HIOKI

8832 · 8833

MEMORY Hi CORDER

GP-IB INTERFACE
RS-232C INTERFACE

INSTRUCTION MANUAL

HIOKI E.E. CORPORATION

Contents

Chapter 1

Outline, Specifications, and Operation of the GP-IB Interface

1-1 Outline

The GP-IB interface was developed as an interface for programmable instruments. It provides great potential for expansion and a wide variety of powerful features.

Interfaces for programmable instruments include not only the GP-IB interface, but also the IEEE Bus, the IEC Bus, and the HP-IB interface (a proprietary standard of Hewlett Packard). All of these are basically the same, except for slight differences in connector pin numbers and signal arrangements. Please pay careful attention to these differences when using any of these interfaces.

For more details on the GP-IB interface, please refer to the following literature.

ANSI/IEEE Standard 488-1978
IEEE Standard Digital Interface for Programmable Instructions

1-2 GP-IB Specifications

1-2-1 Standards conformance

The GP-IB interface described in this manual conforms to IEEE Standard 488-1978.

1-2-2 Interface functions

Function	Contents
SH1	All SH (source handshake) functions
AH1	All AH (acceptor handshake) functions
T6	Basic talker function, serial polling function MLA (My Listen Address) talker release function
L4	Basic listener function MTA (My Talk Address) listener release function
SR1	All SR (Service Request) functions
RL1	All RL (Remote Local) functions
PP0	No PP (Parallel polling) function
DC1	All DC (Device Clear) functions
DT0	No DT (Device Trigger) function
C0	No C (Control) function

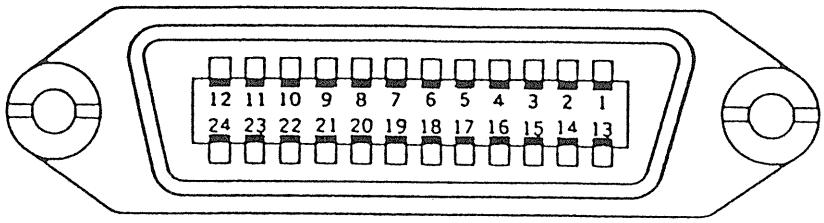
1-2-3 GP-IB Bus signal lines

Bus Structure		Signal Line Note	
Data Bus	DIO1 (Data INput Output 1) 2 (" 2) 3 (" 3) 4 (" 4) 5 (" 5) 6 (" 6) 7 (" 7) 8 (" 8)	Used for data I/O, and for interface and device message I/O.	
Transfer bus	DAV (Data Valid)	Signal indicating information data validity on data bus	Performs acceptor and source handshake.
	NRFD (Not Ready For Data)	Reception preparation ready signal	
	NDAC (Not Data Accepted)	Reception preparation ready signal	
Control bus	ATN (Attention)	Signal indicating that information data on data bus is interface message or device message.	
	IFC (Interface Clear)	Signal to initialize interface bus system.	
	SRQ (Service Request)	Signal to request asynchronous service.	
	REN (Remote Enable)	Signal to switch control remote/local.	
	EOI (End or Identify)	Signal to indicate last data byte.	

1-2-4 Connectors used

On interface: 57LE-20240 (made by DDK) or its equivalent

On cable: 57-10240 (made by DDK) or its equivalent

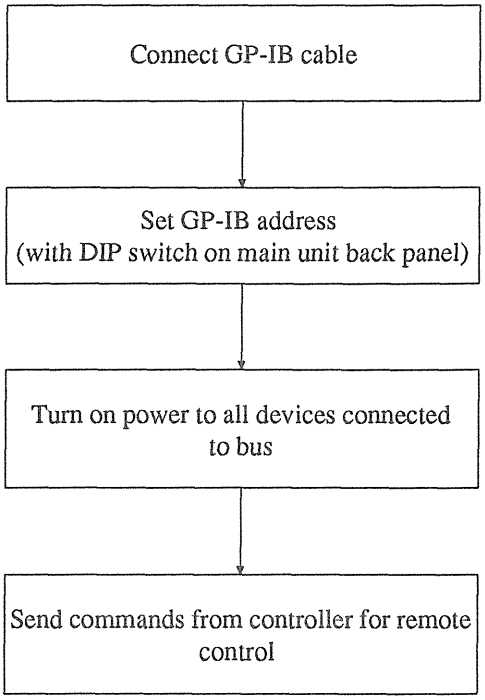


GP-IB Interface Connector Pin Assignments (Interface Connector)

Pin No.	Signal Name	Pin No.	Signal Name
1	DIO1	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

1-3 GP-IB Operation Instructions

1-3-1 Basic operation

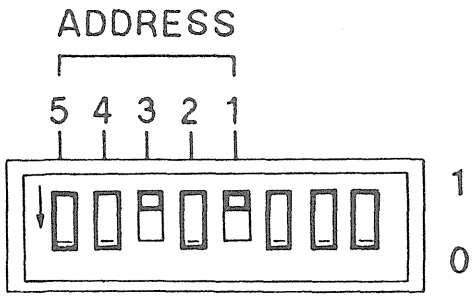


⚠ CAUTION

The GP-IB unit is not isolated from main unit of 8832. Please note that GND of input unit and input terminal and GND of GP-IB are common.

[1] Setting the GP-IB address

Set the GP-IP address with the DIP switch on the back panel.



The address can be set between 0 and 30.

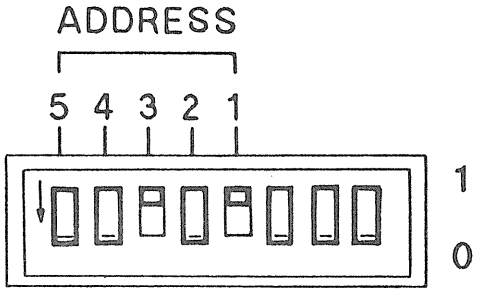
Please ensure that addresses do not overlap with those of the controller or other devices.

Setting example

Example: Setting the address to 5.

$5 = (0\,0\,1\,0\,1)_2$

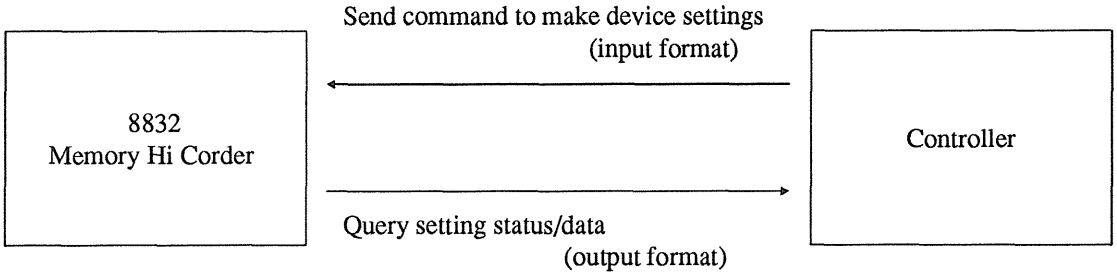
SW	5	4	3	2	1
	0	0	1	0	1



⚠ CAUTION

Address 31 (SW 1 - 5, all 1) is reserved and cannot be set.

1-3-2 I/O Format



[1] Input format

The command input format is as follows:

Command	Parameter	Terminator

[Command]..... Letters indicating the type of command as follows.

For set commands (command to change settings of the 8832), two letters indicating the command.

(Example) FN

For read commands (commands causing the 8832 to output the setup status or data during the next talker action), three letters, the first of which is always Q.

(Example) QFN

[Parameter]..... Numerics specifying the settings

- These are ASCII format numerics. Codes to be applied are limited to 0 - 9 (30H - 39H), + (2BH), - (2DH).
- When using multiple parameters, separate them with a space (Δ) or comma(,).

(Example) AA1, 2, 50, 2, 0

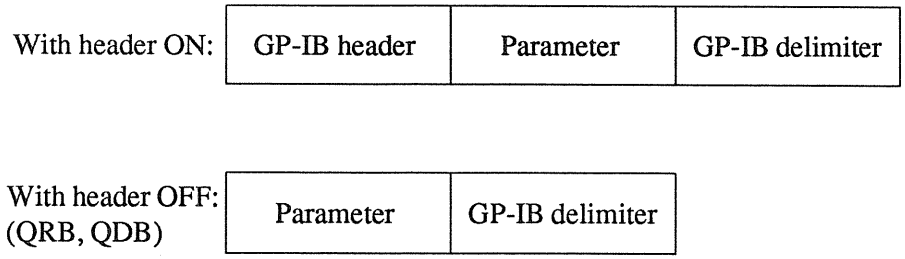
AA1Δ2Δ50Δ2Δ5D0

[Terminator]..... Code indicating the end of instruction

- Colon (:, ASCII Code 3AH) is used as the terminator.
- If another instruction follows, the terminator can be omitted. However, for the CC, CM, SE and NA commands the terminator cannot be omitted.

[2] Output format

The output data format is either of the following two formats. For the QRB and QDB commands, however, it should always be in the format without header.



[GP-IB Header]... Indicating the meaning of output data.

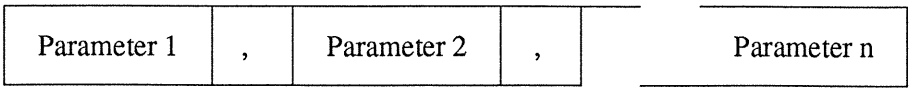
- Made up of the last two letters of a read command and the read command parameters.

  (Example) The GP-IB header of the "QDR1" command is "DR1".

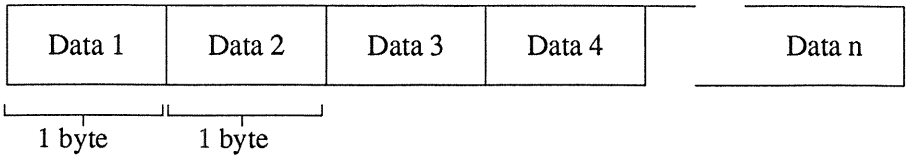
  (Note) The GP-IB header of the "QDAn" command is only "DA".

[Parameter]..... Number indicating settings, character string or data

- Always ASCII format numbers or character strings unless output data is the response to the QRB or QDB command. Multiple parameters are delimited with commas (,).



- The response to the QRB/QDB command is in binary format, in which one byte represents one data item.



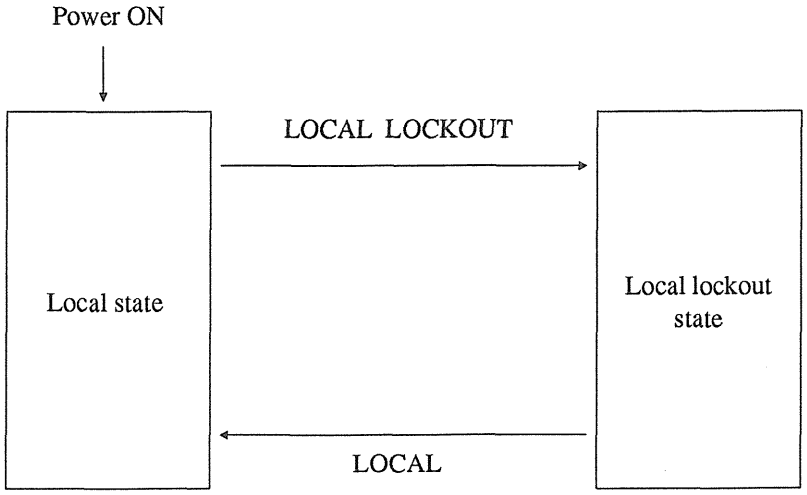
[GP-IB delimiter]... Delimiting code in data string. There are four types of delimiters as follows.

One of these can be selected with the GD command.

CRLF (E01)	0DH	0AH
CR (E01)	0DH	
LF (E01)	0AH	
(EO1).....	None	

1-3-3 Local state and local lockout state

- The GP-IB interface unit is in the local state immediately after power on.
- The interface unit enters the local lockout state when a LOCAL LOCKOUT command (LLO) is received. It returns to the local state when a LOCAL command is received.



[Local state]: State in which external control is possible both through the GP- IB interface and from the control panel.

[Local lock out state]: State in which external control is possible only through the GP-IB interface.

(Program sample 1)... HP9816 (Made by HP)

Local lockout	LOCAL LOCKOUT 7
Local	LOCAL 7

(Program sample 2)... PC9801 (Made by NEC)

Local lockout	ISET REN WBYTE &H11;
Local	IRESET REN

1-3-4 Device clear and buffer memory

- General instructions are first stored in the 256-byte buffer, then executed in FIFO (first-in-first-out) order. However, the device clear instruction is executed immediately regardless of the buffer memory state.
- If a device clear (DCL) or select device clear (SDC) command is received, the following operation takes place:

① START/PRINT/COPY processing is aborted

② Buffer memory is cleared

③ Errors are cancelled and status byte is cleared

④ Serial polling is cancelled

⑤ The following settings are initialized

- SRQ response mask value (MS0) of the MS command
- I/O point (OD1,0) of the OD command

(Program sample 1)...HP9816 (Hewlett-Packard)

```
CLEAR 7
```

(Program sample 2)...PC9801 (NEC)

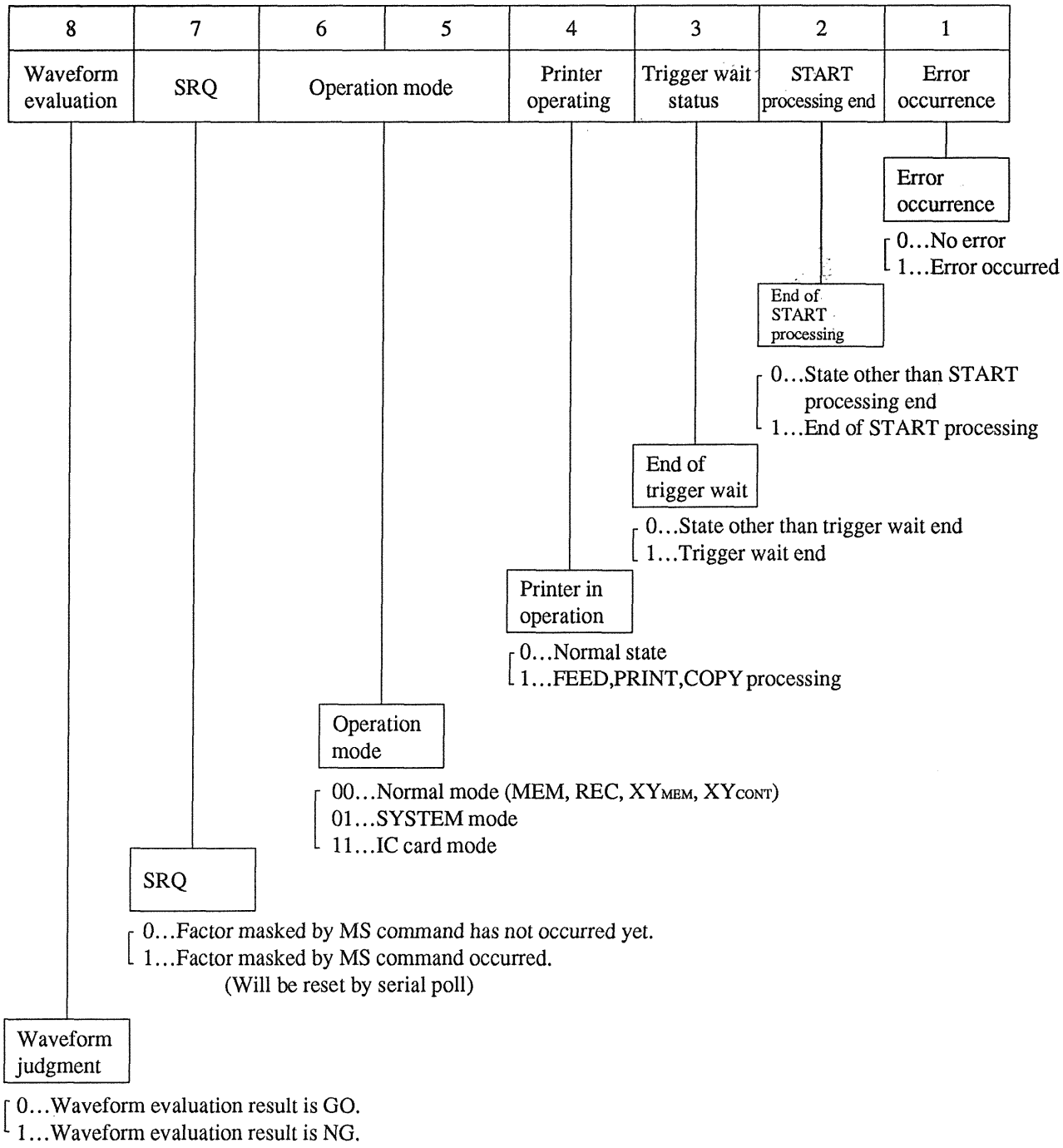
```
WBYTE &H14;
```

(Note) Wait is needed after processing.

1-3-5 Status byte and service request

[1] Status byte

- Bits of the status byte output in response to serial polling (SPE command)* by the controller is as follows:



- The above also applies to the status byte output by the QUS command.
- All bits are "0" in the initial state.

* Function that reads status byte from the unit connected to the controller.

[2] Service request

- Conditions causing an emergency signal (service request) to be sent to the controller can be set with the MS command.

8~3	2	Bit 1
	START processing end	Error occurrence

"MS0" ... No service request

"MS1" ... Makes service request in case of error occurrence.

"MS2" ... Makes service request at end of START processing.

"MS3" ... Makes service request upon error occurrence or at end of START processing.

[3] Serial polling program example

1. For the HP9816 (Hewlett-Packard)

```
100 ON INTR 7 GOSUB 500
110 ENABLE INTR 7 ; 2

500 S=SPOLL (705)

550 ENABLE INTR 7;2
560 RETURN
```

Set SRQ interrupt processing routine and enable interrupt.

Serial poll from address five.
Enable interrupt.

2. For the PC9801 (NEC)

```
100 ON SRQ GOSUB 500
110 SRQ ON

500 POLL 5,S

550 SRQ ON
560 RETURN
```

Set SRQ interrupt processing routine and enable interrupt.

Serial poll from address five.
Enable interrupt.

1-3-6 GP-IB Errors

- Errors are categorized as general errors, warnings, and GP-IB special errors.

Group	Error number	Name	Description
General errors	1~20		
Warnings	31~43		
GP-IB errors	51	Command error	• Undefined command has been entered.
	52	Parameter error	• Input instruction parameter (number) format is erroneous. • Parameter value is inappropriate.
	53	Inappropriate command	• System is in a state in which command cannot be input.
	54	Output request error	• Output request (talker designation) is inappropriate.

- Error numbers output by the QER command are as above.
- If error #54, output request error, occurs, the following data will be output:

NG999, 999

Chapter 2

Outline, Specifications, and Operation of the RS-232C Interface

2-1 Outline

An RS-232C interface is one that conforms to a particular EIA (Electronic Industries Association) standard for serial interface. Such interfaces are generally used for communication over a telephone line.

The RS-232C interface provided by Hioki for the 8832 Memory Hi Corder makes it possible to control operation of the Hi Corder remotely from a personal computer. It also allows the personal computer in input data from the Hi Corder for processing, or to output data to the Hi Corder for plotting.

This booklet explains only how to install and use the RS-232C interface in your 8832 Memory Hi Corder. See the manuals provided with your Memory Hi Corder or personal computer for information on operation of those devices.

2-2 RS-232C Specifications

2-2-1 Standards conformance

The RS-232C interface described in this manual conforms to the EIA RS-232C standard for serial interfaces.

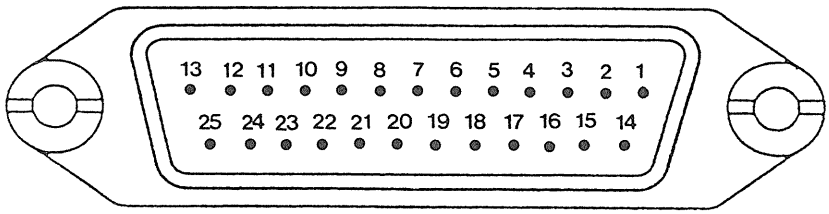
2-2-2 General specifications

Communication mode:	Half duplex
Synchronization:	Start-stop synchronization
Transfer speed:	1200, 2400, 4800, 9600, or 19200 baud (selected by DIP switch)
Stop bits:	1 or 2 (selected by DIP switch)
Data length/parity:	8 bits/no parity 7 bits/no parity 7 bits/even parity 7 bits/odd parity (selected by DIP switch)

2-2-3 Connectors used

On interface: DBLC-J25SAF (made by JAE) or equivalent

On cable: DB-25P (made by JAE) or equivalent

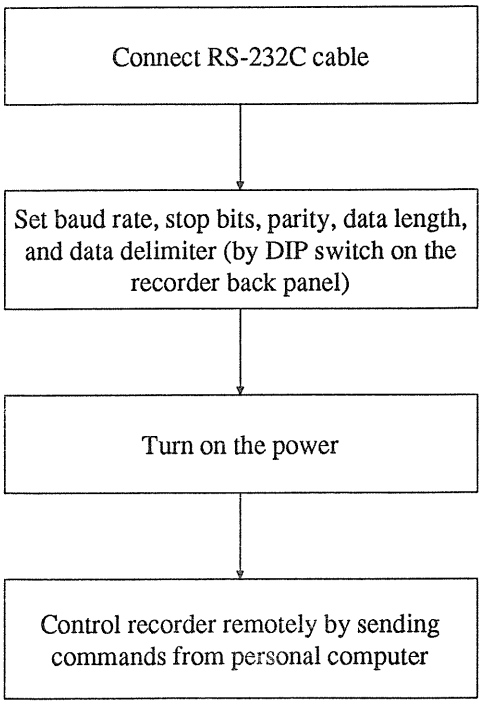


RS-232C Interface Connector Pin Assignments (On Interface)

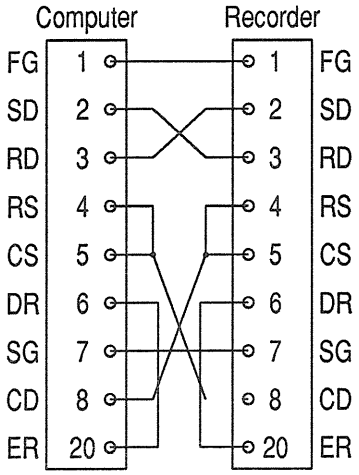
Pin No.	Symbol		Signal Name
	RS-232C	JIS	
1	AA	FG	Frame Ground
2	BA	SD	Send Data
3	BB	RD	Receive Data
4	CA	RS	Request to Send
5	CB	CS	Clear to Send
6	CC	DR	Data Set Ready
7	AB	SG	Signal Ground
8	CF	CD	Data Channel Receive Carrier Detect
20	CD	ER	Data Terminal Ready

2-3 RS-232C Operation Instructions

2-3-1 Basic operation



- Cable connection
  When connecting the Hi Corder RS-232C interface directly to the serial interface on a personal computer, use a cross-connected cable with wiring as shown in the figure below.



[1] Setting the baud rate, stop bits, parity, data length, and data delimiter

Use the DIP switch on the recorder's back panel to set the baud rate, number of stop bits, type of parity, data length, and type of data delimiter. Make sure that the baud rate, stop bits, parity, and data length settings are the same as those of the serial interface on the connected personal computer.

DIP switch

1	2	3	4	5	6	7	8
Baud rate			Data length/parity		Stop bits	Delimiter	

- Baud rate

DIP switch			Baud rate
1	2	3	
0	0	1	19200
0	1	0	9600
0	1	1	4800
1	0	0	2400
1	0	1	1200

- Data length/parity

DIP switch		Data length	Parity
4	5		
0	0	8 bits	None
0	1	7 bits	None
1	0	7 bits	Even
1	1	7 bits	Odd

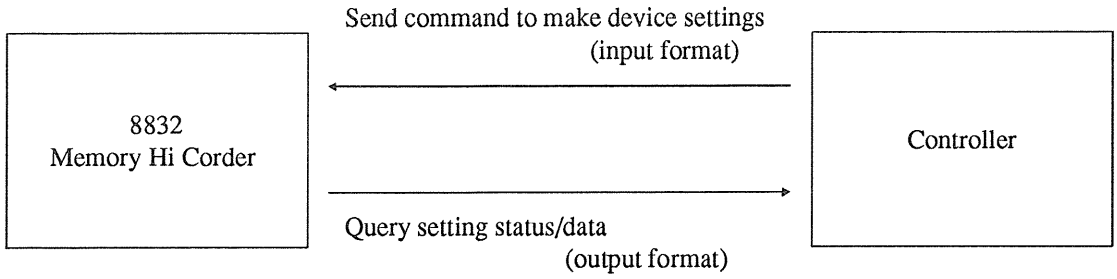
- Stop bits (DIP switch 6)

  0: 1 bit
  1: 2 bits

- Delimiter (DIP switch 7)

  0: CR+LF
  1: CR

2-3-2 I/O Format



[1] Input format

The command input format is as follows:

Command	Parameter	Terminator

[Command]..... Letters indicating the type of command as follows.

For set commands (command to change settings of the 8832), two letters indicating the command.

(Example) FN

For read commands (commands causing the 8832 to output the setup status or data during the next talker action), three letters, the first of which is always Q.

(Example) QFN

[Parameter]..... Numerics specifying the settings

- These are ASCII format numerics. Codes to be applied are limited to 0 - 9 (30H - 39H), + (2BH), - (2DH).
- When using multiple parameters, separate them with a space (Δ) or comma(,).

(Example) AA1, 2, 50, 2, 0

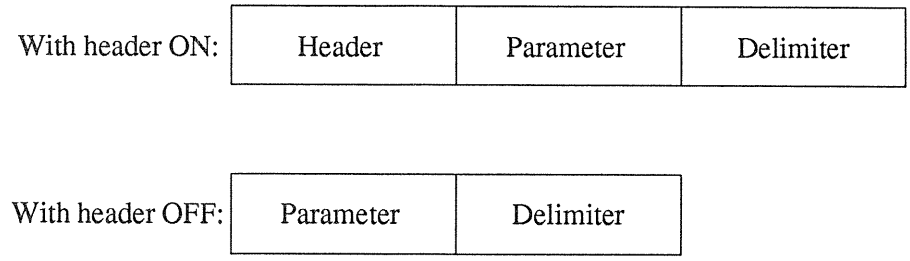
AA1Δ2Δ50Δ2Δ5D0

[Terminator]..... Code indicating the end of instruction

- Colon (:, ASCII Code 3AH) is used as the terminator.
- If another instruction follows, the terminator can be omitted. However, for the CC, CM, SE, and NA commands the terminator cannot be omitted.

[2] Output format

Depending on the header setting, either of two formats can be used for data output as follows.



[Header]... Indicating the meaning of output data.

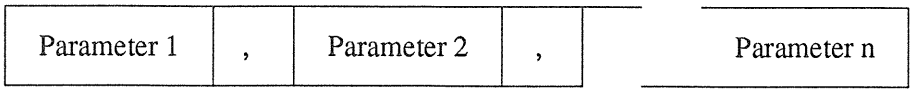
- Made up of the last two letters of a read command and the read command parameters.

  (Example) The header of the "QDR1" command is "DR1".

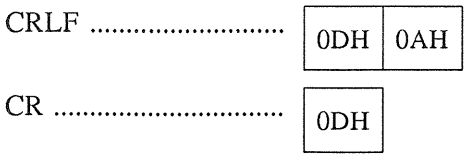
  (Note) The header of the "QDAn" command is only "DA".

[Parameter]..... Numeric values, character strings, and data that indicate settings.

- The parameters always consist of strings of ASCII numerals or characters. Multiple parameters are delimited with commas (,).

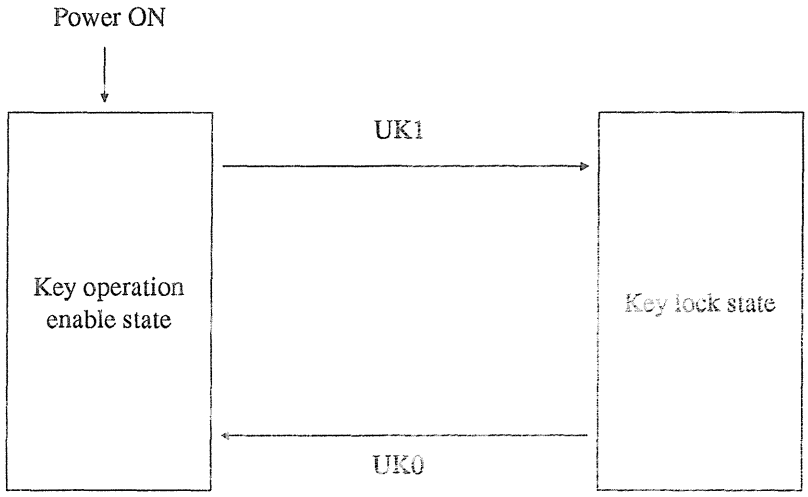


[delimiter]... Delimiting code in data string. There are two types of delimiters as follows.



2-3-3 Key lock status

The key lock state can be set or canceled by sending the UK command to the recorder.



[Key operation enable state]:
The recorder can be controlled either from the keys on the operation panel or by external commands sent through the RS-232C interface.

[Key lock state]: The recorder can be controlled only by external commands sent through the RS-232C interface.

2-3-4 Device clear and buffer memory

- General instructions are first stored in the 256-byte buffer, then executed in FIFO (first-in-first-out) order. However, the device clear instruction is executed immediately regardless of the buffer memory state.
- If a device clear ([ESC]C) command is received, the following operation takes place:

① START/PRINT/COPY processing is aborted

② Buffer memory is cleared

③ Errors are cancelled and status byte is cleared

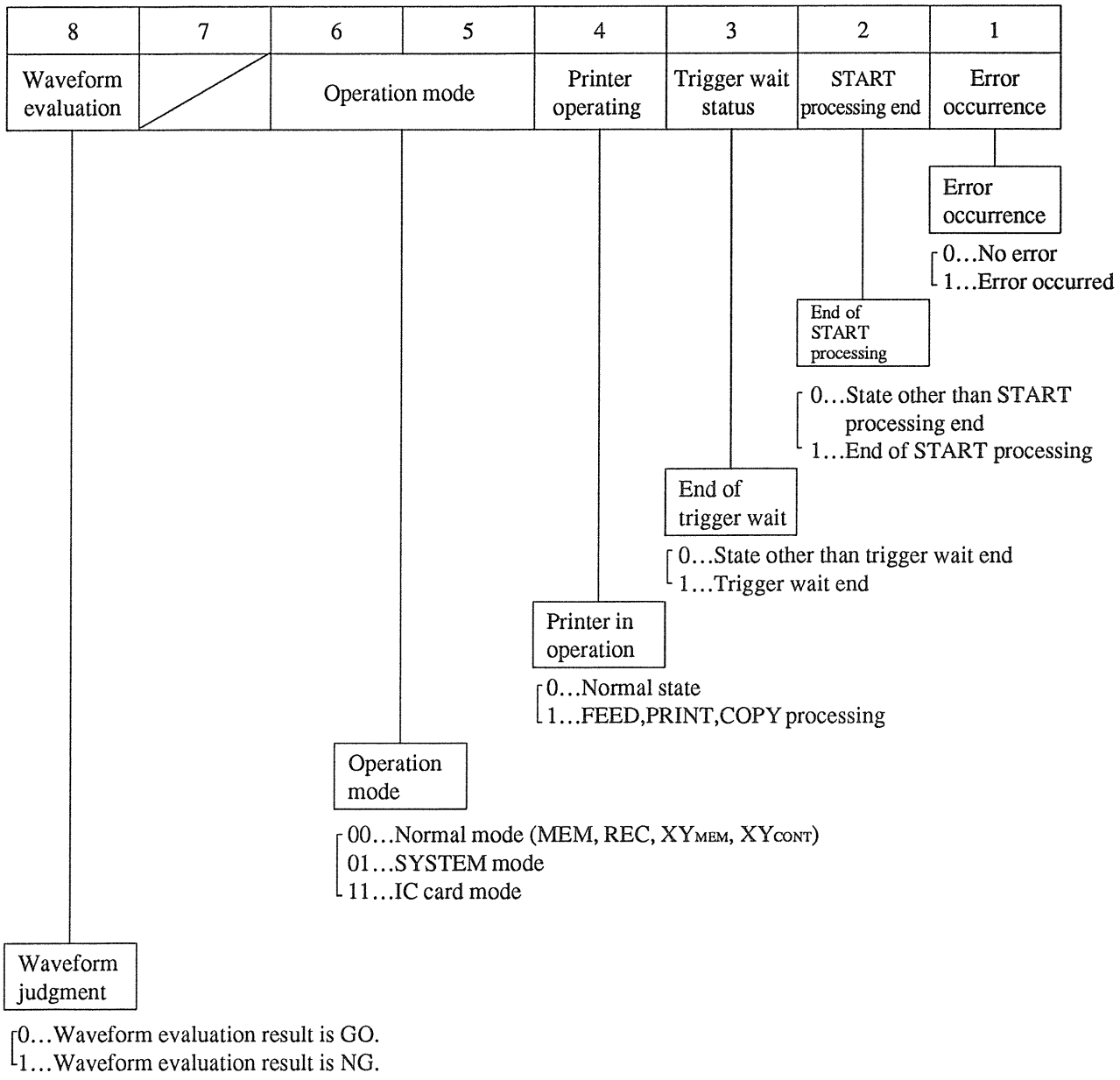
④ The following settings are initialized

- The XO command (XO 0)
- I/O point (OD1,0) of the OD command

(Note) Wait is needed after processing.

2-3-5 Status byte

- Bits of the status byte output in response to the QUS command from the personal computer are as follows.



- The above also applies to the status byte output by the QUS command.
- All bits are "0" in the initial state.

2-3-6 RS-232C Errors

- Errors are categorized as general errors, warnings, and RS-232-C special errors.

Group	Error number	Name	Description
General errors	1~20		
Warnings	31~43		
RS-232C errors	51	Command error	• Undefined command has been entered.
	52	Parameter error	• Input instruction parameter (number) format is erroneous. • Parameter value is inappropriate.
	53	Inappropriate command	• System is in a state in which command cannot be input.
	54	Output request error	• Output request (talker designation) is inappropriate.

- Error numbers output by the QER command are as above.
- If an RS-232C error occurs, the following data will be output:

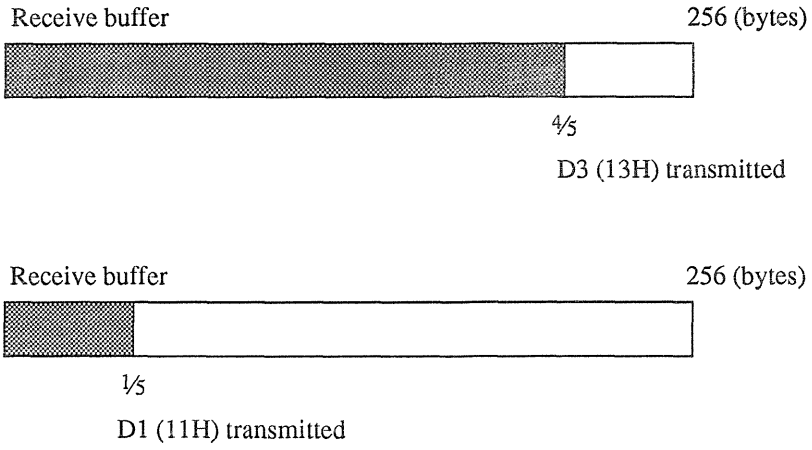
NG999, 999

(Note) **A receive overrun error occurs if a command is sent within about 100 ms after starting the main unit.**
Do not send any commands for at least 100 ms after startup.

2-3-7 Buffer flow control

- When the amount of data in the interface receive buffer exceeds 4/5 of the buffer's capacity, the recorder outputs code D3 (13H) to the personal computer to inform it that little space is available in the buffer.

  When processing reduces the amount of data in the buffer to less than 1/5 of capacity, the recorder outputs code D1 (11H) to the personal computer.



- When sending, the recorder stops transmission upon receiving code D3 (13H), then restarts transmission upon receiving code D1 (11H).

Chapter 3

GP-IB and RS-232C Commands

(Note) **Programs marked [GP-IB] are ones which use GP-IB functions, and programs marked [RS-232C] are ones which use RS-232C functions.**

3-1 Using Set Commands

- When the conditions specified in the command list are met, send set commands in accordance with the <SET COMMAND> format.

[GP-IB]

Program example for HP9816

```
100 ! ---------------------------------- !
110 !   8832      Set Command  HP9816     !
120 ! ---------------------------------- !
130 !
140 Adr=705
150   OUTPUT Adr;"FN1"              !Function MEM
160   OUTPUT Adr;"TD2"              !Time/Div 500us
170   OUTPUT Adr;"SH0"              !Shot     20DIV
180   OUTPUT Adr;"ST"               !< START >
190 END
```

[GP-IB]

Program example for PC9801

```
100 ' ---------------------------------- '
110 '   8832      Set Command  PC9801     '
120 ' ---------------------------------- '
130 '
140 ADR=5
150 ISET IFC
160 ISET REN
170  PRINT@ ADR;"FN1"              'Function MEM
180  PRINT@ ADR;"TD2"              'Time/Div 500us
190  PRINT@ ADR;"SH0"              'Shot     20DIV
200  PRINT@ ADR;"ST"               '< START >
210 IRESET REN
220 END
```

[RS-232C]

Program example for IBM PS/2

```
100  '-----------------------------------------'
110 '      8832  RS-232C SET COMMAND             '
120 '-----------------------------------------'
130 '
140 OPEN "COM1:9600,N,7,1" AS #1     'RS-232C FILE OPEN
150 '
160 PRINT #1,"FN1"                   'FUNCTION : MEM
170 PRINT #1,"TD2"                   'TIME/DIV : 500us
180 PRINT #1,"SH0"                   'SHOT : 20DIV
190 PRINT #1,"ST"                    '<START>
200 '
210 CLOSE #1                         'FILE CLOSE
220 END
```

3-2 Using Read Commands

[GP-IB]

- When the conditions specified in the command list are satisfied, send read commands in accordance with the <READ COMMAND> format to make the 8832 stand by for output. Then, designate the 8832 as the talker and receive data output from the talker.
- Data prepared for output by the <READ COMMAND> will be in the format described in the command list. <READ COMMAND> without SET COMMAND will be in the format shown in parentheses in the command list.

Program sample for HP9816

```
100 ! ---------------------------------- !
110 !   8832    Read Command  HP9816     !
120 ! ---------------------------------- !
130 !
140 DIM Ti$[20]
150 Adr=705
160   OUTPUT Adr;"GH1"            !Header ON
170   OUTPUT Adr;"QFN"            !Read Function
180    ENTER Adr;Ans$
190   OUTPUT Adr;"QRT"            !Read Date & Time
200    ENTER Adr;Da$,Ti$
210   PRINT Ans$,Da$,Ti$
220 END
```

Program sample for PC9801

```
100 ' ---------------------------------- '
110 '   8832     Read Command  PC9801    '
120 ' ---------------------------------- '
130 '
140 ADR=5
150 ISET IFC
160 ISET REN
170  PRINT@ ADR;"GH1"            'Header ON
180  PRINT@ ADR;"QFN"            'Read Function
190   INPUT@ ADR;ANS$
200  PRINT@ ADR;"QRT"            'Read Date & Time
210   INPUT@ ADR;DA$,TI$
220  PRINT ANS$,DA$,TI$
230 WBYTE &H5F;
240 IRESET REN
250 END
```

[RS-232C]

- When the conditions specified in the command list are satisfied, send read commands to the recorder in accordance with the <READ COMMAND> format to make it send data.
- Data prepared for output by the <READ COMMAND> will be in the format described in the command list. <READ COMMAND> without SET COMMAND will be in the format shown in parentheses in the commans list.

Program sample for IBM PS/2

```
100 '-----------------------------------------------'
110 '       8832  RS-232C READ COMMAND               '
120 '-----------------------------------------------'
130 '
140 OPEN  "COM1:19200,N,7,1,LF" AS #1
150 '
160 PRINT #1,"GH1,QFN"                  'HEADER ON, FUNCTION ?
170 INPUT #1,A$
180 PRINT "FUNCTION  = ";A$
190 PRINT #1,"QRT"                      'DATE & TIME ?
200 LINE INPUT #1,A$
210 PRINT "DATE & TIME : ";A$
220 '
230 CLOSE #1
240 END
```

3-3 Using Service Requests

(1) Set the 8832 SRQ response mask value with the MS command, set the SRQ interrupt jump destination on the controller side, then enable SRQ interrupts.

(2) In the SRQ interrupt routine, do serial polling, read the 8832 status byte, and perform processing according to values read, before re-enabling SRQ interrupts and returning.

(Note) **This function is not supported by the RS-232C interface.**

[GP-IB]

Program sample for HP9816.

```
100 ! ---------------------------------- !
110 !   8832   Service Request HP9816    !
120 ! ---------------------------------- !
130 !
140 Adr=705
150 ON INTR 7 GOTO Srq_sub
160 OUTPUT Adr;"MS1"                !SRQ mask
170 ENABLE INTR 7;2                 !Intr enable
180 !
190   Fn=0
200   OUTPUT Adr;"FN";Fn            !Read Date & Time
210    Fn=Fn+1
220    GOTO 200
230 !
240 Srq_sub:                        !SRQ Intr
250   S=SPOLL(Adr)
260   CLEAR 7                       !Buff Clear
270    PRINT " ",SRQ=";S
280   OUTPUT Adr;"MS1"
290   ENABLE INTR 7,2
300 END
```

[GP-IB]

Program sample for PC 9801

```
100 ' ---------------------------------- '
110 '    8832    Service Request PC9801   '
120 ' ---------------------------------- '
130 '
140 ADR=5
150 ISET IFC
160 ISET REN
170 ON SRQ GOSUB *SUB
180 PRINT@ ADR;"MS1"                        'SRQ mask
190 SRQ ON
200 '
210 I=0
220  PRINT@ ADR;"FN"+STR$(I)                'Function Set
230 I=I+1 : GOTO 220
240 '
250 *SUB                                    'SRQ Intr
260    POLL ADR,S
270    IF (S AND &H40)<>0 THEN 320          'SRQ Check
280      DEF SEG=&H60 : A%=PEEK(&H9F3)
290      A%=A% AND &HBF : POKE &H9F3,A%     'SRQ Bit Clear
300      GOTO 340
310 '
320    WBYTE &H14;                          'Buffer Clear
330      PRINT "SRQ=";S
340    PRINT@ ADR;"MS1"                     'SRQ mask
350    SRQ ON
360 IRESET REN
370 END
```

Note for using PC9801

When a PC9801 is used, a SRQ interrupt may occur even if a service request is not made. In this case, it is necessary to forcibly clear the PC9801's SRQ bit as shown in lines 280 to 290 above.

(Excerpt from "PC Note" published by Personal Computers Sales Promotion Headquarters of NEC Corp.)

3-4 Data Input and Output

3-4-1 Output of character string data

- Data output by the cursor read-out output (QVO) or voltage conversion (QVL) command is a character string, including a unit of time or voltage.
- Data output by the trigger detection time output (QTJ) or time output (QRT) is a character string including "-" and ":".

(Output sample) (With header ON)

"QVO" ——→ VO10.1mV, 236us

"QVL" ——→ VL1, 100, 25.2mV

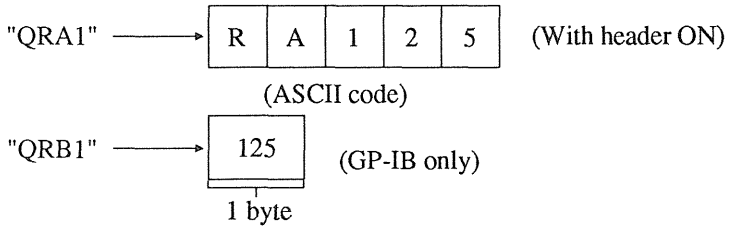
"QTJ" ——→ TJ90-03-15, 15:14:27

"QRT" ——→ RT89-11-30, 09:48:35

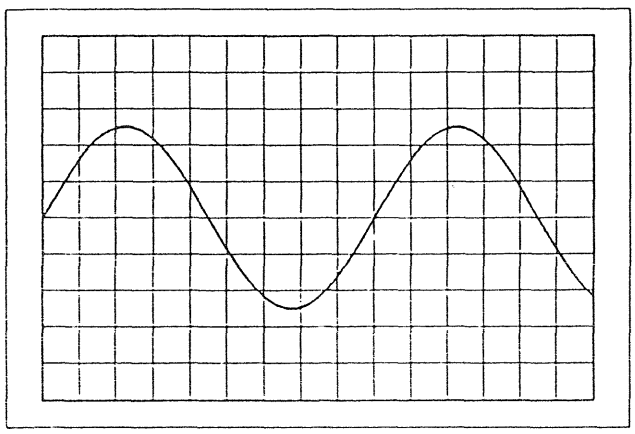
3-4-2 Output of realtime data

- Data measured is output when a QRA or QRB command is received.
- For the QRA command, output data is in ASCII format.
- For the QRB command, output data is in binary format. (GP-IB only)

(Output sample)



- Analog unit data is in the range from -3 to 252. The relationship between numeric values and chart output is as follows:



(Note) Binary data output format of 253, 254 or 255 indicates -3, -2 or -1, respectively. (GP-IB only)

253= $(FD)_{16}$ = –3
254= $(FE)_{16}$ = –2
255= $(FF)_{16}$ = –1

- In the logic unit data, each channel corresponds to a bit.

8	7	6	5	4	3	2	1
A1	A2	A3	A4	B1	B2	B3	B4

0.....Low
1.....Hi

(Example) For data 129, 129 = $(10000001)_2$

A1	A2	A3	A4	B1	B2	B3	B4
Hi	Low	Low	Low	Low	Low	Low	Hi

3-4-3 Storage data output method

(1) Use the QMX command to check whether or not storage data I/O is possible. If data is stored, output is not possible because the value output by QMX becomes "0".

(2) Specify the data channel and point to be output using the OD command. The point shifts automatically as data I/O progresses. If you want to obtain data in sequence, specification is only necessary once.

(3) If you want to obtain data in ASCII format, use the QDA command. If you want to obtain data in binary format, use the QDB command. Data which can be output in one batch is from 1 to 50 for QDA or 1 to 250 for QDB. (GP-IB only)

(Note) • Total processing time is reduced by outputting more data in each batch.
• With binary output, ASCII control codes are also output. If the data string includes a string that is the same as a delimiter, data will not be output correctly. (GP-IB only)

[GP-IB]

Program sample..... Obtaining channel 1 data (2000 pcs) stored with a SHOT length of 40 DIV
(Sample 1: ASCII output with HP9816)

```
100 ! ---------------------------------- !
110 !   8832  Data Out (ASCII) HP9816    !
120 ! ---------------------------------- !
130 !
140 DIM Ch1(2000),Da(49)
150   Adr=705
160     OUTPUT Adr;"GH0,QMX"
170     ENTER Adr;Mx
180     IF Mx<>2000 THEN 310      !Enable Check
190 !
200 OUTPUT Adr;"OD1,0"            !Output Point CH1,No.0
210 FOR I=0 TO 39
220   OUTPUT Adr;"QDA50"
230     ENTER AdR;Da(*)
240     FOR J=0 TO 49
250       Ch1(50*I+J)=Da(J)
260     NEXT J
270 NEXT I
280   OUTPUT Adr;"QDA1"
290    ENTER Adr;Ch1(2000)        !Last Data
300 !
310 END
```

[GP-IB]

(Sample 2: Binary output with HP9816)

```
100 ! ---------------------------------- !
110 !   8832  Data Out (BINARY) HP9816   !
120 ! ---------------------------------- !
130 !
140 DIM Ch1(2000),Da(249)
150   Adr=705
160     OUTPUT Adr;"GH0,QMX"
170     ENTER Adr;Mx
180     IF Mx<>2000 THEN 310      !Enable Check
190 !
200 OUTPUT Adr;"OD1,0"            !Output Point CH1,No.0
210 FOR I=0 TO 7
220   OUTPUT Adr;"QDB250"
230     ENTER Adr USING "B";Da(*)
240     FOR J=0 TO 249
250       Ch1(250*I+J)=Da(J)
260     NEXT J
270 NEXT I
280   OUTPUT Adr;"QDB1"             !LAST DATA
290     ENTER Adr USING "B";Ch1(2000)
300 !
310 END
```

[GP-IB]

(Sample 3: ASCII output with PC9801)

```
100 '-------------------------------------'
110 '  8832   Data Out (ASCII) PC9801    '
120 '-------------------------------------'
140 DEFINT D : DIM D(2000)
150 ADR=5
160 ISET IFC : ISET REN
170 '
180 PRINT@ ADR;"FN1,SH1,ST"       'MEM,40DIV,<START>
190 PRINT@ ADR;"GH0,QMX"          'HEADER OFF
200 INPUT@ ADR;MX
210 IF MX<>2000 THEN 310
220 '
230 PRINT@ ADR;"OD1,0"            'Output Point CH1,No.0
240   FOR I=0 TO 1990 STEP 10
250     PRINT@ ADR;"QDA10"
260     INPUT@ ADR;D(I),D(I+1),D(I+2),D(I+3),D(I+4)
                  ,D(I+5),D(I+6),D(I+7),D(I+8),D(I+9)
270     NEXT I
280     PRINT@ ADR;"QDA1"         'Last Data
290     INPUT@ ADR;D(2000)
300 '
310 WBYTE &H5F;
320 IRESET REN
330 END
```

[RS-232C]

(Sample 2: Binary output with IBM PS/2)

```
100 '-----------------------------------------------'
110 '       8832  RS-232C DATA OUT (ASCII)          '
120 '-----------------------------------------------'
130 CLS
140 DEFINT D : DIM D(2000)
150 OPEN  "COM1:1200,N,8,1,LF" AS #1
160 PRINT #1,"GH0,FN1,SH1,TM0"        'MEM,40DIV,<START>
170 GOSUB 400
180 PRINT #1,"QMX"                    'HEADER OFF ,MAX POINT ?
190 INPUT #1,A$ : PRINT A$ : A%=VAL(A$)
200 PRINT "MAX POINT = ";A%
210 IF A%<>2000 THEN STOP             'SHOT = 40DIV ?
220 '
230 PRINT #1,"OD1,0"                  'OUTPUT POINT CH1,NO.0
240 PRINT TIME$
250  FOR I=0 TO 1990 STEP 10
260 '  GOSUB *WWW :'PRINT I,
270   PRINT #1,"QDA10"                'STORAGE DATA INPUT
280   INPUT #1,D(I),D(I+1),D(I+2),D(I+3),D(I+4),D(I+5)
            ,D(I+6),D(I+7),D(I+8),D(I+9)
290   NEXT I
300   PRINT #1,"QDA1"
310   INPUT #1,D(2000)
320 PRINT TIME$
330 '
340   FOR I=0 TO 2000
350 LOCATE 10,10:   PRINT USING "D(####)  = ####";I,D(I)
360   NEXT I
370 CLOSE #1
380 END
390 '*WWW
400 FOR W=0 TO 200 : NEXT
410 RETURN
```

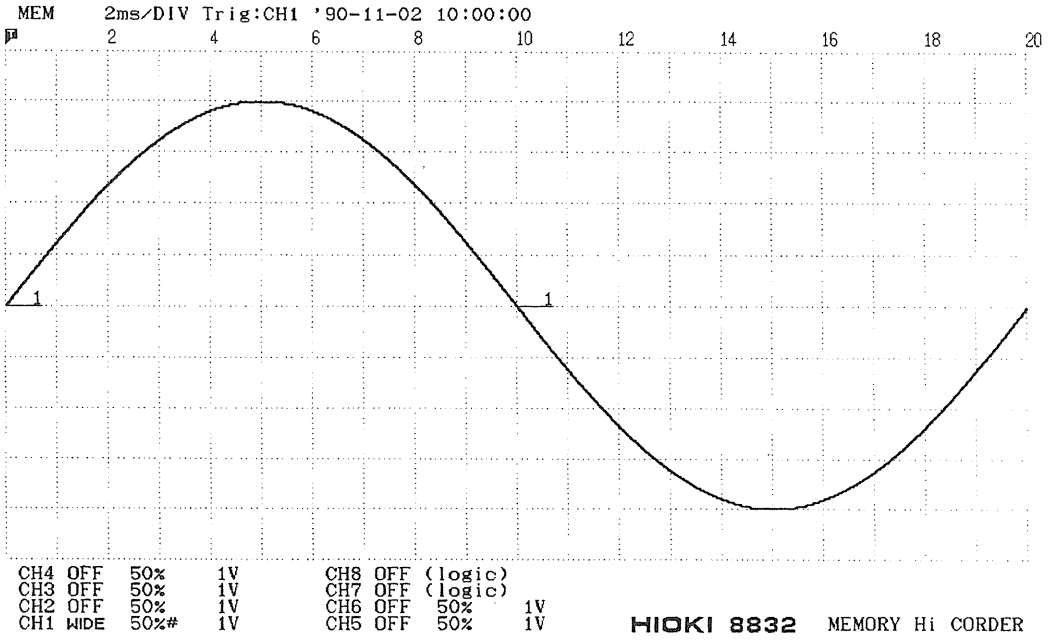
3-4-4 Storage data input

(1) Use thee QMX command to check whether or not data I/O is possible. If the value output by QMX is "0", data I/O is not possible.

(2) Specify the input data channel and point using the OD command, then issue the DA command.

(Note) As in the case of storage data output, processing time is reduced by inputting as much data as possible in each batch.

Program sample..... When data is stored with a recording length of 20DIV, the following program sets a SIN waveform in CH 1.



(Waveform after program execution)

[GP-IB]

(Sample for HP 9816)

```
100 ! ---------------------------------- !
110 !   8832   Data Input       HP9816    !
120 ! ---------------------------------- !
130 !
140 Adr=705
150   OUTPUT Adr;"GH0,QMX"
160     ENTER Adr;Mx
170     IF Mx<>1000 THEN 250
180 !
190   DEG
200   OUTPUT Adr;"OD1,0"
210   FOR I=0 TO 1000
220     OUTPUT Adr;"DA";INT(100*SIN(36*I/100)+125.5)
230   NEXT I
240 !
250 END
```

[GP-IB]

(Sample for PC 9801)

```
100 '-------------------------------------'
110 '  8832   Data Input        PC9801     '
120 '-------------------------------------'
130 ADR=5
140 ISET IFC : ISET REN
150 '
160 PRINT@ ADR;"GH0,QMX"          'HEADER OFF
170 INPUT@ ADR;MX
180 IF MX<>1000 THEN 250
190 '
200 PRINT@ ADR;"OD1,0"            'Input Point CH1,No.0
210   FOR I=0 TO 1000
220     PRINT@ ADR;"DA"+STR$(INT(100*SIN(3.14*I/500)+125))
230   NEXT I
240 '
250 WBYTE &H5F;
260 IRESET REN
270 END
```

[RS-232C]

(Sample for IBM PS/2)

```
100 '-----------------------------------------------'
110 '        8832  RS-232C  DATA INPUT               '
120 '-----------------------------------------------'
130 '
140 OPEN "COM1:19200,N,7,1" AS #1
150 PRINT #1,"GH0,XO1,QMX"                    'MAX POINT?
160 INPUT #1,A%
170 PRINT "MAXPOINT =";A%
180 IF A%<>1000 THEN 340                      'MAX POINT=1000?
190 PRINT #1,"OD1,0"
200 '
210 DIM DAT(1000)
220 PRINT "MAKE DATA"
230 FOR I=0 TO 1000
240   DAT(I)=(INT(100*SIN(3.14*I/500)+125))
250 NEXT I
260 '
270 PRINT "TRANSFER DATA"
280 FOR I=0 TO 1000
290 '    PRINT USING"####";I;
300      PRINT #1,"DA"+STR$(DAT(I))             'DATA INPUT
310 '    FOR J=0 TO 100 : NEXT                  'WAIT
320 NEXT I
330 '
340 CLOSE #1
350 END
```

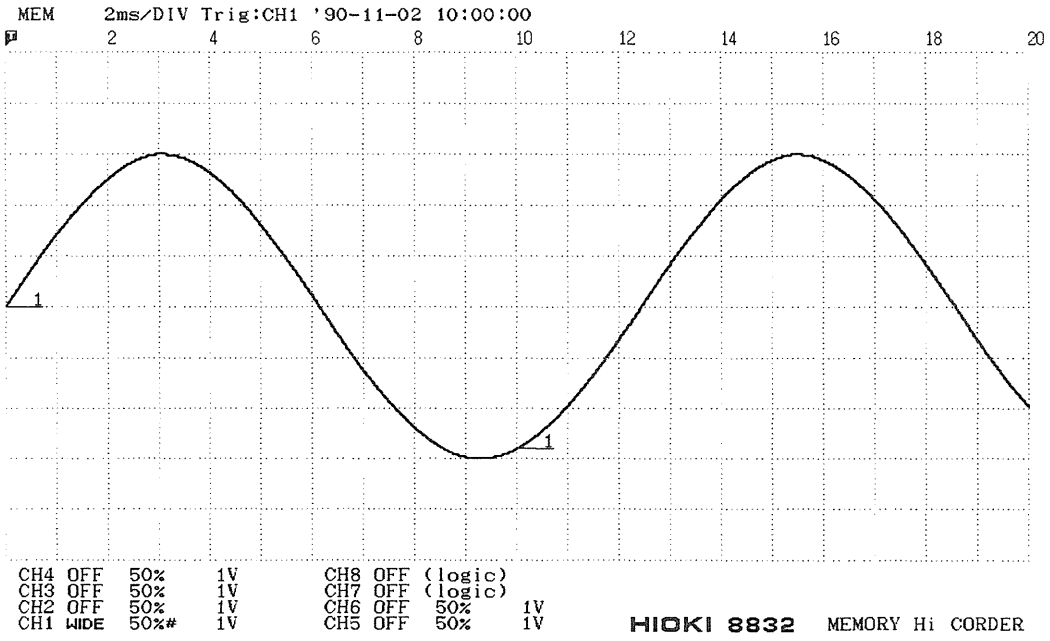
3-5 Using Special Functions

3-5-1 PEAK value computation (QPK)

- When analog units are used, minimum and maximum values of stored data can be computed.
- Calculation of minimum and maximum values is performed on all stored data. Both numeric and converted voltage values are output.
- If scaling is selected, the converted voltage value is output at the scaled value. (See 3-5-8.)
- The output format is as follows:

Minimum value	,	Maximum value	,	Minimum voltage	,	Maximum voltage

(Output sample)



3-5-2 Average value computation (QME)

- When analog units are used, the average and variance of stored data can be computed.
- The following formula is used:

$$\text{Average value } \bar{d} = \frac{1}{n+1} \sum_{i=0}^{n} d_i \qquad d_i\text{: i-th datum}$$

$$\text{Variance value } V = \frac{1}{n+1} \sum_{i=0}^{n} (d_i - \bar{d})^2$$

d_i : i-th datum

$\bar{d}_i$: Average value

(Note) Standard deviation is the square root of the variance.

3-5-3 Area computation (QAR, QFX)

- The area between data from two analog units can be computed.
- The following formula is used, where the unit are $(DIV)^2$.

(Note) If the same channel is specified for both analog units, the area will be computed for that channel against reference zero.

$$\text{Area } A = \frac{1}{25 \times 50} \left\{ \frac{|d_{10} - d_{20}|}{2} + \sum_{i=1}^{n-1} |d_{1i} - d_{2i}| + \frac{|d_{1n} - d_{2n}|}{2} \right\}$$

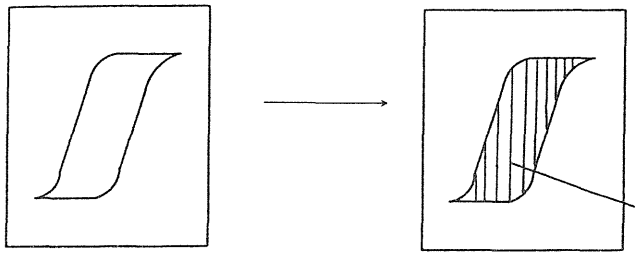
d_{1i} : i-th datum of channel 1

d_{2i} : i-th datum of channel 2

- Areas of XY waveforms can be calculated.
- The XY waveform is drawn by PRINT TYPE LINE and the area enclosed by the line is calculated, including the line itself.

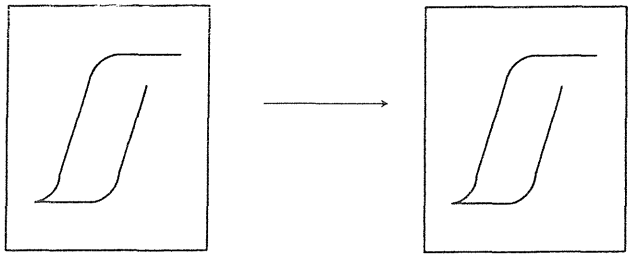
Computed area

XY waveform



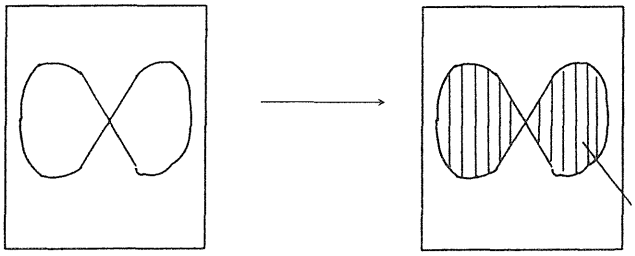
Area computed

XY waveform (if no encircled area)



Area=Line area

XY waveform



Area computed

3-5-4 Differential and integral (DI, IN)

- When analog units are used, differentials and integrals of stored data can be computed.
- For differentials, the source channel must be different from channel in which the computed result is stored. With integrals, both channels can be the same.
- For both differentials and integrals, the coefficient K (1 - 50) must be specified. For differentiation is the interval between points where the difference is obtained;For integration, it is the divisor of added result.
- The computation assumes that the 50% position (125 in data) is the 0 position.
- The following expression is used:

Differential $b_i = (d_{i+K/2} - d_{i-K/2}) + 125 \quad (i = 0, 1, \ldots\ldots, n)$

b_i : i-th computed result datum

d_i : i-th datum of source channel data

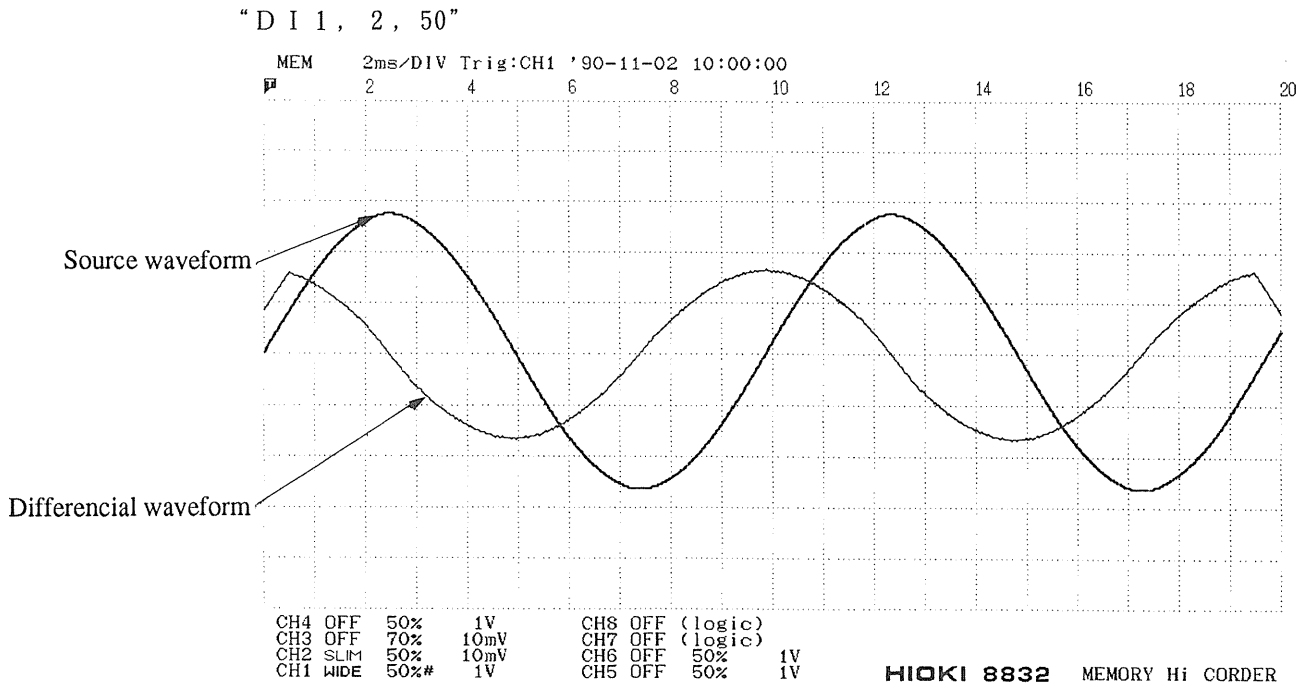
Integral $b_i = \left\{ \frac{1}{K} \sum_{t=0}^{i} (d_t - 125) \right\} + 125 \quad (i = 0, 1, \cdots\cdots, n)$

b_i : i-th computed result datum

d_i : i-th source channel datum

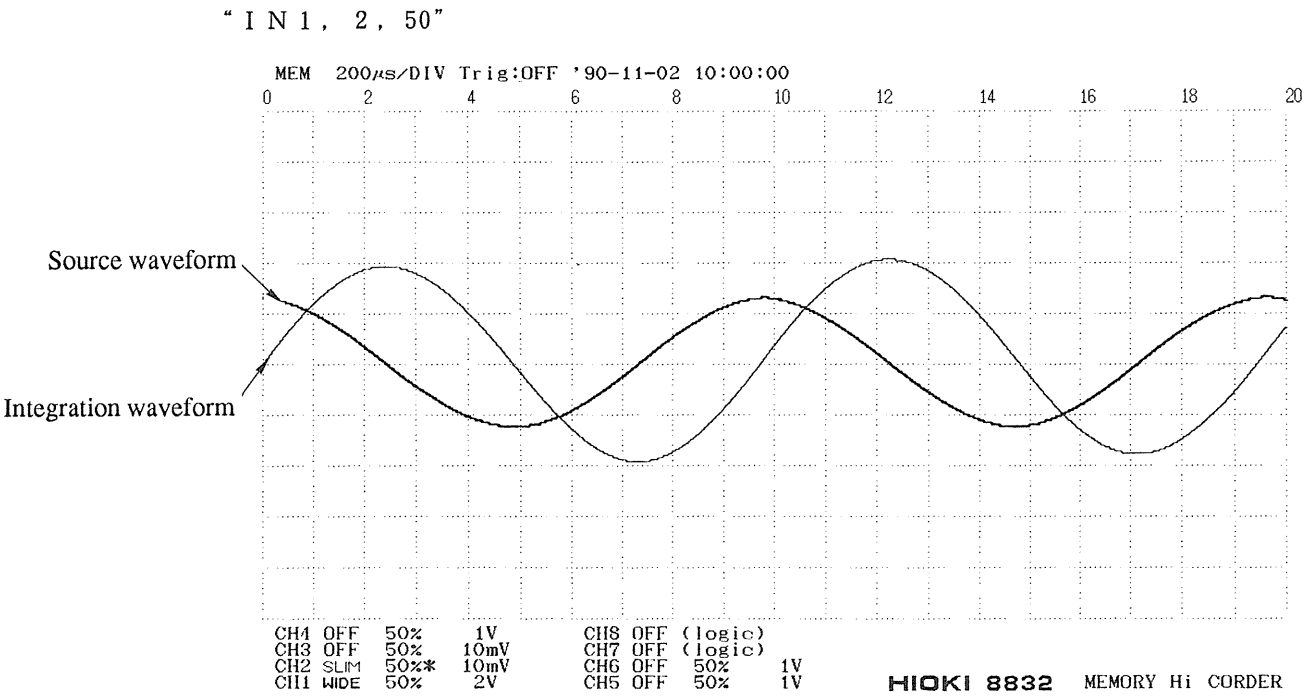
(Computation sample 1)

Differentiation of CH1 waveform (SIN wave) and input of result to CH2.



(Computation sample 2)

Integration of CH1 waveform (SIN wave) and input of result to CH2.



3-5-5 SWAP Function (SW)

- Stored data can be swapped between channels. This is illustrated below.

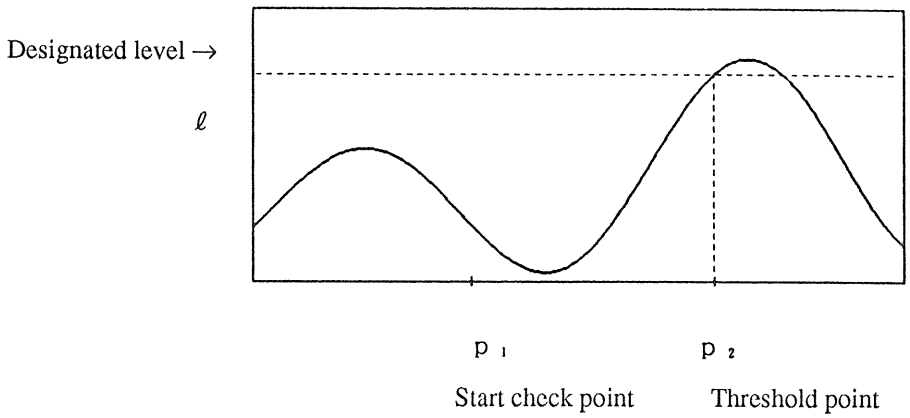
$d_{1i} \rightleftarrows d_{2i}$ (i = 0, 1, ..., n)

d_{1i}: Channel 1's i-th datum

d_{2i}: Channel 2's i-th datum

3-5-6 Computation of the point of threshold passage (QCP)

- The point of passage through a specified threshold (threshold point) can be calculated for stored data, such as the internal trigger for the analog unit.



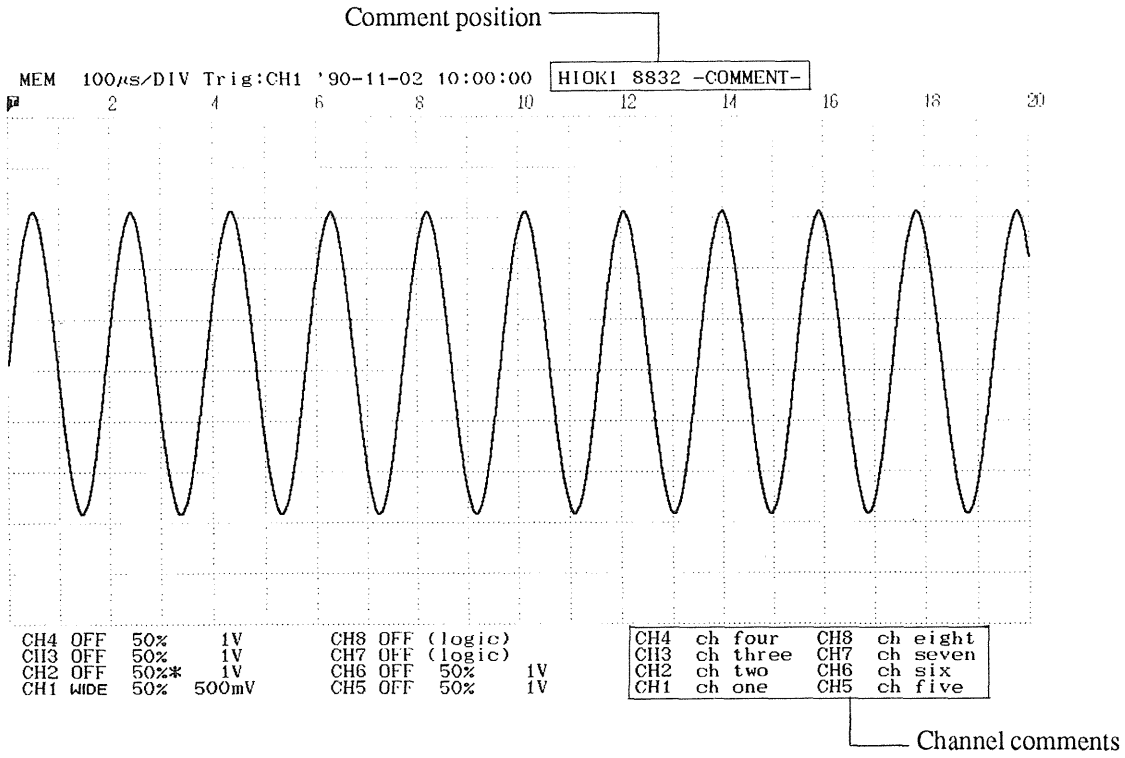
- Starting from a specified start check point P_1 (0-125000), search for the point of passage through the threshold level (-3 - 252) and slope (up or down) at the point of passage. The first point P_2 located will be output.

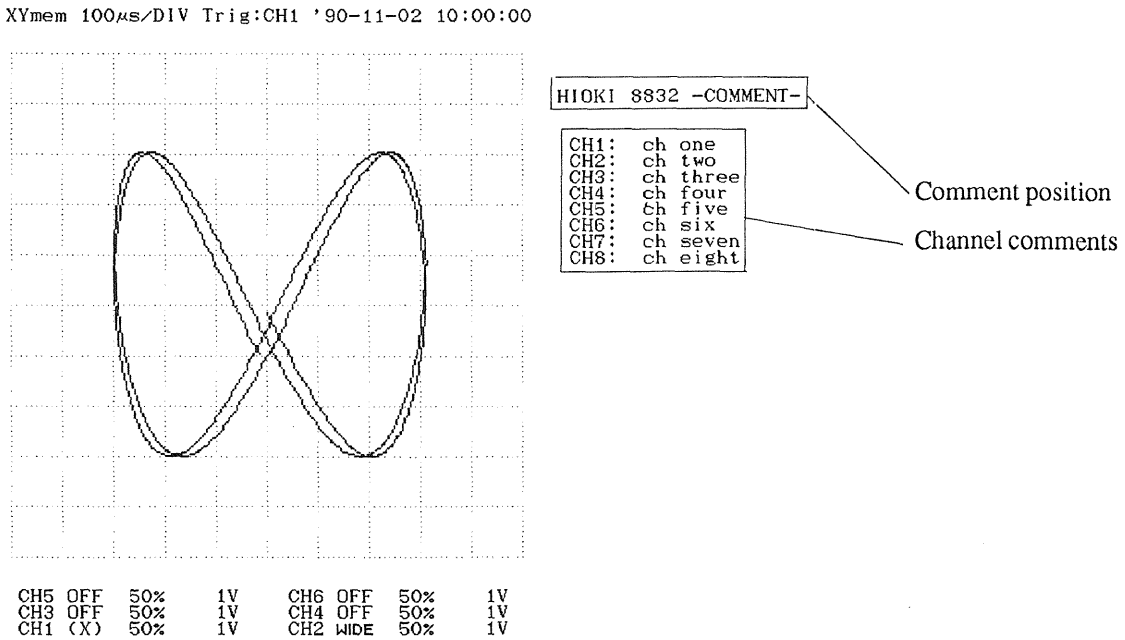
(Note) If the threshold is not passed, "-1" is output.

3-5-7 Comment printing (CO, CM, CN, CC)

- A comment of up to 20 characters can be printed at the top of the chart, and comments of up to 10 characters can be printed for each channel.
- Comments are printed in the locations indicated in the illustration below.
- Characters which can be printed are:
  8832... Letters (uppercase only), numbers (0 - 9), space, ^, ( ), . $ % & = + – * / μ (~u)° (~*)

(Note) Characters other than the above are replaced by spaces.





- The command format is as follows. The command must end with a terminator.

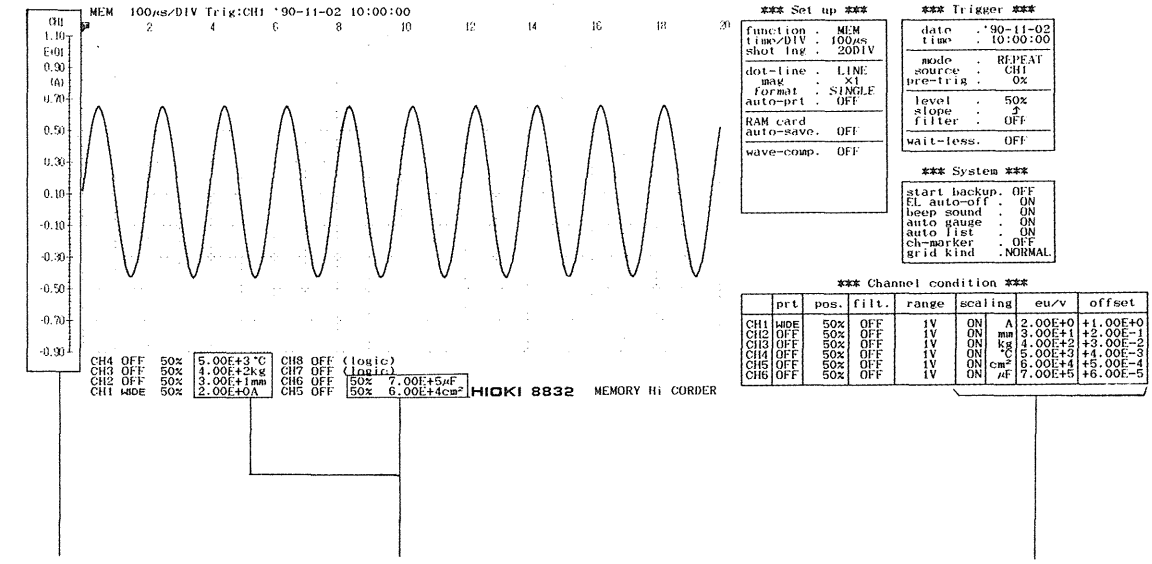
C	M	Comment characters	Terminator

- In order to clear the comment, use:

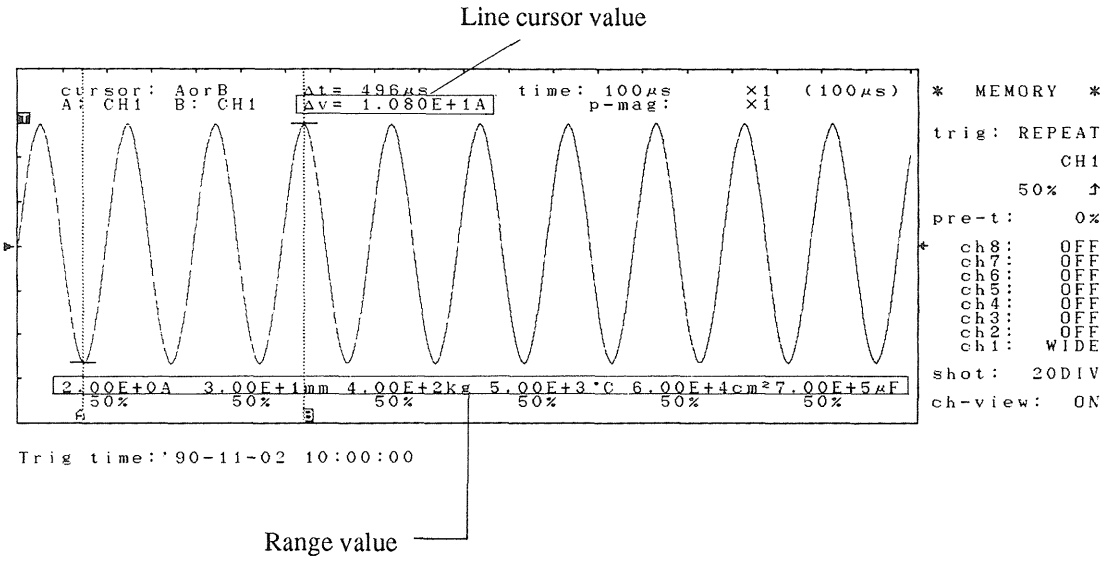
C	M	Terminator

3-5-8 Scaling commands (SC, SE, SV, SO)

- The scaling commands set scaling unit (eu), the conversion value (eu/v), and the offset (offset).
- The scaling settings and related values are printed in the locations indicated in the illustration below.



Gauge display Range value Scaling settings



Unit (eu) setting: (SEc, n$)

Use this command to specify characters (up to 3) displayed for the unit on each channel.

The SE command sets these characters to the channel specified in c.

Characters that can be input for unit display are letters (uppercase and lowercase), spaces, and the symbols $, %, &, =, +, –, *, /, ^{2}, ^{3}, μ, and ° special format is used to input the characters 2, 3, m, and)specify these characters by typing ~ (7EH), followed by the character 2, 3, u, or *. Meanings of each combination are as follows.

^{2} = ~2

^{3} = ~3

μ = ~u

° = ~*

Examples: SE1, OHM

rpm

SE1, cm~2

cm~3

SE1, ~*C

Setting the conversion value: SVc, x. xxE ± x

Input the conversion value in the following format:

x. xxE ± x

Example: SV1, 2.00E+1

Setting the offset: SOc, ± x. xxE ± x

Input the offset in the following format:

± x. xxE ± x

Examples: SO, –1.00E–1

SO, +1.00E+0

Chapter 4

Command List

4-1 Command List

- The [DCL] command is not an ASCII code command. It is a device clear or select device clear command to the GP-IB.
- If a command is given when essential conditions are not satisfied, a command error occurs (error 53).
- In the "Conditions" column, "Storage data I/O possible" means that the function is used when MEM or XY$_{MEM}$ data is present.
- All parameters without explanation are ASCII values.
- **[GP]** indicates GP-IB command, and **[RS]** indicates an RS-232C command.

4-2 8832 Command List

- Execution processing

Item	Set command	Read command	Conditions	Explanation
START processing	ST	—	Ant time other than during display of the SYSTEM screen or an IC card screen	Same as [START] key at time of waveform fetching (Note) Wait at least 100 ms after processing before sending the next command. [RS]
STOP processing	[GP] [DCL]	—		Same as [STOP] key (Note) Wait is needed after processing.
	[RS] [DCL]	—		Same as [STOP] key (Note) Wait is needed after processing.
Print processing	PR	—	MEM XY_{MEM} XY_{CONT}	Same as [PRINT] key
LIST Processing	LS	—	REC MEM XY_{MEM} XY_{CONT}	Same as [FEED]+[COPY] key. *1
COPY Processing	HC	—		Same as [COPY] key
FEED processing	FD ℓ	—		ℓ ..... Paper feed length (in mm) $6 \leqq \ell \leqq 255$
Auto TIME/DIV	AT	—	MEM XY_{MEM} Other than trigger source OFF	Execute time axis automatic setting Same as [←] + [→] key *1
GAUGE processing	GA	—	REC MEM XY_{MEM} XY_{CONT}	Same as [FEED]+[PRINT] key *1

*1 "+" indicates pressing keys simultaneously.

• Execution processing

Item	Set command	Read command	Conditions	Explanation
Back light ON/OFF	EBe	—	XY_{CONT}	Same as [BACK LIGHT] key. e ..... ON/OFF (0 - 1) { 0 ... OFF { 1 ... ON
System reset	SR	—	XY_{CONT}	Initialize 8832 Same as power on while pressing [STOP] key.
Level monitor	LV	—		Level monitor display. Same function as pressing [UP] + [DOWN] *1. Display returns to the normal upon device clear or command execution.

*1 "+" means pressing the keys at the same time.

• Display mode

Item	Set command	Read command	Conditions	Explanation
Screen mode	MW m	QMW	REC MEM XY MEM XY CONT SYSTEM	m ...Screen mode (0-1) { 0...WAVE screen 1...MENU screen
Screen size	WS w	QWS	REC MEM XY MEM XY CONT	w ...Screen size (0-1) { 0...Standard 1...Reduction (1/2)
Function	FN f	QFN		f.....Function(0-4) { 0 ...REC 2 ... XY MEM 1 ...MEM 3 ... XY CONT 4 ... SYSTEM

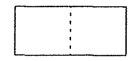
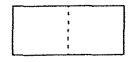
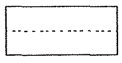
- Measuring condition setting

Item	Set command	Read command	Conditions	Explanation
time/DIV	TD t_1	QTD	REC	t_1....time/DIV (0-11) 0 ...1s 6 ... 1min 1 ...2s 7 ... 2min 2 ...5s 8 ... 5min 3 ...10s 9 ... 10min 4 ...20s 10 ... 20min 5 ...50s 11 ... 50min
	TD t_2	QTD	MEM XY_{MEM}	t_2.... time/DIV(0-14) 0 ...100μs 8 ... 50ms 1 ...200μs 9 ... 100ms 2 ...500μs 10 ... 200ms 3 ...1ms 11 ... 500ms 4 ...2ms 12 ... 1s 5 ...5ms 13 ... 2s 6 ...10ms 14 ... 5s 7 ...20ms
Recording length	SH s_1	QSH	REC	s_1 ...Recording length (0-6) 0 ...20DIV 4 ... 300DIV 1 ...40DIV 5 ... 600DIV 2 ...80DIV 6 ... CONT 3 ...160DIV
	SH s_2	QSH	MEM XY_{MEM}	s_2 ...Recording length (0-7) 0 ...20DIV 4 ... 300DIV 1 ...40DIV 5 ... 600DIV 2 ...80DIV 6 ... 1200DIV 3 ...160DIV 7 ... 2500DIV
Compression/ enlargement rate	MG m	QMG	MEM	m ...Compression/enlargement rate (0-9) 0 ... ×1/100 5 ... ×1/2 1 ... ×1/50 6 ... ×1 2 ... ×1/20 7 ... ×2 3 ... ×1/10 8 ... ×4 4 ... ×1/5 9 ... ×8

• STATUS unit

Item	Set command	Read command	Conditions	Explanation
LCD Clear	DC d	QDC	XY_{CONT}	d....Display clear (0-1) { 0 ...OFF 1 ... ON
Printer output	PO p	QPO	REC	p....Printer (0-1) { 0 ...OFF 1 ...ON
Auto print	PO p	QPO	MEM XY_{MEM}	p....Printer (0-1) { 0 ... OFF 1 ...ON

• WAVE screen

Item	Set command	Read command	Conditions	Explanation
Cursor ON	AB c,t	QAB	REC MEM XY MEM XY CONT	c ....ON/OFF of cursor (0-3) { 0 – OFF 1 – A 2 – A or B 3 – A & B t.....Type of cursor (0-2) { 0 – ↔ Δ t 1 – ↔ Δ f 2 – ↕ (Other than MEM)
A cursor position	MA p	QMA	REC MEM XY MEM XY CONT	p ....A cursor position • Horizontal movement [↔] REC MEM ...(0-503)  0 503 XY MEM XY CONT ...(0-255)  0 255 •Vertical movement [↕] REC XY MEM ...(0-255)  255 0 XY CONT
B cursor position	MB p	QMB	REC MEM XY MEM XY CONT	p ....B cursor position (same as A cursor)
Cursor read output	—	QVA (VO d$)	REC MEM XY MEM XY CONT	d$ ..Cursor read output character string (Example) d$="100.01mV, 10.01ms"

• WAVE screen

Item	Set command	Read command	Conditions	Explanation
Waveform evaluation	WJ	—	MEM Judgment area should exist Judgment mode should exist Stored data should exist	For judgment refer to status byte.
Area generation	WM	—	MEM Judgment mode should exist Stored data should exist	Judged area generation
Channel review	CV c	QCV	REC MEM	c ....ON/OFF (0-1) { 0 ...OFF { 1 ...ON
A cursor channel	AC c	QAC	MEM Drawing is other than OFF	c ....Channel (1 - 6)
B cursor channel	BC c	QAC	MEM Drawing is other than OFF	c ....Channel (1 - 6)

• Trigger unit

Item	Set command	Read command	Conditions	Explanation
Trigger mode	TM c	QTM	MEM XY_{MEM}	c ....Trigger mode (0-2) 0 ...Single 1 ...Repeat 2 ...AUTO
	TM c	QTM	At REC Recording length is other than CONT.	c ....trigger mode (0-1) 0 ...Single 1 ...Repeat
Trigger source	TS s	QTS		s.....Trigger source (0-11) 0 ...OFF 4 ... CH4 8....CH8 1 ...CH1 5 ... CH5 9....EXT 2 ...CH2 6 ... CH6 10....MANU 3 ...CH3 7 ... CH7 11....TIMER
Trigger level	TL ℓ	QTL	Trigger source CH1-CH6	ℓ ..Trigger level (In %) $0 \leqq \ell \leqq 100$
Trigger slope	SL s	QSL	Trigger source CH1-CH6	s.....Trigger slope (0-1) 0 ...UP (↑) 1 ...DOWN (↓)
Trigger filter	TF f	QTF	MEM XY_{MEM}	f..... Trigger filter (0-6) 0 ...OFF 4 ... 0.6DIV 1 ...0.1DIV 5 ... 1.3DIV 2 ...0.2DIV 6 ... 2.6DIV 3 ...0.3DIV
	TF f	QTF	REC XY_{CONT}	f.....Trigger filter (0-1) 0 ...OFF 1 ...ON

• Trigger unit

Item	Set command	Read command	Conditions	Explanation
Trigger timing	TT t	QTT	REC XY_{CONT}	t.....Trigger timing (0-1) 0 ...START 1 ...STOP
Pre-trigger	PT p	QPT	MEM XY_{MEM}	p.... Pre-trigger (0-7) 0 ...0% 4 ... 75% 1 ...5% 5 ... 95% 2 ...25% 6 ... 100% 3 ...50% 7 ... -95%
Trigger waitless	TW w	QTW	MEM XY_{MEM}	w ...Waitless (0-1) 0 ...OFF 1 ...ON
Timer trigger start time	TA mm-dd. hh-mm	QTA	REC MEM XY_{MEM} XY_{CONT}	mm... Month (01-12) *2 dd..... Day (01-31) hh..... Hour (00-23) MM.. Minute (00-59)
Timer trigger stop time	TO mm-dd. hh-MM	QTO	REC MEM XY_{MEM} XY_{CONT}	mm... Month (01-12) *2 dd..... Day (01-31) hh..... Hour (00-23) MM.. Minute (00-59)
Timer trigger interval	IV hh:MM:SS	QIV	REC MEM XY_{MEM} XY_{CONT}	hh..... Hour (00-99) *2 MM.. Minute (00-59) SS..... Second (00-59)
Logic trigger	LA n_1, n_2, n_3	QLA	REC MEM XY_{MEM} XY_{CONT} Trigger source CH7, CH8	n_1 ..... Trigger pattern (low) 1...Low n_2 ..... Trigger pattern (Hi) 1...Hi Each corresponds in 8-ch bit (0-255) n_3 ..... AND/OR (0-1) 0 ....AND 1...OR

*2 Set 2 digits for each item.

• Print unit

Item	Set command	Read command	Conditions	Explanation
DOT, LINE	DL d	QDL	REC XY_{MEM} XY_{CONT}	d....DOT/LINE (0-1) 0 ...DOT 1 ...LINE
	DL d	QDL	MEM	d....DOT/LINE (0-2) 0 ...DOT 1 ...LINE 2 ...SMOOTH
DRAWING	DR c,d	QDR c	REC MEM XY_{MEM} XY_{CONT}	c ....Channel (1-8) Channels 1 to 6 with XY_{MEM} and XY_{CONT} d....Draw type (0-7) Analog / Logic 0 ...OFF / 4 ... OFF 1 ...SLIM / 5 ... Ach 2 ...WIDE / 6 ... Bch 7 ... A&Bch
Print format	FM f	QFM	REC MEM	f.....Print format (0-2) 0 ...SINGLE 1 ...DUAL 2 ...QUAD
Partial print enlargement	PM m	QPM	MEM	m ...Format (0-9) 0 ...× 1/100 1 ...× 1/50 2 ...× 1/20 3 ...× 1/10 4 ...× 1/5 5 ...× 1/2 6 ...× 1 7 ...× 2 8 ...× 4 9 ...× 8

• SPECIAL unit

Item	Set command	Read command	Conditions	Explanation
START state back up	BT t	QBT		t.....Back up (0-1) { 0 ...OFF 1 ...ON
Grid type	GR g	QGR		g....Grid type (0-2) { 0 ...OFF 1 ... NORMAL 2 ...FINE
Auto list	LT ℓ	QLT		ℓ ..Auto list (0-1) { 0 ...OFF 1 ...ON
Auto gage	GG g	QGG		g....Auto gage (0-1) { 0 ...OFF 1 ...ON
EL backlight auto off	EL e	QEL		e ....Auto off (0-1) { 0 ...OFF 1 ...ON
Beep sound	BPb	QBP		b ....Beep sound (0-1) { 0 ...OFF 1 ...ON
Channel marker	MKm	QMK		m ...Channel marker (0-1) { 0 ...OFF 1 ...ON
Self-checking execution	SF t	—	SYSTEM	t.....type of self-check (1-5) { 1 ...ROM/RAM check 2 ... LED check 3 ...Printer check 4 ...Key check 5 ...LCD check

• SPECIAL unit

Item	Set command	Read command	Conditions	Explanation
Clock setting	RT yy-mm-dd, HH:MM:SS	—	SYSTEM	yy.....Year (00-99) mm .....Month (01-12) dd.....Date (01-31) HH .....Hour (00-23) MM.....Minute (00-59) SS.....Second (00-59) (Note) Each to be set using two digits.
Clock output	—	QRT (RT t$)		t$...Clock (character string) (Example): "90-11-08, 15:24:17"
RAM card auto save	IA t	QIA	MEM XY$_{MEM}$	t.....Auto save (0-1) { 0 ... OFF 1 ...ON
Waveform evaluation	WC m, u, d, r, l, s	QWC	MEM	m... Judgment mode (0-4) { 0 ...OFF 3 ... MODE3 1 ...MODE1 4 ... MODE4 2 ...MODE2 u ....Judgment area top (0-255 dot) (×0.04(DIV)) d ....Judgment area bottom (0-255 dot) (×0.04(DIV)) r.....Judgment area right (0-255 dot) (×0.02(DIV)) l.....Judgment area left (0-255 dot) (×0.02(DIV)) s.....Stop mode (0-2) { 0 ...GO 1 ...NG 2 ...GO&NG

• Interface control

Item	Set command	Read command	Conditions	Explanation
GP-IB delimiter	[GP] GD d	—		d....GP-IB output delimiter (0-3) 0 ...CRLF (EOI) 1 ...CR (EOI) 2 ...LF (EOI) 3 ... (EOI)
Header	GH h	—		h....Header (0-1) 0 ...OFF 1 ...ON
Error number output	—	QER (ER e)		e ....Error number (0-54) 0-20...General error 31-43...WARNING 51-54...GP-IB error [GP] Refer to 1-3-6 [RS] Refer to 2-3-6
Machine type and model number output	—	QID (ID i)		i.....Machine's type and model number (8832)
SRQ response masking	[GP] MS m	QMS		m ...SRQ response masking value (0-3) 8-3 | 2 | 1 / | START end | Error occurrence •SRQ clear (Refer to 1-3-5 [2].)
Buffer flow control	[RS] XOx	QXO		x ....Buffer flow control 0 ...OFF 1 ...ON (Refer to 2-3-7)

• Interface control

<table>
<tr><th>Item</th><th>Set command</th><th>Read command</th><th>Conditions</th><th>Explanation</th></tr>
<tr><td rowspan="2">Initialization</td><td>[GP]
[DCL]</td><td>—</td><td></td><td>• GB-IB initialization (Refer to 1-3-4.)
(Note) WAIT is needed after processing.</td></tr>
<tr><td>[RS]
[ESC]C</td><td>—</td><td></td><td>• RS-232C initialization (Refer to 2-3-4.)
(Note) WAIT is needed after processing.</td></tr>
<tr><td>Status byte output</td><td>—</td><td>QUS
(US s)</td><td></td><td>s.....Status byte
<table><tr><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td></tr><tr><td>Waveform judgemnt</td><td>SRQ</td><td colspan="2">Operating mode</td><td>Printer</td><td>WAIT for Trigger</td><td>START</td><td>Error</td></tr></table>[GP] Refer to 1-3-5
[RS] Refer to 2-3-5</td></tr>
<tr><td>Key lock</td><td>[RS]
UKu</td><td>QUK</td><td></td><td>u ....Key lock
{ 0 ...Unlock
{ 1 ...Key lock
{ (Refer to 2-3-3)</td></tr>
</table>

• Input unit

Item	Set command	Read command	Conditions	Explanation
Type of input unit	—	QAM c (AMc,n)		c ....Channel (1-6) n ....Type of input unit (0-15) { 8...8934 Analog unit 9...8932 Analog unit 15...Unloaded
Range of analog unit	—	QAA c (AA c, n_1, n_2, n_3, n_4)	Analog unit	c ....Channel (1-6) n_1...Input voltage range (numeric part) (-1~13) { 0 ...0.1 8....50 1 ...0.2 9....100 2 ...0.5 10....200 3 ...1 11....500 4 ...2 12....1000 5 ...5 13....2000 6 ...10 7 ...20 -1....Undefined range n_2...Input voltage range (unit part) (-1~4) { 0 ...mV/DIV 3 ... Vrms/DIV 1 ...V/DIV 4 ... °C/DIV 2 ...mVrms/DIV -1 ... No unit n_3...Position (-10~10) { -10... -100% -9... -90% 0... 0% 9... 90% 10... 100% n_4...Type of filter (0-2) { 0 ...OFF 1 ... 5Hz 2 ...500Hz

• IC Card

Item	Set command	Read command	Conditions	Explanation
IC card mode start	IC	—		Same as [IC CARD] key
IC card mode end	IE	—	IC card mode	Return to normal mode
INIT processing	II	—	IC card screen	Initializes the RAM card.
KILL processing	IK n	—	IC card screen	n....File number (1-65535)
LOAD	LD n		IC card screen	n....File number (1-65535)
Save file name	NA n$	—	IC card screen	n$..File name (Up to 10 characters) (Note) Terminator required (:)
IC card test	—	QIT (IT t)	IC card screen	t.....Test result 0 ...OK -1 ..NG
SAVE processing	IS t, ch, n	—	IC card screen	t..... Type (0-2) 0 ...FUNC 1 ...WAVE 2 ...AREA ch ..Channel (1-10) 1 ...CH1 5 ...CH5 9...CH1 to CH4 2 ...CH2 6 ...CH6 10...CH5 to CH8 3 ...CH3 7 ...CH7 4 ...CH4 8 ...CH8 n....File name (0-1) 0 ...Skipped 1 ...ON *"ch" is valid for WAVE only.

• IC Card

Item	Set command	Read command	Conditions	Explanation
IC card setting status		QIC (IC i,p,n)		i..... IC card { 0 ... Not set { 1 ... Set p..... Protect { 0 ... Unprotected { 1 ... Protected n..... No. of files { 0~n { –1 .... IC card defective

• Realtime data

Item	Set command	Read command	Conditions	Explanation
Realtime data output (ASCII format)	—	QRA c (RAc,d)		c ....Channel (1-8) d....Data (ASCII format) { Analog ...-3 to 252 { Logic... 0 to 255 (Refer to 3-4-2.)
Output of realtime data (Binary format)	—	[GP] QRB c (d$)		c ... Channel (1-8) d$.. Data (binary format) (0 - 255) (Note) In case of analog unit, 255, 254, and 253 are -1,-2,and -3 respectively (Refer to 3-4-2.)

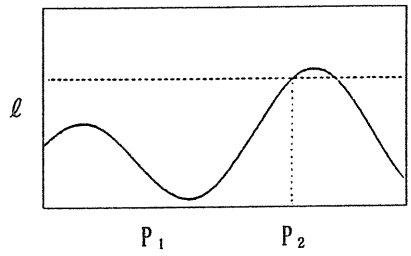
• Stored data I/O

Item	Set command	Read command	Conditions	Explanation
Informs whether stored data I/O possible	—	QMX (MX n)	MEM XY_{MEM}	n....Number of I/O possible data 0....I/O prohibited 1000....20DIV 2000....40DIV 4000....80DIV 8000....160DIV 15000....300DIV 30000....600DIV 60000....1200DIV 125000....2500DIV
I/O point designation	OD c,p	QOD	Stored data I/O possible	c....Channel (1-8) p....Point (0-125000)
Stored data output (ASCII)	—	QDA n (DA $d_1, d_2, d_3 ..., d_n$)	Stored data I/O is possible	n....Number of output data unit (1-50) d_i ...i-th data (ASCII format) Analog...-3 to 252 Logic... 0 to 255 (Refer to 3-4-3.)
Stored data output (Binary)	—	[GP] QDB n (d$)	Stored data I/O is possible	n....Number of output data unit (1 - 250) d$ ..n-byte data (binary format) d_1 d_2 d_3 d_4 ... d_{n-1} d_n No GP-IB header (Refer to 3-4-3.)
Stored data input (ASCII)	DA $d_1, d_2, d_3, d_4, ..., d_n$	—	Stored data I/O is possible	d_i ...Data (ASCII format) Analog... –3 to 252 Logic... 0 to 255 (Refer to 3-4-4.)

• Computation functions

Item	Set command	Read command	Conditions	Explanation
Voltage conversion	—	QVL, c,p (VL c,p d$)	MEM Stored data I/O possible Analog unit	c ....Channel (1-6) p .... Point (0-125000) d$ ...Voltage (character string) (Example) "-43.2mV"
PEAK value computation	—	QPK c (PK c,M_1, M_2, $P_1$$, $P_2$$)	MEM Stored data I/O possible Analog unit	c....Channel (1-6) M_1....Minimum data M_2....Maximum data $P_1$$....Minimum voltage (character string) $P_2$$....Maximum voltage (character string)
Average value / variance computation	—	QME c (ME c,m,v)	MEM Stored data I/O possible Analog unit	c ....Channel (1-6) m ...Average v ....Variance (Note) Standard deviation = $\sqrt{v}$
Area value computation	—	QAR c_1, c_2 (AR c_1, c_2, a)	MEM Stored data I/O possible Analog unit	c_1 ...Channel 1 (1-6) c_2 ...Channel 2 (1-6) a ....Area value (In: DIV^2) (Note) Where $c_1 = c_2$, 1 channel area is calculated against zero position.
XY area value computation	—	QFX c (FX c,a)	XY mem Stored data I/O possible Analog unit	c ....Y-axis channel (2-6) a ....XY area value (In: DIV^2) (Example) "48.686"
Differential computation	DI c_1, c_2, k	—	MEM Stored data I/O possible Analog unit	c_1 ...Source channel (1-6) c_2 ...Destination channel (1-8) k .... Interval (1 - 50) (Note) c_1 not same as c_2
Integral computation	IN c_1, c_2, k	—	MEM Stored data I/O possible Analog unit	c_1 ... Source channel (1-6) c_2 ...Destination channel (1-8) k ....Divider (1 - 50) (Note) c_1=c_2 is possible.

• Computation function

Item	Set command	Read command	Conditions	Explanation
Threshold point computation	—	QCP C, ℓ, s, p_1 (CPc, ℓ, s,p_1, p_2)	MEM Stored data I/O possible Analog unit	c ....Channel (1-6) ℓ ...Threshold level (–3~252) s.....Threshold slope (0-1) 0...UP (↑) 1...DOWN (↓) p_1 ...Start check point (0 - 125000) p_2 ...Threshold point -1 ... Point non-existing 0 - 125000 ... Point existing  (Refer to 3-4-5 [6])
SWAP computation	SW c_1, c_2	—	Stored data I/O possible	c_1 ...Channel (1-8) c_2 ...Channel (1-8) (Note) c_1 not same as c_2

• Special function

Item	Set command	Read command	Conditions	Explanation
Trigger detection time output	—	QTJ (TJ t$)		t$ ... Trigger detection time (character string) (Example) "90-11-08, 15:14:27"
PRINT processing optional comment	CO c	QCO		c ....optional comment ON/OFF (0 - 1) { 0 ... OFF { 1 ...ON
PRINT processing optional comment character	CM m$	QCM		m$...Comment character (Up to 20 characters) (Note) Terminator (:) is required (Refer to 3-5-7.)
PRINT processing ch optional comment	CN c	QCN (CN c)		c ....Channel optional comment ON/OFF (0~1) { 0 ...OFF { 1 ...ON
PRINT processing ch optional comment character	CC c, m$	QCC c		c ....Channel (1-8) m$.Comment string (up to 10 characters) (Note) Requires terminator (:). (Refer to 3-5-7.)
Scaling	SC c, s	QSC c (SC c, s)		c ....Channel (1-6) s.....Scaling ON/OFF { 0 ...OFF { 1 ...ON
Scaling unit	SE c, n$	QSE c (SE c, n$)		c ....Channel (1-6) n$ ..Unit characters (up to 3) To input special characters, (^{2}, ^{3}, μ, and °), input ~ (7EH) and one of characters 2, 3, u, or *. ^{2} = ~2 μ = ~u ^{3} = ~3 ° = ~* (Note) Requires terminator (:).

• Special function

Item	Set command	Read command	Conditions	Explanation
Scaling conversion value	SV c, n	QSV c (SV c, n)		c ..... Channel (1-6) n ..... Conversion value n = x.xxE ± x (x=0~9)
Scaling offset	SO c, n	QSO c (SC c, n)		c ..... Channel (1-6) n ..... Offset n = ± x.xxE ± x (x=0~9)

Chapter 5

Sample Program

(Note) Programs marked [GP-IB] are for the GP-IB interface, and those marked [RS-232C] are for the RS-232C interface.

Programs samples in this chapter are made by setting address of 8832 to 5, and use PC 9801 GP-IB (made by NEC) in MASTER MODE, address number 0.

(Sample 1) Set conditions of stored processing, and run.

[GP-IB]

(Explanation)

Line 140 ... Set 8832 address (number 5) to variables ADR.

Lines 160-170 ... Send interface clear instruction, set system to remote mode.

Lines 190-260 ... Set 8832's function, TRIG condition and PRINT condition.

Line 280 ... Run START processing under set conditions.

```
100 ' ---------------------------------- '
110 '   8832      SAMPLE PROGRAM NO.1     '
120 ' ---------------------------------- '
130 '
140 ADR=5                                 'GP-IB Address=5
150 '
160 ISET IFC                              'Interface Clear
170 ISET REN                              'Remote Enable
180 '
190 PRINT@ ADR;"FN1"                      'Function MEM
200 PRINT@ ADR;"TD3"                      'TIME/DIV 1ms
210 PRINT@ ADR;"SH0"                      'SHOT     20DIV
220 '
230 PRINT@ ADR;"TS1"                      'TRIG Source CH1
240 PRINT@ ADR;"PT1"                      'Pre-TRIG    5%
250 PRINT@ ADR;"TL60"                     'TRIG Level  60%
260 PRINT@ ADR;"SL0"                      'TRIG Slope  UP
270 '
280 PRINT@ ADR;"ST"                       '< START >
290 '
300 WBYTE &H5F;                           'UN TALK
310 IRESET REN
320 END
```

(Sample 1) Set conditions of stored processing, and run.

[RS-232C]

(Explanation)

Line 140 ... Set 8832 communication protocol and open RS-232C file.

Lines 160-230 ... Set 8832's function, TRIG condition and PRINT condition.

Line 250 ... Run START processing under set conditions.

```
100 '-------------------------------------------'
110 '      8832  SMPLE PROGRAM NO.1               '
120 '-------------------------------------------'
130 '
140 OPEN "COM1:9600,N,7,1"  AS #1        'FILE OPEN
150 '
160 PRINT #1,"FN1"                       'FUNCTION MEM
170 PRINT #1,"TD3"                       'TIME/DIV 1ms
180 PRINT #1,"SH0"                       'SHOT 20DIV
190 '
200 PRINT #1,"TS1"                       'TRIG SOURCE CH1
210 PRINT #1,"PT1"                       'PRE-TRIG 5%
220 PRINT #1,"TL60"                      'TRIG LEVEL 60%
230 PRINT #1,"SL0"                       'TRIG SLOPE UP
240 '
250 PRINT #1,"ST"                        '<START>
260 '
270 CLOSE #1                             'CLOSE
280 END
```

(Sample 2) Perform START processing. If trigger is not detected, perform STOP processing.

[GP-IB]

(Explanation)

Lines 210-230 ... Run START processing after clearing status byte.

Lines 250-260 ... Read status byte and ensure start of START processing.

Lines 280-310 ... Continue to check whether or not trigger is detected during a set duration of time.

Lines 320-350 ... If trigger not detected, perform STOP processing (SELECT DEVICE clear).

```
100 ' ---------------------------------- '
110 '   8832     SAMPLE PROGRAM NO.2     '
120 ' ---------------------------------- '
130 '
140 ADR=5                                 'GP-IB Address=5
150 '
160 ISET IFC                              'Interface Clear
170 ISET REN                              'Remote Enable
180 '
190 PRINT@ ADR;"FN1,TD3,SH0"              'MEM,1ms/DIV,20DIV
200 PRINT@ ADR;"TS1,TL60,SL0"             'TRIG CH1,60%,UP
210 PRINT@ ADR;"MS0"                      'Non SRQ
220 '
230 PRINT@ ADR;"ST"                       '< START >
240 '
250     POLL ADR,S                        'START ?
260     IF (S AND &H2)<>0 THEN 250
270 '
280   FOR W=1 TO 100
290     POLL ADR,S                        'TRIG Wait ?
300     IF (S AND &H4)<>0 THEN 370
310   NEXT W
320     PRINT "Not Trigger"
330     LA=&H20+ADR                       'ADR 5
340     WBYTE &H3F,LA,&H4;                '< STOP >
350     GOTO 410
360 '
370     POLL ADR,S                        'END ?
380     IF (S AND &H2)=0 THEN 370
390     PRINT "Storage End"
400 '
410 WBYTE &H5F;                           'UN TALK
420 IRESET REN
430 END
```

(Sample 2) Perform START processing. If trigger is not detected, perform STOP processing.

[RS-232C]

(Explanation)

Lines 150-170 ... Run START processing after clearing status byte.

Lines 190-210 ... Read status byte and ensure start of START processing.

Lines 230-270 ... Continue to check whether or not trigger is detected during a set duration of time.

Lines 280-300 ... If trigger not detected, perform STOP processing.

```
100 '-------------------------------------------'
110 '     8832  SMPLE PROGRAM NO.2                '
120 '-------------------------------------------'
130 '
140 OPEN "COM1:9600,N,7,1" AS #1
150 PRINT #1,"FN1,TD3,SH0"                  'MEM,1ms,20DIV
160 PRINT #1,"TS1,TL60,SL0"                 'TRIG.:CH1,60%,UP
170 PRINT #1,"ST"                           '<START>
180 '
190 FOR W=0 TO 2500:NEXT
200 PRINT #1,"QUS"
210 INPUT #1,A%
220 IF (A% AND &H2)<>0 THEN  200            'START?
230 '
240 FOR W=0 TO 100
250 'FOR F=0 TO 100:NEXT
260 PRINT #1,"QUS"
270 INPUT #1,A%:PRINT NN,A%:NN=NN+1
280 IF (A% AND &H4)<>0 THEN  340            'TRIG WAIT?
290 NEXT W
300 PRINT "NOT TRIGGER"
310  PRINT #1,CHR$(&H1B)"C"                 'STOP
320 GOTO 380
330 '
340 IF (A% AND &H2)=0 THEN  290             'END?
350 PRINT "STRAGE END"
360  PRINT #1,CHR$(&H1B)"C"                 'STOP
370 '
380 CLOSE #1
390 END
```

(Sample 3) Display I/O level of analog unit on CRT for monitoring.

[GP-IB]

(Explanation)

Lines 1190-1340 ... Read existence and category type of CH1 - CH4 unit, and set results as variables.

Lines 1420-1570 ... Read data to become starting point only for analog unit.

Lines 1590-1840 ... Read and plot input level until right margin of screen is reached.

```
1000 ' ---------------------------------- '
1010 '   8832      SAMPLE PROGRAM NO.3     '
1020 ' ---------------------------------- '
1030 '
1040 ADR=5                                  'GP-IB Address=5
1050 SP=2                                   'Set CONST
1060 ISET IFC                               'Interface Clear
1070 ISET REN                               'Remote Enable
1080 ON STOP GOSUB *ENDP : STOP ON          'STOP key ON
1090 SCREEN 3,0:CONSOLE 0,25,0,1:CLS 3      'Clear Display
1100 PRINT@ ADR;"GH0GD0"                    'Header OFF
1110 '
1120 LOCATE 5,0:PRINT "< Level Monitor >"
1130 LOCATE 50,0:PRINT "CH1 :          CH2 :          ";
1140 LOCATE 50,1:PRINT "CH3 :          CH4 :          ";
1150 LOCATE 0,3:PRINT "100"
1160 LOCATE 1,11:PRINT "50"
1170 LOCATE 2,19:PRINT "0"
1180 '
1190 PRINT@ ADR;"QAM1"
1200 INPUT@ ADR;CH1
1210 IF CH1=14 THEN LOCATE 55,0:PRINT "Logic"
1220 IF CH1=15 THEN LOCATE 55,0:PRINT "Nothing"
1230 PRINT@ ADR;"QAM2"
1240 INPUT@ ADR;CH2
1250 IF CH2=14 THEN LOCATE 70,0:PRINT "Logic"
1260 IF CH2=15 THEN LOCATE 70,0:PRINT "Nothing"
1270 PRINT@ ADR;"QAM3"
1280 INPUT@ ADR;CH3
1290 IF CH3=13 THEN LOCATE 55,1:PRINT "Logic"
1300 IF CH3=15 THEN LOCATE 55,1:PRINT "Nothing"
1310 PRINT@ ADR;"QAM4"
1320 INPUT@ ADR;CH4
1330 IF CH4=14 THEN LOCATE 70,1:PRINT "Logic"
1340 IF CH4=15 THEN LOCATE 70,1:PRINT "Nothing"
1350 '
1360 CLS 2
1370 LINE (30,57)-(620,307),7,B,&HCCCC     'Frame
1380 FOR Y=82 TO 282 STEP 25
1390   LINE(30,Y)-(620,Y),7,,&H1010
1400 NEXT Y
1410 '
1420 IF CH1<>9 THEN 1460                    'CH1 not Analog ?
1430 LINE(440,8)-(490,10),6,B
1440 PRINT@ ADR;"QRA1"
1450 INPUT@ ADR;Y10
```

```
1460 IF CH2<>9 THEN 1500                    'CH2 not Analog ?
1470 LINE(560,8)-(610,10),5,B
1480 PRINT@ ADR;"QRA2"
1490 INPUT@ ADR;Y20
1500 IF CH3<>9 THEN 1540                    'CH3 not Analog ?
1510 LINE(440,24)-(490,26),4,B
1520 PRINT@ ADR;"QRA3"
1530 INPUT@ ADR;Y30
1540 IF CH4<>9 THEN 1580                    'CH4 not Analog ?
1550 LINE(560,24)-(610,26),3,B
1560 PRINT@ ADR;"QRA4"
1570 INPUT@ ADR;Y40
1580 '
1590 FOR X=30 TO 620-SP STEP SP
1600 *CH1
1610  IF CH1<>9 THEN *CH2
1620 PRINT@ ADR;"QRA1"
1630 INPUT@ ADR;Y11
1640 LINE(X,307-Y10)-(X+SP,307-Y11),6       'CH1 Line
1650 Y10=Y11
1660 *CH2
1670 IF CH2<>9 THEN *CH3
1680 PRINT@ ADR;"QRA2"
1690 INPUT@ ADR;Y21
1700 LINE(X,307-Y20)-(X+SP,307-Y21),5       'CH2 Line
1710 Y20=Y21
1720 *CH3
1730 IF CH3<>9 THEN *CH4
1740 PRINT@ ADR;"QRA3"
1750 INPUT@ ADR;Y31
1760 LINE(X,307-Y30)-(X+SP,307-Y31),4       'CH3 Line
1770 Y30=Y31
1780 *CH4
1790  IF CH4<>9 THEN 1840
1800 PRINT@ ADR;"QRA4"
1810 INPUT@ ADR;Y41
1820 LINE(X,307-Y40)-(X+SP,307-Y41),3       'CH4 Line
1830 Y40=Y41
1840 NEXT X
1850 GOTO 1360
1860 '
1870 *ENDP
1880 WBYTE &H5F;
1890 IRESET REN
1900 STOP OFF
1910 CLS 3
1920 END
```

(Sample 3) Display I/O level of analog unit on CRT for monitoring.

[RS-232C]

(Explanation)

Lines 1190-1340 ... Read existence and category type of CH1 - CH4 unit, and set results as variables.

Lines 1420-1570 ... Read data to become starting point only for analog unit.

Lines 1590-1840 ... Read and plot input level until right margin of screen is reached.

```
1000 '-------------------------------------'
1010 '    8832  SAMPLE PRPOGRAM  NO.3       '
1020 '-------------------------------------'
1030 '
1040 SP=2                                          'SET CONST
1050 OPEN "COM1:9600,N,7,1" AS #1
1060 PRINT #1,CHR$(27);"C"
1070 SCREEN 9:COLOR 15,1
1080 CLS                  'CLEAR DISPLAY
1090 '
1100 PRINT #1,"GH0"                                'HEADER OFF
1110 '
1120 LOCATE 1,6:PRINT "< LEVEL MONITOR >"
1130 LOCATE 1,51 :PRINT "CH1 :           CH2 :";
1140 LOCATE 2,51 :PRINT "CH3 :           CH4 :";
1150 LOCATE 4,1  :PRINT "100"
1160 LOCATE 13,1 :PRINT " 50"
1170 LOCATE 22,1 :PRINT "  0"
1180 '
1190 PRINT #1,"QAM1"
1200 INPUT #1,CH1%
1210 IF CH1%=14 THEN LOCATE 1,56:PRINT "LOGIC"
1220 IF CH1%=15 THEN LOCATE 1,56:PRINT "NOTHING"
1230 PRINT #1,"QAM2"
1240 INPUT #1,CH2%
1250 IF CH2%=14 THEN LOCATE 1,71:PRINT "LOGIC  "
1260 IF CH2%=15 THEN LOCATE 1,71:PRINT "NOTHING"
1270 PRINT #1,"QAM3"
1280 INPUT #1,CH3%
1290 IF CH3%=14 THEN LOCATE 2,56:PRINT "LOGIC"
1300 IF CH3%=15 THEN LOCATE 2,56:PRINT "NOTHING"
1310 PRINT #1,"QAM4"
1320 INPUT #1,CH4%
1330 IF CH4%=14 THEN LOCATE 2,71:PRINT "LOGIC  "
1340 IF CH4%=15 THEN LOCATE 2,71:PRINT "NOTHING"
1350 '
1360 LINE (30,57)-(620,307),3,B       'FRAME
1370 FOR Y=82 TO 282 STEP 25
1380  LINE(30,Y)-(620,Y),3,,&H1010
1390 NEXT Y
1400 '
1410 IF CH1%<>9 THEN 1450                          'CH1 NOT ANALOG ?
1420 LINE(440,5)-(490,7),15,BF
1430 PRINT #1,"QRA1"
1440 INPUT #1,Y10%
1450 IF CH2%<>9 THEN 1490                          'CH2. NOT ANALOG ?
```

```
1460 LINE(560,5)-(610,7),10,BF
1470 PRINT #1,"QRA2"
1480 INPUT #1,Y20%
1490 IF CH3%<>9 THEN 1530                    'CH3 NOT ANALOG ?
1500 LINE(440,19)-(490,21),11,BF
1510 PRINT #1,"QRA3"
1520 INPUT #1,Y30%
1530 IF CH4%<>9 THEN 1570                    'CH4 NOT ANALOG ?
1540 LINE(560,19)-(610,21),12,BF
1550 PRINT #1,"QRA4"
1560 INPUT #1,Y40%
1570 '
1580 FOR X=30 TO 620-SP STEP SP
1590 'CH1
1610  IF CH1%<>9 THEN 1670
1600  IF CH1%<>9 THEN 1660
1610 PRINT #1,"QRA1"
1620 INPUT #1,Y11%
1630 LINE(X,307-Y10%)-(X+SP,307-Y11%),15     'CH1 LINE
1640 Y10%=Y11%
1650 'CH2
1660 IF CH2%<>9 THEN 1720
1670 PRINT #1,"QRA2"
1680 INPUT #1,Y21%
1690 LINE(X,307-Y20%)-(X+SP,307-Y21%),10     'CH2 LINE
1700 Y20%=Y21%
1710 'CH3
1720 IF CH3%<>9 THEN 1780
1730 PRINT #1,"QRA3"
1740 INPUT #1,Y31%
1750 LINE(X,307-Y30%)-(X+SP,307-Y31%),11     'CH3 LINE
1760 Y30%=Y31%
1770 'CH4
1780 IF CH4%<>9 THEN 1830
1790 PRINT #1,"QRA4"
1800 INPUT #1,Y41%
1810 LINE(X,307-Y40%)-(X+SP,307-Y41%),12     'CH4 LINE
1820 Y40%=Y41%
1830 IF INKEY$<>"" THEN 1870
1840 NEXT X
1850 CLS:GOTO 1120
1860 '
1870 CLOSE #1
1880 END
```

(Sample 4) Save stored data to Disk #1 (sequential file)

[GP-IB]

(Explanation)

Lines 180-190 ... Set destinations for jump when STOP key is pressed or when an error occurs, so system does not end with file open.

Lines 300-320 ... Input channel and file name to be saved.

Lines 400 ... Record data number to be saved into start of file.

Lines 410-450 ... Read stored data from 8832, and save the results sequentially.

```
100 ' ---------------------------------- '
110 '   8832      SAMPLE PROGRAM NO.4     '
120 ' ---------------------------------- '
130 '
140 ADR=5                                  'GP-IB Address=5
150 DR$="1:"                               'FDD No.1
160 ISET IFC                               'Interface Clear
170 ISET REN                               'Remote Enable
180 ON ERROR GOTO *EXIT0                   'If ERROR Then *EXIT0
190 ON STOP GOSUB *EXIT1 : STOP ON         'If STOP  Then *EXIT1
200 '
210 CLS 3:LOCATE 3,3
220 PRINT "< Storage Data SAVE >"
230 PRINT :PRINT
240 PRINT@ ADR;"GH0,QMX"                   'Header OFF
250 INPUT@ ADR;MX                          'Read Max Point
260 IF MX<>0 THEN 290                      'Output OK ?
270   PRINT "No Strage Data !!"
280   GOTO *EXIT2
290 '
300 PRINT "      Max Point=";MX : PRINT
310 INPUT "   Channel(1-4)";CH             'Input Channel
320 INPUT "      File Name";NA$            'Input File Name
330 PRINT :PRINT
340 '
350 DD$=DR$+NA$
360 OPEN DD$ FOR OUTPUT AS #1              'File Open
370 '
380 PRINT@ ADR;"OD"+STR$(CH)+",0"          'Set Output Point
390 '
400 PRINT #1,MX                            'Save Max Point
410 FOR I=0 TO MX
420   PRINT@ ADR;"QDA1"
430   INPUT@ ADR;DT
440   PRINT #1,DT                          'Save Data
450 NEXT I
460 PRINT "     Complete."
470 GOTO *EXIT2
480 '
490 *EXIT0
500 PRINT "         ERROR !!" : GOTO *EXIT2
510 *EXIT1
520 PRINT "         STOP !!"
530 *EXIT2
540 CLOSE #1                               'File Close
550 WBYTE &H5F;                            'UN TALK
560 IRESET REN
570 END
```

(Sample 4) Save stored data to Disk #1 (sequential file)

[RS-232C]

(Explanation)

Lines 160-170 ... Set destinations for jump when STOP key is pressed or when an error occurs, so system does not end with file open.

Lines 280-300 ... Input channel and file name to be saved.

Lines 380 ... Record data number to be saved into start of file.

Lines 390-420 ... Read stored data from 8832, and save the results sequentially.

```
100 '-----------------------------------------'
110 '   8832  SAMPLE PROGLAM   NO.4            '
120 '-----------------------------------------'
130 '
140 OPEN "COM1:9600,N,7,1" AS #1          'FILE OPEN
150 DR$=""                                'FDD NO.1
160 ON ERROR GOTO 470                     'IF ERROR THEN *EXIT0
180 '
190 CLS:LOCATE 3,3
200 PRINT "< STORAGE DATA SAVE >"
210 PRINT:PRINT
220 PRINT #1,"GH0,QMX"                    'HEADER OFF
230 INPUT #1,A% :PRINT "MX=";A%           'READ MAX POINT
240 IF A%<>0 THEN 270                     'OUTPUT OK?
250   PRINT "NO STRAGE DATA"
260   GOTO 510
270 '
280 PRINT "  MAX POINT=";A%  :PRINT
290 INPUT "CHANNEL(1-4)";CH               'INPUT CHANNEL
300 INPUT "  FILENAME? ";NA$              'INPUT FILE NAME
310 PRINT:PRINT
320 '
330 DD$=DR$+NA$
340 OPEN DD$ FOR OUTPUT AS #2             'FILE(FDD) OPEN
350 '
360 PRINT #1,"OD"+STR$(CH)+",0"           'SET OUTPUT POINT
370 '
380 PRINT #2,A% :PRINT "A%=";A%           'SAVE MAX POINT
390 FOR I=0 TO A%
400   PRINT #1,"QDA1"
410   INPUT #1,DT$
420   PRINT #2,DT$                        'SAVE DATA
430 NEXT I
440 PRINT "     COMPLETE."
450 GOTO 510
460 '
470 '  *EXIT0
480  PRINT "    ERROR !!": GOTO 510
490 '  *EXIT1
500 PRINT "    STOP  !!"
510 '  *EXIT2
520 CLOSE #2                              'FILE(FDD) CLOSE
530 CLOSE #1  :ON ERROR GOTO 0            'FILE CLOSE
540 END
```

(Sample 5) Fetch data which was saved in (SAMPLE 4) and set to 8832.

[GP-IB]

(Explanation)

Lines 240-250 ... Specify channel and file.

Lines 320-400 ... Load data and set to 8832.

```
100 ' ---------------------------------- '
110 '   8832      SAMPLE PROGRAM NO.5     '
120 ' ---------------------------------- '
130 '
140 ADR=5                                  'GP-IB Address=5
150 DR$="2:"                               'FDD No.2
160 ISET IFC                               'Interface Clear
170 ISET REN                               'Remote Enable
180 ON ERROR GOTO *EXIT0                   'If ERROR Then *EXIT0
190 ON STOP GOSUB *EXIT1 : STOP ON         'If STOP  Then *EXIT1
200 '
210 CLS 3:LOCATE 3,3
220 PRINT "< Storage Data LOAD >"
230 PRINT :PRINT
240 INPUT "      File Name";NA$            'Input File Name
250 INPUT "   Channel(1-4)";CH             'Input Channel
260 '
270 DD$=DR$+NA$
280 OPEN DD$ FOR INPUT AS #1               'File Open
290 '
300 PRINT@ ADR;"OD"+STR$(CH)+",0"          'Set Input Point
310 '
320 INPUT #1,MX                            'Load Max Point
330 PRINT@ ADR;"GH0,QMX"
340 INPUT@ ADR;MM                          'Read Max Point
350 IF MX<>MM THEN *EXIT0
360 '
370 FOR I=0 TO MX
380   INPUT #1,DT                          'Load Data
390   PRINT@ ADR;"DA"+STR$(DT)
400 NEXT I
410 PRINT "     Complete."
420 GOTO *EXIT2
430 '
440 *EXIT0
450 PRINT "        ERROR !!" : GOTO *EXIT2
460 *EXIT1
470 PRINT "        STOP !!"
480 *EXIT2
490 CLOSE #1                               'File Close
500 WBYTE &H5F;                            'UN TALK
510 IRESET REN
520 END
```

(Sample 5) Fetch data which was saved in (SAMPLE 4) and set to 8832.

[RS-232C]

(Explanation)

Lines 220-230 ... Specify channel and file.

Lines 310-390 ... Load data and set to 8832.

```
100 '-----------------------------------------'
110 '   8832  SAMPLE PROGLAM   NO.5            '
120 '-----------------------------------------'
130 '
140 OPEN "COM1:9600,N,7,1" AS #1          'FILE OPEN
150 DR$=""                                'FDD NO.1
160 ON ERROR GOTO 440                     'IF ERROR THEN *EXIT0
170 '
180 '
190 CLS:LOCATE 3,3
200 PRINT "< STORAGE DATA LOAD >"
210 PRINT:PRINT
220 INPUT "  FILENAME? ";NA$              'INPUT FILE NAME
230 INPUT "CHANNEL(1-4)";CH               'INPUT CHANNEL
240 '
250 DD$=DR$+NA$
260 OPEN DD$ FOR INPUT AS #2              'FILE(FDD) OPEN
270 IF EOF(2) THEN 530                    'IF NOFILE THEN *EXIT3
280 '
290 PRINT #1,"OD"+STR$(CH)+",0"           'SET INPUT POINT
300 '
310 INPUT #2,A%  :PRINT "MX(DATA)=";A%    'LOAD MAX POINT
320 PRINT #1,"GH0,QMX"
330 INPUT #1,MX% :PRINT "MX(8832)=";MX%   'READ MAX POINT
340 IF A%<>MX% THEN 450
350 '
360 FOR I=0 TO A%
370   INPUT #2,DT                         'LOAD DATA
380   PRINT #1,"DA"+STR$(DT)
390 '   FOR W=0 TO 100 :NEXT W            'TIMING ADJUST
400 NEXT I
410 PRINT "     COMPLETE."
420 GOTO 490
430 '
440 ' *EXIT0
450 PRINT "    ERROR !!": GOTO 490
460 ' *EXIT1
470 PRINT "    STOP  !!"
480 ' *EXIT2
490 CLOSE #2                              'FILE(FDD) CLOSE
500 CLOSE #1  :ON ERROR GOTO 0            'FILE CLOSE
510 END
520 ' *EXIT3
530 PRINT "    DON'T EXIST SUCH A FILE !! "
540 GOTO 490
```

Appendices
GB-IB and RS-232C
Command List

GP-IB and RS-232C Command List

	Processing	Format
Execution Processing	START processing	ST
	STOP processing [GP] [RS]	 [DCL] [ESC] C
	PRINT processing	PR
	LIST processing	LS
	COPY processing	HC
	FEED processing	FD ℓ
	Auto TIME/DIVE	AT
	GAUGE processing	GA
	Backlight ON/OFF	EBe
	System reset	SR
	Level monitor	LV
Measuring condition setting	Screen mode	MWm QMW
	Screen size	WSw QWS
	Function	FNf QFN
	TIME/DIV	TDt QTD
	Recording length	SH s QSH

	Processing	Format
Measuring condition setting	Compression rate	MGm QMG
	LCD clear	DCd QDC
	Printer output (REC)	POp QPO
	Auto print (MEM) (XY_{MEM})	POp QPO
WAVE screen	Cursor ON	ABc, t QAB
	A cursor position	MAp QMA
	B cursor position	MBp QMB
	A cursor channel	ACc QAC
	B cursor channel	BCc QBC
	Cursor read output	QVO (VOd$)
	Waveform evaluation	WJ
	Area creation	WM
	Channel view	CVc

	Processing	Format
TRIGGR	Trigger mode	TMc QTM
	Trigger source	TSs QTS
	Trigger level	TL ℓ QTL
	Trigger slope	SLs QSL
	Trigger filter	TFf QTF
	Trigger timing	TTt QTT
	Pre-trigger	PTp QPT
	Trigger waitless	TWw QTW
	Timer trigger start time	TA t$ QTA
	Timer trigger stop time	TO t$ QTO
	Timer trigger interval	IV t$ QIV
	Logic trigger	LA n_1, n_2, n_3 QLA

	Processing	Format
Print	DOT, LINE	DL d QDL
	DRAWING	DC c, d QDRc
	Print format	FMf QFM
	Partial print enlargement rate	PMm QPM
System item	START state backup	BTt QBT
	Grid type	GRg QGR
	Auto list	LTℓ QLT
	Auto gage	GG g QGG
	EL backlight auto off	EL e QEL
	Beep sound	BP b QBP
	Channel marker	MK m QMK
	Self-check execution	SFt
	Clock set	RT t$

	Processing	Format
System item	Clock output	QRT (RT t$)
	RAM card auto save	IAt QIA
	Waveform evaluation	WCm, u, d, r, l, s QWC
Interface control	GP-IB delimiter [GP]	GDd
	Header	GHh
	Error number output	QER (ERe)
	Type and model number output	QID (IDi)
	SRQ response masking [GP]	MSm QMS
	Buffer flow control [RS]	XO x QXO
	Initialization [GP] [RS]	[DCL] [ESC] C
	Status byte output	QUS (USs)
	Key lock [RS]	UK u QUK

	Processing	Format
Input unit	Type of input unit	QAMc (AM c, n)
	Range of analog unit	QAAc (AA c, n_1, n_2, n_3, n_4)
IC card	Start IC card monitor	IC
	End IC card monitor	IE
	INIT processing	II
	KILL processing	IKn
	LOAD	LDn
	SAVE file name	NAn$
	IC card test	QIT (ITt)
	SAVE processing	ISt, ch, n
	IC card setting status	QIC (IC, i, p, n)
Realtime data	Realtime data output (ASCII)	QRAc (RAc, d)
	Realtime data output (BINARY) [GP]	QRB c (d$)

	Processing	Format
Stored data I/O	Stored data	QMX (MXn)
	I/O enable information	ODc,p QOD
	I/O point designation	
	Stored data output (ASCII)	QDAn (DA d_1, ,d_2, d_3, ..., d_n)
	Stored data output (BINARY) [GP]	QDBn (d$)
	Stored data input (ASCII)	DA d_1, d_2, ..., d_n
Computation function	Voltage conversion	QVL c,p (VL c, p, d$)
	PEAK value computation	QPK c (PKc, M_1, M_2, $P_1$$, $P_2$$)
	Average value/variance computation	QME c (ME c, m, v)
	Area value computation	QAR c_1, c_2 (AR c_1, c_2, a)
	XY area computation	QFX c (FX c,a)
	Threshold point computation	QCP c, H , s, p_1 (CP c, H , s, p_1, p_2)

	Processing	Format
Computation function	Differential computation	DI c_1, c_2, k
	Integral computation	INc$_1$, c_2, k
	SWAP computation	SW c_1, c_2
Special function	Trigger detection time output	QTJ (TJt$)
	PRINT processing optional comment	COc QCO
	PRINT processing optional comment character	CMm$: QCM
	PRINT processing ch optional comment	CN c QCN
	PRINT processing ch optional comment character	CC c, m$: QCC c
	Scaling	SC c, s QSC c
	Scaling unit	SE c, n$: QSE c
	Scaling conversion value	SV c, n QSV c
	Scaling offset	SO c, n QSO c

Index

8832 Command Index

HIOKI E.E. CORPORATION

81 Koizumi, Ueda, Nagano 386-11, Japan

TEL:0268-28-0562 FAX:0268-28-0568

TLX:3327508 HIOKI J CABLE: HEWLOV, Ueda

8832A983-01 91-02-0010U 78320100 Printed in Japan