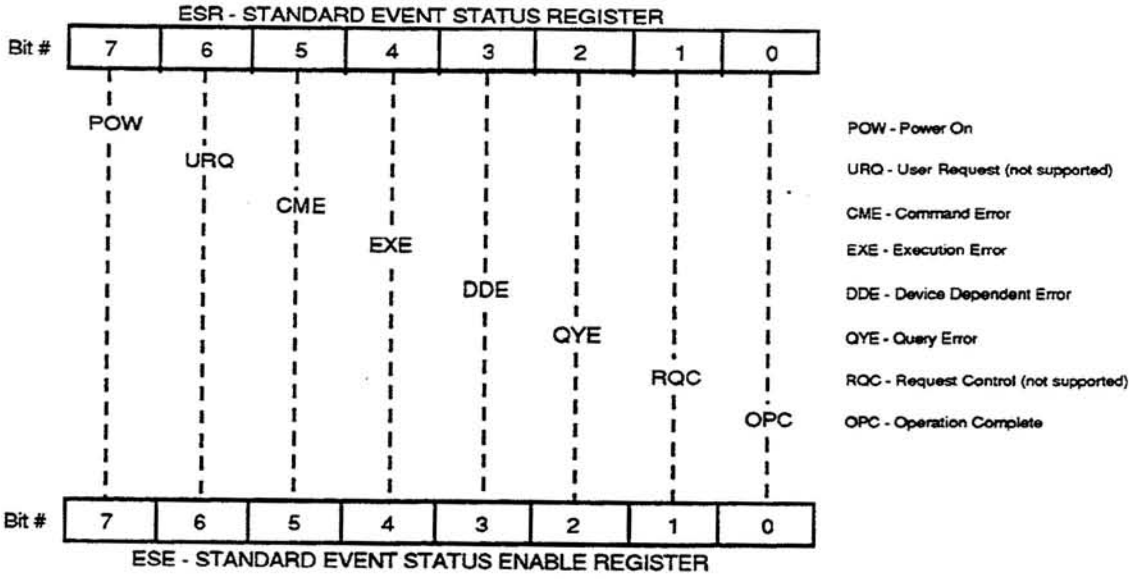

ESR - STANDARD EVENT STATUS REGISTER
Bit #
7
6
5
4
3
2
1
0
POW
URQ
CME
EXE
DDE
QYE
RQC
OPC
POW - Power On
URQ - User Request (not supported)
CME - Command Error
EXE - Execution Error
DDE - Device Dependent Error
QYE - Query Error
RQC - Request Control (not supported)
OPC - Operation Complete
Bit #
7
6
5
4
3
2
1
0
ESE - STANDARD EVENT STATUS ENABLE REGISTER

Figure 10-3. ESR and ESE Registers

For example, if you wanted to generate an SRQ on 1) power on (bit 7), or 2) command error (bit 5), or 3) query error (bit 2), you would first set the ESB bit in the SRE Register (see discussion below) and then send the following:

***ESE 164** (164 = decimal equivalent of binary 10100100)

The ***ESE?** and ***ESR?** command queries allow reading of the Standard Event Status Enable and Standard Event Status Registers, respectively. (Ref. Section 10.5.4)

SRE and STB Registers - The Status Byte (STB) Register of the 2411A has four active bits (see Figure 10-3) which summarize the current status of the event registers, output queue, and certain instrument specific functions. Similar to the ESE Register, the SRE Register provides an 8-bit mask to allow the programmer to enable each STB bit (with the exception of the MSS bit 6) to generate an SRQ.

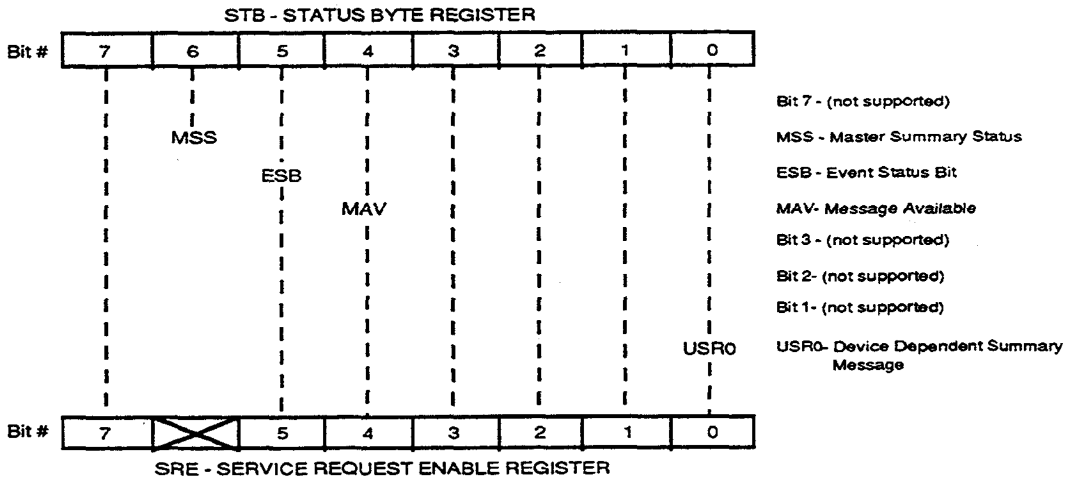


Figure 10-4. SRE and STB Registers

As an example, to enable the ESB and MAV bits, but not the USR0, of the Status Byte Register, you would send the following command:

***SRE 48** (48 = Decimal equivalent of B00110000)

The ***SRE?** and ***STB?** command queries allow reading of the Service Request Enable and Status Byte Registers, respectively. (Ref. Section 10.5.4)

The USR0 (bit 0) is user-defined, and summarizes the instrument status registers (reference Section 10.5.4, Instrument commands).

10.4.4 Functional Elements - Syntax and Nomenclature

In order to establish programming consistency among different manufacturers' devices, IEEE-488.2 has defined a set of rules governing message headers, mnemonics, separators, data types and terminators. The following overview will familiarize the programmer with the fundamentals of these rules. As seen below, the rules for command and query messages are much more flexible than their precise response message counterparts.

<PROGRAM MESSAGE UNIT> - This is the basic message, and represents an operation to be performed by the device. As an example, if you wanted to create a sinewave with 5 cycles in the currently selected waveform, the appropriate <PROGRAM MESSAGE UNIT> would be:

```
WVFM:SINE 5,0
```

Notice that a colon (ASCII 3A) is used to separate the <program mnemonic> **WVFM** from **SINE**. For query responses the similarly structured <RESPONSE MESSAGE UNIT> is used. A complete list of <PROGRAM MESSAGE UNITS> and <RESPONSE MESSAGE UNITS> for the 2411A, with definitions, mnemonics, and limitations is given in Section 10.5.

General rules:

1. <program mnemonic> shall have maximum length of 12 characters.
2. ":" is used to separate <program mnemonics>, and when preceding a <PROGRAM MESSAGE UNIT>, it indicates that the following <program mnemonic> is at the root level.

<PROGRAM MESSAGE UNIT>

1. <program mnemonic> can be either upper or lower case alpha.
2. <white space> is only allowed at the end of the header, or next to a "," or a ";". (See note on <white space> below.)

<RESPONSE MESSAGE UNIT>

1. <program mnemonic> shall contain upper case alpha only.
2. No <white space> allowed in message.

Note: <white space> is defined as a length of 1 or more of <white space characters>. A <white space character> is a single ASCII byte in the range of 00-09, 0B-20 Hex.

<PROGRAM MESSAGE> - This is a series of <PROGRAM MESSAGE UNITS> sent to the device in a single string. For example, to set the currently selected waveform to 1) a sinewave output with a single period, 2) burst mode, 3)a burst of 5, and 4) an amplitude of 2.5V, you would send the following <PROGRAM MESSAGE>:

```
WVFM:SINE 1,0;:MODE BURST;:BURST 5;:AMPL 2.5;:EXEC
     { msg 1 }  { msg 2 }       { msg 3 }  { msg 4 }
```

The semicolon (ASCII 3B) is required to separate <PROGRAM MESSAGE UNITS> within a <PROGRAM MESSAGE>. For the 2411A a <PROGRAM MESSAGE> can be virtually any length. The structure for a <RESPONSE MESSAGE> for query responses is similar.

<PROGRAM MESSAGE TERMINATOR> or **<PMT>** - In order for the device to recognize the end of a <PROGRAM MESSAGE>, a special terminator is required. For command messages, the <PMT> can take one of three different formats:

> **^END** - This is defined as sending EOI TRUE and ATN FALSE with the last byte of the message.
>
> **NL** - NL represents newline and is an ASCII 0A.
>
> **NL^END** - A ^END sent along with NL as the last byte.

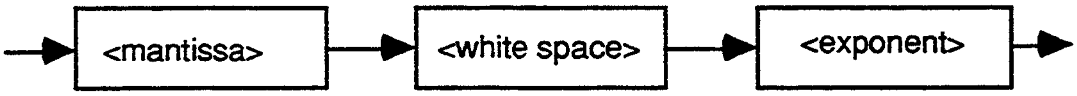
<RESPONSE MESSAGE TERMINATOR> or **<RMT>** - For query messages the required terminator is:

> **NL^END** - A ^END sent along with NL as the last byte.

10.4.5 Data Formats

Many of the <PROGRAM MESSAGE UNITS> and <RESPONSE MESSAGE UNITS> include numeric data in the message, e.g., the '5' in the message **SINE 5,0**. This section defines acceptable data formats, and Section 10.5 indicates which formats are used with specific commands and responses.

<DECIMAL NUMERIC PROGRAM DATA> or **<NRf>** - This is the most flexible of the numeric representations and takes the following general form:



where <mantissa> = [±][x...x][.x...x] with a maximum length of 255 characters (excluding leading zeros), and the

optional <exponent> = E[±][x...x] or e[±][x...x] , with a maximum value of ±32000.

Note in the above definitions: x represents digits 0-9
[ ] means enclosed characters are optional

The following example demonstrates several acceptable ways to represent the number 1,234,567,890 in <NRf> format:

+1234567890	**123456.7890**<white space>**e04**
1.234567890E+9	**+.1234567890E+10**

Three other numeric data formats are used in <RESPONSE MESSAGE UNITS>, and are subsets of the more general <NRf> format.

<NR1 NUMERIC RESPONSE DATA> or **<NR1>** -
[±][x...x]　　e.g. -98765432

<NR2 NUMERIC RESPONSE DATA> or **<NR2>** -
[±]x[x...x].x[x...x]　　e.g. +98765.432

<NR3 NUMERIC RESPONSE DATA> or **<NR3>** -
[±]x[x...x].x[x...x]E±x[x..x]　　e.g. 987.65432E+05

<NON-DECIMAL NUMERIC PROGRAM DATA> - Numeric values may also be represented as a binary, octal, or hex number, as follows:

Binary	- **#Bx**[x...**x**] or **#bx**[x...x]	where x is a **0** or **1**
Octal	- **#Qx**[x...x] or **#qx**[x...x]	where x is a **0** thru **7**
Hex	- **#Hx**[x...x] or **#hx**[x...x]	where x is a **0** thru **F**

<ARBITRARY BLOCK PROGRAM DATA> - This data format is used to speed bus transfer in cases where large amounts of data are sent to or from a device (such as waveform or buffer memory data in the 2411A). Both <INDEFINITE LENGTH> and <DEFINITE LENGTH> block data formats are acceptable.

<INDEFINITE LENGTH> - **#0**[x...x]<RMT> where x is an 8-bit byte of decimal value 0-255, and <RMT> is the message terminator.

<DEFINITE LENGTH> - **#zy..yx...x** where z is a number 1-9, and represents the number of y digit elements. The y digits taken together as a decimal integer equal the number of 8-bit bytes that follow.

For example to send 4 data bytes<DAB> using the <DEFINITE LENGTH> format, you could send:

#14<DAB><DAB><DAB><DAB>
or
#204<DAB><DAB><DAB><DAB>

Refer to Section 10.8 for detailed instructions on how to enter data into the waveform memory of the 2411A.

10.4.6 Error Reporting

There are four basic types of errors that are reported by a device:

Command - In general, when a <PROGRAM MESSAGE> is sent with an error in the syntax, a command error is reported. The command parser (the module that recognizes individual commands) will report the bad command, and look for the next valid command in sequence.

Execution - This error represents either program data which is out of range, or a message which was not properly executed due to some device condition. In this case the faulty command will generate the error, but not be performed.

Device Specific - As the name implies, this error is defined by the specific instrument. Currently there are no Device specific errors generated by the Model 2411A.

Query - When a controller or other device attempts to read data from the Output Queue when no data is present or pending, or when output data is lost, a query error is generated.

To clear an error:

1. Correct the condition which caused the error.
2. Send the ***CLS** command or read the standard event status register by using the ***ESR?** query command.

10.5 REMOTE COMMAND SET

10.5.1 Introduction

Most of the command set has an equivalent front panel counterpart. Therefore, it is highly recommended that the reader be thoroughly familiar with the front panel operation of the Model 2411A before beginning any programming.

*****IMPORTANT NOTE*****
In order to eliminate some of the common errors encountered while programming instruments, the Model 2411A has adopted a unique approach to sending and executing <PROGRAM MESSAGES>. The following discussions on **command sequence** and **command execution** explain this approach.

Command Sequence - Normally a series of commands (or <PROGRAM MESSAGE UNITS>) are sent as a single <PROGRAM MESSAGE> according to the rules outlined in Section 10.4. While these commands are generally executed sequentially, there are certain conditions where the absolute sequential execution of the commands would cause a device error. Take for example the following situation:

Output amplitude is 5Vpp, with offset of 0V

The new test setup calls for an output setting of 1Vpp with +7.5V offset.
(Note: Maximum output amplitude is 10Vpeak)

With most instruments that execute commands sequentially it would be required to first change the amplitude, then change the offset, in order to prevent a device error (caused by the sum of amplitude and offset exceeding 10.4Vpeak; see "ERROR CODES" in the Appendix). Because the Model 2411A is not sequence dependent (except for "WVFM" commands), the command order (within a single <PROGRAM MESSAGE>) is of no consequence . Thus both of the following messages would execute properly without a device error.

AMPL 1.0;OFFSET 7.5;EXEC
or
OFFSET 7.5;AMPL 1.0;EXEC

Command Execution - As already alluded to in the preceding paragraph, in order for the Model 2411A to recognize and execute a command, or series of commands, they usually must be followed by the **EXECUTE** command. This command is equivalent to the front panel <ENTER> button, and allows the programmer to send a complete <PROGRAM MESSAGE> into an input buffer, before executing any of the individual <PROGRAM MESSAGE UNITS>. The advantage of this method is two fold: 1) it allows the Model 2411A to process the commands very quickly as a group, rather than wait for the slower bus transfers to complete, and 2) it offers the non-sequence dependent benefits as outlined above.

10.5.2 Command Set Hierarchy

The command set of the Model 2411A uses a hierarchial structure similar to the file structure on many computer systems. Figure 10-4 shows an example of this structure.

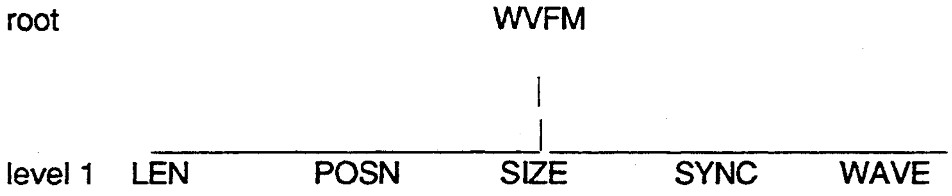


Figure 10-5. Command Hierarchy

While some instruments use several levels within the command set structure, the Model 2411A uses mostly one or two levels. The top level (represented by the mnemonic **WVFM**) is called the root, and the next lower level is level 1. With this structure you must follow a **path** through the root in order to reach the commands on level 1.

Referring to Figure 10-5, if we wanted to execute the command **SIZE**, we would need to indicate the path (through the root) as follows:

WVFM:SIZE 100;

Finally, it is important to note that 1) the path rules of the Model 2411A, allow the programmer to delete the root from the command if the level 1 command has the same root as the preceding command, and 2) if the preceding command is at level 1, you must specify any new root by using a ":" in front of the root mnemonic . To illustrate 1):

WVFM:WAVE 10;:WVFM:SIZE 100;
is the same as
WVFM:WAVE 10;SIZE 100;

10.5.3 Stacked Queries

In general the Model 2411A allows stacked queries, returning the responses in the same order in which the queries were received. The only exceptions to this are the ***IDN?** and ***OPT?** queries. Any queries that are placed after the ***IDN?** or ***OPT?** queries in a <PROGRAM MESSAGE> will be ignored.

10.5.4 Command Set

This section gives a complete explanation of all commands, and their structure for the Model 2411A. The following abbreviations are used for convenience:

<ws> - Whitespace as defined in Section 10.4.4

<NR> - This refers to the flexible numeric representation <NRf>, or non-decimal numeric data as defined in Section 10.4.5.

[] - Brackets indicate that the enclosed characters or parameters are optional. In the case of the command header either the long form or the short form may be used, but not both.

<arbblk> - This refers to arbitrary block data as described in Section 10.4.5.

Root Command [Short Form] ▸ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Common Commands			
***CLS**			Clears all event and status registers. Will also clear output queue if it immediately follows a <PMT>.
***ESE**<ws><8-bit mask> (GPIB only)	<NR>	0/255	Sets the 8-bit mask in the ESE register. (Ref. Section 10.4.3.)
***ESE?** (GPIB only)	<NR1>	0/255	Returns the contents of the ESE register.
***ESR?**	<NR1>	0/255	Returns the contents of the ESR register. Once read, the ESR register is cleared.
***IDN?**			The specific response to this query is: **Pragmatic Instruments, Model 2411A,0,** <firmware level>, where <firmware level> is of the form **Vx.xx** .
***OPC** (GPIB only)			Sets the OPC bit (bit 0) in the ESR register, when all pending instrument operations are complete.
***OPC?**			This query waits for all pending instrument operations to complete, then places an ASCII "1" in the output queue.
***RST**			Equivalent to the front panel RESET CURRent Key, this returns the instrument to a factory default state. (See Section 10.6.)
***SRE**<ws><8-bit mask> (GPIB only)	<NR>	0/255	Sets the 8-bit mask to enable/disable bits in the STB register. (Ref. Section 10.4.3)
***SRE?** (GPIB only)	<NR1>	0/255	Returns the value of the SRE register.
***STB?** (GPIB only)	<NR1>	0/255	Returns the value of the STATUS BYTE. The Master Summary Status bit (bit 6) is cleared with the first read, but all other bits remain unchanged until the conditions are cleared. (Ref. Section 10.4.3)
***TRG**			Equivalent to the front panel TRIG key.
***TST?**			Generally this is a selftest command, however it presently does not initiate any action, except to return an ASCII "1".

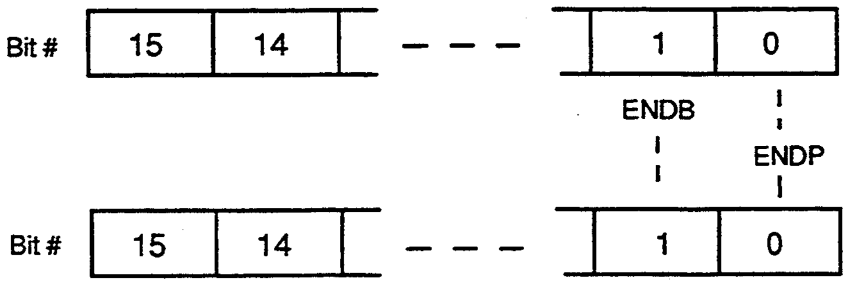
Root Command [Short Form] ➧ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Common Commands (cont.)			
*WAI			This Wait-to-Continue command has no effect, since commands are processed sequentially.
OPT?			No Option Installed: ASCII "0". Option(s) Installed: SEQ, GPIB. (ASCII "0" appears in any one option not installed.)
Configuration Commands CONFIGURE [CONF] ➧HEADERS[HDRS]<ws><state>	ON/OFF		With headers ON, query responses include the header. With headers OFF, responses return only the data.
➧ HEADERS?	0/1		Returns the current header configuration.
System Commands EXECUTE [EXEC]			Instructs the instrument to execute pending commands. Equivalent to the front panel ENTER key. (See Section 10.5.1.)
HOLD<ws><state>	ON/OFF		Holds or releases the present level of the output voltage. Equivalent to a front panel HOLD.
RECALL [RCLL]<ws><memory #>	<NR>	0/30	Recall front panel setups from specified memory.
REF_CLK_ADJ [RADJ]<ws><correction #>	<NR>	-128/ 127	Adjusts reference clock by the specified factor.
REF_CLK_ADJ [RADJ]?	<NR1>	-128/ 127	Returns current ref. clock adjustment factor.
REF_CLOCK[RCLK]<ws><state>	INT/EXT		Sets reference clock source to INTernal or EXTernal.
REF_CLOCK [RCLK]?	INT/EXT		Returns the present state of the reference clock source.
RESET	CURR/ ALL		Resets instrument settings to default values. (See Section 10.6.)
RTNTOSTRT [RTST]	ON/OFF		Returns to the starting point of the output waveform when ON. Equivalent to front panel RTS.
SAMPLECLOCK [SCLK]<ws><frequency>	<NR>	.1/2E6	Sets the sample clock frequency.

Root Command [Short Form] ♦ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
System Commands (cont.)			
SAMPLECLOCK [SCLK]?	<NR3>	0.1/2E6	Returns the present sample clock frequency.
STORE [STOR]<ws><memory #>	<NR>	0/30	Store front panel setups into specified memory.
TGENERATOR[TGEN]<ws><state>	ON/OFF		Sets trigger generator ON or OFF.
TGENERATOR [TGEN]?	ON/OFF		Returns the present state of the trigger generator.
TGRRATE [TGRR]<ws><rate>	<NR>	.02/10	Sets the trigger rate in seconds.
TGRRATE [TGRR]?	<NR2>	.02/10	Returns the current trigger rate.
TRIGGER [TRIG]<ws><state>	ON/OFF/ PULSE		Sets the trigger ON, OFF, or PULSEd.

Instrument Status Commands

The 16-bit Instrument Status Event Register and Instrument Status Enable Register are laid out as follows. The summarized status is routed to Bit #0 (USR0) in the STB Register.

Instrument Status Event Register



Instrument Status Event Enable Register

Bit #15......Bit #2 - not used

ENDP - End Pulse, set high every time a waveform END point is detected.

ENDB - End Block, set high every time an END of waveform burst is detected.

Root Command [Short Form] ♦ Level 1 Command [Short Form] ♦Level 2 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Instrument Status Commands **STATUS** **♦INSTRUMENT [INST]**			
♦CLEAR [CLR]			Clears Instrument Status Register
♦ENABLE [ENBL]	<NR>	0/65535	Sets the Mask in the Instrument Status Event Register. Bit#0 enables ENDP (End Pulse) and Bit#1 enables ENDB (End Block).
♦ENABLE? [ENBL?]	<NR1>	0/65535	Returns the contents of the Instrument Status Event Enable Register.
♦EVENT? [EVNT?]	<NR1>	0/65535	Returns the contents of the Instrument Status Event Register. BIT #0: ENDP (End Pulse) BIT #1: ENDB (End Block)

Root Command [Short Form] ▸ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Output Commands			(Amplitude levels are referenced from 50Ω source impedance into 50Ω load impedance.)
AMPLITUDE [AMPL]<ws><peak-to-peak>	<NR>	.010/10.2	Sets the peak-to-peak output voltage.
AMPLITUDE [AMPL]?	<NR2>	.010/10.2	Returns the peak-to-peak output voltage.
BURST<ws><burst #>	<NR>	1/1048575	Sets the burst count.
BURST?	<NR1>	1/1048575	Returns the present burst number setting.
CLOCK_SEL [CLKSEL]<ws><state>	INT/EXT		Selects either INTernal or EXTernal sample clock source.
CLOCK_SEL [CLKSEL]?	INT/EXT		Returns the present setting for the sample clock source.
DEGL	ON/OFF		Turn the Deglitcher ON or OFF.
DEGL?	ON/OFF		Returns the state of the Deglitcher.
FILTER<ws><state>	LPF40k LPF 700k OFF		Sets output low pass filter to 40k Sets output low pass filter to 700k Turn off output filter.
FILTER?	LPF40k LPF700k OFF		Returns the present output filter state.
FREQUENCY [FREQ]?	<NR3>	0/2E6	Returns the calculated output frequency.
FUNCTION [FUNC]<ws><type>, <wav/seq #>	WAVE/ SEQ <NR> or STDW	WAVE: 0/99 SEQ: 0/99	Selects the specified WAVEform or SEQuence number to send to the output, e.g., FUNC WAVE,1; FUNC WAVE,STDW.
FUNCTION [FUNC]?	WAVE/ SEQ <NR1> or STDW		Returns the currently selected WAVEform or SEQuence number in the form **WAVE[SEQ],<#** or **STDW**>.
MODE<ws><setting>	CONT/ TRIG/ GATE/ BURST/ TOGGLE		Sets the output signal mode to CONTinuous, TRIGgered, GATEd, BURST, or TOGGLEd.
MODE?	" "		Returns present mode of the output signal.
OFFSET [OFST]<ws><level>	<NR>	-10.2/10.2	Sets output offset voltage.
OFFSET [OFST]?	<NR2>	-10.2/10.2	Returns present offset voltage value.
OUTPUT_SWTCH [OUTSW]<ws><state>	ON/MUTE		Turns ON, or MUTEs the output.

Root Command [Short Form] ♦ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Output Commands (cont.)			
OUTPUT_SWTCH [OUTSW]?	ON/MUTE		Returns the state of output switch.
READ_BURST [RBRS]?	<NR1>	1/1048575	Returns the value of the completed burst count
SYNCSEL [SYSEL]<ws><sync #>, <state> Alternately, SYNC 1: ADDR/ENDP SYNC 3: ADDR/WRUN/DAC SYNC 4: ADDR/ENDB	<NR> ADDR/ STATE	1,3,4	Sets the selected SYNC pulse to either an ADDRess, or specific STATE within the waveform. For SYNC1, STATE inserts a pulse at the waveform END Point (or ENDP). For SYNC3, STATE sets the sync pulse high during Waveform RUN (WRUN), and for SYNC4, STATE sets the sync pulse at the END point of each waveform Burst (ENDB) within a sequence. (Ref. SYNC under Waveform Edit Commands.)
SYNCSEL [SYSEL]?<ws><sync #> Query Response SYNC 1: ADDR/ENDP SYNC 3: ADDR/WRUN/DAC SYNC 4: ADDR/ENDB	<NR>	1,3,4	Returns the present state (ADDR or STATE) of the specified SYNC pulse.
TRGINMODE<ws><state>	SYNC/ ASYNC		Sets the trigger input mode to SYNChronous or ASYNChronous. (See Sec. 8)
TRGINMODE?	" "		Returns the present mode of the input trigger.
TRGOUTMODE<ws><state>	SERIAL/ PARALLEL		For multi-instrument triggering, sets outputs to trigger in serial or parallel. (See Sec8)
TRGOUTMODE?	" "		Returns the current output trigger mode.
Waveform Edit Commands *(For Arbitrary Waves only)* **WAVEFORM [WVFM]**			**REFER TO SECTION 10.7 FOR AN OVERVIEW OF WAVEFORM EDITING.**
♦**AM**<ws><# of carrier cycles>, <starting carrier phase>, <# of modulation cycles>, <starting mod. phase>, <modulation index>	<NR> <NR> <NR> <NR> <NR>	0/1E4 -360/360 0/1E4 -360/360 0/200	Generates a sinewave amplitude-modulated by a sinewave, in the selected waveform memory. (Ref. **WAVE** command.) The first two parameters specify the carrier characteristics, and the second two the modulating waveform. Modulation index can vary from 0 to 200%.
♦ **CIRCLE**<ws><# of cycles>, <starting phase>, <invert>	<NR> <NR> NORM/ INVERT	0/1E4 -360/360	Generates a semicircle in the selected waveform memory. The first parameter specifies the number of cycles, while the second specifies starting phase in degrees. The keyword sets the first part of the waveform NORMal or INVERTed (first half cycle positive or negative, resp.).

Root Command [Short Form] ♦ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Waveform Edit Commands (cont.) *(Arbitrary)* **WAVEFORM [WVFM]**			
♦ **DC**<ws><Y value>	<NR>	-32768/ 32767	Generates a horizontal line at "Y value" in the selected waveform memory.
♦ **EXPONENTIAL** **[EXP]**<ws><exponent>, <invert>	<NR> NORM/ INVERT	0/20	Generates a decaying exponential with the specified exponent, e^{-x}, in the selected waveform memory. The keyword sets the waveform to a NORMal or INVERTed (positive going or negative going, resp.) decay.
♦ **FM**<ws><# of carrier cycles>, <starting carrier phase>, <# of modulation cycles>, <starting mod. phase>, <modulation index>	<NR> <NR> <NR> <NR> <NR>	0/1E4 -360/360 0/1E4 -360/360 0/1000	Generates a sinewave frequency-modulated by a sinewave, in the selected waveform memory. The first two parameters specify the carrier characteristics, and the second two the modulating waveform. Modulation index can vary from 0 to 1000.
♦**GAUSSIAN** **[GAUSS]**<ws><exponent>,	<NR>	0/20	Generates a gaussian pulse with the specified exponent, e^{-x^2}. Where x varies between +/- <exponent>.
♦**HAVERSINE** **[HSIN]**<ws><#cycles>	<NR>	0/1E4	Generates a haversine wave, with the number of cycles specified. The basic shape of this waveform is a sinewave shifted by -90^{o}.
WAVEFORM [WVFM] ♦ **LINE**<ws><starting Y value>, <ending Y value>	<NR>	-32768/ 32767	Generates a straight line in the selected waveform memory, with the specified starting and ending Y coordinates.
♦**LINEARSWEEP** **[LINS]**<ws><starting # of cycles>, <ending # of cycles>, <starting phase>	<NR>	.001/1E4 .001/1E4 -360/360	Generates a linearly-swept sinewave, with the number of starting and ending cycles as specified.
♦ **LOGSWEEP** **[LOGS]**<ws><starting # of cycles>, <ending # of cycles>, <starting phase>	<NR>	.001/1E4 .001/1E4 -360/360	Generates a logarithmically-swept sinewave, with the number of starting and ending cycles as specified.
♦ **NOISE**	<NR>		Generates psuedo-random noise in the selected waveform memory. (Ref. **WAVE** command.)

Root Command [Short Form] ▸ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Waveform Edit Commands (cont.)			
(Arbitrary)			
WAVEFORM [WVFM]			
▸ **PULSE**<ws><# of pulses>,	<NR>	1/1E4	Generates a pulse train, with the number of pulses as specified in the first parameter. Delay, rise/high/falltime are all expressed in percentages of the period of the pulse.
<delay>,	<NR>	0/100	
<risetime>,	<NR>	0/100	
<high time>,	<NR>	0/100	
<falltime>,	<NR>	0/100	
<invert>	NORM/ INVERT		
▸ **SAWTOOTH**			
[SAW]<ws><# of cycles>,	<NR>	1/1E4	Generates a sawtooth waveform with the number of cycles as specified. The keyword sets the sawtooth to a NORMal (rising) or INVERTed (falling) waveform.
<duty cycle>	<NR>	0/100	
<invert>	NORM/ INVERT		
▸ **SINE**<ws><# of cycles>,	<NR>	.001/1E4	Generates a sinewave with the number of cycles as specified.
<starting phase>	<NR>	-360/360	
▸**SCM**<ws><# of carrier cycles>	<NR>	0/1E4	Generates a sinewave amplitude-modulated waveform with suppressed carrier, in the selected waveform memory. (Ref. **WAVE** command.) The first two parameters specify the carrier characteristics, and the second two the modulating waveform.
<starting carrier phase>	<NR>	-360/360	
<# of modulation cycles>	<NR>	0/1E4	
<starting modulation phase>	<NR>	-360/360	
▸ **SQUARE**<ws><# of cycles>,	<NR>	1/1E4	Generates a squarewave with the number of cycles as specified. The keyword sets the first half of the squarewave high (NORMal), or low (INVERTed).
[SQU] <duty cycle>	<NR>	0/100	
<invert>	NORM/ INVERT		
▸ **SINE_X_OVR_X**			
[SXX] <ws><# of cycles>,	<NR>	4/1E4	Generates the function sine(x)/x as a waveform, with the number of specified cycles. The keyword sets the waveform NORMal or INVERTed.
<invert>	NORM/ INVERT		
▸**TRIANGLE**<ws><# of cycles>,	<NR>	1/1E4	Generates a triangle wave, in the currently selected waveform memory, with the number of specified cycles. The keyword sets the output to initially rise (NORMal), or fall (INVERTed).
[TRI] <invert>	NORM/ INVERT		
▸ **LENGTH [LEN]**<ws><length>	<NR>	0/65504	Sets the length (in data points) that any succeeding waveform generation function will create, for the selected waveform. (Ref. to **SIZE** and **POSITION** commands, and Section 10.7.) Functional limits are 0 to SIZE-POSITION, otherwise a device error is generated.

Root Command [Short Form] ♦ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Waveform Edit Commands (cont.) *(Arbitrary)* WAVEFORM [WVFM]			
♦ LENGTH [LEN]?	<NR1>	0/65504	Returns the current value for LENGTH.
♦ MAXY<ws><value>	<NR>	-32768/ 32767	Selects the maximum Y value to be produced when generating a waveform. This command is not valid when the standard waveform is selected.
♦ MAXY?	<NR1>	-32768/ 32767	Returns the currently selected maximum Y value.
♦ MINY<ws><value>	<NR>	-32768/ 32767	Selects the minimum Y value to be produced when generating a waveform. This command is not valid when the standard waveform is selected.
♦ MINY?	<NR1>	-32768/ 32767	Returns the currently selected minimum Y value.
♦ POSITION [POSN]<ws><write position>	<NR>	0/65503	Sets the starting position, in the currently selected waveform memory, where new waveform points will be written. The maximum starting position is the size of the memory -1. (Ref. **SIZE** command.) This command is not valid when the standard waveform is selected. After a function such as SINE, etc. is written, **POSITION is automatically incremented** to "POSITION SIZE" to point to the next "new" data point.
♦ POSITION [POSN]?	<NR1>	0/65503	Returns the current starting position for writing into waveform memory.
♦ SIZE<ws><waveform size>	<NR>	0, 32/65504	Sets the memory size of the currently selected waveform, in number of points, . The size can be from zero to the total amount of free memory space. If the selected waveform is the standard waveform (STDW), the existing waveform is stretched or squeezed to fit the new size. If the selected waveform is other than the STDW, if enlarging the size, new points (set to 0) are added at the end of the waveform reducing the size. Sending "zero" size will delete waveform.
♦ SIZE?	<NR1>	0/65504	Returns the present value of SIZE.

Root Command [Short Form] ▸ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Waveform Edit commands (cont.) *(Arbitrary)* **WAVEFORM [WVFM]**			
▸ **SYNC**<ws><sync #>, <start position>, <length>	<NR> <NR> <NR>	1,3,4 0/65503 0/65504	Installs a sync pulse into the specified channel. The start position indicates where in the selected waveform memory the pulse begins, and length specifies the total length of the pulse. Start position can be from 0 to SIZE - 1, while length can range from 0 to SIZE - POSITION.
▸ **SYNC?**<ws><sync #> <start position>, <length>	<NR1> <NR1> <NR1>	1,3,4 0/65503 0/65504	Returns the starting position, and length of the specified sync pulse.
▸ **WAVE**<ws><waveform #>, <selector>	<NR> or STDW WVFM	0/99	Selects either the Standard Waveform (STDW) or one of the numbered locations within the waveform memories. POSITION is set to 0, and LENGTH is set to SIZE. (See Sec. 10-7)
▸ **WAVE?**	<NR1> or STDW WVFM	0/99or STDW	Returns the number of the currently selected waveform memory.

Waveform Edit Commands
(For Standard [STDW] Waves only)

Following are Waveform Edit Commands for Standard (STDW) Waveforms as opposed to the Arbitrary Waveforms described in the preceding section. General rules for STDW drawing commands are:

1. For STDW, the same parameter limits and function drawing rules from the front panel apply.

2. The following commands do NOT apply for STDW:

   POSN, LEN, MINY, MAXY, LINE, MEM

3. The <invert> flag is not allowed except for the following commands:

   EXP, SAW, PULSE

For example:

:wvfm:exp 5, **NORM** is used to draw "Exp+" function
:wvfm:exp 5, **INVERT** is used to draw "Exp-" function

Root Command [Short Form] ▸ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Waveform Edit Commands (cont.) *(Standard [STDW])* (cont.) **WAVEFORM [WVFM]**			**REFER TO SECTION 10.7 FOR AN OVERVIEW OF WAVEFORM EDITING.**
▸ **AM**<ws><# of carrier cycles>, <starting carrier phase> <# of modulation cycles>, <starting mod. phase> <modulation index>	<NR> <NR> <NR> <NR> <NR>	0/1E4 0/360 0/1E4 0/360 0/200	Generates a sinewave amplitude-modulated by a sinewave, in the selected waveform memory. (Ref. **WAVE** command.) The first two parameters specify the carrier characteristics, and the second two the modulating waveform. Modulation index can vary from 0 to 200%.
▸ **CIRCLE**<ws><# of cycles>, <starting phase>	<NR> <NR>	0/1E4 0/360	Generates a semicircle in the selected waveform memory. The first parameter specifies the number of cycles, while the second specifies starting phase in degrees.
▸ **DC**<ws><Y value>	<NR>	-32768/ 32767	Generates a horizontal line at "Y value" in the selected waveform memory.
▸ **EXPONENTIAL** **[EXP]**<ws><exponent>, <invert>	<NR> NORM/ INVERT	0/20	Generates a decaying exponential with the specified exponent, e^{-x}, in the selected waveform memory. The vertical range of the waveform is always between -32768 and 32767. The keyword sets the waveform to a NORMal or INVERTed (positive going or negative going, respectively) decay.
▸ **FM**<ws><# of carrier cycles>, <starting carrier phase> <# of modulation cycles>, <starting mod. phase> <modulation index>	<NR>	0/1E4 0/360 0/1E4 0/360 0/100	Generates a sinewave frequency-modulated by a sinewave, in the selected waveform memory. The first two parameters specify the carrier characteristics, and the second two the modulating waveform. Modulation index can vary from 0 to 100.
▸ **GAUSSIAN** **[GAUSS]**<ws><exponent>,	<NR>	0/20	Generates a gaussian pulse with the specified exponent, e^{-x^2}. Where x varies between +/- <exponent>.
▸ **HAVERSINE** **[HSIN]**<ws><#cycles>	<NR>	0/1E3	Generates a haversine wave, with the number of cycles specified. The basic shape of this waveform is a sinewave shifted by -90°.
▸ **LINEARSWEEP** **[LINS]**<ws><starting # of cycles>, <ending # of cycles>, <constant>	<NR> <NR> <NR>	1/1E3 1/1E3 0/360	Generates a linearly-swept sinewave, with the number of starting and ending cycles as specified. "Constant" must be sent as a third parameter, but will be ignored.

Root Command [Short Form] ▸ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Waveform Edit Commands (cont.) *(Standard [STDW])* (cont.) **WAVEFORM [WVFM]**			
▸ LOGSWEEP			
[LOGS]<ws><starting # of cycles>,	<NR>	.001/1E3	Generates a logarithmically-swept sinewave, with the number of starting and ending cycles as specified. "Constant" must be sent as third parameter, but will be ignored.
<ending # of cycles>,	<NR>	.001/1E3	
<constant>	<NR>	0/360	
▸ NOISE			Generates pseudo-random noise in the selected waveform memory. (Ref. **WAVE** command.)
▸ PULSE<ws><# of pulses>,	<NR>	1/1E3	Generates a pulse train, with the number of pulses as specified in the first parameter. Delay, rise/high/falltime are all expressed in percentages of the period of the pulse. <Invert> specifies whether the pulses will be inverted or not.
<delay>,	<NR>	0/100	
<risetime>,	<NR>	0/100	
<high time>,	<NR>	0/100	
<falltime>,	<NR>	0/100	
<invert>	NORM/ INVERT		
▸ SAWTOOTH			
[SAW]<ws><# of cycles>,	<NR>	1/1E3	Generates a sawtooth waveform with the number of cycles as specified. Duty cycle is 50% unless set otherwise. <Invert> sets the sawtooth to a NORMal (rising) or INVERTed (falling) waveform.
<duty cycle>,	<NR>	0/100	
<invert>	NORM/ INVERT		
▸ SINE<ws><# of cycles>,	<NR>	0/1E3	Generates a sinewave with the number of cycles as specified.
<starting phase>	<NR>	0/360	
▸ SCM<ws><# of cycles>	<NR>	0/1E4	Generates a sinewave amplitude-modulated waveform with suppressed carrier. (Ref. **WAVE** command.) The first two parameters specify the carrier characteristics, and the second two the modulating waveform.
<starting carrier phase>	<NR>	0/360	
<# of modulation cycles>	<NR>	0/1E4	
<starting modulation phase>	<NR>	0/360	
▸ SQUARE<ws><# of cycles>,	<NR>	1/1E3	Generates a squarewave with the number of cycles as specified.
[SQU] <duty cycle>,	<NR>	0/100	
▸ SINE_X_OVR_X **[SXX]**<ws><# of cycles>,	<NR>	4/1E3	Generates the function sine(x)/x as a waveform, with the number of specified cycles.
▸ TRIANGLE<ws><# of cycles>, **[TRI]**	<NR>	1/1E3	Generates a triangle wave, in the currently selected waveform memory, with the number of specified cycles.
▸ LENGTH [LEN]?	<NR1>	0/65536	Returns the current value for length.

Root Command [Short Form] ➧ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Waveform Edit Commands (cont.) *(Standard [STDW])* (cont.) **WAVEFORM [WVFM]**			
➧ **SIZE**<ws><waveform size>	<NR>	32/65536	Sets the memory size of the currently selected waveform, in number of points. The size can be from 32 to the total amount of free memory space. If the selected waveform is the standard waveform (STDW), the existing waveform is stretched or squeezed to fit the new size. If the selected waveform is other than the STDW, if enlarging the size, new points (set to 0) are added at the end of the waveform.
➧ **SIZE?**	<NR1>	0, 32/65536	Returns the present value of SIZE.
➧ **SYNC**<ws><sync #>, <start position>, <length>	<NR> <NR> <NR>	1,3,4, 0/65535 0/65536	Installs a sync pulse into the specified channel. The start position indicates where in the selected waveform memory the pulse begins, and length specifies the total length of the pulse. Start position can be from 0 to SIZE - 1, while length can range from 0 to SIZE - POSITION.
➧ **SYNC?**<ws><sync #> <start position>, <length>	<NR1> <NR1> <NR1>	1,3,4 0/65535 0/65536	Returns the starting position, and length of the specified sync pulse.
➧ **WAVE**<ws><waveform ?#>, <selector>	<NR> or STDW	0/99 (WVFM), or STDW	Selects either the Standard Waveform (STDW) or one of the numbered locations within the waveform or buffer memories. POSITION is set to 0, and LENGTH is set to SIZE. (See Sec. 10-7)
➧ **WAVE?**	<NR1> or STDW	0/99 or STDW	Returns the number of the currently selected waveform number or the Standard Waveform (STDW).

Root Command [Short Form] ♦ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Waveform Transfer Commands **WAVEFORM [WVFM]**			
♦ MEMORY **[MEM]**<ws><address>, <data> [,...]	<NR> <NR> or <arbblk>	0/65504 -32768/ 32767	**This command applies to all waveforms except STDW.** Sends either individual data points or a block of data into the selected wave- form memory, beginnning at the address specified. The data block may be sent either as individual data points in the <NR> format, or as an arbitrary block of data, high byte first (Ref Section 10.7).
♦ MEMORY **[MEM]?**<ws><address>	<NR>	0/65503 for WVFM	Returns a single word of data in the range of -32768 to 32767, beginning at the specified address. (For both ARB and STDW waves).
♦ MEM_BLOCK **[MBLK]?**<ws><address>, <length>	<NR> <NR>	0/65503 0/2048	Returns the number of data points as specified in length (each data point is a 2-byte word, high byte first), beginning at the address as stipulated. The response is made up of the data in the definite length arbitrary block data format (Ref. Section 10.7)

Root Command [Short Form] ♦ Level 1 Command [Short Form]	Data Format	Limits Min/Max	Command Description
Sequence Generator Commands (See Section 10.5.5 for Applications)			
WAVEFORM [WVFM]			
♦ ADDSEQUENCE **[ADDSEQ]**<ws><sequence #>, <waveform #>, <burst count>, <sequence step #> [,...]	<NR> <NR> <NR> <NR>	0/99 0/99 0/1048575 0/999	Adds to the specified sequence a series of waveforms. Waveform parameters come in triplets. The first specifies the number of the waveform, the second, the number of times the waveform is repeated, and the third indicates the sequence step # for that waveform.
♦ ADDSEQUENCE? **[ADDSEQ]?**<ws><sequence #>, <step #>	<NR1> <NR1>	0/99 0/999	Returns the waveform #, burst count, and step number (all as integers) of the specified sequence and step number.
♦ AUTOSEQUENCE **[AUTO]**<ws><1st step #>, <step increment>	<NR> <NR>	0/999 0/999	Configures the automatic sequence step generator to begin at the selected first sequence step, and increment by the step number to the next sequence step number. This command is used in conjunction with **SEQ, SEQB.**
♦SEQUENCE **[SEQ]**<ws><sequence #> [,<waveform #> etc.]	<NR> <NR>	0/99 0/99	Constructs a sequence with the specified sequence number, composed of a series of waveforms as specified. The burst count is set to 1 for each waveform. *If no waveform number is supplied, the sequence is deleted.*
♦ SEQBURST **[SEQB]**<ws><sequence #>, <waveform #>, <burst count> [,...]	<NR> <NR> <NR>	0/99 0/99 0/1048575	Constructs a sequence with the specified sequence number, composed of a series of waveforms. Waveform parameters come in pairs, with the first being the number of the next waveform in sequence, and the second being the number of times the waveform is repeated.
♦ SEQBURSTNUM **[SEQBN]**<ws><sequence #>, <waveform #>, <burst count>, <sequence step #> [,...]	<NR> <NR> <NR> <NR>	0/99 0/99 0/1048575 0/999	Constructs a sequence with the specified sequence number, composed of a series of waveforms. Waveform parameters come in triplets. The first specifies the number of the waveform, the second, the number of times the waveform is repeated, and the third indicates the sequence step # for that waveform.

10.5.5 Sequence Generator Application Notes

ADDSEQUENCE [ADDSEQ] adds to the specific sequence file, one or more sequence steps. This is the only way to modify an existing sequence file using GPIB or RS-232 once it has been entered. The first parameter is the sequence file to be modified. After the first parameter, the following parameters always come in triplets. The first of these three being the number of the waveform to be sequenced. The second is the number of times this waveform will be repeated, and the third indicates the sequence step # for that waveform.

Example:
Assume that sequence file #10 already exists and steps 35 and 53 are NOT used. The following command will add two steps (35 and 53) to sequence #10. Step #35 will be wave #3 repeated 5 times. Step #53 will be wave #2 repeated 4 times.

```
WVFM:ADDSEQ 10,3,5,35,2,4,53
```

AUTOSEQUENCE [AUTO] confiures the automatic sequence step number generator by setting the sequence step number to start at first parameter and the increment value for the following sequence step numbers. This command is used for the SEQUENCE [SEQ] and SEQBURST [SEQB] commands only.

Example:
The following command will start numbering the sequence steps at 10 and increment them by 10, i.e., the first sequence step will be 10, the second sequence step will be 20, etc.

```
WVFM:AUTO 10,10
```

SEQUENCE [SEQ] makes a new sequence file, DELETING the existing sequence file if one already exists. The first parameter is the sequence file to create. The following parameters are the waves that are to be sequenced. For the SEQUENCE [SEQ] command, the Burst count (the number of times the wave is repeated) is always set to one. If no waves are supplied, then this command will only delete the specified waveform. The AUTO command is used in conjunction with this command.

Example:
The following command will delete sequence file number 23 (if it exists).

```
WVFM:SEQ 23
```

The following commands will create a sequence file (#12) that will sequence waves 3, 1, 5, 8 and 3. Step 10 will be wave 3 repeated 1 time. Step 20 will be wave 1 repeated 1 time. Step 30 will be wave 5 repeated 1 time. Step 40 will be wave 8 repeated 1 time. Step 50 will be wave 3 repeated 1 time.

```
AUTO 10,10
WVFM:SEQ 12,3,1,5,8,3
```

SEQBURST [SEQB] makes a new sequence file, DELETING the existing sequence file if one already exists. The first parameter is the sequence file to create. The following parameters are pairs, the first parameter of the pair is the wave number, the second parameter is the burst count (the number of times the wave is repeated). The AUTO command is used in conjunction with this command.

Example:
The following commands will create sequence file #5. Step 10 will be wave 2 repeated 4 times. Step 20 will be wave 6 repeated 1,000 times. Step 30 will be wave 45 repeated 10,000 times.

```
AUTO 10,10
WVFM:SEQB 5,2,4,6,1000,45,10000
```

SEQBURSTNUM [SEQBN] makes a new sequence file, DELETING the existing sequence file if there is one. The first parameter is the sequence file to create. The following parameters are in triplets. First of the three is the wave number, second is the burst count (the number of times the wave is repeated), and the third and last indicates the sequence step number.

Example:
The following command will create sequence file #23. Step 5 will be wave 1 repeated 2 times. Step 10 will be wave 3 repeated 4 times. Step 15 will be wave 5 repeated 6 times.

```
WVFM:SEQBN 23,1,2,5,3,4,10,5,6,15
```

10.5.6 Programming Example

The following programming example shows how to communicate with the 2411A over the GPIB, using a National Instruments AT-GPIB card, installed in an IBM compatible PC.

The program illustrates in Quick Basic the following:

1) Read 40 words of data, one at a time from the selected memory.
2) Read a block of 10 data words from the selected memory.
3) Create 3 waveforms in waveform memories 0 to 2, then set up these waveforms in a sequence with a single burst for each waveform.

Quick Basic Programming Example

```
DECLARE SUB SendArb (ArbString$)
DECLARE SUB WaitDelay (Sec!)
REM $INCLUDE: 'qbdecl.bas'
COMMON SHARED Arb%
CrLf$ = CHR$(13) + CHR$(10)

GOSUB InitGpibResetArb

StartProgram:
DO
        PRINT "Read 40 points using mem command"
        FOR Address = 0 TO 39
            CALL ibwrt(Arb%, "wvfm:mem?" + STR$(Address) + CrLf$)
            GOSUB ReadArb
            PRINT Address; ","; ArbString$,
        NEXT Address

        PRINT "Read a block of 10 using Mblk command"
            CALL ibwrt(Arb%, "wvfm:Mblk? 0,10" + CrLf$)
            GOSUB ReadArb
            PRINT "Response : >> "; ArbString$; " <<"

        PRINT "Load Sinewave into waveform 0"
            CALL ibwrt(Arb%, "wvfm:Wave 0;MaxY 32767;MinY -32768;PosN 0;Sine
            1,0" + CrLf$)
            CALL ibwrt(Arb%, ":Func Wave,0;:exec" + CrLf$)
            GOSUB WaitForInput

        PRINT "Load Line into waveform 1"
            CALL ibwrt(Arb%, "wvfm:Wave 1;MaxY 32767;MinY -32768;PosN 0;Line
            32767,-32768" + CrLf$)
            CALL ibwrt(Arb%, ":Func Wave,1;:exec" + CrLf$)
            GOSUB WaitForInput

        PRINT "Load LinearSweep into waveform 2"
            CALL ibwrt(Arb%, "wvfm:Wave 2;MaxY 32767;MinY -32768;PosN 0;LinS
            1,100,0" + CrLf$)
            CALL ibwrt(Arb%, ":Func Wave,2;:exec" + CrLf$)
            GOSUB WaitForInput
            PRINT "Now sequence waveforms 0 to 2"
                  CALL ibwrt(Arb%, "Wvfm:Auto 10,10" + CrLf$)
                  CALL ibwrt(Arb%, "Wvfm:SeqB 1,0,1,1,1,2,1" + CrLf$)
                  CALL ibwrt(Arb%, ":Func Seq,1;:exec" + CrLf$)
                  GOSUB WaitForInput

EndProgram:
STOP
LOOP
END

WaitForInput:
               CALL ibloc(Arb%)
              INPUT ">> Hit Return to Continue <<"; Scratch$
             RETURN

ReadArb:
            TmpStr$ = SPACE$(100)      'allocate space for gpib response
            WaitDelay (.075)
            CALL ibrd(Arb%, TmpStr$)   'read Arb
            IF ibcnt% > 1 THEN ArbString$ = LEFT$(TmpStr$, ibcnt% - 1) ELSE ArbString$ =
            "TimedOut"
            RETURN

InitGpibResetArb:
            PRINT
            PRINT "Initializing GPIB & resetting ARB"
            PRINT
            CALL IBDEV(0, 16, 0, 12, 1, 0, Arb%)'connect Arb at add 16
            PRINT Arb%
            CALL SendArb("*RST")
            CALL SendArb("*CLS")
            RETURN

SUB SendArb (OutputString$)
            CALL ibwrt(Arb%, OutputString$)
            CALL ibwrt(Arb%, "*ESR?")
            InputString$ = SPACE$(100)
            CALL ibrd(Arb%, InputString$)
            IF VAL(InputString$) = 4 THEN PRINT ">>QUERY ERROR<<": CALL ibwrt(Arb%,
            "*CLS")
            IF VAL(InputString$) > 0 THEN PRINT ">>ESR ERROR<< read :"; VAL(InputString$):
            CALL ibwrt(Arb%, "*CLS")
            END SUB

SUB WaitDelay (Sec)
 StrtTime = TIMER
 CurrentTime = TIMER
 StpTime = StrtTime + (Sec)
 DO WHILE CurrentTime < StpTime
 CurrentTime = TIMER
 LOOP
'    PRINT StrtTime, CurrentTime
END SUB
```

10.6 RESET AND FACTORY DEFAULTS

The following Table shows the parameter values which are affected for both factory default, and reset setting conditions. Factory default can be accessed via the front panel RESET key, and selecting the ALL function, or over the GPIB with the **RESET-ALL** command. Reset settings can be accessed either from the front panel RESET, CURRent function or over the GPIB with the ***RST** or **RESET-CURR** commands.

Parameter	Default Value	Factory Set	*RST
Reference Clock Adjust	0	√	√
Reference Clock Select	INTernal	√	√
Sample Clock Freq.	1MHz	√	√
Sample Clock Select	INTernal	√	√
Trigger Generator Rate	50 ms	√	√
Trigger Generator State	OFF	√	√
Amplitude	5.000 V	√	√
Burst Count	3	√	√
Filter	OFF	√	√
Function	STDW Wave	√	√
Mode	CONTinuous	√	√
Offset	0 V	√	√
Output Switch	MUTE	√	√
Trigger Input Mode	ASYNChronous	√	√
Trigger Output Mode	SERIAL	√	√
Waveform (Edit) Functions:			
AM <carrier cycles>	20	√	√
<phase>	0	√	√
<mod. cycles>	1	√	√
<phase>	0	√	√
<mod. index>	100%	√	√
Circle	1 cycle, 0 phase	√	√
DC	0	√	√
Exponential	+5	√	√
FM <carrier cycles>	20	√	√
<phase>	0	√	√
<mod. cycles>	1	√	√
<phase>	0	√	√
<mod. index>	10	√	√
Gaussian Exponent	2.00	√	√
HSIN	1.00	√	√
Linear Sweep			
<start cycles>	1	√	√
<end cycles>	10	√	√
Log Sweep			
<start cycles>	1	√	√
<end cycles>	10	√	√

Parameter	Default Value	Factory Set	*RST
Pulse <# of pulses>	1	√	√
<delay>	0%	√	√
<risetime>	10%	√	√
<hightime>	30%	√	√
<falltime>	10%	√	√
SCM <carrier cycles>	20	√	√
<phase>	0	√	√
<mod. cycles>	1	√	√
<phase>	0	√	√
Sawtooth <cycles>	1	√	√
<duty cycle>	100%: SAW+	√	√
	0: SAW-	√	√
Sine	1 cycle, 0 phase	√	√
Sine X Over X	1 cycle	√	√
Square <cycles>	1	√	√
<duty cycle>	50%	√	√
Triangle	1 cycle	√	√
Sync Sel (1,3 & 4)	ADDRess	√	√
Waveform Memory:			
WAVE	00,01,02,03,04		
Size	2000 points	√	Unchanged
Data	Affected	√	Unchanged
SYNC start/length	0/1	√	Unchanged
STDW			
Size	1000 points	√	Unchanged
Data	1 cycle sinewave	√	Unchanged
SYNC start/length	0/1	√	Unchanged
Sequence Generator:			
SEQ	000,001,002	√	Unchanged
GPIB			
Address	16	√	Unchanged
RS-232C			
BAUD	1.2 k	√	Unchanged
PAR	none	√	Unchanged
BITS	8d1s	√	Unchanged
HAND	sw	√	Unchanged

10.7 WAVEFORM EDITING PRINCIPLES

In order to successfully apply the waveform editing commands of the Model 2411A, it is important to understand the principles behind memory allocation, size and the active memory area. For the following discussion refer to Figure 10-6.

There are 65536 words of active waveform memory. The active memory is divided up into an area designated for the Standard Waveform (STDW), and the rest of the memory which can be partioned into waveform files numbered 0 to 99. The horizontal size (in digital words) of each partioned waveform can be set by the **SIZE** command, and thus the total number of waveforms is limited by the cumulative size of the individual waveforms.

The minimum and maximum y values for the two memories are -32768 and +32767, respectively.

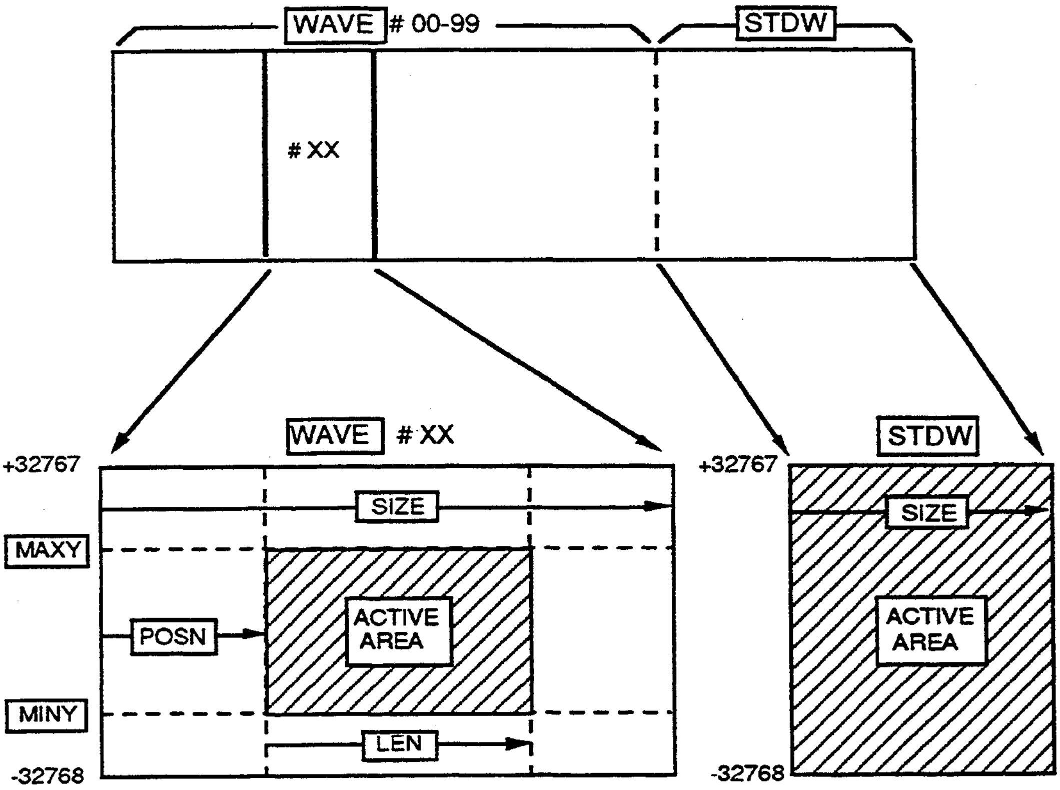


Figure 10-6. Waveform Editing Commands

When using the Waveform editing commands to operate on a specific waveform memory, it is important to realize that these commands only effect the active area (See Note below). In other words, if you wanted to create a sinewave with three cycles, starting at 0 phase, you would send the following command:

```
WVFM:SINE  3,0;:EXEC
```

This command would place three cycles of sinewaves into the active area, with a maximum and minimum y value the same as the active area. The commands **LEN, MINY, MAXY**, and **POSN** all affect the actual dimensions of the active area as follows:

LEN - The length command determines the actual horizontal length of the active area. This maximum length is SIZE-POSN.

MINY - The minimum y command sets the lowest y value for the active area.

MAXY - The maximum y command sets the highest y value for the active area.

POSN - The position command determines the horizontal starting point for the active area. Note that once a waveform is written into the active area (using the waveform edit commands), POSN is changed to the point POSN + LEN.

Important Note: For the Standard Waveform (STDW), the active area is always the entire area defined by SIZE. Thus, none of the four commands mentioned above effect the STDW.

10.8 WAVEFORM MEMORY FORMAT

This section describes the formats of the commands which enter data into the wave memory of a Pragmatic Instruments arbitrary waveform generator via the GPIB or RS-232C interface.

10.8.1 Decimal Waveform Download

The contents of the waveform memory for wave X (X = 0..99) are changed by a single command formatted according to the IEEE 488.2-1987 standard, and has the following syntax:

```
<command header> <start address>,<data>[,<data> ... ];
```

As a command for the GPIB or RS-232C a <command header>, followed by the address
of the first memory cell to be set, followed by one or more <data> items and terminated by a semicolon. The <start address> and <data> items are separated by commas, with a space or tab separating the <command header> from the <start address> item. The numbers specified by the <data> items are stored in successive addresses of the wave memory.

The <command header> is a string, whose value depends only on the wave to be set:

Wave	Command Header Value
0	WVFM:WAVE 0;MEM
1	WVFM:WAVE 1;MEM
99	WVFM:WAVE 99;MEM

The <start address> is a decimal number between 0 and 65471. It sets the starting address where the <data> will be downloaded in waveform memory.

The <data> items are decimal numbers between -32768 and 32767. If the AMPL (Amplitude) parameter is set to 10V, the following data: -32768, 0, 32767 would produce -10 Volts, 0 Volts, and 10 Volts, respectively, on the output.

Examples:

An 8 point positive Ramp down loaded into wave 1, address 0.

```
WVFM:WAVE 1;MEM 0,0,4681,9362,14043,18724,23405,28086,32767;
```

An 8 point sine wave down loaded into wave 2, address 48.

```
WVFM:WAVE 2;MEM 48,0,23169,32767,23169,0,-23170,-32768,-23170;
```

10.8.2 Binary Waveform Download

The 2411A also supports binary format for waveform <data> items. Binary format is the fastest way to transfer data.

```
<command header> <start address>,<binary data><\n>
```

The <command header> and <start address> are the same as above. The <binary data> has the following format:

```
<#><num><length><hi byte><lo byte>[<hi><lo>...]<\n>
```

Where:
<#> is the pound sign, ASCII(35).
<num> is the number of digits in the length parameter. 1 to 9.
<length> is the length in bytes (two bytes per waveform data point). 2 to 4096.
<hi byte> contains the upper eight bits of a waveform data point .
<lo byte> contains the lower eight bits of a waveform data point .
<\n> is the new-line character, ASCII(10).

NOTE:

1. The range of a waveform data point is 0 to 65535. If the AMPL (Amplitude) parameter is set to 10V, the following waveform data point values: 0, 32768, and 65535, produce -10 Volts, 0 Volts, and 10 Volts, respectively, on the output. The waveform data point is converted to <hi byte><lo byte> format for binary downloading. Conversion from waveform data point to <hi byte><lo byte> can be accomplished as follows:

   QuickBASIC

   ```
   high_byte = FIX( pointdata / 256)
   low_byte  = pointdata MOD 256
   ```

   C

   ```
   high_byte = pointdata / 256
   low_byte  = pointdata % 256
   ```

   Where point data is between 0 and 65535.

2. Since two bytes are required for each waveform data point, the length of bytes sent must always be even. The MAXIMUM <length> of a binary block is 4096 bytes or 2048 waveform data points.
3. For <binary data> larger than 2048 points, the data must be broken up into multiple binary blocks and sent. An example of downloading binary data larger than 2048 points is included in a QuickBASIC program at the end of this document.

Example:

An 8 point positive Ramp down loaded into wave 1, address 0.

WVFM:WAVE 1;MEM 0,#40016***binary_data***\n

Where ***binary_data*** are the following values as bytes sent to the GPIB or RS-232C

```
|128| 0|146|73|164|146|182|219|201|36|219|109|237|182|255|255|
```

The complete command as bytes in memory would look like the following:

```
|'W'|'V'|'F'|'M'|':'|'W'|'A'|'V'|'E'|' '|'1'|';'|'M'|
|'E'|'M'|' '|'0'|','|'#'|'4'|'0'|'0'|'1'|'6'|128| 0 |
|146| 73|164|146|182|219|201| 36|219|109|237|182|
|255|255| 10|
```

Where |'A'|represents one byte in memory containing the character 'A' (i.e the value ASCII(65)) and |219| represents one byte in memory whose value is 219.
Therefore: |'A'| and |65| are equal in value.

In QuickBASIC

```
A$= "WVFM:WAVE 1;MEM 0,#40016"+CHR$(128)+CHR$(0)
  +CHR$(146)+CHR$(73)+CHR$(164)+CHR$(146)
  +CHR$(182)+CHR$(219)+CHR$(201)+CHR$(36)
  +CHR$(219)+CHR$(109)+CHR$(237)+CHR$(182)
  +CHR$(255)+CHR$(255)+CHR$(10)
```

In C

```
{
int i;
char wvfm_buffer[100];
char data[16]= {128,0,146,73,164,146,182,219,201,36,219,109,237,182,255,255}
sprintf(wvfm_buffer,"WVFM:WAVE 1;MEM 0,#40016");
for(i=0; i < 16; i++)
  wvfm_buffer[i+24] = data[i]; /* 24 bytes in header */
  wvfm_buffer[16+24] = '\n'; /* end of block */
}
```

Example:

An 8 point sine wave down loaded into wave 2, address 48.

WVFM:WAVE 2;MEM 48,#40016***binary_data***\n

Where ***binary_data*** are the following values as bytes sent to the GPIB or RS-232C.

```
|128| 0|218|129|255|255|218|129|128| 0|37|126| 0| 0| 37|126|
```

The complete command as bytes in memory would look like the following:

```
|'W'|'V'|'F'|'M'|':'|'W'|'A'|'V'|'E'|' '|'2'|';'|'M'|'E'|
|'M'|' '|'4'|'8'|','|'#'|'4'|'0'|'0'|'1'|'6'|128| 0 |218|
|129|255|255|218|129|128| 0 | 37|126| 0 | 0 | 37|126| 10|
```

In BASIC

```
A$= "WVFM:WAVE 2;MEM 48,#40016"+CHR$(128)+CHR$(0)
  +CHR$(218)+CHR$(129)+CHR$(255)+CHR$(255)
  +CHR$(218)+CHR$(129)+CHR$(128)+CHR$(0)+CHR$(37)+CHR$(126)
  +CHR$(0)+CHR$(0)+CHR$(37)+CHR$(126)+CHR$(10)
```

In C

```
{
int i;
char wvfm_buffer[100];
char data[16]= {128,0,218,129,255,255,218,129,128,0,37,126,0, 0,37,126}
sprintf(wvfm_buffer,"WVFM:WAVE 2;MEM 48,#40016");
for(i=0; i < 16; i++)
  wvfm_buffer[i+25] = data[i]; /* 25 bytes in header */
  wvfm_buffer[16+25] = '\n'; /* end of block */
}
```

```
REM*****************************************************************************************
REM ****        2411A BINARY DOWNLOAD TEST PROGRAM FOR GPIB                            ****
REM*****************************************************************************************
REM  Language    : Microsoft Quick Basic
REM  Computer    : IBM AT or better
REM  GPIB Board : National Instruments AT-GPIB, IEEE488.2
REM
REM  Function    : The program downloads a 8400 point Sine Wave into
REM              Wave 1, starting at Address 0.
REM              The data transfer occurs in 5 blocks, the first
REM              four blocks containing 2048 Data Points and the
REM              last containing 208 Data Points.
REM*****************************************************************************************
REM $INCLUDE: 'qbdecl.bas'

COMMON SHARED Arb%

  PRINT
  PRINT "Initializing GPIB & resetting ARB"
  PRINT
  CALL IBDEV(0, 16, 0, 12, 1, 0, Arb%)'connect Arb at add 16
  CALL ibwrt(Arb%, "*idn?")
  ArbResponse$ = SPACE$(100)
  CALL IBRD(Arb%, ArbResponse$)
  IF IBCNT% > 0 THEN
    PRINT "Unit is a : "; MID$(ArbResponse$, 1, IBCNT%)
    GOTO GpibWasOk
    END IF
  PRINT ">> GPIB ERROR <<"
  PRINT "Corrective Action REQUIRED"
  STOP

GpibWasOk:
SetupParameters:

  MaxBlockSize& = 2048
  Length& = 8400
  DIM ArbData&(Length&)


  NumBlocks& = INT(Length& / MaxBlockSize&)
  NumPartial& = Length& - NumBlocks& * MaxBlockSize&

  PRINT "Calculating a sample Sinewave"

  Pie = 3.141593

  FOR x& = 0 TO Length&
    ArbData&(x&) = 32767*SIN((x& / Length& * 10)*2*Pie) + 32767
    IF ArbData&(x&) > 65535 OR ArbData&(x&) < 0 THEN STOP
  NEXT x&
```

```
PRINT "Setting up machine"

CALL ibwrt(Arb%, "*cls")        'Start clean
CALL ibwrt(Arb%, "outsw on;:exec")
CALL ibwrt(Arb%, ":wvfm:wave 1;size " + STR$(Length&) + ";len " + STR$(Length&))
CALL ibwrt(Arb%, ":wvfm:wave 1;posn 0;miny -2047;maxy 2047")
CALL ibwrt(Arb%, ":Func Wave,1;:exec")
CALL ibwrt(Arb%, ":wvfm:Wave 1;Dc 0")
GOSUB CheckStatus

BlockNum& = 0

PRINT "Starting Binary Download of : "; Length&; " points"
PRINT "Start > "; TIME$

DO WHILE (BlockNum& < NumBlocks&)
   BlockSize& = MaxBlockSize&
   GOSUB MakeHeader
   StartIndex& = (BlockNum& * MaxBlockSize&)
   BlockNum& = BlockNum& + 1
   StopIndex& = (BlockNum& * MaxBlockSize&) - 1
   PRINT "Assembling Block : "; BlockNum&; " in progress";
   GOSUB AssembleData
   PRINT " > DownLoading < ";
    CALL ibwrt(Arb%,"disp 'Block "+STR$(BlockNum&)+" processing'" +CHR$(10))
   CALL ibwrt(Arb%, "wvfm:mem " + STR$(StartIndex&) + "," + Header$
   + BinData$ +  CHR$(10))
   PRINT " >> completed <<"
   GOSUB CheckStatus
LOOP

DO WHILE (NumPartial&)
   StartIndex& = (BlockNum& * MaxBlockSize&)
   BlockSize& = NumPartial&
   GOSUB MakeHeader
   PRINT "Assembling a partial bock of : "; BlockSize&;
    CALL ibwrt(Arb%, "disp 'Partial Block "+STR$(BlockNum&)+"'"+ CHR$(10))
   StopIndex& = NumPartial& - 1 + StartIndex&
   GOSUB AssembleData
   PRINT " > DownLoading > ";
   CALL ibwrt(Arb%, "wvfm:mem " + STR$(StartIndex&)+","+ Header$
   + BinData$+CHR$(10))
   PRINT " >> completed <<"
   GOSUB CheckStatus
   NumPartial& = 0
LOOP

PRINT " Stop > "; TIME$
CALL IBLOC(Arb%)
INPUT " >> Hit Return to continue <<", Scratch$

END
```

```
MakeHeader:

   temp1$ = "#6000000"
   temp2$ = LTRIM$(STR$(2 * BlockSize&))
   Header$ = LEFT$(temp1$, LEN(temp1$) - LEN(temp2$)) + temp2$
   RETURN

AssembleData:

   CurrentPosition& = 1
   BinData$ = SPACE$(2 * (StopIndex& - StartIndex& + 1))
   FOR IndexPointer& = StartIndex& TO StopIndex&
   MID$(BinData$,CurrentPosition&,2) = CHR$(INT(ArbData&( IndexPointer&) / 256))
   + CHR$(ArbData&(IndexPointer&) AND 255)
   CurrentPosition& = CurrentPosition& + 2
   NEXT IndexPointer&
   RETURN

CheckStatus:

   CALL ibwrt(Arb%, "*ESR?")
   InputString$ = SPACE$(100)
   CALL IBRD(Arb%, InputString$)
   IF VAL(InputString$) AND 1 THEN
      PRINT ">> Operation Complete <<";
   END IF
   IF VAL(InputString$) AND 2 THEN
      PRINT ">> Request Control ( N/I ) <<";
   END IF
   IF VAL(InputString$) AND 4 THEN
      PRINT ">> QUERY ERROR <<";
   END IF
   IF VAL(InputString$) AND 8 THEN
      PRINT ">> Device Dependant ERROR <<";
   END IF
   IF VAL(InputString$) AND 16 THEN
      PRINT ">> Execution ERROR <<";
   END IF
   IF VAL(InputString$) AND 32 THEN
      PRINT ">> Command ERROR <<";
   END IF
   IF VAL(InputString$) AND 64 THEN
      PRINT ">> User Request ( N/I ) <<";
   END IF
   IF VAL(InputString$) AND 128 THEN
      PRINT ">> Power On <<";
   END IF
   IF VAL(InputString$) > 0 THEN
      PRINT " Status read : "; VAL(InputString$)
   END IF
   IF VAL(InputString$) > 0 THEN CALL ibwrt(Arb%, "*CLS")
   RETURN
```

APPENDIX

MENU LOGIC TREE

This Menu Tree provides a complete list of all setup and control display menus.

Pressing a hard key opens a menu with additional choices. The submenus are shown horizontally to the right and below the main menu selections. The second branch submenus are shown vertically below the submenu selections. Keys are identified as shown below: Many entries provide softkey selections of DO, OK and CANCEL.

□ HARDKEY	▨ FIRST-LEVEL SOFTKEY	[SECOND-LEVEL SOFTKEY] [THIRD-LEVEL SOFTKEY]

WAVEFORM PARAMETER KEY GROUP

MODE	CONT	TRIG	GATE	BRST
	TOGL	□	□	□
FUNC	WAV# Set Waveform Number	SEQ# Set Sequence Number	□	VIEW [ALL] [STRT] [STOP]
	SIN Set Phase Set Number	SQR Set Duty Cycle Set Number	TRI Set Number	SAW+ Set Duty Cycle Set Number
	SAW- Set Duty Cycle Set Number	DC Set Value	EXP+ Set Time Constant	EXP- Set Time Constant
	AM Set Carrier Freq. Set Mod. Freq. Set Carrier Phase Set Mod. Phase Set % Modulation	SCM Set Carrier Freq. Set Mod. Freq. Set Carrier Phase Set Mod. Phase	FM Set Carrier Freq. Set Mod. Freq. Set Carrier Phase Set Mod. Phase Set Mod. Index	HSIN Set Number

WAVEFORM PARAMETER KEY GROUP (Cont.)

SLIN
Set Beginning and End Frequencies

SLOG
Set Beginning and End Frequencies

PLS+
Set Delay %
Set Rise %
Set High %
Set Fall %

PLS-
Set Delay %
Set Rise %
Set High %
Set Fall %

GAUS
Set Exponent Power

SX/X
Set Number

CIRC
Set Phase
Set Number

NOIS

CLOCK FREQ
Set Sample Clock
Observe Frequency

AMPL OFST
Set Amplitude
Set Offrset

SETUP, UTILITY AND EDIT KEY GROUP

SETUP

WAV#

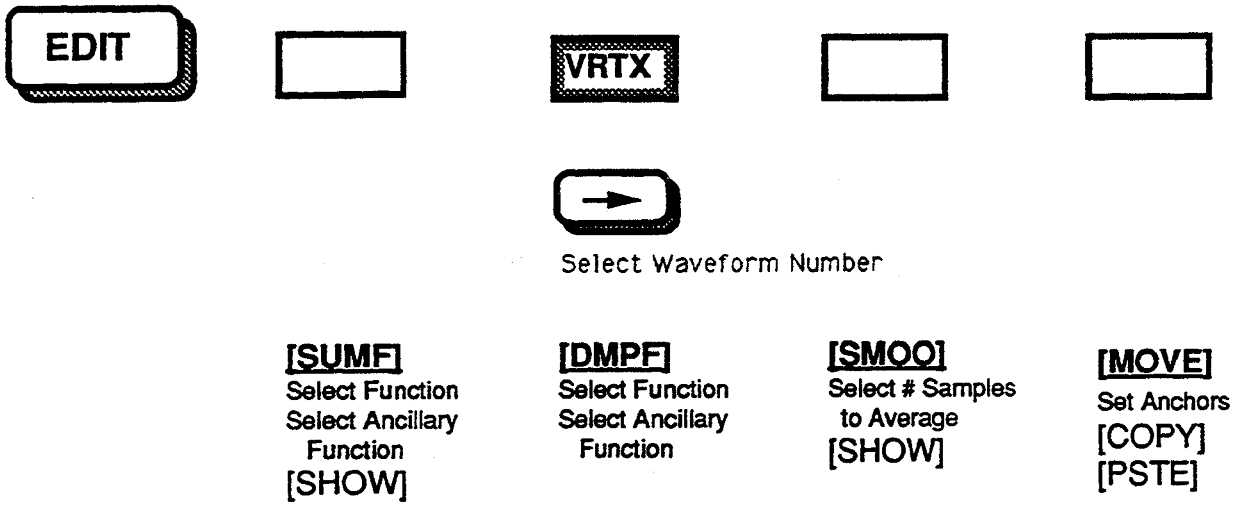
[SYNC]

[SYN1]
[SYN3]
[SYN4]
Set Address and Length

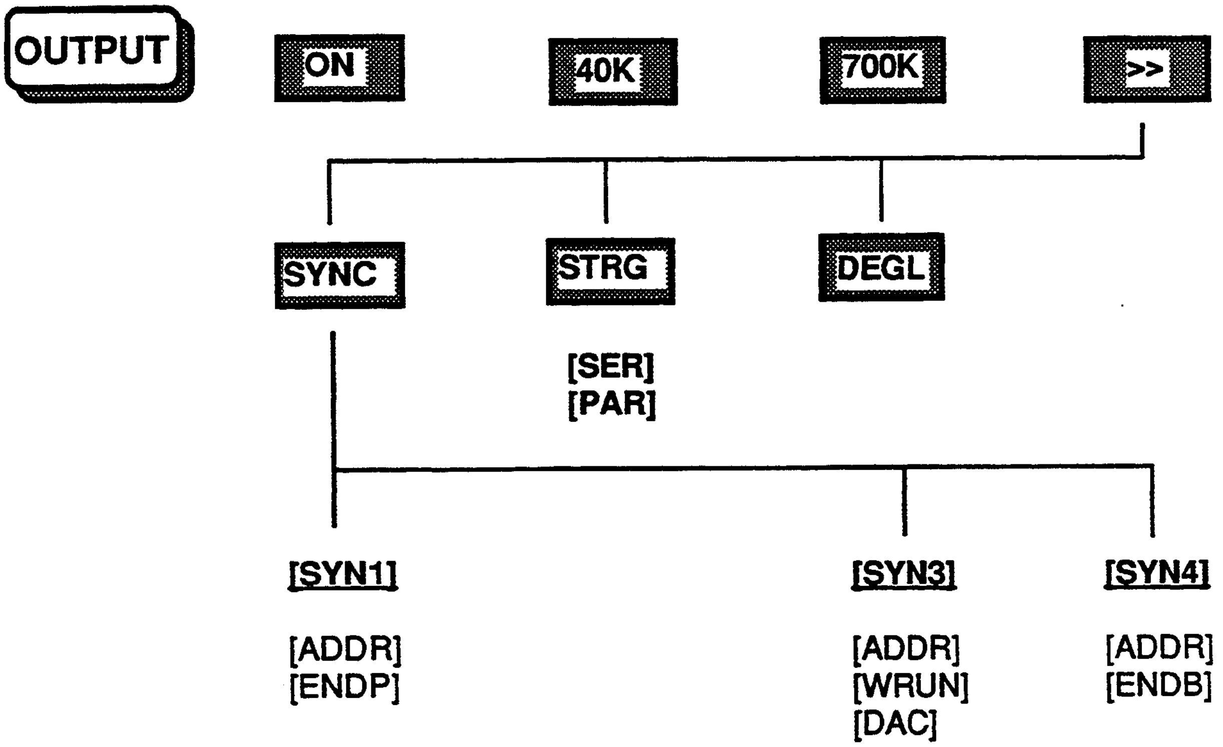
[DEL]

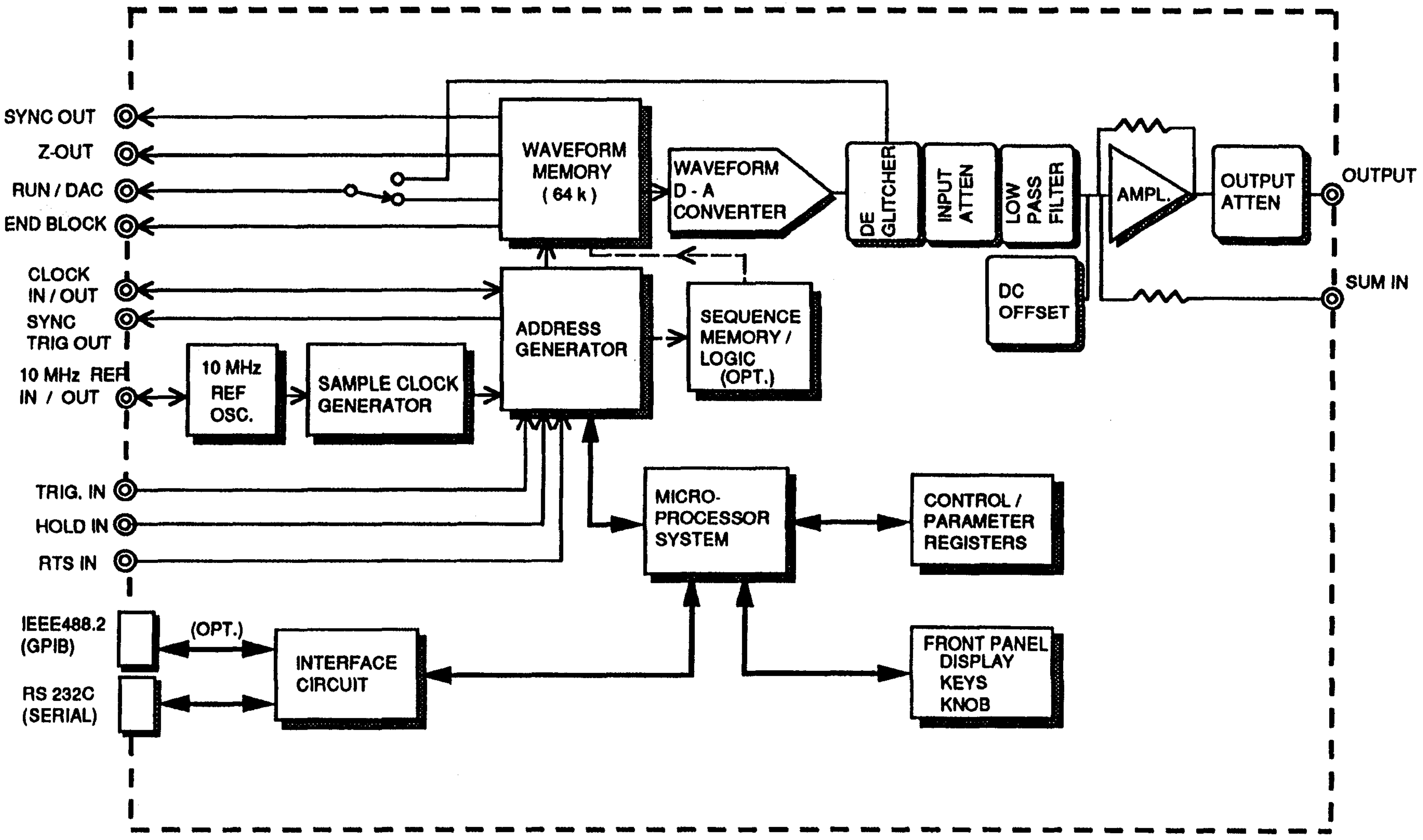
[NEW]
Set New Waveform Number and Length

[LEN]
Set Waveform Length


EDIT
VRTX
→
Select Waveform Number
[SUMF]
Select Function
Select Ancillary Function
[SHOW]
[DMPF]
Select Function
Select Ancillary Function
[SMOO]
Select # Samples to Average
[SHOW]
[MOVE]
Set Anchors
[COPY]
[PSTE]

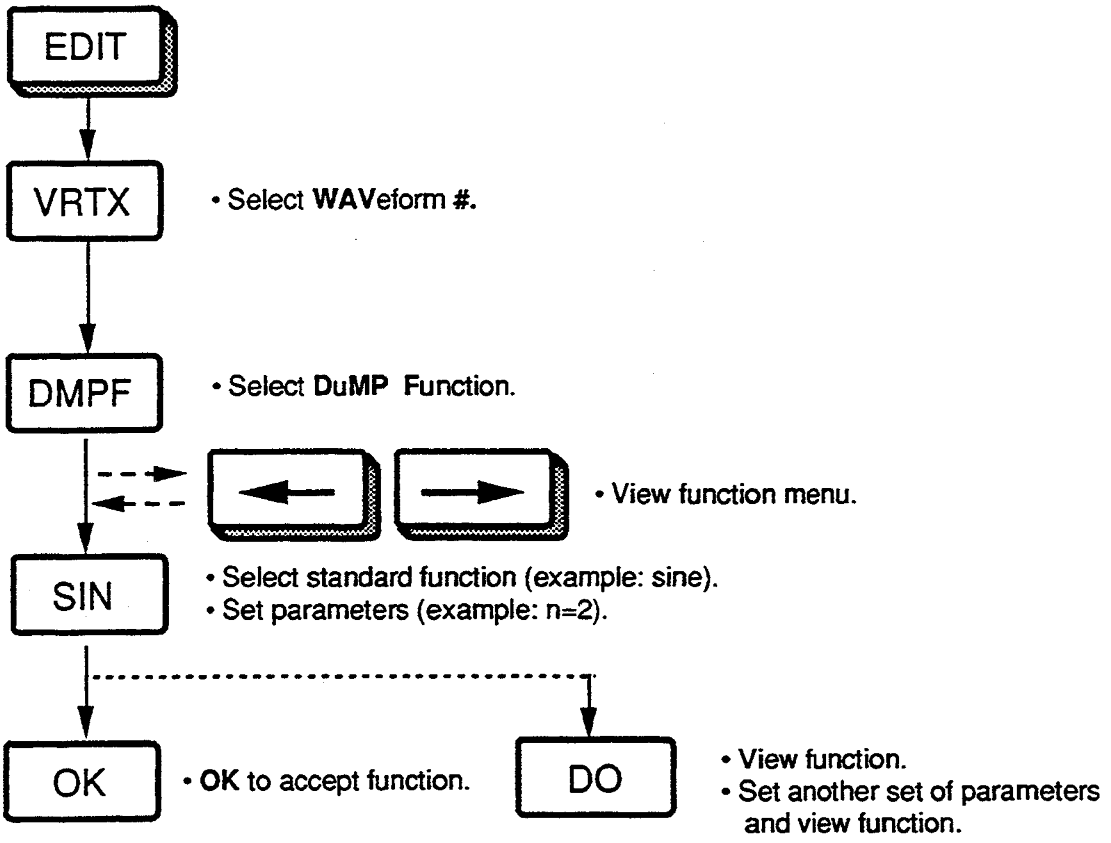
OUTPUT KEY


OUTPUT
ON
40K
700K
>>
SYNC
STRG
DEGL
[SER]
[PAR]
[SYN1]
[ADDR]
[ENDP]
[SYN3]
[ADDR]
[WRUN]
[DAC]
[SYN4]
[ADDR]
[ENDB]

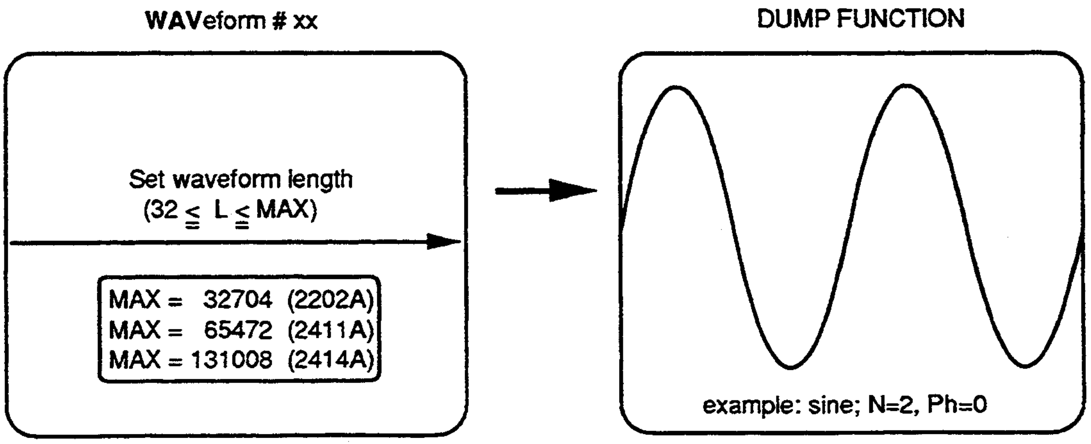


2411A BLOCK DIAGRAM

Draw Waveform - DUMP FUNCTION



- Select **WAV**eform **#**.
- Select **DuMP** Function.
- View function menu.
- Select standard function (example: sine).
- Set parameters (example: n=2).
- **OK** to accept function.
- View function.
- Set another set of parameters and view function.

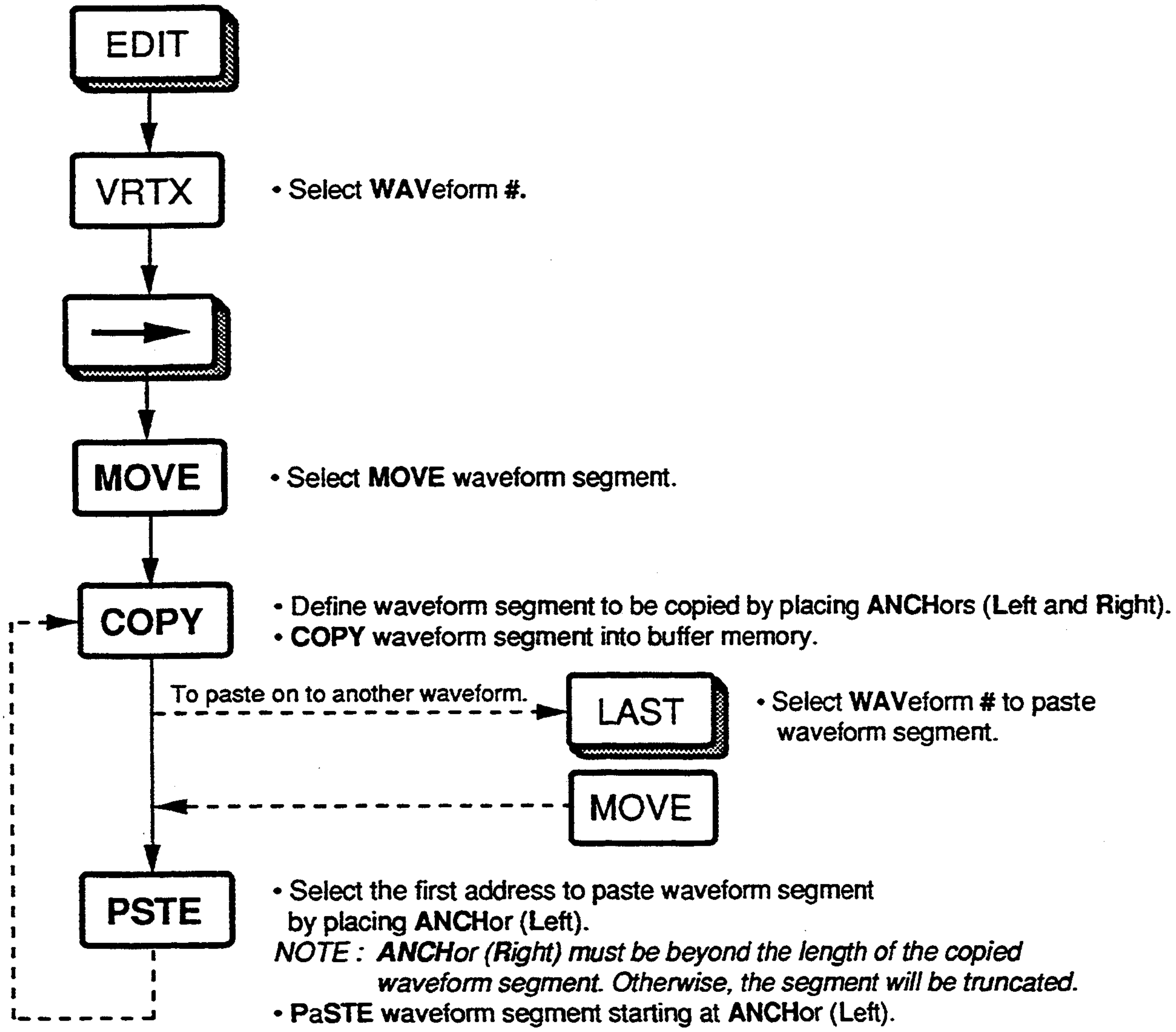


Set waveform length
(32 ≦ L ≦ MAX)

```
MAX =  32704  (2202A)
MAX =  65472  (2411A)
MAX = 131008  (2414A)
```

example: sine; N=2, Ph=0

Draw Waveform - MOVE



- Select **WAV**eform **#**.

- Select **MOVE** waveform segment.

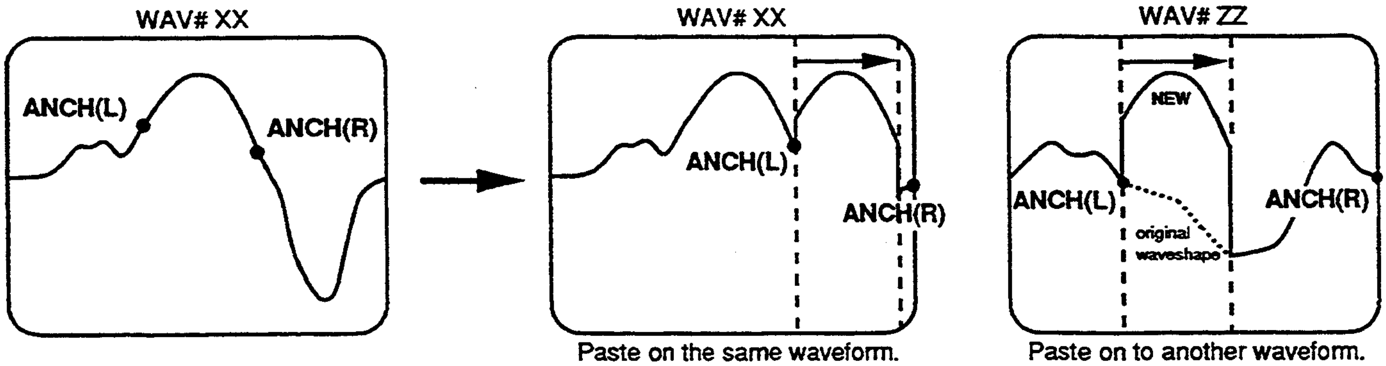
- Define waveform segment to be copied by placing **ANCH**ors (**Left** and **Right**).
- **COPY** waveform segment into buffer memory.

- Select **WAV**eform **#** to paste waveform segment.

- Select the first address to paste waveform segment by placing **ANCH**or (Left).

*NOTE : **ANCHor (Right)** must be beyond the length of the copied waveform segment. Otherwise, the segment will be truncated.*

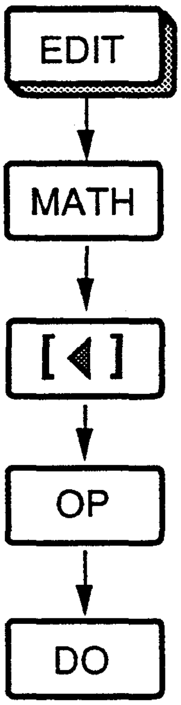
- **PaSTE** waveform segment starting at **ANCH**or (Left).



Paste on the same waveform.

Paste on to another waveform.

Waveform MATH



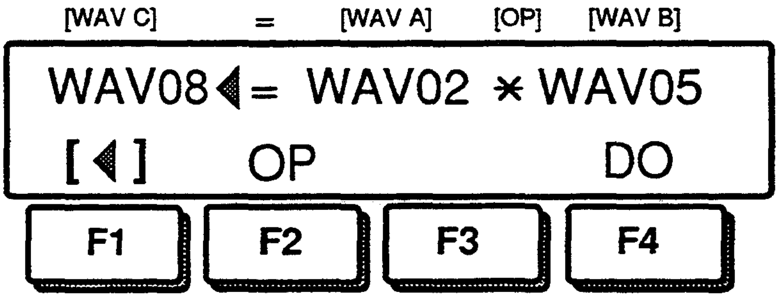
- Select **MATH.**

- Select **WAV**eforms **A,B** and **C** by pressing [◀].
- Select **WAV**eform number by rotating knob.

- Select **OP**eration (+,-, x).

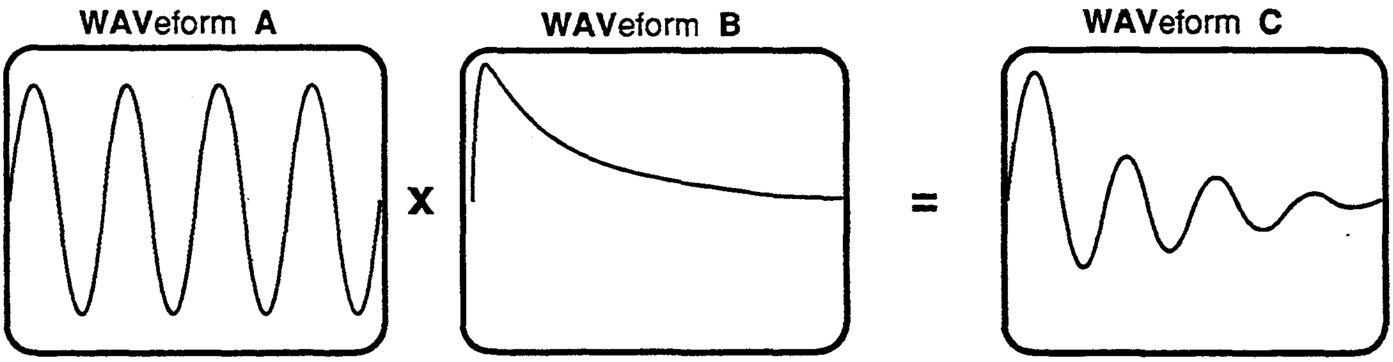
- Compute.
- Place the result in **WAV**eform **C.**

EXAMPLE : DISPLAY and SOFT KEYS

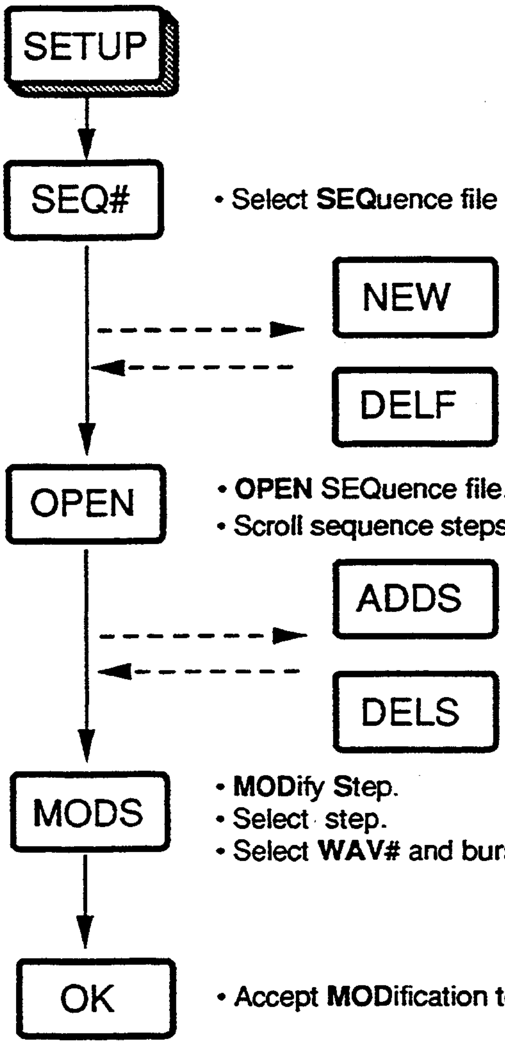


```
WAV08◀= WAV02 * WAV05
[◀]    OP          DO
```

WAV02 and WAV05 are multiplied. The result is placed in WAV08.



Create Waveform by SEQUENCE



SETUP

SEQ#
- Select **SEQ**uence file number.

NEW
- Create **NEW** sequence file number.

DELF
- **DELete** currently selected sequence **F**ile.

OPEN
- **OPEN** SEQuence file.
- Scroll sequence steps with knob. (2414A only)

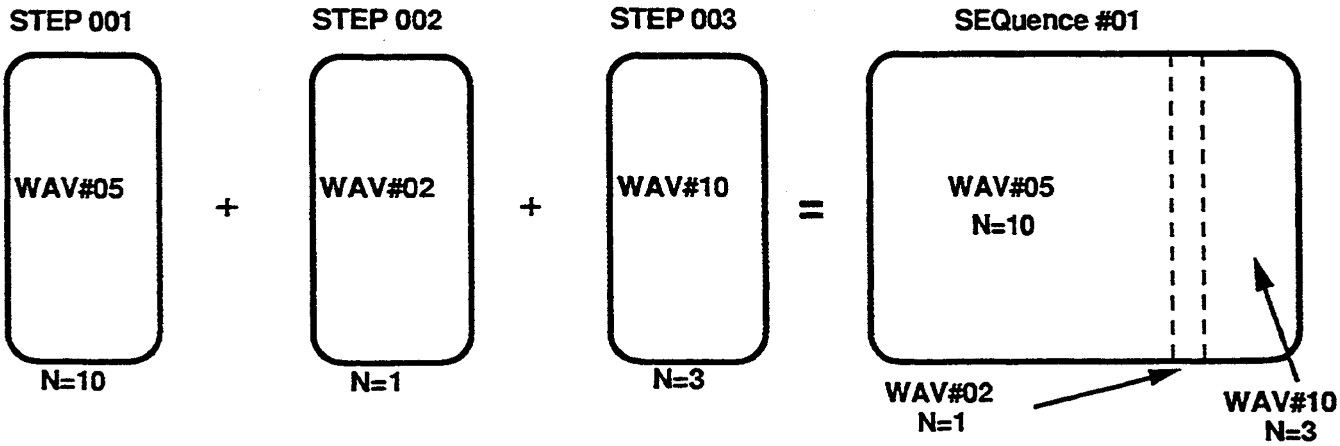
ADDS
- **ADD** new **S**tep.

DELS
- **DEL**ete currently selected **S**tep.

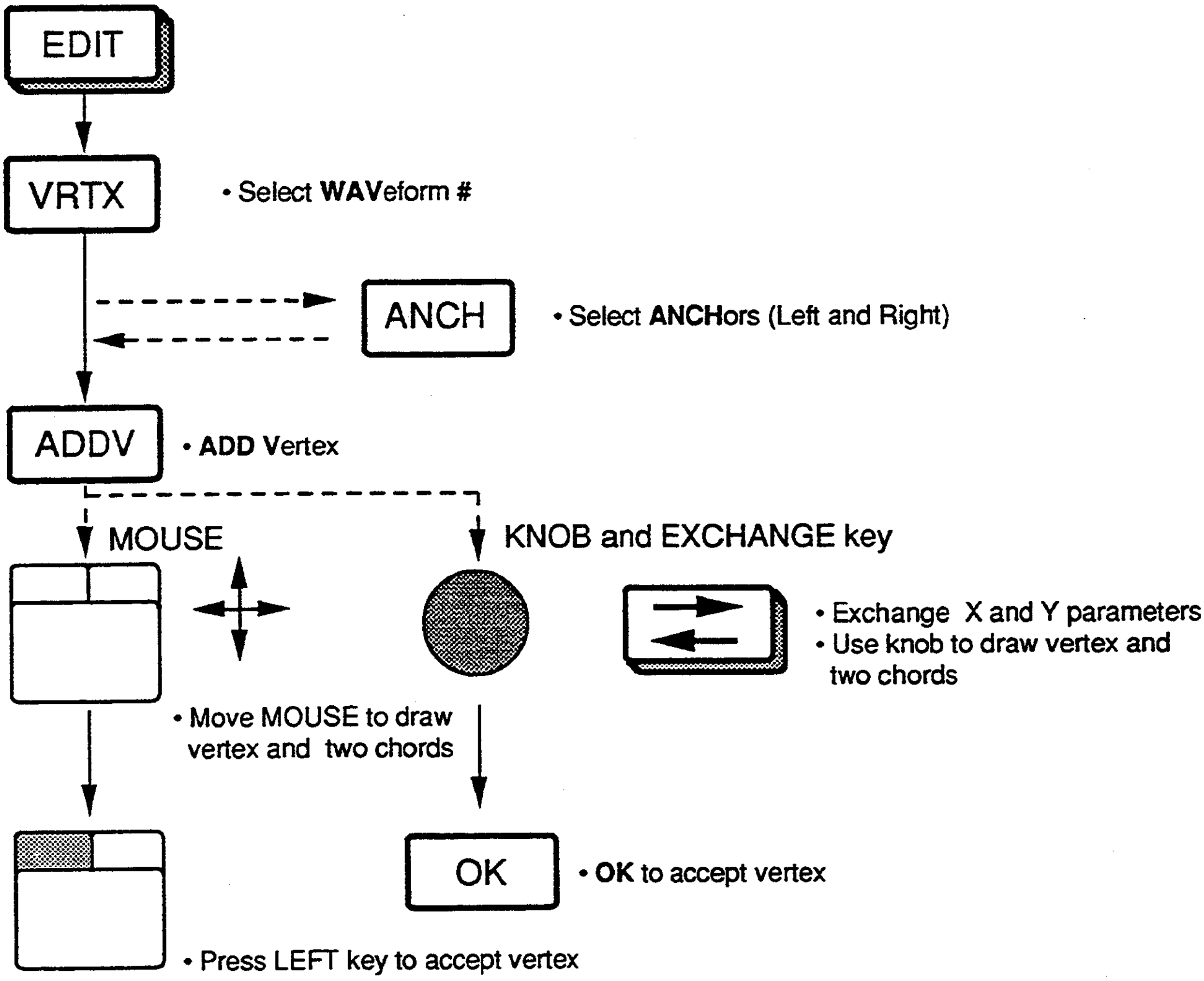
MODS
- **MOD**ify **S**tep.
- Select step.
- Select **WAV#** and burst count (**N**).

OK
- Accept **MOD**ification to **S**tep.

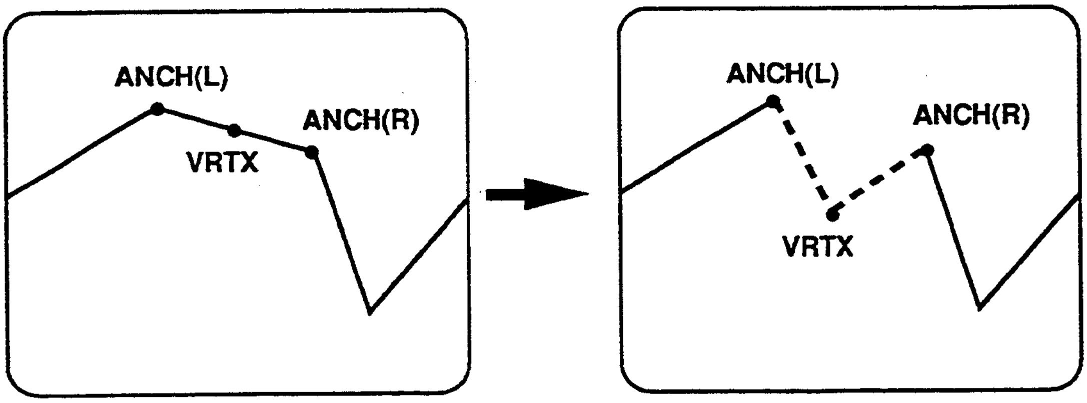
EXAMPLE



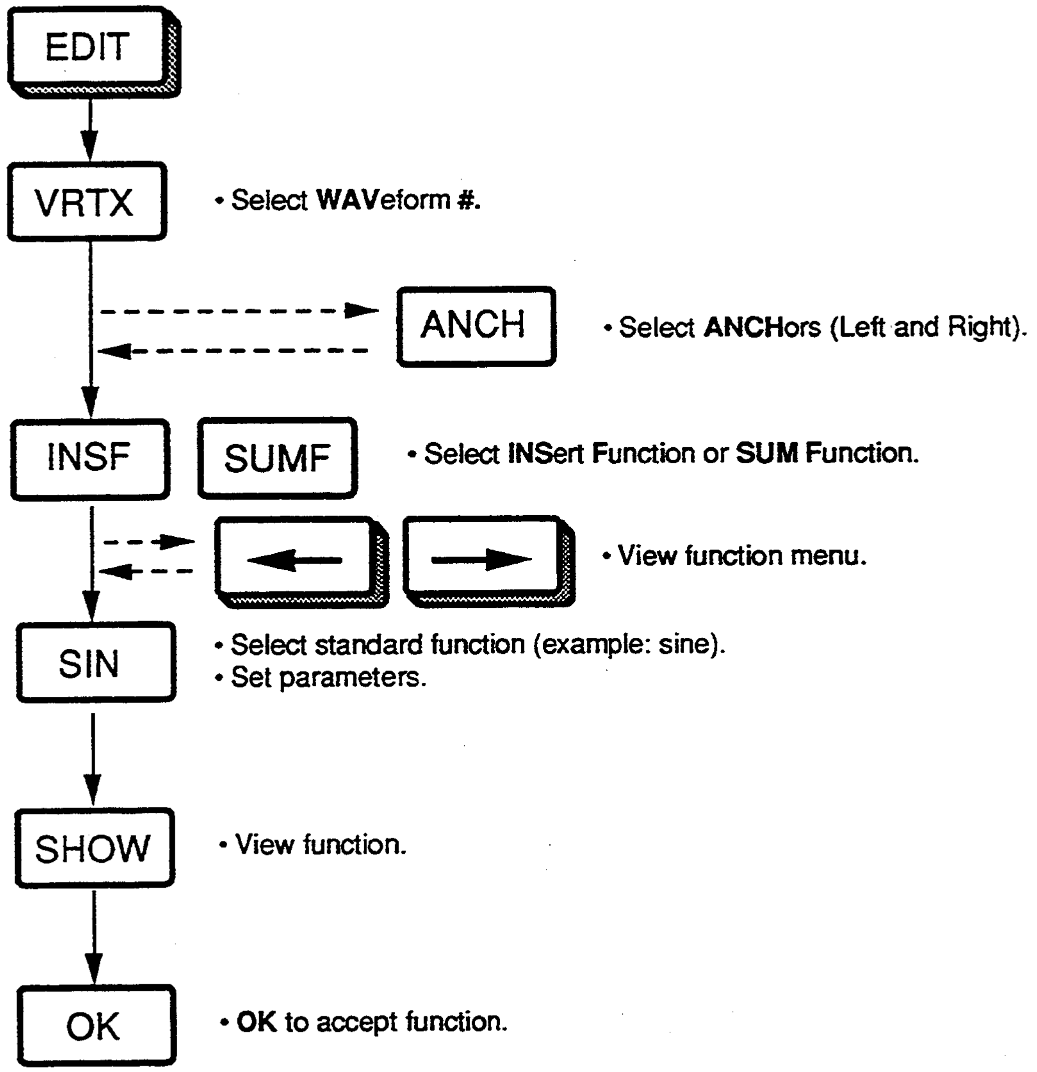
Draw Waveform - VERTEX



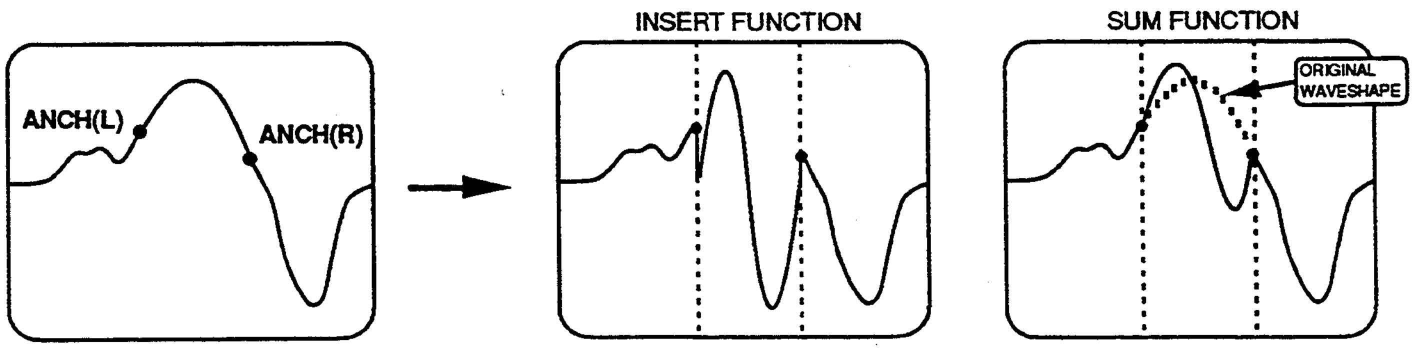
- Select **WAV**eform #
- Select **ANCH**ors (Left and Right)
- **ADD** Vertex
- Move MOUSE to draw vertex and two chords
- Exchange X and Y parameters
- Use knob to draw vertex and two chords
- **OK** to accept vertex
- Press LEFT key to accept vertex



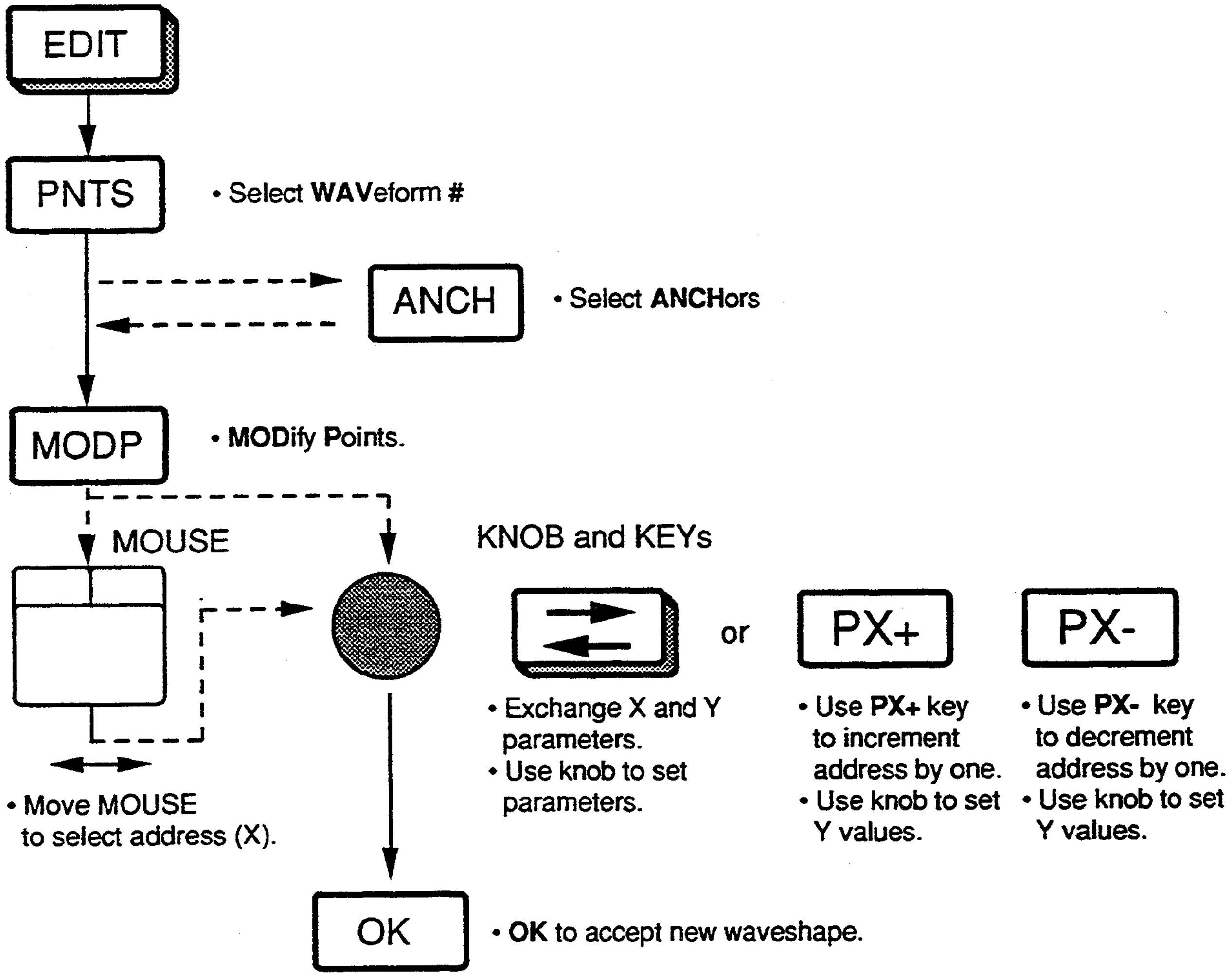
Draw Waveform - INSERT or SUM FUNCTION



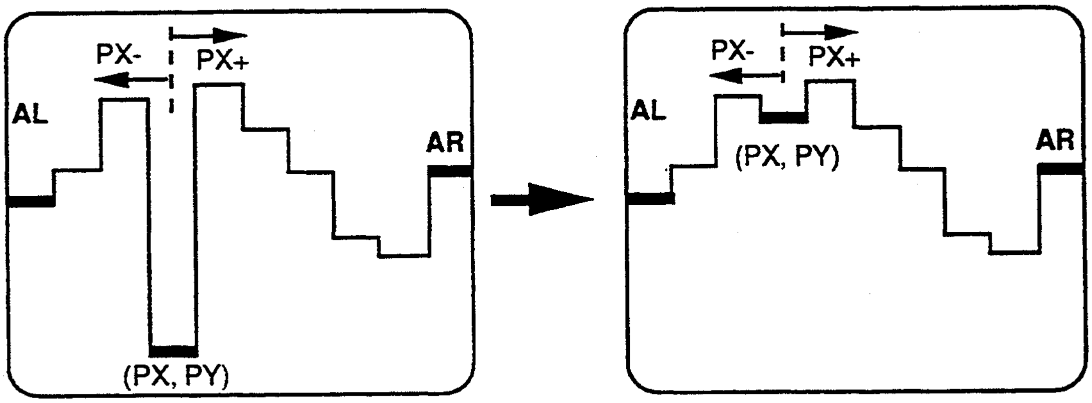
- Select **WAV**eform **#.**
- Select **ANCH**ors (Left and Right).
- Select **INS**ert Function or **SUM** Function.
- View function menu.
- Select standard function (example: sine).
- Set parameters.
- View function.
- **OK** to accept function.



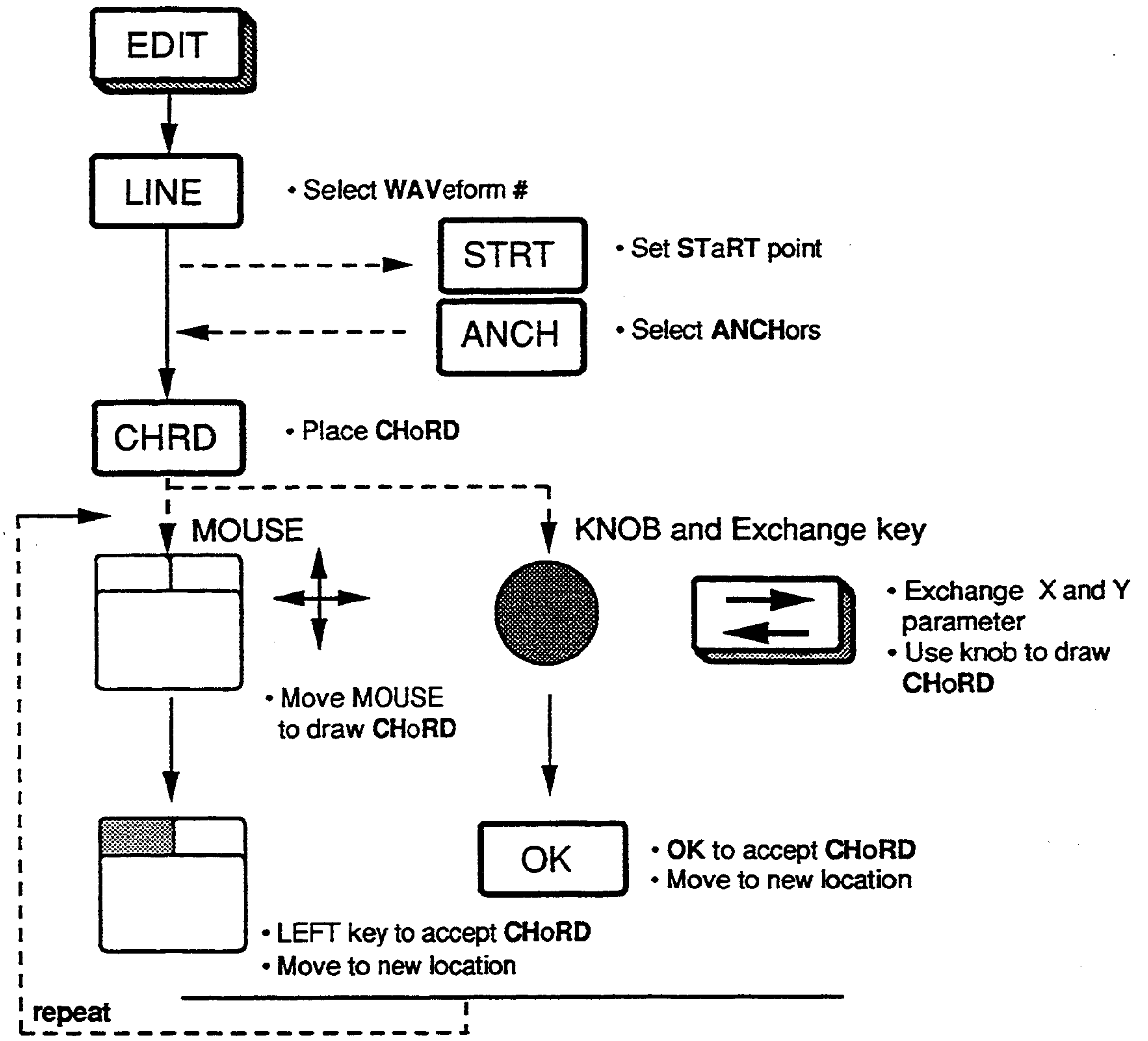
Draw Waveform - POINT (2414A only)



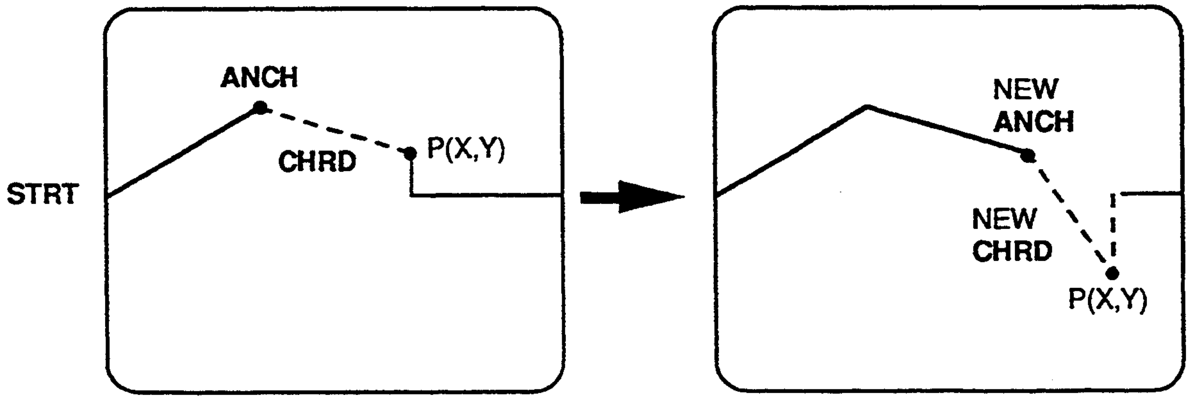
- Select **WAV**eform #
- Select **ANCH**ors
- **MOD**ify **P**oints.
- Move MOUSE to select address (X).
- Exchange X and Y parameters.
- Use knob to set parameters.
- Use **PX+** key to increment address by one.
- Use knob to set Y values.
- Use **PX-** key to decrement address by one.
- Use knob to set Y values.
- **OK** to accept new waveshape.



Draw Waveform - LINE



- Select **WAV**eform #
- Set **STaRT** point
- Select **ANCH**ors
- Place **CHoRD**
- Move MOUSE to draw **CHoRD**
- Exchange X and Y parameter
- Use knob to draw **CHoRD**
- **OK to accept CHoRD**
- Move to new location
- LEFT key to accept **CHoRD**
- Move to new location



DICTIONARY OF TERMS

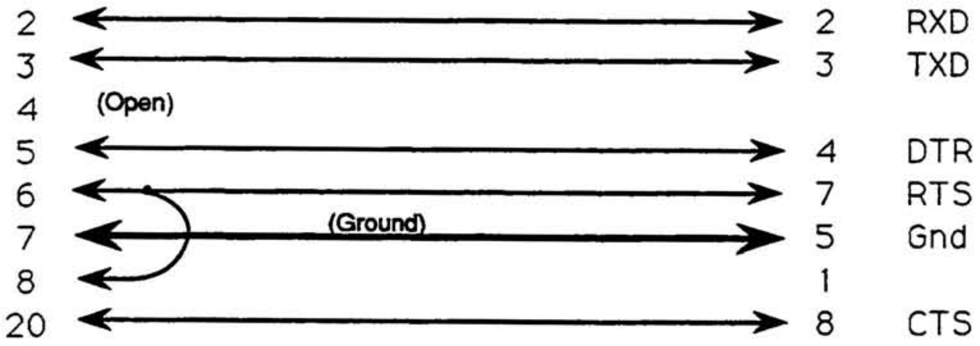
ADDS	Add Step
ADDR	Address
ADDV	Add Vertex
AM	Amplitude Modulation
AMPL	Amplitude
ANCH	Anchor
ASNC	Asynchronous
BRST	Burst
CANC	Cancel
CHRD	Chord
CIRC	Circle
CONT	Continuous
DAC	Digital-to- Analog Converter
DEL	Delete
DELF	Delete File
DELS	Delete Step
DMPF	Dump Function
ENDB	End Block
ENDP	End Pulse
EXP	Exponential
FLTR	Filter
FM	Frequency Modulation
FREQ	Frequency
FUNC	Function
GAUS	Gaussian
GPIB	General Purpose Interface Bus
HSIN	Haversine
INSF	Insert Function
LCD	Liquid-Crystal Display
LEN	Length
MBST	Monitor Burst Count
NOIS	Noise
OFST	Offset
OP	Operation
PAR	Parallel
PLS	Pulse
R232	RS-232C Interface Bus
RCLK	Reference Clock
RECL	Recall
RTS	Return To Start
SAW	Sawtooth
SCAL	Scale
SCLK	Sample Clock
SCM	Suppressed-Carrier Modulation
SEQ#	Sequence Number
SER	Serial
SIN	Sine
SLIN	Linear Sweep
SLOG	Logarithmic Sweep
SMOO	Smooth
SQR	Square
STDW	Standard Wave
STOR	Store
STRG	Synchronous Trigger
STRT	Start
SUMF	Sum Function
SX/X	Sine X/X
TGEN	Trigger Generator
TGIN	Trigger In
TOGL	Toggle
TRI	Triangle
TRIG	Trigger
UTIL	Utilities
VRTX	Vertex
WAV#	Waveform Number
WRUN	Waveform Run
ZLVL	Z-Axis Amplitude

RS-232C ADAPTER CABLES

The following wiring diagrams illustrate proper interconnects between the serial port of an IBM (or compatible) PC and the Model 2411A RS-232C connector.

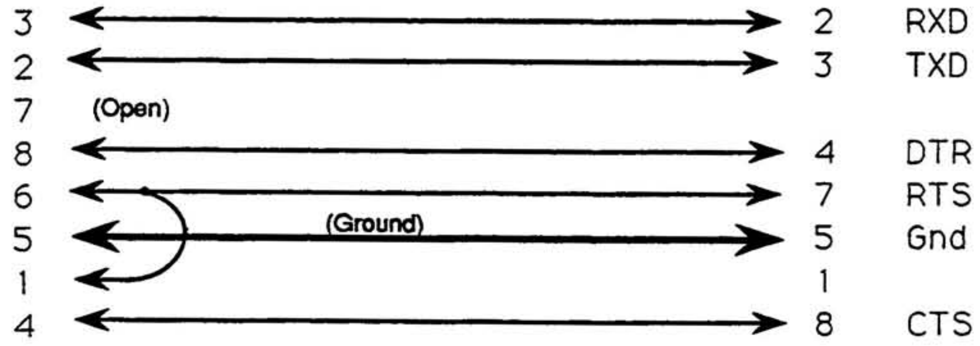
IBM PC
(DB-25, female)

Model 2411A
(DB-9, female)



IBM PC
(DB-9, female)

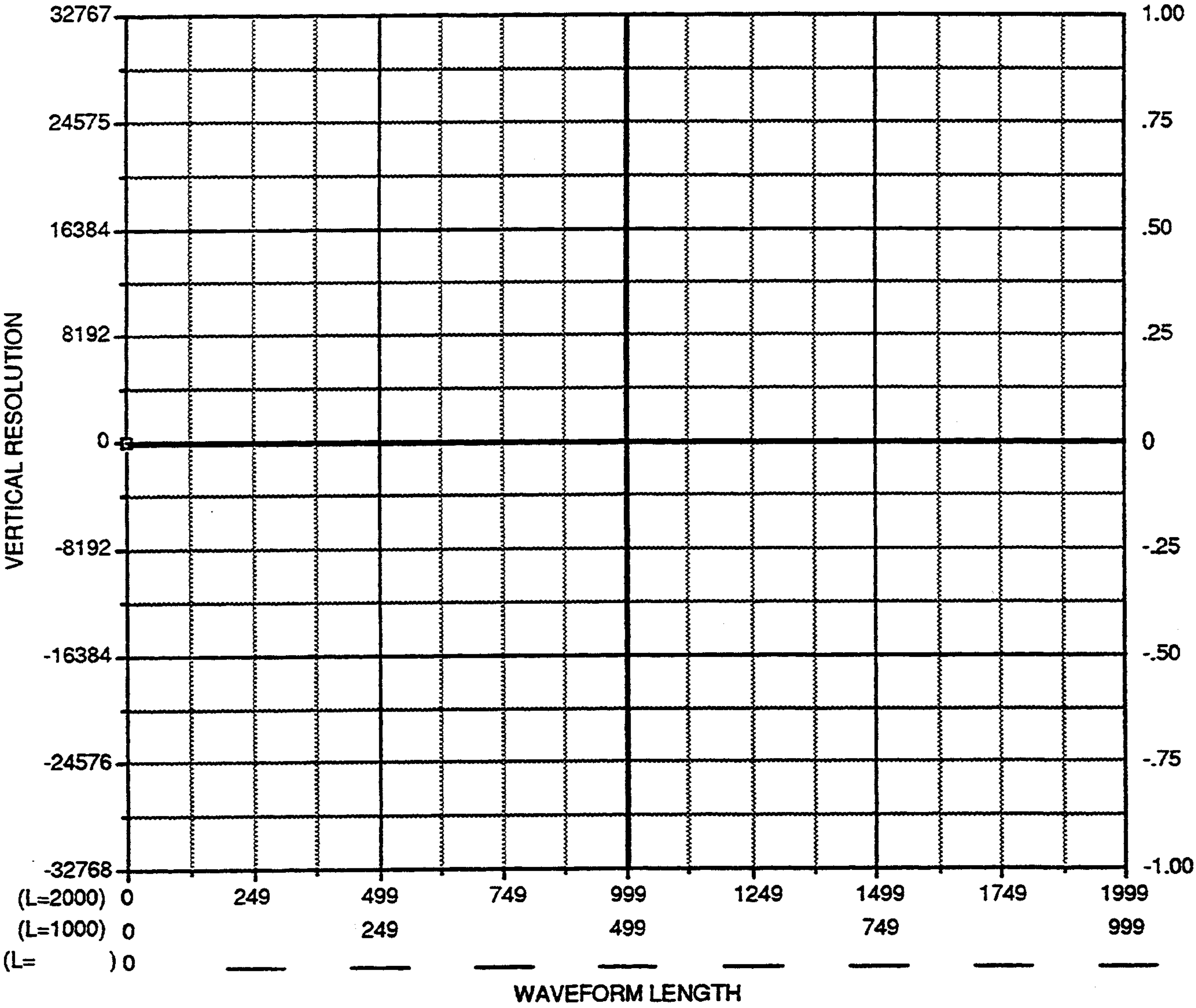
Model 2411A
(DB-9, female)



WAVEFORM DESIGN SHEET - 2411A

WAV # ________ WAVEFORM TYPE__________________________________

LENgth________



APPLICATION

WAVEFORM / SEQUENCE WORK SHEET

TEGAM 2202A / 2411A / 2414A

STANDARD WAVEFORM

	LENGTH
STDW	

SEQUENCE # ________

APPLICATION ________________________

____

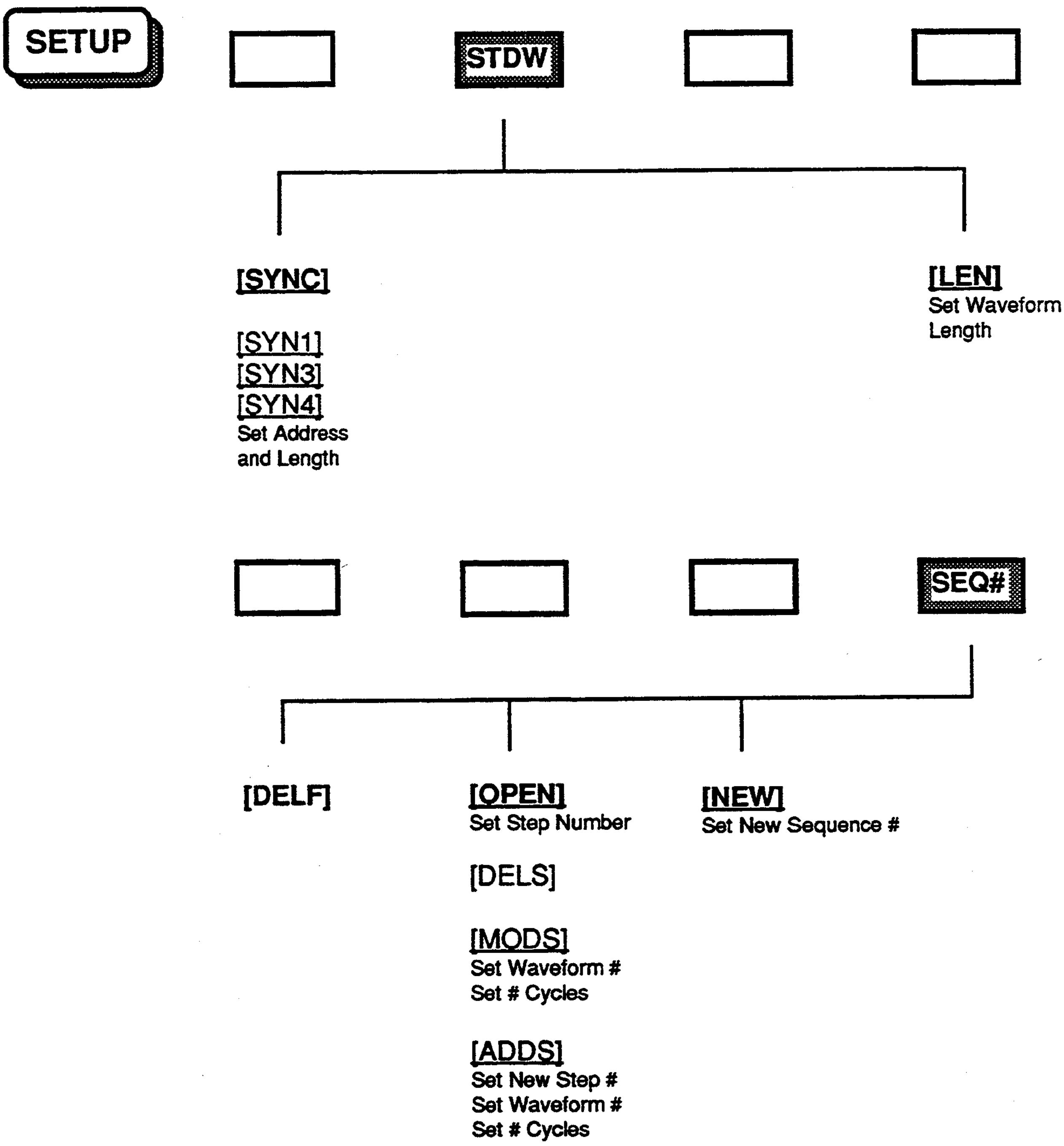
____

CUSTOM WAVEFORMS

WAV #	LENGTH	TYPE

STEP #	WAV #	NUMBER of BURST

SETUP, UTILITY AND EDIT KEY GROUP (Cont.)



SETUP

STDW

[SYNC]

[SYN1]
[SYN3]
[SYN4]
Set Address
and Length

[LEN]
Set Waveform
Length

SEQ#

[DELF]

[OPEN]
Set Step Number

[DELS]

[MODS]
Set Waveform #
Set # Cycles

[ADDS]
Set New Step #
Set Waveform #
Set # Cycles

[NEW]
Set New Sequence #

SETUP, UTILITY AND EDIT KEY GROUP (Cont.)

UTIL

SCLK

[INT]
[EXT]

RCLK

[INT]
[EXT]
[ADJ]
Set Reference
Clock Vernier

TGIN

[SYNC]
[ASNC]

TGEN

[OFF]
[ON]
Set Internal
Trigger Rep Rate

MBST

Monitor Burst
Count

GPIB

Set Address

R232

[BAUD]

[1.2K]
[2.4K]
[9.6K]
[19K2]

[PAR]

[ODD]
[EVEN]
[NONE]

[BITS]

[7D1S]
[7D2S]
[8D1S]
[8D2S]

[HAND]

[SW]
[HW]

EDIT

LINE

[STRT]
Set X Address (LX)
Set Y Address (LY)

[ANCH]
Set Left Anchor (AL)
Set Right Anchor (AR)

[CHRD]
Set Chord X
Address (CX)
Set Chord Y
Address (CY)

[ZLVL]
Set Z Amplitude

VRTX

[ADDV]
Set Vertex X
Address (VX)
Set Vertex Y
Address (VY)

[SCAL]
Set Digital
Amplitude (DA)
Set Digital
Offset (DO)

[ANCH]
Set Left Anchor (AL)
Set Right Anchor (AR)

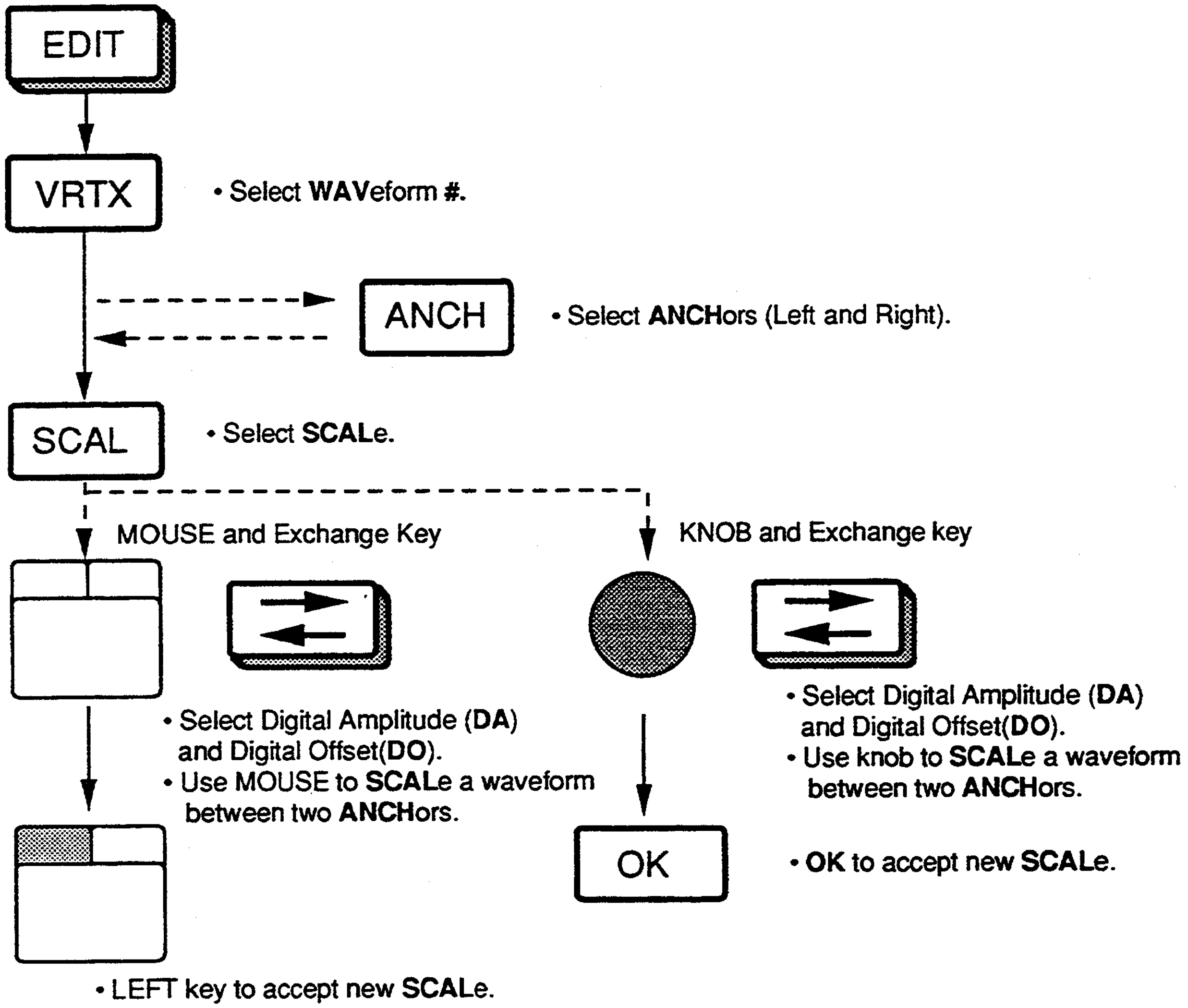
[INSF]
Select Function
[SHOW]
[-->]
Set Waveform
Parameters

MATH

[◀]
Select Three
Waveform Numbers

[OP]
[A*B]
[A+B]
[A-B]

SCALE Waveform



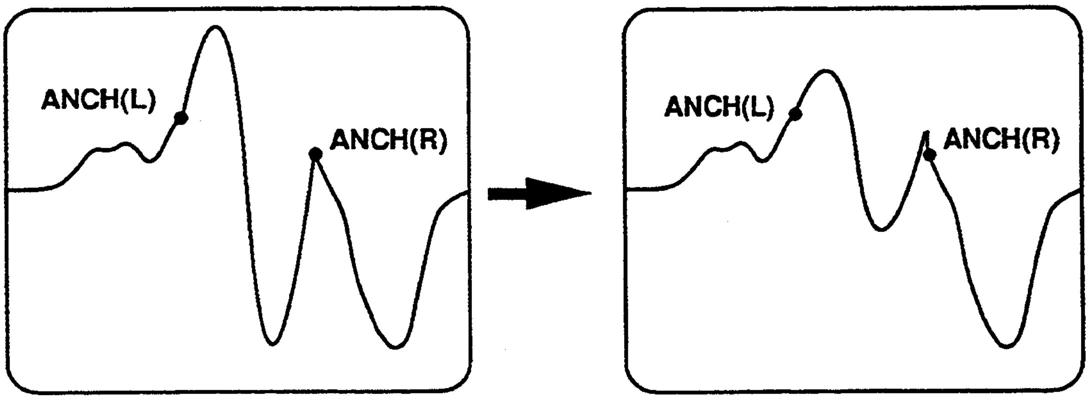
- Select **WAV**eform **#.**
- Select **ANCH**ors (Left and Right).
- Select **SCAL**e.

MOUSE and Exchange Key

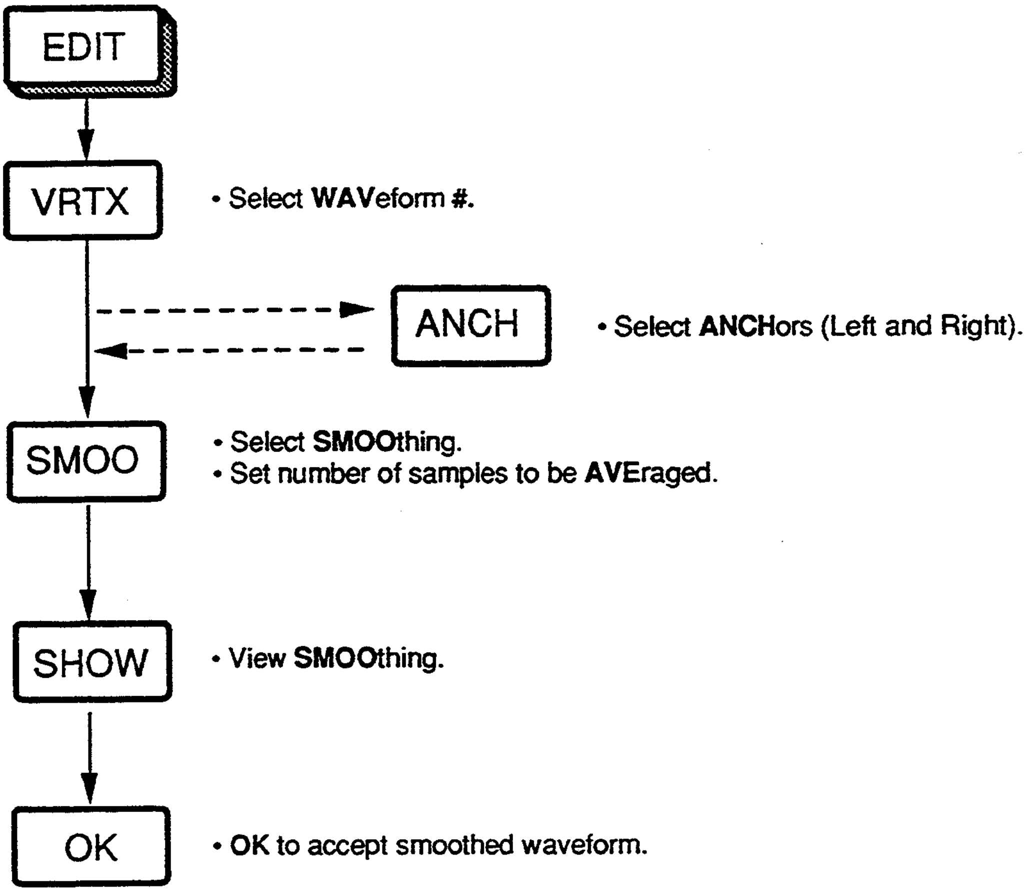
- Select Digital Amplitude (**DA**) and Digital Offset(**DO**).
- Use MOUSE to **SCAL**e a waveform between two **ANCH**ors.
- LEFT key to accept new **SCAL**e.

KNOB and Exchange key

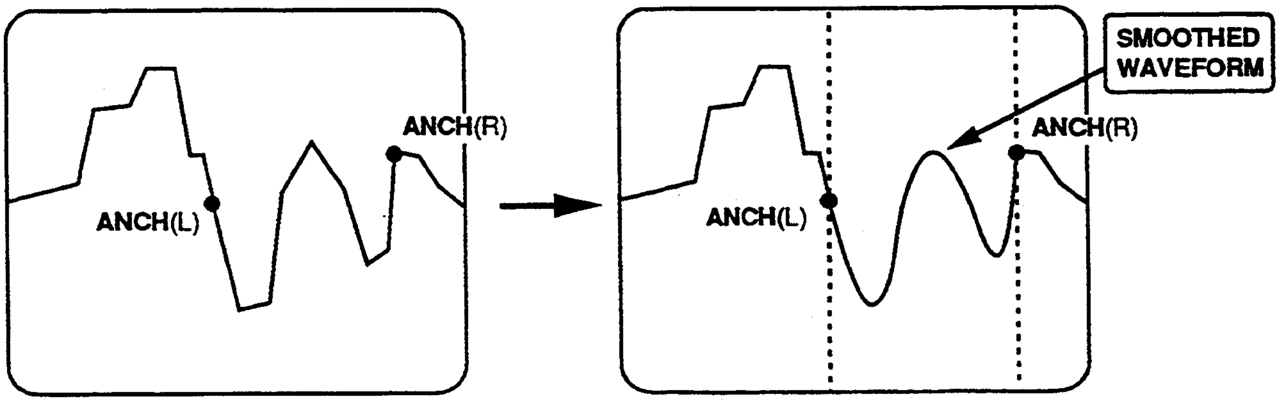
- Select Digital Amplitude (**DA**) and Digital Offset(**DO**).
- Use knob to **SCAL**e a waveform between two **ANCH**ors.
- **OK** to accept new **SCAL**e.



SMOOTH Waveform



- Select **WAV**eform **#.**
- Select **ANCH**ors (Left and Right).
- Select **SMOO**thing.
- Set number of samples to be **AVE**raged.
- View **SMOO**thing.
- **OK** to accept smoothed waveform.



Roots and First Level Commands - 2202A / 2411A

		F1	F2	F3	F4		F1	F2	F3	F4
MODE	1	CONT	TRIG	GATE	BRST	2	TOGL			
FUNC	1	WAV#	SEQ#		VIEW	4	AM	SCM	FM	HSIN
	2	SIN	SQR	TRI	SAW+	5	SLIN	SLOG	PLS+	PLS--
	3	SAW--	DC	EXP+	EXP--	6	GAUS	SX/X	CIRC	NOIS

CLOCK / FREQ
- Set Sample **CLOCK** frequency
- Monitor **FREQ**uency of each waveform frame

AMPL / OFST
- Set output **AMPL**itude and **OFfSeT**

		F1	F2	F3	F4		F1	F2	F3	F4
SETUP	1	WAV#	STDW		SEQ#					
EDIT	1	LINE	VRTX	MATH						
VRTX	1	ADDV	SCAL	ANCH	INSF	2	SUMF	DMPF	SMOO	MOVE
OUT PUT	1	ON	FLTR (2202A) 40 K (2411A)	*700K *2411A only*	»	2	SYNC	STRG	* DEGL *2411A only*	»
STOR / RECL	1	STOR	RECL							
UTIL	1	SCLK	RCLK	TGIN	TGEN	2	MBST		GPIB	R232

Menu and Display Control Keys

ENTER
- Execute and set commands and parameters

CLEAR
- Clears numerical parameters

SHIFT
- Shift numerical keypads to secondary operations

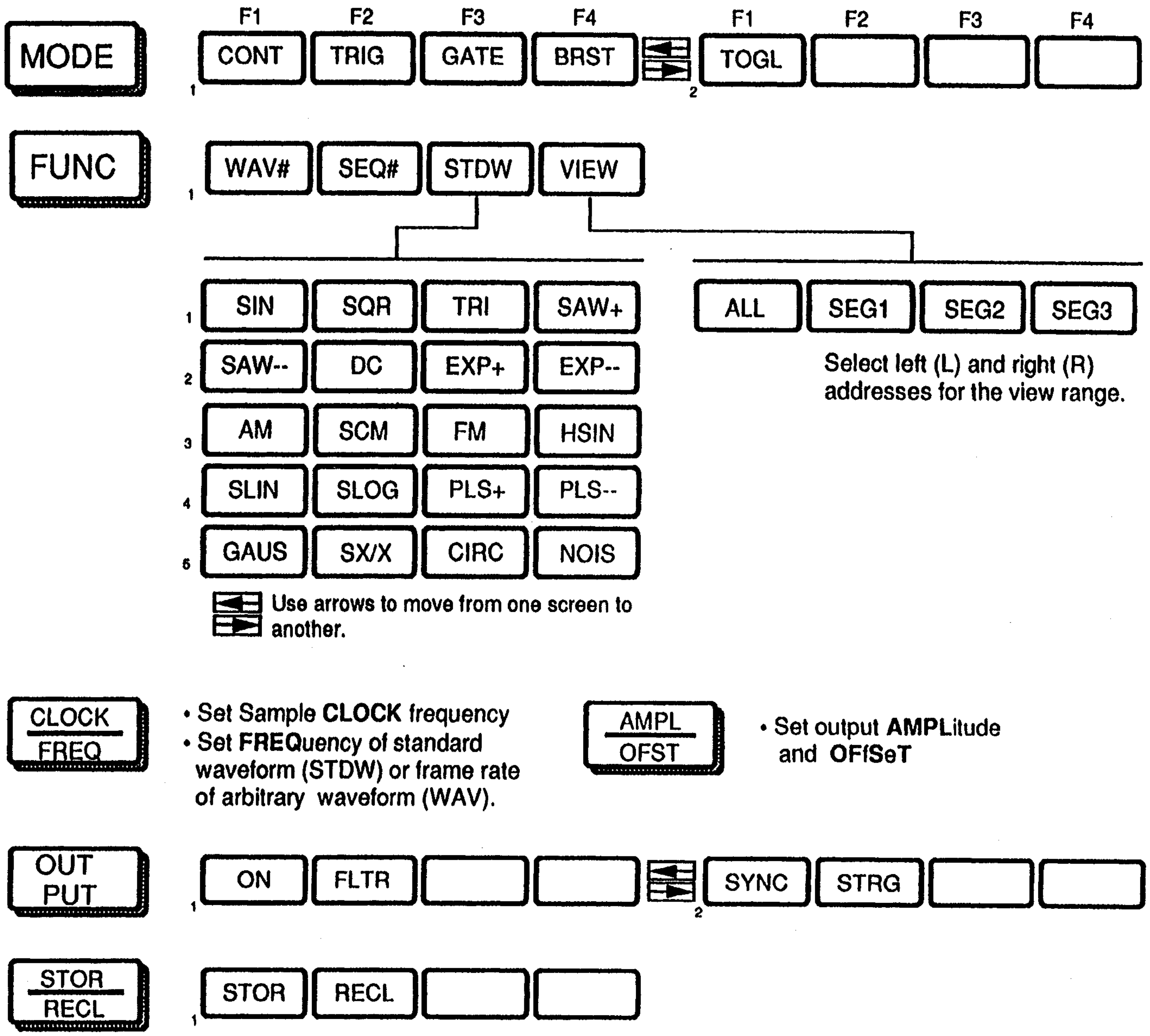
LAST
- Move up **MENU**

⇄
- Exchange Numerical Parameter field

→
- Move MENU to right

←
- Move MENU to left

Roots and Commands - 2414A



MODE

	F1	F2	F3	F4		F1	F2	F3	F4
1	CONT	TRIG	GATE	BRST	⇆ 2	TOGL			

FUNC

1	WAV#	SEQ#	STDW	VIEW

STDW:

1	SIN	SQR	TRI	SAW+
2	SAW--	DC	EXP+	EXP--
3	AM	SCM	FM	HSIN
4	SLIN	SLOG	PLS+	PLS--
5	GAUS	SX/X	CIRC	NOIS

⇆ Use arrows to move from one screen to another.

VIEW:

ALL	SEG1	SEG2	SEG3

Select left (L) and right (R) addresses for the view range.

CLOCK / FREQ

- Set Sample **CLOCK** frequency
- Set **FREQ**uency of standard waveform (STDW) or frame rate of arbitrary waveform (WAV).

AMPL / OFST

- Set output **AMPL**itude and **OF**f**S**e**T**

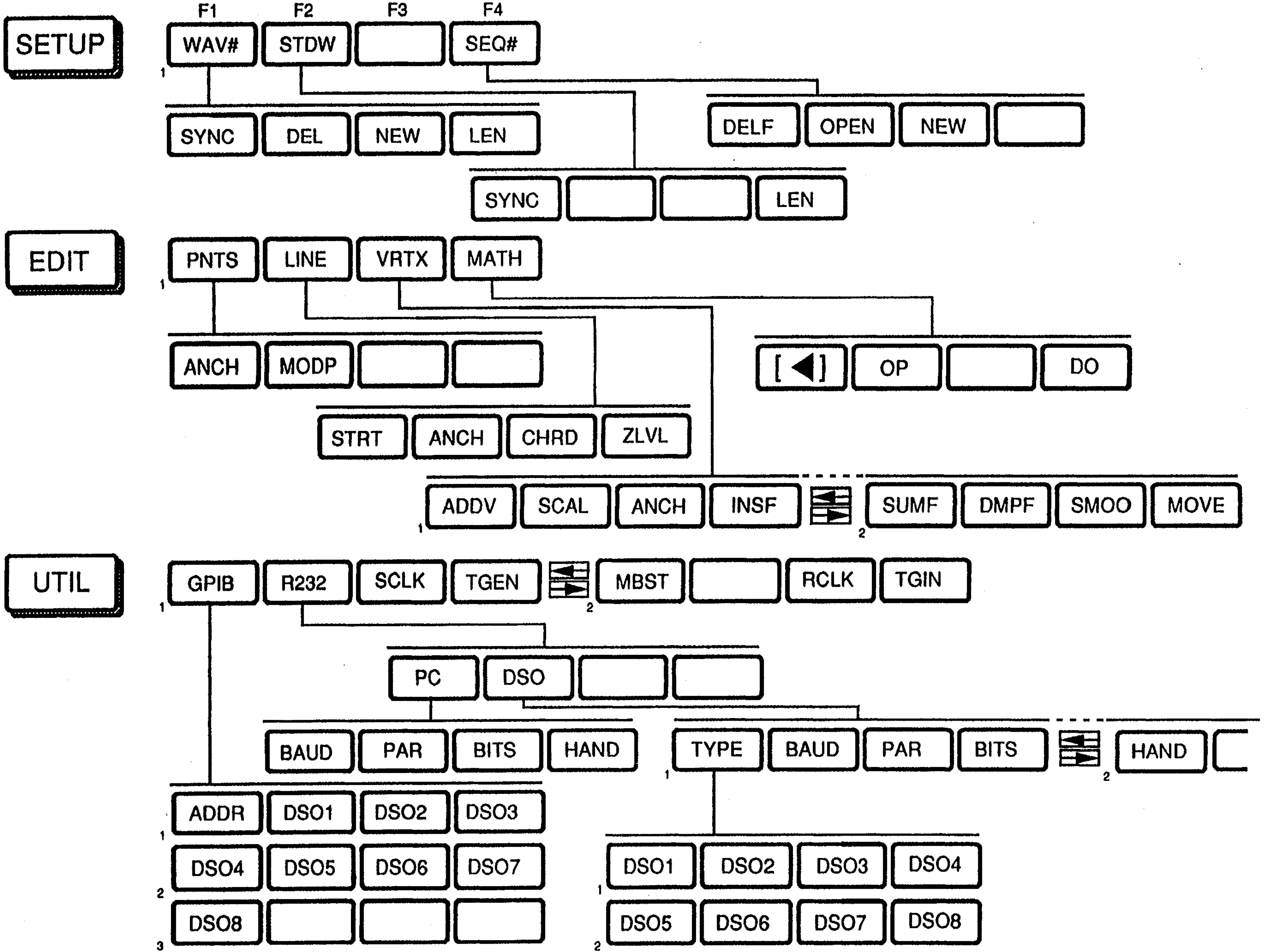
OUT PUT

1	ON	FLTR			⇆ 2	SYNC	STRG		

STOR / RECL

1	STOR	RECL		

Roots and Commands - 2414A



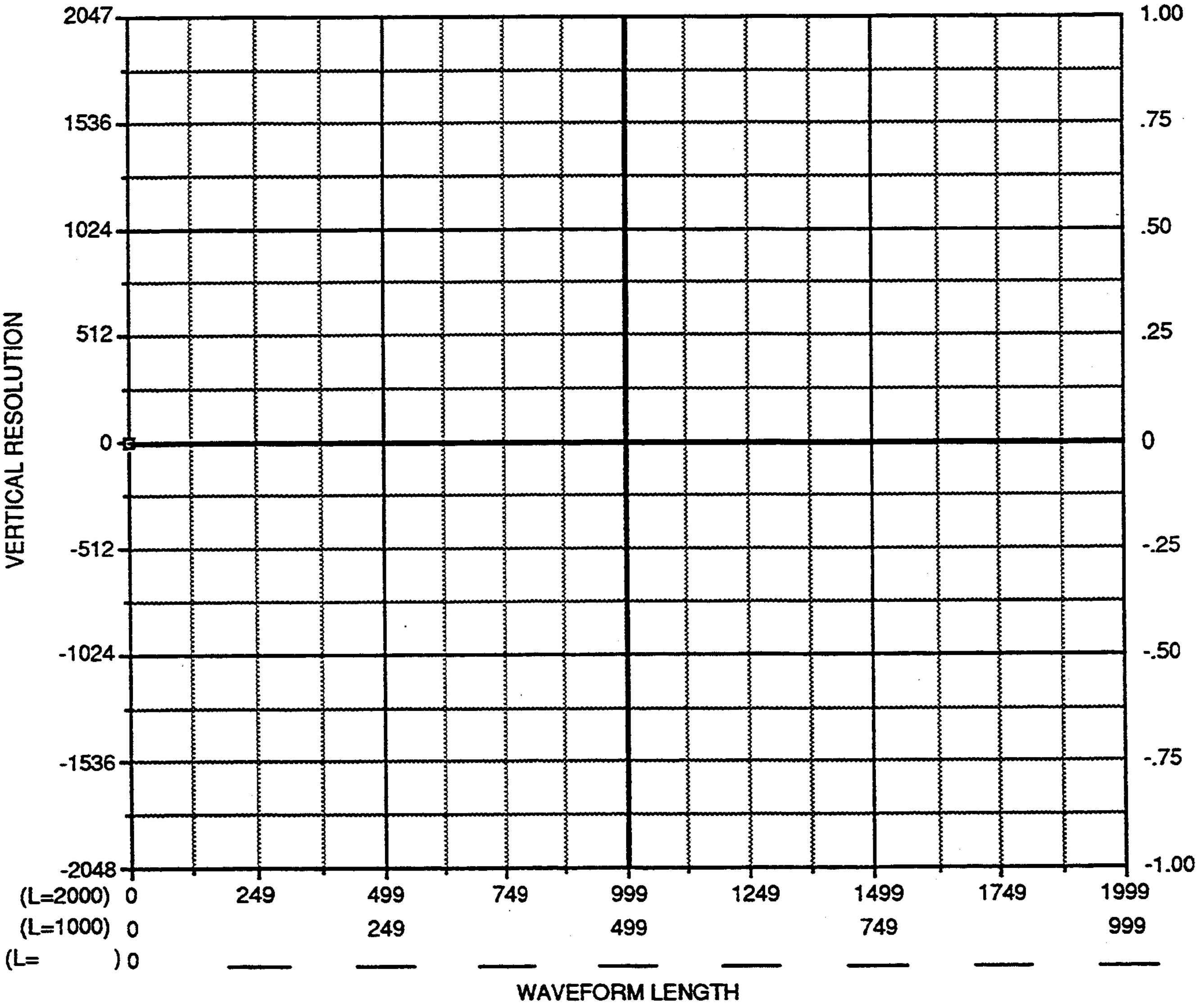
⇆ Use arrows to move from one screen to another.

⇆ Use arrows to move from one screen to another.

WAVEFORM DESIGN SHEET - 2202A / 2414A

WAV # ________ **WAVEFORM TYPE** ________________________________

LENgth ________



APPLICATION

WAVEFORM / SEQUENCE WORK SHEET

TEGAM 2411A

STANDARD WAVEFORM

▨	LENGTH
STDW	

SEQUENCE # ________

APPLICATION ____________________

____

____

CUSTOM WAVEFORMS

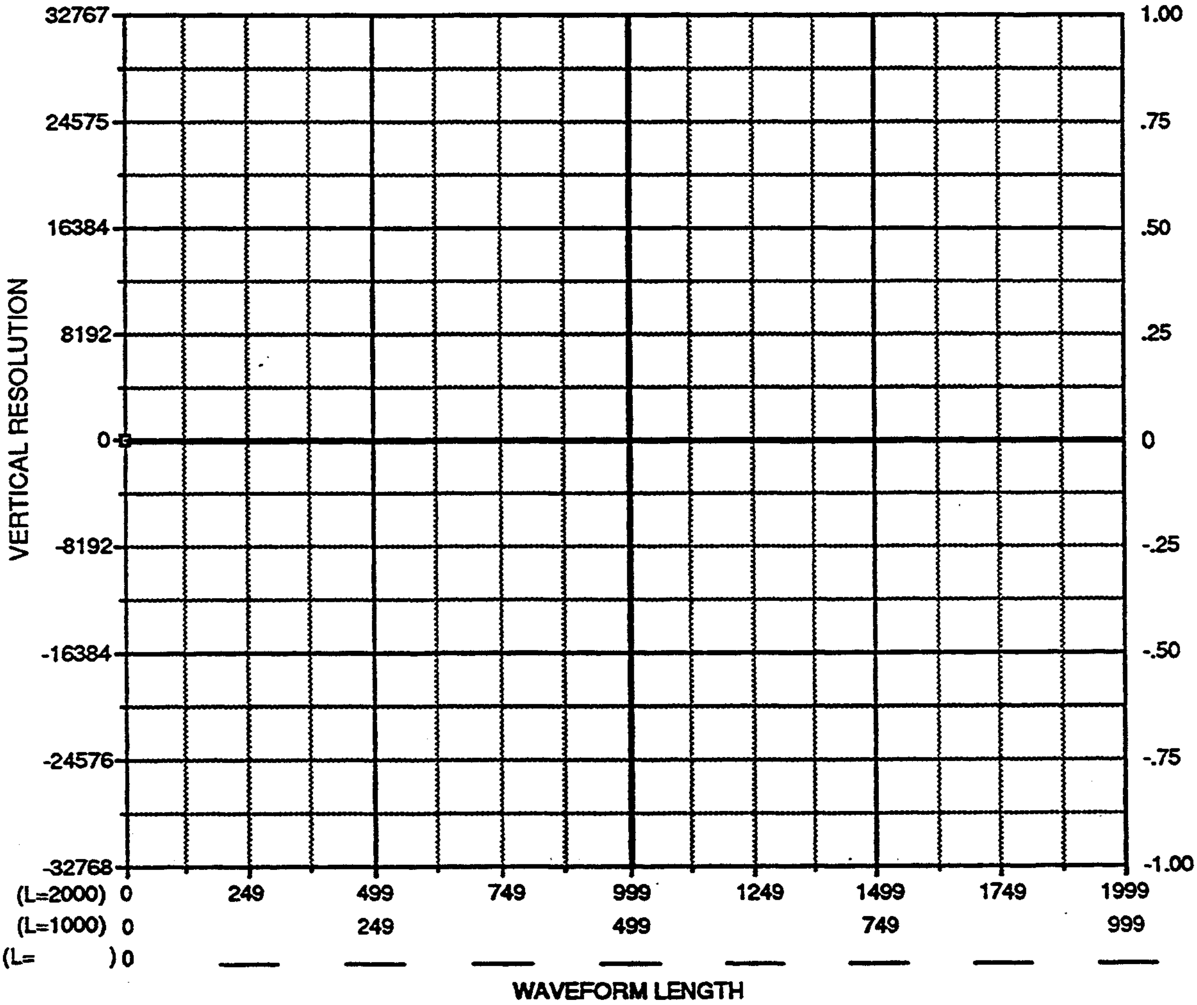
WAV #	LENGTH	TYPE

STEP #	WAV #	NUMBER of BURST

WAVEFORM DESIGN SHEET

WAV #________ WAVEFORM TYPE____________________________________

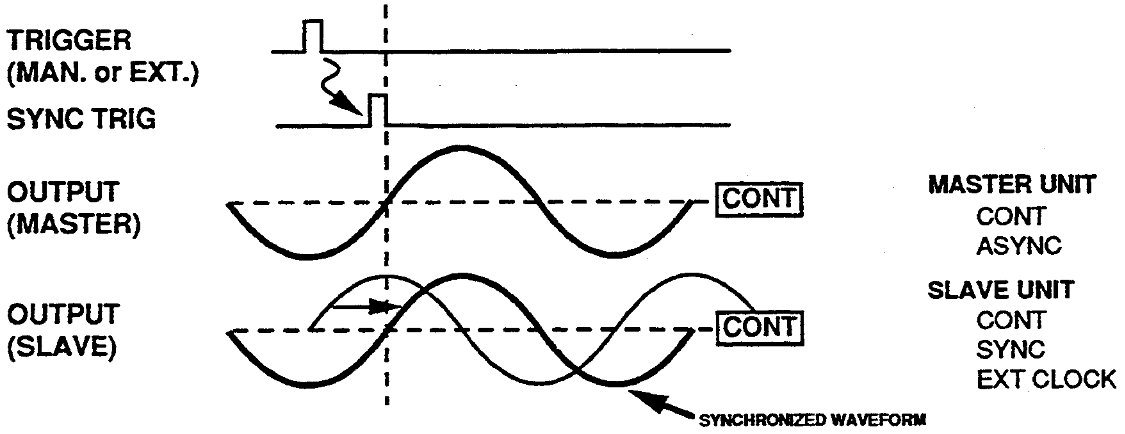
LENgth________



APPLICATION

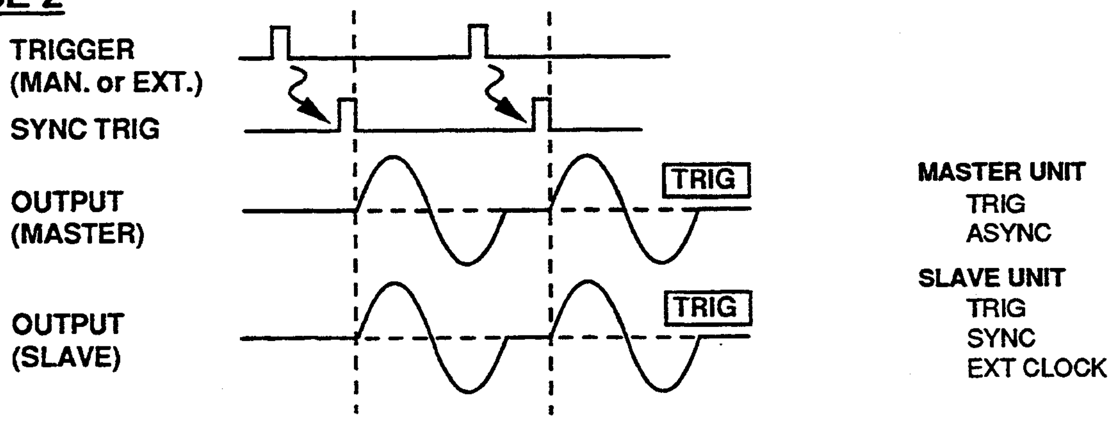
SYNCHRONOUS OPERATION - Parallel

CASE 1



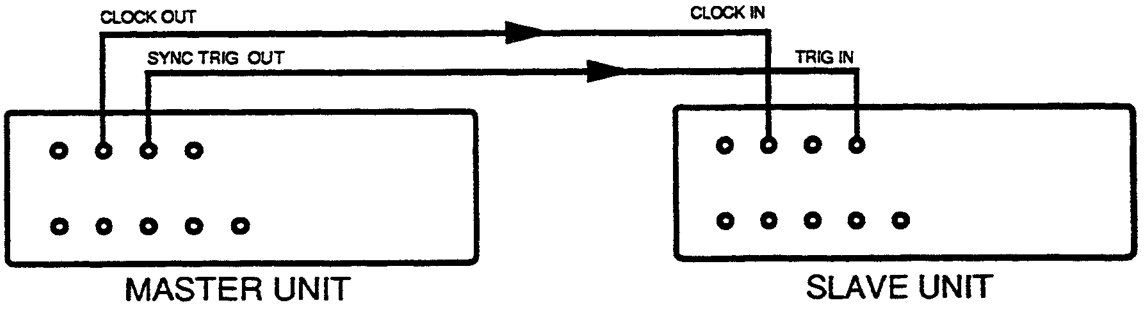
SYNC TRIG pulse synchronizes both Master and Slave units whenever Trigger (Manual or External) is given.

CASE 2



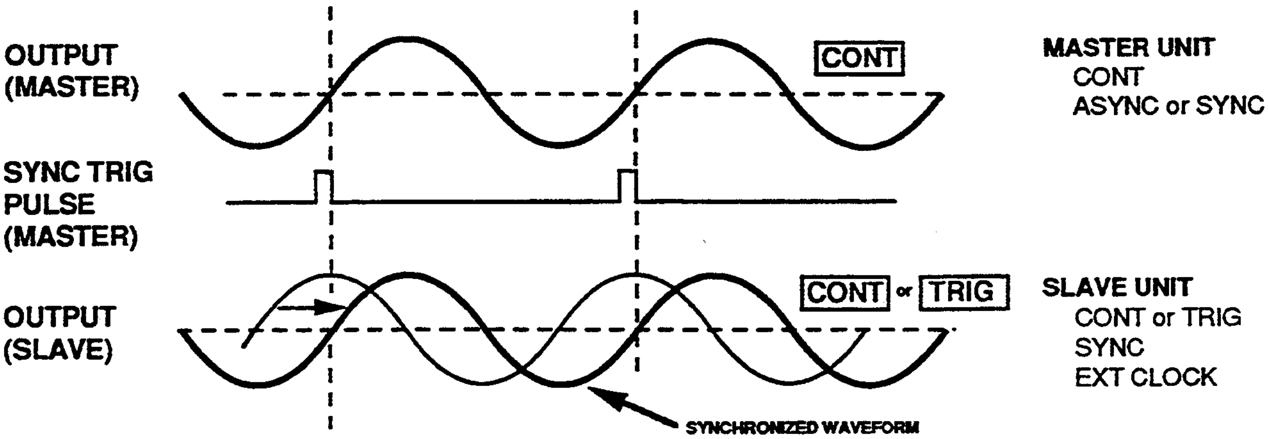
CONNECTION DIAGRAM

(REAR BNC CONNECTORS)



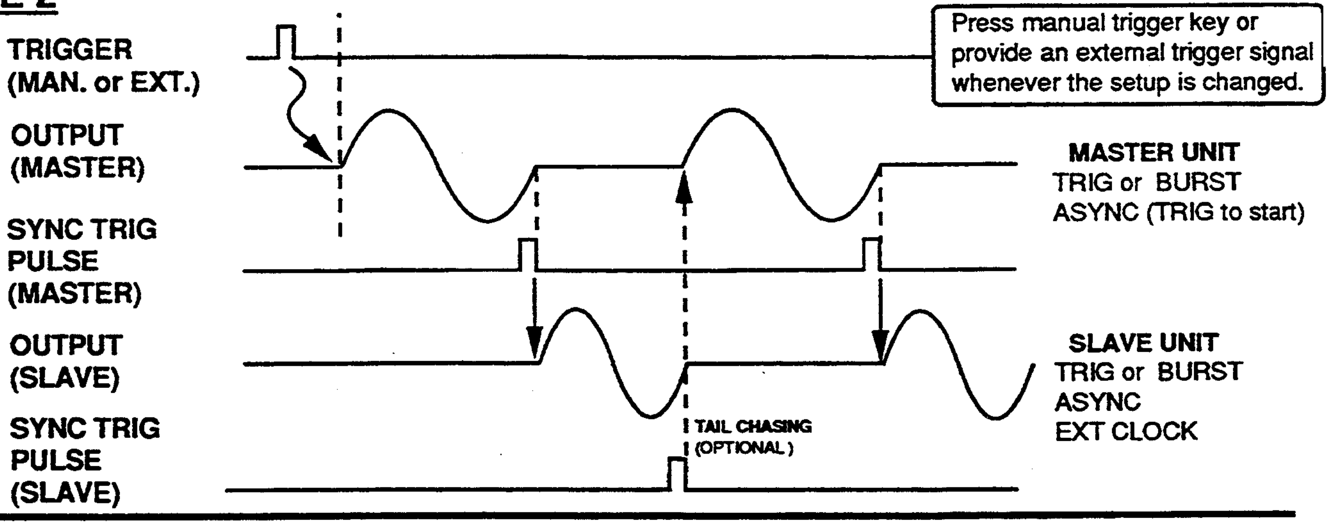
SYNCHRONOUS OPERATION - Serial

CASE 1



SYNC TRIG pulse synchronizes both Master and Slave units on every cycle.
In **CONT**inuous mode, Waveform Length of Slave unit must be **equal to** the Waveform Length of the Master unit.
In **TRIG**ger mode, Waveform Lengh of Slave unit must be **equal to** or **less than** the Waveform Length of the Master unit.

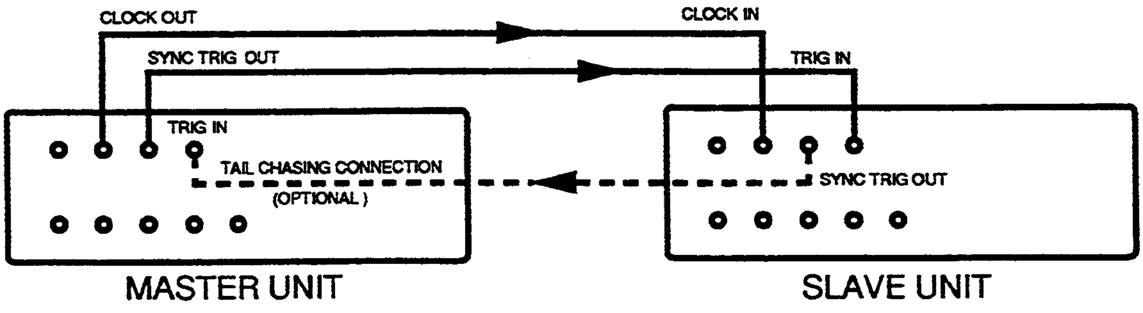
CASE 2



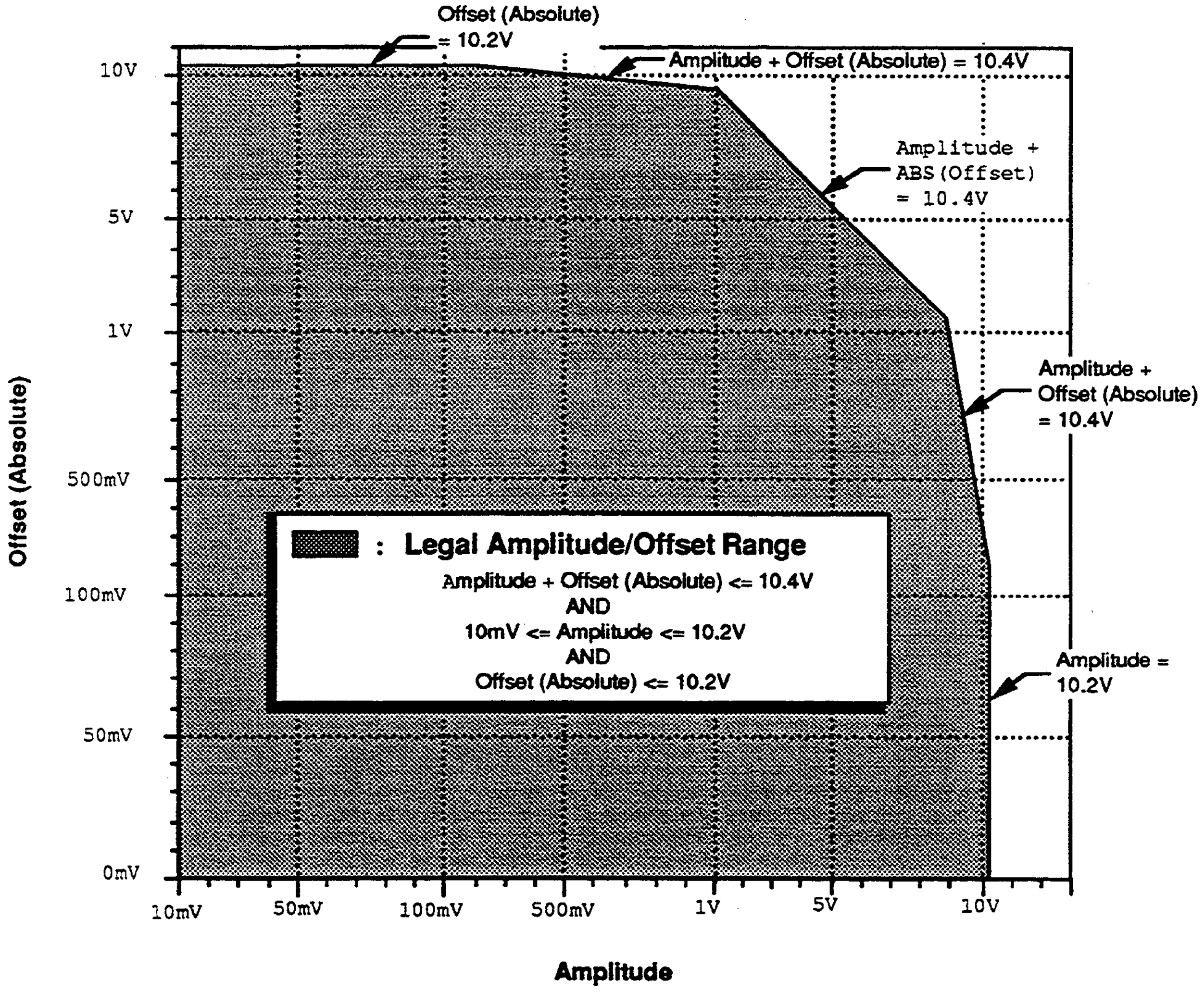
Press manual trigger key or provide an external trigger signal whenever the setup is changed.

CONNECTION DIAGRAM

(REAR BNC CONNECTORS)



AMPLITUDE & OFFSET RANGES

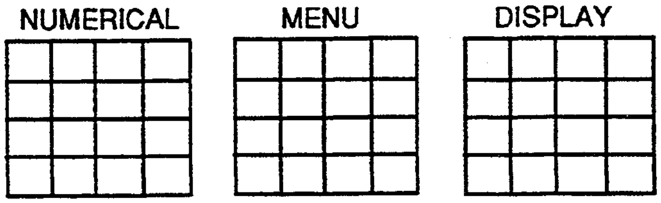


QUICK REFERENCE GUIDE

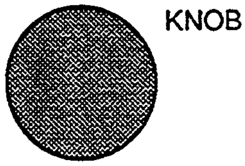
2202A / 2411A / 2414A
Quick Reference Guide

Using Your Keyboard, Knob, or Mouse

You can use a keyboard, a knob, or an optional mouse as your input device. Turn the following pages to see how the instrument may be used for your applications.



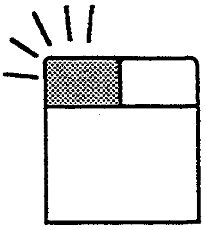
- MENU selection
- Display Control
- Numerical Entry



- Numerical Entry of Parameters
- Numerical Selection in MENU

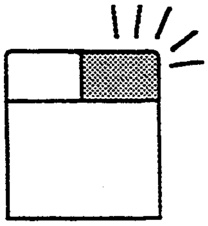
ENTER

- Complete COMMAND and Execute



MOUSE
Left Button

- Select an item in **EDIT**
- Responds **OK** to complete a command in **EDIT**



MOUSE
Right Button

- Responds to **CANCel** a command in **EDIT**

Waveform and Sequence Output

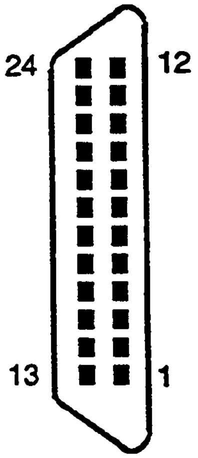
LAST	• Move up **MENU**
F 1	SOFT KEYS • Select an item above the keys
F 2	
F 3	
F 4	
⇄	• Exchange Numerical Parameter field

MODE	• Select **MODE**
FUNC	• Select **SEQ**uence,**WAV**eform, and standard function
CLOCK / FREQ	• Set Sample **CLOCK** frequency • Monitor **FREQ**uency of each waveform frame • Set **FREQ**uency of each waveform frame (2414A only)
AMPL / OFST	• Set output **AMPL**itude and **OF**f**S**e**T**
→	• Move MENU to right
←	• Move MENU to left

GPIB AND RS-232 CONNECTOR PINOUTS

GPIB

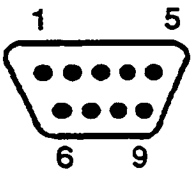
GPIB 24 PIN
(REAR VIEW)



24: GND
23: GND
22: GND
21: GND
20: GND
19: GND
18: GND
17: REN
16: DIO8
15: DIO7
14: DIO6
13: DIO5

12: SHIELD (CHASSIS GND)
11: ATN
10: SRQ
9: IFC
8: NDAC
7: NRFD
6: DAV
5: EOI
4: DIO4
3: DIO3
2: DIO2
1: DIO1

RS-232C



1: 6.2k Pulldown
2: RXD
3: TXD
4: DTR
5: Ground
6: DSR
7: RTS
8: CTS
9: NC

STANDARD WAVESHAPE EQUATIONS

The following ten equations show the mathematical basis for the algorithms used to create the indicated waveforms:

$\mathrm{fm}(t) = \sin[\omega_c t + \beta \cos(\omega_m t + \phi_m) + \phi_c]$ β = Modulation Index

$\mathrm{am}(t) = \sin(\omega_c t + \phi_c) * [1 + M \sin(\omega_m t + \phi_m)]$ M = Modulation Index

$\mathrm{scm}(t) = \sin(\omega_c t + \phi_c) * \sin(\omega_m t + \phi_m)$

$\mathrm{exp+}(t) = e^{-kt}$ k = Exponential Time Constant

$\mathrm{exp-}(t) = 1 - e^{-kt}$ k = Exponential Time Constant

$\mathrm{sinx/x}(t) = \frac{\sin(t)}{t}$

$\mathrm{gauss}(t) = e^{-t^2}$

$\mathrm{circle}(t) = \sqrt{1 - (1 - 2t)^2}$

$\mathrm{linsweep}(t) = \sin\left(\omega_{\mathrm{begin}}\, t + K \int t \, dt\right)$ K is the slope of the sweep = $\frac{(\omega_{\mathrm{end}} - \omega_{\mathrm{begin}})}{2}$

$\mathrm{logsweep}(T) = \sin\left(\omega_{\mathrm{begin}} T * \int_0^T e^{tK} dt\right)$ $K = \frac{\log(\omega_{\mathrm{end}}) - \log(\omega_{\mathrm{begin}})}{\mathrm{SweepTime}}$

INDEX

INDEX