SIGNAL RECOVERY

Using the Model 7225 and 7265 Lock-in Amplifiers with software written for the SR830

APPLICATION NOTE
AN 1002

Introduction

The majority of user-developed software programs written to operate the SR830 lock-in amplifier are easily modified to use the ***SIGNAL RECOVERY*** models 7225 and 7265 Digital Signal Processing (DSP) instruments instead. This Application Note describes the changes required by first considering how the behavior of the GPIB and RS232 interfaces differs between the instruments and then listing the most commonly used SR830 commands and their 7225/7265 equivalents.

The ***SIGNAL RECOVERY*** models 7225 and 7265 are both dual phase lock-in amplifiers and share the same command set, except that some of the command parameters have a wider range for the latter unit, to accommodate its increased operating frequency range. There are also a few commands which are exclusive to the 7265. In the rest of this Application Note the term 7265 can be taken to include the 7225, except where otherwise noted.

General

Perhaps the most important difference between the instruments is that in the SR830 the user needs to specify the interface to which the response to a command should be sent, using the **OUTX** command as the first command in a program. There is no equivalent command for the 7265 or 7225 since they always generate a response to the same interface port at which they received a command.

GPIB Interface

Both the SR830 and ***SIGNAL RECOVERY*** 7265 instruments support the IEEE-488.1 (1978) standard, but only the SR830 supports the common standards of the IEEE-488.2 (1987) standard. However this is rarely a problem since the commands associated with the latter are not particularly useful when controlling a specialized instrument such as a lock-in amplifier.

The 7265's GPIB address is set using the GPIB SETTINGS MENU in the range 0 to 31.

The serial poll status byte bit allocations differ between the instruments, as follows:

Bit	SR830 Serial Poll Status Byte	***SIGNAL RECOVERY*** 7265 Serial Poll Status Byte
bit 0	no scan in progress	command complete
bit 1	no command execution in progress	invalid command
bit 2	bit in error status byte has been set	command parameter error
bit 3	bit in LIA status byte has been set	reference unlock
bit 4	data available in output buffer	input or output overload
bit 5	bit in standard status byte has been set	new ADC values available after external trigger
bit 6	asserted SRQ	asserted SRQ
bit 7	not used	data available in output buffer

Hence the following bits are equivalent:-

SR830 Serial Poll Status Byte		***SIGNAL RECOVERY*** 7265 Serial Poll Status Byte
bit 1	no command execution in progress	bit 0 - command complete
bit 4	data available in output buffer	bit 7 - data available
bit 6	asserted SRQ	bit 7 - asserted SRQ

Information indicated by the assertion of the other four functional bits in the SR830's Status Byte can generally be obtained from the 7265's Status Byte (accessed via a serial poll or **ST** command and by using the commands **N** (Overload Byte) and **M** (Monitor Curve Acquisition).

Although both instruments will allow the user to simply send a command over the GPIB and then read the response, the recommended method with the 7265 is to use the serial poll status byte as part of a single write/read routine. This ensures that the response read back from the instrument is that generated by the command sent to it and in addition allows any error conditions which might be present to be immediately identified.

For example, with the 7265 this recommendation translates into the following Visual Basic code element, based on the use of a National Instruments PCI-GPIB interface card:-

```
' send the command
Call ibwrt(devAssign, sndcmd)           ' sends command to lock-in
Call ibrsp(devAssign, SPSB)             ' reads serial poll status byte
cmddone = SPSB And 1                    ' command complete = LSB of SPSB

cmdrsp = "No response to this command"  ' initializes command response string

' next section performs repeated serial polls in a loop, waiting for
' command done to clear, implying that command is being processed. But on slow
' computers and with write-only commands, command done might clear and be
' reasserted without the serial poll detecting it. Hence a timeout loop using
' k as a counter is included.
k = 0                                   ' k is used to implement timeout loop
While k < 1000 And cmddone = 1          ' start loop with timeout
Call ibrsp(devAssign, SPSB)             ' read SPSP
cmddone = SPSB And 1                    ' wait for command done to clear
k = k + 1                               ' increment timeout counter
Wend                                    ' loop

' command done should now be cleared, implying instrument is processing command
' next section performs repeated serial polls in a loop until command done
' is re-asserted. Each time the data available bit is also detected and if
' asserted a GPIB read is performed.
While cmddone = 0
Call ibrsp(devAssign, SPSB)             ' serial poll
cmddone = SPSB And 1                    ' command done = bit 0
dataavail = (SPSB And 128)              ' data available = bit 7
If dataavail = 128 Then                 ' if data is available
cmdrsp = Space$(32)                     ' define buffer into which to read response
Call ibrd(devAssign, cmdrsp)            ' read response
cmdrsp = Left(cmdrsp, ibcnt)            ' ibcnt is a global variable returned by
                                        ' the NI GPIB handler software containing
                                        ' the number of bytes read. Use this to
                                        ' trim response to that number of bytes
End If
Wend                                    ' loop until command done is asserted
Invalidcmd = (SPSB And 2)
Paramerr = (SPSB And 4)
Refunlock = (SPSB And 8)
Overload = (SPSB And 16)
```

The 7265 can be set to accept and generate a Carriage Return character, Carriage Return/Line Feed character pair or a GPIB EIO as input and output terminators for GPIB communications. These three options include the two available on the SR830.

RS232 Interface

Both instruments are fitted with RS232 interfaces, although that on the SR830 is configured as a DCE and has a 25-pin connector, whilst the 7265 is a DTE and uses the more modern 9-pin connector. Consequently the user will need a

different type of RS232 cable to couple the 7265 to his computer; the connection diagram for the most popular combination is given in Figure 1.

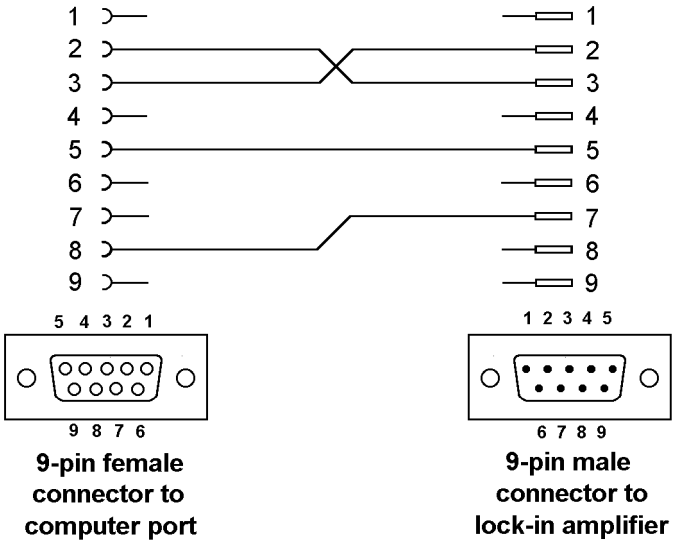


Figure 1, RS232 Cable to Connect 7265 to PC Computer

The other important difference between the instruments' RS232 interfaces is that the 7265 uses a character-by-character software handshake rather than CTS/DTR hardware or DRQ software handshaking. Hence the user's program must send each command one character at a time and wait until the instrument echoes it back before sending the next character.

For example, a Qbasic code segment to achieve this is:

```
'send the command cmd$ - which includes CR terminator - one character at a time.
'Note ; after print statement avoids QBasic sending extra CR or LF characters.
   FOR charcount = 1 TO LEN(cmd$)
   sentchar$ = MID$(cmd$, charcount, 1)
   PRINT #1, sentchar$;

   'check for each character being echoed by the lock-in
   DO
   WHILE LOC(1) = 0: WEND
   recdchar$ = INPUT$(1, #1)
   LOOP UNTIL recdchar$ = sentchar$
   NEXT charcount
```

Like the SR830, the 7265 accepts a Carriage Return or Carriage Return/Line Feed character pair as input terminator and normally generates a Carriage Return/Line Feed character pair as an output terminator. Unlike the SR830, the 7265 also normally sends a prompt character after the terminator, either an asterisk "*" or question mark "?". The former indicates that the instrument is functioning correctly, whilst the latter implies that an error condition is present. The user's program can then issue **ST** (status) and **N** (overload status) commands to determine the source of error.

Multiple Commands and Delimiters

Both instruments will accept multiple commands separated by semi-colons on a single line. Multiple responses are normally separated by a comma delimiter, although the 7265 allows the user to change this to any other printable ASCII character.

Communications Monitor

The 7265 includes a communications monitor display. Unlike the SR830 receive input buffer display, this shows not only the commands received from the computer, but also responses generated by the lock-in amplifier, making program debugging easier. The 7225 does not have such a display.

Command Set

In both instruments, GPIB and RS232 commands are identical as far as possible, and have the general format of a command mnemonic followed by one or more parameters.

In the SR830, the present setting of an instrument control is determined by adding a question mark "?" character after the command mnemonic and omitting any parameter(s), whereas in the 7265 the same effect is achieved by simply omitting the command parameter(s). The instrument then responds with the present value(s) of the requested command's parameter(s). Commands with multiple parameters require commas between parameters in the SR830, but not with the ***SIGNAL RECOVERY*** 7265.

The following table lists in alphabetical order the most commonly used SR830 commands and their 7265 equivalents, together with important notes about the parameter range. Note that [n] indicates that the parameter n is optional. Although the table makes translating programs easier, the reader may need to refer to the instruments' instruction manuals as well.

SR830 Command	***SIGNAL RECOVERY*** 7265 Command	Purpose	Notes
***CLS**	No equivalent command	Clear Status Registers	
***ERRS? [i]**	No equivalent command	Read Error Status Byte	
***ESR? [i]**	No equivalent command	Read Standard Event Status Byte	Information corresponding to bits 4 and 5 of standard event status byte can be obtained from the 7265's response to a serial poll or **ST** command
***IDN?**	**ID**	Read Instrument Identity	7265 returns only "7265", 7225 returns only "7225"
***LIAS? [i]**	No equivalent command	Read LIA Status Byte	Information corresponding to bits 0 to 3 of the LIA status byte can be obtained from 7265's response to a serial poll or **ST** command and /or the response to an **N** command
***RST**	**ADF 1**	Reset all instrument settings to default values	Communications interface settings are not changed
***STB? [i]**	**ST**	Read Serial Poll Status Byte	Note bit allocations differ between the SR830 and 7265 as described earlier in this document
AGAN	**ASEN**	Auto Sensitivity/Gain	**AGAN** is inoperative at time constants longer than 1 second
AOFF i	**AXO**	Auto-Offset	In the SR830, i selects output to be offset. In the 7265 both X and Y are auto-offset
APHS	**AQN**	Auto Phase	Operates once per call
ARSV	**AUTOMATIC 1**	Auto Reserve - AC Gain	**ARSV** operates once per call; Automatic AC Gain on (n = 1) adjusts AC Gain with FS sensitivity in the 7265
AUXV i, x	**DAC. n_1 n_2**	Set Auxiliary DAC Voltages	i = n_1 = 1 to 4 sets DAC 1 to DAC 4 outputs. x and n_2 are in volts, with 1 mV resolution
AUXV? i	**DAC. n_1**	Read Auxiliary DAC Voltages	As above
FMOD i	**IE n**	Set Reference Channel Source	i = 1 and n = 0: internal i = 0: external SR830 n = 1: external TTL 7265 n = 2: external analog 7265

SR830 Command	***SIGNAL RECOVERY*** 7265 Command	Purpose	Notes
FMOD?	**IE**	Read Reference Channel Source	As above
FPOP 1, j	**CH 1 n**	Set CH1 Analog Output Type	j = 1, n = 0: X%, see manuals for other outputs
FPOP? 1	**CH 1**	Read CH1 Analog Output Type	As above
FPOP 2, j	**CH 2 n**	Set CH2 Analog Output Type	j = 1, n = 0: Y%, see manuals for other outputs
FPOP? 2	**CH 2**	Read CH2 Analog Output Type	As above
FREQ f	**OF. n**	Set Internal Reference (Oscillator) frequency	f and n are in hertz. Ranges are limited as follows:- SR830: 1 mHz to 102 kHz 7225: 1 mHz to 120 kHz 7265: 1 mHz to 250 kHz
FREQ?	**FRQ.**	Read Reference Frequency	As above. Note that in the 7265 the oscillator frequency is not necessarily the same as the reference frequency.
HARM i	**REFN n**	Set Reference Harmonic Control	i and n indicate reference harmonic number, with range limited as follows:- SR830: 1 to 19999 7225: 1 to 32 7265: 1 to 65535
HARM?	**REFN**	Read Reference Harmonic Control	As above
ICPL i	**CP n**	Set Input Coupling mode	i = n = 0 : AC i = n = 1: DC
ICPL?	**CP**	Read Input Coupling mode	As above
IGND i	**FLOAT n**	Set Input Connector Float/Ground Control	i = 0 and n = 1: Float i = 1 and n = 0: Ground
IGND?	**FLOAT**	Read Input Connector Float/Ground Control	As above
ILIN i	**LF n_1 n_2**	Set Line Frequency notch filter	i n_1 Function 0 0 No filters 1 1 F 2 2 2F 3 3 F & 2F n_2 sets the filter frequency in the 7225/7265 n_2 = 0: 60 Hz, n_2 = 1: 50 Hz
ILIN?	**LF**	Read Line Frequency notch filter	As above
ISRC 0	**IMODE 0; VMODE 1**	Set Input Mode to A input, voltage mode	
ISRC 1	**IMODE 0; VMODE 3**	Set Input Mode to A-B input, voltage mode	
ISRC 2	**IMODE 1**	Set Input Mode to current - high bandwidth	
ISRC 3	**IMODE 2**	Set Input Mode to current - low noise	
ISRC?	**IMODE; VMODE**	Read Input Mode	As above

SR830 Command	***SIGNAL RECOVERY*** **7265 Command**	Purpose	Notes
LOCL i	**REMOTE n**	Set Local/Remote status	i = n = 0: Front Panel Active i = n = 1: Remote Operation Only
LOCL?	**REMOTE**	Read Local/Remote status	As above
OAUX? i	**ADC. n**	Read Auxiliary ADC Voltages	i = 1 to 4, n = 1 or 2: Response is in volts, with 1 mV resolution.
OEXP 1 [, j, k]	**XOF 1 n_2; EX n_3**	Set X Output Offset and Expand	j = n_3 =0: No expansion j = n_3 = 1: ×10 Expansion k = n_2/100 = Percentage offset
OEXP? 1	**XOF 1; EX**	Read X Output Offset and Expand	As above
OEXP 2 [, j, k]	**YOF 1 n_2; EX n_3**	Set Y Output Offset and Expand	j = n_3 =0: No expansion j = n_3 = 1: ×10 Expansion k = n_2/100 = Percentage offset
OEXP? 2	**YOF 1; EX**	Read Y Output Offset and Expand	As above
OFLT i	**TC n**	Set Output Filters Time Constant	The available SR830 time constants are in a 1-3-10 sequence from 10 µs (i = 0)to 30 ks (i = 19), whilst the ***SIGNAL RECOVERY*** 7265 range from 10 µs (n = 0) to 100 ks in a 1-2-5-10 sequence.
OFLT?	**TC**	Read Output Filters Time Constant	As above
OFSL i	**SLOPE n**	Set Output Filters Slope	n or i — Selection 0 — 6 dB/octave 1 — 12 dB/octave 2 — 18 dB/octave 3 — 24 dB/octave
OFSL?	**SLOPE**	Read Output Filters Slope	As above
OUTP? 1	**X.**	Read X Output Value	Response is in volts
OUTP? 2	**Y.**	Read Y Output Value	Response is in volts
OUTP? 3	**MAG.**	Read Magnitude (R) Value	Response is in volts
OUTP? 4	**PHA.**	Read Signal Phase (θ) Value	Response is in degrees
OUTX [?] [i]	No equivalent command	Select Output Interface	The 7265 always responds to the port at which it received a command.
OVRM [?] i	No equivalent command	Set/Read Front Panel Lockout	Use 7265's **REMOTE** command
PHAS x	**REFP. n**	Set Reference Phase	x and n are in degrees
PHAS?	**REFP.**	Read Reference Phase	Response is in degrees
PAUS	**HC**	Pause data acquisition to buffer	
REST	**NC**	Reset Data Buffers	Commands erase data buffers
RMOD i	**ACGAIN n**	Set Dynamic Reserve or AC GAIN	i is limited to 0, 1 or 2. n ranges from 0 to 9
RMOD?	**ACGAIN**	Read Dynamic Reserve or AC GAIN	As above
RSET i	No equivalent command	Recall setup from buffer i	Use series of instrument commands with required settings instead

SR830 Command	***SIGNAL RECOVERY*** **7265 Command**	Purpose	Notes
RSLP[?]	No equivalent command	Set/Read External Reference Trigger Threshold	Use 7265 **IE n** and appropriate reference connector
SEND 0;STRT	**TD**	Start data acquisition to buffer - one-shot	In the SR830, acquisition may not start until trigger is received
SEND 1;STRT	**TDC**	Start data acquisition to buffer - loop mode	In the SR830, acquisition may not start until trigger is received
SENS i	**SEN n**	Set full-scale sensitivity	When the 7265 and SR830 are in voltage input mode or wide bandwidth current mode (**IMODE 1**), then n = i +1. Hence for a full scale sensitivity of 100 nV, i = 5 and n = 6. When the 7256 is in low noise current mode (**IMODE 2**), then n = i + 6. Hence for full-scale sensitivity of 10 nA, i = 21 and n = 27
SENS?	**SEN**	Read full-scale sensitivity	As above
SLVL x	**OA. n**	Set Oscillator Amplitude	x and n are in volts rms, with range limited as follows:- SR830: 0.004 to 5.000 7225: 0.001 to 5.000 7265: 0.000001 to 5.000 (accuracy is not specified for <100 µV
SLVL?	**OA.**	Read Oscillator Amplitude	Response is in volts rms
SNAP i [,j,k,l,m,n]	**STAR n** followed by ***** commands	Read Output(s) sampled at same time	Use **STAR n** to select 7265 outputs equivalent to those specified by parameters i to n to be read and then use ***** commands to read data
SPTS?	**M**	Read Number of points in buffer	The **M** command returns four values; the number of points stored is the fourth response
SRAT i	**STR n**	Set Curve Buffer Storage Rate Control	i sets time per point according to table in range 1.9 ms to 16 s per point. n sets time per point in milliseconds in range 1.25 ms (n = 0) to 1E6.
SRAT?	**STR**	Read Curve Buffer Storage Rate Control	As above
SSET i	No equivalent command	Save setup to buffer i	
STRT	**TD**	Start data acquisition to buffer	
SYNC i	**SYNC n**	Set Synchronous Time Constant	n = i = 0: Sync filter off n = i = 1: Sync filter on
SYNC?	**SYNC**	Read Synchronous Time Constant	As above
TRIG	No equivalent command	Software Trigger	
TRCA? i, j ,k	**DC. n**	Read points stored in buffer i/n	Response is in ASCII floating-point numbers
TSTR 1; STRT	**TDT 0**	Start curve acquisition on-trigger	

Example - Basic Signal Recovery

This section shows how a typical program coded to operated the SR830 needs to be modified to operate the 7265. The program, which is listed simply as the required sequence of commands, excluding details of the write/read subroutine to send them, sets the lock-in amplifier's controls and then records the chosen outputs, perhaps as a function of time.

Assuming the output sampling rate is less than a few points per second then there is no need to use the internal curve buffers. The commands to achieve this would therefore be similar to the following sequence:

SR830	***SIGNAL RECOVERY* 7265**	**Notes**
OUTX 1	No equivalent command	Set GPIB as the interface to which responses are sent
FMOD 0	IE 2	Set reference source to external front panel input
ISRC 0	IMODE 0; VMODE 1	Single-ended voltage input mode
IGND 0FLOAT 1		Float input connector shell with respect to chassis ground
ILIN 0	LF 0 0	Turn off line frequency rejection filter
OFLT 8TC 11		Set time constant to 100 ms
AGAN	AS	Auto-Sensitivity/Gain
APHS	AQN	Auto Phase

Then the outputs could be read as follows:

OUTP? 1	X.	Reads X output in volts
OUTP? 2	Y.	Reads Y output in volts
OUTP? 3	MAG.	Reads Magnitude in volts
OUTP? 4	PHA.	Reads Phase in degrees
FREQ?	FRQ.	Reads reference frequency in hertz

The controlling program would send a new output command each time a new reading were required. Note that at an output filter slope of 12 dB/octave a good “rule of thumb” is to wait for a period of five time-constants after the input signal has changed before recording a new value. Hence when recording the measured signal as an experimental parameter is changed, the program should issue the commands to whatever equipment causes this change, wait for five time-constants, and then record the required output.

Conclusions

In the majority of cases, programs written to control the SR830 lock-in amplifier use only a small number of the available instrument commands. In such cases, modifying the program to operate the ***SIGNAL RECOVERY*** 7225 or 7265 instruments is not nearly as large a task as might at first be thought.

Changing to the ***SIGNAL RECOVERY*** units allows the user to take advantage, maybe at a later date, of the richer feature set of these instruments, including such items as the extended frequency range, dual reference and harmonic modes, the transient recorder facility and the more powerful output data curve buffer.

SIGNAL RECOVERY

SIGNAL RECOVERY is part of AMETEK Advanced Measurement Technology, Inc

801 SOUTH ILLINOIS AVENUE
OAK RIDGE
TN 37831-2011
USA
Phone: +1 865 482 4411
Fax: +1 865 483 0396

SPECTRUM HOUSE
1 MILLARS BUSINESS CENTRE, FISHPONDS CLOSE
WOKINGHAM, BERKS RG41 2TZ
UNITED KINGDOM
Phone: +44 (0)118 936 1210
Fax: +44 (0)118 936 1211

E-mail: info@signalrecovery.com **Web Site: www.signalrecovery.com**

V5.01/08UK © 2008 AMETEK Advanced Measurement Technology, Inc