HIOKI

INSTRUCTION MANUAL

8853

MEMORY HiCORDER

FFT FUNCTION

GP-IB INTERFACE

HIOKI E.E. CORPORATION

Contents

Safety Notes

- This Instruction Manual provides information and warnings essential for operating this equipment in a safe manner and for maintaining it in safe operating condition.
- Before using this equipment, be sure to carefully read the following safety notes. Also, be sure to read "Notes on Use" at the beginning of the 8853 Instruction Manual.

⚠ WARNING **This equipment is designed according to IEC348 Electrical Measurement Equipment Safety Standards, and has been tested for safety prior to shipment. During high voltage measurement, incorrect measurement procedures could result in injury or death, as well as damage to the equipment. Please read this manual carefully and be sure that you understand its contents before using the equipment. The manufacturer disclaims all responsibility for any accident or injury except that resulting due to defect in its product.**

Safety symbols

The following symbols are used in this Instruction Manual to indicate the relative importance of cautions and warnings.

	This symbol is affixed to locations on the equipment where the operator should consult corresponding topics in this manual (which are also marked with the ⚠ symbol) before using relevant functions of the equipment. In the manual, this mark indicates explanations which it is particularly important that the user read before using the equipment.
	Indicates a grounding terminal.
	Indicates a fuse.

⚠ DANGER	Indicates that incorrect operation presents extreme danger of accident resulting in death or serious injury to the user.
⚠ WARNING	Indicates that incorrect operation presents significant danger of accident resulting in death or serious injury to the user.
CAUTION	Indicates that incorrect operation presents possibility of injury to the user or damage to the equipment.
NOTE	Denotes items of advice related to performance of the equipment or to its correct operation.

⚠ DANGER

- **To avoid the danger of electric shock or damage to the unit, never apply more than 450 V (either AC or DC) between a pair of input units or between an input unit and the frame.**
  In particular, if a power line capable of carrying a current is connected, and applies an excess voltage, there is a danger of a short circuit accident.
- **If any metal parts of the input cables are exposed, there is a danger of electric shock. Use only the 9574 input cables supplied whose metal parts are not exposed.**
- **Normally keep all four input units installed permanently. If a unit is not fitted, it must be replaced by a blank panel. If the unit is operated with an input unit not in place, it poses a shock hazard.**
- **To prevent the danger of electric shock, be sure to ground the unit.**
  - **Connect the protective ground terminal to ground first.**
  - **Using the three-core power cord supplied provides grounding.**
  - **Insert the three-core power cord only in the socket equipped with the contact for the protective ground.**
  - **Do not use the extension cord without the protective conductor.**
  - **If using the three- to two-core conversion plug supplied, the unit is not grounded, so, connect the earth cord of the conversion plug or the protective ground terminal to ground.**
  - **Do not disconnect the protective ground.**
- **To prevent the danger of electric shock, always check that the input cables are disconnected, turn off the power switch, and remove the power cord, before replacing the fuse.**
- **To prevent fire hazard, use only a fuse with the correct voltage and current rating specified on the rear panel.**

⚠ WARNING

- **When the 8853 unit and also the measuring object are grounded, be extremely careful in connecting the grounds, because if connecting the ground of the logic probe to other than that of the measuring object, the unit or the measuring object may be damaged because of a short-circuit.**
- **To prevent damage to the 8853 unit, never exceed the limits in the table below for the various input/output terminals.**

Input/output terminals	Maximum capacity
Analog inputs	500 V DC + AC peak
EXT TRIG START STOP	-5 V to +10 V
TRIG OUT GO · NG	-20 V to +30 V 500 mA max. 200 mW max.

- **The unit should always be operated in the range of 5°C to 40°C and 35 % to 80 % relative humidity. To avoid damage to the unit, do not use in direct sunlight, in dusty conditions or in the presence of corrosive gases.**

Chapter Summary

FFT Function

Chapter 1 describes the outline and concept of FFT analysis.

Chapter 2 describes the analysis function of the FFT function.

Chapter 3 describes the screen displays of the FFT function.

Chapter 4 describes the operation procedures for FFT analysis.

GP-IB Interface

Chapter 5 gives an overview of the GP-IB interface.

Chapter 6 contains the GP-IB specifications.

Chapter 7 describes the operation procedures.

Chapter 8 describes the GP-IB command list. It can be used for the command index.

Chapter 9 describes the details of the commands.

Chapter 10 describes the example program to operate GP-IB interface.

Chapter 11 describes the method of setting plotter output.

Chapter 12 contains the standard related to the GP-IB.

Appendix Refer to this section in the event of operating problems with the GP-IB interface.

Chapter 1 What is the FFT Function?

1.1 Introduction

- This allows a Fourier transform of the sampled waveform to be calculated, giving a frequency spectrum.
- The 3 types of analyses; storage waveform [STORAGE], linear spectrum [LINEAR], and power spectrum [POWER] are possible.
- FFT analysis can be performed on waveform data sampled in the memory recorder function.

Specifications

FFT channel mode	1 channel FFT
FFT analysis channel	Any channel (channel 1 to 4)
FFT analysis mode	STORAGE Storage waveform LINEAR Linear spectrum POWER Power spectrum
Frequency range	4 Hz to 5 MHz (for the time axis range)
Dynamic range	72 dB (theoretical value)
Number of samples	800 points
Frequency resolution	1/400

■ Window function	RECTAN	Rectangular window function
	HANNING	Hanning window function
■ Display format	SINGLE	Single screen display
	DUAL	Dual screen display
■ X-axis setting	Time	Time axis display
	Linear-Hz	Frequency display
	Log-Hz	Displays the frequency spectrum using a logarithmic scale.
■ Y-axis setting	Linear-Real	Indicates the real-number part of the data in voltage units.
	Linear-Imag	Indicates the imaginary-number part of the data in voltage units.
	Linear-Mag	Indicates the data in voltage units.
	Log-Mag	Indicates the data in decibels.
	Phase	Indicates the phase information in degrees.
■ Reference data	NEW DATA	Calculated as waveform is read
	MEM DATA	Calculation done from waveform in memory
■ Intermittent compression	Can be set in MEM DATA. (×1, ×1/2, ×1/5)	
■ Display scale	AUTO	Automatic setting
	MANUAL	Manual setting (-9.999E+9 to +9.999E+9)
■ Waveform decision function	Can be set in SINGLE format. (OUT, ALL-OUT)	
■ Averaging interval	OFF, 2 to 256	
■ A and B cursors	Trace cursor	

1.2 Finding Reference Material in this Manual

(1) FFT analysis (See Chapter 2.)

FFT analysis settings are described in Chapter 4.

(2) Trigger functions (See Chapter 8 of the 8853 Instruction Manual.)

Depending on the application, there is a wide range of trigger types to choose from.

(3) Waveform decision function (See Section 4.22.)

- The FFT analysis waveform decision is based on an arbitrary area.
- This allows abnormal waveforms to be detected and recorded.

(4) Data saving (See Section 4.21.)

The floppy disk, hard disk, and magneto-optical disk drive provide a storage mechanism for measurement data, setting information and waveform decision areas.

(5) Averaging function (See Section 4.7.)

This enables noise components to be removed.

(6) Comment function (See Section 12.4 of the 8853 Instruction Manual.)

This provides a convenient means of annotating lists.

(7) Screen auto off function (See Section 12.5.1 of the 8853 Instruction Manual.)

Turns the display off automatically if no user operation occurs for 10 minutes.

(8) Grid setting (See Section 12.5.2 of the 8853 Instruction Manual.)

The grid can be selected as required on the graph.

(9) Start key backup function (See Section 12.5.3 of the 8853 Instruction Manual.)

If the power fails during measurement, using this function enables measurement to restart when power is restored.

(10) Beep sound setting (See Section 12.5.5 of the 8853 Instruction Manual.)

When an error occurs, or a warning is issued, or when a waveform decision produces the result NG (fail), it is possible to arrange for a "beep" sound to be produced.

(11) List and gauge functions (See Section 12.5.6 of the 8853 Instruction Manual.)

Voltage axis scales and listings of settings on printed recordings.

(12) GP-IB interface (See Chapter 5 to 12.)

Connection to a computer via the GP-IB interface.

(13) Self check functions (See Section 12.8 of the 8853 Instruction Manual.)

This performs simple tests on the unit's functioning.

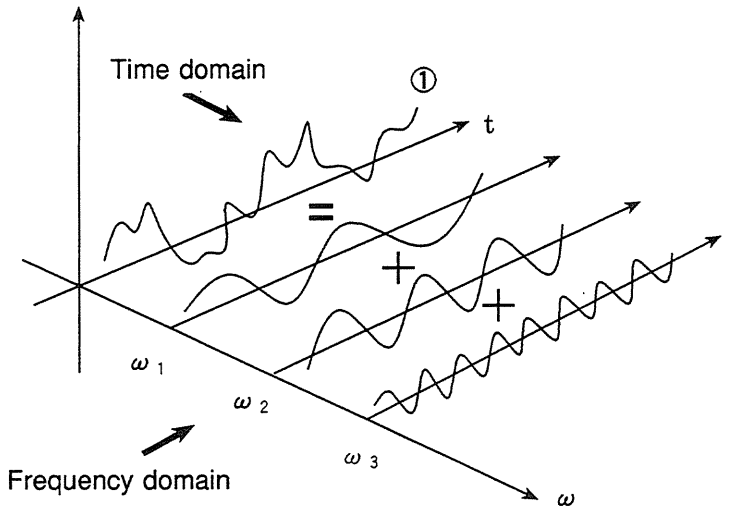
1.3 Basic Concept of FFT Analysis

- FFT stands for Fast Fourier Transform, which is calculation method used to decompose a time domain waveform into frequency components.
- By using the FFT calculations, various analyses can be made.
- There are the following two methods for performing FFT analysis on the unit.
  - ① NEW DATA — Samples the new waveform data in the FFT function, and performs FFT analysis.
  - ② MEM DATA — Performs FFT analysis on the waveform data sampled in the memory recorder function.

- Analysis results can be displayed in graphic form, and printed out.

(1) Concept of time domain and frequency domain

- The signals measured by this memory recorder have values which correspond to time, that is the signals are functions of time.
  Waveform ① shown in the figure below is an example of such a signal. Signals which are expressed as a function of time are called time domain signals.
- In reality, a signal consists of a number of sine-waves of different frequencies, called frequency components, which combine to create the final shape of the waveform. Expressing waveform ①, the source signal, as a function of its frequency components yields a frequency domain representation.
- Often, the characteristics of a signal which cannot be easily analyzed in the time domain, can be clearly revealed by the frequency domain representation.



(2) Physical meaning of Fourier transformation analysis

The following equations define the Fourier transformation and the Inverse Fourier transformation.

$$F(\omega) = \Im[f(t)] = \int_{-\infty}^{+\infty} f(t) \cdot \exp(-j\omega t)dt \qquad (\mathrm{I})$$

$$f(t) = \Im^{-1}[F(\omega)] = \frac{1}{2\pi}\int_{-\infty}^{+\infty} F(\omega) \cdot \exp(j\omega t)d\omega \qquad (\mathrm{II})$$

j	imaginary number unit
$f(t)$	non-cyclic function
$\Im$	Fourier transformation
exp	natural logarithm

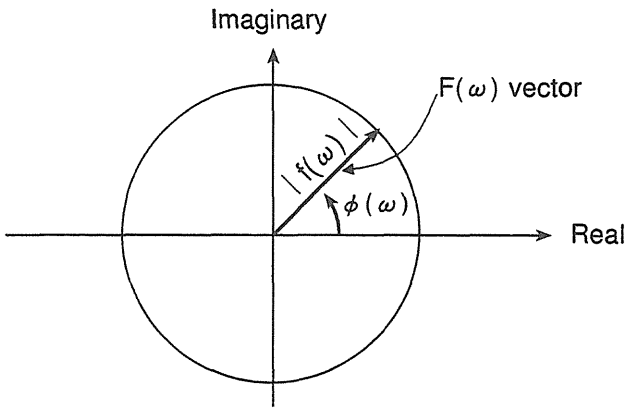
The function $F(\omega)$ generally results in a complex number, and can be expressed as follows.

$$F(\omega) = |F(\omega)| \cdot \exp(j\phi(\omega)) = |F(\omega)|\angle\phi(\omega) \qquad (\mathrm{III})$$

$$\Im[f(t)] = F(\omega) \qquad (\mathrm{IV})$$

$\lvert F(\omega) \rvert$	Absolute-value spectrum of f(t) (Magnitude spectrum)
$\phi(\omega)$	Unit spectrum of the phase of f(t)

When conversion is made from the time domain to the frequency domain, the magnitude information and phase information are clearly expressed as indicated in equation (III). The figure below shows F (ω) in vector form.



1.4 Aliasing Distortion

(1) A/D conversion

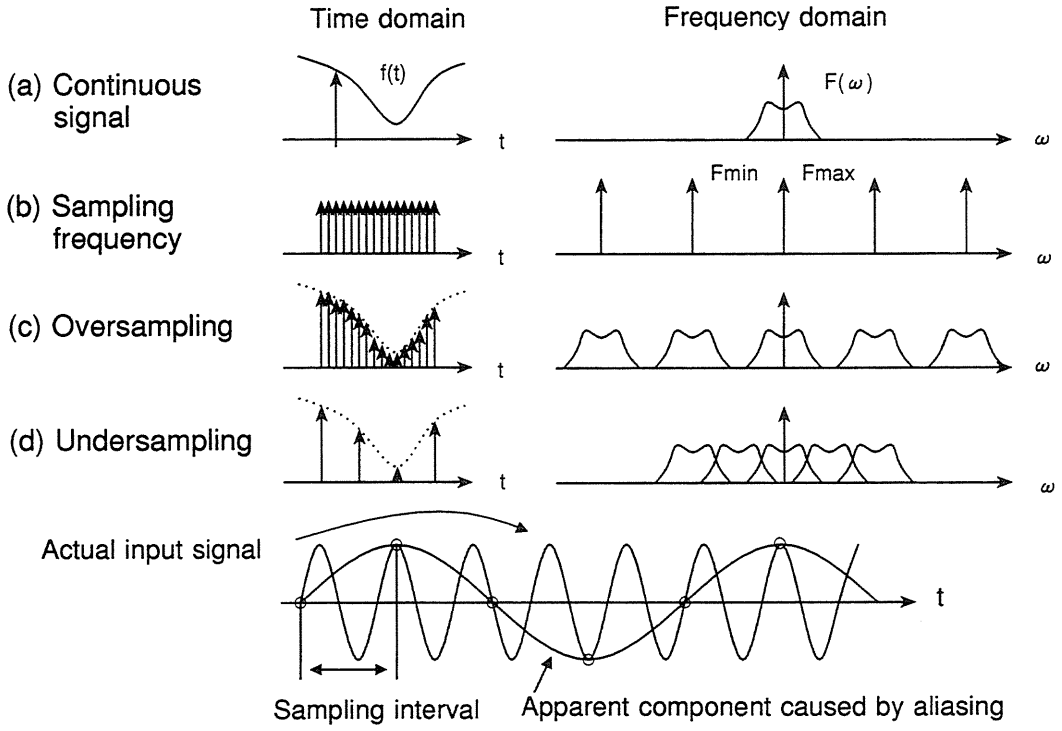
- In the 8853, the input signal is converted from analog into digital form (A/D conversion), and all processing is performed on the digital values. (This process of A/D conversion is referred to as sampling.)
- If the frequency of the signal being measured is significantly higher than the sampling rate (sampling interval), it is possible for sampling to produce an apparent signal which is actually nonexist.
- The phenomenon where the spectrum of an undersampled signal overlaps onto images of itself (as shown in the figure below, (d)) is referred to as frequency aliasing.
- The sampling theorem gives the lowest sampling frequency before the spectra begin to overlap. This sampling frequency is known as the Nyquist frequency.

  $Fs = 2 \cdot Fmax$

  Fmax Highest frequency in input signal

  Fs Nyquist frequency

- If sampling is performed with a sampling frequency which is lower than the Nyquist frequency, the digital signal appears to contain frequency components which do not exist in the original signal.



(2) Anti-aliasing filter

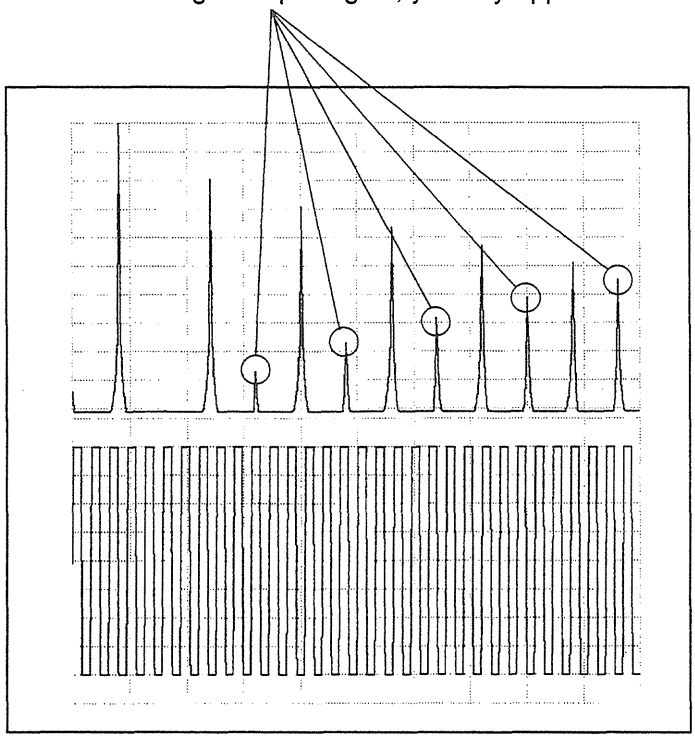
- If the input signal is regarded as having an unlimited bandwidth, the components exceeding the Nyquist frequency exist, and aliasing distortion is a consequence of sampling.
- For an FFT operation, a consequence of aliasing distortion is that a number of frequency spectra appear that do not actually exist in the original input signal.
- In order to prevent this phenomenon, provide, before the sampling, a low-pass filter having a cut-off frequency half of the sampling frequency. (This filter is called an anti-aliasing filter.)

With the 8853, this anti-aliasing filter function is not available. Kindly use it bearing in mind the characteristics of digital signals.

These spectra are caused by aliasing distortion of frequency components which are higher than half the sampling frequency of the A/D converter. (They do not exist in the original input signal, yet they appear in the spectrum.)

FFT calculated waveform (Power spectrum)

Storage waveform

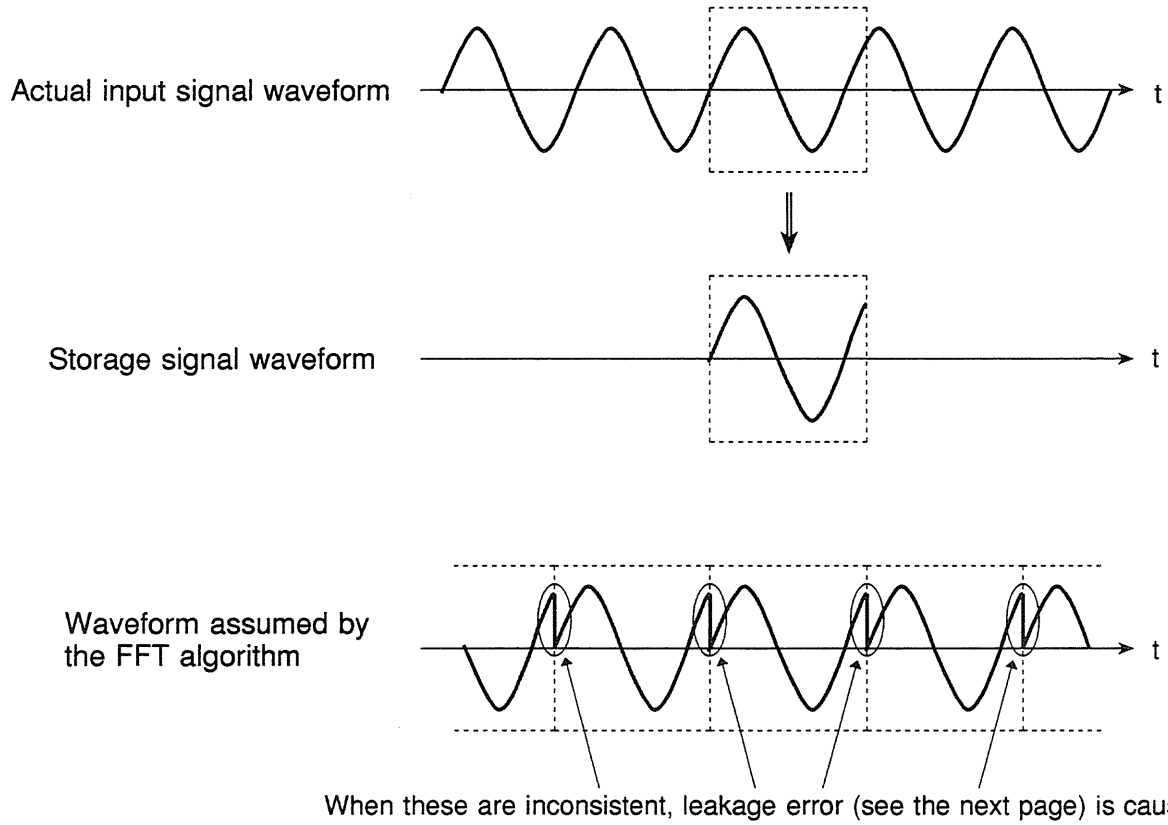


The edge of the square-wave contains very high-frequency components.

1.5 Windows and Leakage

(1) Window processing

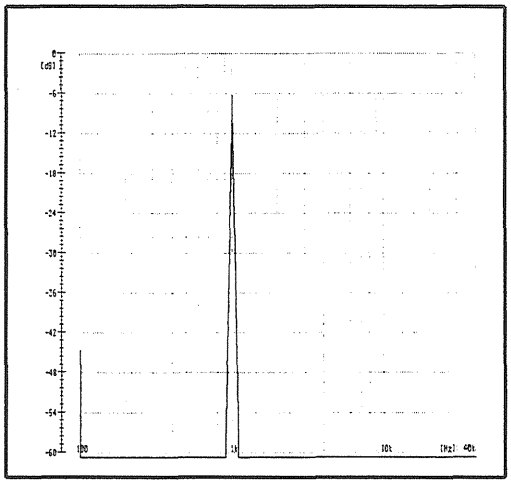
- The Fourier transformation theorem is defined as an integration between negative infinity and positive infinity. However, for actual measurements, this calculation is impossible.
- Therefore, only a limited portion of the continuous signal is clocked in (This is called window processing.), and the frequency spectrum is calculated.
- In terms of the FFT algorithm, the input signal is assumed to be a periodic function for the calculation, and it is assumed that the data for this limited time period is repeated.



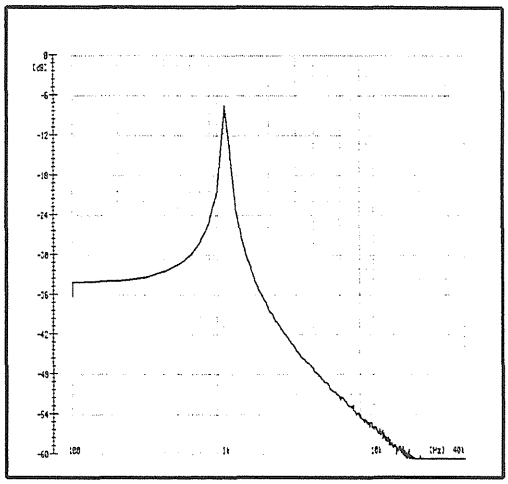
Due to the phase difference between the beginning and end points of the waveform of the stored signal, the waveform acquired by the FFT algorithm may differ from the waveform of the actual input signal.

(2) Leakage error

- Differences between the waveform of the signal acquired by the FFT algorithm and the waveform of the actual signal cause the error in the calculation results. (This error is called leakage.)
- Leakage error is caused by the fact that the values of the beginning and end points of the acquired (limited time period) signal acquired by window processing are inconsistent.



Spectrum Having Small Leakage
The width of the spectrum is narrow.

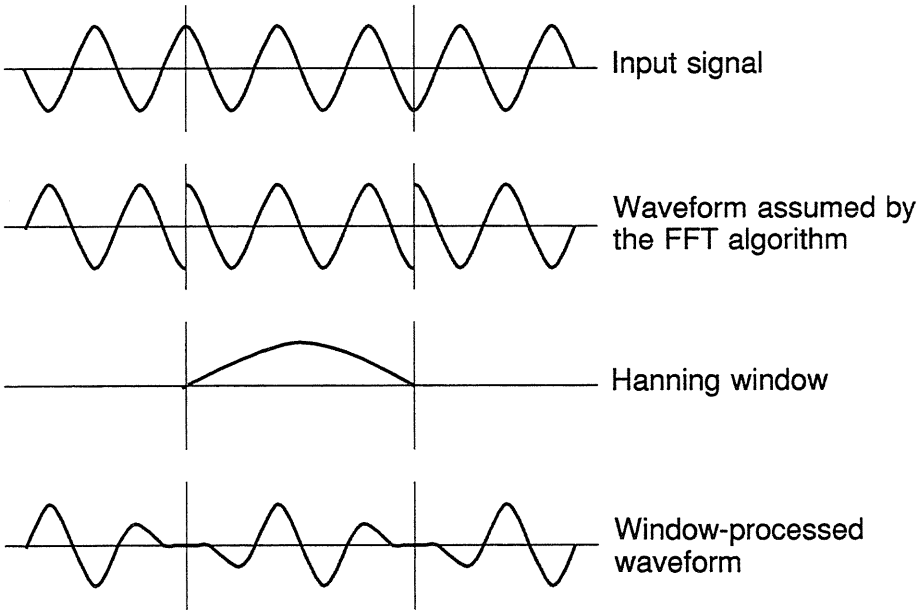


Spectrum Having Large Leakage
The spectrum spreads over a wide frequency range.

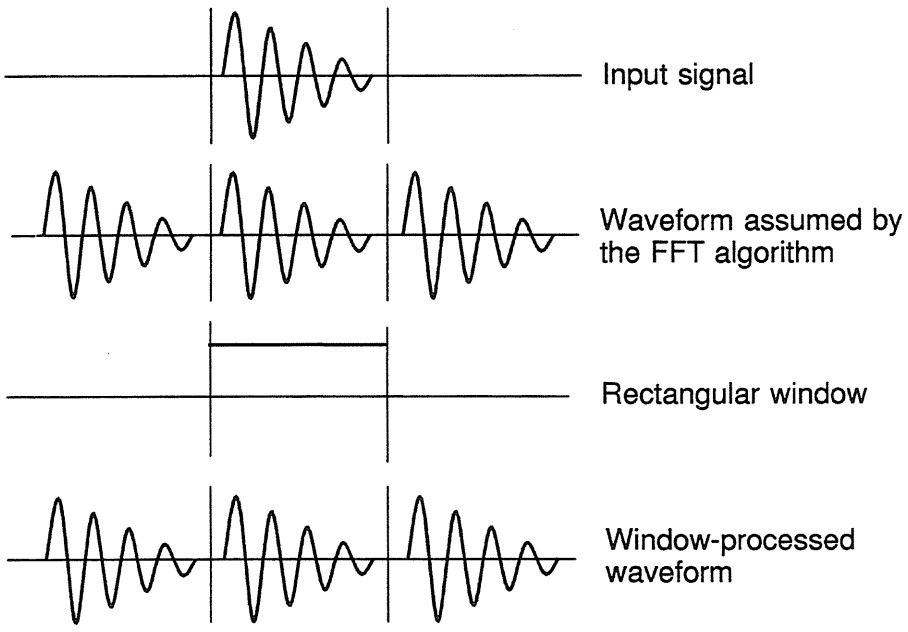
(3) Window function

- When sampling an input signal over a limited time period, the leakage error can be reduced by modifying the input signal as it is sampled.
- For input of a periodic function, a spectrum having a small leakage error can be obtained by performing an FFT operation emphasizing the middle portion of the clocked-in waveform.
- The function applied to the input values when clocking in the input signal, to reduce the leakage error is referred to as window function.
- For the 8853, the Hanning and rectangular window functions are used.
- Basically, the "rectangular" window is effective for single waveforms, and the "Hanning" window is effective for continuous waveforms.

Hanning window



Rectangular window



Chapter 2 Analysis Functions

2.1 Storage waveform [STORAGE]

■ Displays the time domain waveform of the input signal.

Function fa

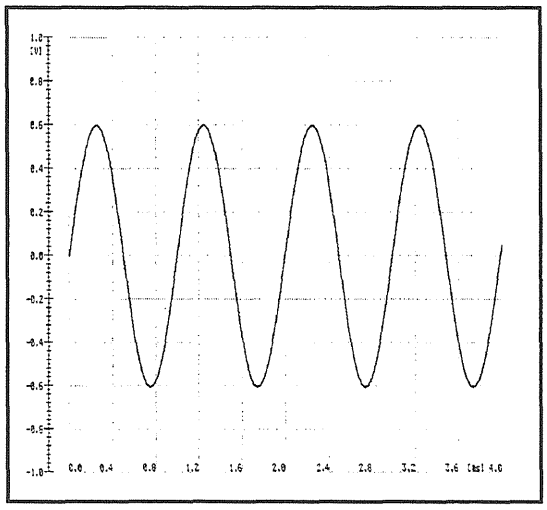
Horizontal axis x-axis (time)

- Time axis display
- The same as the memory recorder. Indicates the value of the specified TIME/DIV frequency range. (See Section 4.3.)

Vertical axis y-axis (volt)

- Voltage axis display
- Indicates the value of the voltage axis range of the input unit.

Example



When the gauge function is in use. (See Section 12.5.6 of the 8853 Instruction Manual.)

2.2 Linear spectrum [LINEAR]

- The frequency domain waveform of the input signal
- Includes magnitude and phase information.
- Major applications
  - Determining the peaks (maxima) of waveform frequency components.
  - Determining the levels of high and low harmonics.
  - Determining the frequency characteristics of a filter or the like by using an impulse signal.

Function

$$Fa = \Im(fa)$$
$$= |Fa| \cdot \exp(ja)$$
$$= |Fa| \cdot (\cos \angle a + j \sin \angle a)$$

Vertical axis	Meaning
Linear Real (real-number part)	| Fa | · cos ∠ a
Linear Imag (imaginary-number part)	| Fa | · sin ∠ a
Linear Mag (magnitude)	| Fa |
Log Mag (logarithmic magnitude)	20·log | Fa |
Phase (phase)	∠ a

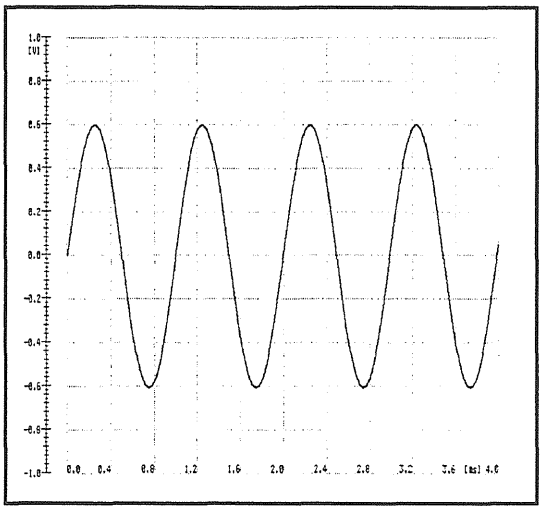
Horizontal axis x-axis

Linear Hz	Indicates the frequency spectrum in linear units. (Range: from DC to the frequency range value)
Log Hz	Indicates the frequency spectrum in logarithmic units. (Range: from 1/400 the frequency range value to the frequency range value)

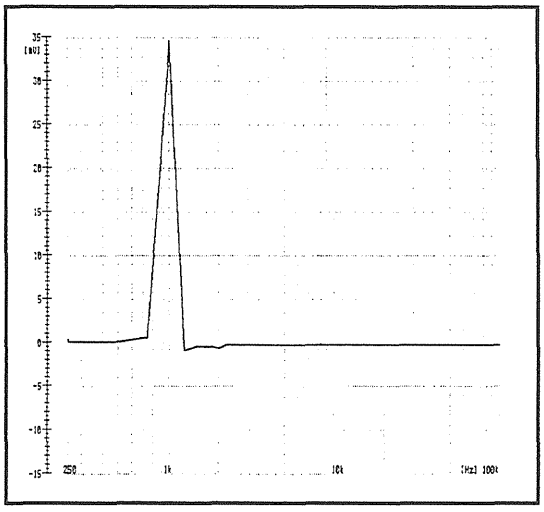
Vertical axis y-axis

Linear Real	Indicates the real-number part of the data in voltage units (V).
Linear Imag	Indicates the imaginary-number part of the data in voltage units (V).
Linear Mag	Indicates the data values in voltage units (V).
Log Mag	Indicates the data in logarithmic units (dB). (0 dB reference value: 1 V peak, 1 Vrms)
Phase	Indicates the phase information in degrees.

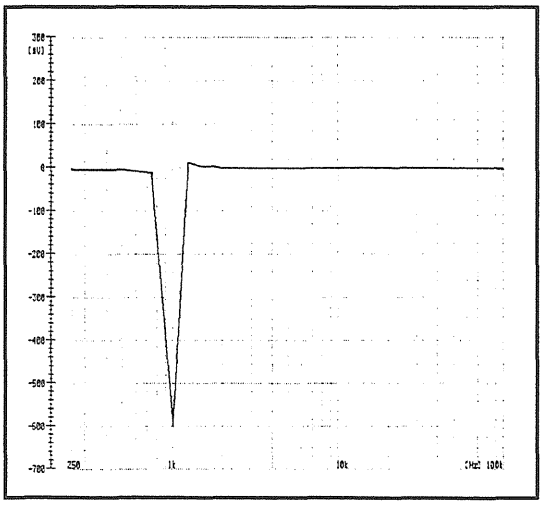
Example



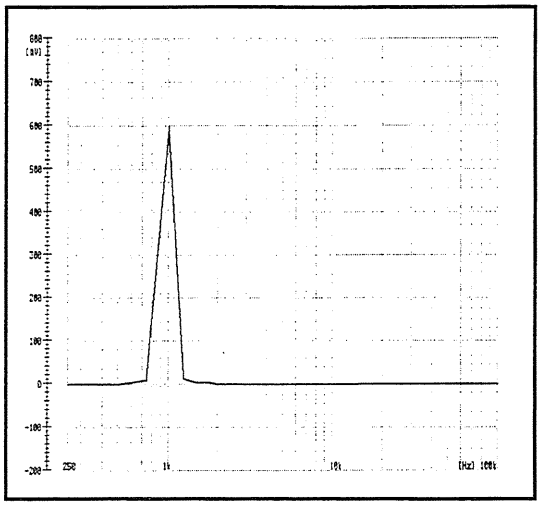
Storage waveform



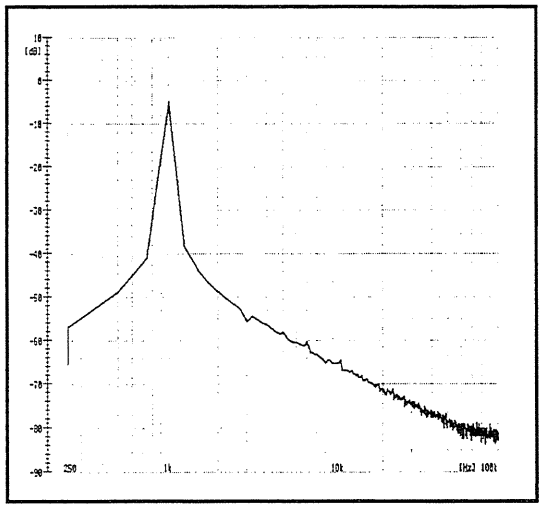
Y-axis: Linear Real
X-axis: Log Hz



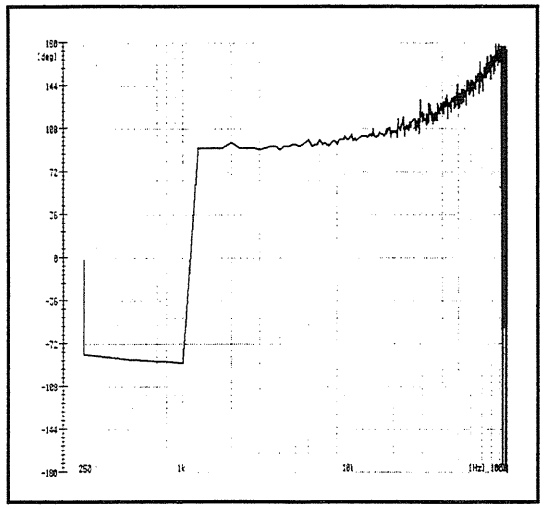
Y-axis: Linear Imag
X-axis: Log Hz



Y-axis: Linear Mag
X-axis: Log Hz



Y-axis: Log Mag
X-axis: Log Hz



Y-axis: Phase
X-axis: Log Hz

2.3 Power spectrum [POWER]

- The energy spectrum of the input signal
- Consists of only magnitude information.
- Major applications
  - Determining the peaks of waveform frequency components. (Because the differences of level appear to be larger than with a linear spectrum, this is suitable for finding the peaks.)
  - Determining the energy levels of high and low harmonics.

Function

$$Gaa = \frac{1}{2} \cdot Fa^* \cdot Fa$$
$$= \frac{1}{2} \cdot \{Re^2(Fa) + Im^2(Fa)\}$$
$$= \frac{1}{2} \cdot |Fa|^2$$

DC components:

$$Gaa = Fa^* \cdot Fa$$
$$= Re^2(Fa) + Im^2(Fa)$$
$$= |Fa|^2$$

Fa* is the complex conjugate of Fa.

Vertical axis	Meaning
Linear Mag (magnitude)	Gaa
Log Mag (logarithmic magnitude)	10 · log(Gaa)

Horizontal axis x-axis

Linear Hz — Indicates the frequency spectrum in linear units. (Range: from DC to the frequency range value)

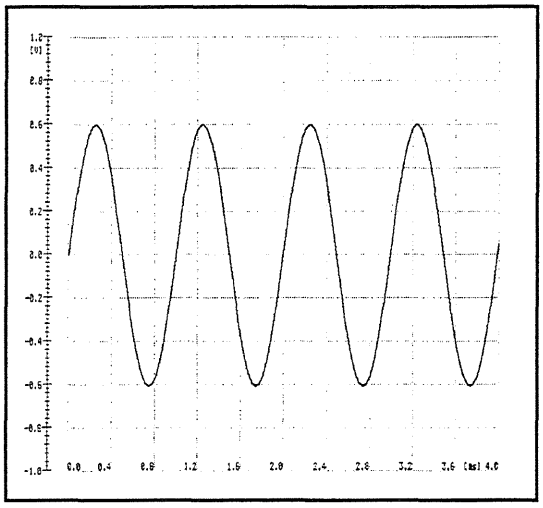
Log Hz — Indicates the frequency spectrum in logarithmic units. (Range: from 1/400 the frequency range value to the frequency range value)

Vertical axis y-axis

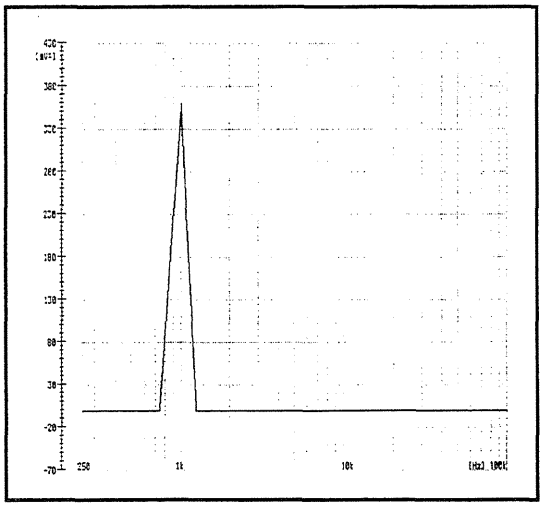
Linear Mag
- Indicates the binary exponential value of the data in voltage units (voltage)^2 (V^2).
- This indicates the energy component of the signal.

Log Mag
- Indicates the data in logarithmic units (dB). (0 dB reference value: 1 V^{2}peak, 1 V^{2}rms)
- This indicates the energy component of the signal.

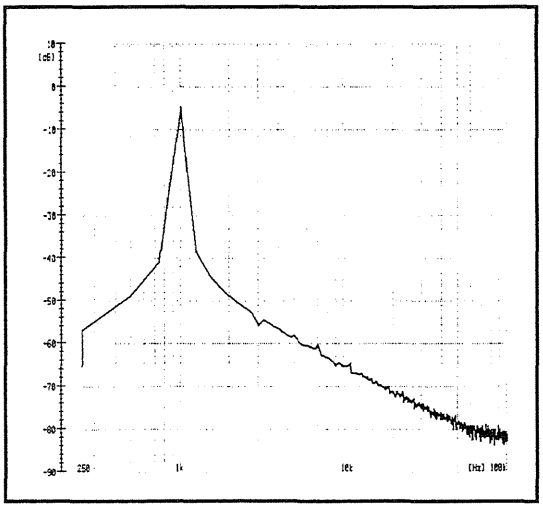
Example



Storage waveform



Y-axis: Linear Mag
X-axis: Log Hz



Y-axis: Log Mag
X-axis: Log Hz

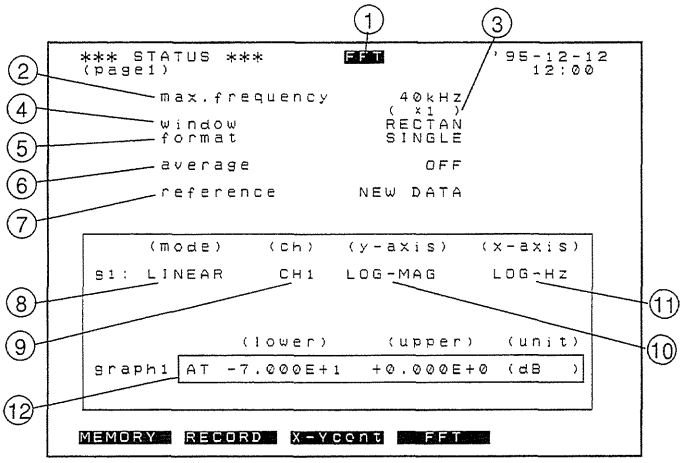
Chapter 3
Display Screens

3.1 Display Screens

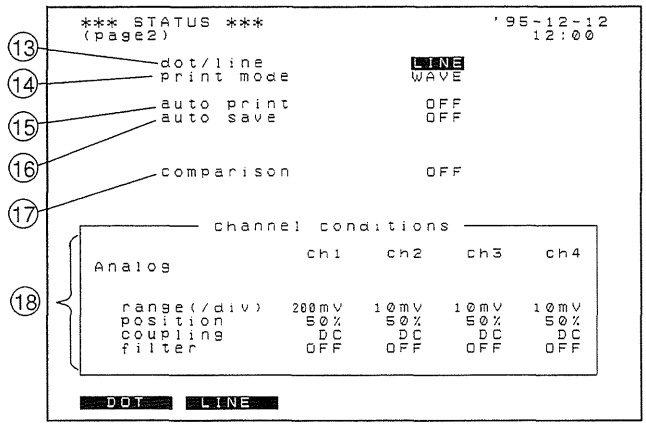
- This section describes the status, trigger and display screens.
- The system screen is described in Chapter 12 of the 8853 Instruction Manual, and the floppy disk and SCSI control screens are described in Chapter 13 of the 8853 Instruction Manual.

3.2 Status Screen

- Press the [STATUS] key, and the status screen appears.
- Pressing the [STATUS] key alternately switches between page 1 and page 2.
- On page 1, the settings related to FFT analyses are made.
- On page 2, the settings of interpolation, printer, waveform decision, channels, etc. are made.



page 1



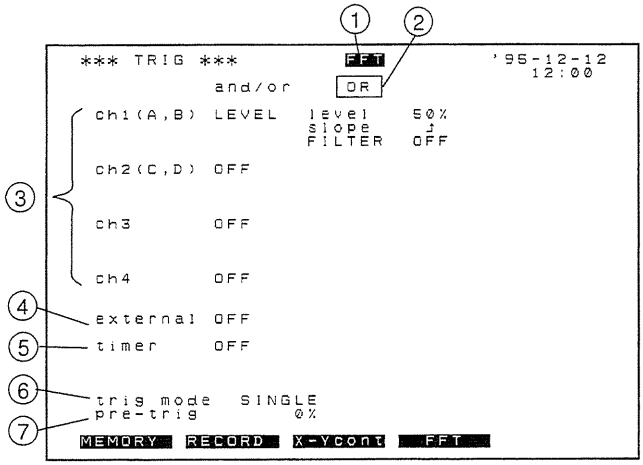
page 2

① Function mode
② Frequency range
③ Intermittent compression
④ Window
⑤ Format
⑥ Averaging
⑦ Reference data
⑧ Analysis mode
⑨ Analysis channel
⑩ Y-axis display
⑪ X-axis display
⑫ Display scale
⑬ Interpolation function
⑭ Printing mode
⑮ Auto-print
⑯ Auto-save
⑰ Waveform decision
⑱ Channel settings

	Selections		Explanation	Ref. Section
① Function mode	MEM, REC, X-Y, FFT		Selects the function mode.	4.2
② Frequency range	4 Hz to 5 MHz		Sets the frequency range.	4.3
③ Intermittent compression	×1, ×1/2, ×1/5		Sets the intermittent compression.	4.4
④ Window	RECTAN, HANNING		Sets the window function.	4.5
⑤ Format	SINGLE, DUAL		Sets the format type for display and recording.	4.6
⑥ Averaging	OFF, 2 to 256		Sets the averaging interval.	4.7
⑦ Reference data	NEWDATA, MEMDATA		Selects the data for FFT analysis	4.8
⑧ Analysis mode	STORAGE, LINEAR, POWER		Selects the FFT analysis mode.	4.9
⑨ Analysis channel	CH1 to CH4		Selects the channel for FFT analysis.	4.10
⑩ Y-axis display	LIN-REAL, LIN-IMAG, LIN-MAG, LOG-MAG, PHASE, (volt)		Sets the Y-axis (vertical axis).	4.11
⑪ X-axis display	LIN-Hz, LOG-Hz, (time)		Sets the X-axis (horizontal axis).	4.12
⑫ Display scale	AUTO, MANUAL (-9.999E+9 to +9.999E+9)		Selects the method of setting the display scale. With the manual setting, sets the upper and lower limits.	4.13
⑬ Interpolation function	DOT, LINE		Sets whether linear interpolation is performed or not.	4.15
⑭ Printing mode	WAVE, DATA		Sets the printing mode.	4.20
⑮ Auto-print	OFF, ON		Sets whether or not to automatically print out after calculation is performed.	4.20
⑯ Auto-save	OFF, FD, SCSI		Sets whether or not to automatically save data after calculation is performed.	4.21
⑰ Waveform decision	OFF, OUT, ALL-OUT		Sets the waveform decision mode.	4.22
⑱ Channel settings	range (/div)	10, 20, 50, 100, 200, 500 m, 1, 2, 5, 10, 20, 50 V	Sets the voltage axis range, position, input coupling, and low-pass filter.	4.16
	position	-100 % to 100 % (1% step)		
	coupling	GND, AC, DC		
	filter	OFF, 500 k, 500, 5 Hz		

3.3 Trigger Screen

- Press the TRIG key, and the trigger screen appears.
- For trigger settings, see Chapter 8 "Trigger Functions" of the 8853 Instruction Manual.

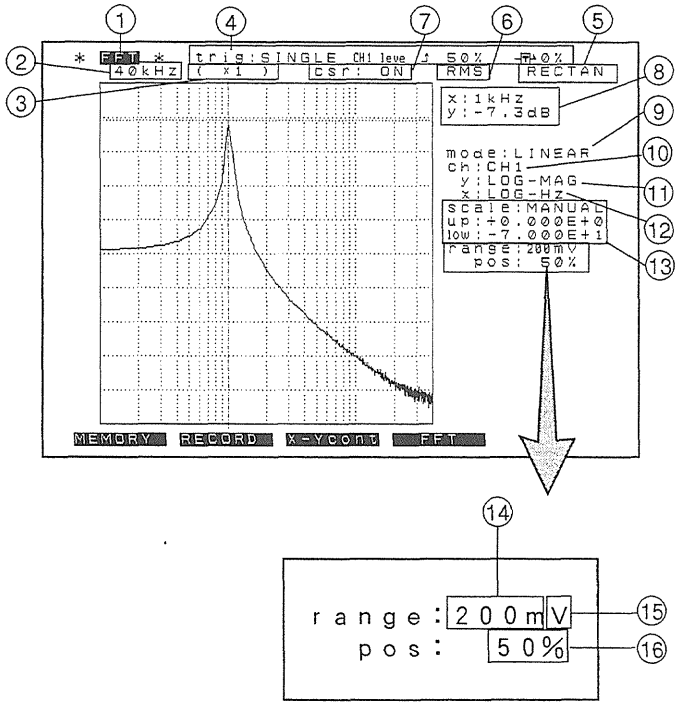


① Function mode
② Trigger logical operator
③ Internal triggers
④ External trigger
⑤ Timer trigger
⑥ Trigger mode
⑦ Pre-trigger

	Selections	Explanation	Ref. Section
① Function mode	MEM, REC, X-Y, FFT	Selects the function mode.	4.2
② Trigger logical operator	OR, AND	Sets the internal, external and timer trigger logical operator.	Chapter 8 of the 8853 Instruction Manual
③ Internal triggers	OFF, LEVEL, WINDOW, LOGIC, GLITCH, TIMEOUT	Selects the type of the internal trigger.	
④ External trigger	OFF, ON	Sets the external trigger.	
⑤ Timer trigger	OFF, ON	Sets the timer trigger.	
⑥ Trigger mode	SINGLE, REPEAT, AUTO	Sets the mode in which the trigger is activated.	
⑦ Pre-trigger	-950 % to 100 %	Sets the recorded range before the trigger.	

3.4 Display Screen

Press the DISP key, and the display screen appears.



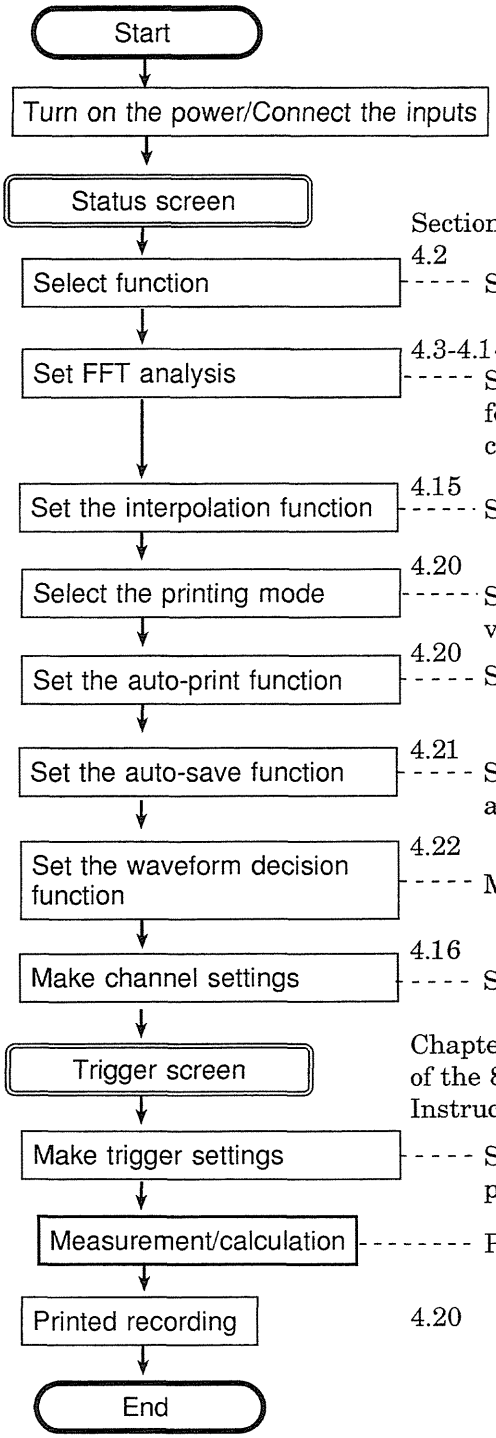
① Function mode
② Frequency range
③ Intermittent compression
④ Settings related to the trigger
⑤ Window
⑥ Display value setting
⑦ Trace cursor setting
⑧ Trace cursor readout values
⑨ Analysis mode
⑩ Analysis channel
⑪ Y-axis display
⑫ X-axis display
⑬ Display scale
⑭ Voltage axis range
⑮ Input coupling
⑯ Position

	Selections	Explanation	Ref. Section
① Function mode	MEM, REC, X-Y, FFT	Selects the function mode.	4.2
② Frequency range	4 Hz to 5 MHz	Sets the frequency range.	4.3
③ Intermittent compression	×1, ×1/2, ×1/5	Sets the intermittent compression.	4.4
④ Settings related to the trigger	SINGLE, REPEAT, AUTO CH1 to CH4 ↑, ↓ 0 % to 100 % -950 % to 100 %	Sets the trigger mode, trigger channel, trigger slope, trigger level, and pre-trigger.	Chapter 8 of the 8853 Instruction Manual
⑤ Window	RECTAN, HANNING	Sets the window function.	4.5
⑥ Display value setting	PEAK, RMS	Sets the method of displaying the voltage value.	4.14
⑦ Trace cursor setting	OFF, ON	Enables or disables the trace cursor.	4.19
⑧ Trace cursor readout values		Displays the trace cursor readout values.	4.19
⑨ Analysis mode	STORAGE, LINEAR, POWER	Selects the FFT analysis mode.	4.9
⑩ Analysis channel	CH1 to CH4	Selects the channel for FFT analysis.	4.10
⑪ Y-axis display	LIN-REAL, LIN-IMAG, LIN-MAG, LOG-MAG, PHASE, (volt)	Sets the Y-axis (vertical axis).	4.11
⑫ X-axis display	LIN-Hz, LOG-Hz, (time)	Sets the X-axis (horizontal axis).	4.12
⑬ Display scale	AUTO, MANUAL (-9.999E+9 to +9.999E+9)	Selects the method of setting the display scale. With the manual setting, sets the upper and lower limits.	4.13
⑭ Voltage axis range	10 mV to 50 V	Sets the voltage axis range.	4.16
⑮ Input coupling	⏚: GND, ~: AC, ⎓: DC	Sets the input coupling.	4.16
⑯ Position	-100 % to 100 %	Sets the position.	4.16

Chapter 4
User Operations

4.1 Flow of User Operations

The flowchart on the next page illustrates the sequence of operations involved in using the FFT function. (See Section 4.23 for Operation Example.)



Section

4.2 ----- Select the FFT function.

4.3-4.14 ----- Set the frequency range, intermittent compression, window, format, averaging, reference data, analysis mode, analysis channel, x-axis and y-axis, display scale, and display value.

4.15 ----- Set dot or line display.

4.20 ----- Set the mode for outputting as a waveform or as numerical values.

4.20 ----- Set whether or not to print automatically.

4.21 ----- Select whether or not to save data to external memory automatically.

4.22 ----- Make settings for deciding the FFT analysis result.

4.16 ----- Set the voltage axis range, position, input coupling and filter.

Chapter 8 of the 8853 Instruction Manual

----- Set the trigger types, trigger conditions, trigger modes, and the pre-trigger.

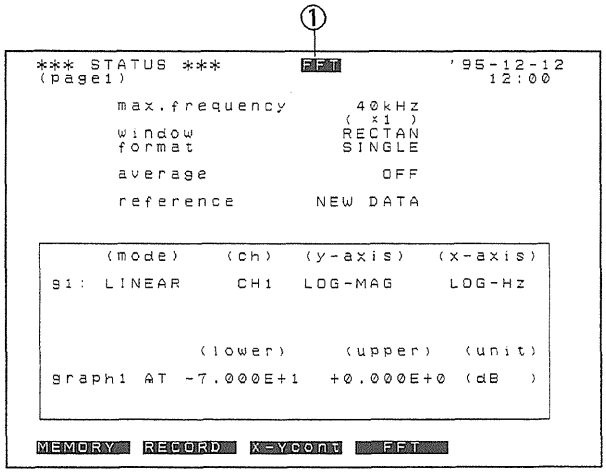
-------- Press the START key to start measurement and calculation.

4.20

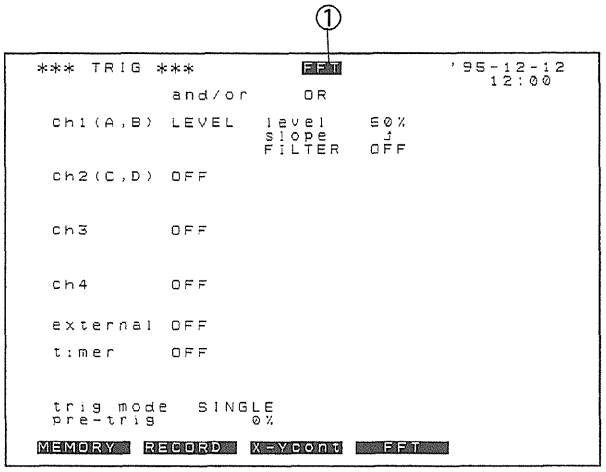
4.2 Function Selection

- The 8853 has four function: memory recorder (Chapter 5 of the 8853 Instruction Manual), recorder (Chapter 6 of the 8853 Instruction Manual), X-Y recorder (Chapter 7 of the 8853 Instruction Manual), and FFT.
- In this case, select the FFT function.

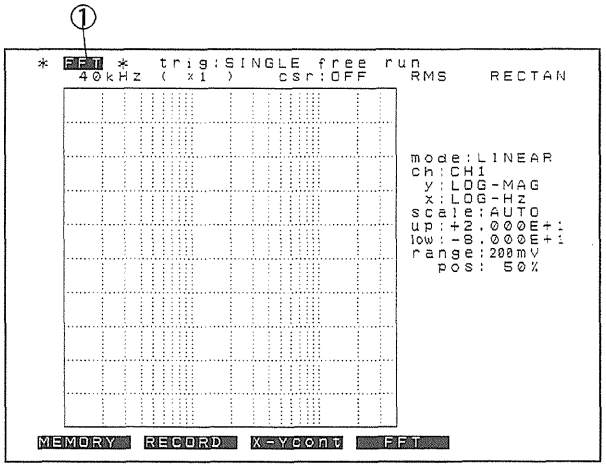
Procedure (Setting screen: status (page 1), trigger and display)



Status screen (page 1)



Trigger screen



Display screen

1. Using the cursor keys, move the flashing cursor to ① as shown in the figure on the left.
2. Press the **FFT** soft key.
   This selects the FFT function.

Soft key indication

MEMORY : Memory recorder function
RECORD : Recorder function
X-Ycont : X-Y recorder function
FFT : FFT function

NOTE

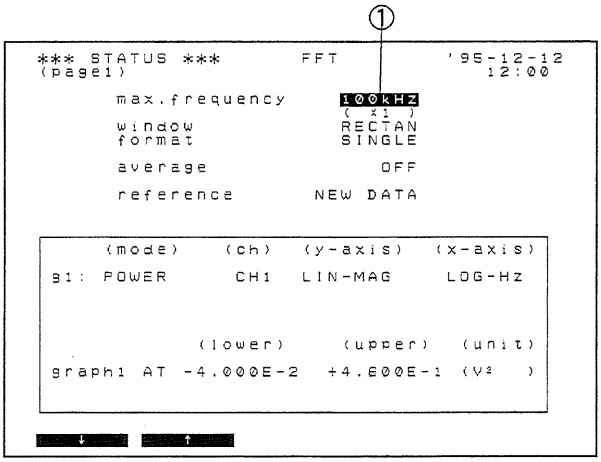
The function indication in ① on the display screen is abbreviated:

MEM : MEMORY
REC : RECORDER
XYc : X-Ycont

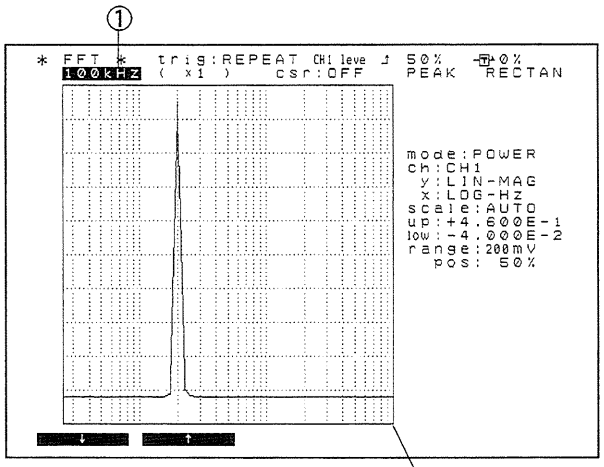
4.3 Setting the Frequency Range (max. frequency)

- Set FFT for the frequency range of the waveform to be captured.
- The frequency range corresponds to the time axis range of the memory recorder function.
  (The frequency range correspondence table is available on the next page.)

Procedure (Setting screen: status (page 1) and display)



Status screen (page 1)



Display screen

This is the maximum frequency value.

1. Move the flashing cursor to the "max. frequency" item, ① as shown in the figure on the left.
2. By using the soft keys or the rotary knob, set the frequency range.

[4 Hz, 10 Hz, 20 Hz, 40 Hz, 100 Hz, 200 Hz, 400 Hz, 1 kHz, 2 kHz, 4 kHz, 10 kHz, 20 kHz, 40 kHz, 100 kHz, 200 kHz, 400 kHz, 1 MHz, 2 MHz, 4 MHz, 5 MHz]

By using the (▼TIME/DIV▲) key, you can set the frequency range without moving the flashing cursor.

NOTE

If using the rotary knob, set the rotary knob operation to "Altering numerical values (VALUE), (Two LEDs light.). See Section 4.2, "Operation Keys" of the 8853 Instruction Manual.

Time axis range, window width, and frequency resolution corresponding to each frequency range.

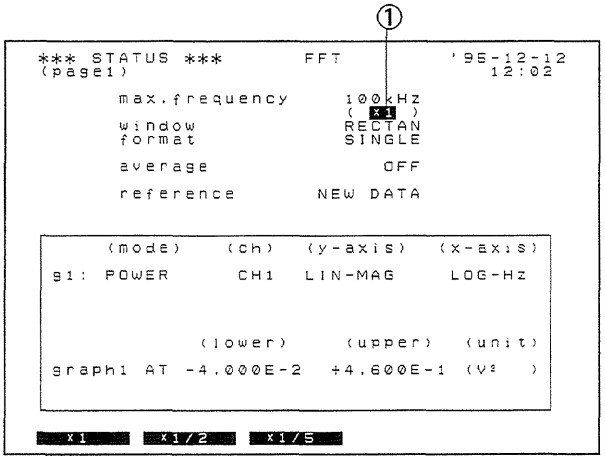
Frequency range	Time axis range in the STORAGE analysis mode	Time axis range in the memory recorder function	Window width	Frequency resolution
5 MHz	8 μs	4 μs	80 μs	12.5 kHz
4 MHz	10 μs	5 μs	100 μs	10 kHz
2 MHz	20 μs	10 μs	200 μs	5 kHz
1 MHz	40 μs	20 μs	400 μs	2.5 kHz
400 kHz	100 μs	50 μs	1 ms	1 kHz
200 kHz	200 μs	100 μs	2 ms	500 Hz
100 kHz	400 μs	200 μs	4 ms	250 Hz
40 kHz	1 ms	500 μs	10 ms	100 Hz
20 kHz	2 ms	1 ms	20 ms	50 Hz
10 kHz	4 ms	2 ms	40 ms	25 Hz
4 kHz	10 ms	5 ms	100 ms	10 Hz
2 kHz	20 ms	10 ms	200 ms	5 Hz
1 kHz	40 ms	20 ms	400 ms	2.5 Hz
400 Hz	100 ms	50 ms	1 s	1 Hz
200 Hz	200 ms	100 ms	2 s	500 mHz
100 Hz	400 ms	200 ms	4 s	250 mHz
40 Hz	1 s	500 ms	10 s	100 mHz
20 Hz	2 s	1 s	20 s	50 mHz
10 Hz	4 s	2 s	40 s	25 mHz
4 Hz	10 s	5 s	100 s	10 mHz

In the STORAGE analysis mode (see Section 4.9), 20 divisions of a waveform is compressed to 10 divisions, so the time axis range is twice that for the memory recorder function.

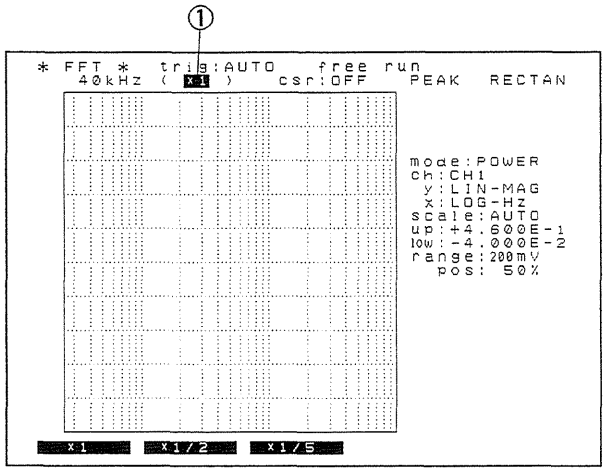
4.4 Intermittent Compression

- Normally, FFT analysis is performed for 20 divisions of waveform data (800 points of the sampled data). (40 sampling per one division)
- By taking the sampled data at intervals, FFT analysis can be performed for 40 or 100 divisions of waveform data.
- This is available, when the reference data (see Section 4.8) is MEM DATA.

Procedure (Setting screen: status (page 1) and display)



Status screen (page 1)



Display screen

1. Move the flashing cursor to ① as shown in the figure on the left.
2. Set the intermittent compression ratio by using the soft keys.

Soft key indication

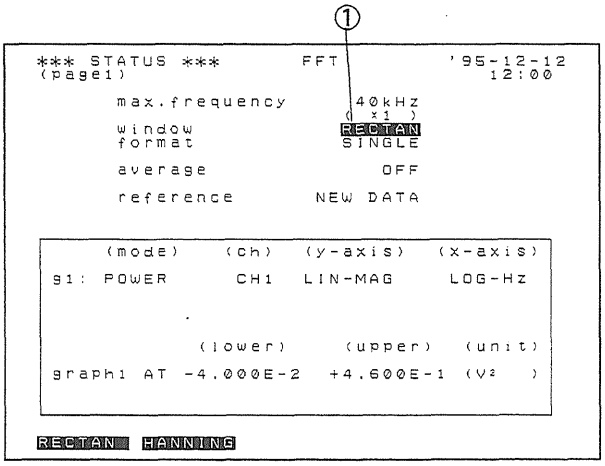
X1 : Normal
X1/2 : 1/2 compression
X1/5 : 1/5 compression

- Normal
  FFT analysis is performed for 800 points of the sampled data in order. (20 divisions of waveform data) (No intermittent compression)
- 1/2 compression
  FFT analysis is performed for 800 points of the sampled data, taken one every other data point. (40 divisions of waveform data)
- 1/5 compression
  FFT analysis is performed for 800 points of the sampled data, taken one every five data points. (100 points of waveform data)

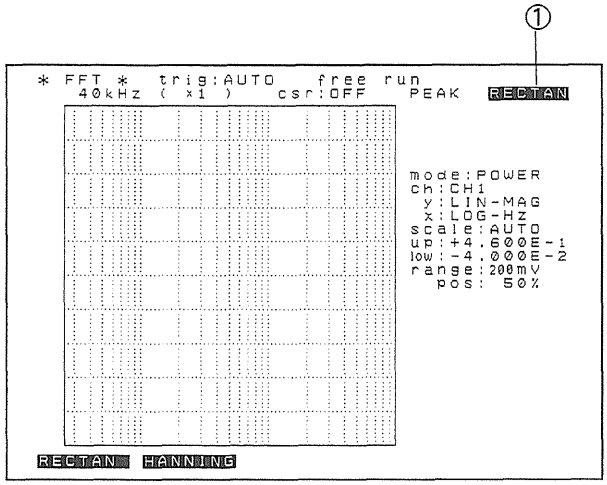
4.5 Setting the Window Function (window)

- The window function defines the segment of the input signal that will be processed.
- Window processing can be used to minimize leakage error (see Section 1.5).

Procedure (Setting screen: status (page 1) and display)



Status screen (page 1)



Display screen

1. Move the flashing cursor to the "window" item, ① as shown in the figure on the left.
2. Select the window function by using the soft keys.
   [RECTAN, HANNING]

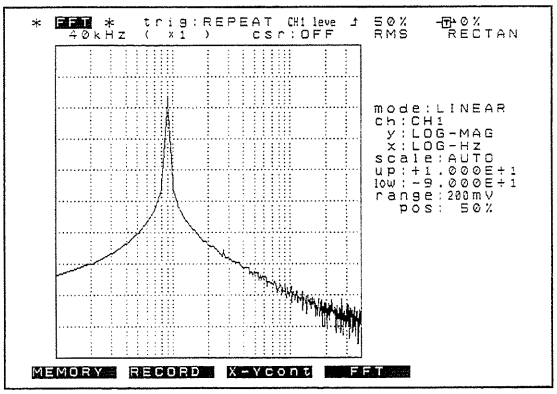
Soft key indication

RECTAN : Rectangular function
Effective on discrete waveform.

HANNING : Hanning function
Effective on continuous waveform.

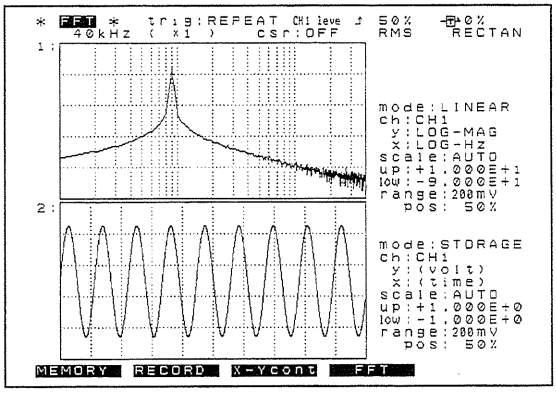
4.6 Format Selection (format)

· Select the format for displaying the FFT analysis result.
· There are two possibilities: SINGLE and DUAL.



SINGLE format

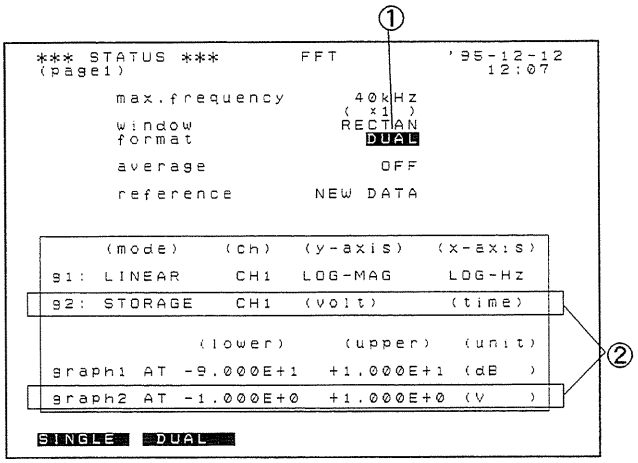
Displays the analysis result in a single graph.



DUAL format

Displays the analysis result in two graphs.

Procedure (Setting screen: status)

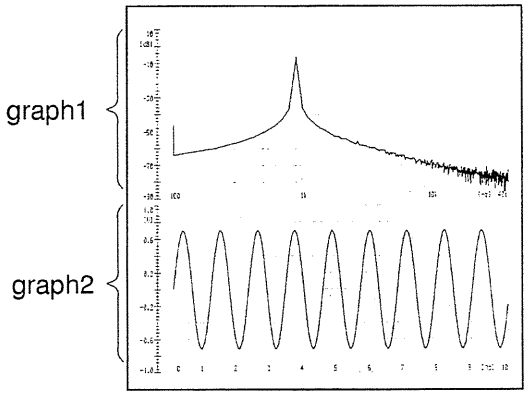


Status screen (page 1)

1. Move the flashing cursor to the "format" item, ① as shown in the figure on the left.
2. Select the format by using the soft keys.

Soft key indication

SINGLE : Single format
DUAL : Dual format



DUAL format

Displays the analysis result in two graphs.

· If selecting the DUAL format, the items for graph 2, ② as shown in the figure on the left are displayed.
· Outputs on the printer in the same format as on the screen.

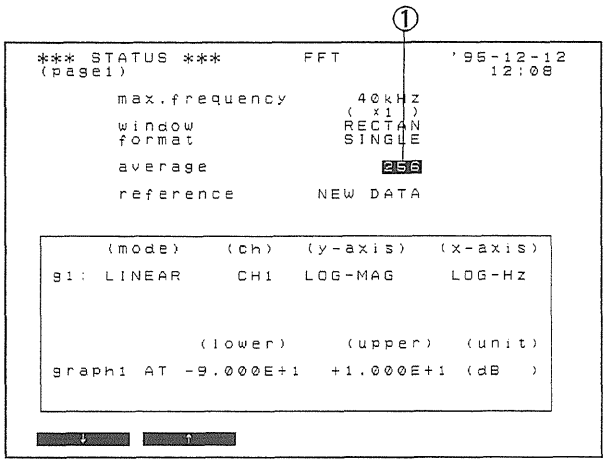
4.7 Averaging Function (average)

- Noise components can be removed by using averaging.
- With periodic waveform signals, it is possible to isolate the significant signal when the input signal contains much random noise. It is also possible to increase the reliability of unstable phenomena.
- A variety of types of averaging according to the analysis modes (see Section 4.9) is as follows.

Analysis mode	Averaging
Linear spectrum	Time domain additive averaging *
Power spectrum	Frequency domain power spectrum additive averaging
Storage waveform	Only the most recently captured waveform is displayed. Averaging is not performed.

* With time axis averaging, addition is performed in sync with the trigger. This type of averaging is meaningless unless sync is obtained.
Ensure that the trigger mode is either SINGLE or REPEAT.

Procedure (Setting screen: status (page 1))



Status screen (page 1)

1. Move the flashing cursor to the "average" item, ① as shown in the figure on the left.
2. By using the soft keys or the rotary knob, set the number of times for averaging.
   [OFF, 2, 4, 8, 16, 32, 64, 128, 256]

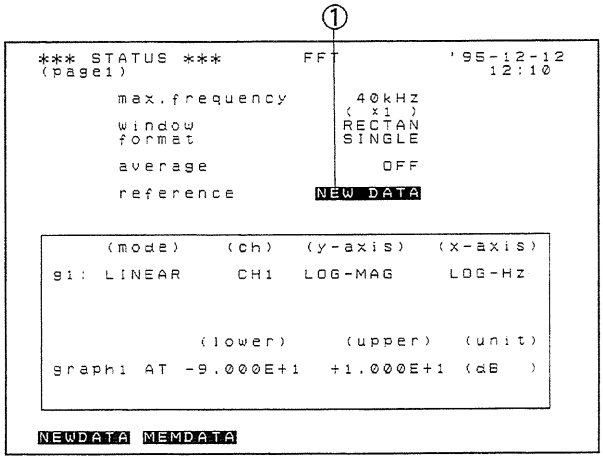
NOTE

- When using the averaging function, only the data for the analysis channel (see Section 4.10) are held. If switching to the channel not to have been analyzed after measurement, no data are held.
- When the reference data (see Section 4.8) is MEM DATA, the averaging function cannot be used.

4.8 Setting the Reference Data (reference)

Select the data to be used for FFT analysis.

Procedure (Setting screen: status (page 1))



Status screen (page 1)

1. Move the flashing cursor to the "reference" item, ① as shown in the figure on the left.
2. Select the desired reference data by using the soft keys.

Soft key indication

NEWDATA : 800 points of waveform data are captured with the FFT function, and FFT analysis is performed.

MEMDATA : FFT analysis is performed using the first 800 points (20 divisions) of waveform data captured using the memory recorder function.

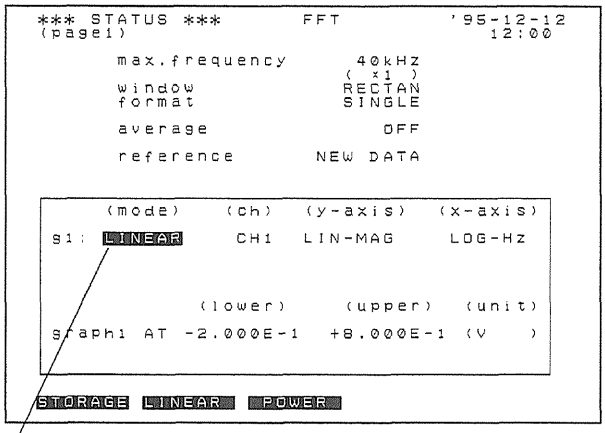
When MEM DATA is selected

- If the waveform data in the memory recorder function is less than 800 points (20 divisions), the remainder is filled by 0 V.
- When using the A and B cursors (as vertical or trace cursors) in the memory recorder function, FFT analysis is performed using the 800 points of data that follow whichever of the two cursors is first.
- FFT analysis can be performed using the 40 or 100 divisions of data, by using the intermittent compression (see Section 4.4).
- The averaging function (see Section 4.7) cannot be used.

4.9 Setting the Analysis Mode (mode)

Select the FFT analysis mode.

Procedure (Setting screen: status (page 1) and display)



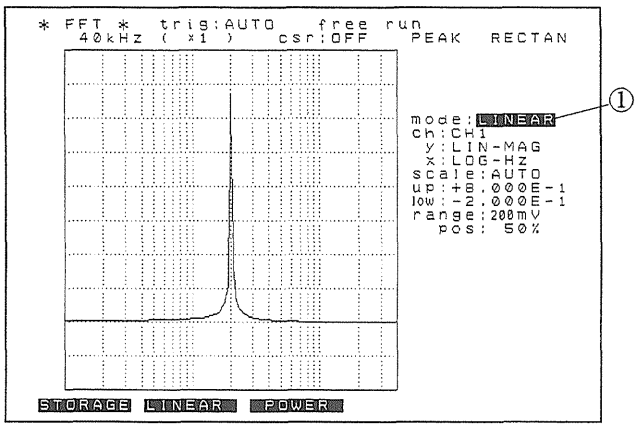
Status screen (page 1)

1. Move the flashing cursor to the "(mode)" item, ① as shown in the figure on the left.
2. Select the analysis mode by using the soft keys.

Soft key indication

STORAGE : Storage waveform
LINEAR : Linear spectrum
POWER : Power spectrum

For the detailed explanation of each analysis mode, refer to Chapter 2.



Display screen

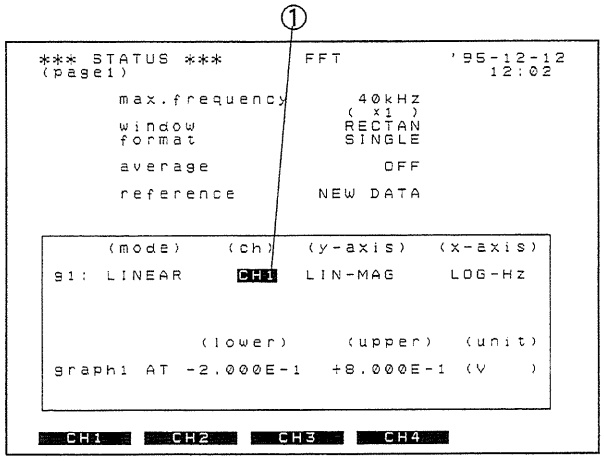
NOTE

- When scaling (see Section 12.3 of the 8853 Instruction Manual) is being performed, FFT analysis is performed using the values after scaling.
- In DUAL format, the setting items for graph 2 are also displayed. (Set in the same manner as graph 1.)

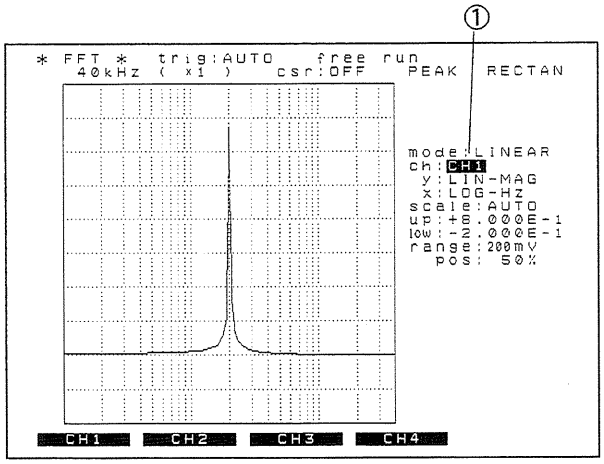
4.10 Setting the Analysis Channel (ch)

Select the channel for FFT analysis.

Procedure (Setting screen: status (page 1) and display)



Status screen (page 1)



Display screen

1. Move the flashing cursor to the "(ch)" item, ① as shown in the figure on the left.
2. Select the analysis channel by using the soft keys.

Soft key indication

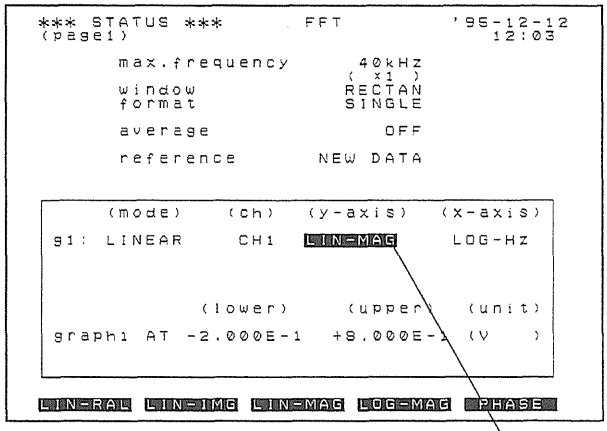
CH1 : Channel 1
CH2 : Channel 2
CH3 : Channel 3
CH4 : Channel 4

NOTE In DUAL format, the setting items for graph 2 are also displayed. (Set in the same manner as graph 1.)

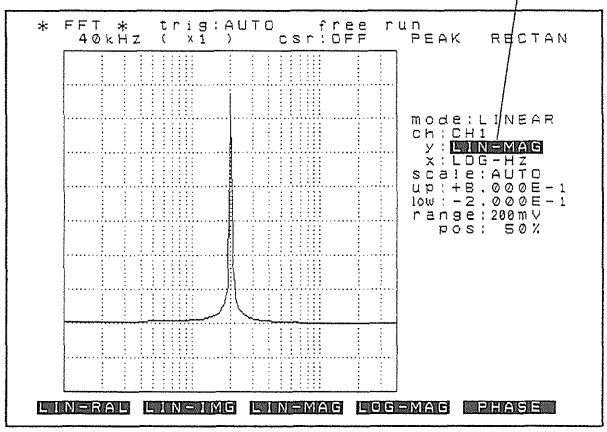
4.11 Setting the Y-axis (y-axis)

Set the y-axis (vertical axis) for display of FFT analysis results.

Procedure (Setting screen: status (page 1) and display)



Status screen (page 1)



Display screen

1. Move the flashing cursor to the "y (y-axis)" item, ① as shown in the figure on the left.
2. Select the y-axis display by using the soft keys.

Soft key indication

* **LIN-RAL** : Linear-Real
Indicates the real-number part of the data in voltage units.

* **LIN-IMG** : Linear-Imag
Indicates the imaginary-number part of the data in voltage units.

LIN-MAG : Linear-Mag
Indicates the data values in voltage units.

LOG-MAG : Log-Mag
Indicates the data in decibels.

* **PHASE** : Phase
Indicates the phase information in degrees.

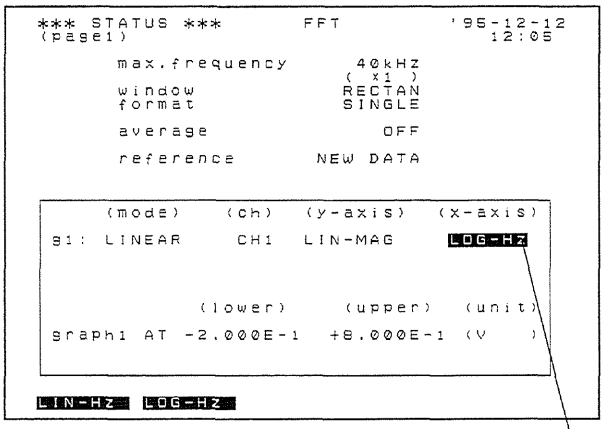
NOTE

- When the analysis mode is POWER (power spectrum), the soft key indications with the "*" marks above do not appear.
- When the analysis mode is STORAGE (storage waveform), the y-axis is fixed for "(volt)", and the flashing cursor skips the "y-axis" item.
- In DUAL format, the setting items for graph 2 are also displayed. (Set in the same manner as graph 1.)

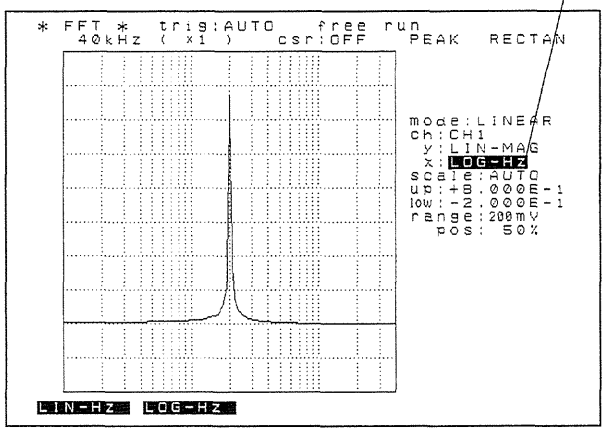
4.12 Setting the X-axis (x-axis)

Set the x-axis (horizontal axis) for display of FFT analysis results.

Procedure (Setting screen: status (page 1) and display)



Status screen (page 1)



Display screen

1. Move the flashing cursor to the "x (x-axis)" item, ① as shown in the figure on the left.
2. Select the x-axis display by using the soft keys.

Soft key indication

LIN-Hz : Linear-Hz
Indicates the frequency spectrum in linear units.

LOG-Hz : Log-Hz
Indicates the frequency spectrum in logarithmic units.

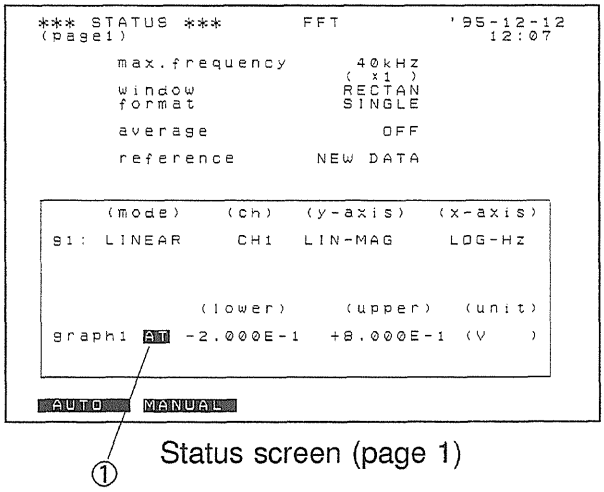
NOTE

- When the analysis mode is STORAGE (storage waveform), the x-axis is fixed for "(time)", and the flashing cursor skips the "x-axis" item.
- In DUAL format, the setting items for graph 2 are also displayed. (Set in the same manner as graph 1.)

4.13 Setting the Display Scale (scale)

- Sets the y-axis (vertical axis) scale (upper and lower limits) for displaying FFT analysis results.
- There are the following two methods of setting the display scale.
  ① AUTO: The upper and lower limits for the y-axis are set automatically.
  ② MANUAL: The upper and lower limits for the y-axis are set to any value.

Procedure (Setting screen: status (page 1) and display)



Status screen (page 1)

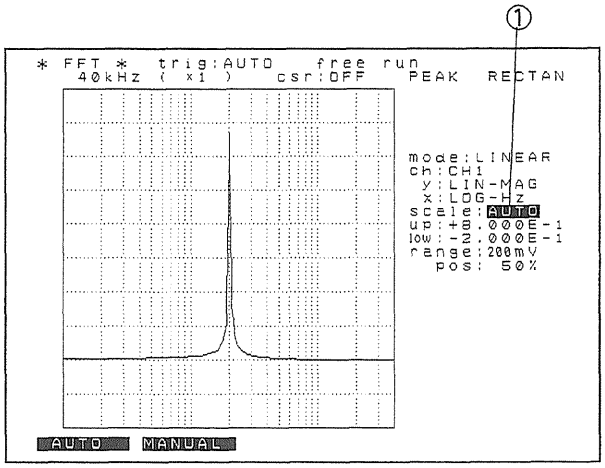
1. Move the flashing cursor to ① as shown in the figure on the left.
2. Select the method of setting the display scale by using the soft keys.

Soft key indication

AUTO : Scale set automatically

MANUAL : Scale set to any value (-9.999E+9 to +9.999E+9)

For the setting, refer to the next page.



Display screen

The function indication in ① on the status screen is abbreviated.

AT : AUTO
MN : MANUAL

NOTE

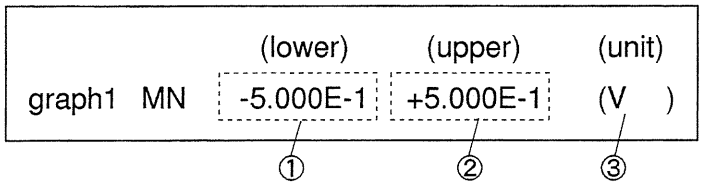
- In DUAL format, the setting items for graph 2 are also displayed. (Set in the same manner as graph 1.)
- If altering the upper and lower limits, the display scale is set to MANUAL automatically.

Setting the upper and lower limits

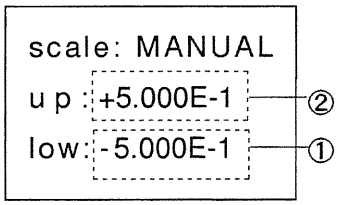
Setting range : -9.999E+9 to +9.999E+9.

Procedure (Setting screen: status (page 1) and display)

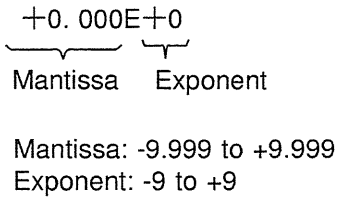
- Sets the lower limit in the "low (lower)" item, ① as shown in the figure below.
- Set the upper limit in the "up (upper)" item, ②.
- Move the flashing cursor to each individual digit, and set the upper and lower limits by using the soft keys or the rotary knob(VALUE).
- The unit is displayed in ③.



Status screen



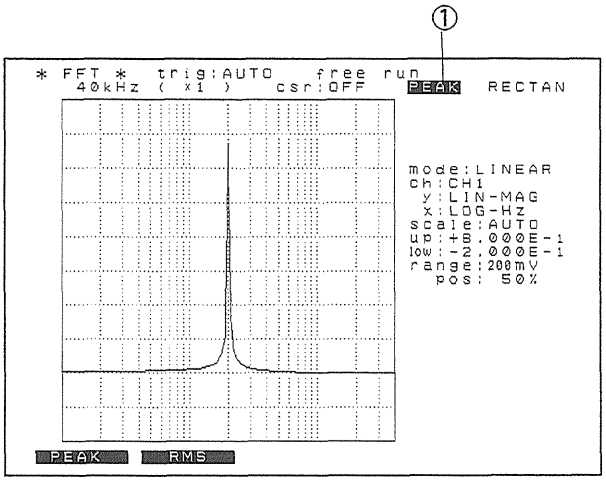
Display screen



4.14 Setting the Display Value (peak/rms)

- Set the method of displaying the voltage value (peak or effective value) of FFT analysis results.
- The setting is available, when the y-axis setting (see Section 4.11) is other than "Phase" in linear spectrum or power spectrum.

Procedure (Setting screen: display)



Display screen

1. Move the flashing cursor to ① as shown in the figure on the left.
2. Set the method of displaying the voltage value by using the soft keys.

Soft key indication

PEAK : Peak value

RMS : Effective value

Relationship between the decibel (dB) and voltage

The relationship between decibel (dB) and measurement voltage (V) in the FFT analysis is expressed in the following equations.

Linear spectrum

① When the display value is a peak value $$dB=20\cdot\log_{10}\frac{V}{1\ V\ peak}$$

② When the display value is an effective value $$dB=20\cdot\log_{10}\frac{V}{1\ V\ rms}$$

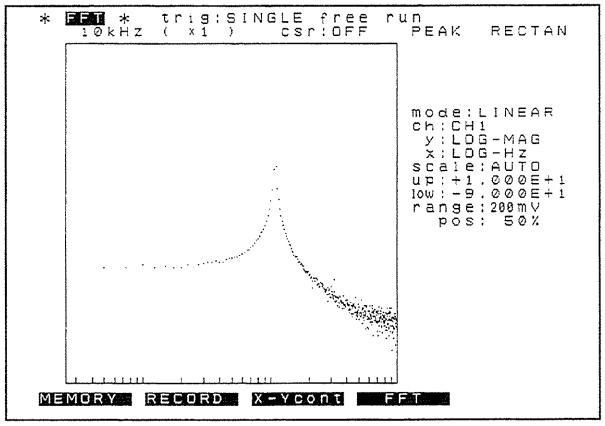
Power spectrum

① When the display value is a peak value $$dB=10\cdot\log_{10}\left(\frac{V}{1\ V\ peak}\right)^2$$

② When the display value is an effective value $$dB=10\cdot\log_{10}\left(\frac{V}{1\ V\ rms}\right)^2$$

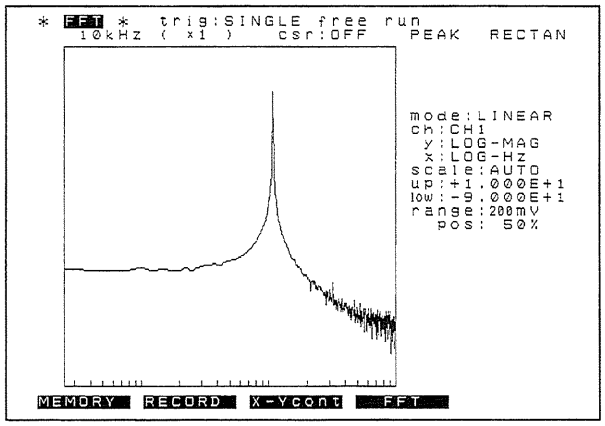
4.15 Setting the Interpolation Function (dot/line)

This function determines whether to display the FFT analysis results as detached points (dot mode) or with straight-line interpolation (line mode).



Dot mode

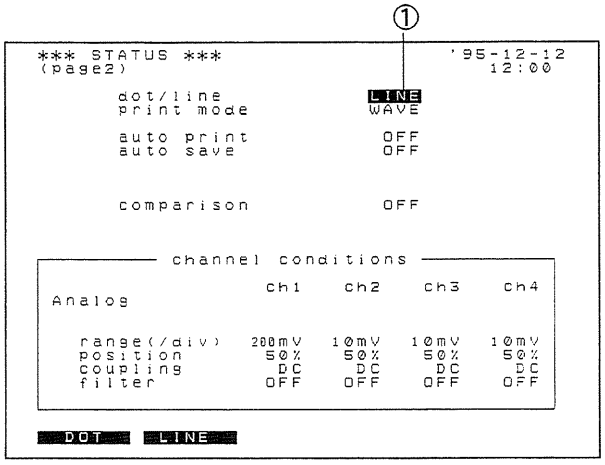
- No linear interpolation.
- The analysis results are displayed exactly as measured.



Line mode

- Linear interpolation.
- This gives a more readable display.

Procedure (Setting screen: status (page 2))



Status screen (page 2)

1. Move the flashing cursor to the "dot/line" item, ① as shown in the figure on the left.
2. Set the interpolation function by using the soft keys.

Soft key indication

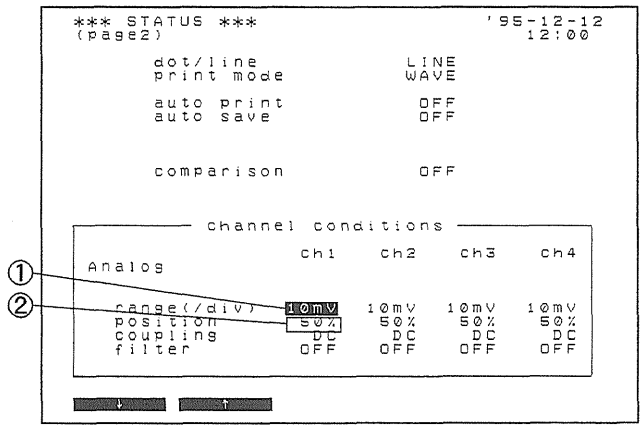
DOT : Dot mode

LINE : Line mode

4.16 Setting the Voltage Axis Range, Position, Input Coupling and Filter

- These settings determine the voltage axis range, position, input coupling, and filter for each channel.
- The settings are the same as in the memory recorder function. For details, see Section 5.3.9 of the 8853 Instruction Manual.

Procedure (Setting screen: status and display)



Status screen (page 2)

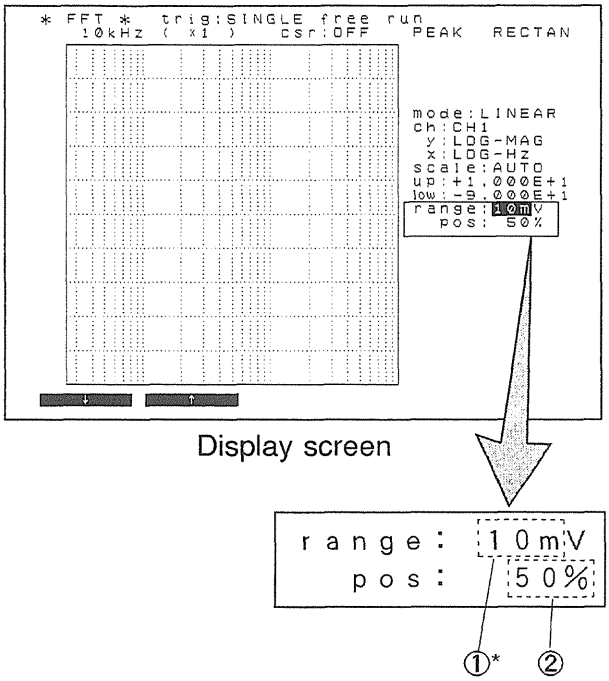
(1) Voltage axis range (range/div) setting

The voltage value for one division is set.

Move the flashing cursor to ① as shown in the figure on the left or below for each channel, and make the setting by using the soft keys or the rotary knob.

[10 mV, 20 mV, 50 mV, 100 mV, 200 mV, 500 mV, 1 V, 2 V, 5 V, 10 V, 20 V, 50 V]

You can also use the ▼CH1-CH4▲ key for each channel, without moving the flashing cursor.



Display screen

* "V" is omitted.

(2) Position setting

The position of the zero voltage is set.

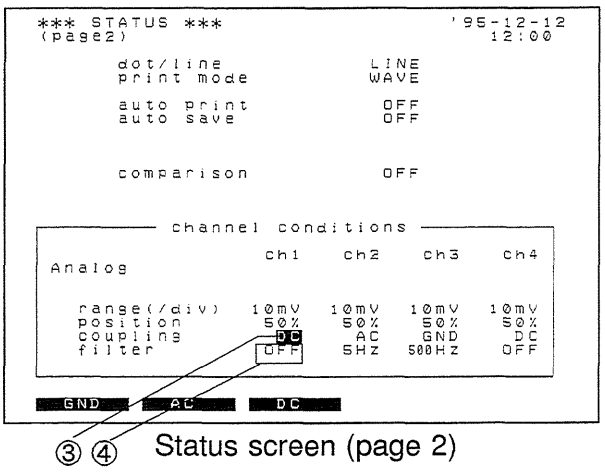
Move the flashing cursor to ② as shown in the figures on the left for each channel, and make the setting by using the soft keys or the rotary knob.

Soft key indication

↓ : 1 % decrement
↑ : 1 % increment
10↓ : 10 % decrement
10↑ : 10 % increment

[Range: -100 % to 100 % (in 1 % steps)]

Procedure (Setting screen: status and display)



(3) Input coupling setting

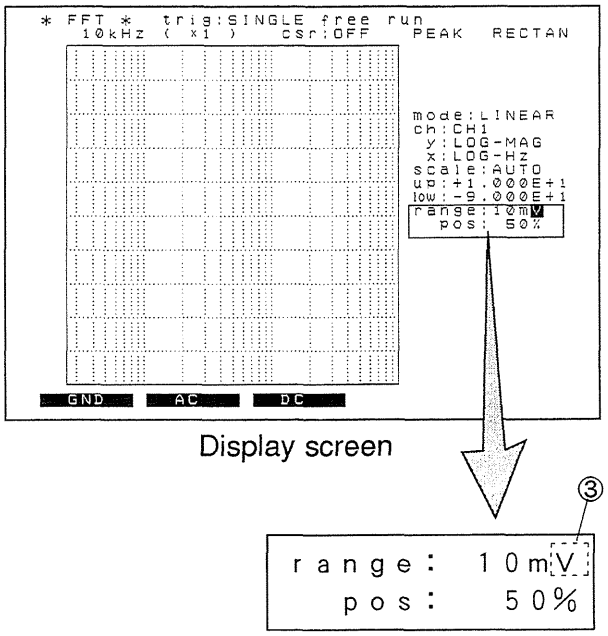
The method of coupling the input signal is set.

Move the flashing cursor to ③ as shown in the figures on the left for each channel, and make the setting by using the soft keys.

Soft key indication

GND	⏚	: GND coupling
AC	~	: AC coupling
DC	V	: DC coupling

↑
└ Indications of ③ on the display screen.



(4) Low-pass filter setting (On the status screen only)

The low-pass filter in the input unit itself is set.

Move the flashing cursor to ④ as shown in the figure on the left for each channel, and make the setting by using the soft keys.

Soft key indication

OFF	: OFF
500kHz	: Cutoff frequency approx. 500 kHz
500Hz	: Cutoff frequency approx. 500 Hz
5Hz	: Cutoff frequency approx. 5 Hz

The setting cannot be make on the display screen.

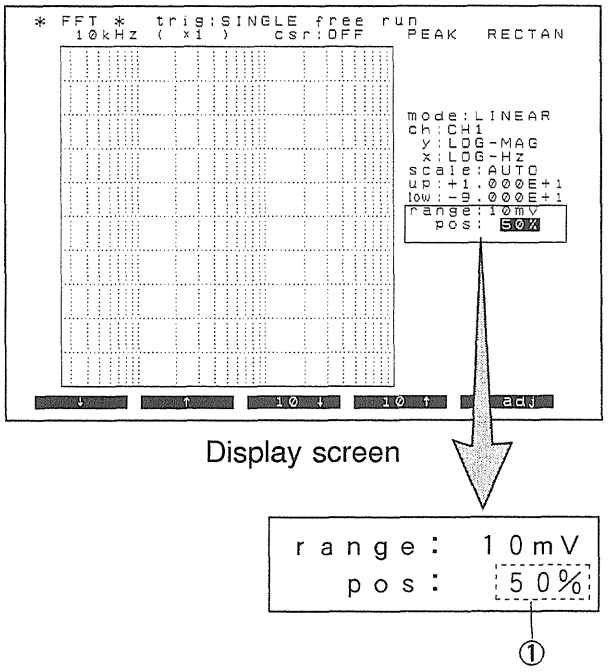
On the display screen, only the voltage axis range, position and input coupling for the analysis channel (see Section 4.11) are displayed.

4.17 Zero Adjustment

- This function provides for accurate adjustment of the waveform to the origin position when a zero voltage is input.
- Use it to ensure accurate results from FFT analysis.

Procedure (Setting screen: display)

Allow at least 60 minutes after powering on before carrying out this procedure, to ensure that the internal temperature of the input units has stabilized.



1. Move the flashing cursor to the position of ① as shown in the figure on the left. (The item for setting the position)
2. Press the **0 adj** soft key to carry out zero adjustment.

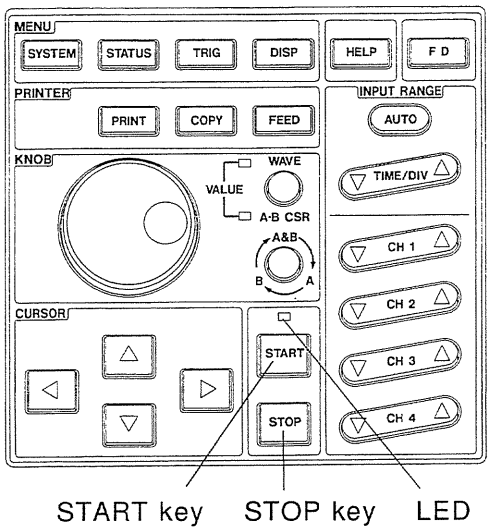
NOTE

- Zero adjustment is not possible while the unit is measuring.
- In DUAL format, zero adjustment can also be carried out in ① as shown in the figure above for graph 2.
- Perform zero adjustment after changing the input units.
- Perform zero adjustment also after carrying out a system reset (see Section 16.5 of the 8853 Instruction Manual) by powering on the unit while holding down the STOP key.

4.18 Starting and Stopping Measurement and Calculation Operation

- The START and STOP keys control the measurement and calculation operation mode of the unit.
- The LED above the START key is lit during measurement and calculation.
- When using the scaling function (see Section 12.3 of the 8853 Instruction Manual), the calculation is performed for the scaled values.

Procedure



1. Press the START key.
   Measurement and calculation start.
2. Press the STOP key.
   Measurement and calculation stop.

NOTE

- When the reference data is MEM DATA, measurement and calculation operation stops automatically after the calculation is carried out, regardless of the trigger mode (see Section 8.6 of the 8853 Instruction Manual).
- When the reference data is NEW DATA, measurement and calculation operation is as follows.

(1) The trigger modes, and starting and stopping

① When the trigger mode is SINGLE:

- When the trigger is activated, the unit captures the waveform of length equal to 800 points, and performs the calculation.
- After displaying the calculation results, it terminates measurement automatically.

② When the trigger mode is REPEAT:

- When the trigger is activated, the unit captures the waveform of length equal to 800 points, and performs the calculation.
- It remains in measurement operation mode, and if the trigger is activated again, repeats the waveform capture, each time performing the calculation.
- It continues measurement until the STOP key is pressed.

③ When the trigger mode is AUTO:

- When the trigger is activated, the unit captures the waveform of length equal to 800 points, and performs the calculation.
- Whether or not the trigger is activated, it captures the waveform of length equal to 800 points, and performs the calculation, after about one second.
- It captures the waveform repeatedly, performing the calculation until the STOP key is pressed.

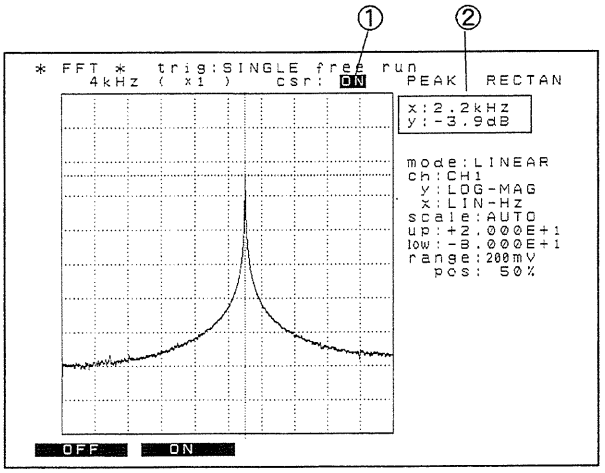
(2) Stopping

- Even if the STOP key is pressed, until capturing the waveform of length equal to 800 points, and displaying the calculation results, the unit continues measurement. (The auto-save and auto-print functions, however, are disabled.)
- At the point, pressing the STOP key second time abandons the measurement. (If the trigger mode is REPEAT or AUTO, the immediately previous calculation result is displayed.)
- Even if the STOP key is pressed, thereafter if the START key is pressed before measurement terminates, this applies a restart, and the unit starts measurement again.

4.19 Using the A and B Cursors (csr)

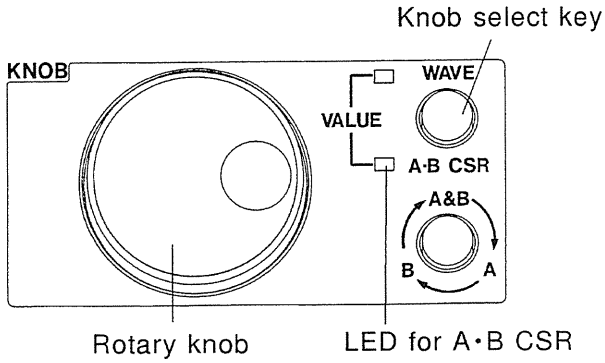
- In the FFT function, only one trace cursor can be used.
- As the trace cursor is moved, the trace point (the intersection of the waveform and the trace cursor) traces the waveform, and reads off the x-axis and y-axis values.
- When the scaling function (see Section 12.3 of the 8853 Instruction Manual) is being used, the scaled values are displayed.

Procedure (Setting screen: display)



Display screen

1. Move the flashing cursor to the "csr" item, ① as shown in the figure on the left.
2. Press the **ON** soft key.
   - A trace cursor appears.
   - A trace cursor disappears with the **OFF** soft key.
3. Press the knob select key so that only the LED for A·B CSR is lit.
4. The rotary knob controls the position of the cursor.
5. The values which are read off are shown in ②. [x: x-axis value, y: y-axis value]



NOTE

- In DUAL format, the cursor can only be used in graph 1. (It cannot be used in graph 2.)
- The values are read off as the value (peak or effective value) selected in "Setting the Display Value" (see Section 4.14).

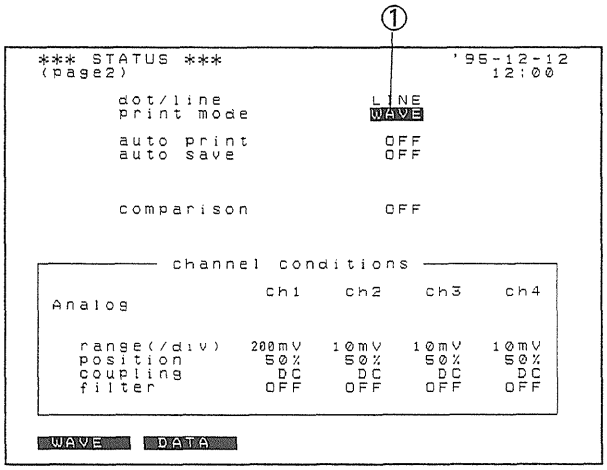
4.20 Recording on the Printer (print mode)

(1) Setting the print mode

There are the following two modes for outputting the analysis result on the printer.

① WAVE: Output as a waveform
② DATA: Output as numerical values

Procedure (Setting screen: status (page 2))



Status screen (page 2)

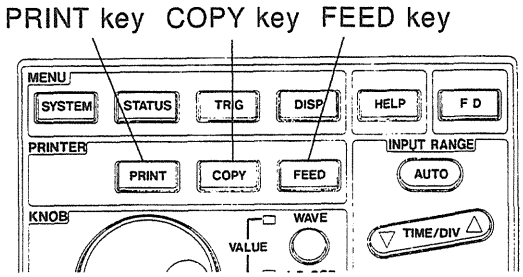
1. Move the flashing cursor to the "print mode" item, ① as shown in the figure on the left.
2. Select the print mode by using the soft keys.

Soft key indication

WAVE : Waveform
DATA : Numerical values

(2) Printing method

There are three printing methods: manual print, screen dump and auto-print.

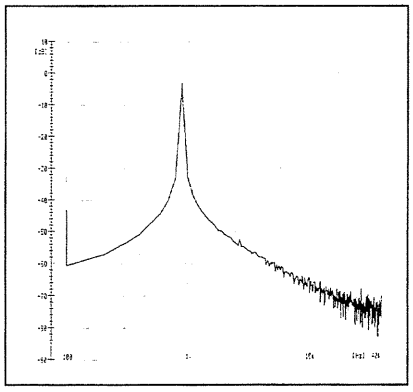


① Manual print

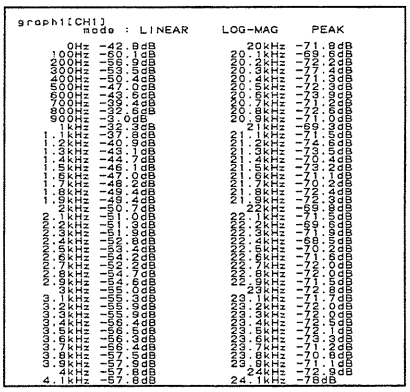
Prints out the analysis result as a waveform or numerical values.

Procedure (Setting screen: display)

After calculation is finished, press the [PRINT] key.



Waveform



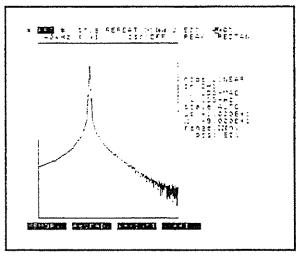
Numerical values

② Screen dump function

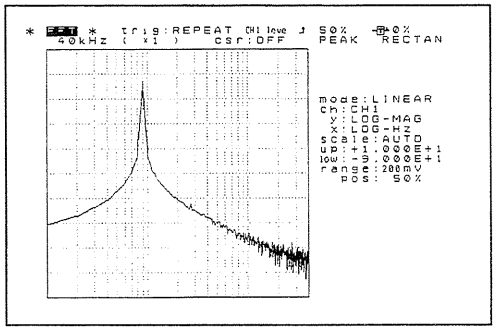
Prints out an exact copy of the screen.

Procedure

- Press the [COPY] key on the screen to be dumped.
- The screen dump output designation can be set on the system screen. (See Section 12.7 of the 8853 Instruction Manual.) (If the output designation is not set to the printer, press the COPY key again to output to the printer.)
- The copy size can be set on the system screen. (See Section 12.5.9 of the 8853 Instruction Manual.)



SMALL

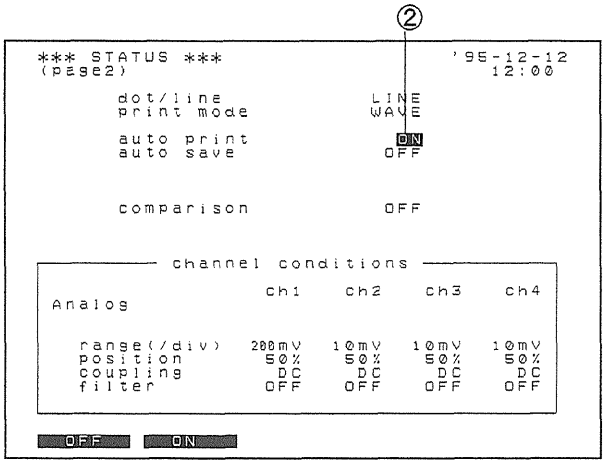


LARGE

③ Auto print

When calculation is finished, prints out the analysis result automatically.

Procedure (Setting screen: status (page 2))



Status screen (page 2)

1. Move the flashing cursor to the "auto print" item, ② as shown in the figure on the left.
2. Set auto print ON or OFF, by using the soft keys. [OFF, ON]

NOTE

- The recording paper is fed, while holding down the FEED key.
- On the system, status, or trigger screen, pressing the PRINT key produces a listing of settings (see Section 4.25).

4.21 External Memory (FD, HD, MO)

- It is possible to record the data on the floppy disk, hard disk or the magneto-optical disk.
- The auto save function (see the next page) enables the measurement data to be recorded automatically.
- For details, see Chapter 13 of the 8853 Instruction Manual.

What can be recorded and how much

① Setting state (FUNC)

Records the setting state.

Memory capacity — 720 K/1.2 M-byte floppy disk: 2 clusters (1 cluster is 1024 bytes)
1.44 M-byte floppy disk: 4 clusters (1 cluster is 512 bytes)

② Measurement data (WAVE)

Saves an FFT analyzed waveform.

Memory capacity (unit: cluster)

Format	Analysis mode	Floppy disk	
		720 K/1.2 M bytes	1.44 M bytes
SINGLE	STORAGE	14 (19)	28 (38)
	LINEAR	9 (14)	18 (28)
	POWER		
DUAL	STORAGE+STORAGE	26 (36)	52 (72)
	STORAGE+LINEAR	22 (31)	44 (62)
	STORAGE+POWER		
	LINEAR+LINEAR	17 (26)	34 (52)
	LINEAR+POWER		
	POWER+POWER		

( ): In case of averaging

Saves as the value (peak or effective value) selected in "Setting the Display Value" (see Section 4.14).

③ Waveform decision area (AREA)

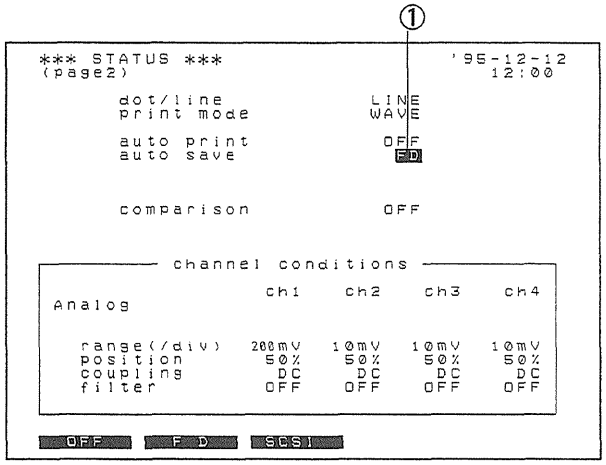
Saves a decision area for the waveform decision function.

Memory capacity — 720 K/1.2 M-byte floppy disk: 21 clusters (1 cluster is 1024 bytes)
1.44 M-byte floppy disk: 42 clusters (1 cluster is 512 bytes)

Auto save function

- When calculation is finished, saves the analysis result automatically.
- The file name is "#AUTO○○○. FFT". (○○○: Number from 001)

Procedure (Setting screen: status (page 2))



Status screen (page 2)

1. Move the flashing cursor to the "auto save" item, ① as shown in the figure on the left.
2. Set the save destination by using the soft keys.

Soft key indication

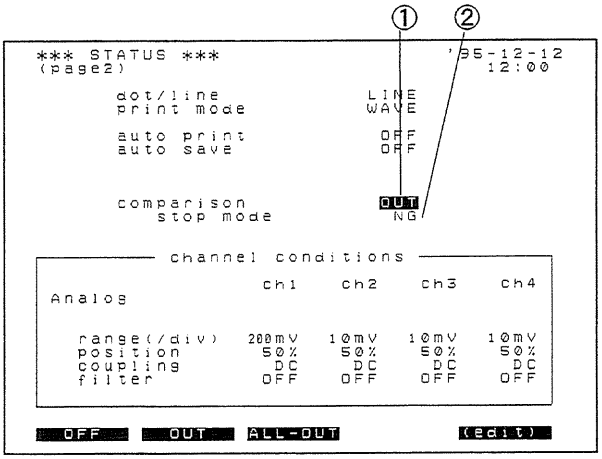
OFF : OFF
FD : Floppy disk
SCSI : Hard disk or magneto-optical disk (Saves in the current directory.)

For details, see Chapter 13 of the 8853 Instruction Manual.

4.22 Waveform Decision Function (comparison)

- This function provides a pass/fail decision (GO/NG) for the FFT analysis result with respect to an arbitrarily defined decision area.
- This function can be used in SINGLE format. (Can not be used in DUAL format.)
- For details, see Chapter 10 "Waveform Decision Function" of the 8853 Instruction Manual.

Procedure (Setting screen: status (page 2))



Status screen (page 2)

1. Move the flashing cursor to the "comparison" item, ① as shown in the figure on the left.
2. Select the required decision mode by using the soft keys.

Soft key indication

OFF : Decision is not made.
OUT : Decision result is fail if the waveform leaves the decision area at any point.
ALL-OUT : Decision result is fail if the waveform is entirely outside the decision area.
(edit) : Graphics editor screen (see Section 10.4 of the 8853 Instruction Manual) appears.

3. Move the flashing cursor to the "stop mode" item, ②.
4. Select the required stop mode by using the soft keys.

Soft key indication

GO : Stop operation only after a pass.
NG : Stop operation only after a fail.
GO&NG : Stop operation regardless of the decision result.
(edit) : Graphics editor screen (see Section 10.4 of the 8853 Instruction Manual) appears.

For the procedure for setting up the decision area, see Section 10.4 "Using the Graphics Editor" of the 8853 Instruction Manual.

4.23 Operation Example

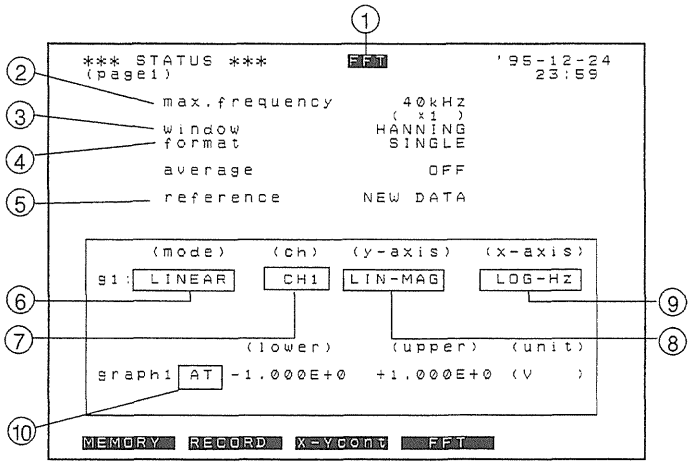
This example illustrates the basic procedure using the FFT function to measure the linear spectrum of a 3 V p-p 1 kHz sine wave input.

(1) Power on the unit/Input connection

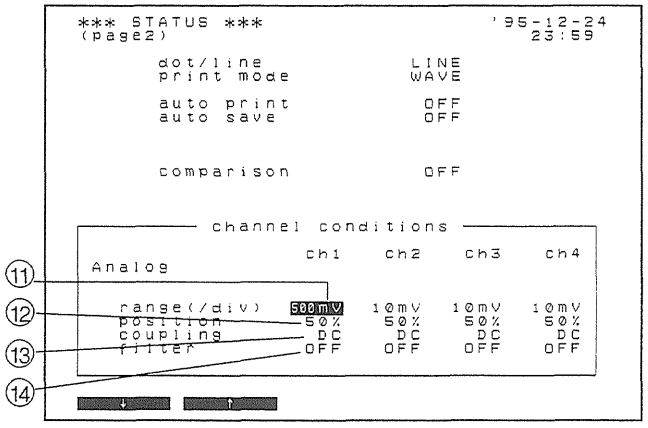
- Connect the power cord to the 8853 and press the power switch.
- Connect a signal generator to the input terminals of channel 1 (the 8945 analog unit).
- Set the signal generator so that it outputs a 3 V p-p 1 kHz sine wave.

(2) Settings on the status screen

Using the cursor keys and the soft keys, make settings as shown in the figure below.



Status screen (page 1)

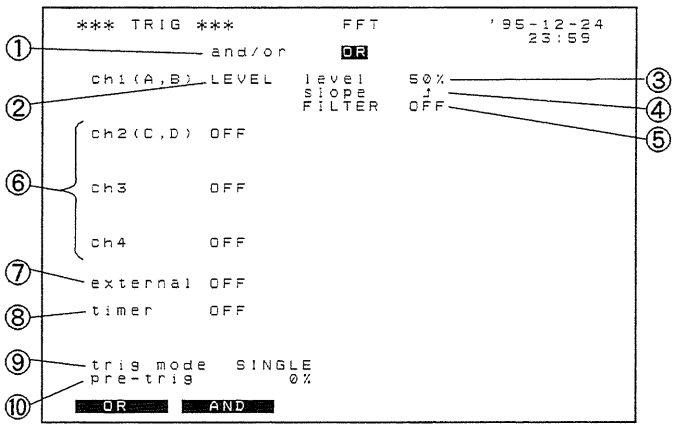


Status screen (page 2)

① Function mode	FFT	
② Frequency range max. frequency	40 kHz	
③ Window window	HANNING	
④ Format format	SINGLE	
⑤ Reference data reference	NEW DATA	
⑥ Analysis mode (mode)	LINEAR	
⑦ Analysis channel (ch)	CH1	
⑧ Y-axis (y-axis)	LIN-MAG	
⑨ X-axis (x-axis)	LOG-Hz	
⑩ Display scale	AT	
⑪ Voltage axis range range (/div)	500 mV	ch 1 only
⑫ Position position	50 %	ch 1 only
⑬ Input coupling coupling	DC	ch 1 only
⑭ Filter filter	OFF	ch 1 only

(3) Settings on the trigger screen (See Chapter 8 of the 8853 Instruction Manual.)

Using the cursor keys and the soft keys, make settings as shown in the figure below.



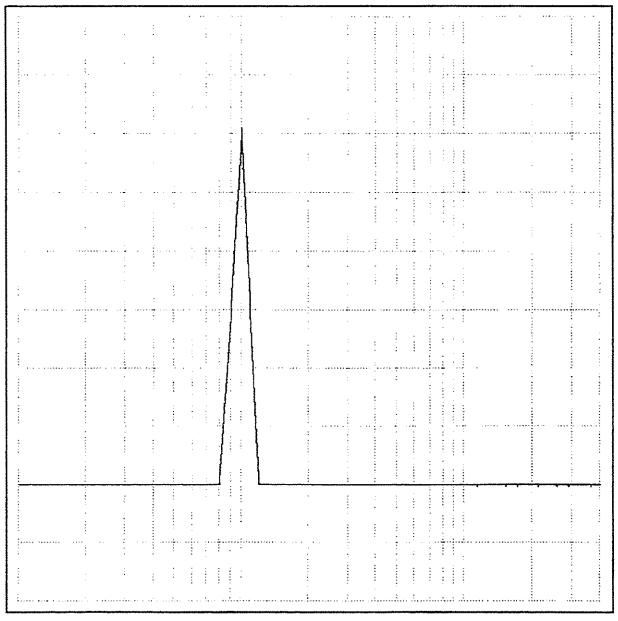
Trigger screen

① Trigger logical operator and/or	OR
② ch1 (A, B)	LEVEL
③ Trigger level level	50 %
④ Trigger direction slope	↑
⑤ Filter FILTER	OFF
⑥ ch2 to ch4	OFF
⑦ External trigger external	OFF
⑧ Timer trigger timer	OFF
⑨ Trigger mode trig mode	SINGLE
⑩ Pre-trigger pre-trig	0 %

(The items of ③, ④ and ⑤ are shown, after the setting for ② is made.)

(4) Measurement and calculation

- Press the START key to start measurement.
- 800 points of data are captured, and FFT analyses are performed.
- The analysis result is displayed on the screen, and measurement and calculation automatically terminate.
- The analysis result can be printed out by using the PRINT key.

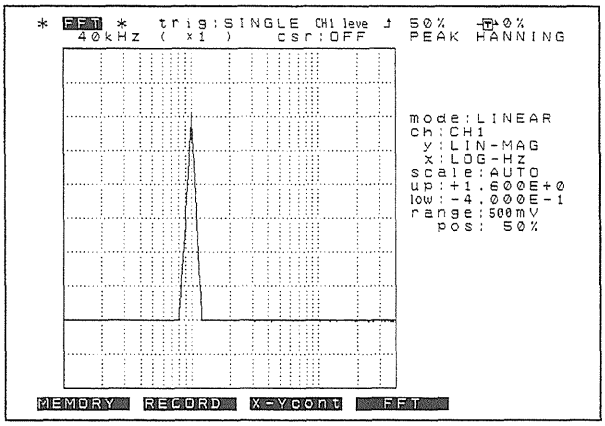


4.24 Interpreting Waveform Displays and Recordings

This section illustrates examples of the display and printed recording (manual print).

(1) SINGLE format

Display



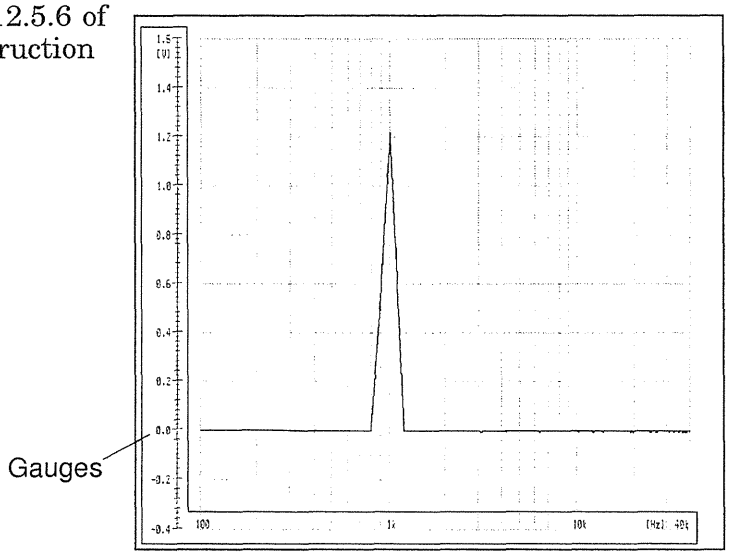
Linear Spectrum

Y-axis: Linear-Mag
X-axis: Log-Hz

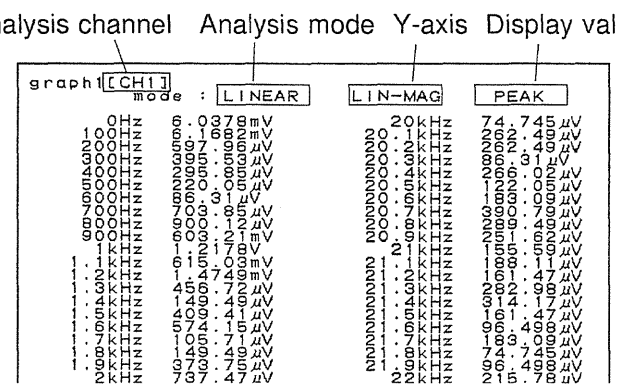
Manual print

[Waveform] When using the gauge function

(See Section 12.5.6 of the 8853 Instruction Manual.)



[Numerical values]



(2) DUAL format

Display

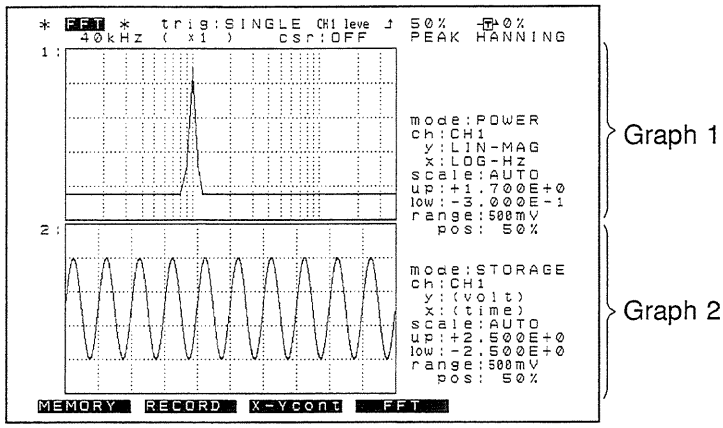
[Graph 1]

Power Spectrum

Y-axis: Linear-Mag
X-axis: Log-Hz

[Graph 2]

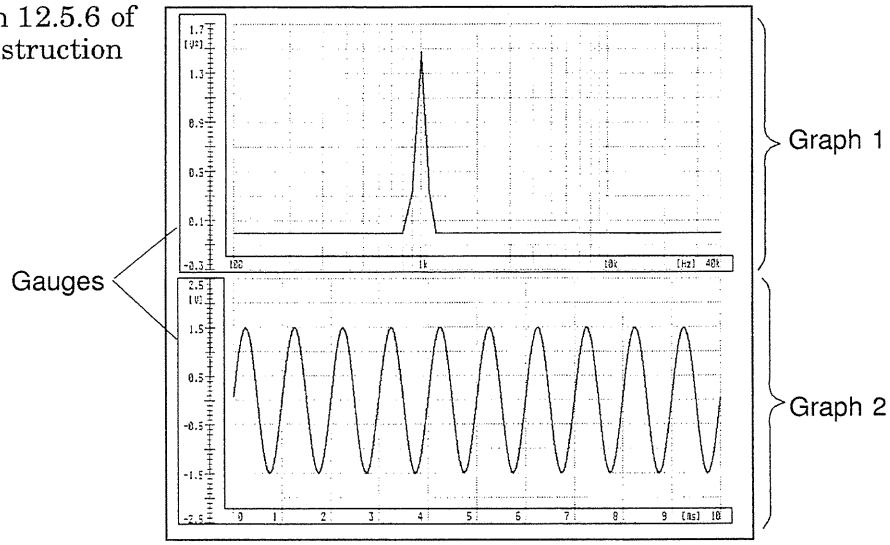
Storage waveform



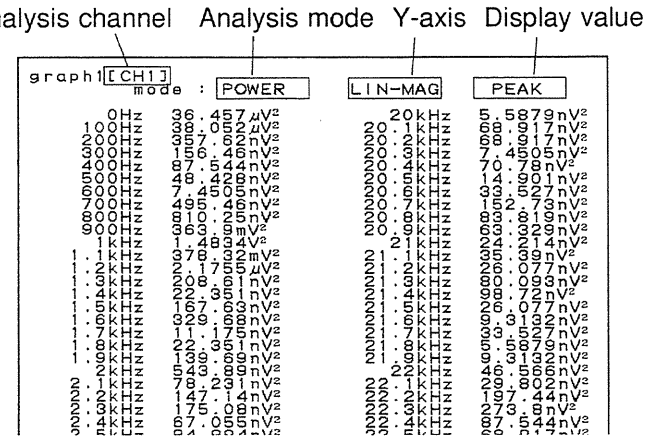
Manual print

[waveform] When using the gauge function

(See Section 12.5.6 of the 8853 Instruction Manual.)



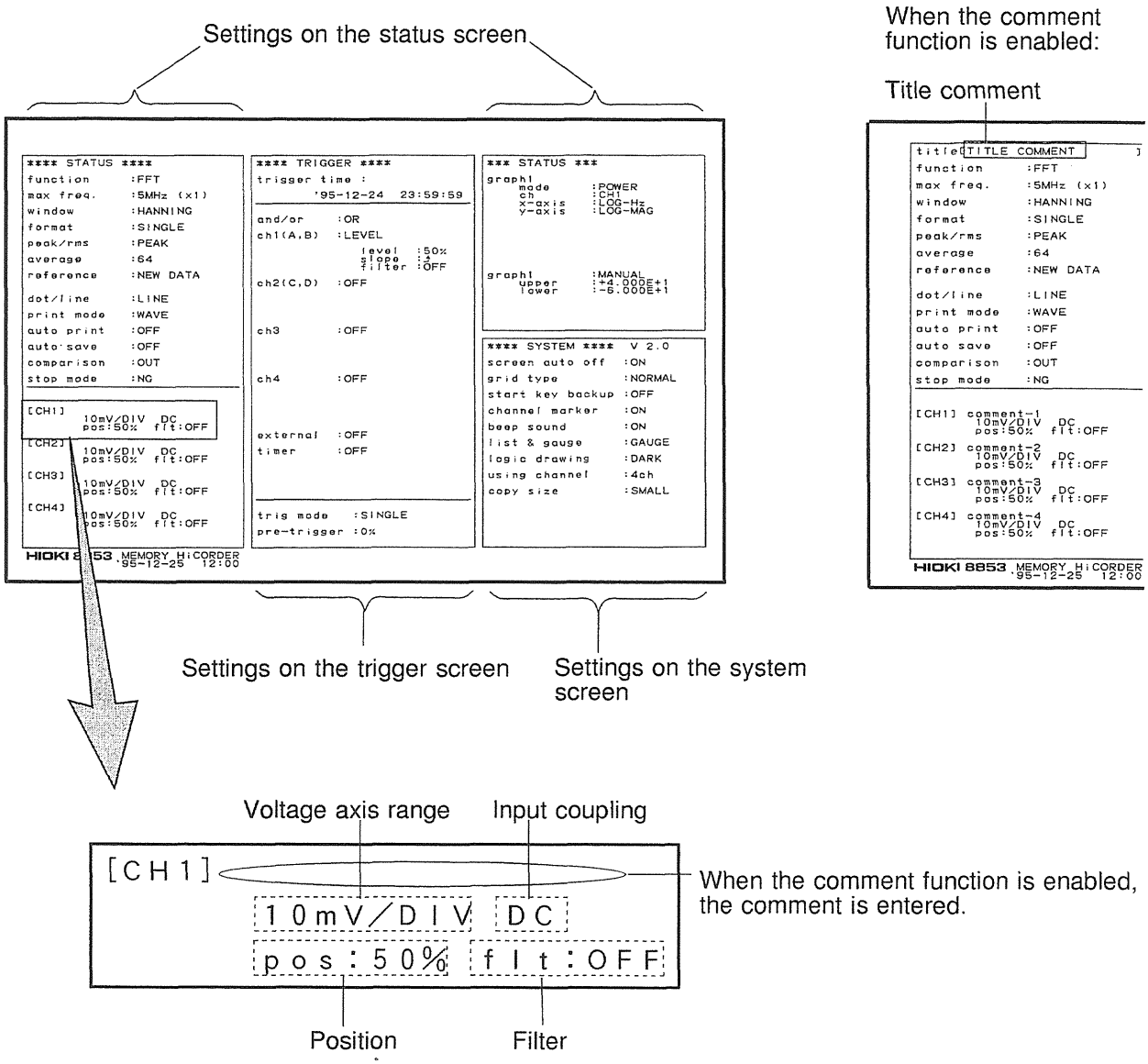
[Numerical values]



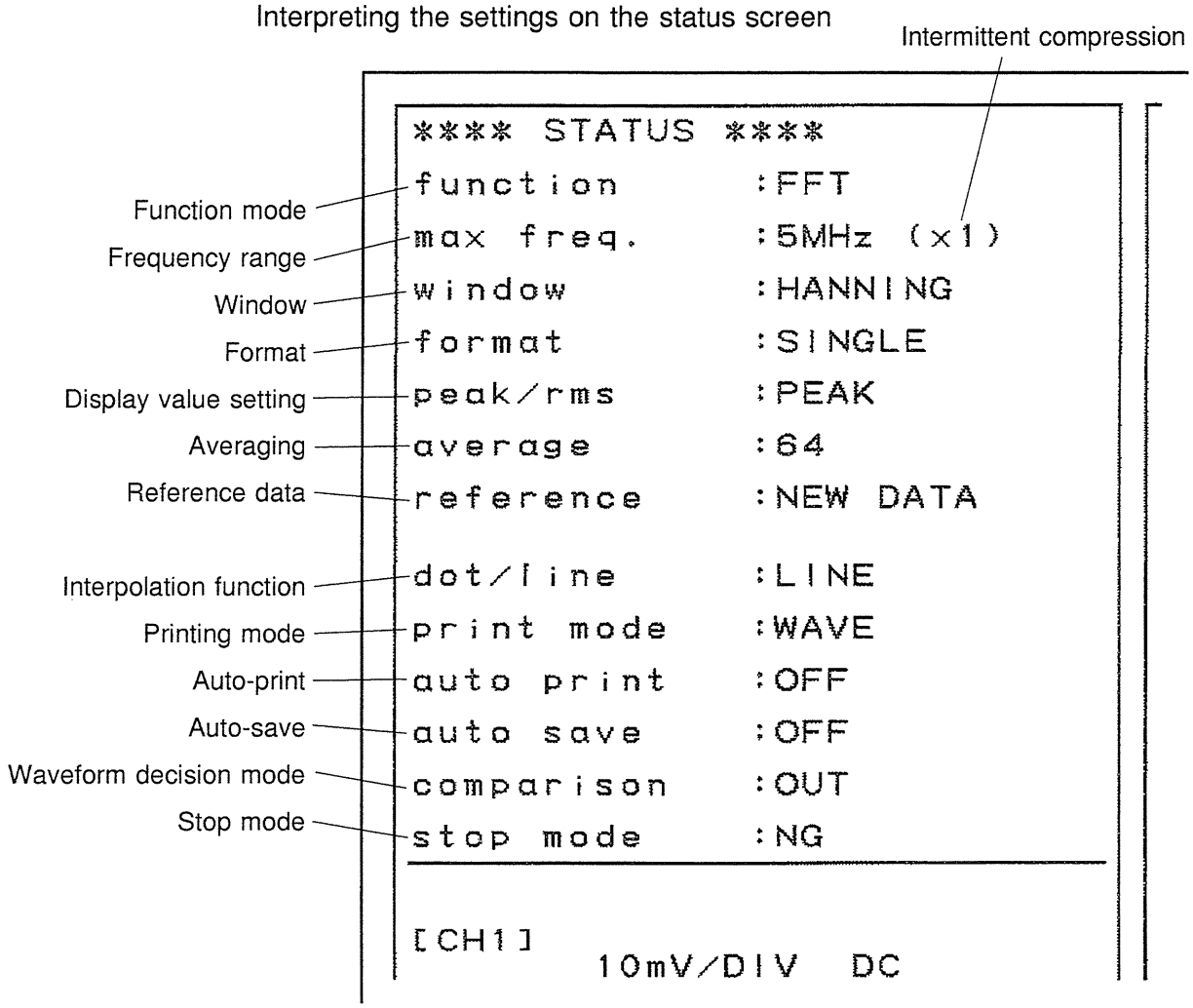
The data for graph 2 are printed after graph 1.

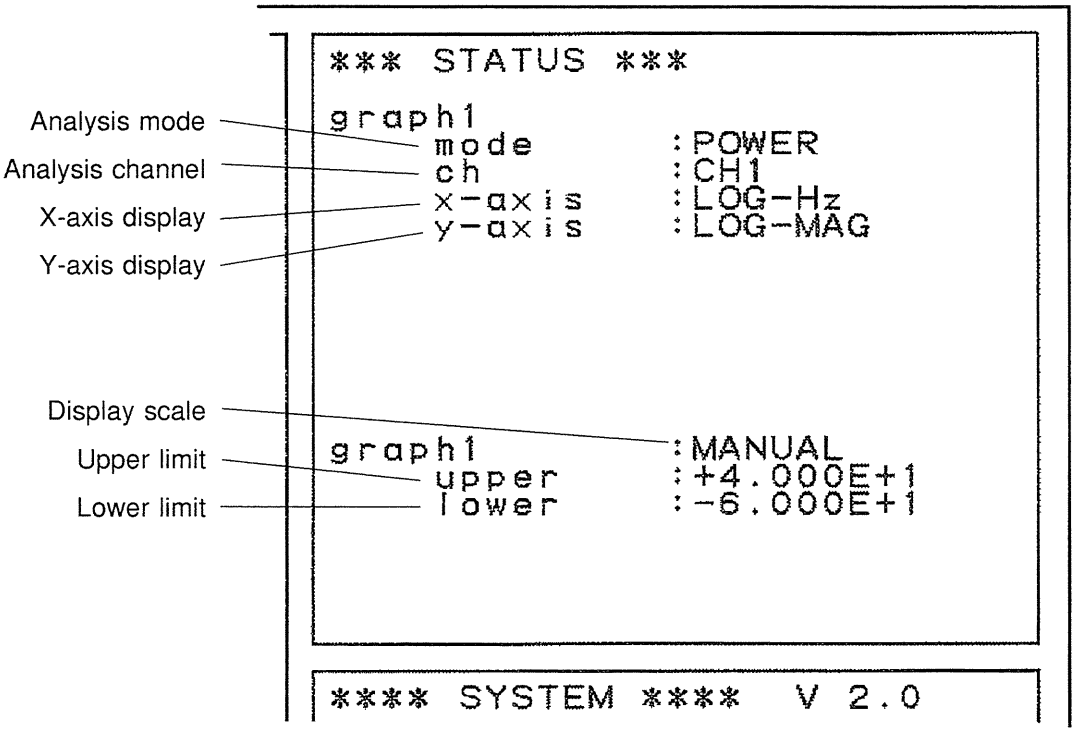
4.25 Interpreting Listing

- This section illustrates the listing.
- The listing is printed by enabling the list function (see Section 12.5.6 of the 8853 Instruction Manual) or pressing the [PRINT] key on the status, trigger or system screen.



Interpreting the settings on the status screen





In DUAL format, the settings for graph 2 are also displayed.

Chapter 5
GP-IB Outline

The GP-IB (General Purpose Interface Bus) was developed as an interface for general use by programmable instrumentation, and as an interface is rich in expandability and has many distinctive features.

There are various interfaces with specific names apart from the GP-IB, such as the IEEE-488 bus, the IEC bus, and the HP-IB which is an internal standard within the Hewlett-Packard Company. These are basically the same standard, but, because the number of connector pins and the arrangement of the signals and so on differ, much care should be exercised.

In this explanation of management and operation, only the GP-IB related resources of the 8853 will be described.

If more detailed knowledge of the GP-IB interface is required, reference should be made to the following literature:
The Institute of Electrical and Electronics Engineers, Inc.: "IEEE Standard Digital Interface for Programmable Instrumentation", IEEE Std 488.1-1987, IEEE Std 488.2-1987 (1987)

Chapter 6
GP-IB Specification

6.1 Standards

IEEE Standard 488.1-1987
IEEE Standard 488.2-1987

6.2 Interface Functions

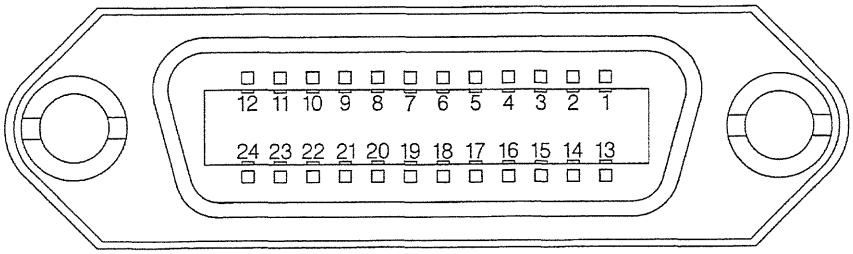
Function	Implementation
SH1	SH (Source Handshake) - All Functions
AH1	AH (Acceptor Handshake) - All Functions
T5	Basic Talk Function, Serial Poll Function, Talk Only Function MLA (My Listen Address) Talk Release Function
L4	Basic Listener Function MTA (My Talk Address) Listen Release Function
SR1	SR (Service Request) - All Functions
RL1	RL (Remote/Local) - All Functions
PP0	PP (Parallel Poll) - No Function
DC1	DC (Device Clear) - All Functions
DT0	DT (Device Trigger) - No Function
C0	C (Control) - No Function

6.3 GP-IB Signal Lines

Bus Signal Lines		Remarks	
Data bus	DIO 1 (Data Input Output 1) DIO 2 (Data Input Output 2) DIO 3 (Data Input Output 3) DIO 4 (Data Input Output 4) DIO 5 (Data Input Output 5) DIO 6 (Data Input Output 6) DIO 7 (Data Input Output 7) DIO 8 (Data Input Output 8)	These are used for: Input and output of data. Input and output of interface messages. Input and output of device messages.	
Transfer bus	DAV (Data Valid)	Signal which indicates data bus information validity.	These perform acceptor and source handshake.
	NRFD (Not Ready For Data)	Input preparation completed signal.	
	NDAC (Not Data Accepted)	Input completed signal.	
Control bus	ATN (Attention)	Signal which indicates that the information on the data bus is an interface message or a device message.	
	IFC (Interface Clear)	Signal which sets the interface bus system to the initial condition.	
	SRQ (Service Request)	Signal which requests a non-synchronous service.	
	REN (Remote Enable)	Signal which performs changeover of remote and local control.	
	EOI (End or Identify)	Indicates the last byte of data.	

6.4 Connector Pin Assignment

On the 8853 57LE-20240 (made by DDK) or compatible.
On the cable 57-10240 (made by DDK) or compatible.

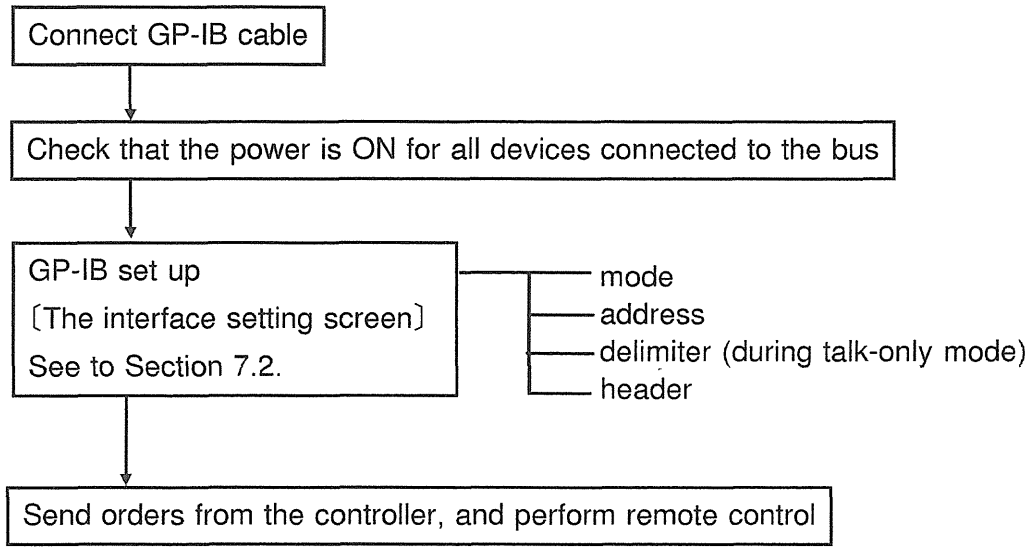


Pin arrangement diagram for the GP-IB interface connector on the 8853

Pin number	Name of signal line	Pin number	Name of signal line
1	DIO 1	13	DIO 5
2	DIO 2	14	DIO 6
3	DIO 3	15	DIO 7
4	DIO 4	16	DIO 8
5	EOI	17	REN
6	DAV	18	GND
7	NRFD	19	GND
8	NDAC	20	GND
9	IFC	21	GND
10	SRQ	22	GND
11	ATN	23	GND
12	SHIELD	24	LOGIC GND

Chapter 7
Method of Operation

7.1 Operational Procedure



⚠ WARNING

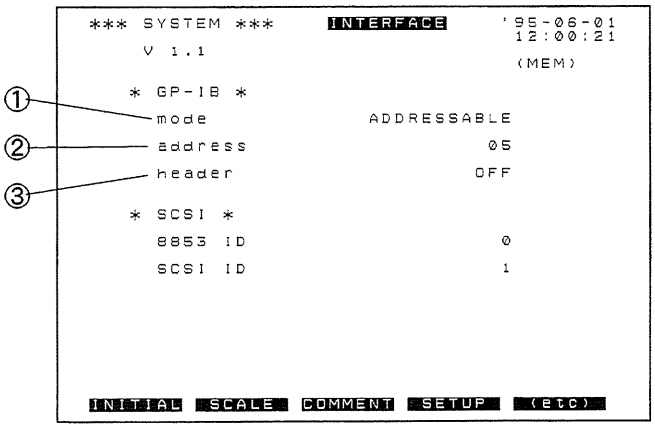
- The GP-IB interface is not isolated from the 8853 system.
- Exercise caution, because the ground of the logic and analog inputs, and the GP-IB interface ground are connected.

7.2 Setup Procedure

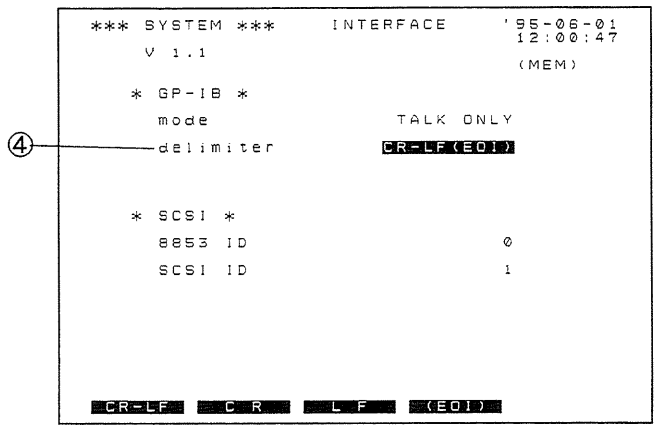
- On the 8853, set the GP-IB address for the unit, and select whether or not to use headers mode, and delimiter in messages output by the 8853.
- Use the interface setting screen, accessed from the "system" screen.

Procedure

1. Press the SYSTEM key to display the system screen.
2. Press the **INTER** soft key, and the interface setting screen appears.



System screen (INTERFACE)



In talk-only mode

3. Set the GP-IB operation mode for this unit.

   Set the GP-IB address for this unit on the bus in mode item (① on the left figure).

   Soft key indication

   ADDRESS : (ADDRESSABLE) Assign a device address, so this unit can be used both as talker and listener.

   TALK : (TALK ONLY) Use this unit as talker only. (Only when using a plotter.)

   DISABLE : Do not use the GP-IB interface.

4. Set the address.

   Use the ↓ and ↑ soft keys, or the rotary knob to adjust the numerical value in address item (② on the left upper figure).
   [0 to 30]

5. Enable or disable the headers.

   Select whether or not this unit as talker should output an identifying header at the beginning of each message it sends in header item (③ on the left upper figure).
   [OFF, ON]

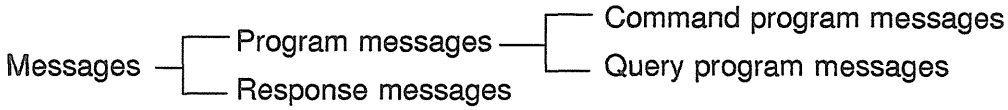
6. Select the delimiter for talk-only mode.

   Select the appropriate delimiter sequence for the plotter being used in delimiter item (④ on the left figure).

   [CR-LF(EOI), CR(EOI), LF(EOI), (EOI)]

7.3 Receive and Send Protocols

(1) Messages



- Data received or sent by the GP-IB interface is called a message.
- Of these, program messages are those received by the unit from the controller, while response messages are those sent from the unit to the controller.
- Program messages are command messages or query messages.
- After a query message has been received, a response message is produced the moment that its syntax has been checked.

Command messages are orders for control of the device, such as for making settings or for reset or the like.

Query messages are orders for responses relating to the results of operation, results of measurement, or the state of device settings.

Response messages are sent in response to query program messages.

(2) Command syntax

(When no ambiguity would arise, the term "command" is henceforth used to refer to both command and query program messages.)

- The 8853 accepts commands without distinction between lower case and upper case letters.
- It generates response messages in the long form (when headers are enabled) and in upper case letters.
- The names of commands for the 8853 are as far as possible mnemonic.
- All commands have a long form, and an abbreviated short form.
- In command references in this manual, the short form is written in upper case letters, and then this is continued in lower case letters so as to constitute the long form.

Example

DISPlay
Short form
Long form

DISPLAY
DISP
} Either will be accepted.

DISPLA
DISPL
DIS
} Any one will generate an error.

(3) Command program headers

- Commands must have a header.
- A header identifies the command in question.
- There are three kinds of header: the simple command type, the compound command type, and standard command type.

① Simple command type header

The first word constitute the header.

Example :HEADer ON

Simple command type header — Data

② Compound command type header

A header made up from a plurality of simple command type headers marked off by colons.

Example :CONFigure:TDIV 1.E-3

Simple command type header — Data

Compound command type header

③ Standard command type header

A command beginning with an asterisk and stipulated by IEEE 488.2

Example *RST

(4) Query program headers

- These are for commands used for interrogating the unit about the result of an operation or about a setting.
- These can be recognized as queries by a question mark appearing after the program header.
- The structure of the header is identical to that of a command program header, with "?" always being affixed to the last command.
- There are queries possible in each of the three previously described types of command form.

Example :HEADer? ON

Query program header — Data

(5) Response messages

- Response messages relating to queries are made up from header portions (which also may be absent due to header disablement) and data portions identical to those of program messages.
- As a general rule are sent in an identical format to the format of the program message corresponding to their originating query.

(6) Terminators and separators

① Message terminator

- A terminator is used in order to separate the transmission of one message from another, and this terminator is not itself included in the message.
- On the 8853, LF, EOI, or LF+EOI is used as the message terminator.
- LF+EOI is also used as the response message terminator.

② Message unit separator

A semicolon ";" is used as a message unit separator when it is desired to set out several messages on a single line.

Example :CONFIGURE:TDIV 1. E-3;:CONFIGURE:SHOT 15

Message unit separator

③ Header separator

A space " " is used as a header separator to separate the header from the data.

Example :CONFIGURE:SHOT 15

Header separator

④ Data separator

A comma "," is used as a data separator for separating several data items from one another.

Example :DISPLAY:DRAW CH1, DARK

Data separator

(7) The command tree

- The rule when writing several messages of compound command form on the same line, when no colon is prefixed to the next header after the semicolon (the message unit separator), is that that header is considered as continuing on from the header before the last colon in the message directly preceding.
- This corresponds to the general concept of the current directory in the directory structure of UNIX or MS-DOS, and this directly preceding header is called the "current path".

To reiterate, the colon at the beginning of a command forces the search for the command to begin from the root. Thus in Example 1:

Example 1 :CONFIGURE:TDIV 1.E-3

The first colon indicates that the "CONFIGURE" command is at the root level.

- Both Example 2 and Example 3 are messages setting TIME/AXIS to 1 ms and recording length to 15 divisions.

  Example 2 :CONF:TDIV 1. E-3;:CONF:SHOT 15

  Example 3 :CONF:TDIV 1. E-3;SHOT 15

  With Example 2, because there is a colon directly after the semicolon, the current position is the "root". Accordingly the reference of the next command is performed from the root.

  On the other hand, with Example 3, because with ":CONF:TDIV 1. E-3;" the current path has become ":CONF", it is now possible to omit the ":CONF" before "SHOT".

(8) Data format

The 8853 uses character data, character string data and decimal data.

① Character data

- The first character must be alphabetic.
- The characters after the first character can only be alphabetic characters, numerals, or underline characters (_).
- As alphabetic characters, during sending only upper case letters are used, but during receiving both upper case and lower case letters are permitted.

② Character string data

- Character string data is enclosed within quotation marks.
- The data is composed of 7 bit ASCII characters.
- Characters which cannot be handled by the 8853 are replaced by spaces.
- When the 8853 is sending, only the double quotation mark (") is used as a quotation mark, but when receiving both this double quotation mark and also the single quotation mark (') are accepted.

③ Decimal data

- Decimal data values are represented in what is termed NR format.
- There are three types of NR format from NR1 to NR3, and each of these can appear as either a signed number or an unsigned number. Unsigned numbers are taken as positive.
- If the accuracy of a numerical value exceeds the range with which the 8853 can deal, it is rounded off. (5 and above is rounded up; 4 and below is rounded down).

  NR1 format - integer data
  Examples: +15, -20, 25
  NR2 format - fixed point numbers
  Examples: +1.23, -4.56, 7.89
  NR3 format - floating point numbers
  Examples: +1.0E-3, -2.3E+3

  } NRf format (includes all three formats on the left)

- When the 8853 is receiving it accepts NRf format, but when it is sending it utilizes whichever one of the formats NR1 to NR3 is indicated in the particular command.

7.4 Remote Local

(1) Local state

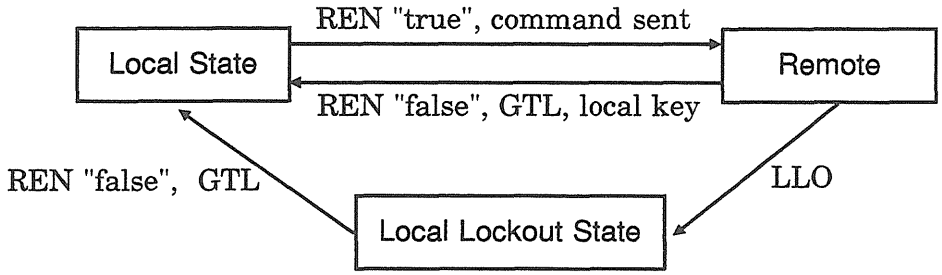
This is the state in which the 8853 is controlled by its keys. When the power is turned on, the 8853 always comes up in local state.

(2) Remote state

- In this state the 8853 is controlled from the GP-IB interface (the REN line is "true"), and its keys are disabled.
- When in the remote state, the 8853 returns to local state if the local key (the [LCL] soft key) is pressed.

(3) Local lockout state

- When an LLO (Local Lockout) command (this is a GP-IB universal command) is sent, even if the local key is pressed, the 8853 is prevented from returning to the local state. (This state is called the local lockout state.)
- In order to return the 8853 from the local lockout state to the local state, it is necessary either (a) to send a GTL (Go To Local) command (this is a GP-IB universal command), or (b) to turn the power to the 8853 temporarily off and then on again, or (c) to bring the line REN to "false".
- If a command is sent with REN in the "false" state, then the only way to return to the local state is with the local key.



Program example HP-9816 (Hewlett-Packard)

local lockout	LOCAL LOCKOUT 7
local	LOCAL 7

7.5 Device Clear

- When the 8853 receives the device clear command, it executes following processing:

· Clears the input buffer
· Clears the output queue.

- The device clear command is exemplified by the following:

HP 9816 (made by Hewlett-Packard)
CLEAR 7

7.6 The Status Byte and the Event Registers

(1) The status byte

- Each bit of the status byte is a summary (logical OR) of the event register corresponding to that bit.
- The status byte and each event register has an enable register corresponding to it, and according to the setting of this enable register (which starts off at zero when the power is turned on) it is possible to mask the service requests originating from each event.

Status byte bit settings

bit 7	Unused: 0
bit 6 RQS MSS	Set when a service request is issued.
bit 5 ESB	Event summary bit. Shows a summary of the standard event status register.
bit 4 MAV	Message available. Shows that a message is present in the output queue.
bit 3	Unused: 0
bit 2	Unused: 0
bit 1	Unused: 0
bit 0 ESB0	Event summary bit 0 Shows a summary of event status register 0.

- The following commands are used for reading the status byte, and for setting the service request enable register and for reading it.

  - Reading the status byte *STB?
  - Setting the service request enable register *SRE
  - Reading the service request enable register *SRE?

(2) Standard event status register (SESR)

- The summary of this register is set in bit 5 of the status byte.
- Each bit is masked by setting the standard event status enable register (which starts off at zero when the power is turned on).

● The circumstances when the contents of the standard event status register are cleared are as listed below.

① When the *CLS command is received.

② When the contents have been read by an *ESR? query.

③ When the power is turned off and turned on again.

Bit allocations in the standard event status register

bit 7 PON	The power has been turned on again. Since this register was last read, the unit has been powered off and on.
bit 6 URQ	User request: not used.
bit 5 CME	Command error. There is an error in a command that has been received; either an error in syntax, or an error in meaning.
bit 4 EXE	Execution error. An error has occurred while executing a command. Range error; Mode error.
bit 3 DDE	Device dependent error. It has been impossible to execute some command, due to an error other than a command error, a query error, or an execution error.
bit 2 QYE	Query error. The queue is empty, or data loss has occurred (queue overflow).
bit 1	Request for controller right (not used) Unused: 0
bit 0 OPC	Operation finished. Only set for the *OPC command.

● Command used

- Read the standard event status register *ESR?
- Set the standard event status enable register *ESE
- Read the standard event status enable register *ESE?

(3) Event status register 0 (ESR0)

- The summary of this register is set in bit 0 of the status byte.
- Each bit is masked when the event status enable register 0 (which starts off at zero when the power is turned on) is set.

● The circumstances when the contents of event status register 0 are cleared are as listed below.

① When the *CLS command is received.

② When the contents have been read by an :ESR0? query.

③ When the power is turned off and turned on again.

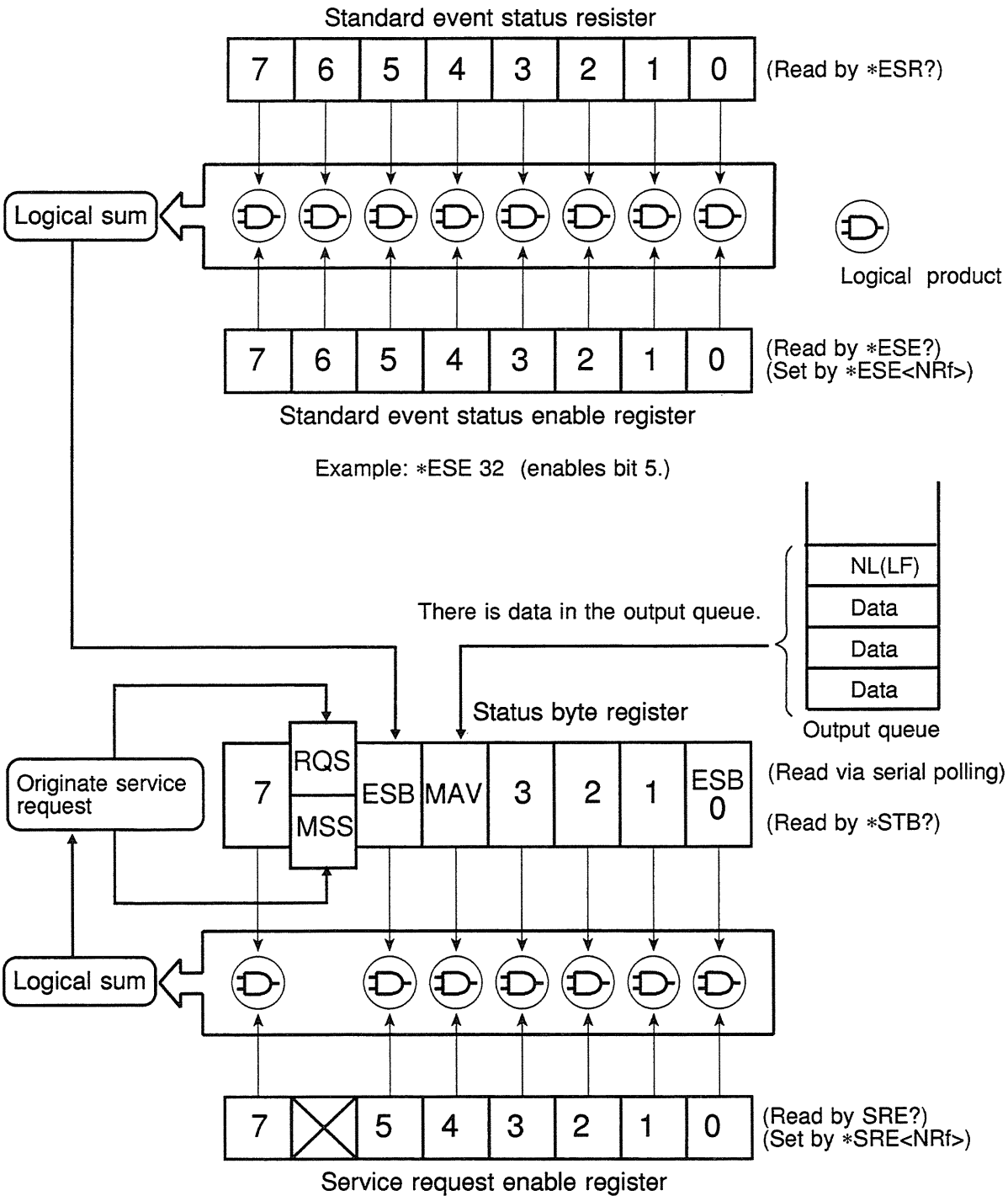
The bits of event status register 0

bit 7	Waveform decision fail (NG).
bit 6	Unused.
bit 5	Waveform parameter calculation finished.
bit 4	Waveform processing calculation finished.
bit 3	Printer operation finished (print, or copy output).
bit 2	Trigger wait finished (set when the trigger event occurs).
bit 1	Measurement operation concluded (set by STOP).
bit 0	Error not related to the GP-IB interface; printer error etc.

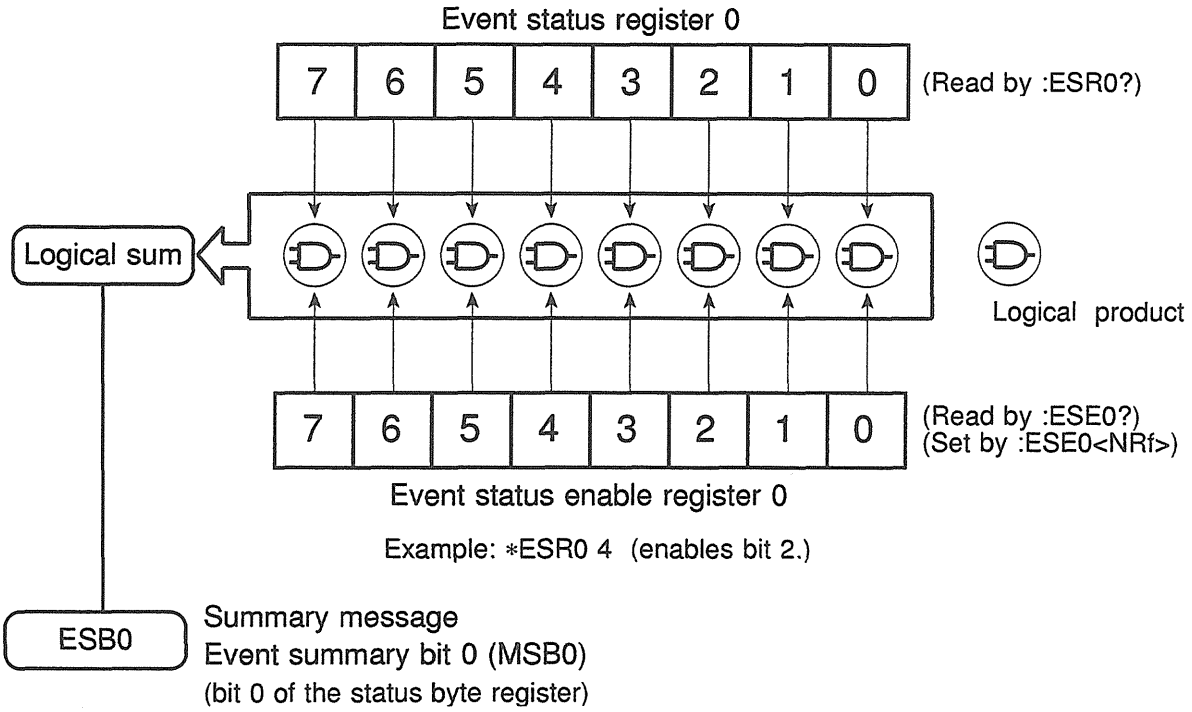
● Command used

- Reading event status register 0 :ESR0?
- Setting event status enable register 0 :ESE0
- Reading event status enable register 0 :ESE0?

Status byte data structure



Event status register 0 data structure



7.7 The Input Buffer and the Output Queue

(1) Input buffer

The 8853 has an input buffer of 512 bytes capacity.
Messages which are received are put into this buffer and executed in order.
(However, an ABORT command is executed instantly as soon as it is received.)

(2) Output queue

- The 8853 has an output queue of 256 bytes capacity.
- Response messages are accumulated in this queue and are read out from the controller.
- If the length of a response message has exceeded 256 bytes, a query error occurs.

● The circumstances when the output queue is cleared are as listed below:

① When the controller has read out its entire contents.

② When a device clear is issued.

③ When the power is turned off and turned on again.

④ Upon receipt of the next message.

7.8 GP-IB Errors

- When a command which has been received contains an error, that one of bits 2 to 5 of the standard event status register which corresponds to the event is set.
- If a command has given rise to an error (apart from an execution error), commands accumulated in the input buffer and waiting for execution after that command are ignored.

Bit allocations in the standard event status register

bit 7 PON	The power has been turned on again. Since this register was last read, the unit has been powered off and on.
bit 6 URQ	User request: not used.
bit 5 CME	Command error. There is an error in a command that has been received; either an error in syntax, or an error in meaning.
bit 4 EXE	Execution error. An error has occurred while executing a command. Range error; Mode error.
bit 3 DDE	Device dependent error. It has been impossible to execute some command, due to an error other than a command error, a query error, or an execution error.
bit 2 QYE	Query error. The queue is empty, or data loss has occurred (queue overflow).
bit 1	Request for controller right (not used). Unused: 0
bit 0 OPC	Operation finished. Only set for the *OPC command.

Chapter 8
GP-IB Commands

8.1 Command Summary

(1) Standard commands specified by IEEE 488.2

Command	Data (for a query, response data)	Explanation	Ref page
*IDN?	Maker's name, model number, serial number, software version (not used, zero)	Queries device ID.	105
*OPT?	Whether channel 1 to channel 4 input unit exists. (0=no present, 1=input unit present)	Queries device option provision.	105
*RST		Device initial setting.	106
*TST?	*A*<NR1> (0 = normal, 1 = failure)	Queries the result of the self-test.	106
*OPC		Sets the LSB of SESR after all action has been completed.	106
*OPC?	*A*<NR1>	Queries whether all action has been completed. ASCII [1] is the response.	107
*WAI		Wait until action fully completed.	107
*CLS		Clears the status byte and associated queues.	107
*ESE *A*	*A* = 0 to 255	Sets SESER.	108
*ESE?	*A*<NR1> 0 to 255	Queries SESER.	
*ESR?	*A*<NR1>	Queries SESR.	108
*SRE *A*	*A* = 0 to 255	Sets SRER.	109
*SRE?	*A*<NR1> 0 to 63, 128 to 191	Queries SRER.	
*STB?	*A*<NR1> 0 to 255	Reads the STB and the MSS bit, without performing serial polling.	109

Command	Data (for a query, response data)	Explanation	Ref page
:ESE0 *A* #	*A* = 0 to 255	Writes ESER0.	110
:ESE0? #	*A*<NR1> 0 to 255	Reads ESER0.	
:ESR0? #	*A*<NR1> 0 to 255	Reads ESR0.	110

#: specific to the 8853.

(2) Commands specific to the 8853.

1. Commands of execution control etc. (Common to all functions)

Command	Data (for a query, response data)	Explanation	Ref page
:STARt		Same as the START key.	111
:STOP		Same as the STOP key.	111
:ABORT		Forced halt.	111
:PRINt		Same as the PRINT key.	111
:HCOPy		Same as the COPY key.	112
:FEED *A*	*A* = 1 to 255 (unit; mm)	Feeds the paper the specified distance.	112
:AUTO		Sets the time axis and the voltage axis automatically. (Only the memory recorder function)	112
:ERRor?	*A*<NR1> error number	Queries 8853 error number.	112
:HEADer *A$*	*A$* = OFF, ON	Enables and disables headers.	113
:HEADer?	*A$*	Queries header enablement.	
:FUNCtion *A$*	*A$* = MEM, REC, XYC, FFT	Changes the function.	113
:FUNCtion?	*A$*	Queries the function.	

MEM	memory recorder function	REC	recorder function
XYC	XY recorder function	FFT	FFT function
All	MEM, REC, XYC, and FFT function		

2. CONFigure command (Setting and querying the time axis range, the recording length, etc.)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CONFigure				
:TDIV *A*	*A* = time axis range (unit; seconds)	Sets the time axis range.	MEM REC	114
:TDIV?	*A*<NR3> (unit; seconds)	Queries the time axis range.		
:SHOT *A*	*A* = recording length (unit; DIV)	Sets the recording length.	MEM REC	114
:SHOT?	*A*<NR1> (unit; DIV)	Queries the recording length.		
:FORMat *A$*	*A$* = SINGle, DUAL, QUAD, XY (MEM) SINGle, DUAL, QUAD (REC) SINGle, DUAL (FFT)	Sets the format.	MEM REC FFT	115
:FORMat?	*A$*	Queries the format.		
:DOTLine *A$*	*A$* = DOT, LINE	Sets the interpolation function.	MEM XYC FFT	115
:DOTLine?	*A$*	Queries the interpolation function.		
:OVWRite *A$*	*A$* = OFF, ON	Enables and disables waveform superimposition.	MEM	116
:OVWRite?	*A$*	Queries waveform superimposition enablement.		
:ATPRint *A$*	*A$* = OFF, ON	Enables and disables auto print.	MEM FFT	116
:ATPRint?	*A$*	Queries auto print enablement.		
:ATSAve *A$*	*A$* = OFF, FD, SCSI	Enables and disables auto save. FD: Floppy disk SCSI: SCSI	MEM FFT	116
:ATSAve?	*A$*	Queries auto save enablement.		
:SMOOth *A$*	*A$* = OFF, ON	Enables and disables smooth printing.	MEM	117
:SMOOth?	*A$*	Queries smooth printing enablement.		
:ROLL *A$*	*A$* = OFF, ON	Enables and disables roll mode.	MEM	117
:ROLL?	*A$*	Queries roll mode enablement.		
:AVERage *A*	*A* = 0, 2, 4, 8, 16, 32, 64, 128, 256 (0: OFF)	Sets the count for averaging.	MEM FFT	117
:AVERage?	*A*<NR1>	Queries the current setting of the count for averaging.		
:MEMDiv *A$*	*A$* = OFF, SEQ, MULTI	Sets the memory segmentation function.	MEM	118
:MEMDiv?	*A$*	Queries the memory segmentation function.		

MEM memory recorder function
REC recorder function
XYC XY recorder function
FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CONFigure				
:MAXBlock *A*	*A* = 2 to 63	Sets the memory block number (in multi-block function).	MEM	118
:MAXBlock?	*A*<NR1>	Queries the memory block number.		
:STTBlock *A*	*A* = 1 to number of blocks	Sets the start block (in sequential save function).	MEM	119
:STTBlock?	*A*<NR1>	Queries the start block.		
:ENDBlock *A*	*A* = 1 to number of blocks	Sets the end block (in sequential save function)	MEM	119
:ENDBlock?	*A*<NR1>	Queries the end block.		
:USEBlock *A*	*A* = 1 to number of memory segmentations	Sets the number of the memory block used (in sequential save and multi-block function).	MEM	119
:USEBlock?	*A*<NR1>	Queries the number of the memory block used.		
:REFBlock *A*	*A* = 0, 1 to number of memory segmentations (0: OFF)	Sets the reference block (in multi-block function).	MEM	120
:REFBlock?	*A*<NR1>	Queries the reference block.		
:PRINt *A$*	*A$* = OFF, ON	Enables and disables printer output.	REC	120
:PRINt?	*A$*	Queries printer output enablement.		
:WVCOmp *A$*	*A$* = OFF, OUT, ALLOut	Sets the waveform decision mode.	MEM FFT	120
:WVCOmp?	*A$*	Queries the waveform decision mode.		
:CMPStop *A$*	*A$* = GO, NG, G_N	Sets the waveform decision stop mode.	MEM FFT	121
:CMPStop?	*A$*	Queries the waveform decision stop mode.		
:MAXFreq *A*	*A* = frequency range (unit; Hz)	Sets the frequency range.	FFT	121
:MAXFreq?	*A*<NR3>	Queries the frequency range.		
:FFTWind *A$*	*A$* = RECTan, HANNing	Sets FFT window.	FFT	121
:FFTWind?	*A$*	Queries FFT window.		
:FFTRef *A$*	*A$* = NEW, MEM	Designates the source for FFT analysis data.	FFT	122
:FFTRef?	*A$*	Queries the current FFT analysis data source.		

MEM memory recorder function
REC recorder function
XYC XY recorder function
FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CONFigure	(*G$* = G1, G2)			
:FFTMode *G$,A$*	*A$* = STR, LIN, PSP	Sets the FFT analysis mode.	FFT	122
:FFTMode? *G$*	*G$,A$*	Queries the current FFT analysis mode.		
:FFTYaxis *G$,A$*	*A$* = LINREal, LINIMag, LINMAg, LOGMAg, PHASE	Sets the FFT y-axis.	FFT	123
:FFTYaxis? *G$*	*G$,A$* or *G$,*(volt)	Queries the present FFT y-axis setting.		
:FFTXaxis *G$,A$*	*A$* = LINhz, LOGhz	Sets the FFT x-axis.	FFT	124
:FFTXaxis? *G$*	*G$,A$* or *G$,*(time)	Queries the present FFT x-axis setting.		
:FFTSCale *G$,A$*	*A$* = AUTO, MANUal	Sets the FFT display scaling method for a graph.	FFT	124
:FFTSCale? *G$*	*G$,A$*	Queries the current FFT display scaling method for a graph.		
:FFTUp *G$,A*	*A* = -9.999E+9 to +9.999E+9	Sets the vertical axis upper limit value for FFT display.	FFT	125
:FFTUp? *G$*	*G$,A*<NR3>	Queries the current vertical axis upper limit value for FFT display.		
:FFTLow *G$,A*	*A* = -9.999E+9 to +9.999E+9	Sets the vertical axis lower limit value of FFT display.	FFT	125
:FFTLow? *G$*	*G$,A*<NR3>	Queries the current vertical axis lower limit value of FFT display.		
:FFTPrint *A$*	*A$* = WAVE, DATA	Sets FFT data printer output.	FFT	126
:FFTPrint?	*A$*	Queries FFT data printer output.		
:FFTThin *A$*	*A$* = X1, X1_2, X1_5	Sets the intermittent compression ratio for FFT analysis.	FFT	126
:FFTThin?	*A$*	Sets the intermittent compression ratio for FFT analysis.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

3. TRIGger command (Setting and querying trigger.)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:TRIGger	(*ch$* = CH1 to CH4)			
:SOURce *A$*	*A$* = OR, AND	Sets the inter-trigger AND/OR setting.	All	127
:SOURce?	*A$*	Queries the inter-trigger AND/OR setting.		
:KIND *ch$,A$*	*A$* = OFF, LEVEL, LOGIc, WINDow, TIMEout, GLITch (LOGIC: CH1, CH2 only)	Sets type of trigger.	All	127
:KIND? *ch$*	*ch$,A$*	Queries type of trigger.		
:LEVEl *ch$,A*	*A* = 0 to 100 (unit; %)	Sets the trigger level.	All	128
:LEVEl? *ch$*	*ch$,A*<NR1>	Queries the trigger level.		
:SLOPe *ch$,A$*	*A$* = UP, DOWN	Sets the trigger direction (slope).	All	128
:SLOPe? *ch$*	*ch$,A$*	Queries the trigger direction (slope).		
:FILTer *ch$,A*	*A* = 0.2 to 4000 (0: OFF) (LOGIC; CH1,CH2 only)	Sets the trigger filter width.	All	129
:FILTer? *ch$*	*ch$,A*<NR1>	Queries the trigger filter width.		
:WIDTh *ch$,A*	*A* = 2 to 4000	Sets the width for glitch detection or timeout trigger.	MEM FFT	129
:WIDTh? *ch$*	*ch$,A*<NR1>	Queries the width for glitch detection or timeout trigger.		
:FILTEvent *ch$,A$*	*A$* = FILTer, EVENt	Switches the trigger filter and event trigger.	MEM FFT	130
:FILTEvent? *ch$*	*ch$,A$*	Queries the trigger filter and event trigger.		
:UPPEr *ch$,A*	*A* = 1 to 100 (unit; %)	Sets upper limit level of window trigger.	All	130
:UPPEr? *ch$*	*ch$,A*<NR1>	Queries upper limit level of window trigger.		
:LOWEr *ch$,A*	*A* = 0 to 99 (unit; %)	Sets lower limit level of window trigger.	All	131
:LOWEr? *ch$*	*ch$,A*<NR1>	Queries lower limit level of window trigger.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:TRIGger	(*ch$* = CH1 to CH4)			
:LOGPat *ch$,'A$'*	*ch$* = CH1 or CH2 *A$* = XXXXXXXX trigger pattern (X, 0, 1)	Sets the trigger pattern for a logic trigger.	All	131
:LOGPat? *ch$*	*ch$,"A$"*	Queries the trigger pattern for a logic trigger.		
:LOGAnd *ch$,A$*	*ch$* = CH1 or CH2 *A$* = OR, AND	Sets AND/OR for the logic trigger pattern.	All	132
:LOGAnd? *ch$*	*ch$,A$*	Queries AND/OR for the logic trigger pattern.		
:EVENt *ch$,A*	*A* = 0, 2 to 4000 (0; OFF)	Sets count for event trigger.	MEM FFT	132
:EVENt? *ch$*	*ch$,A*<NR1>	Queries count for event trigger.		
:EXTErnal *A$*	*A$* = OFF, ON	Enables and disables external trigger.	All	133
:EXTErnal?	*A$*	Queries external trigger enablement.		
:TIMEr *A$*	*A$* = OFF, ON	Enables and disables timer trigger.	All	133
:TIMEr?	*A$*	Queries timer trigger.		
:TMSTArt *month,day, hour,min*	*month* = 1 to 12 *day* = 1 to 31 *hour* = 0 to 23 *min* = 0 to 59	Sets start time of timer trigger.	All	134
:TMSTArt?	*month,day,hour,min* all <NR1>	Queries start time of timer trigger.		
:TMSTOp *month,day, hour,min*	Same as :TMSTArt	Sets stop time of timer trigger.	All	134
:TMSTOp?	Same as :TMSTArt	Queries stop time of timer trigger.		
:TMINTvl *hour,min,sec*	*hour* = 0 to 23 *min* = 0 to 59 *sec* = 0 to 59	Sets time interval for timer trigger.	All	135
:TMINTvl?	*hour,min,sec* all <NR1>	Queries time interval for timer trigger.		
:MODE *A$*	*A$* = SINGle, REPEat (REC) SINGle, REPEat, AUTO (MEM, FFT)	Sets trigger mode.	MEM REC FFT	135
:MODE?	*A$*	Queries trigger mode.		

MEM memory recorder function
REC recorder function
XYC XY recorder function
FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:TRIGger	(*ch$* = CH1 to CH4)			
:PRETrig *A*	*A* = 0, 2, 5, 10, ... 90, 95, 100, and -950 to -50 in 50 % steps.	Sets pre-trigger.	MEM FFT	136
:PRETrig?	*A*<NR1> (unit; %)	Queries pre-trigger.		
:TIMIng *A$*	*A$* = START, STOP, S_S	Sets trigger timing.	REC XYC	136
:TIMIng?	*A$*	Queries trigger timing.		
:TRGTime? (*A*)	*A* = block number in memory segmentation (0 to maximum number of blocks) *hour,min,sec* (all <NR1>)	Queries the currently set time point for trigger detection.	All	137
:TRGDate? (*A*)	*A* = block number in memory segmentation (0 to maximum number of blocks) *year,month,day* (all <NR1>)	Queries the currently set date for trigger detection.	All	137

4. UNIT command (Setting and querying input channel)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:UNIT	(*ch$* = CH1 to CH4)			
:RANGe *ch$,A*	*A* = voltage axis range (unit; volts)	Sets input channel voltage axis range.	All	138
:RANGe? *ch$*	*ch$,A*<NR3>	Queries input channel voltage axis range.		
:POSItion *ch$,A*	*A* = Position value (unit; %)	Sets the origin position for an input channel.	All	138
:POSItion? *ch$*	*ch$,A*<NR1>	Queries the origin position for an input channel.		
:COUPling *ch$,A$*	*A$* = GND, DC, AC	Sets input channel coupling.	All	139
:COUPling? *ch$*	*ch$,A$*	Queries input channel coupling.		
:FILTer *ch$,A$*	*A* = 0, 5.0E5, 5.0E2, 5 (0: OFF)	Sets input channel filter.	All	139
:FILTer? *ch$*	*ch$,A*<NR3>	Queries input channel filter.		
:ADJUST		Carries out zero adjustment.	All	139

MEM memory recorder function
REC recorder function
XYC XY recorder function
FFT FFT function
All MEM, REC, XYC, and FFT function

5. DISPlay command (Setting and querying changeover of the screen mode, waveform display)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:DISPlay	(*ch$* = CH1 to CH4) (*G$* = G1, G2)			
:CHANge *A$*	*A$* = SYSTem, STATus, TRIGger, DISPlay	Changes over the display screen.	All	140
:CHANge?	*A$*	Queries the display screen.		
:DRAWing *ch$,A$*	*A$* = OFF, LIGHt, DARK	Sets display and recording intensity for waveform.	MEM REC XYC	140
:DRAWing? *ch$*	*ch$,A$*	Queries display and recording of a waveform.		
:LOGDraw *ch$,A$*	*A$* = OFF, ON	Enables and disables display for a logic waveform.	MEM REC	141
:LOGDraw? *ch$*	*ch$,A$*	Queries display for a logic waveform.		
:PAGE *A*	*A* = 1 to 7 (system screen) *A* = 1, 2 (status screen)	Changes over the page of the screen.	All	141
:PAGE?	*A*<NR1>	Queries the page of the screen.		
:GRAPh *ch$,G$*		Sets waveform display graph in DUAL and QUAD format.	MEM REC	142
:GRAPh? *ch$*	*ch$,G$*	Queries waveform display graph in DUAL and QUAD format.		
:XMAG *A$*	(MEM) *A$* = X10, X5, X 2, X1, X1_2, X1_5, X1_10, X1_20, X1_50, X1_100, X1_200, X1_500, X1_1000, X1_2000, X1_4000 (REC) *A$* = X1, X1_2, X1_5, X1_10, X1_20, X1_50	Sets the magnification/ compression factor on the time axis.	MEM REC	142
:XMAG?	*A$*	Queries the magnification/ compression factor on the time axis.		
:ZOOM *A$*	*A$* = OFF, ON	Enables and disables the zoom function.	MEM	143
:ZOOM?	*A$*	Queries the zoom function enablement.		
:ZOOMMag *A$*	*A$* is same as (MEM) of XMAG	Sets the zoom magnification.	MEM	143
:ZOOMMag?	*A$*	Queries the zoom magnification.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:DISPlay	(*ch$* = CH1 to CH4) (*G$* = G1, G2)			
:YMAG *ch$,A$*	*A$* = X1_2, X1, X2, X5, X10	Sets the magnification/ compression factor on the voltage axis.	MEM REC	144
:YMAG? *ch$*	*ch$,A$*	Queries the magnification/ compression factor on the voltage axis.		
:YZOOm *ch$,A$*	*A* = 0 to 100 %	Sets the waveform display position on the voltage axis.	MEM REC	144
:YZOOm? *ch$*	*ch$,A*<NR1>	Queries the waveform display position on the voltage axis.		
:WAVE *A$*	*A$* = ACUR (A cursor), TRIG (trigger point), POINT (the point set with :MEMory:POINt)	Executes waveform display.	MEM	145
:DIVMap *A$*	*A$* = OFF, ON	Enables and disables the memory segmentation screen.	MEM	145
:DIVMap?	*A$*	Queries the memory segmentation screen enablement.		
:CALCEdit *A$*	*A$* = OFF, ON	Enables and disables the waveform calculation processing screen.	MEM	146
:CALCEdit?	*A$*	Queries enablement of the waveform calculation processing screen.		
:MEASEdit *A$*	*A$* = OFF, ON	Enables and disables the waveform parameter calculation screen.	MEM	146
:MEASEdit?	*A$*	Queries enablement of the waveform parameter calculation screen.		
:XAXIs *ch$*		In XY format, sets the X axis.	MEM XYC	147
:XAXIs?	*ch$*	In XY format, queries the X axis.		
:XYCLr *A$*	*A$* = OFF, ON	Enables and disables the display clear function.	XYC	147
:XYCLr?	*A$*	Queries enablement of the display clear function.		
:FFTCH *G$,ch$*	*G$* = G1, G2	Sets the FFT analysis channel for the designated graph.	FFT	148
:FFTCH? *G$*	*G$,ch$*	Queries the FFT analysis channel for the designated graph.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

6. CURSor command (Cursor setting and reading)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CURSor	(*ch$* = CH1 to CH4)			
:MODE *A$*	*A$* = OFF, TIME, VOLT (REC) OFF, TIME, VOLT, TRACe (except XY format in MEM) OFF, Xcur, Ycur (XY format in MEM, XYC) OFF, ON (FFT)	Sets the A and B cursor type.	All	149
:MODE?	*A$*	Queries the A and B cursor type.		
:ABCUrsor *A$*	*A$* = A, A_B	Chooses between the A and the A&B cursors.	MEM REC XYC	149
:ABCUrsor?	*A$*	Queries between the A and the A&B cursors.		
:ACHAnnel *ch$*		Sets the A cursor channel.	MEM REC XYC	150
:ACHAnnel?	*ch$*	Queries the A cursor channel.		
:BCHAnnel *ch$*		Sets the B cursor channel.	MEM REC XYC	150
:BCHAnnel?	*ch$*	Queries the B cursor channel.		
:YDISp *A$*	*A$* = PEAK, RMS	Sets the display method of the FFT voltage value.	FFT	151
:YDISp?	*A$*	Queries the display method of the FFT voltage value.		
:APOSition *A*	(vertical cursor, trace cursor) *A* = 0 to amount of stored data (MEM, REC) 0 to 400 (XYC) (horizontal cursor) *A* = 0 to 250	Sets the position of the A cursor.	MEM REC XYC	151
:APOSition?	*A*<NR1>	Queries the position of the A cursor.		
:BPOSition *A*	Same as :APOSition	Sets the position of the B cursor.	MEM REC XYC	153
:BPOSition?	*A*<NR1>	Queries the position of the B cursor.		
:FPOSition *A*	*A* = 0 to 799 (STORAGE) 0 to 399 (except STORAGE)	Sets the position of the FFT cursor.	FFT	153
:FPOSition?	*A*<NR1>	Queries the position of the FFT cursor.		

MEM memory recorder function
REC recorder function
XYC XY recorder function
FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CURSor	(*ch$* = CH1 to CH4)			
:DTREad?	*A$,B$* *A$* = readout value (t, Δt) *B$* = readout value (1/t, 1/Δt, V, ΔV)	Queries the cursor readout value (t).	MEM REC	154
:DVREad?	*A$* = readout value (V, ΔV)	Queries the cursor readout value (V).	MEM REC XYC	154
:FFTRead?	*A$,B$* *A$* = x-axis readout value. *B$* = y-axis readout value.	Queries the FFT cursor readout value.	FFT	154

7. MEMory command (Setting and querying input and output, etc., from the memory)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:MEMory	(*ch$* = CH1 to CH4)			
:POINt *ch$,A*	*A* = 0 to recording length × 40 (2000000 max.)	Sets point in memory for input and output.	MEM	155
:POINt?	*ch$,A*<NR1>	Queries point in memory for input and output.		
:MAXPoint?	*A*<NR1> = 0 (not stored) 600 to 2000000 (÷ 40 = number of divisions)	Queries the amount of data stored.	MEM	155
:ADATa *B,C,...*	*B,C,...* = -48 to 4047	Input data to memory (ASCII).	MEM	156
:ADATa? *A*	*A* = 1 to 40 (number of output units) *B,C,...*<NR1> = -48 to 4047	Output data from memory (ASCII).		
:VDATa *B,C,...*	*B,C,...* = voltage values (unit; V)	Input data to memory (voltage values).	MEM	157
:VDATa? *A*	*A* = 1 to 10 (amount of data) *B,C,...*<NR3> = voltage value (unit; V)	Output stored data (voltage values).		
:LDATa *B,C,...*	*B,C,...* = 0 to 15	Input logic data to memory.	MEM	158
:LDATa? *A*	*A* = 1 to 40 (amount of output data) Response data <NR1> = 0 to 15	Output logic data from memory.		
:AREAI? *ch$*	*A*<NR1> = -48 to 4047	Real time data output (ASCII)	MEM	159
:VREAI? *ch$*	*A*<NR3> = voltage value (units V)	Real time data output (voltage value)	MEM	159

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:MEMory	(*ch$* = CH1 to CH4)			
:LREAl? *ch$*	*A*<NR1> = 0 to 15	Logic real time data output	MEM	159
:BDATa? *A*	*A* = 1 to 125 (amount of output data) Response data, binary, integer data	Performs binary transfer for stored data.	MEM	160
:BREAl? *ch$*	Response data, binary, integer data	Real time data output (binary)	MEM	160
:FFTPOint *A*	*A* = 0 to 799 (STORAGE) 0 to 399 (except STORAGE)	Sets the output point for FFT data.	FFT	161
:FFTPOint?	*A*<NR1>	Queries the current output point for FFT data.		
:FFTData?	*A*<NR3> *A* = y-axis data	Output FFT data.	FFT	161

8. SYSTem command (Setting and querying the system screen)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:SYSTem				
:TIME *hour, min,sec*	*hour* = 0 to 23 *min* = 0 to 59 *sec* = 0 to 59	Sets the time.	All	162
:TIME?	*hour,min,sec* (all <NR1>)	Queries the current time.		
:DATE *year,month,day*	*year* = 0 to 99 *month* = 1 to 12 *day* = 1 to 31	Sets the calendar.	All	162
:DATE?	*year,month,day* (all <NR1>)	Queries the calendar.		
:DATAClear		Clear data.	All	162
:CRTOff *A$*	*A$* = ON, OFF	Enables and disables the screen saver.	All	163
:CRTOff?	*A$*	Queries enablement of the screen saver.		
:GRID *A$*	*A$* = OFF, NORMal, FINE	Sets the grid type.	All	163
:GRID?	*A$*	Queries the grid type.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:SYSTem	(*ch$* = CH1 to CH4)			
:STARt *A$*	*A$* = ON, OFF	Enables and disables start key backup.	All	163
:STARt?	*A$*	Queries start key backup enablement.		
:CHMArk *A$*	*A$* = ON, OFF	Enables and disables channel markers.	All	164
:CHMArk?	*A$*	Queries enablement of channel markers.		
:BEEPer *A$*	*A$* = ON, OFF	Enables and disables the beep sound.	All	164
:BEEPer?	*A$*	Queries beep sound enablement.		
:LIST *A$*	*A$* = OFF, LIST, GAUGE, L_G	Sets list and gauge functions.	All	164
:LIST?	*A$*	Queries list and gauge functions.		
:USECH *A*	*A* = 1, 2, 4	Sets number of channels used.	All	165
:USECH?	*A*<NR1>	Queries number of channels used.		
:LOGDraw *A$*	*A$* = DARK, LIGHt	Sets the logic waveform's dark / light setting.	All	165
:LOGDraw?	*A$*	Queries the logic waveform's dark / light setting.		
:COPYSize *A$*	*A$* = LARGE, SMALL	Sets the CRT copy size.	All	165
:COPYSize?	*A$*	Queries the CRT copy size.		
:SCSI *A$,B*	*A$* = 8853, SCSI *B* = 0 to 7	Sets the SCSI interface device address ID.	All	166
:SCSI? *A$*	*A$,B*<NR1>	Queries the SCSI interface device address ID.		
:COPYPlot *A$*	*A$* = PRINter, PLOTter, FD, SCSI	Sets the CRT copy output device.	All	166
:COPYPlot?	*A$*	Queries the CRT copy output device.		
:PEN *A$,B*	*A$* = AREA, FRAME, CHAR CH1 to CH4 *B* = 0 to 8 (0; OFF)	Sets the plotter pen.	All	167
:PEN? *A$*	*A$,B*<NR1>	Queries the plotter pen.		

MEM memory recorder function
REC recorder function
XYC XY recorder function
FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:SYSTem	(*ch$* = CH1 to CH4)			
:BMPKind *A$*	*A$* = MONO, COLOR	Sets the type of CRT copy file output.	All	167
:BMPKind?	*A$*	Queries the type of CRT copy file output.		
:BMPColor *A$ to D$*	*A$ to D$* = BLACK, BLUE, RED, MAGENTA, GREEN, CYAN, YELLOW, ORANGE	Sets the color of CRT copy file output.	All	168
:BMPColor?	*A$ to D$*	Queries the color of CRT copy file output.		
:DISKMode *A$*	*A$* = FD, SCSI, FD_SCSI	Sets the FD key.	All	168
:DISKMode?	*A$*	Queries the FD key.		

9. SCALing command (Setting and querying scaling)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:SCALing	(*ch$* = CH1 to CH4)			
:SET *ch$,A$*	*A$* = OFF, SCI, ENG	Enables and disables scaling.	All	169
:SET? *ch$*	*ch$,A$*	Queries scaling enablement.		
:VOLT *ch$,A*	*A* = -9.999E+9 to 9.999E+9	Sets the scaling conversion value.	All	169
:VOLT? *ch$*	*ch$,A*<NR3>	Queries the scaling conversion value.		
:OFFSet *ch$,A*	*A* = -9.999E+9 to +9.999E+9	Sets scaling offset.	All	170
:OFFSet? *ch$*	*ch$,A*<NR3>	Queries scaling offset.		
:UNIT *ch$,'A$'*	*A$* = scaling unit (7 characters)	Sets scaling unit.	All	170
:UNIT? *ch$*	*ch$,"A$"*	Queries scaling unit.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

10. COMMent command (Setting and querying comments)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:COMMent	(*ch$* = CH1 to CH4)			
:TITLe *A$,'B$'*	*A$* = ON, OFF *B$* = comment string (up to 20 characters)	Sets a title comment.	All	171
:TITLe?	*A$,"B$"*	Queries a title comment.		
:CH *ch$,A$,'B$'*	*ch$* = CH1 to CH4, CHA to CHD *A$* = ON, OFF *B$* = comment string (up to 20 characters)	Sets a comment for a particular channel.	All	172
:CH? *ch$*	*ch$,A$,"B$"*	Queries comment for a particular channel.		

11. CALCulate command (Calculation setting and querying)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CALCulate	(*ch$* = CH1 to CH4)			
:WVCALc *A$*	*A$* = ON, OFF, EXEC (execute)	Enables and disables waveform processing calculation.	MEM	173
:WVCALc?	*A$*	Queries enablement of waveform processing calculation.		
:Z1 *A$,B$,C$, D$*	*A$, B$, C$* = A to P *D$* = PLUS, MINUs, MULTi, DIVI	Sets the coefficients for the waveform processing calculation equation for Z1.	MEM	173
:Z1?	*A$,B$,C$,D$*	Queries the coefficients for the waveform processing calculation equation for Z1.		
:Z2 *A$,B$,C$, D$*	*A$, B$, C$* = A to P *D$* = PLUS, MINUs, MULTi, DIVI	Sets the coefficients for the waveform processing calculation equation for Z2.	MEM	174
:Z2?	*A$,B$,C$,D$*	Queries the coefficients for the waveform processing calculation equation for Z2.		
:Z3 *A$,B$,C$, D$*	*A$, B$, C$* = A to P *D$* = PLUS, MINUs, MULTi, DIVI	Sets the coefficients for the waveform processing calculation equation for Z3.	MEM	174
:Z3?	*A$,B$,C$,D$*	Queries the coefficients for the waveform processing calculation equation for Z3.		

MEM memory recorder function
REC recorder function
XYC XY recorder function
FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CALCulate	(*ch$* = CH1 to CH4)			
:Z4 *A$,B$,C$, D$*	*A$, B$, C$* = A to P *D$* = PLUS, MINUs, MULTi, DIVI	Sets the coefficients for the waveform processing calculation equation for Z4.	MEM	174
:Z4?	*A$,B$,C$,D$*	Queries the coefficients for the waveform processing calculation equation for Z4.		
:X1 *A$,ch$,B$*	*A$* = OFF (*ch$, B$* are disregarded) PAR, ABS, EXP, LOG, SQR, MOV, SLI, DIF, INT, DIF2, INT2 *B$* = A to P (when *A$* = MOV, a value from 1 to 4000, when *A$* = SLI, a value from -4000 to 4000)	Sets calculation equation for X1.	MEM	175
:X1?	*A$,ch$,B$*	Queries calculation equation for X1.		
:X2 *A$,ch$,B$*	Same as X1 (*ch$* = CH1 to CH4, Z1)	Sets calculation equation for X2.	MEM	176
:X2?	*A$,ch$,B$*	Queries calculation equation for X2.		
:X3 *A$,ch$,B$*	Same as X1 (*ch$* = CH1 to CH4, Z1, Z2)	Sets calculation equation for X3.	MEM	176
:X3?	*A$,ch$,B$*	Queries calculation equation for X3.		
:X4 *A$,ch$,B$*	Same as X1 (*ch$* = CH1 to CH4, Z1 to Z3)	Sets calculation equation for X4.	MEM	177
:X4?	*A$,ch$,B$*	Queries calculation equation for X4.		
:Y1 *A$,ch$,B$*	Same as X1 (*ch$* = CH1 to CH4)	Sets calculation equation for Y1.	MEM	178
:Y1?	*A$,ch$,B$*	Queries calculation equation for Y1.		
:Y2 *A$,ch$,B$*	Same as X1 (*ch$* = CH1 to CH4, Z1)	Sets calculation equation for Y2.	MEM	179
:Y2?	*A$,ch$,B$*	Queries calculation equation for Y2.		
:Y3 *A$,ch$,B$*	Same as X1 (*ch$* = CH1 to CH4, Z1, Z2)	Sets calculation equation for Y3.	MEM	179
:Y3?	*A$,ch$,B$*	Queries calculation equation for Y3.		

MEM memory recorder function
REC recorder function
XYC XY recorder function
FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CALCulate	(*ch$* = CH1 to CH4)			
:Y4 *A$,ch$,B$*	Same as X1 (*ch$* = CH1 to CH4, Z1 to Z3)	Sets calculation equation for Y4.	MEM	180
:Y4?	*A$,ch$,B$*	Queries calculation equation for Y4.		
:FACTor *A$,B*	*A$* = A to P *B* = -9.999E+9 to +9.999E+9	Sets the value of calculation equation coefficient a to p.	MEM	180
:FACTor? *A$*	*A$,B*<NR3>	Queries the value of calculation equation coefficient a to p.		
:Z1DIsplay *ch$,A$,upper, lower*	*ch$* = CH1 to CH4, NONE *A$* = AUTO, MANUal (for MANUal) *upper, lower* = -9.999E+9 to +9.999E+9 (units V)	Sets the channel for receipt of the calculated result of the waveform processing calculation equation for Z1.	MEM	181
:Z1DIsplay?	*ch$,A$,upper,lower*	Queries the channel for receipt of the calculated result of the waveform processing calculation equation for Z1.		
:Z2DIsplay *ch$,A$,upper, lower*	Same as Z1DIsplay	Sets the channel for receipt of the calculated result of the waveform processing calculation equation for Z2.	MEM	181
:Z2DIsplay?	*ch$,A$,upper,lower*	Queries the channel for receipt of the calculated result of the waveform processing calculation equation for Z2.		
:Z3DIsplay *ch$,A$,upper, lower*	Same as Z1DIsplay	Sets the channel for receipt of the calculated result of the waveform processing calculation equation for Z3.	MEM	182
:Z3DIsplay?	*ch$,A$,upper,lower*	Queries the channel for receipt of the calculated result of the waveform processing calculation equation for Z3.		
:Z4DIsplay *ch$,A$,upper, lower*	Same as Z1DIsplay	Sets the channel for receipt of the calculated result of the waveform processing calculation equation for Z4.	MEM	182
:Z4DIsplay?	*ch$,A$,upper,lower*	Queries the channel for receipt of the calculated result of the waveform processing calculation equation for Z4.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

Command	Data (for a query, response data)	Explanation	Function	Ref page
:CALCulate	(*ch$* = CH1 to CH4)			
:MEASure *A$*	*A$* = ON, OFF, EXEC (execute)	Enables and disables waveform parameter calculation.	MEM	183
:MEASure?	*A$*	Queries enablement of waveform parameter calculation.		
:MEASPrint *A$*	*A$* = OFF, PRINTer, FD, SCSI	Sets printing of waveform parameter calculation values output device.	MEM	183
:MEASPrint?	*A$*	Queries printing of waveform parameter calculation values output device.		
:MEASSet *NO$,A$,ch$*	*A$* = OFF, AVE, RMS, PP, MAX, MAXT, MIN, MINT, AREA, PERI, FREQ, RISE, FALL, XYAREA *ch$* = ALL, CH1 to CH4	Sets waveform parameter calculation.	MEM	184
:MEASSet? *NO$*	*NO$,A$,ch$*	Queries waveform parameter calculation.		
:ANSWer? *NO$,ch$*	*A$* = OFF, AVE, RMS, PP, MAX, MAXT, MIN, MINT, AREA, PERI, FREQ, RISE, FALL, XYAREA *B*<NR3> = calculation result NONE, 0 (when there is no calculation result.)	Queries a waveform parameter calculation result.	MEM	185
:COMP *NO$,A$*	*A$* = ON, OFF	Enables or disables waveform parameter decision.	MEM	185
:COMP? *NO$*	*NO$,A$*	Queries enablement of waveform parameter decision.		
:COMPArea *NO$,up,low*	*up, low* = -9.999E+9 to +9.999E+9	Sets upper limit and lower limit values for waveform parameter decision.	MEM	186
:COMPArea? *NO$*	*NO$,up*<NR3>,*low*<NR3>	Queries upper limit and lower limit values for waveform parameter decision.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

12. DISK command (Commands relating to the floppy disk drive, the hard disk drive and the magneto optical disk)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:DISK	(*ch$* = CH1 to CH4)			
:MODE *A$*	*A$* = OFF, FD, SCSI	Enables or disables the floppy disk (FD) screen or small computer system interface (SCSI) screen..	All	187
:MODE?	*A$*	Queries enablement of the FD screen or the SCSI screen.		
:SAVE '*NAME1$*.*NAME2$*',*A$*,*B$*	*NAME1$* = file name (up to 8 characters) *NAME2$* = file extension (up to 3 characters) *A$* = type of data to save Wave: measurement data (MEM, FFT) Func: conditions of creation Area: waveform decision area (MEM, FFT) (when *A$* = Wave in MEM, FFT) *B$* = channels to save ALL, CH1 to CH4	Saves a file.	All	187
:LOAD *NO* (,ch$)	*NO* = file number	Loads a file.	All	188
:DELEte *NO*	*NO* = file number	Deletes a file.	All	188
:FORMat (*A$*)	*A$* = 2HD, 2HC (Effective with *A$* when FD is 2HD.)	Formats a FD, a HD or a MO.	All	188
:MKDIR '*NAME$*	*NAME$* = directory name (up to 12 characters)	Creates a directory.	All	189
:CHDIR *NO*	*NO* = file number	Changes the current directory.	All	189
:DIR?	*A$* = directory name	Queries the current directory.	All	189

MEM	memory recorder function	REC	recorder function
XYC	XY recorder function	FFT	FFT function
All	MEM, REC, XYC, and FFT function		

Command	Data (for a query, response data)	Explanation	Function	Ref page
:DISK	(ch$ = CH1 to CH4)			
:INFOr? *'NAME$*	[In the file] *"NAME$",A,B$,"DATE$","TIME$",C* [In the directory] *"NAME$",A,"DATE$","TIME$"* *NAME$* = file name *A* = file number (if no file exists, then -1) *B$* = type of data saved WAVE: measurement data FUNC: conditions of creation AREA: waveform decision area N: no such file *DATE$* = year/month/day of save *TIME$* = hour:min:sec of save *C* = file size	Queries the file information.	All	190
:NINFor? *NO*	*NO,"NAME$"* *NO* = file number *NAME$* = file name	Queries filename.	All	190
:FILE?	*A*<NR1> = number of files	Queries number of files.	All	191
:FREE?	*A$* = allowable number of clusters	Queries the allowable number of clusters.	All	191

MEM	memory recorder function	REC	recorder function
XYC	XY recorder function	FFT	FFT function
All	MEM, REC, XYC, and FFT function		

13. GRAPh command (Commands relating to the graphics editor)

Command	Data (for a query, response data)	Explanation	Function	Ref page
:GRAPh				
:EDIT *A$*	*A$* = OFF, ON	Enables and disables the editor.	MEM FFT	192
:EDIT?	*A$*	Queries editor enablement.		
:STORage		Loads a waveform into the editor.	MEM FFT	192
:PARAllel *high,low,right, left*	*high* = 0 to 9.96 (div) *low* = 0 to 9.96 (div) *right* = 0 to 14.975 (div) *left* = 0 to 14.975 (div)	Carries out a parallel movement of the drawing.	MEM FFT	192
:LINE *X1,Y1, X2,Y2*	*X1,X2* = x-coordinates *Y1,Y2* = y-coordinates	Draws a line from (X1,Y1) to (X2,Y2).	MEM FFT	193
:PAINT *X,Y*	*X* = x-coordinate *Y* = y-coordinate	Paints the enclosed plane surrounding the point specified by (X,Y).	MEM FFT	194
:REVErse		Reverses black-and-white.	MEM FFT	194
:ERASe *X1,Y1, X2,Y2*	*X1,X2* = x-coordinates *Y1,Y2* = y-coordinates	Erases from (X1,Y1) to (X2,Y2).	MEM FFT	194
:CLEAr *X1,Y1, X2,Y2*	*X1,X2* = x-coordinates *Y1,Y2* = y-coordinates all <NR1>	Clears the rectangle with the points (X1,Y1) and (X2,Y2) at diagonally opposite corners.	MEM FFT	195
:ALLClear		Clears the entire drawing.	MEM FFT	195
:UNDO		Reverses the effect of the immediately previous editor command.	MEM FFT	195
:SAVE		Saves the decision area created with the editor.	MEM FFT	195
:POINt *X,Y,A*	*X* = x-coordinates, *Y* = y-coordinates *A* = 0, 1	Sets waveform decision area data.	MEM FFT	196
:POINt? *X,Y*	*X,Y,A* all <NR1>	Queries waveform decision area data.		

MEM memory recorder function REC recorder function
XYC XY recorder function FFT FFT function
All MEM, REC, XYC, and FFT function

Chapter 9
Command Reference

9.1 Command Reference

- The following sections describe the format and functions of individual commands.
- The following is an example of how the descriptions are organized. (Describes items of ① to ⑤ in the figure below.)

Example

①→ ■ Changes and queries the function selection.

②→ **Syntax**

command	:FUNCtion *A$*
query	:FUNCtion?
response	*A$* = MEM : memory recorder function REC : recorder function XYC : XY recorder function FFT : FFT function

③→ **Explanation**

- Switches to the function designated by *A$*.
- Returns the name of the current function as character data.

④→ **Example**

```
:FUNCTION MEM
```

The function is set to the memory recorder function.

⑤→ **When allowed** In all functions.

① Command function

② Command syntax

- command the syntax of a command program message
- query the syntax of a query program message
- response the format of the response message
- parameters
  - *A, B, C,...* Numerical data (e.g. 1.5, 10E-3)
  - *A$, B$, ...* Character data (e.g. A, B1, GND, OFF)
  - *"A", "A$",...* Character string data (e.g. "1.5", "mA")

  [Single quotation marks (') can be used instead of double quotation marks (") .]

The format of numerical data follows the formats <NR1>, <NR2>, and <NR3>, described (8) in Section 3.3.

Example
A<NR1> Numerical parameter in NR1 format
B<NR2> Numerical parameter in NR2 format
C Numerical parameter in NRf format

If no format is mentioned, <NRf> format (i.e. any of the above) is accepted.

NR format
NR1 format: integer data (examples: +15, -5, 10)
NR2 format: fixed point numbers (examples: +1.234, -12.12, 6.8)
NR3 format: floating point numbers (examples: +1.0E-3, -2.3E+4)
The term "NRf format" includes all these three formats.

- When the 8853 is receiving a command or query program message, it accepts format.
- When it is sending it utilizes whichever one of the formats <NR1> to <NR3> is indicated in the particular command.
- Response messages may or may not have headers prefixed, according to the setting made by the :HEADER command.

③ Explanation of the command function

④ Example of command use

⑤ This lists the functions in which the command may be used.

MEM memory recorder function
REC recorder function
XYC X-Y recorder function
FFT FFT function
All Any of the MEM, REC, XYC and FFT functions

Execution of commands

- Commands are input into the input buffer and are executed in order.
- However the :ABORT command is executed immediately, even if commands are waiting in the input buffer - more precisely, at the instant its terminator is received.
- Commands other than those which can be handled by the 8853 in its current state are not executed but generate execution errors. (This happens, for example, when in recorder function it is attempted to execute an memory recorder function setting.
- Almost all commands cannot be executed during measurement operation.

9.2 Standard Commands Stipulated by IEEE 488.2

A. System data commands and queries

1. *IDN? query

■ Queries device ID (Identification code)

Syntax

query	*IDN?
response	HIOKI,8853,0,V2.00

① ② ③ ④

Explanation

① First field	Manufacturer's name
② Second field	Model name
③ Third field	Serial number (not used: 0)
④ Fourth field	Software version

2. *OPT? query

■ Queries device option provision.

Syntax

query	*OPT?
response	A<NR1>,B<NR1>,C<NR1>,D<NR1>

Explanation

Whether or not input unit present is returned as an NR1 numerical value.

A : whether or not channel 1 input unit present.
B : whether or not channel 2 input unit present.
C : whether or not channel 3 input unit present.
D : whether or not channel 4 input unit present.

0 = no present
1 = input unit present

B. Internal operation commands and queries

1. *RST command

■ Device initial setting.

Syntax command *RST

Explanation
- Initializes the 8853 (same as system reset).
- It does not clear GP-IB related items (the event registers, the enable registers, the input buffer and the output queue).

2. *TST? query

■ Queries the result of the self-test.

Syntax query *TST?
response *A*<NR1>
A = 0 : normal, 1: failure

Explanation The result of the self-test of the 8853 is returned as an NR1 numerical value.

C. Synchronous commands and queries

1. *OPC command

■ After all action has been completed during execution, sets the LSB (bit 0) of SESR (the standard event status register).

Syntax command *OPC

Explanation When the command preceding the *OPC command completes execution, the LSB of SESR is set.

Example *A$;B$;*OPC;C$*
After the execution of the command *A$* and *B$* is completed, the LSB of SESR is set.

2. *OPC? query

■ After execution is completed, replies with ASCII [1].

Syntax

query	*OPC?
response	1

Explanation When the command preceding the *OPC command completes execution, the response of ASCII [1] is made.

3. *WAI command

■ After all execution is completed, subsequently performs the following command.

Syntax

command	*WAI

Example

```
A$;B$;*WAI;C$
```

The command following *WAI is not executed until the execution of the command *A$* and *B$* is completed.

D. Status and event control commands and queries

1. *CLS command

■ Clears the status byte and associated queues (except for the output queue).

Syntax

command	*CLS

Explanation

- This instruction clears the event register associated with each bit of the status byte register.
- It also clears the status byte register.
- Because it does not clear the output queue, it has no effect upon bit 4 (MAV) of the status byte.

2. *ESE command

■ Writes the standard event status enable register (SESER).

Syntax

command *ESE *A*

A = 0 to 255

Explanation

- Sets the mask pattern of SESER to a value in the range 0 to 255.
- Outside this range, an execution error occurs. The initial value (when the power is turned on) is 0.

Example

*ESE 36

Bit 5 and bit 2 of SESER are set.

3. *ESE? query

■ Reads the standard event status enable register (SESER).

Syntax

query *ESE?

response *A*<NR1>

A = 0 to 255

Explanation

The contents of SESER as set by the *ESE command are returned as an integral value in the range 0 to 255.

4. *ESR? query

■ Reads out and clears the contents of the standard event status register (SESR).

Syntax

query *ESR?

response *A*<NR1>

Explanation

The contents of SESR are returned as an NR1 numerical value.

5. *SRE command

■ Writes the service request enable register (SRER).

Syntax

command *SRE *A*

A = 0 to 255

Explanation

- Sets the mark pattern of SRER to a value in the range of 0 to 255.
- Outside this range, an execution error occurs.
- The value of bit 6 is disregarded. The initial value (when the power is turned on) is 0.

Example

*SRE 33

Bits 5 and 0 of SRER are set.

6. *SRE? query

■ Reads the service request enable register (SRER).

Syntax

query *SRE?

response *A*<NR1>

A = 0 to 63, 128 to 191

Explanation

- The contents of SRER as set by the *SRE command are returned as an NR1 numerical value in the range 0 to 63, 128 to 191.
- Bit 6 is always 0.

7. *STB? query

■ Reads the status byte and MSS bit, without performing serial polling.

Syntax

query *STB?

response *A*<*NR*1>

A = 0 to 255

Explanation

- This is the same as reading out the status byte with serial polling.
- Bit 6 is not RQS, but is MSS.

8. :ESE0 command

■ Writes event status enable register 0 (ESER0).
(Commands specific to the 8853)

Syntax

command :ESE0 *A*
A = 0 to 255

Explanation

- Sets the mask pattern of ESER0 to a value in the range of 0 to 255.
- Outside this range, an execution error occurs. The initial value (when the power is turned on) is 0.

Example

```
:ESE0 36
```

This sets bit 5 and bit 2 of ESER0.

9. :ESE0? query

■ Reads event status enable register 0 (ESER0).
(Commands specific to the 8853)

Syntax

query :ESE0?
response *A*<NR1>
A = 0 to 255

Explanation

The contents of ESER0 are returned as an NR1 numerical value.

10. :ESR0? query

■ Reads event status register 0 (ESR0).
(Commands specific to the 8853)

Syntax

query :ESR0?
response *A*<NR1>
A = 0 to 255

Explanation

The contents of ESR0 are returned as an NR1 numerical value, and ESR0 is cleared.

9.3 Commands Specific to the 8853

1. Execution control commands (common to all functions)

■ Performs starting.

Syntax command :STARt

Explanation
- Same as the START key of the 8853.
- Starts waveform sampling operation.

When allowed In all functions.

■ Performs stopping.

Syntax command :STOP

Explanation
- Same as the STOP key of the 8853.
- Terminates at the instant that waveform sampling operation is completed.
- With the :STOP command, printer operation is not stopped. (Use the :ABORT command to stop operation.)

When allowed In all functions.

■ Aborts processing.

Syntax command :ABORT

Explanation
- Same as the STOP key of the 8853.
- Forced halt. Terminates even if waveform sampling operation is not yet completed.
- Also stops printer operation.

When allowed In all functions.

■ Performs printing.

Syntax command :PRINt

Explanation Same as the PRINT key of the 8853.

When allowed In all functions.

■ Screen dump function. (Hard copy of CRT)

Syntax command :HCOPy

Explanation
- Same as the COPY key of the 8853.
- Executes a hard copy of the screen.

When allowed In all functions.

■ Feeds printer paper.

Syntax command :FEED *A*

A = 1 to 255

Explanation Feeds the paper by a distance from 1 to 255 in millimeters determined by the numerical value.

When allowed In all functions.

■ Performs automatic range setting.

Syntax command :AUTO

Explanation
- Same as the AUTO key of the 8853.
- Sets the time axis range and the voltage axis range automatically and executes measurement.

When allowed In the memory recorder function.

■ Queries the 8853 error number.

Syntax query :ERRor?

response *A*<NR1>

A = error no.

Explanation
- The number of error or warning that has occurred on the 8853 is returned in <NR1> as a numerical value. (See to Appendix 1, in the 8853 instruction manual.)
- If an error occurs during execution of :ERROR? then the error number is cleared.

When allowed In all functions.

■ Enables and disables headers, and queries header enablement.

Syntax

command	:HEADer *A$*
query	:HEADer?
response	*A$*
	A$ = OFF, ON

Explanation

command	Sets header enablement. When headers are enabled, responses to queries are prefixed by headers; when headers are disabled, responses are not so prefixed.
query	Returns whether or not headers are prefixed to responses to queries. The initial toggle state for headers (when the power is turned on) is OFF.

Example ① When headers are disabled: response to :HEADER? is OFF.
② When headers are enabled: response to :HEADER? is :HEADER ON.

When allowed In all functions.

■ Changes and queries the function selection.

Syntax

command	:FUNCtion *A$*
query	:FUNCtion?
response	*A$*
	A$ = MEM : memory recorder function
	REC : recorder function
	XYC : XY recorder function
	FFT : FFT function

Explanation

command	Switches to the function designated by *A$*.
query	Returns the name of the current function as character data.

Example

```
:FUNCTION MEM
```

The function is set to the memory recorder function.

When allowed In all functions.

2. CONFigure command (Sets and queries time axis range, recording length, etc.)

■ Sets and queries the time/axis range.

Syntax

command	:CONFigure:TDIV *A*
query	:CONFigure:TDIV?
response	*A*<NR3>

Explanation

command	Sets the time axis range to a numerical value (unit seconds).
query	Returns the currently set value of the time axis range as an NR3 numerical value. (If an attempt is made to set the time per division to a non-permitted value, it will be set to the next range above that value.)

Example

```
:CONFIGURE:TDIV +500.0E-6
```

Sets the time axis range to 500 μ s.

When allowed In the memory recorder function and the recorder function.

■ Sets and queries the recording length.

Syntax

command	:CONFigure:SHOT *A*
query	:CONFigure:SHOT?
response	*A*<NR1>

Explanation

command	Sets the numerical value of the recording length (unit divisions).
query	Returns the currently set value of the recording length as an NR1 numerical value. 0 means CONT. (in the recorder function)

Example

```
:CONFIGURE: SHOT 15
```

Sets the recording length to 15 divisions.

When allowed In the memory recorder function and the recorder function.

■ Sets and queries the format.

Syntax

command	:CONFigure:FORMat *A$*
query	:CONFigure:FORMat?
response	*A$*
	A$ = SINGle, DUAL, QUAD, XY : MEM
	SINGle, DUAL, QUAD : REC
	SINGle, DUAL : FFT

Explanation

command	Sets the format.
query	Returns the current format as character data.

Example

```
:CONFIGURE:FORMAT SINGLE
```

Sets the format to SINGLE.

When allowed In the memory recorder function, the recorder function, and the FFT function.

■ Sets and queries the interpolation function.

Syntax

command	:CONFigure:DOTLine *A$*
query	:CONFigure:DOTLine?
response	*A$*
	A$ = DOT, LINE

Explanation

command	Sets the interpolation function (DOT or LINE).
query	Returns the currently setting of the interpolation as character data.

Example

```
:CONFIGURE:DOTLINE DOT
```

Sets the interpolation function to DOT.

When allowed In the memory recorder function, the XYC recorder function, and the FFT function.

CONFigure

■ Sets and queries the waveform superimposition function.

Syntax

command :CONFigure:OVWRite *A$*
query :CONFigure:OVWRite?
response *A$*
A$ = OFF, ON

Explanation

command: Enables and disables screen waveform superimposition.
query: Returns the current setting of the waveform superimposition enablement as character data.

Example

```
:CONFIGURE:OVWRITE ON
```
Sets the screen waveform superimposition to ON.

When allowed In the memory recorder function.

■ Sets and queries the auto print function.

Syntax

command :CONFigure:ATPRint *A$*
query :CONFigure:ATPRint?
response *A$*
A$ = OFF, ON

Explanation

command: Toggles the auto print function on and off.
query: Returns the current setting of the auto print function as character data.

Example

```
:CONFIGURE:ATPRINT ON
```
Sets the auto print function to ON.

When allowed In the memory recorder function and the FFT function.

■ Sets and queries the auto save function (output device).

Syntax

command :CONFigure:ATSAve *A$*
query :CONFigure:ATSAve?
response *A$*
A$ = OFF
FD : Auto save to the floppy disk
SCSI : Auto save to the SCSI (small computer system interface)

Explanation

command: Toggles the auto save function on and off (output device).
query: Returns the current setting of the auto save function as character data.

Example

```
:CONFIGURE:ATSAVE FD
```
Auto save to the floppy disk.

When allowed In the memory recorder function and the FFT function.

■ Enables and disables, and queries the smooth printing function.

Syntax

command	:CONFigure:SMOOth *A$*
query	:CONFigure:SMOOth?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables and disables the smooth printing function.
query	Returns the current enablement state of the smooth printing function as character data.

Example

```
:CONFIGURE:SMOOTH ON
```

Sets the smooth printing function to ON.

When allowed In the memory recorder function.

■ Enables and disables, and queries the roll mode function.

Syntax

command	:CONFigure:ROLL *A$*
query	:CONFigure:ROLL?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables and disables the roll mode function.
query	Returns the current enablement state of the roll mode function as character data.

Example

```
:CONFIGURE:ROLL ON
```

Sets the roll mode function to ON.

When allowed In the memory recorder function.

■ Sets and queries the count for averaging.

Syntax

command	:CONFigure:AVERage *A*
query	:CONFigure:AVERage?
response	*A*<NR1>
	A = 0, 2, 4, 8, 16, 32, 64, 128, 256 (0 : OFF)

Explanation

command	Sets the count for averaging.
query	Returns the current setting of the count for averaging as an NR1 numerical value. Setting 0 disables averaging. When averaging is disabled, the query returns OFF.

Example

```
:CONFIGURE:AVERAGE 32
```

Sets the count for averaging to 32.

When allowed In the memory recorder function and the FFT function.

CONFigure

■ Sets and queries memory segmentation.

Syntax

command	:CONFigure:MEMDiv *A$*
query	:CONFigure:MEMDiv?
response	*A$*

A$ = OFF
SEQ : sequential save
MULTI : multi-block

Explanation

command	Sets the of memory segmentation.
query	Returns the current setting for memory segmentation as character data.

Example

```
:CONFIGURE:MEMDIV SEQ
```

Sets the method of memory segmentation to sequential save.

When allowed In the memory recorder function.

■ Sets and queries the number of memory blocks. (divisions)

Syntax

command	:CONFigure:MAXBlock *A*
query	:CONFigure:MAXBlock?
response	*A*<NR1>

A = 2 to 63

Explanation

command	Sets the number of memory blocks (number of memory segmentations).
query	Returns the current number of memory blocks as an NR1 numerical value.

Example

```
:CONFIGURE:MAXBLOCK 15
```

Sets the number of memory blocks to 15.

When allowed In the memory recorder function, when the multi-block function is in use. Query is also possible, when the sequential save function is in use.

■ Sets and queries the start block. (start block)

Syntax

command	:CONFigure:STTBlock *A*
query	:CONFigure:STTBlock?
response	*A*<NR1>
	A = 1 to number of memory segmentations

Explanation

command	Sets the start block.
query	Returns the current start block as an NR1 numerical value.

Example

```
:CONFIGURE:STTBLOCK 1
```

Sets the start block to 1.

When allowed In the memory recorder, when the sequential save function is in use.

■ Sets and queries the end block. (end block)

Syntax

command	:CONFigure:ENDBlock *A*
query	:CONFigure:ENDBlock?
response	*A*<NR1>
	A = 1 to number of memory segmentations

Explanation

command	Sets the end block.
query	Returns the current end block as an NR1 numerical value.

Example

```
:CONFIGURE:ENDBLOCK 15
```

Sets the end block to 15.

When allowed In the memory recorder, when the sequential save function is in use.

■ Sets and queries the memory block used. (using block)

Syntax

command	:CONFigure:USEBlock *A*
query	:CONFigure:USEBlock?
response	*A*<NR1>
	A = 1 to number of segmentations

Explanation

command	Sets the using block.
query	Returns the currently used block as an NR1 numerical value.

Example

```
:CONFIGURE:USEBLOCK 15
```

Sets the block used to 15.

When allowed In the memory recorder function, when the memory segmentation function is in use.

CONFigure

■ Sets and queries the reference block. (ref block)

Syntax

command	:CONFigure:REFBlock *A*
query	:CONFigure:REFBlock?
response	*A*<NR1>

A = 0, 1 to number of memory segmentations (0: OFF)

Explanation

command	Sets the ref block.
query	Returns the current reference block as an NR1 numerical value.

Setting 0 disables averaging. When averaging is disabled, the query returns OFF.

Example

```
:CONFIGURE:REFBLOCK 15
```

Sets the reference block to 15.

When allowed In the memory recorder function, when the multi-block function is in use.

■ Sets and queries printer output.

Syntax

command	:CONFigure:PRINt *A$*
query	:CONFigure:PRINt?
response	*A$*

A$ = OFF, ON

Explanation

command	Sets the printer output.
query	Returns the currently set state of the printer output as character data.

Example

```
:CONFIGURE:PRINT ON
```

Sets the printer output to ON.

When allowed In the recorder function.

■ Sets and queries the waveform decision mode.

Syntax

command	:CONFigure:WVCOmp *A$*
query	:CONFigure:WVCOmp?
response	*A$*

A$ = OFF, OUT, ALLOut

Explanation

command	Sets the waveform decision mode.
query	Returns the current waveform decision mode as character data.

Example

```
:CONFIGURE:WVCOMP OUT
```

Sets the waveform decision mode to OUT.

When allowed In the memory recorder function and the FFT function.

■ Sets and queries the waveform decision stop mode.

Syntax

command	:CONFigure:CMPStop *A$*
query	:CONFigure:CMPStop?
response	*A$*
	A$ = GO, NG, G_N (G_N: GO&NG)

Explanation

command	Sets the stop mode during waveform decision.
query	Returns the current stop mode as character data.

Example

```
:CONFIGURE:CMPSTOP GO
```

Sets the stop mode during waveform decision to GO.

When allowed In the memory recorder function and the FFT function.

■ Sets and queries the FFT frequency range.

Syntax

command	:CONFigure:MAXFreq *A*
query	:CONFigure:MAXFreq?
response	*A*<NR3>

Explanation

command	Sets the frequency range as a numerical value (unit: Hz).
query	Returns the currently set frequency range as a numerical value in <NR3> format.
	If an attempt is made to set an unacceptable value, then the frequency range is set to the next higher value.

Example

```
:CONFIGURE:MAXFREQ 100
```

The frequency range is set to 100 Hz.

When allowed In the FFT function.

■ Sets and queries the FFT window function.

Syntax

command	:CONFigure:FFTWind *A$*
query	:CONFigure:FFTWind?
response	*A$*
	A$ = RECTan : rectangular window
	HANNing : hanning window

Explanation

command	Sets the window function as specified by *A$*.
query	Returns the current window function as character data.

Example

```
:CONFIGURE:FFTWIND HANNING
```

The window function is set to hanning window.

When allowed In the FFT function.

CONFigure

■ Sets and queries the FFT data source.

Syntax

command	:CONFigure:FFTRef *A$*
query	:CONFigure:FFTRef?
response	*A$*

A$ = NEW : new data
MEM : data stored in the memory recorder function

Explanation

command	Designates the source for FFT data as specified by *A$*.
query	Returns the current FFT data source as character data.

Example

```
:CONFIGURE:FFTREF NEW
```

New data is used as FFT data.

When allowed In the FFT function.

■ Sets and queries the FFT analysis mode.

Syntax

command	:CONFigure:FFTMode *G$,A$*
query	:CONFigure:FFTMode? *G$*
response	*G$,A$*

G$ = G1 : graph 1
G2 : graph 2
A$ = STR : storage waveform
LIN : linear spectrum
PSP : power spectrum

Explanation

command	Sets the FFT analysis mode. G2 can be designated even if the display format is SINGLE, but this does not affect the display.
query	Returns the current FFT analysis mode as character data.

Example

```
:CONFIGURE:FFTMODE G1,STORAGE
```

The FFT analysis mode for graph 1 is set to the stored waveform.

When allowed In the FFT function.

■ Sets and queries the y-axis (vertical axis) in the FFT analysis mode.

Syntax

command :CONFigure:FFTYaxis *G$,A$*
query :CONFigure:FFTYaxis? *G$*
response *G$,A$*
G$ = G1 : graph 1
G2 : graph 2
A$ = LINREal (*) : Linear-Real (linear real axis magnitude)
LINIMag (*) : Linear-Imag (linear imaginary axis magnitude)
LINMAg : Linear-Mag (linear magnitude)
LOGMAg : Log-Mag (logarithmic magnitude)
PHASE (*) : Phase (phase)
(volt) : in the FFT analysis mode; storage waveform (response only)

Explanation

command
Sets the y-axis for the displaying the FFT analysis result.
G2 can be designated even if the display format is SINGLE, but this does not affect the display.
When the analysis mode is stored waveform, the setting is not available.
The character data marked by (*) can be set only when the FFT analysis mode is Linear spectrum.

query
Returns the current y-axis (vertical axis) setting as character data.

Example

```
:CONFIGURE:FFTYAXIS G1,LINREAL
```

The setting for the y-axis of graph 1 is set to Linear-Real.

When allowed

In the FFT function (analysis mode; Linear spectrum or Power spectrum).

CONFigure

■ Sets and queries the x-axis (horizontal axis) in the FFT analysis mode.

Syntax

command: :CONFigure:FFTXaxis *G$,A$*
query: :CONFigure:FFTXaxis? *G$*
response: *G$,A$*
G$ = G1 : graph 1
G2 : graph 2
A$ = LINhz : Linear-Hz
LOGhz : Log-Hz
(time) : in the FFT analysis mode; storage waveform (response only)

Explanation

command: Sets the x-axis (horizontal axis) for the display of the FFT analysis result.
G2 can be designated even if the display format is SINGLE, but this does not affect the display.
When the analysis mode is the stored waveform, the setting is not available.
query: Returns the current x-axis setting as character data.

Example

```
:CONFIGURE:FFTXAXIS G1,LOGHz
```

The setting for the x-axis of graph 1 is set to Log-Hz.

When allowed

In the FFT function (analysis mode; Linear spectrum or Power spectrum).

■ Sets and queries the display scaling method for the FFT analysis result.

Syntax

command: :CONFigure:FFTSCale *G$,A$*
query: :CONFigure:FFTSCale? *G$*
response: *G$,A$*
G$ = G1 : graph 1
G2 : graph 2
A$ = AUTO : automatic setting
MANUal : manual setting

Explanation

command: Sets the display scaling method for the graph number designated by *G$*.
G2 can be designated even if the display format is SINGLE, but this does not affect the display.
query: Returns the current display scaling method for the graph number designated by *G$* as character data.

Example

```
:CONFIGURE:FFTSCALE G1,AUTO
```

The scaling method for graph number 1 is set to automatic.

When allowed

In the FFT function.

■ Sets and queries the FFT display scale vertical axis upper limit.

Syntax

command :CONFigure:FFTUp *G$,A*
query :CONFigure:FFTUp? *G$*
response *G$,A*<NR3>
G$ = G1 : graph 1
G2 : graph 2
A = -9.999E+9 to +9.999E+9

Explanation

command: Sets the FFT display scale vertical axis upper limit for the graph number designated by *G$* to the value designated by *A*. G2 can be designated even if the display format is SINGLE, but this does not affect the display.

query: Returns the current FFT display scale vertical axis upper limit for the graph number designated by *G$* as a numerical value in <NR3> format.

Example

```
:CONFIGURE:FFTUP G2,100
```

The FFT display scale vertical axis upper limit for graph 2 is set to 100.

When allowed In the FFT function.

■ Sets and queries the FFT display scale vertical axis lower limit.

Syntax

command :CONFigure:FFTLow *G$,A*
query :CONFigure:FFTLow? *G$*
response *G$,A*<NR3>
G$ = G1 : graph 1
G2 : graph 2
A = -9.999E+9 to +9.999E+9

Explanation

command: Sets the FFT display scale vertical axis lower limit for the graph number designated by *G$* to the value designated by *A*. G2 can be designated even if the display format is SINGLE, but this does not affect the display.

query: Returns the current FFT display scale vertical axis lower limit for the graph number designated by *G$* as a numerical value in <NR3> format.

Example

```
:CONFIGURE:FFTLOW G2,10
```

The FFT display scale vertical axis lower limit for graph 2 is set to 10.

When allowed In the FFT function.

CONFigure

■ Sets and queries the FFT data printer output style.

Syntax

command	:CONFigure:FFTPrint *A$*
query	:CONFigure:FFTPrint?
response	*A$*

A$ = WAVE : waveform data
DATA : numerical data

Explanation

command	Sets the printer output style to be waveform or logging (numerical data).
query	Returns the current setting of the printer output style as character data.

Example

```
:CONFIGURE:FFTPRINT WAVE
```

Sets the printer output style to be waveform.

When allowed In the FFT function.

■ Sets and queries the intermittent compression ratio of the FFT analysis data.

Syntax

command	:CONFigure:FFTThin *A$*
query	:CONFigure:FFTThin?
response	*A$*

A$ = X1 : ×1 (no intermittent compression)
X1_2 : ×1/2 compression
X1_5 : ×1/5 compression

Explanation

command	Sets the intermittent compression ratio of the FFT analysis data as character data.
query	Returns the current setting of the intermittent compression ratio as character data.

Example

```
:CONFIGURE:FFTTHIN X1_5
```

Sets the intermittent compression ratio to 1/5.

When allowed In the FFT function.

3. TRIGger command (Sets and queries trigger)

■ Sets and queries the trigger logical operator (AND/OR).

Syntax

command	:TRIGger:SOURce *A$*
query	:TRIGger:SOURce?
response	*A$*

A$ = OR, AND

Explanation

command	Sets the logical operator (AND/OR) for the internal, external and timer triggers.
query	Returns the current setting of the trigger logical operator (AND/OR) as character data.

Example

```
:TRIGGER:SOURce OR
```

Sets the trigger source to OR.

When allowed In all functions.

■ Sets and queries the type of trigger.

Syntax

command	:TRIGger:KIND *ch$,A$*
query	:TRIGger:KIND? *ch$*
response	*ch$,A$*

ch$ = CH 1 to CH4

A$ = OFF

LEVEl : level trigger
LOGIc : logic trigger
WINDow : window trigger
TIMEout : time out trigger
GLITch : glitch detection trigger

Explanation

command	Sets the type of trigger for the channel designated by *ch$*.
query	Returns as character data the type of the current trigger for the channel designated by *ch$*.

Example

```
:TRIGGER:KIND CH1,LEVEL
```

Sets channel 1 to level trigger.

When allowed In all functions.

TRIGger

■ Sets and queries trigger level.

Syntax

command	:TRIGger:LEVEl *ch$,A*
query	:TRIGger:LEVEl? *ch$*
response	*ch$,A*<NR1>
	ch$ = CH1 to CH4
	A = 0 to 100 (%)

Explanation

command	Sets the trigger level of the level, glitch detection, or time out trigger, of the channel designated by *ch$*.
query	Returns the current trigger level as an NR1 numerical value.

Example

```
:TRIGGER:LEVEL CH1,50
```

Sets the trigger level of channel 1 to 50%.

When allowed In all functions.

■ Sets and queries trigger direction (slope).

Syntax

command	:TRIGger:SLOPe *ch$,A$*
query	:TRIGger:SLOPe? *ch$*
response	*ch$,A$*
	ch$ = CH1 to CH4
	A$ = UP (rising : ⤴)
	DOWN (falling : ⤵)

Explanation

command	Sets the trigger direction of the level of the channel designated by *ch$*.
query	Returns the current trigger direction as a character value.

Example

```
:TRIGGER:SLOPE CH1,UP
```

Sets the trigger direction of channel 1 to rising.

When allowed In all functions.

■ Sets and queries the filter width for level or logic trigger.

Syntax

command	:TRIGger:FILTer *ch$,A*
query	:TRIGger:FILTer? *ch$*
response	*ch$,A*<NR*1*>

ch$ = CH1 to CH4 (level trigger)
ch$ = CH1, CH2 (logic trigger)
A = 0, 2 to 4000 (0: OFF)

Explanation

command: Sets the filter width for a trigger of the channel designated by *ch$* as a numerical value from 2 to 4000. (Level trigger, Logic trigger)
The recorder or the X-Y recorder is set only by ON or OFF. (Numerical value other than 0 is regarded as ON)

query: Returns the current filter width as NR1 numerical value.

Example

```
:TRIGGER:FILTER CH1,10
```

Sets the filter width of channel 1 to 10.

When allowed In all functions.

■ Sets and queries the width for glitch detection or timeout trigger.

Syntax

command	:TRIGger:WIDTh *ch$,A*
query	:TRIGger:WIDTh? *ch$*
response	*ch$,A*<NR1>

ch$ = CH1 to CH4
A = 2 to 4000

Explanation

command: Sets the width of the glitch detection trigger and time out trigger for the channel designated by *ch$*.
Sets the trigger width as a numerical value in the range 2 to 4000.

query: Returns the current setting (width) as a numerical value in NR1.

Example

```
:TRIGGER:WIDTH CH1,10
```

Sets the width of channel 1 to 10.

When allowed In the memory recorder function and the FFT function.

TRIGger

■ Selects and queries the trigger filter or event trigger.

Syntax

command :TRIGger:FILTEvent *ch$,A$*
query :TRIGger:FILTEvent? *ch$*
response *ch$,A$*
ch$ = CH1 to CH4
A$ = FILTer : trigger filter
EVENt : event trigger

Explanation

command: Specifying FILTer selects the trigger filter (level trigger), and specifying EVENt selects the event trigger.
A character string is returned indicating whether the channel specified by ch$ is using the trigger filter (level trigger) or event trigger.

Example

```
:TRIGGER:FILTEVENT CH1,EVENT
```

Channel 1 is used event trigger.

When allowed In the memory recorder function and the FFT function.

■ Sets and queries upper limit level for a window trigger.

Syntax

command :TRIGger:UPPEr *ch$,A*
query :TRIGger:UPPEr? *ch$*
response *ch$,A*<NR1>
ch$ = CH1 to CH4
A = 1 to 100 (%)

Explanation

command: Sets the upper limit level of the window trigger of the channel designated by *ch$*.
query: Returns the current upper limit level of the window trigger as an NR1 numerical value.

Example

```
:TRIGGER:UPPER CH1,80
```

Sets the upper limit level of the window trigger of channel 1 to 80 %.

When allowed In all functions.

■ Sets and queries lower limit level for a window trigger.

Syntax

command	:TRIGger:LOWEr *ch$,A*
query	:TRIGger:LOWEr? *ch$*
response	*ch$,A*<NR1>

ch$ = CH1 to CH4
A = 0 to 99 (%)

Explanation

command	Sets the lower limit level of the window trigger of the channel designated by *ch$*.
query	Returns the current lower limit level of the window trigger as an NR1 numerical value.

Example

```
:TRIGGER:LOWER CH1,20
```

Sets the lower limit level of the window trigger of channel 1 to 20 %.

When allowed In all functions.

■ Sets and queries the trigger pattern for a logic trigger.

Syntax

command	:TRIGger:LOGPat *ch$,'A$'*
query	:TRIGger:LOGPat? *ch$*
response	*ch$,'A$'*

ch$ = CH1, CH2
A$ = XXXXXXXX (trigger pattern)
1-A-4 1-B-4 (CH1)
1-C-4 1-D-4 (CH2)
Trigger pattern : X, 0, 1

Explanation

command	Sets the trigger pattern for the logic trigger of the channel designated by *ch$* to that specified by the given character data. (character is X except X, 0, and 1)
query	Returns the current trigger pattern for the logic trigger as that specified by the given character data.

Example

```
:TRIGGER:LOGPAT CH1,'10XX10XX'
```

Sets the trigger pattern for channel 1 to "10XX10XX".

When allowed In all functions.

TRIGger

■ Sets and queries the logical operator (AND/OR) for the trigger pattern of a logic trigger.

Syntax

command	:TRIGger:LOGAnd *ch$,A$*
query	:TRIGger:LOGAnd? *ch$*
response	*ch$,A$*
	ch$ = CH1, CH2
	A$ = OR, AND

Explanation

command	Sets the AND/OR logical operator for the trigger pattern of a logic trigger.
query	Returns the present AND/OR setting as a character string.

Example

```
:TRIGGER:LOGAND CH1,OR
```

Sets the AND/OR logical operator for the trigger pattern of channel 1 to OR.

When allowed In all functions.

■ Sets and queries number of setting (COUNT) for an event trigger.

Syntax

command	:TRIGger:EVENt *ch$,A*
query	:TRIGger:EVENt? *ch$*
response	*ch$,A*<NR1>
	ch$ = CH1 to CH4
	A = 0, 2 to 4000

Explanation

command	Sets the number of settings for the event trigger of the channel designated by *ch$*. 0 means OFF.
query	Returns the current number of counts for the event trigger as an NR1 numerical value.

Example

```
:TRIGGER:EVENT CH1,10
```

Sets to 10 the number of counts for the event trigger of channel 1.

When allowed In the memory recorder function and the FFT function.

■ Sets and queries external trigger.

Syntax

command	:TRIGger:EXTErnal *A$*
query	:TRIGger:EXTErnal?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables and disables external trigger.
query	Returns the current external trigger enablement state as character data.

Example

```
:TRIGGER:EXTERNAL OFF
```

Sets the external trigger to OFF.

When allowed In all functions.

■ Sets and queries whether the timer trigger is on or off.

Syntax

command	:TRIGger:TIMEr *A$*
query	:TRIGger:TIMEr?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables or disables the timer trigger.
query	Returns the current enablement state of the timer trigger as character data.

Example

```
:TRIGGER:TIMER ON
```

Sets the timer trigger to ON.

When allowed In all functions.

TRIGger

■ Sets and queries the start instant for the timer trigger.

Syntax

command	:TRIGger:TMSTArt *month,day,hour,min*
query	:TRIGger:TMSTArt?
response	*month,day,hour,min*
	month = 1 to 12
	day = 1 to 31
	hour = 0 to 23
	min = 0 to 59

Explanation

command	Sets the start instant for the timer trigger.
query	Returns the current setting for the timer trigger start instant as NR1 numerical values.

Example

```
:TRIGGER:TMSTART 7, 5, 9, 30
```

Sets the start instant for the timer trigger to 09:30 on July 5th.

When allowed In all functions.

■ Sets and queries the stop instant for the timer trigger.

Syntax

command	:TRIGger:TMSTOp *month,day,hour,min*
query	:TRIGger:TMSTOp?
response	*month,day,hour,min*
	month = 1 to 12
	day = 1 to 31
	hour = 0 to 23
	min = 0 to 59

Explanation

command	Sets the stop instant for the timer trigger.
query	Returns the current setting for the timer trigger stop instant as NR1 numerical values.

Example

```
:TRIGGER:TMSTOP 7, 5, 10, 30
```

Sets the stop instant for the timer trigger to 10:30 on July 5th.

When allowed In all functions.

■ Sets and queries the time interval for the timer trigger.

Syntax

command	:TRIGger:TMINTvl *hour,min,sec*
query	:TRIGger:TMINTvl?
response	*hour,min,sec*
	hour = 0 to 23
	min = 0 to 59
	sec = 0 to 59

Explanation

command	Sets the time interval for the timer trigger.
query	Returns the current setting for the timer trigger time interval as NR1 numerical values.

Example

```
:TRIGGER:TMINTVL 1,20,30
```

Sets the time interval for the timer trigger to one hour, twenty minutes, and thirty seconds.

When allowed In all functions.

■ Sets and queries trigger mode.

Syntax

command	:TRIGger:MODE *A$*
query	:TRIGger:MODE?
response	*A$*
	A$ = SINGle, REPEat, AUTO (MEM)
	SINGle, REPEat (REC)
	SINGle, REPEat, AUTO (FFT)

Explanation

command	Sets the trigger mode.
query	Returns the current trigger mode as character data.

Example

```
:TRIGGER:MODE REPEAT
```

Sets the trigger mode to repeat.

When allowed In the memory recorder function, the recorder function, and the FFT function.

TRIGger

■ Sets and queries pre-trigger.

Syntax

command	:TRIGger:PRETrig *A*
query	:TRIGger:PRETrig?
response	*A*<NR1>

A = 0, 2, 5, 10, 20,..., 80, 90, 100, -950 to -50 (unit %)
(-950 to -50; 50 % step)

Explanation

command	Sets pre-trigger value to a numerical value (in percent). If an attempt is made to set a value which cannot be set on the 8853, setting is performed to the next higher permitted value.
query	Returns the currently set pre-trigger value as an NR1 numerical value.

Example

```
:TRIGGER:PRETRIG 10
```

Pre-trigger value is set to 10%.

When allowed In the memory recorder function and the FFT function.

■ Sets and queries trigger timing.

Syntax

command	:TRIGger:TIMIng *A$*
query	:TRIGger:TIMIng?
response	*A$*

A$ = START
STOP
S_S (START&STOP)

Explanation

command	Sets the trigger timing.
query	Returns the currently set trigger timing as a character string.

Example

```
:TRIGGER:TIMING START
```

Sets the trigger timing to START.

When allowed In the recorder function and the XYC recorder function.

■ Queries the time point for trigger detection.

Syntax

query :TRIGger:TRGTime? (*A*)

response *hour*<NR1>,*min*<NR1>,*sec*<NR1>

A = block number during memory segmentation
hour = 0 to 23
min = 0 to 59
sec = 0 to 59

Explanation

query Returns the currently set time point for trigger detection as a numerical value in NR1 format.
During memory segmentation, returns the time point for trigger detection in the memory block whose block number is specified.

Example

```
:TRIGGER:TRGTIME?
```

The currently set time point for trigger detection is queried.

When allowed In all functions.

■ Queries the date for trigger detection.

Syntax

query :TRIGger:TRGDate? (*A*)

response *year*<NR1>,*month*<NR1>,*day*<NR1>

A = block number during memory segmentation
year = 0 to 99
month = 1 to 12
day = 1 to 31

Explanation

query Returns the currently set date for trigger detection as a numerical value in NR1 format.
During memory segmentation, returns the date for trigger detection in the memory block whose block number is specified.

Example

```
:TRIGGER:TRGDATE?
```

The currently set date for trigger detection is queried.

When allowed In all functions.

4. UNIT command (Sets and queries input channel)

■ Sets and queries the voltage axis range of an input channel.

Syntax

command	:UNIT:RANGe *ch$,A*
query	:UNIT:RANGe? *ch$*
response	*ch$,A*<NR3>
	ch$ = CH1 to CH4
	A = voltage axis range (unit V)

Explanation

command	Sets the voltage axis range for the channel designated by *ch$*. *ch$* to a numerical value.
query	Returns the current voltage axis range for the channel designated by *ch$* as an NR3 numerical value.

Example

```
:UNIT:RANGE CH1,+20.E-3
```

Sets the voltage axis range for channel 1 to 20 mV.

When allowed In all functions.

■ Sets and queries input channel origin position.

Syntax

command	:UNIT:POSItion *ch$,A*
query	:UNIT:POSItion? *ch$*
response	*ch$,A*<NR1>
	ch$ = CH1 to CH4
	A = -100 to 100 (%)

Explanation

command	Sets the origin position for the channel designated by *ch$* in the range.
query	Returns the current origin position for the channel designated by *ch$* as an NR1 numerical value (unit percent).

Example

```
:UNIT:POSITION CH1,50
```

Sets the origin position for channel 1 to 50 %

When allowed In all functions.

■ Sets and queries input coupling for an input channel.

Syntax

command	:UNIT:COUPling *ch$,A$*
query	:UNIT:COUPling? *ch$*
response	*ch$,A$*
	ch$ = CH1 to CH4
	A$ = GND, AC, DC

Explanation

command	Sets the input coupling for the channel designated by *ch$*.
query	Returns the current input coupling for the channel designated by *ch$* as character data.

Example

```
:UNIT:COUPLING CH1,DC
```

Sets the input coupling for channel 1 to DC.

When allowed In all functions.

■ Sets and queries the filter for an input channel.

Syntax

command	:UNIT:FILTer *ch$,A*
query	:UNIT:FILTer? *ch$*
response	*ch$,A*<NR3>
	ch$ = CH1 to CH4
	A = 0, 5.0E5, 5.0E2, 5 (0 : OFF)

Explanation

command	Sets the filter for the channel designated by *ch$*.
query	Returns the current filter setting for the channel designated by *ch$* as an NR3 numerical value.

Example

```
:UNIT:FILTER CH1,5
```

Sets the filter for channel 1 to 5 Hz.

When allowed In all functions.

■ Carries out zero adjustment for the input units.

Syntax command :UNIT:ADJUST

Explanation command Carries out zero adjustment for the input units.

When allowed In all functions.

5. DISPlay command (Sets and queries changeover of the screen mode and waveform display.)

■ Sets and queries the screen mode.

Syntax

command	:DISPlay:CHANge *A$*
query	:DISPlay:CHANge?
response	*A$*

A$ = SYSTem : system screen
STATus : status screen
TRIGger : trigger screen
DISPlay : display screen

Explanation

command	Changes the screen mode.
query	Returns the current screen mode as character data.

Example

```
:DISPLAY:CHANGE DISPLAY
```

Switches to the display mode.

When allowed In all functions.

■ Sets and queries waveform display style.

Syntax

command	:DISPlay:DRAWing *ch$,A$*
query	:DISPlay:DRAWing? *ch$*
response	*ch$,A$*

ch$ = CH1 to CH4
A$ = OFF, LIGHt, DARK

Explanation

command	Sets the waveform display style for the channel designated by *ch$* to OFF, LIGHT (low intensity), or DARK (high intensity).
query	Returns the current display or recording style setting for the channel designated by *ch$* as character data.

Example

```
:DISPLAY:DRAWING CH1,DARK
```

Displays the channel 1 waveform the DARK.

When allowed In the memory recorder function, the recorder function, and the XYC recorder function.

■ Enable and disables, and queries, display of logic wave forms.

Syntax

command	:DISPlay:LOGDraw *ch$,A$*
query	:DISPlay:LOGDraw? *ch$*
response	*ch$,A$*
	ch$ = CH1 to CH4
	A$ = OFF, ON

Explanation

command	Enables and disables display of logic waveform.
query	Returns the current enablement state of logic waveform display as character data.

Example

```
:DISPLAY:LOGDRAW CH1,ON
```

Enables display of the channel 1 logic waveform.

When allowed In the memory recorder function and the recorder function.

■ Sets and queries changeover of the page of the screen.

Syntax

command	:DISPlay:PAGE *A*
query	:DISPlay:PAGE?
response	*A*<NR1>
	[On the status screen]
	A = 1 : page 1
	2 : page 2 (no XY recorder function)
	[On the system screen]
	A = 1 : INITIALIZE
	2 : SCALING
	3 : COMMENT
	4 : SETUP
	5 : INTERFACE
	6 : CRTCOPY
	7 : SELF CHECK

Explanation

command	Changes over the page of the status or system screen according to the corresponding numerical value.
query	Returns the current page of the status or system screen as a corresponding NR1 numerical value.

Example

```
:DISPLAY:CHANGE SYSTEM
:DISPLAY:PAGE 4
```

Changes over the SETUP screen accessed from the system screen.

When allowed In all functions.

DISPlay

■ Sets and queries waveform display graph in DUAL and DUAL (print QUAD) format.

Syntax

command :DISPlay:GRAPh *ch$,G$*
query :DISPlay:GRAPh? *ch$*
response *ch$,G$*
ch$ = CH1 to CH4
G$ = G1, G2 : graph 1, graph 2

Explanation

command: Sets the waveform display graph on the screen.
query: On the screen, returns the current waveform display graph for a channel as character data.

Example

```
:DISPLAY:GRAPH CH1,G1
```

Displays the channel 1 waveform in display graph 1.

When allowed In the memory recorder function and the recorder function.

■ Sets and queries magnification/compression factor on the time axis.

Syntax

command :DISPlay:XMAG *A$*
query :DISPlay:XMAG?
response *A$*
[MEM]
A$ = X10, X5, X2, X1, X1_2, X1_5, X1_10, X1_20, X1_50, X1_100, X1_200, X1_500, X1_1000, X1_2000, X1_4000
[REC]
A$ = X1, X1_2, X1_5, X1_10, X1_20, X1_50
For example X1_2 means 1/2, X1_5 means 1/5.

Explanation

command: Sets the magnification/compression factor on the time axis according to character data.
When the zoom function is used, sets the magnification/compression factor on the time axis for the lower graph.
query: Returns the current magnification/compression factor on the time axis as character data.

Example

```
:DISPLAY:XMAG X1_10
```

Sets the compression ratio along the time axis to be 1/10.

When allowed In the memory recorder function and the recorder function.

■ Enables and disables, and queries the zoom function.

Syntax

command	:DISPlay:ZOOM *A$*
query	:DISPlay:ZOOM?
response	*A$*

A$ = OFF, ON

Explanation

command	Enables and disables the zoom function.
query	Returns the current enablement state of the zoom function as character date.

Example

```
:DISPLAY:ZOOM ON
```

Enables the zoom function.

When allowed In the memory recorder function.

■ Sets and queries magnification/compression factor on the time axis, when the zoom function is used.

Syntax

command	:DISPlay:ZOOMMag *A$*
query	:DISPlay:ZOOMMag?
response	*A$*

A$ = X10, X5, X2, X1, X1_2, X1_5, X1_10, X1_20, X1_50, X1_100, X1_200, X1_500, X1_1000, X1_2000, X1_4000

For example X1_2 means 1/2, X1_5 means 1/5.

Explanation

command	Sets the magnification/compression factor on the time axis for the upper graph, when the zoom function is used.
query	Returns as character data the current magnification/compression factor on the time axis for the upper graph in the zoom function.

Example

```
:DISPLAY:ZOOMMAG X1_100
```

Sets to be 1/100 the compression ratio along the time axis for the upper graph in the zoom function.

When allowed In the memory recorder function.

DISPlay

■ Sets and queries magnification/compression factor on the voltage axis.

Syntax

command	:DISPlay:YMAG *ch$,A$*
query	:DISPlay:YMAG? *ch$*
response	*ch$,A$*
	ch$ = CH1 to CH4
	A$ = X1_2, X1, X2, X5, X10 (X1_2 : X1/2)

Explanation

command	Sets the magnification/compression factor on the voltage axis for the channel designated by *ch$* according to the character data.
query	Returns the current magnification/compression factor on the voltage axis for the channel designated by *ch$* as character data.

Example

```
:DISPLAY:YMAG X2
```

Sets the magnification ratio on the voltage axis to be ×2.

When allowed In the memory recorder function and the recorder function.

■ Sets and queries waveform display position on the voltage axis.

Syntax

command	:DISPlay:YZOOm *ch$,A*
query	:DISPlay:YZOOm? *ch$*
response	*ch$,A*<NR1>
	ch$ = CH1 to CH4
	A = 1 to 100 (%)

Explanation

command	Sets the waveform display position on the voltage axis. Sets the percentage of the displayed position on full scale in the center of the display screen.
query	Returns the current waveform display position on the voltage axis as an NR1 numerical value.

Explanation

```
:DISPLAY:YZOOM CH1,40
```

Displays the position of 40 % on full scale on channel 1 in the center of the display screen.

When allowed In the memory recorder function and the recorder function.

■ Performs waveform display.

Syntax

command :DISPlay:WAVE *A$*

A$ = ACUR (the A cursor)
TRIG (the trigger point)
POINT (the point set by :MEMory:POINt.)

Explanation

command Displays the waveform on the screen from the position indicated by *A$*.
Displays the waveform from the position of the last 60 points if within the last 60 points of data are indicated.

Example

```
:DISPLAY:WAVE ACUR
```

Displays the waveform from the position of A cursor.

When allowed

In the memory recorder function (when *A$* = ACUR, the A cursor must be displayed).

■ Enables and disables the memory segmentation screen.

Syntax

command :DISPlay:DIVMap *A$*
query :DISPlay:DIVMap?
response *A$*

A$ = ON : Enter the memory segmentation screen.
OFF : Exit from the memory segmentation screen.

Explanation

command Enables and disables the memory segmentation screen.
query Returns the current memory segmentation screen enablement as character data.

Example

```
:DISPLAY:DIVMAP ON
```

Displays the memory segmentation screen.

When allowed

In the memory recorder function.

DISPlay

■ Enables and disables the waveform processing calculation screen.

Syntax

command :DISPlay:CALCEdit *A$*
query :DISPlay:CALCEdit?
response *A$*
A$ = ON : Enter the waveform processing calculation screen.
OFF : Exit from the waveform processing calculation screen.

Explanation

command: Enables and disables the waveform processing calculation screen.
query: Returns the current waveform processing calculation screen enablement as character data.

Example

```
:DISPLAY:CALCEDIT ON
```

Displays the waveform processing calculation screen.

When allowed

In the memory recorder function.

■ Enables and disables the waveform parameter calculation screen.

Syntax

command :DISPlay:MEASEdit *A$*
query :DISPlay:MEASEdit?
response *A$*
A$ = ON : Enter the waveform parameter calculation screen.
OFF : Exit from the waveform parameter calculation screen.

Explanation

command: Enables and disables the waveform parameter calculation screen.
query: Returns the current waveform parameter calculation screen enablement as character data.

Example

```
:DISPLAY:MEASEDIT ON
```

Displays the waveform parameter calculation screen.

When allowed

In the memory recorder function.

■ Sets and queries the x-axis, in the XY format.

Syntax

command	:DISPlay:XAXIs *ch$*
query	:DISPlay:XAXIs?
response	*ch$*
	ch$ = CH1 to CH4

Explanation

command	Sets the x-axis channel in the XY format.
query	Returns the current x-axis channel in the XY format.

Example

```
:DISPLAY:XAXIS CH1
```

Sets channel 1 to the x-axis.

When allowed In the memory recorder function (in XY format) and in the XYC recorder function.

■ Sets and queries display clearing function.

Syntax

command	:DISPlay:XYCLr *A$*
query	:DISPlay:XYCLr?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables or disables display clearing function.
query	Returns the enablement of display clearing function.

Example

```
:DISPLAY:XYCLR ON
```

Sets the display clearing to ON.

When allowed In the XYC recorder function.

DISPlay

■ Sets and queries the FFT analysis channel.

Syntax

command	:DISPlay:FFTCH *G$,ch$*
query	:DISPlay:FFTCH? *G$*
response	*G$,ch$*

G$ = G1 : graph 1
G2 : graph 2
ch$ = CH1 to CH4

Explanation

command: Sets the FFT analysis channel for the graph designated by *G$*. G2 can be designated even if the display format is SINGLE, but this does not affect the display.

query: Returns the current FFT analysis channel for the graph designated by *G$* as character data.

Example

```
:DISPLAY:FFTCH G1,CH1
```

Sets the FFT analysis channel for graph 1 to channel 1.

When allowed

In the FFT function.

6. CURSor command (Cursor setting and reading)

■ Turns on and off, and queries, the A and B cursors.

Syntax

command :CURSor:MODE *A$*
query :CURSor:MODE?
response *A$*
A$ = OFF, TIME, VOLT, TRACe (MEM)
OFF, Xcur, Ycur (MEM (XY format))
OFF, TIME, VOLT (REC)
OFF, Xcur, Ycur (XYC)
OFF, ON (FFT)

Explanation

command: Sets the A and B cursor type (vertical cursor, horizontal cursor, trace cursor).
TIME, Xcur : vertical cursor
VOLT, Ycur : horizontal cursor
TRACe : trace cursor

query: Returns the current A and B cursor type as character data.

Example

```
:CURSOR:MODE TIME
```

Sets vertical cursors.

When allowed

In all functions.

■ Selects between, and queries, A only or A and B cursors.

Syntax

command :CURSor:ABCUrsor *A$*
query :CURSor:ABCUrsor?
response *A$*
A$ = A : only A cursor
A_B : both A and B cursors

Explanation

command: Selects between A only or A and B cursors.
query: Returns whether currently the A cursor only or both A and B cursors are in use, as character data.

Example

```
:CURSOR:ABCURSOR A
```

Sets A cursor.

When allowed

In the memory recorder function, the recorder function, and the XYC recorder function.

CURSor

■ Sets and queries the channel for the A cursor.

Syntax

command	:CURSor:ACHAnnel *ch$*
query	:CURSor:ACHAnnel?
response	*ch$*
	ch$ = CH1 to CH4

Explanation

command	Sets the channel for the A cursor.
query	Returns the current A cursor channel as character data.

Example

```
:CURSOR:ACHANNEL CH1
```

Uses the A cursor to channel 1.

When allowed

In the memory recorder function, recorder function, and XYC recorder function, during use of the trace cursor or the horizontal cursor.

■ Sets and queries the channel for the B cursor.

Syntax

command	:CURSor:BCHAnnel *ch$*
query	:CURSor:BCHAnnel?
response	*ch$*
	ch$ = CH1 to CH4

Explanation

command	Sets the channel for the B cursor.
query	Returns the current B cursor channel as character data.

Example

```
:CURSOR:BCHANNEL CH1
```

Uses the B cursor to channel 1.

When allowed

In the memory recorder function, recorder function, and XYC recorder function, during use of the trace cursor or the horizontal cursor.

■ Sets and queries the method of displaying FFT analysis voltage value.

Syntax

command	:CURSor:YDISp *A$*
query	:CURSor:YDISp?
response	*A$*
	A$ = PEAK : peak value
	RMS : rms value

Explanation

command	Sets the method of displaying the FFT voltage value.
query	Returns the current FFT trace cursor readout value setting as character data.

Example

```
:CURSOR:YDISP RMS
```

Sets the FFT trace cursor readout value as RMS value.

When allowed In the FFT function.

■ Sets and queries the position of the A cursor.

Syntax

command	:CURSor:APOSition *A*
query	:CURSor:APOSition?
response	*A*<NR1>
	[vertical cursor, trace cursor]
	A = 0 to number of stored data values
	(40 × recording length) (MEM, REC)
	A = 0 to 400 (XYC, MEM(XY format))
	[horizontal cursor]
	A = 0 to 250

Explanation

command	Sets the A cursor position.
query	Returns the current A cursor position as an NR1 numerical value.

Example

```
:CURSOR:APOSITION 400
```

Moves the A cursor position to 400 points (10 divisions).

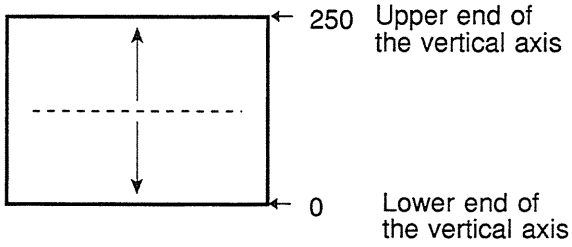
When allowed In the memory recorder function, the recorder function, and the XYC recorder function.

CURSor

The cursor position (MEM, REC, XYC)

Horizontal cursor

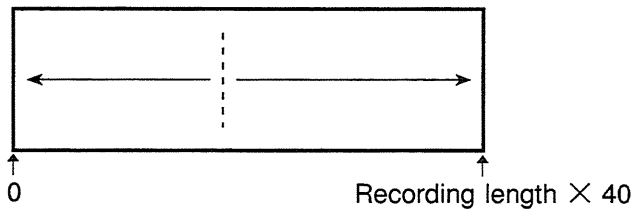
Lower end of the vertical axis : 0
Upper end of the vertical axis : 250



Vertical cursor or trace cursor

① In the memory recorder function and the recorder function

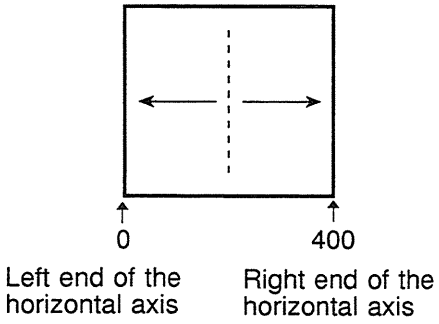
The cursor position is an indication of the number of stored data.
(1 DIV = 40 points)



When recording length is 15 division, the number of stored data is 600 points (15 divisions × 40 points). Therefore the cursor position indication lies in the range from 0 to 600.

② In the XYC recorder function and the memory recorder function (X-Y format)

Left end of the horizontal axis : 0
Right end of the horizontal axis : 400



■ Sets and queries the position of the B cursor.

Syntax

command	:CURSor:BPOSition *A*
query	:CURSor:BPOSition?
response	*A*<NR1>
	Same as the A cursor setting. (Refer to the previous page.)

Explanation

command	Sets the B cursor position.
query	Returns the current B cursor position as an NR1 numerical value.

When allowed In the memory recorder function, the recorder function, and the XYC recorder function.

■ Sets and queries the position of the FFT trace cursor.

Syntax

command	:CURSor:FPOSition *A*
query	:CURSor:FPOSition?
response	*A*<NR1>
	A = 0 to 799 (analysis mode; stored waveform)
	0 to 399 (analysis mode; linear spectrum, power spectrum)

Explanation

command	Sets the FFT trace cursor position.
query	Returns the current FFT trace cursor position as an NR1 numerical value.

Example

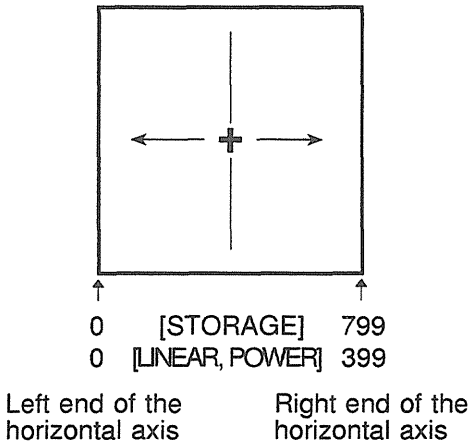
```
:CURSOR:FPOSITION 100
```

Move the FFT trace cursor position to 100 points.

When allowed In the FFT function.

The cursor position (FFT)

Left end of the horizontal axis : 0
Right end of the horizontal axis : depends on the analysis mode
(Stored waveform: 799, linear spectrum: 399, power spectrum: 399)



CURSor

■ Queries the cursor readout value (t).

Syntax

query :CURSor:DTREad?

response "*A* unit" ("*B* unit")

A$ = t or Δt readout value

B$ = 1/t or 1/Δt readout value (vertical cursor only)

= V or ΔV readout value (trace cursor only)

Explanation

query Returns the cursor readout value (t), (1/t) as a line of character data.

Example

```
:CURSOR:DTREAD?
```

Queries the A cursor readout value.

When allowed In the memory recorder function and recorder function. (t or Δt is being shown on the display.)

■ Sets and queries the cursor readout value (V).

Syntax

query :CURSor:DVREad?

response "*A* unit"

A$ = V or ΔV readout value

Explanation

query Returns the cursor readout value (V) as a line of character data.

Example

```
:CURSOR:DVREAD?
```

Queries the cursor readout value.

When allowed In the memory recorder function, the recorder function, and the XYC recorder function. (V or ΔV is being shown on the display.)

■ Queries the FFT cursor readout value.

Syntax

query :CURSor:FFTRead?

response "*A$* unit","*B$* unit"

A$ = x-axis readout value

B$ = y-axis readout value

Explanation

query Returns the current cursor readout value in the FFT function as a line of character data.

Example

```
:CURSOR:FFTREAD?
```

Queries the FFT cursor readout position.

When allowed In the FFT function (provided that the cursor is on).

7. MEMory command (Sets and queries input and output from the memory)

■ Sets and queries the point in memory for input/output.

Syntax

command	:MEMory:POINt *ch$,A*
query	:MEMory:POINt?
response	*ch$,A*<NR1>
	ch$ = CH1 to CH4
	A = 0 to recording length × 40 (2000000 max.)

Explanation

command	Sets the input/output point in memory. 1 DIV is 40 points.
query	Returns the current input/output point in memory as an NR1 numerical value.

Example

```
:MEMORY:POINT CH1,100
```

Sets the input/output point for channel 1 to the 100th location from the start of memory.

When allowed In the memory recorder function.

■ Queries the number of data samples stored.

Syntax

query	:MEMory:MAXPoint?
response	*A*<NR1>
	A = 0 (no data stored), 600 to 2000000 (divided by 40 gives the number of divisions)

Explanation

query	Returns the number of data samples stored in the memory. 1 DIV is 40 points.

Example

query	`:MEMORY:MAXPOINT?`
response	`:MEMORY:MAXPOINT 600` (when headers are on)

The number of data samples stored in the memory is 600 (for 15 divisions).

When allowed In the memory recorder function.

MEMory

■ Inputs data to memory, and outputs stored data.

Syntax

command	:MEMory:ADATa *B,C,...*
query	:MEMory:ADATa? *A*
response	*B,C,...* all <NR1>
	B, C,... = -48 to 4047 (data for storage)
	A = 1 to 40 (number of data values to be output)

Explanation

command

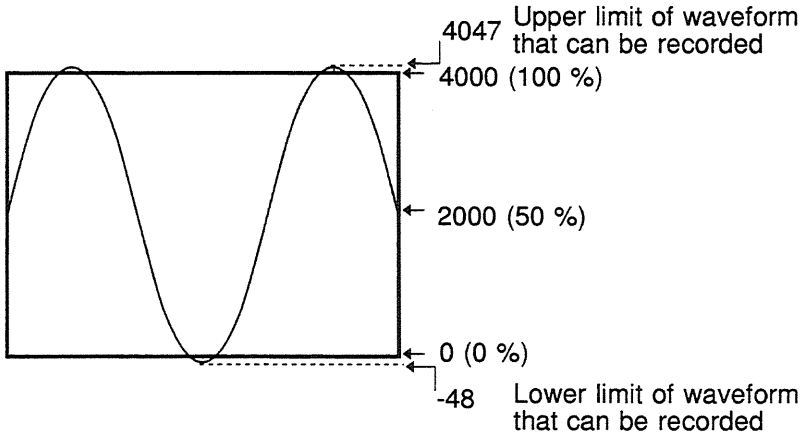
Puts the data of the data portion into the memory at the channel and point set by the :MEMory:POINt command.
If there are several data values, they are input in order from the point set by the :MEMory:POINt command. The input/output point is incremented by the number of data values.

query

The number of data values specified by A are output from the memory channel and point set by the :MEMory:POINt command.
The input/output point is incremented by the number of data values.
This cannot be executed during measurement operation.

Relationship between data values in memory and measured voltages

The following figure illustrates the relationship between the data values (-48 to 4047) input and output using the :MEMory:ADATa command and the measured voltage values.



Example

```
:MEMORY:POINT CH1,0
:MEMORY:ADATA? 10
```

Sets the input/output point to channel 1 and data value zero in memory, then outputs 10 stored data values.

When allowed

In the memory recorder function, provided that stored data is present, and provided that the input/output point is lower than the amount of data stored.

■ Input voltage data to memory, and output voltage data from memory.

Syntax

command	:MEMory:VDATa *B,C,...*
query	:MEMory:VDATa? *A*
response	*B,C,...* all <NR3>
	B,C... = voltage values (unit V)
	A = 1 to 10 (amount of data)

Explanation

command

Puts the data values (voltage values) in the data portion into the memory at the channel and point set by the :MEMory:POINt command.

If there are several data values, they are input in order from the point set by the :MEMory:POINt command. The input/output point is incremented by the number of data values.

query

The number of stored data values specified by *A* are output as voltage values from the memory channel and point set by the :MEMory:POINt command.
The input/output point is incremented by the number of data values.

When scaling, the scaled values are input and output.
When calculating the waveform, calculated results are input and output.
This cannot be executed during measurement operation.

Example

```
:MEMORY:POINT CH1,0
:MEMORY:VDATA? 10
```

Sets the input/output point to channel 1 and data value zero in memory, then outputs 10 stored data values as voltage values.

When allowed

In the memory recorder function, provided that stored data is present, and provided that the input/output point is lower than the amount of data stored.

MEMory

■ Input logic data to memory, and output logic data from memory.

Syntax

command :MEMory:LDATa *B,C...*
query :MEMory:LDATa? *A*
response *B,C....* all <NR1>
B, C.... = 1 to 15 (logic data)
A = 1 to 40 (number of data values to be output)

Explanation

command

Puts the data values (logic values) in the data portion into the memory at the channel and point set by the :MEMory:POINt command.
If there are several data values, they are input in order from the point set by the :MEMory:POINt command. The input/output point is incremented by the number of data values.

query

The number of stored data values specified by A are output as logic values from the memory channel and point set by the :MEMory:POINt command.
The input/output point is incremented by the number of data values.
This cannot be executed during measurement operation.

The following is the correspondence between the channels set by the :MEMory:POINt command and the logic channel groups:

CH1 : CHA1 to A4
CH2 : CHB1 to B4
CH3 : CHC1 to C4
CH4 : CHD1 to D4

The eight logic channels in each group are encoded as binary bits in the NR1 data value, as shown in the following example.

7	6	5	4	3	2	1	0
0	0	0	0	A4	A3	A2	A1

Example

```
:MEMORY:POINT CH1,0
(query) :MEMORY:LDATA? 1
(response) :MEMORY:LDATA 10 (when headers are on)
```

In this case channels A1 to A4 are as follows;

7	6	5	4	3	2	1	0
0	0	0	0	1	0	1	0
				A4	A3	A2	A1

LOW : 0
HIGH : 1

When allowed

In the memory recorder function, provided that stored data is present, and provided that the input/output point is lower than the amount of data stored.

■ Outputs real time data (in ASCII).

Syntax

query	:MEMory:AREAl? *ch$*
response	*A*<NR1>
	ch$ = CH1 to CH4
	A = -48 to 4047

Explanation

query: Returns the value input on the channel designated by *ch$*.

Example

query	`:MEMORY:AREAL? CH1`
response	`:MEMORY:AREAL 125` (When headers are on)

When allowed Providing that measurement operation is not taking place.

■ Outputs real time data (voltage values).

Syntax

query	:MEMory:VREAl? *ch$*
response	*A*<NR3>
	ch$ = CH1 to CH4
	A = a voltage value (unit V)

Explanation

query: Returns as a voltage value the value input on the channel designated by *ch$*.

Example

query	`:MEMORY:VREAL? CH1`
response	`:MEMORY:VREAL 5.5E-2` (When headers are on)

When allowed Providing that measurement operation is not taking place.

■ Outputs real time data (logic).

Syntax

query	:MEMory:LREAl? *ch$*
response	*A*<NR1>
	ch$ = CH1 to CH4
	A = 0 to 15

Explanation

query: Returns as an NR1 numerical value, the value input on the channel designated by *ch$*.
The correspondence between the logic channel groups and the response data is the same as that of LDAT on the previous page.

Example

query	`:MEMORY:LREAL? CH1`
response	`:MEMORY:LREAL 9` (When headers are on)

Indicates that the current logic data for CHA4 to CHA1 is 1001.

When allowed Providing that measurement operation is not taking place.

MEMory

■ Binary transfer of stored data.

Syntax

query :MEMory:BDATa? *A*

response #0＊＊＊＊＊＊ ・・・・

A = 1 to 125

Explanation

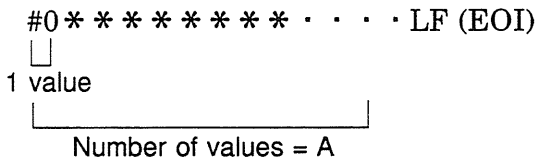
Outputs the data stored by a :MEMory:POINt specification in binary format. The input/output point is incremented by the number of data values. It is not possible to input data in binary format.

The format of the output data is as follows:

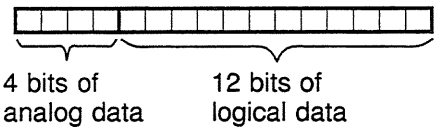
① Initially: "#0" (Indicates binary format.)

② After "#0", the number of data values specified by *A* (each value is one byte) is transmitted.

③ The data is followed by LF (0AH) + EOI.



④ Output data consists of 16 bits, including 12 bits of analog data and 4 bits of logical data.



Example

```
:MEMORY:POINT CH1,0
:MEMORY:BDATA? 10
```

This sets the input/output point to channel 1, and stored data value to address 0 in memory, then outputs 10 data values in binary format.

When allowed

In the memory recorder function, provided that stored data is present, and provided that the input/output point is lower than the amount of data stored.

■ Outputs real time data (binary)

Syntax

query :MEMory:BREAI? *ch$*

response #0＊

ch$ = CH1 to CH4

Explanation

query — Outputs in binary format the value input on the channel designated by *ch$*.

When allowed

Providing that measurement operation is not taking place.

■ Sets and queries the output point for FFT data.

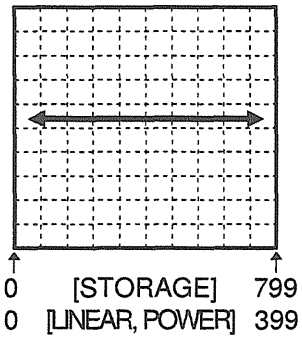
Syntax

command :MEMory:FFTPOint *A*
query :MEMory:FFTPOint?
response *A*<NR1>
A = 0 to 799 (in analysis mode; stored waveform)
0 to 399 (in analysis mode; linear spectrum, power spectrum)

Explanation

command: Sets the output point for FFT data.
In DUAL format, sets the output point only for the graph 1.

query: Returns the current output point for FFT data as a numerical value in <NR1> format.



Example

```
:MEMORY:FFTPOINT 50
```

Sets the output point for FFT data to 50.

When allowed In the FFT function.

■ Outputs FFT analysis data (in ASCII).

Syntax

query :MEMory:FFTData?
response *A*<NR3>
A = y-axis data

Explanation

query: Returns the y-axis FFT data at the output point specified by the ":MEMory:FFTPOint" command.
When this command is executed, the specified output point is increased by one.

Example

```
:MEMORY:FFTPOINT 50
:MEMORY:FFTDATA?
```

Outputs the y-axis FFT data at the point 50. (At this time the point is 51.)

When allowed In the FFT function.

8. SYSTem command (Sets and queries the system screen)

■ Sets the time, and queries the current time.

Syntax

command	:SYSTem:TIME *hour,min,sec*
query	:SYSTem:TIME?
response	*hour,min,sec*
	hour = 0 to 23
	min = 0 to 59
	sec = 0 to 59

Explanation

command	Sets the time.
query	Returns the current time.

Example

```
:SYSTEM:TIME 10,0,0
```

Sets the built-inclock to 10:00 on the dot.

When allowed In all functions.

■ Sets the calendar date, and queries the current calendar date.

Syntax

command	:SYSTem:DATE *year,month,day*
query	:SYSTem:DATE?
response	*year,month,day*
	year = 0 to 99
	month = 1 to 12
	day = 1 to 31

Explanation

command	Sets the date on the internal calendar.
query	Returns the current date.

Example

```
:SYSTEM:DATE 95,6,1
```

Sets the internal calendar to June first, 1995.

When allowed In all functions.

■ Clearing waveform data.

Syntax command :SYSTem:DATAClear

Explanation command Clear the waveform data.

When allowed In all functions.

■ Enables and disables, and queries the screen auto off (screen saver) function.

Syntax

command	:SYSTem:CRTOff *A$*
query	:SYSTem:CRTOff?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables or disables the screen saver function.
query	Returns the current enablement state of the screen saver function as character data.

Example

```
:SYSTEM:CRTOFF ON
```

Sets the screen saver function to ON.

When allowed In all functions.

■ Sets and queries the grid type.

Syntax

command	:SYSTem:GRID *A$*
query	:SYSTem:GRID?
response	*A$*
	A$ = OFF, NORMal, FINE

Explanation

command	Sets the type of grid displayed.
query	Returns the current grid setting as character data.

Example

```
:SYSTEM:GRID NORMAL
```

Sets the grid type to NORMAL.

When allowed In all functions.

■ Enables and disables, and queries the start key backup function.

Syntax

command	:SYSTem:STARt *A$*
query	:SYSTem:STARt?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables and disables the start key backup function.
query	Returns the current enablement state of the start key backup function as character data.

Example

```
:SYSTEM:START ON
```

Sets the start key backup function to ON.

When allowed In all functions.

SYSTem

■ Enables and disables, and queries the channel marker.

Syntax

command	:SYSTem:CHMArk *A$*
query	:SYSTem:CHMArk?
response	*A$*
	A$ = OFF, ON

Explanation

command	Makes the corresponding channel marker setting.
query	Returns the current channel marker setting as character data.

Example

```
:SYSTEM:CHMARK ON
```

Sets the channel marker to ON.

When allowed In all functions.

■ Enables and disables, and queries the sound of beeper.

Syntax

command	:SYSTem:BEEPer *A$*
query	:SYSTem:BEEPer?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables and disables the beeper sound.
query	Returns the current enablement state of the beeper sound as character data.

Example

```
:SYSTEM:BEEPER ON
```

Sets the beeper sound to ON.

When allowed In all functions.

■ Sets and queries the list function and the gauge function.

Syntax

command	:SYSTem:LIST *A$*
query	:SYSTem:LIST?
response	*A$*
	A$ = OFF, LIST, GAUGE, L_G (L_G: LIST&GAUGE)

Explanation

command	Sets the list function and the gauge function according to a character string.
query	Returns the current setting for the list function and the gauge function as a character string.

Example

```
:SYSTEM:LIST LIST
```

Sets the list function.

When allowed In all functions.

■ Sets and queries the number of channels used.

Syntax

command	:SYSTem:USECH *A*
query	:SYSTem:USECH?
response	*A*<NR1>
	A = 1, 2, 4

Explanation

command	Sets the number of channels used to a numerical value.
query	Returns the current number of channels used as an NR1 numerical value.

Example

```
:SYSTEM:USECH 4
```

Sets the number of channel used to 4.

When allowed In all functions.

■ Sets and queries the logic waveform's dark/light setting.

Syntax

command	:SYSTem:LOGDraw *A$*
query	:SYSTem:LOGDraw?
response	*A$*
	A$ = DARK, LIGHt

Explanation

command	Sets the darkness of logic waveform display. (DARK/LIGHT)
query	Returns the current darkness of logic waveform display as character data.

Example

```
:SYSTEM:LOGDRAW DARK
```

Sets the logic waveform display darkness to DARK.

When allowed In all functions.

■ Sets and queries the CRT copy size.

Syntax

command	:SYSTem:COPYSize *A$*
query	:SYSTem:COPYSize?
response	*A$*
	A$ = LARGE, SMALL

Explanation

command	Sets the CRT copy size.
query	Returns the current CRT copy size as character data.

Example

```
:SYSTEM:COPYSIZE SMALL
```

Sets the CRT copy size to SMALL.

When allowed In all functions.

SYSTem

■ Sets and queries the SCSI interface device address ID.

Syntax

command	:SYSTem:SCSI *A$,B*
query	:SYSTem:SCSI? *A$*
response	*A$,B*<NR1>

A$ = 8853 : 8853
SCSI : hard disk drive or MO
B = 0 to 7 : device address ID

Explanation

command	Sets the device address ID designated by *A$*.
query	Returns as an NR1 numerical value the setting for the device address ID designated by *A$*.

Example

```
:SYSTEM:SCSI 8853,1
```

Sets the SCSI interface device address ID for the 8853 to 1.

When allowed In all functions.

■ Sets and queries the CRT copy output device for the screen display.

Syntax

command	:SYSTem:COPYPlot *A$*
query	:SYSTem:COPYPlot?
response	*A$*

A$ = PRINter
PLOTter
FD : floppy disk
SCSI : SCSI interface

Explanation

command	Sets the CRT copy output device for the screen display.
query	Returns the CRT copy output device setting as character data.

Example

```
:SYSTEM:COPYPLOT PLOTTER
```

Sets so that the CRT copy is output to the plotter.

When allowed In all functions.

■ Sets and queries the plotter pen.

Syntax

command	:SYSTem:PEN *A$,B*
query	:SYSTem:PEN? *A$*
response	*A$,B*<NR1>

A$ = AREA : waveform decision area
FRAME
CHAR : character
CH1 to CH4
B = 0 to 8 (0: OFF)

Explanation

command	Sets the plotter pen number for the setting designated by *A$*.
query	Returns as an numerical value the pen number setting for the setting designated by *A$*.

Example

```
:SYSTEM:PEN AREA 1
```

Uses the plotter pen 1 to draw the waveform decision area.

When allowed In all functions.

■ Sets and queries the type of CRT copy file output.

Syntax

command	:SYSTem:BMPKind *A$*
query	:SYSTem:BMPKind?
response	*A$*

A$ = MONO : monochrome
COLOR

Explanation

command	Sets the type of CRT copy file (monochrome or color) that is output to floppy disk or the SCSI interface.
query	Returns the file type as character data.

Example

```
:SYSTEM:BMPKIND MONO
```

CRT copies are output as monochrome files.

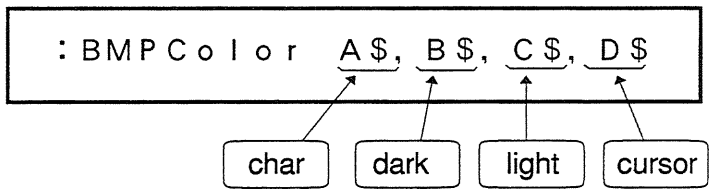
When allowed In all functions

SYSTem

■ Sets and queries the color of CRT copy file output.

Syntax

command	:SYSTem:BMPColor *A$,B$,C$,D$*
query	:SYSTem:BMPColor?
response	*A$,B$,C$,D$*



Refer to Section 12.7, "CRT Copy Output Setting" in the 8853 manual.

A$* to *D$ = BLACK, BLUE, RED, MAGENTA, GREEN, CYAN, YELLOW or ORANGE

Explanation

command	Sets the color used for output of CRT copies to floppy disk or the SCSI interface when color CRT copy output is selected.
query	Returns the color setting as character data.

Example

```
:SYSTEM:BMPCOLOR BLACK,BLUE,RED,CYAN
```

Sets char to black, dark to blue, and light to red, and the cursor to cyan.

When allowed In all functions

■ Sets and queries the FD key.

Syntax

command	:SYSTem:DISKMode *A$*
query	:SYSTem:DISKMode?
response	*A$*

A$ = FD : FD screen
SCSI : SCSI screen
FD_SCSI : FD screen or SCSI screen

Explanation command — Sets the screen that is displayed when the FD key is pressed.

When allowed In all functions

9. SCALing command (Sets and queries scaling)

■ Enables and disables, and queries the scaling function.

Syntax

command	:SCALing:SET *ch$,A$*
query	:SCALing:SET? *ch$*
response	*ch$,A$*

ch$ = CH1 to CH4

A$ = OFF

SCI : conventional scientific floating-point notation

ENG : floating-point notation using powers of 1000

Explanation

command	Enables or disables the scaling function.
query	Returns the current state of enablement of the scaling function as character data.

Example

```
:SCALING:SET CH1,SCI
```

Sets the scaling function for channel 1 to SCI.

When allowed In all functions.

■ Sets and queries the scaling conversion value.

Syntax

command	:SCALing:VOLT *ch$,A*
query	:SCALing:VOLT? *ch$*
response	*ch$,A*<NR3>

ch$ = CH1 to CH4

A = scaling conversion value (EU/V)

(-9.999E+9 to +9.999E+9)

Explanation

command	Sets the scaling conversion value for the channel designated by *ch$*.
query	Returns the scaling conversion value for the channel designated by *ch$* as an NR3 numerical value.

Example

```
:SCALING:VOLT CH1,+2.0E-3
```

Sets the scaling conversion value (EU/V) for channel 1 to +2.0E-3.

When allowed In all functions.

SCALing

■ Sets and queries the scaling offset.

Syntax

command :SCALing:OFFSet *ch$,A*
query :SCALing:OFFSet? *ch$*
response *ch$,A*<NR3>
ch$ = CH1 to CH4
A = scaling offset (EU offset)
(-9.999E+9 to +9.999E+9)

Explanation

command: Sets the scaling offset for the channel designated by *ch$*.
query: Returns the scaling offset for the channel designated by *ch$* as an NR3 numerical value.

Example

```
:SCALING:OFFSET CH1,+1.0E-3
```

Sets the scaling offset (EU offset) for channel 1 to +1. 0E-3.

When allowed In all functions.

■ Sets and queries the scaling unit.

Syntax

command :SCALing:UNIT *ch$,'A$'*
query :SCALing:UNIT? *ch$*
response *ch$,"A$"*
ch$ = CH1 to CH4
A$ = scaling unit (within 7 characters)

Explanation

command: Sets the scaling unit for the channel designated by *ch$*. Double quotation marks (") can be used instead of single quotation marks (').

Characters that can be used are shown below (characters other than the below are replaced by spaces).

A to Z	a to z	+	−	*
/	%	=	~.(=•)	(space)
^2 (=^{2})	^3 (=^{3})	~u (=μ)	~o (=Ω)	~c (=°)

query: Returns the scaling unit for the channel designated by *ch$* as character string data.

Example

```
:SCALING:UNIT CH1,'mA'
```

Sets the scaling unit for channel 1 to milliamps.

When allowed In all functions.

10. COMMent command (Sets and queries comments)

■ Enables and disables, and queries title comments, and inputs comment characters.

Syntax

command :COMMent:TITLe *A$,'B$'*
query :COMMent:TITLe?
response *A$,"B$"*
A$ = OFF, ON
B$ = comment characters (up to 20 characters)

Explanation

command: Enables and disables comments, and inputs a string of comment characters. (Comments may be omitted.)

Characters that can be used are shown below (characters other than the below are replaced by spaces).

A to Z	a to z	0 to 9	+	−
*	/	%	=	(
)	#	&	.	^
,	~u (= μ)	~c (= °)	(space)	

Double quotation marks (") can be used instead of single quotation marks (').

query: Returns the current enablement state of title comments, and the characters of the comment if any, as character data.

Example

```
:COMMENT:TITLE ON,"HIOKI 8853"
```

Inputs "HIOKI 8853" as a title comment.

When allowed

In all functions.

COMMent

■ For each channel, enables and disables and queries comments, and inputs comment characters.

Syntax

command	:COMMent:CH *ch$,A$,'B$'*
query	:COMMent:CH? *ch$*
response	*ch$,A$,"B$"*

ch$ = CH1 to CH4, CHA to CHD
A$ = OFF, ON
B$ = comment characters (up to 20 characters)

Explanation

command

Enables and disables comments for the channel specified by *ch$*, and inputs a string of comment characters. (Comments may be omitted.)
Characters that can be used are the same as in :TITLe.
Double quotation marks (") can be used instead of single quotation marks (').

query

Returns the enablement state of comments for the channel specified by *ch$*, and the characters of the comment if any, as character data.

Example

```
:COMMENT:CH CH1,ON,'ch1=TEST'
```

Sets the comment display for channel 1 to "ch1=TEST".

When allowed In all functions.

11. CALCulate command (Calculation setting and querying)

■ Enables and disables, and queries waveform processing calculation.

Syntax

command	:CALCulate:WVCALc *A$*
query	:CALCulate:WVCALc?
response	*A$*

A$ = OFF, ON, EXEC (execute)

Explanation

command	Enables or disables, according to character data, the execution of waveform processing calculation.
query	Returns, as character data, whether execution of waveform processing calculation is enabled or disabled. Only valid when execution (EXEC) is enabled.

Example

```
:CALCULATE:WVCALC ON
```

Sets the waveform processing calculation to ON.

When allowed In the memory recorder function.

■ Sets and queries the coefficients for the waveform processing calculation equation for Z1.

Syntax

command	:CALCulate:Z1 *A$,B$,C$,D$*
query	:CALCulate:Z1?
response	*A$,B$,C$,D$*

A$, B$, C$ = A to P

D$ = PLUS : +
MINUs : -
MULTi : *
DIVI : /

A$, B$, C$, D$ are used to set up the calculation equation for Z1 in the following way:

Z1 = *A$* X1 *D$* *B$* Y1 + *C$*

(Syntax of :Z2 to :Z4 commands are same as the :Z1 command.)

Explanation

command	Sets the coefficients for the waveform processing calculation equation for Z1 according to the character data.
query	Returns the current coefficients for the waveform processing calculation equation for Z1 as character data.

Example

```
:CALCULATE:Z1 A,B,C,PLUS
```

Sets up the calculation equation for Z1 to be Z1 = aX1+bY1+c.

When allowed In the memory recorder function.

CALCulate

■ Sets and queries the coefficients for the waveform processing calculation equation for Z2.

(For details, refer to the explanation for the :Z1 command.)

Syntax

command	:CALCulate:Z2 *A$,B$,C$,D$*
query	:CALCulate:Z2?
response	*A$,B$,C$,D$*

Explanation

command	Sets the coefficients for the waveform processing calculation equation for Z2 according to the character data.
query	Returns the current coefficients for the waveform processing calculation equation for Z2 as character data.

When allowed In the memory recorder function.

■ Sets and queries the coefficients for the waveform processing calculation equation for Z3.

(For details, refer to the explanation for the :Z1 command.)

Syntax

command	:CALCulate:Z3 *A$,B$,C$,D$*
query	:CALCulate:Z3?
response	*A$,B$,C$,D$*

Explanation

command	Sets the coefficients for the waveform processing calculation equation for Z3 according to the character data.
query	Returns the current coefficients for the waveform processing calculation equation for Z3 as character data.

When allowed In the memory recorder function.

■ Sets and queries the coefficients for the waveform processing calculation equation for Z4.

(For details, refer to the explanation for the :Z1 command.)

Syntax

command	:CALCulate:Z4 *A$,B$,C$,D$*
query	:CALCulate:Z4?
response	*A$,B$,C$,D$*

Explanation

command	Sets the coefficients for the waveform processing calculation equation for Z4 according to the character data.
query	Returns the current coefficients for the waveform processing calculation equation for Z4 as character data.

When allowed In the memory recorder function.

■ Sets up and queries the calculation equation for X1.

Syntax

command :CALCulate:X1 *A$,ch$,B$*
query :CALCulate:X1?
response *A$,ch$,B$*

A$* = OFF : (in this case, *ch$ and *B$* are disregarded)
PAR : (
ABS : Absolute value
EXP : Exponential
LOG : Common logarithm
SQR : Square root
MOV : Moving average
DIF : Differentiation once
INT : Integration once
DIF2 : Differentiation twice
INT2 : Integration twice
SLI : Parallel displacement

ch$ = CH1 to CH4

B$ = A to P
1 to 4000 (when *A$* is set to MOV)
-4000 to 4000 (when *A$* is set to SLI)

A$*, *B$, and *ch$* are used to set up the calculation equation in the following way:

X1 = *A$* (*ch$* +*B$*)

or, when *A$* is set to MOV or SLI:

X1 = MOV or SLI (*ch$*,*B$*)

(Refer to Section 11.2.2, "Calculation Setting" in the 8853 manual.)

(Syntax of the :X2 , :X3, and :X4 commands are same as :X1 command except "*ch$*".)

Explanation

command — Sets the X1 calculation equation for the waveform processing calculation equation for Z1 according to the character or numerical data.

query — Returns the current X1 calculation equation for the waveform processing calculation equation for Z1 as character or numerical data.

Example 1

```
CALCULATE:X1 ABS,CH1,A
```

Sets up the calculation equation for X1 to be X1 = ABS(ch1 + a).

Example 2

```
CALCULATE:X1 MOV,CH1,50
```

Sets up the calculation equation for X1 to be X1 = MOV (ch1,50).

When allowed

In the memory recorder function.

CALCulate

■ Sets up and queries the calculation equation for X2.

(For details, refer to the explanation for the :X1 command.)

Syntax

command	:CALCulate:X2 *A$,ch$,B$*
query	:CALCulate:X2?
response	*A$,ch$,B$*
	ch$ = CH1 to CH4, Z1

Explanation

command	Sets the X2 calculation equation for the waveform processing calculation equation for Z2 according to the character or numerical data.
query	Returns the current X2 calculation equation for the waveform processing calculation equation for Z2 as character or numerical data.

When allowed In the memory recorder function.

■ Sets up and queries the calculation equation for X3.

(For details, refer to the explanation for the :X1 command.)

Syntax

command	:CALCulate:X3 *A$,ch$,B$*
query	:CALCulate:X3?
response	*A$,ch$,B$*
	ch$ = CH1 to CH4, Z1, Z2

Explanation

command	Sets the X3 calculation equation for the waveform processing calculation equation for Z3 according to the character or numerical data.
query	Returns the current X3 calculation equation for the waveform processing calculation equation for Z3 as character or numerical data.

When allowed In the memory recorder function.

■ Sets up and queries the calculation equation for X4.

(For details, refer to the explanation for the :X1 command.)

Syntax

command	:CALCulate:X4 *A$,ch$,B$*
query	:CALCulate:X4?
response	*A$,ch$,B$* *ch$* = CH1 to CH4, Z1 to Z3

Explanation

command	Sets the X4 calculation equation for the waveform processing calculation equation for Z4 according to the character or numerical data.
query	Returns the current X4 calculation equation for the waveform processing calculation equation for Z4 as character or numerical data.

When allowed In the memory recorder function.

CALCulate

■ Sets up and queries the calculation equation for Y1.

Syntax

command :CALCulate:Y1 *A$,ch$,B$*
query :CALCulate:Y1?
response *A$,ch$,B$*

A$* = OFF : (in this case, *ch$ and *B$* are disregarded)
PAR : (
ABS : Absolute value
EXP : Exponential
LOG : Common logarithm
SQR : Square root
MOV : Moving average
DIF : Differentiation once
INT : Integration once
DIF2 : Differentiation twice
INT2 : Integration twice
SLI : Parallel displacement

ch$ = CH1 to CH4
B$ = A to P
1 to 4000 (when *A$* is set to MOV)
-4000 to 4000 (when *A$* is set to SLI)

A$, B$, and *ch$* are used to set up the calculation equation in the following way:

Y1 = *A$* (*ch$* + *B$*)

or, when *A$* is set to MOV or SLI:

Y1 = MOV or SLI (*ch$,B$*)

(Refer to Section 11.2.2, "Calculation Setting" in the 8853 manual.)

(Syntax of the :Y2, Y3, and :Y4 commands are same as :Y1 command except "*ch$*.")

Explanation

command — Sets the Y1 calculation equation for the waveform processing calculation equation for Z1 according to the character or numerical data.

query — Returns the current Y1 calculation equation for the waveform processing calculation equation for Z1 as character or numerical data.

Example 1

```
:CALCULATE:Y1 ABS,CH1,A
```

Sets up the calculation equation for Y1 to be Y1 = ABS (ch1 + a)

Example 2

```
:CALCULATE:Y1 MOV,CH1,50
```

Sets up the calculation equation for Y1 to be Y1 = MOV (ch1,50)

When allowed

In the memory recorder function.

Sets up and queries the calculation equation for Y2.

(For details, refer to the explanation for the :Y1 command.)

Syntax

command	:CALCulate:Y2 *A$,ch$,B$*
query	:CALCulate:Y2?
response	*A$,ch$,B$* *ch$* = CH1 to CH4, Z1

Explanation

command	Sets the Y2 calculation equation for the waveform processing calculation equation for Z2 according to the character or numerical data.
query	Returns the current Y2 calculation equation for the waveform processing calculation equation for Z2 as character or numerical data.

When allowed In the memory recorder function.

Sets up and queries the calculation equation for Y3.

(For details, refer to the explanation for the :Y1 command.)

Syntax

command	:CALCulate:Y3 *A$,ch$,B$*
query	:CALCulate:Y3?
response	*A$,ch$,B$* *ch$* = CH1 to CH4, Z1, Z2

Explanation

command	Sets the Y3 calculation equation for the waveform processing calculation equation for Z3 according to the character or numerical data.
query	Returns the current Y3 calculation equation for the waveform processing calculation equation for Z3 as character or numerical data.

When allowed In the memory recorder function.

CALCulate

■ Sets up and queries the calculation equation for Y4.

(For details, refer to the explanation for the :Y1 command.)

Syntax

command	:CALCulate:Y4 *A$,ch$,B$*
query	:CALCulate:Y4?
response	*A$,ch$,B$* *ch$* = CH1 to CH4, Z1 to Z3

Explanation

command	Sets the Y4 calculation equation for the waveform processing calculation equation for Z4 according to the character or numerical data.
query	Returns the current Y4 calculation equation for the waveform processing calculation equation for Z4 as character or numerical data.

When allowed In the memory recorder function.

■ Sets and queries numerical values for coefficients a to p of the waveform processing calculation equation.

Syntax

command	:CALCulate:FACTor *A$,B*
query	:CALCulate:FACTor? *A$*
response	*A$,B*<NR3> *A$* = A to P *B* = -9.999E+9 to +9.999E+9

Explanation

command	Sets to the given numerical value the one of the coefficients a to p which is designated in *A$*.
query	Returns as an NR 3 numerical value the current value of that one of the coefficients a to p which is designated in *A$*.

Example

```
:CALCULATE:FACTOR A,+1.234E+1
```

Sets the coefficient a to be equal to +1.234E+1.

When allowed In the memory recorder function.

■ Sets and queries the display channel for the calculated result of the waveform processing calculation equation for Z1.

Syntax

command :CALCulate:Z1DIsplay *ch$,A$,upper,lower*
query :CALCulate:Z1DIsplay?
response *ch$,A$,upper,lower*
ch$ = CH1 to CH4, NONE
A$ = AUTO, MANUal
upper, lower = -9.999E+9 to +9.999E+9
(if *A$* = AUTO, *upper* and *lower* may be omitted.)

(Syntax of :Z2DIsplay to :Z4DIsplay commands are same as :Z1DIsplay.)

Explanation

command: Displays the calculated result of the waveform processing calculation equation for Z1 on the channel designated by *ch$* within the range from lower to upper (unit volts - however, if scaling is being performed, in those units).

query: Returns the currently set display channel, scale setting lower limit, and upper limit for display of the calculated result of the waveform processing calculation equation for Z1.

Example

```
:CALCULATE:Z1DISPLAY CH1,MANUAL,+5.000E+0,+0.000E+0
```

Displays the calculated result of the waveform processing calculation equation for Z1 on channel 1 within the range from 0 volts to 5 volts.

When allowed In the memory recorder function.

■ Sets and queries the display channel for the calculated result of the waveform processing calculation equation for Z2.

(For details, refer to the explanation for the :Z1DIsplay command.)

Syntax

command :CALCulate:Z2DIsplay *ch$,A$,upper,lower*
query :CALCulate:Z2DIsplay?
response *ch$,A$,upper,lower*

Explanation

command: Displays the calculated result of the waveform processing calculation equation for Z2 on the channel designated by *ch$* within the range from lower to upper (unit volts - however, if scaling is being performed, in those units).

query: Returns the currently set display channel, scale setting lower limit, and upper limit for display of the calculated result of the waveform processing calculation equation for Z2.

When allowed In the memory recorder function.

CALCulate

■ Sets and queries the display channel for the calculated result of the waveform processing calculation equation for Z3.

(For details, refer to the explanation for the :Z1DIsplay command.)

Syntax

command	:CALCulate:Z3DIsplay *ch$,A$,upper,lower*
query	:CALCulate:Z3DIsplay?
response	*ch$,A$,upper,lower*

Explanation

command	Displays the calculated result of the waveform processing calculation equation for Z3 on the channel designated by *ch$* within the range from lower to upper (unit volts - however, if scaling is being performed, in those units).
query	Returns the currently set display channel, scale setting lower limit, and upper limit for display of the calculated result of the waveform processing calculation equation for Z3.

When allowed In the memory recorder function.

■ Sets and queries the display channel for the calculated result of the waveform processing calculation equation for Z4.

(For details, refer to the explanation for the :Z1DIsplay command.)

Syntax

command	:CALCulate:Z4DIsplay *ch$,A$,upper,lower*
query	:CALCulate:Z4DIsplay?
response	*ch$,A$,upper,lower*

Explanation

command	Displays the calculated result of the waveform processing calculation equation for Z4 on the channel designated by *ch$* within the range from lower to upper (unit volts - however, if scaling is being performed, in those units).
query	Returns the currently set display channel, scale setting lower limit, and upper limit for display of the calculated result of the waveform processing calculation equation for Z4.

When allowed In the memory recorder function.

■ Enables and disables, and queries waveform parameter calculation.

Syntax

command	:CALCulate:MEASure *A$*
query	:CALCulate:MEASure?
response	*A$*
	A$ = OFF, ON, EXEC (execute)

Explanation

command	Enables or disables, according to character data, the execution of waveform parameter calculation.
query	Returns, as character data, whether execution of waveform parameter calculation is enabled or disabled. Only valid when execution (EXEC) is enabled.

Example

```
:CALCULATE:MEASURE ON
```

Sets the waveform parameter calculation to ON.

When allowed In the memory recorder function.

■ Sets and queries waveform parameter calculation value output device.

Syntax

command	:CALCulate:MEASPrint *A$*
query	:CALCulate:MEASPrint?
response	*A$*
	A$ = OFF : no output
	PRINter
	FD : floppy disk
	SCSI : SCSI interface

Explanation

command	Sets the output device of waveform parameter calculation values according to character data.
query	Returns the output device of waveform parameter calculation values as character data.

Example

```
:CALCULATE:MEASPRINT PRINTER
```

Sets so that the waveform processing calculation result is output to the printer.

When allowed In the memory recorder function.

CALCulate

■ Sets and queries waveform parameter calculations.

Syntax

command :CALCulate:MEASSet *NO$,A$,ch$*
query :CALCulate:MEASSet? *NO$*
response *NO$,A$,ch$*

NO$ = *NO1* to *NO4*
A$ = OFF
MIN : minimum value
MAX : maximum value
MINT : time to minimum value
MAXT : time to maximum value
PP : peak value
AVE : average value
RMS : effective value
AREA : area value
PERI : period
FREQ : frequency
RISE : rise time
FALL : fall time
XYAREA : X-Y area value
ch$ = CH1 to CH4, ALL
[During *A$* = XYAREA]
ch$ = x-axis channel, y-axis channel

Explanation

command: Sets the channel and the calculation item of the waveform parameter calculation designated by *NO$*.
query: Returns the channel and the calculation item of the waveform parameter calculation designated by *NO$*.

Example 1

```
:CALCULATE:MEASSET NO1,MAX,CH1
```

Sets the calculation to be of the maximum value on channel 1.

Example 2

```
:CALC:MEASS NO2,XYAREA,CH1,CH2
```

If the x-axis is channel 1 and the y-axis is channel 2, sets X-Y area value calculation.

When allowed

In the memory recorder function.

■ Queries result of waveform parameter calculation.

Syntax

query :CALCulate:ANSWer? *NO$,ch$*
response *A$,B*<NR 3>
NO$ = NO1 to NO4
ch$ = CH1 to CH4
A$ = OFF, AVE, RMS, PP, MAX, MAXT, MIN, MINT, AREA, PERI, FREQ, RISE, FALL, XYAREA
NONE : no calculation result
B = calculation result

Explanation

query Returns the calculation result for the waveform parameter calculation item and result specified by *NO$* and *ch$*.
When *A$* is "NONE", there is no calculation result.

Example

query
```
CALCULATE:ANSWER? NO1,CH1
```
Queries the calculation result of NO1 for the channel 1.
response
```
CALCULATE:ANSWER MIN,-1.2345E-2
```
(When headers are on)

When allowed In the memory recorder function.

■ Enables and disables, and queries decision for waveform parameter calculation.

Syntax

command :CALCulate:COMP *NO$,A$*
query :CALCulate:COMP? *NO$*
response *NO$,A$*
NO$ = NO1 to NO4
A$ = OFF, ON

Explanation

command Enables and disables, according to the character data, the decision of waveform parameter calculation.
query Returns, as character data, the enablement state of the decision of waveform parameter.

Example
```
:CALCULATE:COMP NO1,ON
```
Sets the decision of the calculation result of NO1 to ON.

When allowed In the memory recorder function.

CALCulate

■ Sets and queries upper and lower limits for the decision on the waveform parameter.

Syntax

command	:CALCulate:COMPArea *NO$,upper,lower*
query	:CALCulate:COMPArea? *NO$*
response	*NO$,upper,lower*
	NO$ = NO1 to NO4
	upper, lower = -9.999E+9 to +9.999E+9

Explanation

command: Sets, according to the numerical values supplied, the upper limit value and the lower limit value used when performing a decision on the waveform parameter designated by *A$*.

query: Returns, as NR 3 numerical values, the upper limit and the lower limit used when performing a decision on the waveform parameter designated by *A$*.

Example

```
:CALCULATE:COMPAREA NO1,+1.000E+0,-1.000E+0
```

Sets the decision value for the waveform parameter calculation NO1 to be in the range -1.000E+0 < NO1 < +1.000E+0

When allowed

In the memory recorder function.

12. DISK command (Setting and querying relating to the floppy disk drive, and the hard disk drive, and the magneto optical)

■ Enables and disables, and querying the floppy disk screen, and the SCSI control screen.

Syntax

command :DISK:MODE *A$*
query :DISK:MODE?
response *A$*

A$ = FD : floppy disk (FD) control screen
SCSI : small computer system interface (SCSI) control screen
OFF : except FD or SCSI screen

Explanation

command: Enters the FD or SCSI screen.
query: Returns the currently set screen as character data.

Example

```
:DISK:MODE FD
```

Enters the floppy disk control screen.

When allowed In all functions.

■ Saves a file.

Syntax

command :DISK:SAVE '*NAME1$. NAME2$*',*A$*,*B$* (when *A$* = Wave)
:DISK:SAVE '*NAME1$. NAME2$*',*A$* (when *A$* = Func or Area)

NAME1$ = file name (within 8 characters)
NAME2$ = extension (within 3 characters)
A$ = type of saved information
Wave : measurement data (MEM, FFT)
Func : conditions of creation
Area : waveform decision area (MEM, FFT)
B$* = saved channels (only when *A$ = Wave)
ALL, CH1 to CH4

Explanation

command: Saves the information specified by *A$*.
If an attempt is made to save to a filename that already exists, an execution error is generated.
Double quotation marks (") can be used instead of single quotation marks (').

Example

```
:DISK:SAVE 'TEST. DAT', WAVE, ALL
```

Saves all channels of measurement data under the file name 'TEST. DAT'.

When allowed When the FD or SCSI screen is displayed.

DISK

■ Loads a file.

Syntax command :DISK:LOAD *NO* (,*ch$*)
NO = file number
ch$ = CH1 to CH4 (effective when a file is WAVE.)

Explanation command Loads the data in the file numbered *NO.*
When a file is WAVE (measurement data), and a saved channel (refer to the :DISK:SAVE command.) is CH1 to CH4, the measurement data is loaded to the channel specified by *ch$*.
When *ch$* is omitted, it is loaded to the saved channel.

Example
```
:DISK:LOAD 1
```
Loads the data of the file numbered 1.

When allowed When the FD or SCSI screen is displayed.

■ Deletes a file.

Syntax command :DISK:DELEte *NO*
NO = file number

Explanation command Deletes the file whose number is specified by *NO.*

Example
```
:DISK:DELETE 1
```
Deletes the file of the file numbered 1.

When allowed When the FD or SCSI screen is displayed.

■ Formats a floppy disk, a hard disk, or a magneto optical disk.

Syntax command :DISK:FORMat (*A$*)
A$ = 2HD, 2HC (effective only when FD is 2HD)

Explanation command Formats a floppy disk, a hard disk or a magneto optical disk. Selects either, 2HD (1.2 Mbyte) or 2HC (1.44 Mbyte) format for 2HD floppy disks.

When allowed When the FD or SCSI screen is displayed.

■ Creates a directory on the hard disk or the magneto optical disk.

Syntax

command :DISK:MKDIR '*NAME$*'

"*NAME$*" = subdirectory name (up to 12 characters)

Explanation

command Creates a subdirectory in the current directory on the hard disk or the magneto optical disk.

Double quotation marks (") can be used instead of single quotation marks (').

Example

```
:DISK:MODE SCSI
:DISK:MKDIR 'TEST'
```

Creates a subdirectory 'TEST'.

When allowed When the SCSI screen is displayed.

■ Changes the current directory on the hard disk or the magneto optical disk.

Syntax

command :DISK:CHDIR *NO*

NO = file number (directory)

Explanation

command Changes the current directory to the directory specified by *NO* on the hard disk or the magneto optical disk.

When allowed When the SCSI screen is displayed.

■ Queries the current directory on the hard disk or the magneto optical disk.

Syntax

query :DISK:DIR?

response *A$*

A$ = directory name

Explanation

query Returns the current directory name on the hard disk or the magneto optical disk as character data.

When allowed When the SCSI screen is displayed.

DISK

■ Queries information about a file.

Syntax

query :DISK:INFOr? *'NAME$'*

response *"NAME$",A,B$,"DATE$","TIME$",C* (file)

"NAME$",A,"DATE$","TIME$" (directory)

NAME$ = file name

A = file number (if no such file exists, -1)

B$ = type of information saved:

WAVE : measurement data

FUNC : conditions of creation

AREA : waveform decision area

N : no such file

DATE$ = date of save "year-month-day"

TIME$ = time of save "hour:minute:second"

C = size of file

Explanation

query Returns information about the file whose name is specified in *NAME$*.

If no such file exists, returns as the following way:

"NAME$",-1,N,"----","-:-:-",0

Double quotation marks (") can be used instead of single quotation marks (').

When allowed When the FD or SCSI screen is displayed.

■ Queries the filename.

Syntax

query :DISK:NINFor? *NO*

response *NO,"NAME$"*

NO = file number

NAME$ = file name

Explanation

query Returns the filename of the file whose number is specified in *NO*.

Example

query `:DISK:NINFOR? 1`

response `:DISK:NINFOR 1,"TEST.DAT"`

When allowed When the FD or SCSI screen is displayed.

■ Queries the number of files.

Syntax

query	:DISK:FILE?
response	*A*<NR1>

A = number of files

Explanation

query

Returns the total number of files which are currently saved on the floppy disk.
Returns the number of files (including directories) in the current directory on the hard disk or the magneto optical disk.

When allowed

When the FD or SCSI screen is displayed.

■ Queries the allowable number of clusters for the floppy disk, the hard disk or the magneto optical disk.

Syntax

query	:DISK:FREE?
response	*A$*

A$ = allowable number of clusters

Explanation

query

Returns the allowable number of clusters for the floppy disk or the hard disk or the magneto optical disk.

When allowed

When the FD or SCSI screen is displayed.

13 GRAPh command (Commands relating to graphics editor)

■ Enables and disables, and queries the enablement of the graphics editor.

Syntax

command	:GRAPh:EDIT *A$*
query	:GRAPh:EDIT?
response	*A$*
	A$ = OFF, ON

Explanation

command	Enables and disables the graphic editor mode.
query	Returns whether or not the graphic editor mode is enabled as character data.

Example

```
:GRAPH:EDIT ON
```

Sets the graphic editor mode to ON.

When allowed In the memory recorder function (SINGLE, XY format) and the FFT function (SINGLE format).

■ Loads a waveform into the editor.

Syntax

command	:GRAPh:STORage

Explanation

command	Loads a waveform displaying on the screen into the editor.

When allowed In the memory recorder function and the FFT function, when in the graphics editor mode.

■ Parallel Command

Syntax

command	:GRAPh:PARAllel *high,low,right,left*
	high = 0 to 9.960 (div)
	low = 0 to 9.960 (div)
	right = 0 to 14.975 (div)
	left = 0 to 14.975 (div)

Explanation

command	Carries out a parallel movement of the drawing, creates the decision area. The *high* and *low* parameters are set in units of 0.04 steps, and the *right* and *left* parameters in units of 0.025 steps.

When allowed In the memory recorder function and the FFT function, when in the graphics editor mode.

Line command

Syntax command :GRAPh: LINE *X1,Y1,X2,Y2*

X1, *X2* = x-coordinates

Y1, *Y2* = y-coordinates

Explanation command Draws a line from (*X1,Y1*) to (*X2,Y2*).

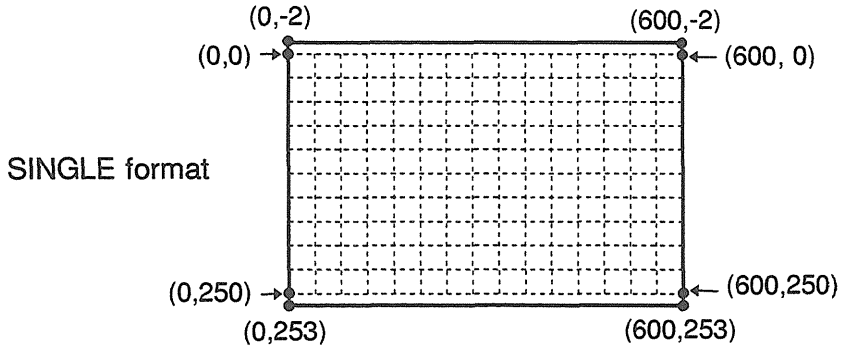
Example `:GRAPH: LINE 10, 20, 100, 200`

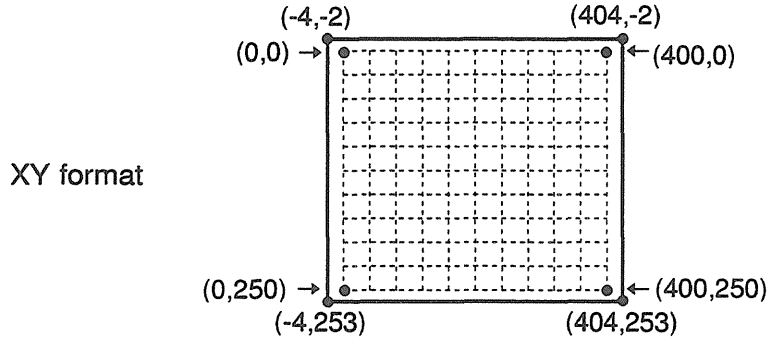
Draws a line from (10,20) to (100,200).

When allowed In the memory recorder function and the FFT function, when in the graphics editor mode.

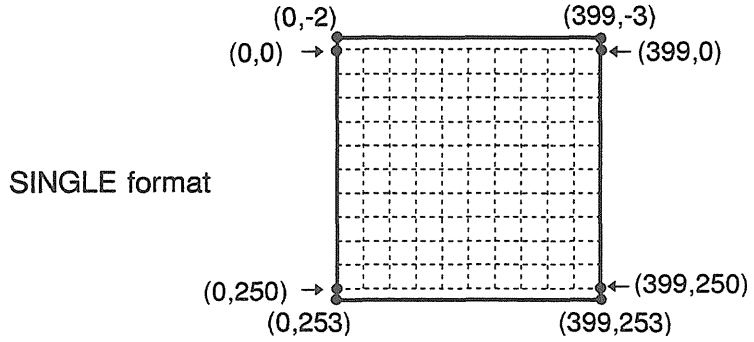
The x- and y-coordinates

- In the memory recorder function





- In the FFT function



GRAPh

■ Paint command

Syntax command :GRAPh:PAINT *X,Y*
X = x-coordinate
Y = y-coordinate

Explanation command Paints the enclosed plane surrounding the point specified by X and Y.
Refer to the :GRAPh:LINE command for details of X and Y coordinates.

When allowed In the memory recorder function and the FFT function, when in the editor mode.

■ The reverse command.

Syntax command :GRAPh:REVErse

Explanation command Reverses black and white in the prepared area.

When allowed In the memory recorder function and the FFT function, when in the editor mode.

■ Erase command

Syntax command :GRAPh:ERASe *X1,Y1,X2,Y2*
X1, X2 = x-coordinates
Y1, Y2 = y-coordinates

Explanation command Erases the line from (*X1,Y1*) to (*X2,Y2*).
Refer to the :GRAPh:LINE command for details of X and Y coordinates.

When allowed In the memory recorder function and the FFT function, when in the editor mode.

Clear command

Syntax command :GRAPh:CLEAr *X1,Y1,X2,Y2*
X1, *X2* = x-coordinates
Y1, *Y2* = y-coordinates

Explanation command Clears the rectangle with the points (*X1,Y1*) and (*X2,Y2*) at diagonally opposite corners.
Refer to the :GRAPh:LINE command for details of X and Y coordinates.

When allowed In the memory recorder function and the FFT function, when in the editor mode.

The all clear command.

Syntax command :GRAPh:ALLClear

Explanation command Clears the entire drawing.

When allowed In the memory recorder function and the FFT function, when in the editor mode.

Undo command

Syntax command :GRAPh:UNDO

Explanation Cancels the effect of the immediately previous editor command.

When allowed In the memory recorder function and the FFT function, when in the editor mode.

Saves the drawing (decision area)

Syntax command :GRAPh:SAVE

Explanation Saves the decision area created with the editor.

When allowed In the memory recorder function and the FFT function, when in the editor mode.

■ Sets and queries decision area data points.

Syntax

command	:GRAPh:POINt *X,Y,A*
query	:GRAPh:POINt? *X,Y*
response	*X,Y,A* all <NR1>

X = x-coordinate
Y = y-coordinate
A = 0 : for a point outside it
1 : for a point within the decision area

Explanation

command

Creates the decision area by dots.
When A = 1, at the coordinates indicated by X and Y is a point within the decision area.
When A = 0, at the coordinates indicated by X and Y is a point outside it.

query

Returns the value *A* at the coordinates indicated by *X* and *Y*.
Refer to the :GRAPh:LINE command for details of X and Y.

When allowed

In the memory recorder function and the FFT function, when in the editor mode.

Chapter 10
Example Programs

The programs in this chapter run on an IBM-PC(VGA) series computer.

Example 1 Using a setting command

Send the command in the format specified, when the conditions for the command to be acceptable are met.

```
100 ' ------------------------------------
110 ' 8853 Set command
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%            'GP-IB Address = 5
160 CALL SENDIFC(BOAD%)                        'Clear interface
170 CALL ENABLEREMOTE(BOAD%, ADRS%(0))         'Enable remote
180 GOSUB 270                                  'Function MEM
190 GOSUB 270                                  'Time/Div 500us
200 GOSUB 270                                  '15DIV
210 GOSUB 270                                  '< START >
220 CALL ENABLELOCAL(BOAD%, ADRS%(0))          'Enable operations
230 END
240 ' ------------------------------------
250 ' Send data
260 ' ------------------------------------
270 READ COMMAND$
280 CALL SEND(BOAD%, ADRS%(0), COMMAND$, NLEND%)
290 RETURN
300 ' ------------------------------------
310 ' data table
320 ' ------------------------------------
330 DATA ":FUNCTION MEM"
340 DATA ":CONFIGURE:TDIV +500.e-6"
350 DATA ":CONFIGURE:SHOT 15"
360 DATA ":START"
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
160-170	Send interface clear, and switch to remote mode.
180	Select memory recorder function.
190	Time/division is 500 μ s.
200	Recording length is 15 divisions.
210	Enter measurement operation mode.
220	End remote mode.

Example 2 Using a query

(1) Send the query in the format specified, when the conditions for the query to be acceptable are met.
Next switch the 8853 to be the talker, and receive the output data.

(2) The response data from the query is returned in the format specified for the corresponding command.

```
100 ' ------------------------------------
110 ' 8853 Receive command
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
160 CALL SENDIFC(BOAD%)                          'Clear interface
170 CALL ENABLEREMOTE(BOAD%, ADRS%(0))           'Enable remote
180 GOSUB 300                                    'Header OFF
190 GOSUB 300                                    'Read FUNCTION
200 GOSUB 360:ANS$ = READING$
210 GOSUB 300                                    'Read TIME
220 GOSUB 360:TM$ = READING$
230 PRINT ANS$, TM$
240 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%, UNT$)    'UN TALK
250 CALL ENABLELOCAL(BOAD%, ADRS%(0))            'Enable operations
260 END
270 ' ------------------------------------
280 ' Send data
290 ' ------------------------------------
300 READ COMMAND$
310 CALL SEND(BOAD%, ADRS%(0), COMMAND$, NLEND%)
320 RETURN
330 ' ------------------------------------
340 ' Receive data
350 ' ------------------------------------
360 READING$ = SPACE$(30)
370 CALL RECEIVE(BOAD%, ADRS%(0), READING$, STOPEND%)
380 LENGS% = IBCNT% - 1
390 READING$ = LEFT$(READING$, LENGS%)
400 RETURN
410 ' ------------------------------------
420 ' data table
430 ' ------------------------------------
440 DATA ":HEADER OFF"
450 DATA ":FUNCTION?"
460 DATA ":SYSTEM:TIME?"
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
160-170	Send interface clear, and switch to remote mode.
180	Disable headers.
190-200	Ask function, and load into ANS$.
210-220	Ask current time, and load into TM$.
240	Release talker.
250	End remote mode.

Example 3 Using service requests

(1) Using the *SRE and *ESE commands, this program sets the service request response enable, and sets the jump address in the controller for a service request interrupt. It then enables the service request interrupt.

(2) The service request interrupt handling routine uses serial polling to read the 8853 status byte, then carries out appropriate processing depending on the value of the status byte.
It then re-enables the service request interrupt, and returns.

```
100 ' ------------------------------------
110 ' 8853 Service request
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
160 CALL SENDIFC(BOAD%)                          'Clear interface
170 CALL ENABLEREMOTE(BOAD%, ADRS%(0))           'Enable remote
180 ON PEN GOSUB 330
190 SRE$="*SRE 32":ESE$="*ESE 60":SCL$="*CLS"
200 CALL SEND(BOAD%, ADRS%(0), SRE$, NLEND%)     'Mask SRQ
210 CALL SEND(BOAD%, ADRS%(0), ESE$, NLEND%)     'Mask SESER
220 CALL SEND(BOAD%, ADRS%(0), SCL$, NLEND%)     'Clear status byte
230 PEN ON
240 FUN$=":FUNCTION MEM"
250 CALL SEND(BOAD%, ADRS%(0), FUN$, NLEND%)     'Set FUNCTION
260 I% = 0
270 AVR$=":CONFIGURE:AVERAGE "+STR$(I%)
280 CALL SEND(BOAD%, ADRS%(0), AVR$, NLEND%)     'Set AVERAGE
290 I% = I% + 50:GOTO 270
300 ' ------------------------------------
310 ' Service request operation
320 ' ------------------------------------
330 CALL IBRSP(ADRS%, S%)
340 DCL$ = CHR$(DCL%):CALL IBCMD(BOAD%, DCL$)    'Clear buffer
350 PRINT "SQR="; S%
360 CALL SEND(BOAD%, ADRS%(0), SRE$, NLEND%)     'Mask SRQ
370 CALL SEND(BOAD%, ADRS%(0), ESE$, NLEND%)     'Mask SESER
380 PEN ON
390 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%, UNT$)    'UN TALK
400 CALL ENABLELOCAL(BOAD%, ADRS%(0))            'Enable operations
410 END
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
160-170	Send interface clear, and switch to remote mode.
180	Set jump address for service request.
200	Enable bit 5 (ESB) of the status byte by the service request enable register.
210	Enable bits 2, 3, 4, and 5 of the standard event status register by the standard event status enable register.
220	Clear the status byte associated queue.
230	Enable the service request interrupt.
250	Set the function.
280	Set the averaging. (Error source)
330-340	Serial polling to read the status byte.
380	Enable service request interrupt.
390-400	Release talker and remote mode.

Example 4 Outputting stored data

(1) Using the :MEMORY:MAXPOINT? query, this program checks whether data can be output from memory. If this query returns zero, no data is stored, and it cannot therefore be output.
(2) Next, the program specifies the channel and point for output, using the :MEMORY:POINT command. As data is input or output, the point is incremented automatically. If capturing data consecutively, it is sufficient to specify the point once only.
(3) To capture data in ASCII format use the :MEMORY:ADATA? query, and to capture data as voltage values use the :MEMORY:VDATA? query.
The number of data samples which may be output in one set is 1 to 40 using :ADATA? and 1 to 10 using the :VDATA? query.
Note: Outputting data in bigger sets reduces the overall processing time.

Read data (1200 samples) for channel 1 when stored with a 30-division recording length.

```
100 ' ------------------------------------
110 ' 8853 Data out
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 DIM D(1201)
150 ESR$ = ":ESRO?":VDT$ = ":MEMORY:VDATA? 10"
160 BOAD% = 0
170 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
180 CALL SENDIFC(BOAD%)                          'Clear interface
190 CALL ENABLEREMOTE(BOAD%, ADRS%(0))           'Enable remote
200 GOSUB 470                                    'Enable ESRO
210 GOSUB 470                                    'MEM, 30DIV
220 GOSUB 470                                    'Trigger mode SINGLE
230 GOSUB 470                                    '<START>
240 CALL SEND(BOAD%, ADRS%(0), ESR$, NLEND%)
250 GOSUB 530:STS% = VAL(READING$)
260 IF (STS% AND 2) = 0 THEN 240                 '<START> stopped?
270 GOSUB 470                                    'Check STORAGE data
280 GOSUB 530:MAX% = VAL(READING$)
290 IF MAX% <> 1200 THEN 410
300 GOSUB 470                                    'Set point ch1, 0
310 FOR I% = 0 TO MAX% - 10 STEP 10
320 CALL SEND(BOAD%, ADRS%(0), VDT$, NLEND%)
330 GOSUB 530
340 FOR II% = 0 TO 9
350 D(I%+II%) = VAL(MID$(READING$, (12*II%+1), 11))
360 NEXT II%
370 NEXT I%
380 GOSUB 470
390 GOSUB 530:D(1200) = VAL(READING$)            'Last Data
400 FOR I% = 0 TO 1200:PRINT D(I%):NEXT I%       'Print data
410 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%, UNT$)    'UN TALK
420 CALL ENABLELOCAL(BOAD%, ADRS%(0))            'Enable operations
430 END
440 ' ------------------------------------
450 ' Send data
460 ' ------------------------------------
```

```
470 READ COMMAND$
480 CALL SEND(BOAD%, ADRS%(0), COMMAND$, NLEND%)
490 RETURN
500 '  ------------------------------------
510 '  Receive data
520 '  ------------------------------------
530 READING$ = SPACE$(128)
540 CALL RECEIVE(BOAD%, ADRS%(0), READING$, STOPEND%)
550 LENGS% = IBCNT% - 1
560 READING$ = LEFT$(READING$, LENGS%)
570 RETURN
580 '  ------------------------------------
590 '  data table
600 '  ------------------------------------
610 DATA ":ESE0 2"
620 DATA ":FUNCTION MEM;:CONFIGURE:SHOT 30"
630 DATA ":TRIGGER:MODE SINGLE"
640 DATA ":START"
650 DATA ":HEADER OFF;:MEMORY:MAXPOINT?"
660 DATA ":MEMORY:POINT CH1,0"
670 DATA ":MEMORY:VDATA? 1"
```

Line	*Comment*
170	Set ADRS%(0) to address of 8853.
180-190	Send interface clear, and switch to remote mode.
210	Set memory recorder function and 30-division recording length.
230	Enter measurement operation mode.
240-260	Wait for end of measurement operation.
270-280	Disable headers, and read number of stored data samples into MAX%.
300	Set output data to be from channel 1, point 0.
310-370	Set size of output data set to be 10 samples, and read as voltage values.
410-420	Release talker and remote mode.

Example 5 Inputting storage data

(1) Using the :MEMORY:MAXPOINT? query, this program checks whether data can be input to memory. If this query returns zero, the state is such as not to store data, and it cannot therefore be input.

(2) Next, the program specifies the channel and point for input, using the :MEMORY:POINT command, and then uses the :MEMORY:ADATA command to input data.
Note: As with output, it is more efficient to input data in bigger sets.

With the unit storing with a 30-division recording length, write sine wave data into memory for channel 1.

```
100 ' ------------------------------------
110 ' 8853 Data input
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 HEA$ = ":HEADER OFF;:MEMORY:MAXPOINT?"
160 ADT$ = ":MEMORY:ADATA"
170 PNT$ = ":MEMORY:POINT CH1,0"
180 WAV$ = ":DISPLAY:CHANGE DISPLAY"
190 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
200 CALL SENDIFC(BOAD%)                          'Clear interface
210 CALL ENABLEREMOTE(BOAD%, ADRS%(0))           'Enable remote
220 CALL SEND(BOAD%, ADRS%(0), HEA$, NLEND%)     'Header off
230 MXP$ = SPACE$(10)
240 CALL RECEIVE(BOAD%, ADRS%(0), MXP$, STOPEND%) 'Maxpoint?
250 MAX% = VAL(MXP$)
260 IF MAX% <> 1200 THEN 340
270 CALL SEND(BOAD%, ADRS%(0), PNT$, NLEND%)     'Set point CH1,0
280 FOR I% = 0 TO MAX%
290 VOLT% = 1500 * SIN(3.14 * I% / 500) + 2000
300 SND$ = ADT$ + STR$(VOLT%)
310 CALL SEND(BOAD%, ADRS%(0), SND$, NLEND%)
320 NEXT I%
330 CALL SEND(BOAD%, ADRS%(0), WAV$, NLEND%)     'Wave display
340 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%, UNT$)    'UN TALK
350 CALL ENABLELOCAL(BOAD%, ADRS%(0))            'Enable operations
360 END
```

Line	*Comment*
190	Set ADRS%(0) to address of 8853.
200-210	Send interface clear, and switch to remote mode.
220-250	Read maximum number of data samples in memory into MAX%.
270	Set input data to be to channel 1, point 0.
280-320	Write the sine wave.
340-350	Release talker and remote mode.

Example 6 Making storage condition settings

```
100 ' ------------------------------------
110 ' 8853 Sample program No.1
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
160 '
170 CALL SENDIFC(BOAD%)                          'Clear interface
180 CALL ENABLEREMOTE(BOAD%, ADRS%(0))           'Enable remote
190 '
200 GOSUB 410                                    'FUNCTION MEM
210 GOSUB 410                                    'TIME/DIV 1ms
220 GOSUB 410                                    'SHOT 15DIV
230 '
240 GOSUB 410                                    'Trigger source OR
250 GOSUB 410                                    'LEVEL trigger
260 GOSUB 410                                    'Pre-trigger 5%
270 GOSUB 410                                    'LEVEL 60%
280 GOSUB 410                                    'SLOPE UP
290 GOSUB 410                                    'CH2 trigger OFF
300 GOSUB 410                                    'CH3 trigger OFF
310 GOSUB 410                                    'CH4 trigger OFF
320 '
330 GOSUB 410                                    '<START>
340 '
350 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%, UNT$)    'UN TALK
360 CALL ENABLELOCAL(BOAD%, ADRS%(0))            'Enable operations
370 END
380 ' ------------------------------------
390 ' Send data
400 ' ------------------------------------
410 READ COMMAND$
420 CALL SEND(BOAD%, ADRS%(0), COMMAND$, NLEND%)
430 RETURN
440 ' ------------------------------------
450 ' data table
460 ' ------------------------------------
470 DATA ":FUNCTION MEM"
480 DATA ":CONFIGURE:TDIV 1.E-3"
490 DATA ":CONFIGURE:SHOT 15"
500 DATA ":TRIGGER:SOURCE OR"
510 DATA ":TRIGGER:KIND CH1,LEVEL"
520 DATA ":TRIGGER:PRETRIG 5"
530 DATA ":TRIGGER:LEVEL CH1,60"
540 DATA ":TRIGGER:SLOPE CH1,UP"
550 DATA ":TRIGGER:KIND CH2,OFF"
560 DATA ":TRIGGER:KIND CH3,OFF"
570 DATA ":TRIGGER:KIND CH4,OFF"
580 DATA ":START"
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
170-180	Send interface clear, and switch to remote mode.
200-310	Set the 8853 function, trigger conditions, etc.
330	Enter measurement operation mode with the conditions set.
350	Release talker.
360	End remote mode.

Example 7 Start measurement operation mode, and if no trigger is detected execute a STOP.

```
100 ' ------------------------------------
110 ' 8853 Sample program No.2
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
160 CALL SENDIFC(BOAD%)                          'Clear interface
170 CALL ENABLEREMOTE(BOAD%,ADRS%(0))            'Enable remote
180 '
190 GOSUB 520                                    'Enable SESER bit
200 GOSUB 520                                    'TIME/DIV 1ms,SHOT 15DIV
210 GOSUB 520                                    'Trigger source OR
220 GOSUB 520                                    'LEVEL trigger CH1,CH2
230 GOSUB 520                                    'Trigger OFF CH3,CH4
240 GOSUB 520                                    'Trigger CH1,60%,UP
250 GOSUB 520                                    'Trigger CH2,60%,UP
260 GOSUB 520                                    'Trigger MODE SINGLE
270 '
280 GOSUB 520                                    '<START>
290 '
300 ESR$ = ":ESR0?"
310 '
320 FOR W% = 1 TO 100
330 CALL SEND(BOAD%,ADRS%(0),ESR$,NLEND%)
340 GOSUB 580
350 IF (ESR0% AND &H4) <> 0 THEN 410
360 NEXT W%
370 PRINT "Not Trigger"
380 GOSUB 520
390 GOTO 460
400 '
410 CALL SEND(BOAD%,ADRS%(0),ESR$,NLEND%)
420 GOSUB 580
430 IF (ESR0% AND &H2) = 0 THEN 410
440 PRINT "Storage end"
450 '
460 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%,UNT$)     'UN TALK
470 CALL ENABLELOCAL(BOAD%,ADRS%(0))             'Enable operations
480 END
490 ' ------------------------------------
500 ' Send data
510 ' ------------------------------------
520 READ COMMAND$
530 CALL SEND(BOAD%,ADRS%(0),COMMAND$,NLEND%)
540 RETURN
550 ' ------------------------------------
560 ' Receive data
570 ' ------------------------------------
580 READING$ = SPACE$(10)
590 CALL RECEIVE(BOAD%,ADRS%(0),READING$,STOPEND%)
600 ESR0% = VAL(READING$)
610 RETURN
620 ' ------------------------------------
```

```
630 ' data table
640 ' ------------------------------------
650 DATA "*CLS;:ESE0 6;:FUNCTION MEM"
660 DATA ":CONFIGURE:TDIV 1.E-3;SHOT 15"
670 DATA ":TRIGGER:SOURCE OR"
680 DATA ":TRIGGER:KIND CH1,LEVEL;KIND CH2,LEVEL"
690 DATA ":TRIGGER:KIND CH3,OFF;KIND CH4,OFF"
700 DATA ":TRIGGER:LEVEL CH1,60;SLOPE CH1,UP"
710 DATA ":TRIGGER:LEVEL CH2,60;SLOPE CH2,UP"
720 DATA ":TRIGGER:MODE SINGLE"
730 DATA ":START"
740 DATA ":ABORT"
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
160-170	Send interface clear, and switch to remote mode.
190-260	Set the function and trigger conditions. Clear event status register 0. Clear the standard event status register.
280	Enter measurement operation mode.
320-360	At fixed intervals, check whether the trigger has been applied. Read event status register 0, and check if bit 2 is set. When it is, go to line 420.
370-390	If no trigger has been detected, abort measurement.
410-440	If a trigger has been detected, read event status register 0, and check that bit 1 is set, confirming that measurement operation has started.
460-470	Release talker and remote mode.

Example 8 Checking which input units are present, and displaying their input ranges on the screen.

```
100 ' ------------------------------------
110 ' 8853 Sample program No.3
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 SCREEN 9
150 BOAD% = 0
160 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
170 CALL SENDIFC(BOAD%)                          'Clear interface
180 CALL ENABLEREMOTE(BOAD%,ADRS%(0))            'Enable remote
190 HEA$ = ":HEADER OFF"
200 CH1$ = ":MEMORY:AREAL? CH1"
210 CH2$ = ":MEMORY:AREAL? CH2"
220 CH3$ = ":MEMORY:AREAL? CH3"
230 CH4$ = ":MEMORY:AREAL? CH4"
240 CALL SEND(BOAD%,ADRS%(0),HEA$,NLEND%)        'Header OFF
250 CLS
260 LOCATE 3,5:PRINT "<LEVEL MONITOR>"
270 LOCATE 4,1:PRINT "100"
280 LOCATE 13,1:PRINT " 50"
290 LOCATE 22,1:PRINT "  0"
300 LOCATE 1,52:PRINT "CH1            CH2"
310 LOCATE 2,52:PRINT "CH3            CH4"
320 LINE (30,57)-(620,307),7,B,&HCCCC            'Frame
330 FOR Y% = 82 TO 282 STEP 25
340 LINE (30,Y%)-(620,Y%),7,,&H1010
350 NEXT Y%
360 '
370 LINE (440,8)-(490,10),6,B
380 CALL SEND(BOAD%,ADRS%(0),CH1$,NLEND%)        'CH1 ADATA
390 GOSUB 760:Y10% = ADT% / 16
400 LINE (560,8)-(610,10),5,B
410 CALL SEND(BOAD%,ADRS%(0),CH2$,NLEND%)        'CH2 ADATA
420 GOSUB 760:Y20% = ADT% / 16
430 LINE (440,24)-(490,26),4,B
440 CALL SEND(BOAD%,ADRS%(0),CH3$,NLEND%)        'CH3 ADATA
450 GOSUB 760:Y30% = ADT% / 16
460 LINE (560,24)-(610,26),3,B
470 CALL SEND(BOAD%,ADRS%(0),CH4$,NLEND%)        'CH4 ADATA
480 GOSUB 760:Y40% = ADT% / 16
490 '
500 FOR X% = 30 TO 618 STEP 2
510 CALL SEND(BOAD%,ADRS%(0),CH1$,NLEND%)        'CH1 ADATA
520 GOSUB 760:Y11% = ADT% / 16
530 LINE (X%,307-Y10%)-(X%+2,307-Y11%),6
540 Y10% = Y11%
550 CALL SEND(BOAD%,ADRS%(0),CH2$,NLEND%)        'CH2 ADATA
560 GOSUB 760:Y21% = ADT% / 16
570 LINE (X%,307-Y20%)-(X%+2,307-Y21%),5
580 Y20% = Y21%
590 CALL SEND(BOAD%,ADRS%(0),CH3$,NLEND%)        'CH3 ADATA
600 GOSUB 760:Y31% = ADT% / 16
610 LINE (X%,307-Y30%)-(X%+2,307-Y31%),4
620 Y30% = Y31%
```

```
630 CALL SEND(BOAD%,ADRS%(0),CH4$,NLEND%)        'CH4 ADATA
640 GOSUB 760:Y41% = ADT% / 16
650 LINE (X%,307-Y40%)-(X%+2,307-Y41%),3
660 Y40% = Y41%
670 NEXT X%
680 IF INKEY$ = "" GOTO 250
690 SCREEN 0
700 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%,UNT$)     'UN TALK
710 CALL ENABLELOCAL(BOAD%,ADRS%(0))             'Enable operations
720 END
730 ' ------------------------------------
740 ' Receive data
750 ' ------------------------------------
760 READING$ = SPACE$(32)
770 CALL RECEIVE(BOAD%,ADRS%(0),READING$,STOPEND%)
780 ADT% = VAL(READING$)
790 RETURN
```

Line	*Comment*
160	Set ADRS%(0) to address of 8853.
170-180	Send interface clear, and switch to remote mode.
240	Disable headers.
250-350	Screen display.
370-480	Read real time data for the channels into variables.
500-670	Read real time data for the channels and display.
700-710	Release talker and remote mode.

Example 9 Saving stored data onto drive 2 (sequential file)

```
100 ' ------------------------------------
110 ' 8853 Sample program No.4
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
160 HEA$ = ":HEADER OFF;:MEMORY:MAXPOINT?"
170 ADT$ = ":MEMORY:ADATA? 1"
180 CALL SENDIFC(BOAD%)                          'Clear interface
190 CALL ENABLEREMOTE(BOAD%,ADRS%(0))            'Enable remote
200 ON ERROR GOTO 500
210 '
220 CLS:LOCATE 2,10
230 PRINT "< Storage Data SAVE >"
240 PRINT:PRINT
250 CALL SEND(BOAD%,ADRS%(0),HEA$,NLEND%)        'Header OFF
260 GOSUB 590:MAX% = VALUE%                      'Max point?
270 IF MAX% <> 0 THEN 300                        'Output ready?
280 PRINT "No storage data !!"
290 GOTO  520
300 '
310 PRINT " Max point=";MAX%:PRINT
320 INPUT " Channel(CH1-CH4)";CH$                'Input channel No.
330 INPUT " File name";NA$                       'Input (drive)+filename
340 PRINT:PRINT
350 '
360 OPEN NA$ FOR OUTPUT AS #1                    'Open file
370 '
380 PNT$ = ":MEMORY:POINT "+CH$+",0"             'Set output point
390 CALL SEND(BOAD%,ADRS%(0),PNT$,NLEND%)
400 '
410 PRINT #1,MAX%                                'Save max point
420 FOR I% = 0 TO MAX%
430 CALL SEND(BOAD%,ADRS%(0),ADT$,NLEND%)
440 GOSUB 590                                    'Get ADATA
450 PRINT #1,VALUE%                              'Save ADATA
460 NEXT I%
470 PRINT " Completed."
480 GOTO 520
490 '
500 PRINT "ERROR !!"
510 '
520 CLOSE #1                                     'Close file
530 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%,UNT$)     'UN TALK
540 CALL ENABLELOCAL(BOAD%,ADRS%(0))             'Enable operations
550 END
560 ' ------------------------------------
570 ' Receive data
580 ' ------------------------------------
590 READING$ = SPACE$(30)
600 CALL RECEIVE(BOAD%,ADRS%(0),READING$,STOPEND%)
610 VALUE% = VAL(READING$)
620 RETURN
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
180-190	Send interface clear, and switch to remote mode.
200	Set the jump addresses for if an error occurs, to ensure that the program does not exit with the file left open.
250-260	Disable headers, and read the number of stored data values into MAX%.
310-330	Input the channels to be saved and the filename.
390	Set the stored data output point.
410	Write the number of data values saved, at the beginning of the file.
420-460	Read the stored data from the 8853, and save sequentially.
530-540	Release talker and remote mode.

Example 10 Reading the data saved in Example 9, and loading it into the 8853.

```
100 ' ------------------------------------
110 ' 8853 Sample program No.5
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
160 HEA$ = ":HEADER OFF;:MEMORY:MAXPOINT?"
170 DIS$ = ":DISPLAY:CHANGE DISPLAY"
180 CALL SENDIFC(BOAD%)                          'Clear interface
190 CALL ENABLEREMOTE(BOAD%,ADRS%(0))            'Enable remote
200 ON ERROR GOTO 470
210 '
220 CLS:LOCATE 2,10
230 PRINT "< Storage Data LOAD >"
240 PRINT:PRINT
250 INPUT " Channel(CH1-CH4)";CH$                'Input channel No.
260 INPUT " File name";NA$                       'Input (drive)+filename
270 '
280 OPEN NA$ FOR INPUT AS #1                     'Open file
290 '
300 PNT$ = ":MEMORY:POINT "+CH$+",0"             'Set output point
310 CALL SEND(BOAD%,ADRS%(0),PNT$,NLEND%)
320 '
330 INPUT #1,MAX%                                'Load max point
340 CALL SEND(BOAD%,ADRS%(0),HEA$,NLEND%)        'Header OFF
350 GOSUB 560                                    'Max point?
360 IF VALUE% <> MAX% THEN 470                   'Input ready?
370 '
380 FOR I% = 0 TO MAX%
390 INPUT #1,DAT%                                'Load ADATA
400 ADT$ = ":MEMORY:ADATA "+STR$(DAT%)
410 CALL SEND(BOAD%,ADRS%(0),ADT$,NLEND%)        'Set ADATA
420 NEXT I%
430 PRINT " Completed."
440 CALL SEND(BOAD%,ADRS%(0),DIS$,NLEND%)        'Display wave
450 GOTO 490
460 '
470 PRINT "ERROR !!"
480 '
490 CLOSE #1                                     'Close file
500 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%,UNT$)     'UN TALK
510 CALL ENABLELOCAL(BOAD%,ADRS%(0))             'Enable operations
520 END
530 ' ------------------------------------
540 ' Receive data
550 ' ------------------------------------
560 READING$ = SPACE$(30)
570 CALL RECEIVE(BOAD%,ADRS%(0),READING$,STOPEND%)
580 VALUE% = VAL(READING$)
590 RETURN
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
180-190	Send interface clear, and switch to remote mode.
200	Set the jump addresses for if an error occurs, to ensure that the program does not exit with the file left open.
250-260	Specify the filename to be opened and channel.
310	Set the stored data input point.
340-350	Read the number of stored data values into VALUE%.
380-420	Read the data from the file, and write to memory on the 8853.
500-510	Release talker and remote mode.

Example 11 Setting measurement conditions, and starting measurement operation after synchronizing with the *OPC command

```
100 ' ------------------------------------
110 ' 8853 Sample program No.6
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
160 CALL SENDIFC(BOAD%)                          'Clear interface
170 CALL ENABLEREMOTE(BOAD%,ADRS%(0))            'Enable remote
180 ON PEN GOSUB 330
190 GOSUB 450                                    'Mask SRQ
200 GOSUB 450                                    'Mask SESER
210 GOSUB 450                                    'Clear statusbyte
220 PEN ON
230 '
240 GOSUB 450                                    'Set FUNCTION
250 GOSUB 450                                    'TIME/DIV 1ms
260 GOSUB 450                                    'SHOT 15DIV
270 '
280 GOSUB 450                                    'CH1 <- LEVEL TRIG.
290 GOSUB 450                                    'Pre-TRIG. 5%
300 GOSUB 450                                    'LEVEL 60%,SLOPE UP
310 GOTO 310
320 '
330 CALL IBRSP(ADRS%,S%)
340 DCL$ = CHR$(DCL%):CALL IBCMD(BOAD%,DCL$)     'Clear buffer
350 PRINT "START OK "
360 GOSUB 450                                    '< START >
370 '
380 PEN OFF
390 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%,UNT$)     'UN TALK
400 CALL ENABLELOCAL(BOAD%,ADRS%(0))             'Enable operations
410 END
420 ' ------------------------------------
430 ' Send data
440 ' ------------------------------------
450 READ COMMAND$
460 CALL SEND(BOAD%,ADRS%(0),COMMAND$,NLEND%)        'Mask SRQ
470 RETURN
480 ' ------------------------------------
490 ' DATA table
500 ' ------------------------------------
510 DATA "*SRE 32"
520 DATA "*ESE 1"
530 DATA "*CLS"
540 DATA ":FUNCTION MEM"
550 DATA ":CONFIGURE:TDIV 1.E-3"
560 DATA ":CONFIGURE:SHOT 15"
570 DATA ":TRIGGER:KIND CH1,LEVEL"
580 DATA ":TRIGGER:PRETRIG 5"
590 DATA ":TRIG:LEVEL CH1,60;SLOPE CH1,UP;*OPC"
600 DATA ":START"
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
160-170	Send interface clear, and switch to remote mode.
180	Set jump address for if a service request is received.
190	Enable bit 5 (ESB) of the status byte by the service request enable register.
200	Enable bit 0 of the standard event status register by the standard event status enable register.
210	Clear the status byte associated queue.
220	Enable the service request interrupt.
240-300	Set the measurement conditions.
310	Wait for a service request.
330-340	Serial polling to read the status byte.
360	After confirming the completion of condition setting, start measurement operation.
380	Disable service request interrupt.
390-400	Release talker and remote mode.

Example 12 Using service requests to display errors

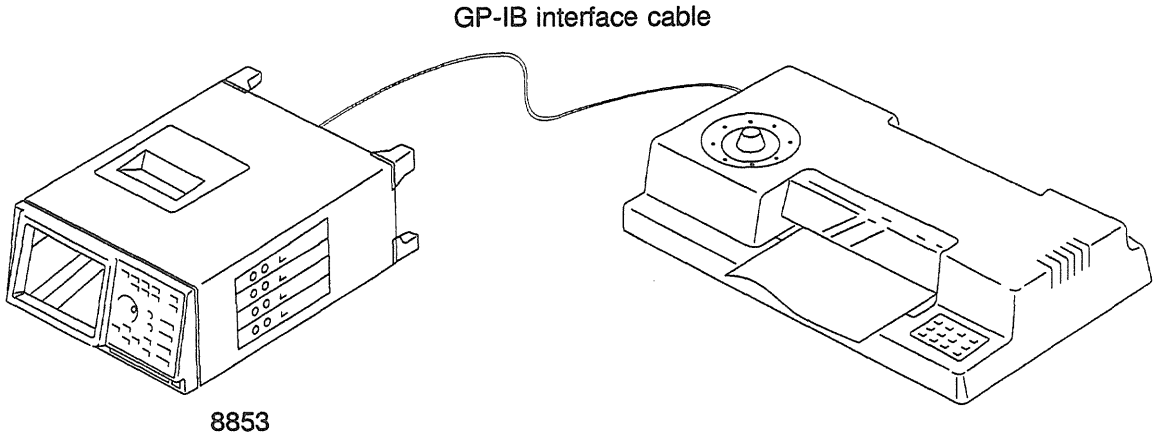
```
100 ' ------------------------------------
110 ' 8853 Sample program No.7
120 ' You must merge this code with DECL.BAS
130 ' ------------------------------------
140 BOAD% = 0
150 ADRS%(0) = 5:ADRS%(1) = NOADDR%              'GP-IB Address = 5
160 CALL SENDIFC(BOAD%)                          'Clear interface
170 CALL ENABLEREMOTE(BOAD%, ADRS%(0))           'Enable remote
180 ON PEN GOSUB 340
190 SRE$="*SRE 32":ESE$="*ESE 60"
200 SCL$="*CLS":ESR$="*ESR?"
210 CALL SEND(BOAD%, ADRS%(0), SRE$, NLEND%)     'Mask SRQ
220 CALL SEND(BOAD%, ADRS%(0), ESE$, NLEND%)     'Mask SESER
230 CALL SEND(BOAD%, ADRS%(0), SCL$, NLEND%)     'Clear statusbyte
240 PEN ON
250 FUN$=":FUNCTION MEM"
260 CALL SEND(BOAD%, ADRS%(0), FUN$, NLEND%)     'Set FUNCTION
270 I% = 5
280 AVR$=":CONFIGURE:AVERAGE "+STR$(I%)
290 CALL SEND(BOAD%, ADRS%(0), AVR$, NLEND%)     'Set AVERAGE
300 I% = I% + 50:GOSUB 480:GOTO 280
310 ' ------------------------------------
320 ' Service request operation
330 ' ------------------------------------
340 CALL IBRSP(ADRS%, S%)
350 DCL$ = CHR$(DCL%):CALL IBCMD(BOAD%, DCL$)    'Clear buffer
360 CALL SEND(BOAD%, ADRS%(0), ESR$, NLEND%)     'ERROR kind?
370 CMD$ = SPACE$(8)
380 CALL RECEIVE(BOAD%, ADRS%(0), CMD$, STOPEND%)  'receive ERROR
390 B = VAL(CMD$)
400 IF (B AND &H4) <> 0 THEN PRINT "Query ERROR"
410 IF (B AND &H8) <> 0 THEN PRINT "Machine ERROR"
420 IF (B AND &H10) <> 0 THEN PRINT "Execute ERROR"
430 IF (B AND &H20) <> 0 THEN PRINT "Command ERROR"
440 PEN OFF
450 UNT$ = CHR$(UNT%):CALL IBCMD(BOAD%, UNT$)    'UN TALK
460 CALL ENABLELOCAL(BOAD%, ADRS%(0))            'Enable operations
470 END
480 FOR J%=0 TO 1000
490 NEXT J%
500 RETURN
```

Line	*Comment*
150	Set ADRS%(0) to address of 8853.
160-170	Send interface clear, and switch to remote mode.
180	Set jump address for if a service request is received.
210	Enable bit 5 (ESB) of the status byte by the service request enable register.
220	Enable bits 2, 3, 4, and 5 of the standard event status register by the standard event status enable register.
230	Clear the status byte associated queue.
240	Enable the service request interrupt.
260	Set the function.
290	Set averaging. (Error source)
340	Serial polling to read the status byte.
380	Read the standard event status register.
400-430	From the value read, determine the error, and display it.
440	Disable service request interrupt.
440-460	Release talker and remote mode.

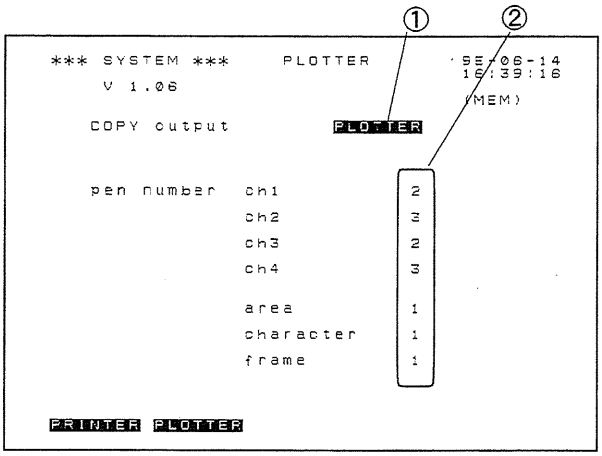
Chapter 11
Plotter Output

11.1 Settings for plotter output

- It is possible to use an HP-GL compatible plotter to output the waveform from the 8853 display screen.
- If there is a comment appended to a channel, it will also appear on the plot. (For more details of comments, see Section 12.4, "Appending Comments" in the instruction manual for the 8853.)
- Connect the plotter to the 8853 using the GP-IB interface cable.



Procedure



CRT copy output setting screen

1. Press the SYSTEM key, to display the system screen.
2. Press the CRTCOPY soft key to get the CRT copy output settings screen.
3. Set the CRT copy output device in COPY output. (① on the left figure.)

   Press the PLOTTER soft key.

Soft key indication

PLOTTER ·····Outputs to plotter

PRINTER
F D
SCSI

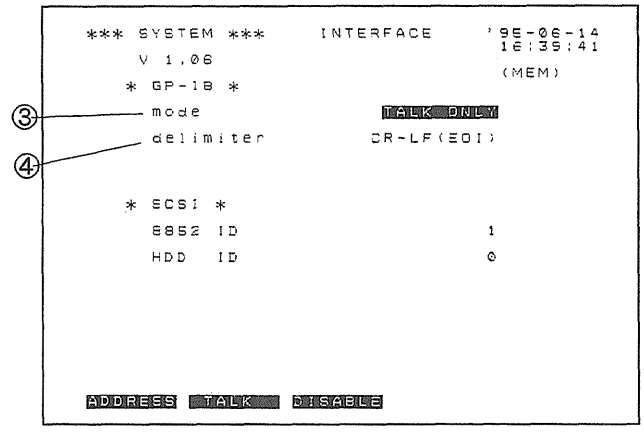
} For details see Section 12.7 in the instruction manual for the 8853.

4. Set the pen number to be used by each channel in pen number as ② on the upper figure. [—, 1 to 8] (—: No pen)

· It can assign a color for each waveform, area, character or frame.

· Use ↓ and ↑ soft keys or the rotary knob to make the settings of the pen number.

ch1	Waveform of analog CH1 and logic CHA.
ch2	Waveform of analog CH2 and logic CHB.
ch3	Waveform of analog CH3 and logic CHC.
ch4	Waveform of analog CH4 and logic CHD.
area	The waveform decision area.
character	The characters, trigger mark, cursor readout value, waveform parameter calculation result and comments.
frame	The frame, grids and A and B cursors.



INTERFACE setting screen

5. Next carry out the GP-IB settings on the interface setting screen.
   Press the SYSTEM key, then press the INTER soft key.
6. Set mode in mode item (③ on the left figure).
   Press the TALK soft key to select talk only mode for the plotter.

7. Set delimiter in the "delimiter" item (④ on the left figure).
   - The delimiter sequence required depends on the plotter being used.
   - Consult the documentation accompanying the plotter.

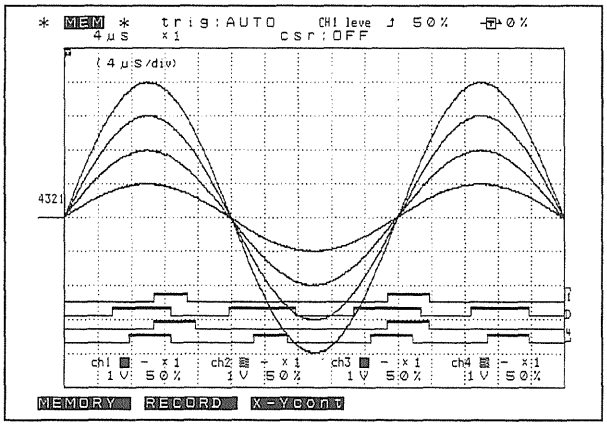
Soft key indication

CR-LF	: CR-LF(EOI)
C R	: CR (EOI)
L F	: LF (EOI)
(EOI)	: (EOI)

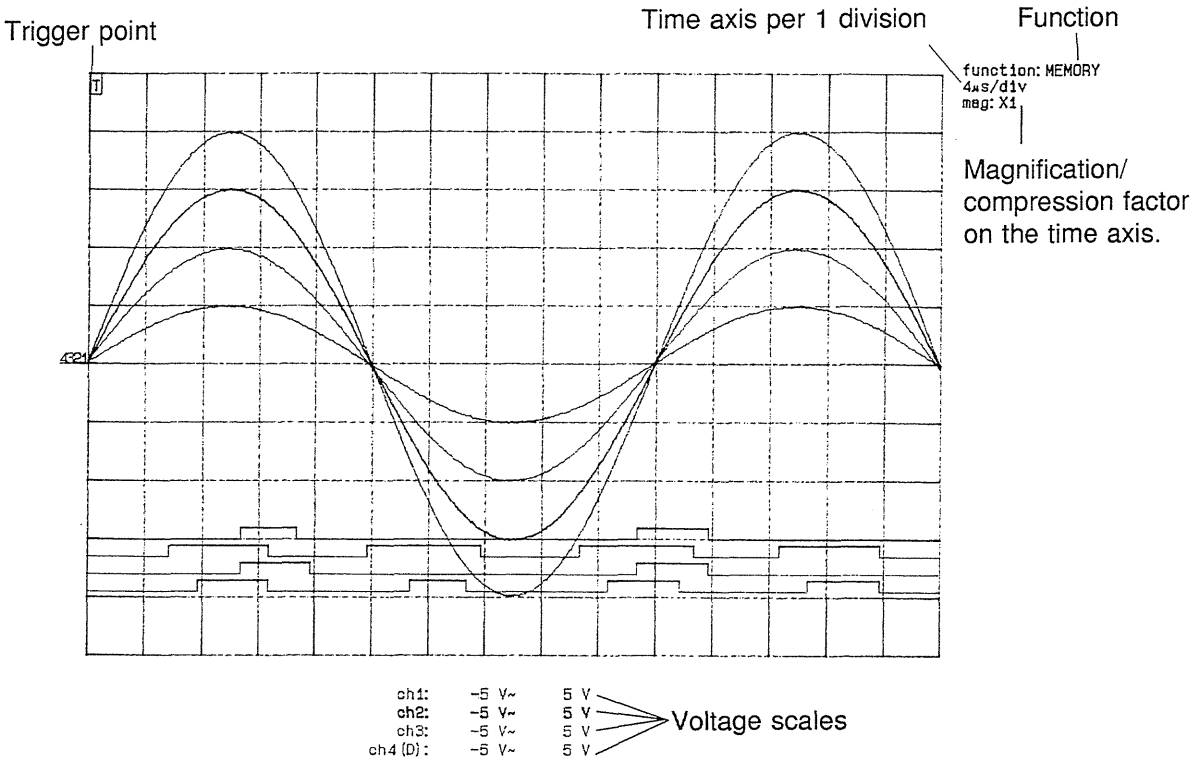
8. Press the DISP key to show the display screen.
9. Press the COPY key to begin plotter output.

The plotter output appears as shown in the following figure.

① Display screen



② Waveform on the plotter



NOTE

- In overwriting mode (see section 5.3.18 in the 8853 instruction manual), only the last waveform captured will be printed.
- The compression waveform and the waveform in the recorder function are traced twice.

Chapter 12 Device Compliance Statement

The following information relates to the compliance with the IEEE 488.2 standard.

① IEEE 488.1 interface functions

These are detailed in Section 6.2, "Interface functions".

② Operations with a device address other than 0 through 30

It is not possible to set to other than 0 through 30.

③ Timing of changed device address recognition

A change of address is recognized immediately after powering on.

④ Device settings at powering on, including all commands which further restrict the initial setting

The status information is cleared. However, the :MEMORY:POINT command and the :MEMORY:FFTPOINT command are reinitialized, and all other items are preserved.

⑤ List of message exchange options

(a) Input buffer capacity and operation

The 8853 has an input buffer of 512 bytes capacity. If the data accumulated in this buffer exceeds 512 bytes the buffer full, and until a space again becomes available in the buffer the IEEE 488.1 bus goes into the waiting state.

(b) Queries to which multiple response message units are returned

There are no query to return multiple response message.

(c) Queries producing responses are syntax checking is performed

On the 8853, all queries produce responses when syntax checking is performed.

(d) Whenever any queries produce responses when read

There are no queries which produce response messages at the instant they are read in by the controller.

(e) Whether any commands are coupled

There are no relevant commands.

⑥ Summary of functional elements for use when constructing device specific commands, and whether compound commands or program headers can be used

Program message, program message terminator, program message unit, program message unit separator, command message unit, query message unit, command program header, query program header, program data, character program data, decimal program data, character string program data, and compound commands program headers.

⑦ Buffer capacity limitations for block data

Block data is not used.

⑧ Summary of program data elements used in expressions, and deepest nesting level allowable in sub-expressions, including syntax restrictions imposed by the device

Sub-expressions are not used. Character data and decimal data are the only program data elements used.

⑨ Response syntax for queries

Response syntax is detailed in Chapter 9, "Commands Reference."

⑩ Transmission congestion relating to device to device messages which do not conform to the general principles for basic response messages

There are no device to device messages.

⑪ Response capacity for block data

Block data does not appear in responses.

⑫ Summary of standard commands and queries used

This appears in Chapter 8, "Command Summary."

⑬ Device state after a calibration query has been completed without any problem

The "*CAL?" query is not used.

⑭ When using the "*DDT" command, the maximum length of block used in a trigger macro definition

The "*DDT" command is not used.

⑮ When a macro command is being executed, the maximum length of macro label, the maximum length of block for defining a macro, and how echoing is managed when expanding a macro

Macros are not used.

⑯ For queries related to identification, explanation of the response to the "*IDN?" query

This is detailed in Section 9.2, "Standard Commands Stipulated by IEEE 488.2".

⑰ Capacity of the user data storage area reserved for when the "*PUD" command and the "*PUD?" query are being executed

The "*PUD" command and the "*PUD?" query are not used. Further, there is no user data storage area.

⑱ Resources when the "*RDT" command and the "*RDT?" query are being used

The "*RDT" command and the "*RDT?" query are not used.

⑲ Conditions which are influenced when "*RST", "*LRN?", "*RCL", and "*SAV" are used

"*LRN?", "*RCL", and "*SAV" are not used. The "*RST" command returns the 8853 to its initial state.

⑳ Scope of the self-testing executed as a result of the "*TST?" query

Checks the internal ROM and RAMs.

㉑ Additional organization of the status data used in a device status report

This is detailed in Section 7.6, "The Status Byte and the Event Registers".

㉒ Whether commands are overlap or sequential type

All the commands are sequential commands except :ABORT command. An :ABORT command is executed instantly as soon as it is transmitted.

㉓ Criterion relating to the functions required at the instant that the termination message is produced, as a response to each command

Termination occurs when the command has been parsed.

Appendix

Troubleshooting the GP-IB faults

Check the items in the following table in the event of operating problems with the GP-IB interface.

Symptom	Likely causes and remedies
The GP-IB does not operate at all.	Is the cable properly connected?
	Is the GP-IB address of the 8853 unit correctly set? Does it clash the address of other equipment on the same bus?
	Are all the devices that are connected powered on?
Although a command was transmitted, the unit did not operate.	Use the "*ESR?" query to check the standard event status register for anomalies.
An attempt to read data using the CALL RECEIVE statement causes the GP-IB bus to hang.	Each and every CALL RECEIVE statement must be preceded by a query.
	Is the query transmitted incorrect?
The 8853 keys stop working after using GP-IB communications.	Press the soft key to end the remote operating state.
	Has an LLO (local lock-out) command been sent to the 8853? Send a GTL command to return to the local state.
Even though a number of queries were sent, only one response was received.	Has an error occurred?
	The response should be read immediately after each query. To read several responses in one operation, the corresponding queries must be combined into a single line using the message separator.
A service request is sometimes not issued.	Are service request enable register and the event status enable registers set correctly?
	At the end of the SRQ handling routine, use a "*CLS" command to clear all of the event registers. (If a bit in the event registers is not cleared, the same event occurring again will not generate a service request.)

Index

- A -

- C -

- D -

- E -

- F -

- G -

- H -

- I -

- L -

- M -

- N -

- O -

- P -

- Q -

- R -

- S -

- T -

- U -

- V -

- W -

- X -

- Y -

- Z -

HIOKI

HIOKI E.E. CORPORATION

81 Koizumi, Ueda, Nagano 386-11, Japan
TEL: 0268-28-0562 FAX: 0268-28-0568

8853A983-01 96-03-0001H