Model 7210

Multi-Channel
DSP Lock-in Amplifier

Instruction Manual
190480-A-MNL-F

Acknowledgement

This instrument was originally developed in conjunction with:

Dr E. Tokunaga

Masumoto Single Quantum Dot Project
Japan Science and Technology Corporation
Tsukuba Research Consortium
5-9-9 Tokodai
Tsukuba
Ibaraki 300-2635
Japan

FCC Notice

This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with this manual, may cause interference to radio communications. As temporarily permitted by regulation, operation of this equipment in a residential area is likely to cause interference, in which case the user at his own facility will be required to take whatever measures may be required to correct the interference.

Company Names

SIGNAL RECOVERY is part of Advanced Measurement Technology, Inc, a division of AMETEK, Inc. It includes the businesses formerly trading as EG&G Princeton Applied Research, EG&G Instruments (Signal Recovery), EG&G Signal Recovery and PerkinElmer Instruments (Signal Recovery)

Declaration of Conformity

This product conforms to EC Directives 89/336/EEC Electromagnetic Compatibility Directive, amended by 92/31/EEC and 93/68/EEC, and Low Voltage Directive 73/23/EEC amended by 93/68/EEC.

This product has been designed in conformance with the following IEC/EN standards:

EMC: BS EN55011 (1991) Group 1, Class A (CSPIR 11:1990)
BS EN50082-1 (1992):
IEC 801-2:1991
IEC 801-3:1994
IEC 801-4:1988

Safety: BS EN61010-1: 1993 (IEC 1010-1:1990+A1:1992)

Trademarks

Table of Contents

Chapter One, Introduction

Chapter Two, Installation & Initial Checks

Chapter Three, Technical Description

Chapter Four, Front and Rear Panels

Chapter Five, Computer Operation

Appendix A, Specifications

Appendix B, Pinouts

Appendix C, Demonstration Programs

Appendix D, Cable Diagrams

Appendix E, Alphabetical Listing of Commands

Index

Introduction

Chapter 1

1.1 How to Use This Manual

This manual gives detailed instructions for setting up and operating the SIGNAL RECOVERY Model 7210 Multi-Channel Digital Signal Processing (DSP) dual phase lock-in amplifier. It is split into the following chapters:-

Chapter 1 - Introduction
Provides an introduction to the manual, briefly describes what a lock-in amplifier is and the types of measurements it may be used for, and lists the major specifications of the model 7210.

Chapter 2 - Installation and Initial Checks
Describes how to install the instrument and gives a simple test procedure which may be used to check that the unit has arrived in full working order.

Chapter 3 - Technical Description
Provides an outline description of the design of the instrument and discusses the effect of the various controls. A good understanding of the design will enable the user to get the best possible performance from the unit.

Chapter 4 - Front and Rear Panels
Describes the connectors, controls and indicators which are found on the unit and which are referred to in the subsequent chapters.

Chapter 5 - Computer Operation
This chapter provides detailed information on operating the instrument from a computer over the GPIB (IEEE-488) or RS232 interfaces. It includes information on how to establish communications, the functions available, the command syntax and a detailed command listing.

Appendix A
Gives the detailed specifications of the model 7210.

Appendix B
Details the pinouts of the multi-way connectors on the rear panel.

Appendix C
Lists three simple terminal programs which may be used as the basis for more complex user-written programs.

Appendix D
Shows the connection diagrams for suitable RS232 null-modem cables to couple the unit to an IBM-PC or 100% compatible computer.

Appendix E
Gives an alphabetical listing of the computer commands for easy reference.

New users are recommended to unpack the instrument and carry out the procedure in chapter 2 to check that it is working satisfactorily. They should then make themselves familiar with the information in chapters 3 and 4 before turning to chapter 5 for information on how to operate the instrument. Once the structure of the computer commands is familiar, appendix E will prove convenient as it provides a complete alphabetical listing of these commands in a single easy-to-use section.

1.2 What is a Lock-in Amplifier?

In its most basic form the single-channel lock-in amplifier is an instrument with dual capability. It can recover signals in the presence of an overwhelming noise background or alternatively it can provide high resolution measurements of relatively clean signals over several orders of magnitude and frequency.

The model 7210 extends this capability to experiments requiring several detection channels, where previously the only way of implementing them has been with multiple single-channel instruments. It therefore offers a cost-effective and physically much more compact solution to such requirements.

The model 7210 lock-in amplifier can function as a:-

- AC Signal Recovery Instrument
- Vector Voltmeter
- Phase Meter
- Frequency Meter

These characteristics, all available in a single compact unit, make it an invaluable addition to any laboratory.

1.3 Key Specifications and Benefits

The SIGNAL RECOVERY Model 7210 represents a significant advance in the application of DSP technology in the design of a lock-in amplifier. Until now, units have been restricted to a single signal channel, allowing only one, or at most two, signals to be measured at any one time. The model 7210, with its use of the latest technology, allows up to 32 signals to be measured simultaneously. What is more, units can be linked together to give more detection channels. For example, four units would give 128 channels, while sixteen would give 512 channels.

The unit can effectively operate as 32 parallel dual-phase lock-in amplifiers, running at the same external reference frequency, measuring 32 signals and generating 32 pairs of X and Y outputs. It can also operate in a *tandem* mode in which it generates a second reference signal which is an integer division of the external reference. This second reference is applied to the external experiment in such a way as to induce an amplitude modulation on the signal at the first reference frequency.

The amplitude modulation is detected by the first set of demodulators, which run at the external frequency, and then further demodulated by a second set of demodulators running at the generated reference frequency, to give a second set of X and Y outputs per channel. This detection method would previously have required two lock-in amplifiers connected in series, so in this mode the 32-channels of the 7210 are equivalent to 64 dual phase lock-in amplifiers. To date, no other commercially available instrument matches this capability.

Key specifications include:

- 32 Dual-Phase Detection Channels
- Up to sixteen instruments can be interconnected to give 512 detection channels
- External Frequency range: 20 Hz to 50.5 kHz
- Voltage, Wide Bandwidth Current or Low Noise Current input modes, depending on input signal board configuration
- Dual phase demodulators with 5-digit readings of X and Y outputs
- Very low phase noise of $< 0.0001^{\circ}$ rms
- Output time constant: 2 ms to 1 ks
- Tandem Reference Mode - Unit generates a second reference frequency by integer division of the first reference frequency. This second frequency is used in the experiment to amplitude modulate the signal at the first frequency. The instrument then uses a second stage of demodulation to detect the resulting modulation at the X output of the first stage demodulation
- 8-bit programmable digital output port for control of peripheral devices
- Full range of auto-modes
- Internal data buffer capable of storing up to 4000 sets of 32-output readings at intervals of down to 2 ms per set
- Data buffer can be used in first-in, first out (FIFO) mode for continuous data acquisition

Installation & Initial Checks

Chapter 2

2.1 Installation

2.1.01 Introduction

Installation of the model 7210 in the laboratory or on the production line is very simple. It can be operated on almost any laboratory bench or be rack mounted using the integral flanges at the user's convenience. With an ambient operating temperature range of 0 °C to 30 °C, it is highly tolerant to environmental variables, needing only to be protected from exposure to dust, corrosive agents and liquids.

The instrument uses forced-air ventilation and as such should be located so that the ventilation holes on the bottom and/or top cover and rear panels are not obstructed. This condition is best satisfied by leaving a space of at least 2" (5 cm) between the top/bottom covers and rear panels and any adjacent surface

2.1.02 Inspection

Upon receipt the model 7210 Lock-in Amplifier should be inspected for shipping damage. If any is noted, SIGNAL RECOVERY should be notified immediately and a claim filed with the carrier. The shipping container should be saved for inspection by the carrier.

2.1.03 Line Cord Plug

A standard IEC 320 socket is mounted on the rear panel of the instrument and a suitable line cord is supplied.

2.1.04 Line Voltage Selection and Line Fuses

Before plugging in the line cord, ensure that the model 7210 is set to the voltage of the AC power supply to be used.

A detailed discussion of how to check and, if necessary, change the line voltage setting follows.

CAUTION: The model 7210 may be damaged if the line voltage is set for 110 V AC operation and it is turned on with 220 V AC applied to the power input connector.

The model 7210 can operate from any one of four different line voltage ranges, 90-110 V, 110-130 V, 200-240 V, and 220-260 V, at 50-60 Hz. The change from one range to another is made by repositioning a plug-in barrel selector internal to the Line Input Assembly on the rear panel of the unit. Instruments are normally shipped from the factory with the line voltage selector set to 110-130 V AC, unless they are destined for an area known to use a line voltage in the 220-260 V range, in which case, they are shipped configured for operation from the higher range.
The line voltage setting can be seen through a small rectangular window in the line input assembly on the rear panel of the instrument (figure 2-1). If the number

showing is incorrect for the prevailing line voltage (refer to table 2-1), then the barrel selector will need to be repositioned as follows.

Observing the instrument from the rear, note the plastic door immediately adjacent to the line cord connector (figure 2-1) on the right-hand side of the instrument. When the line cord is removed from the rear-panel connector, the plastic door can be opened outwards by placing a small, flat-bladed screwdriver in the slot on the topside and levering gently. This gives access to the fuse and to the voltage barrel selector, which is located at the right-hand edge of the fuse compartment. Remove the barrel selector with the aid of a small screwdriver or similar tool. With the barrel selector removed, four numbers become visible on it: 100, 120, 220, and 240, only one of which is visible when the door is closed. Table 2-1 indicates the actual line voltage range represented by each number. Position the barrel selector such that the required number (see table 2-1) will be visible when the barrel selector is inserted and the door closed.

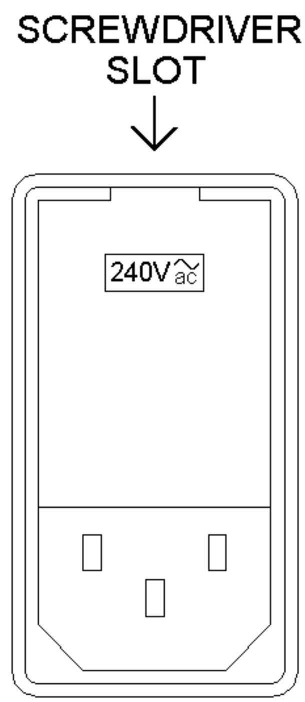

Figure 2-1, Line Input Assembly

VISIBLE #	VOLTAGE RANGE
100	90 - 110 V
120	110 - 130 V
220	200 - 240 V
240	220 - 260 V

Table 2-1, Range vs. Barrel Position

Next check the fuse rating. For operation from a nominal line voltage of 100 V or 120 V, use a 20 mm slow-blow fuse rated at 2.0 A, 250 V. For operation from a nominal line voltage of 220 V or 240 V, use a 20 mm slow-blow fuse rated at 1.0 A, 250 V.

To change the fuse, first remove the fuse holder by pulling the plastic tab marked with an arrow. Remove the fuse and replace with a slow-blow fuse of the correct voltage and current rating. Install the fuse holder by sliding it into place, making sure the arrow on the plastic tab is pointing downwards. When the proper fuse has been installed, close the plastic door firmly. The correct selected voltage setting should now be showing through the rectangular window. Ensure that only fuses with the required current and voltage ratings and of the specified type are used for replacement. The use of makeshift fuses and the short-circuiting of fuse holders is prohibited and potentially dangerous.

2.2 Initial Checks

2.2.01 Introduction

The following procedure checks the performance of a single model 7210. In general, this procedure should be carried out after inspecting the instrument for obvious shipping damage. If several instruments have been supplied already interconnected as a system, then separate instructions apply, but the given procedure is still valid for checking a single unit.

NOTE: Any damage must be reported to the carrier and to SIGNAL RECOVERY ***immediately. In addition the shipping container must be retained for inspection by the carrier.***

Note that this procedure is intended to demonstrate that the instrument has arrived in good working order, not that it meets specifications. Each instrument receives a careful and thorough checkout before leaving the factory, and normally, if no shipping damage has occurred, will perform within the limits of the quoted specifications. If any problems are encountered in carrying out these checks, contact SIGNAL RECOVERY or the nearest authorized representative for assistance.

2.2.02 Procedure

1) Ensure that the model 7210 is set to the line voltage of the power source to be used, as described in section 2.1.05.

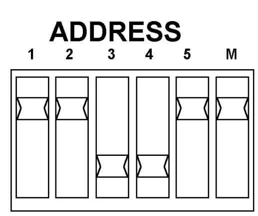

Figure 2-2, Model 7210 GPIB Address and Master/Slave Switch

2) Locate the GPIB Address and Master/Slave switch on the lower left-hand side of the rear-panel, as shown in figure 2-2
3) Confirm that the switches are set as shown in figure 2-2. This configures the

instrument as a Master unit and the GPIB address as 12.

Note:- If several units have been supplied as a system then the switches may be set to different positions. Record the existing settings before changing them to match the above so that you can restore them on completion of this procedure.

4) Connect an RS232 null modem cable, SIGNAL RECOVERY part number C01003, between the **RS232** connector on the rear-panel of the instrument and a serial port on a PC compatible computer running Windows 95 or 98.

5) Start Windows HyperTerminal or other RS232 terminal emulator software. Set the communications port properties in the software, for the port to which the instrument is cabled, to 9600 baud, 7 data bits, 1 stop bit, even parity and no flow control.

6) With the rear-panel mounted power switch (located above the line power input connector) set to **0** (off), plug in the line cord to an appropriate line source.

7) Turn the model 7210's power switch to the **I** (on) position.

8) The instrument's front panel **PWR**, **UNLK**, and **MSTR** LEDs should light and "Model 7210" will appear on the computer monitor.

9) Type "ID" and press the return key to send the ID command to the instrument. The response should be "7210", a newline, carriage return and a "?" character.

10) Use a signal generator capable of generating a 1 kHz output sinewave in the range 0.1 V to 1.0 V rms and fitted with a TTL sync output. Connect the sync output to the front panel **REF 1 IN** connector. The **UNLK** LED should extinguish.

11) Type "FRQ1" and press the return key to send the FRQ1 command to the instrument. The response should be "1000", a newline, carriage return and a "*" character., indicating a detected reference frequency on 1.000 kHz

12) If the unit is fitted with voltage mode input signal boards (7210/99), set the signal generator output amplitude to 1 V rms and apply this to the front panel signal input connector marked "**1**".

    If it is fitted with wide bandwidth current mode input signal boards (7210/98), set the signal generator output amplitude to 1 V rms and apply this via series resistor of 1 MΩ (1% accuracy) to the front panel signal input connector marked "**1**". This resistor generates a nominal current of 1 µA rms.

    If it is fitted with low noise current mode input signal boards (7210/97)), set the signal generator output amplitude to 0.1 V rms and apply this via a series resistor of 1 MΩ (1% accuracy) to the front panel signal input connector marked "**1**". This resistor generates a nominal current of 1 µA rms.

13) Type "SEN1 1 9" and press the return key to set the channel 1 full-scale sensitivity to the least sensitive range (1 V for voltage mode inputs, 1 µA for wide bandwidth current mode inputs and 100 nA for low noise current mode

inputs) and wait a second or so to allow the channel to settle. Then type "AQN1 1; X1 1" and press the return key to perform an autophase and read the X1 output. The response should be "10000", a newline and carriage return and a "*" character, although the actual number received may differ by up to ±500 counts, depending on the real level of applied signal.

14) Turn off the model 7210 and, if the DIP switch settings were changed at step 2, restore them to their initial values.

This completes the initial checks. Even though the procedure leaves many functions untested, if the indicated results were obtained then the user can be reasonably sure that the unit incurred no hidden damage in shipment and is in good working order.

Technical Description

Chapter 3

3.1 Introduction

The model 7210 is a very sophisticated instrument and its tandem demodulation mode means that its operation is more complex than conventional lock-in amplifiers. This chapter discusses the operating modes it offers and describes how these are implemented by considering the instrument as a series of functional blocks. A good understanding of the design will allow the user to make best use of the instrument's versatility.

3.2 Operating Modes

3.2.01 Introduction

Throughout this manual, reference is made to the two different operating modes within the instrument, so in order to aid the reader's understanding they are defined here.

3.2.02 Single Reference Mode

This is the conventional mode of operation common to all lock-in amplifiers. The instrument measures the amplitude of the two components of the signal at its inputs that are in-phase and in quadrature (i.e. 90° out of phase) with an internally generated sinusoidal demodulator signal. This demodulator signal is in turn phase locked to the applied external reference signal.

The two measured components per signal channel are conventionally known as the X and Y channel outputs. All 32 signal channels are measured with respect to the same external reference signal, so the instrument generates 64 output values.

3.2.03 Tandem Reference Mode

If an amplitude-modulated sinusoidal *carrier* signal is applied to a conventional lock-in amplifier which is operated at the carrier frequency and reference phase adjusted to yield zero Y channel output, then the X output signal will be the modulating signal. This only applies if the output time constant of the first lock-in is sufficiently short to allow the modulation to pass.

If this X output signal is then applied to a second lock-in amplifier, but this time running at the modulating frequency, then this second lock-in can directly measure the amplitude of the modulation.

In the past this type of experiment would have required two lock-in amplifier, with a physical cable coupling the X output of one to the input of the second. However, the 7210 includes this capability as a standard feature, subject to only a few limitations.

In order to allow the second demodulator to run synchronously with the first, it is desirable for its reference frequency to be the result of an integer division of the first reference frequency. This condition is best satisfied by ensuring that the second reference frequency be internally generated by the instrument and made available via a connector so that it can be used as the source of modulation for the signal.

Consequently the 7210 is fitted with two reference connectors; **REF 1 IN** is used to apply the external reference frequency at which the first demodulation stage operates, and the second. **REF 2 OUT** outputs a TTL reference waveform at the frequency of the second stage. The user can specify the divisor used to generate the second reference from the first.

It will be appreciated that in tandem mode there are four outputs per signal channel, an X and Y pair from the first stage and an X and Y pair from the second. To avoid confusion, the outputs from the first stage, *even when the unit is operating in single reference mode*, are referred to as X1 and Y1 and those from the second as X2 and Y2.

It will also be seen that in Tandem mode the instrument generates 128 output values.

3.3 Principles of Operation

3.3.01 Block Diagram

The model 7210 uses 32 ADC's and digital signal processors (DSP's), a microprocessor and very low-noise analog circuitry to achieve its specifications. A block diagram of the instrument is shown in figure 3-1. The sections that follow describe how each functional block operates and the effect it has on the instrument's performance.

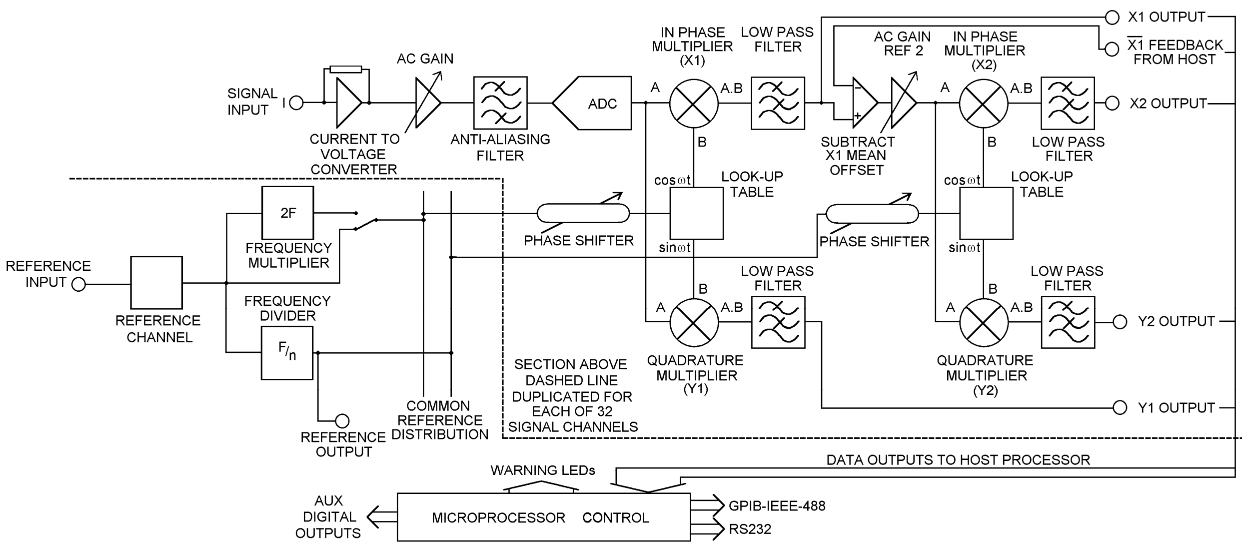

Figure 3-1, Model 7210 - Block Diagram

3.3.02 Signal Channel Inputs

The signal input amplifier depends on the type of input signal boards fitted:

7210/99 Signal Board - Voltage Mode Inputs
In this case the input amplifier is a buffer stage with an input impedance of 10 MΩ. The frequency response (-3 dB) extends from 20 Hz to 50.5 kHz

7210/98 Signal Board - Wide Bandwidth Current Mode Inputs
In this case the input amplifier is a single-ended current mode device with a fixed transimpedance setting of 10E6 V/A. Hence an applied current of 1 µA will give a 1 V signal into the next stage. The frequency response (-3 dB) extends from 20 Hz to 50.5 kHz

7210/97 Signal Board - Low Noise Current Mode Inputs
In this case the input amplifier is a single-ended current mode device with a fixed transimpedance setting of 10E7 V/A. Hence an applied current of 100 nA will give a 1 V signal into the next stage. The frequency response (-3 dB) extends from 20 Hz to 5.0 kHz.

3.3.03 AC Gain and Dynamic Reserve

The signal channel contains a number of analog filters and amplifiers whose overall gain is defined by the AC Gain parameter, which is specified in terms of decibels (dB). For each value of AC Gain there is a corresponding value of the INPUT LIMIT parameter, which is the maximum instantaneous (peak) voltage or current that can be applied to the input without causing input overload, as shown in table 3-1 below.

AC Gain (dB)	INPUT LIMIT (V) Voltage Mode	INPUT LIMIT (nA) Wide Bandwidth Current Mode	INPUT LIMIT (nA) Low Noise Current Mode
0	3.1	3100	310
10	1.5	150	15
20	0.31	31	3.1
30	0.15	15	1.5
40	0.031	3.1	0.31
50	0.015	1.5	0.15
60	0.0031	0.31	0.031

Table 3-1, Input Limit vs AC Gain

It is a basic property of the digital signal processing (DSP) lock-in amplifier that the best demodulator performance is obtained by presenting as large a signal as possible to the main analog-to-digital converter (ADC). Therefore, in principle, the AC Gain value should be made as large as possible without causing the signal channel amplifier or converter to overload. This constraint is not too critical however and the use of a value 10 or 20 dB below the optimum value makes little difference. Note that as the AC Gain value is changed, the demodulator gain (described later in section 3.3.08) is also adjusted in order to maintain the selected full-scale sensitivity.

The full-scale sensitivity is set by a combination of AC Gain and demodulator gain. Since the demodulator gain is entirely digital, changes in full-scale sensitivity which do not change the AC Gain do not cause any of the errors which might arise from a change in the AC Gain.

The user is prevented from setting an illegal AC Gain value, i.e. one that would result in overload on a full-scale input signal. Similarly, if the user selects a full-scale sensitivity which causes the present AC Gain value to be illegal, the AC Gain will change to the nearest legal value.

In practice, this system is very easy to operate. However, the user may prefer to make use of the AUTOMATIC AC Gain feature which gives very good results in most cases. When this is active the AC Gain is automatically controlled by the instrument, which determines the optimum setting based on the full-scale sensitivity currently being used.

At any given setting, the ratio

$$\mathrm{DR} = 0.7 \times \frac{\text{Input Limit}}{\text{Full-Scale Sensitivity}}$$

represents the factor by which the largest acceptable sinusoidal interference input exceeds the full-scale sensitivity and is called the Dynamic Reserve of the lock-in amplifier at that setting. (The factor 0.7 is a peak-to-rms conversion). The dynamic reserve is often expressed in decibels, for which

$$\mathrm{DR(in\ dB)} = 20 \times \log(\mathrm{DR(as\ a\ ratio)})$$

Applying this formula to the model 7210 at the maximum value of INPUT LIMIT and the smallest available value of FULL-SCALE SENSITIVITY, gives a maximum available dynamic reserve of better than 80 dB.

3.3.04 Anti-Aliasing Filter

Prior to the main analog-to-digital converter (ADC) the signal passes through an anti-aliasing filter to remove unwanted frequencies which would cause a spurious output from the ADC due to the sampling process.

Consider the situation when the lock-in amplifier is measuring a sinusoidal signal of frequency f_{signal} Hz, which is sampled by the main ADC at a sampling frequency $f_{sampling}$ Hz. In order to ensure correct operation of the instrument the output values representing the f_{signal} frequency must be uniquely generated by the signal to be measured, and not by any other process.

However, if the input to the ADC has, in addition, an unwanted sinusoidal signal with frequency f_1 Hz, where f_1 is greater than half the sampling frequency, then this will appear in the output as a sampled-data sinusoid with frequency less than half the sampling frequency, $f_{alias} = |f_1 - nf_{sampling}|$, where n is an integer. This alias signal is indistinguishable from the output generated when a genuine signal at frequency f_{alias} is sampled. Hence if the frequency of the unwanted signal were such that the alias signal frequency produced from it was close to, or equal to, that of the wanted signal then it is clear that a spurious output would result.

To overcome this problem the signal is fed through the anti-aliasing filter which restricts the signal bandwidth. The filter takes the form of two stacked third-order Butterworth low-pass filters with a nominal -3 dB cut-off frequency of 70 kHz. The signal channel also incorporates AC coupling at several points, so the overall effect is to restrict the signal bandwidth to nominally 20 Hz to 51 kHz.

It should be noted that the dynamic range of a lock-in amplifier is normally so high that practical anti-alias filters are not capable of completely removing the effect of a full-scale alias. For instance, even if the filter gives 100 dB attenuation, an alias at the input limit and at the reference frequency will give a one percent output error when the dynamic reserve is 60 dB.

In a typical low-level signal recovery situation, many unwanted inputs need to be dealt with and it is normal practice to make small adjustments to the reference frequency until a clear point on the frequency spectrum is reached. In this context an unwanted alias is treated as just another interfering signal and its frequency is avoided when choosing the reference frequency.

3.3.05 Main Analog-to-Digital Converter

Following the anti-alias filter the signal passes to the analog-to-digital converter running at a sampling rate of nominally 250 kHz. This rate is not fixed, but is automatically adjusted so that it is an exact integer multiple of the applied **REF 1 IN** reference frequency. The reason for doing this is that it means that the internally generated digital representations of a cosine and sine wave by which the sampled signal is multiplied in the demodulators are exactly correct, with no need to interpolate between values stored in the look-up tables. It also means that the output filters are inherently synchronous, reducing the noise seen at the output of asynchronous filters operated at short time-constants. At reference frequencies below 200Hz the sampling rate drops to nominally 40 kHz. Again, the rate is not fixed but is set to be an integer multiple of the reference frequency.

The output from the converter feeds a digital signal processor, the demodulator DSP, which implements all four digital multipliers (demodulators) and the first stages of the output low-pass filtering. However, before discussing this further, the reference channel, which provides the other inputs to the demodulators, will be described.

3.3.06 Reference Channel

The reference channel contains the reference trigger/phase-locked loop, a reference multiplier and a reference divider, and two digital phase-shifters per signal channel. (see figure 3-1).

In single reference mode, the reference channel locks to the applied external reference signal and its output, referred to REF 1, is fed either directly or via a 2F frequency multiplier to all 32 of the phase shifters used for the first stage of demodulation. The 2F multiplier allows the instrument to detect at the second harmonic of the reference frequency rather than at is fundamental frequency, should this be required.

In tandem mode, the reference channel output is also fed to the 32 of the phase shifters used for the first stage of demodulation, but the 2F detection option is not allowed. In addition, it feeds a programmable divider that generates the second reference frequency, REF2, that in turn is applied to the second set of 32 phase shifters used for the second stage of demodulation. REF2 is also output as a TTL signal the front-panel **REF 2 OUT** connector.

3.3.07 Phase-Shifters

Each of the 32 channels includes two digital phase shifters that can be used to bring the signal and corresponding reference waveforms applied to the X1 and X2 demodulators into alignment. The output from the phase shifter is used to derive cosinusoidal and sinusoidal waveforms which are then applied to the X1 and Y1 demodulators respectively.

If the reference input is a sinusoid applied to the **REF 1 IN** socket, the reference phase is defined as the phase of the X1 demodulation function with respect to the reference input.

This means that when the reference phase is zero and the signal input to the demodulator is a full-scale sinusoid in phase with the reference input sinusoid, the X1 channel output of the demodulator is a full-scale positive value and the Y1 channel output is zero.

The circuits connected to the **REF 1 IN** socket detect a positive-going crossing of the mean value of the applied reference voltage. Therefore when the reference input is not sinusoidal, its effective phase is the phase of a sinusoid with a positive-going zero crossing at the same point in time, and accordingly the reference phase is defined with respect to this waveform.

3.3.08 Demodulator DSP

The demodulator DSP implements the four demodulators and the first stages of the four output filters. Each demodulator consists of a multiplier which simply multiplies the signal input values, (in the first stage the value from the ADC and in the second the X1 output values) with corresponding values of a unity-amplitude cosine or sine waveform at the same point in time.

Each demodulator output is digitally scaled to provide the demodulator gain control.

As discussed earlier in section 3.3.04 this gain is adjusted as the AC Gain is adjusted to maintain a given full-scale sensitivity.

Each demodulator output is then filtered by a digital finite impulse response (FIR) filter before passing to the host microprocessor for further filtering if required. As with most lock-in amplifiers, the output filters in the model 7210 are described as having a characteristic "slope". This may seem somewhat strange, and a few words of explanation may be helpful.

In traditional audio terminology, a second-order low-pass filter is described as having a slope of 12 dB per octave. This term has become part of the accepted terminology relating to lock-in amplifier output filters and is used in the model 7210 to apply to the envelope of the frequency response function of the FIR output filters. For the sake of simplicity, all the filters in the 7210 have a fixed slope of 12dB/octave.

3.3.09 Main Microprocessor - General

All functions of the instrument are under the control of a microprocessor which in addition drives the front-panel LEDs and supports the RS232 and GPIB (IEEE-488) computer interfaces. This processor also drives the instrument's 8-bit digital (TTL) programmable output port, which may be used for controlling auxiliary apparatus.

The microprocessor has access to 128 data memory locations that may be used to store X1, Y1, X2 and Y2 values on receipt of a GPIB trigger or command. This ensures that all outputs are sampled at the same time. Once stored, the acquired values can be downloaded to the controlling PC for further data manipulation.

A particularly useful feature of the design is that only part of the controlling firmware program code, which the microprocessor runs, is permanently resident in the instrument. The remainder is held in a flash EEPROM and can be updated via the RS232 computer interface. It is therefore possible to change the functionality of the instrument, perhaps to include a new feature or update the computer command set, simply by connecting it to a computer and running an update program.

3.3.10 Main Microprocessor - Auto Functions

The microprocessor also controls the instrument's auto functions, which are control operations executed by means of a single command. These functions allow easier, faster operation in most applications, although direct manual operation or special purpose control programs may give better results in certain circumstances. During application of several of the auto functions, decisions are made on the basis of output readings made at a particular moment. Where this is the case, it is important for the output time constant set by the user to be long enough to reduce the output noise to a sufficiently low level so that valid decisions can be made and that sufficient time is allowed for the output to settle.

The following sections contain brief descriptions of the auto functions. Note that for each of the 32 channels, there are auto-functions applying to both the REF1 and, in tandem mode, REF2 demodulators.

Auto-Sensitivity
This function operates at all reference frequencies in the conventional reference mode

and the first stage of the tandem mode, but only for **REF 2 OUT** reference frequencies above 100 mHz in the tandem mode.

The Auto-Sensitivity operation first increases the AC gain until input overload occurs, and then decreases it by one step to remove the overload. The full-scale sensitivity range is then adjusted until the signal magnitude (i.e. the square root of the sum of the squares of X1 and Y1, or X2 and Y2) lies in the range 110% to 30% of full-scale.
In the presence of noise, or a time-varying input signal, it may be a long time before the Auto-Sensitivity sequence comes to an end, and the resulting setting may not be what is really required.

Note: The auto-sensitivity operation can take a significant time (in some cases over a minute) per channel to complete, so it should be used with care.

Auto-Phase
In an Auto-Phase operation the value of the signal phase is computed and an appropriate phase-shift is then introduced into the reference channel so as to bring the value of the signal phase to zero. The intended result is to null the output of the Y channel while maximizing the output of the X channel.

Any small residual phase can normally be removed by calling Auto-Phase for a second time, after a suitable delay to allow the outputs to settle.

The Auto-Phase facility is normally used with a clean signal which is known to be of stable phase. It usually gives very good results provided that the X channel and Y channel outputs are steady when the procedure is called.

Auto-Measure
This function is essentially a combination of an auto sensitivity operation followed by an auto-phase performed first on the main (first stage) demodulators, and, if they are active, then on the second (tandem) demodulators.

The Auto-Measure function is intended to give a quick setting of the instrument which will be approximately correct in typical simple measurement situations. For optimum results in any given situation, it may be convenient to start with Auto-Measure and to make subsequent modifications to individual controls.

Note: The auto-measure operation can take a significant time (in some cases over a minute) per channel to complete, so it should be used with care.

The following auto function operates on the complete instrument, not on each channel.

Auto-Default
With an instrument of the complexity of the model 7210 where there are many controls of which only a few are regularly adjusted, it is very easy to overlook the setting of one of them. Consequently an Auto-Default function is provided, which sets all the controls to a defined state. This is most often used as a rescue operation to bring the instrument into a known condition when it is giving unexpected results and is equivalent to cycling the instrument's power.

3.3.11 Main Microprocessor - Curve Buffer

The microprocessor has access to a 128,000 point memory which can be used for storage of sets of 32 selected instrument outputs as curves, prior to their transfer to a computer via one of the interfaces. This function can store the X1, X2, Y1, Y2 outputs, as well as the REF1 and REF2 frequencies. In addition the buffer can be used in first-in, first out (FIFO) mode for experiments when acquisition lengths become limited only by the supporting computer's ability to process the data.

This completes the description of the main functional blocks of the instrument.

3.4 General

3.4.01 Accuracy

When the demodulator is operating under correct conditions, the absolute gain accuracy of the instrument is limited by the analog components in the signal channel, and the absolute phase accuracy is limited by the analog components in both the signal channel and the reference channel. The resulting typical accuracy is ±0.5 percent of the full-scale sensitivity and ±2.0 degrees respectively, for an AC Gain of 0 dB. When the higher values of AC Gain are in use, the errors tend to increase above 5 kHz.

3.4.02 Power-up Defaults

Unlike most other SIGNAL RECOVERY **lock-in amplifiers, instrument settings are not retained on power down.** Consequently the user must ensure that all controls are reset as required by the controlling program each time the instrument is powered-up.

Front and Rear Panels

4.1 Front Panel

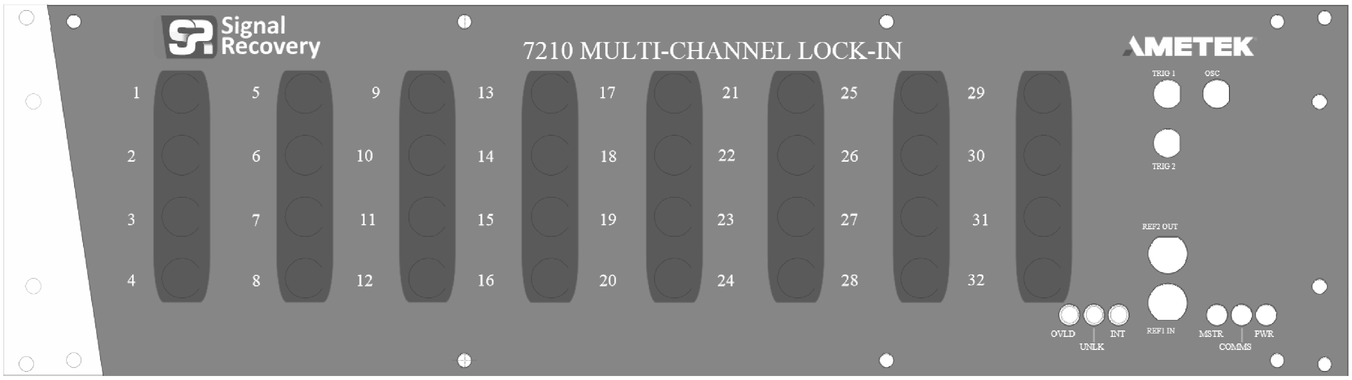

Figure 4-1, Model 7210 Front Panel Layout

As shown in figure 4-1, there are 36 BNC connectors and 6 LED indicators, mounted on the model 7210's front panel. The following sections describe the function and location of these items.

4.1.01 Signal Input Connectors

The 32 connectors numbered 1 to 32 are the signal input connectors.

4.1.02 REF 1 IN Connector

This is the input connector for the external reference signal, which must lie in the frequency range 20 Hz to 50.5 kHz. When a suitable source of reference is connected the **UNLK** (Unlock) LED indicator will be extinguished.

Note: When several units are interconnected to give more detection channels, the reference signal should be connected to the one configured as the Master (see section 4.2.07). To minimize coherent pick-up, it is suggested that the reference be applied to the 7210 with the final set of channels (e.g. channels 97 - 128 in a 128-channel system).

4.1.03 REF 2 OUT Connector

This is the output connector for the second reference signals that the unit generates in Tandem reference mode. The signal, which is a TTL level, is generated by dividing the applied reference signal by an integer number and lies in the frequency range 0.001 Hz to 100 Hz. Note that the maximum frequency of the **REF 2** out signal is one half of the applied reference frequency.

Note: When several units are interconnected to give more detection channels, the second reference signal should be taken from the one configured as the Master (see section 4.2.07)

4.1.04 TRIG 1 and TRIG 2 Connectors

The **TRIG 1** connector is a TTL input used for triggering data acquisition to the internal curve buffer, when this has been configured and armed using the TDT (take data triggered) command. When using the curve buffer, it is also possible to configure the instrument so that trigger events cause a set of "illegal" values to be stored as a marker into the curve that is being taken at the time they occur. When the data is then downloaded, these points are easily identified.

4.1.05 Indicators

Although the instrument is designed for operation from a computer, the six LED status indicators provide useful feedback to the user. They have the following significance.

OVLD
This LED lights when any one of the signal channels is in input and/or output overload.

UNLK
This LED lights to indicate that the reference channel is not locked to the external reference signal. Check the applied signal amplitude, frequency and connections.

INT
This LED is provided for possible future expansion.

MSTR
This indicator lights when the unit is configured as the Master instrument in a series of interconnected instruments. The Master/Slave setting is adjusted using the rear panel DIP switches and only one instrument in a series should be configured as the Master, with the others being set as Slaves.

COMMS
This indicator flashes whenever communication activity is taking place over the RS232 or GPIB interfaces.

PWR
This LED lights whenever line power is applied and the unit is switched on.

4.2 Rear Panel

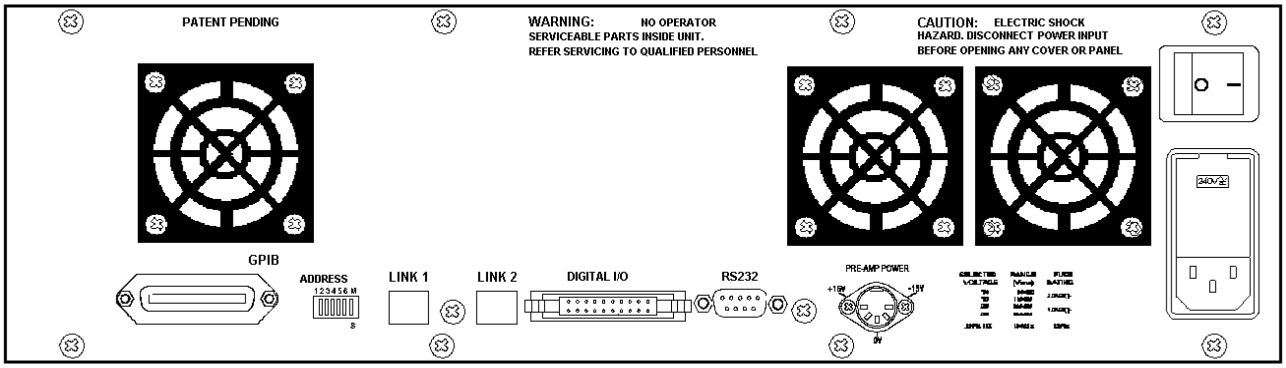

Figure 4-2, Model 7210 Rear Panel Layout

As shown in figure 4-2, the line power switch, line power voltage selector, RS232 connector, GPIB (IEEE-488) connector, digital output port and two RJ45 multipole connectors, and a preamplifier power connector are mounted on the rear panel of the instrument. Brief descriptions of these are given in the following text.

4.2.01 Line Power Switch

Press the end of the switch marked **I** to turn on the instrument's power, and the other end marked **O** to turn it off.

4.2.02 Line Power Input Assembly

This houses the line voltage selector and line input fuse. To check, and if necessary change, the fuse or line voltage see the procedure in section 2.1.04.

4.2.03 RS232 Connector

This 9-pin D type RS232 interface connector implements pins 1, 2, 3 and 7 (Earth Ground, Transmit Data, Receive Data, Logic Ground) of a standard DTE interface. To make a connection to a PC-compatible computer, it is normally sufficient to use a three-wire cable connecting Transmit Data to Receive Data, Receive Data to Transmit Data, and Logic Ground to Logic Ground. Appendix D shows the connection diagrams of cables suitable for computers with 9-pin and 25-pin serial connectors. Pinouts for this connector are given in appendix B.

4.2.04 GPIB Connector

The GPIB interface connector conforms to the IEEE-488 1978 Instrument Bus Standard. The standard defines all voltage and current levels, connector specifications, timing and handshake requirements.

4.2.05 DIGITAL I/O Connector

This connector provides eight TTL output lines, each of which can be set high or low via the computer interfaces. It is most commonly used for controlling auxiliary apparatus, such as lamps, shutters and heaters. Pinouts for this connector are given in appendix B.

4.2.06 PREAMP POWER Connector

This connector supplies ±15 V at up to 250 mA and can be used for powering any of several optional remote preamplifiers available from SIGNAL RECOVERY, including the model 7210/90 four channel optical input preamplifier that is specifically intended for use with the model 7210. Pinouts for this connector are given in appendix B.

4.2.07 LINK 1 and LINK 2 connectors

These two RJ45 connectors are used to connect two or more 7210 instruments together to build multi-channel instruments with more than 32 signal channels. A cable should be connected from the **LINK 2** socket on the 7210 that is set to be the Master unit to the **LINK 1** socket on the second unit, configured as a Slave. If more than two 7210's are interconnected, then the **LINK 2** socket on the first Slave should be coupled to the **LINK 1** socket on the second Slave, and so on.

4.2.08 GPIB Address & Master/Slave Switch

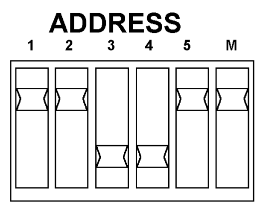

Figure 4-3, Model 7210 GPIB Address and Master/Slave Switch

The GPIB Address and Master/Slave switch sets the address and status as follows:-

GPIB ADDRESS
The GPIB address is set in the range 1 to 30 by moving the five left-hand DIP switches so that they represent the binary coded decimal (BCD) version of the required address, where switch 1 is the least significant bit, and the up position is logic "0". Hence, for example, to set the address to 8 whose 5-bit BCD equivalent is 01000, the switches should be in the following positions, reading from left to right:

UP, UP, UP, DOWN, UP.

Figure 4-3 shows the address set to the SIGNAL RECOVERY default value of 12.

Note: All instruments connected to the bus must have a unique address.

Master/Slave Switch
This switch defines whether the instrument is the master unit (switch in UP position) or a slave unit (switch in DOWN position) when several 7210's are interconnected via their **LINK** connectors.

Note: Only one instrument in an interconnected series of units should be configured as the Master; all others must be set to be Slaves. If only one instrument is in use, it must be set as the Master.

Computer Operation

Chapter 5

5.1 Introduction

The model 7210 includes both RS232 and GPIB (IEEE-488) interface ports, designed to allow the lock-in amplifier to be completely controlled from a remote computer. All of the instrument's controls other than the GPIB address and master/slave status, can be operated, and all of the outputs can be read, via these interfaces.

This chapter describes the capabilities of the instrument when operated remotely and discusses how this is done

5.2 RS232 and GPIB Operation

5.2.01 Introduction

Control of the lock-in amplifier from a computer is accomplished by means of communications over the RS232 or GPIB interfaces. The communication activity consists of the computer sending commands to the lock-in amplifier, and the lock-in amplifier responding, either by sending back some data or by changing the setting of one of its controls. The commands and responses are encoded in standard 7-bit ASCII format, with one or more additional bits as required by the interface (see below).

The two ports cannot be used simultaneously, but when a command has been completed, the lock-in amplifier will accept a command at either port. Also when the test echo facility has been activated all output from the computer to the GPIB can be monitored by a terminal attached to the RS232 connector.

Although the interface is primarily intended to enable the lock-in amplifier to be operated by a computer program specially written for an application, it can also be used in the direct, or terminal, mode. In this mode the user enters commands on a keyboard and reads the results on a monitor screen.

The simplest way to establish the terminal mode is to connect a standard terminal, or a terminal emulator, to the RS232 port. A terminal emulator is a computer which runs special-purpose software that makes it act as a terminal. In the default (power-up) state of the port, the lock-in amplifier sends a convenient prompt character when it is ready to receive a command, and echoes each character that is received.

Microsoft Windows 95/98 includes a program called HyperTerminal, usually to be found in the Accessories group, which can be used as a terminal emulator. Alternatively a simple terminal program with minimal facilities can be written in a few lines of BASIC code (see appendix C.1).

5.2.02 RS232 Interface - General Features

The RS232 interface in the model 7210 is implemented with three wires; one carries digital transmissions from the computer to the lock-in amplifier, the second carries digital transmissions from the lock-in amplifier to the computer and the third is the Logic Ground to which both signals are referred. The logic levels are ±12 V referred to Logic Ground, and the connection may be a standard RS232 cable in conjunction with a null modem or, alternatively, may be made up from low-cost general purpose cable. The pinout of the RS232 connectors are shown in appendix B and cable diagrams suitable for coupling the instrument to a computer are shown in appendix D.

The main advantages of the RS232 interface are:

1) It communicates via a serial port which is present as standard equipment on nearly all computers, using leads and connectors which are available from suppliers of computer accessories or can be constructed at minimal cost in the user's workshop.

2) It requires no more software support than is normally supplied with the computer, for example Microsoft's GWBASIC, QBasic or Windows HyperTerminal.

A single RS232 transmission consists of a start bit followed by 7 or 8 data bits, an optional parity bit, and 1 stop bit. The rate of data transfer depends on the number of bits per second sent over the interface, usually called the baud rate. In the model 7210 the baud rate can be set to a range of different values up to 19,200, corresponding to a minimum time of less than 0.5 ms for a single character.

Mostly for historical reasons, there are a very large number of different ways in which RS232 communications can be implemented. Apart from the baud rate options, there are choices of data word length (7 or 8 bits), parity check operation (even, odd or none), and number of stop bits (1 or 2). With the exception of the number of stop bits, which is fixed at 1, these settings may be adjusted using the RS232 Settings menu, discussed in chapter 5. They may also be adjusted by means of the RS command.

NOTE: In order to achieve satisfactory operation, the RS232 settings must be set to exactly the same values in the terminal or computer as in the lock-in amplifier.

5.2.03 Choice of Baud Rate

Where the lock-in amplifier is connected to a terminal or to a computer implementing an echo handshake, the highest available baud rate of 19,200 is normally used if, as is usually the case, this rate is supported by the terminal or computer. Lower baud rates may be used in order to achieve compatibility with older equipment or where there is some special reason for reducing the communication rate.

5.2.04 Choice of Number of Data Bits

The model 7210 lock-in amplifier uses the standard ASCII character set, containing 127 characters represented by 7-bit binary words. If an 8-bit data word is selected, the most significant bit is set to zero on output from the lock-in amplifier and ignored on input. The result is that either the 8-bit or the 7-bit option may be used, but the 7-bit option can result in slightly faster communication.

5.2.05 Choice of Parity Check Option

Parity checks are not required at the baud rates available in the model 7210, that is up to 19,200 baud, with typical cable lengths of up to a few meters. Therefore no software is provided in the model 7210 for dealing with parity errors. Where long cables are in use, it may be advisable to make use of a lower baud rate. The result is that any of the parity check options may be used, but the no-parity option will result in slightly faster communication.

Where the RS232 parameters of the terminal or computer are capable of being set to any desired value, a choice must be made. In the model 7210 the combination set at the factory is even parity check, 7 data bits, and one stop bit (fixed).

5.2.06 GPIB Interface - General Features

The GPIB is a parallel digital interface with 8 bi-directional data lines, and 8 further lines which implement additional control and communication functions. Communication is through 24-wire cables (including 8 ground connections) with special-purpose connectors which are constructed in such a way that they can be stacked on top of one another to enable numerous instruments to be connected in parallel. By means of internal hardware or software switches, each instrument is set to a different address on the bus which is a number in the range 1 to 30. In the model 7210 the address is set using the rear-panel DIP switches (see section 4.2.07).

A most important aspect of the GPIB is that its operation is defined in minute detail by the IEEE-488 standard, usually implemented by special-purpose semiconductor devices that are present in each instrument and communicate with the instrument's microprocessor. The existence of this standard greatly simplifies the problem of programming the bus controller, i.e. the computer, to implement complex measurement and test systems involving the interaction of numerous instruments. There are fewer interface parameters to be set than with RS232 communications.

The operation of the GPIB requires the computer to be equipped with special-purpose hardware, usually in the form of a plug-in card, and associated software which enable it to act as a bus controller. The control program is written in a high-level language, usually BASIC or C, containing additional subroutines implemented by software supplied by the manufacturer of the interface card.

Because of the parallel nature of the GPIB and its very effective use of the control lines, including the implementation of a three-wire handshake (see below), comparatively high data rates, up to a few hundred thousand bytes per second, are possible. In typical setups the data rate of the GPIB itself is not the factor that limits the rate of operation of the control program, but rather the rate with which it can process the data received.

5.2.07 Handshaking and Echoes

A handshake is a method of ensuring that the transmitter does not send a byte until the receiver is ready to receive it, and, in the case of a parallel interface, that the receiver reads the data lines only when they contain a valid byte.

GPIB Handshaking

The GPIB interface includes three lines (*DAV, *NRFD, *NDAC) which are used to implement a three-wire handshake. The operation of this is completely defined by the IEEE-488 standard and is fully automatic, so that the user does not need to know anything about the handshake when writing programs for the GPIB. Note that each command must be correctly terminated, which is best done by arranging for the software to assert the GPIB line EOI (end or identify) with the transmission of the last character. This avoids the need to send additional characters, such as the carriage return and/or line feed characters and thereby improves the data transfer rate.

RS232 Handshaking

In the RS232 standard there are several control lines called handshake lines (RTS, DTR outputs and CTS, DSR, DCD inputs) in addition to the data lines (TD output and RD input). However, these lines are not capable of implementing the handshaking function required by the model 7210 on a byte-by-byte basis and are not connected in the model 7210 apart from the RTS and DTR outputs which are constantly asserted.

Note that some computer applications require one or more of the computer's RS232 handshake lines to be asserted. If this is the case, and if the requirement cannot be changed by the use of a software switch, the cable may be used in conjunction with a null modem. A null modem is an adapter which connects TD on each side through to RD on the other side, and asserts CTS, DSR, and DCD on each side when RTS and DTR are asserted on the opposite sides.

With most modern software there is no need to assert any RS232 handshake lines and a simple three-wire connection can be used. The actual handshake function is performed by means of bytes transmitted over the interface.

The more critical handshake is the one controlling the transfer of a command from the computer to the lock-in amplifier, because the computer typically operates much faster than the lock-in amplifier and characters can easily be lost if the command is sent from a program. (Note that because of the limited speed of human typing, there is no problem in the terminal mode.) To overcome this problem an echo handshake is used. This works in the following way: after receiving each character, the lock-in amplifier sends back an echo, that is a character which is a copy of the one that it has just received, to indicate that it is ready to receive the next character. Correspondingly, the computer does not send the next character until it has read the echo of the previous one. Usually the computer makes a comparison of each character with its echo, and this constitutes a useful check on the validity of the communications.

Where the echo is not required, it can be suppressed by negating bit 3 in the RS232 parameter byte. The default (power-up) state of this bit is for it to be asserted.

The program RSCOM2.BAS in section C.2 illustrates the use of the echo handshake.

5.2.08 Terminators

In order for communications to be successfully established between the lock-in amplifier and the computer, it is essential that each transmission, i.e. command or command response, is terminated in a way which is recognizable by the computer and the lock-in amplifier as signifying the end of that transmission.

In the model 7210 there are three input termination options for GPIB communications, selected by means of the GP command. The lock-in amplifier may be set to expect the <CR> byte (ASCII 13) or the <CR,LF> sequence (ASCII 13 followed by ASCII 10) to be appended by the controller as a terminator to the end of each command. Alternatively instead of a terminator it may expect the EOI signal line (pin 5 on the GPIB connector) to be asserted during the transmission of the last character of the command. The third option is normally to be preferred with modern interface cards which can easily be set to a wide variety of configurations.

The selected GPIB termination option applies also to the output termination of any responses sent back by the lock-in amplifier to the controller, i.e. the lock-in amplifier will send <CR> or <CR,LF> or no byte as appropriate. In all cases the lock-in amplifier asserts the EOI signal line during the transmission of the last byte of a response.

In RS232 communications, the lock-in amplifier automatically accepts either <CR> or <CR,LF> as an input command terminator, and sends out <CR,LF> as an output response terminator except when the "no prompt" bit (bit 4 in the RS232 parameter byte) is set, in which case the terminator is <CR>. The default (power-up) state of this bit is zero.

5.2.09 Command Format

The simple commands listed in section 5.3 have one of four forms:

CMDNAME terminator
CMDNAME n terminator
CMDNAME [n] terminator
CMDNAME n_1 [n_2] terminator

where CMDNAME is an alphanumeric string that defines the command, and n, n_1, n_2 are parameters separated by spaces. When n is not enclosed in square brackets it must be supplied. [n] means that n is optional. n_1 [n_2] means that n_1 is required and may optionally be followed by n_2. Upper-case and lower-case characters are equivalent. Terminator bytes are defined in section 5.2.08.

Where the command syntax includes optional parameters and the command is sent without the optional parameters, the response consists of a transmission of the present values of the parameter(s).

Any response transmission consists of one or more numbers followed by a response terminator. In the case of commands that are equivalent to compound commands (see section 5.2.11), each response number is separated by a byte called a delimiter. Some commands have an optional floating point mode which is invoked by appending a **.** (full stop) character to the end of the command and before the parameters. This allows some parameters to be entered or read in floating point format. The floating point output format is given below.

±1.234E±01

The number of digits between the decimal point and the exponent varies depending on the number but is a minimum of one and a maximum of eight. The input format is not as strict but if a decimal point is used there must be a digit before it. An exponent is optional.

5.2.10 Delimiters

Responses from the instrument consist of one or more numbers. In the case of commands that are equivalent to compound commands, such as "XY1", (see section 5.2.11), each number is separated by a byte called a delimiter. This delimiter can be any printing ASCII character and is selected by the DD command. By default, it is a comma character (ASCII decimal code 44).

In the case of commands that are not equivalent to compound commands, such as "X1 0", then the responses are separated in different ways, depending on whether the GPIB or RS232 interfaces are being used.

RS232 Interface
Each response is followed by a carriage return/line feed pair.

GPIB Interface
Each response is followed by the selected terminator, either CR, CR/LF or EOI only, with EOI being asserted with the last character of the response, CR. or LF as appropriate. Bit 7 in the serial poll status byte (which indicates data available) remains asserted until all response values have been read.

5.2.11 Compound Commands

A compound command consists of two or more simple commands separated by semicolons (ASCII 59) and terminated by a single command terminator. No more than one of these commands should be ones that generate a response, since otherwise it would not be possible to identify which command the response was generated by.

5.2.12 Status Byte, Prompts and Overload Byte

An important feature of the IEEE-488 standard is the serial poll operation by which a special byte, the status byte, may be read at any time from any instrument on the bus. This contains information which must be urgently conveyed from the instrument to the controller.

The function of the individual bits in the status byte is instrument dependent, apart from bit 6 (the request service bit) whose functions are defined by the standard.

In the model 7210, bits 0 and 7 signify "command complete" and "data available" respectively. In GPIB communications, the use of these bits can lead to a useful simplification of the control program by allowing the use of a single subroutine which is the same for all commands, whether or not they send a response over the bus. The subroutine should carry out the following sequence of events:

1) Send the command

2) Perform repeated serial polls testing both bit 0 (command complete) and bit 7 (data available) and, if bit 7 is asserted then perform a read operation. This cycle (i.e. test bit 0 (command complete) and test bit 7 (data available)) should then continue until the lock-in amplifier asserts bit 0 to indicate that the command-response sequence is complete, so that the instrument will then be ready for the next command.

This procedure deals successfully with commands generating multiple responses.

In RS232 communications, comparatively rapid access to the status byte is provided by the prompt character which is sent by the lock-in amplifier at the same time as bit 0 becomes asserted in the status byte. This character is sent out by the lock-in amplifier after each command response (whether or not the response includes a transmission over the interface) to indicate that the response is finished and the instrument is ready for a new command. The prompt takes one of two forms. If the command contained an error, either in syntax or by a command parameter being out of range, or alternatively if an overload or reference unlock is currently being reported by the front panel indicators, the prompt is ? (ASCII 63). Otherwise the prompt is * (ASCII 42).

These error conditions correspond to the assertion of bits 1, 2, 3 or 4 in the status byte. When the ? prompt is received by the computer, the ST command may be issued in order to discover which type of fault exists and to take appropriate action.

The prompts are a rapid way of checking on the instrument status and enable a convenient keyboard control system to be set up simply by attaching a standard terminal, or a simple computer-based terminal emulator, to the RS232 port. Where the prompt is not required it can be suppressed by setting the "no prompt" bit, bit 4 in the RS232 parameter byte. The default (power-up) state of this bit is zero.

Because of the limited number of bits in the status byte, it can indicate that an overload exists but cannot give more detail. Two auxiliary commands, OVL and OVR n, give details of the location of the overload.

A summary of the bit assignments in the status byte is given below.

Bit	Status Byte
bit 0	command complete
bit 1	invalid command
bit 2	command parameter error
bit 3	reference unlock
bit 4	input or output overload
bit 5	GPIB Group Execute Trigger or instrument GET 1 command received - Data output buffer update suspended until GET 0 command is received
bit 6	asserted SRQ
bit 7	data available

5.2.13 Service Requests

The interface defined by the IEEE-488 standard includes a line (pin 10 on the connector) called the SRQ (service request) line which is used by the instrument to signal to the controller that urgent attention is required. At the same time that the instrument asserts the SRQ line, it also asserts bit 6 in the status byte. The controller responds by executing a serial poll of all the instruments on the bus in turn and testing bit 6 of the status byte in order to discover which instrument was responsible for asserting the SRQ line. The status byte of that instrument is then further tested in order to discover the reason for the service request and to take appropriate action.

In the model 7210 the assertion of the SRQ line is under the control of a byte called the SRQ mask byte which can be set by the user with the MSK command or via the GPIB Settings menu. If any bit in the status byte becomes asserted, and the corresponding bit in the mask byte has a non-zero value, the SRQ line is automatically asserted. If the value of the mask byte is zero, the SRQ line is never asserted.

Hence, for example, if the SRQ mask byte is set to 16, a service request would be generated as soon as an overload occurred; if the SRQ mask byte were set to 0, then service requests would never be generated.

5.3 Command Descriptions

This section lists the commands in logical groups so that, for example, all commands associated with setting controls which affect the signal channel are shown together. Appendix E gives the same list of commands but in alphabetical order.

In the following commands the parameter n_1 is commonly used to signify which channel(s) of the 32 within the instrument will be affected by the command, as follows:-

n_1	Significance
0	All 32 channels are set to the same value
1	Channel 1
2	Channel 2
..	
31	Channel 31
32	Channel 32

For example, AQN 0 will perform an auto-phase operation on all 32 channels, but AQN 5 will perform it only on Channel 5.

Commands that elicit a response where n_1 is equal to 0 generate 32 response values in the order Channel 1 to Channel 32. In the case of the RS232 interface, each response is terminated with a carriage return/line feed pair, while when using the GPIB interface the final character of each response is indicated by the GPIB line EOI being asserted.

5.3.01 Signal Channel

ACGAIN n_1 [n_2] AC Gain control

Sets or reads the gain of the signal channel amplifier. Values of n_2 from 0 to 6 can be entered, corresponding to the range 0 dB to 60 dB in 10 dB steps.

SEN1 n_1 [n_2]
SEN1. n_1 Full-scale sensitivity control

In single reference mode, the value of n_2 sets the overall full-scale sensitivity of the channel(s) specified by n_1 according to the following table:

n_2	full-scale sensitivity		
	Voltage Mode	Wideband Current Mode	Low Noise Current Mode
1	100 µV	100 pA	10 pA
2	300 µV	300 pA	30 pA
3	1 mV	1 nA	100 pA
4	3 mV	3 nA	300 pA
5	10 mV	10 nA	1 nA
6	30 mV	30 nA	3 nA
7	100 mV	100 nA	10 nA
8	300 mV	300 nA	30 nA
9	1 V	1 µA	100 nA

In tandem mode, the value of n_2 sets the full-scale sensitivity of the main (first stage) demodulator(s) for the channel(s) specified by n_1 according to the following table:

n_2	full-scale sensitivity		
	Voltage Mode	Wideband Current Mode	Low Noise Current Mode
1	100 µV	100 pA	10 pA
2	300 µV	300 pA	30 pA
3	1 mV	1 nA	100 pA
4	3 mV	3 nA	300 pA
5	10 mV	10 nA	1 nA
6	30 mV	30 nA	3 nA
7	100 mV	100 nA	10 nA
8	300 mV	300 nA	30 nA
9	1 V	1 µA	100 nA

In either mode, SEN1. n_1 reads the sensitivity of the channel(s) specified by n_1 in floating-point mode.

SEN2 n_1 [n_2]
SEN2. n_1 Full-scale sensitivity control

In tandem mode, the value of n_2 sets the full-scale sensitivity of the second stage demodulator(s) for the channel(s) specified by n_1 according to the following table:

n_2	full-scale sensitivity		
	Voltage Mode	Wideband Current Mode	Low Noise Current Mode
1	100 µV	100 pA	10 pA
2	300 µV	300 pA	30 pA
3	1 mV	1 nA	100 pA
4	3 mV	3 nA	300 pA
5	10 mV	10 nA	1 nA
6	30 mV	30 nA	3 nA
7	100 mV	100 nA	10 nA
8	300 mV	300 nA	30 nA
9	1 V	1 µA	100 nA

SEN2. n_1 reads the sensitivity of the channel(s) specified by n_1 in floating-point mode.

For example, let the instrument have voltage mode inputs, be in tandem mode with an applied signal consisting of a 10 mV rms 50 kHz sinusoidal waveform modulated to 50% at the REF 2 OUT frequency, with the reference phases correctly adjusted and SEN1 for the relevant channel set to 5. In this case the X1 output will report nominally 100%, while the X2 output, if SEN2 for the same channel is also set to 5, will report nominally 50%.

AS1 n_1 Perform an Auto-Sensitivity operation

Main (first stage) demodulator(s)

AS2 n_1 Perform an Auto-Sensitivity operation

Tandem (second stage) demodulator(s)

ASM n_1 Perform an Auto-Measure operation

Operates on both the main (first stage) and tandem (second stage) demodulator(s)

Note: The auto-sensitivity and auto-measure operations can take a significant time (in some cases over a minute) per channel to complete, so they should be used with care.

AUTOMATIC n_1 [n_2] AC Gain automatic control

n_2	Status
0	AC Gain is under manual control via the ACGAIN command
1	Automatic AC Gain control is activated, with the gain being adjusted according to the full-scale sensitivity setting

OFFSET [n] Automatically Set/Read Interdemodulator Offset Value

When the instrument is operating in tandem mode, the output of the first stage of the demodulator is a DC level with an AC modulation, at the second reference frequency. In order to allow the (wanted) AC signal to occupy as much as possible of the second demodulator's input dynamic range, it is desirable that this DC level is reduced or removed. The OFFSET command allows this to be done.

When the command is sent without the parameter "n", the instrument automatically calculates the optimum offset for each channel and applies it, with the value remaining in effect until next changed or until the instrument is powered down.

The command also operates, for diagnostic purposes only, so that when sent with a parameter "n" in the range 1 to 32, the present offset setting for the corresponding channel (in arbitrary units) is reported.

OVL Locate overload conditions

Causes the lock-in amplifier to respond with four 8-bit numbers expressed as decimal integers in the range 0 to 255, with each response separated from the next by the defined delimiter character. The first number reports the overload conditions in channels 1 to 8, the second in channels 9 to 16, the third in channels 17 to 24 and the fourth in channels 25 to 32.

Within each number, each bit corresponds to the logical ORing of all the bits in the corresponding channel's overload byte. Hence, for example, if channels 1 and 3 are in overload but all other 32 channels are not, then the response would be:

5,0,0,0

In normal use bit 4 in the ST command or GPIB serial poll status byte can be used to identify that an overload has occurred, and when this happens the OVL command can identify which channel(s) are in overload. Finally the OVR command can be used to determine the nature of the overload in the relevant channel(s).

OVR n_1 Report overload byte

Causes the lock-in amplifier to respond, for the channel specified by n_1, with the overload byte, an integer between 0 and 31, which is the decimal equivalent of a binary number with the following bit-significance:

Bit 0	input overload
Bit 1	X1 channel output overload (> ±300 %FS)
Bit 2	Y1 channel output overload (> ±300 %FS)
Bit 3	X2 channel output overload (> ±300 %FS)
Bit 4	Y2 channel output overload (> ±300 %FS)
Bit 5	not used
Bit 6	not used
Bit 7	not used

Note that if bit 1 is set then bits 3 and 4 are meaningless, since if the X1 output is in overload it will not be feeding any valid signal forward in the second (tandem) demodulator stage.

5.3.02 Reference Channel

FRQ1[.] Reference frequency meter

The FRQ1 command causes the lock-in amplifier to respond with 0 if the main (first stage) reference channel is unlocked, or with the reference input frequency if it is locked. In both fixed and floating point mode the response frequency is in Hz.

FRQ2[.] [n] Set/Read Reference output frequency

The value of n sets or reads the frequency of the REF OUT signal when the instrument is operating in Tandem mode. In fixed point mode, n is in millihertz, and in floating point mode, it is in hertz. Note that because of the finite number of possible frequency divisors, the requested value of n may not be the exact frequency generated. To confirm the exact frequency, send the required value of n as a write command FRQ2[.] n and then read the actual value with a write/read command FRQ2[.]

Once set, the frequency of the **REF 2 OUT** signal remains at the same integer division of the applied reference frequency as long as the latter does not change by more than ± 100 Hz. Greater changes cause the reference channel to re-lock and the instrument to determine new divisor to give a **REF 2 OUT** frequency as close as possible to that requested.

REFMODE [n] Reference mode selector

n	Mode
0	Single Reference mode
1	Tandem Mode
2	Fast TC Single Reference mode

The selected mode applies to all 32 channels

REFMODE 2 disables the second stage (tandem) demodulators and allows time constants down to 1 ms to be specified, by extending the legal range of the TC command parameter.

REFN1 [n] Reference harmonic mode control

The value of n sets the main (first stage) reference channel to harmonic detection, as follows:-

n	Mode
1	1F detection
2	2F detection (only available when REFMODE = 0)

The selected mode applies to all 32 channels

REFP1[.] n_1 [n_2] Reference phase control

In fixed point mode n_2 sets the phase of the channel(s) specified by n_1 for the main (first stage) demodulator(s) in millidegrees in the range ±360000.

In floating point mode n_2 sets the phase of the channel(s) specified by n_1 in degrees

REFP2[.] n_1 [n_2] Reference phase control

In fixed point mode n_2 sets the phase of the channel(s) specified by n_1 for the tandem (second stage) demodulator(s) in millidegrees in the range ±360000.

In floating point mode n_2 sets the phase of the channel(s) specified by n_1 in degrees.

AQN1 n_1 Auto-Phase (auto quadrature null)
Main (first stage) demodulator(s)

AQN2 n_1 Auto-Phase (auto quadrature null)
Tandem (second stage) demodulator(s)

5.3.03 Signal Channel Output Filters

TC1 n_1 [n_2]
TC1. n_1 Main (first stage) filter time constant control

n_2	time constant	
-2	1 ms	(REFMODE = 2 only)
-1	2 ms	(REFMODE = 2 only)
0	4 ms	
1	10 ms	
2	30 ms	
3	100 ms	
4	300 ms	
5	1 s	
6	3 s	
7	10 s	
8	30 s	
9	100 s	
10	300 s	
11	1 ks	

The TC1. n_1 command is only used for reading the time constant, and reports the current setting in seconds. Hence if a TC1 1 2 command were sent, TC1 1 would report 2 and TC1. 1 would report 3.0E-02, i.e. 0.03 s or 30 ms.

TC2 n_1 [n_2]
TC2. n_1 Tandem (second stage) filter time constant control

n_2	time constant
2	30 ms
3	100 ms
4	300 ms
5	1 s
6	3 s
7	10 s
8	30 s
9	100 s
10	300 s
11	1 ks

The TC2. n_1 command is only used for reading the time constant, and reports the current setting in seconds. Hence if a TC2 1 3 command were sent, TC2 1 would report 3 and TC2. 1 would report 1.0E-01, i.e. 0.1 s or 100 ms.

5.3.04 Instrument Outputs

BX1 X channel output - Main (first stage) - Binary Mode

This command causes the lock-in amplifier to respond with 64 bytes of data being the X1 demodulator output for all 32 channels. Each value is a sixteen-bit signed integer in the range ±30000, full-scale being ±10000, represented by two bytes in the order High Byte - Low Byte, with the data being transferred in the order Channel 1 to Channel 32.

BY1 Y channel output - Main (first stage) - Binary Mode

This command causes the lock-in amplifier to respond with 64 bytes of data being the Y1 demodulator output for all 32 channels. Each value is a sixteen-bit signed integer in the range ±30000, full-scale being ±10000, represented by two bytes in the order High Byte - Low Byte, with the data being transferred in the order Channel 1 to Channel 32.

BX2 X channel output - Tandem (second stage) - Binary Mode

This command causes the lock-in amplifier to respond with 64 bytes of data being the X2 demodulator output for all 32 channels. Each value is a sixteen-bit signed integer in the range ±30000, full-scale being ±10000, represented by two bytes in the order High Byte - Low Byte, with the data being transferred in the order Channel 1 to Channel 32.

BY2 Y channel output - Tandem (second stage)- Binary Mode

This command causes the lock-in amplifier to respond with 64 bytes of data being the Y2 demodulator output for all 32 channels. Each value is a sixteen-bit signed integer in the range ±30000, full-scale being ±10000, represented by two bytes in the order High Byte - Low Byte, with the data being transferred in the order Channel 1 to Channel 32.

X1[.] n_1 X channel output - Main (first stage)

In fixed point mode causes the lock-in amplifier to respond with the X1 demodulator output of the channel(s) specified by n_1 in the range ±30000, full-scale being ±10000.

In floating point mode causes the lock-in amplifier to respond with the X1 demodulator output of the channel(s) specified by n_1 in amps.

X2[.] n_1 X channel output - Tandem (second stage)

In fixed point mode causes the lock-in amplifier to respond with the X2 demodulator output of the channel(s) specified by n_1 in the range ±30000, full-scale being ±10000.

In floating point mode causes the lock-in amplifier to respond with the X2 demodulator output of the channel(s) specified by n_1 in amps.

XY1[.] n_1 X, Y channel outputs - Main (first stage)

Equivalent to the compound command X1[.] n_1;Y1[.] n_1

XY2[.] n_1 X, Y channel outputs - Tandem (second stage)

Equivalent to the compound command X2[.] n_1;Y2[.] n_1

Y1[.] n_1 Y channel output - Main (first stage)

In fixed point mode causes the lock-in amplifier to respond with the Y1 demodulator output of the channel(s) specified by n_1 in the range ±30000, full-scale being ±10000.

In floating point mode causes the lock-in amplifier to respond with the Y1 demodulator output of the channel(s) specified by n_1 in amps.

Y2[.] n_1 Y channel output - Tandem (second stage)

In fixed point mode causes the lock-in amplifier to respond with the Y2 demodulator output of the channel(s) specified by n_1 in the range ±30000, full-scale being ±10000.

In floating point mode causes the lock-in amplifier to respond with the Y2 demodulator output of the channel(s) specified by n_1 in amps.

5.3.05 Auxiliary Output

BYTE [n] Digital port output control

The value of n, in the range 0 to 255, determines the bits to be output on the rear panel digital output port. Hence, for example, if BYTE = 0, all outputs are low, and when BYTE = 255, all are high.

5.3.06 Output Data Curve Buffer

CBD [n] Curve buffer define

Defines which data outputs are stored in the curve buffer when subsequent TD (take data), TDT (take data triggered) or TDC (take data continuously) commands are issued.

Up to eight curve sets may be stored. Each set either consists of single values per curve, or an array of 32 values representing a full set of 32 output readings defining a curve.

The actual curve sets that will be saved are specified by the CBD parameter, which is an integer between 1 and 247, being the decimal equivalent of an 8-bit binary word. When a given bit is asserted, the corresponding output is selected for storage. When a bit is negated, the output is not stored. The bit function and range for each output are shown in the table below:

Bit	Decimal value	Values/Curve Point	Output and range
0	1	32	X1 Outputs (±10000 FS)
1	2	32	Y1 Outputs (±10000 FS)
2	4	1	Ref 1 frequency (Hz)
3	8	-	Reserved for future use
Tandem Mode Only (i.e. REFMODE = 1)			
4	16	32	X2 Outputs (±10000 FS)
5	32	32	Y2 Outputs (±10000 FS)
6	64	1	Ref 2 frequency bits 0 to 15 (mHz)
7	128	1	Ref 2 frequency bits 16 to 32 (mHz)

Curve sets 6 and 7 store the reference frequency F2 in millihertz. The calculation needed to translate the two 16-bit values to one 32-bit value is:

Reference Frequency F2 = (65536 × value in Curve 7) + (value in Curve 6)

Note that the CBD command directly determines the allowable parameters for the DC and DCB commands. It also interacts with the LEN command and affects the values reported by the M and MAXLEN commands.

No provision is made for storing the instrument's 32 input channel sensitivities when using the curve buffer, so these should not be changed during acquisition since it would not then be possible to correctly interpret the stored X1, Y1, X2 and Y2 values.

LEN [n] Curve sets length control

The value of n specifies the curve sets buffer length in effect for data acquisition. The maximum value is 2,000,000,000, although this is only useful if using the DCFIFO or DCBFIFO readout commands. In normal operation the curve buffer can hold a maximum of 128,000 curve sets, divided between those to be stored. Some examples of the maximum LEN parameter for this mode are shown below.

CBD	max LEN parameter	
1	4000	(128,000 ÷ 32)
2	4000	(128,000 ÷ 32)
3	2000	(128,000 ÷ 64)
4	100,000	(128,000 ÷ 1)
5	3878	(128,000 ÷ 33)
6	3878	(128,000 ÷ 33)
7	1969	(128,000 ÷ 65)

MAXLEN Read maximum curve length

This command reports the maximum curve sets buffer length corresponding to the current setting of CBD when using normal readout (i.e. not DCFIFO or DCBFIFO commands).

NC New curves

Initializes the curve sets storage memory and status variables. All record of previously taken data is removed.

STR [n] Storage interval control

Sets the time interval between successive curve sets being acquired under the TD or TDC commands. n specifies the time interval in milliseconds. When REFMODE ≠ 3 the resolution is 4 ms, input values being rounded up to a multiple of 4. When REFMODE = 3, the resolution is 2 ms, and input values are rounded up to a multiple of 2 The longest interval that can be specified is 1000000 s corresponding to one point in about 12 days. The shortest interval is 2 ms or 4 ms, depending on the REFMODE setting.

TD Take data

Initiates data acquisition. Acquisition starts at the current position in the curve sets buffer and continues at the rate set by the STR command until the buffer is full, or until the HC command is issued.

TDT n Take data triggered

Sets the instrument so that data acquisition will be initiated on receipt of a trigger at the **TRIG IN 1** connector on the front panel. Two triggered modes are possible, as set by the value of n:

n	function
0	One complete set of curves, consisting of the number of curve sets specified by the LEN command parameter, is acquired for each trigger
1	One curve is acquired for each trigger. Hence in order to store a complete set of curves, the number of triggers applied must equal the number of curve sets specified by the LEN command parameter. Note that in this mode the maximum trigger rate is 200 Hz and the storage interval control setting has no effect.

A curve set is stored on the next interrupt after the trigger's rising edge. The minimum trigger pulse high time and the minimum trigger pulse low time are both 2 ms. The minimum pulse period is therefore 4 ms. The trigger has to return low again for a minimum of 2 ms before another trigger can be detected. Although curves may be stored if the minimum pulse high and low times are not adhered to, curves will not be stored on every pulse.

TDC Take data continuously

Initiates data acquisition. Acquisition starts at the current position in the curve buffer and continues at the rate set by the STR command until halted by an HC command. The buffer is circular in the sense that when it has been filled, new data overwrites earlier points.

TMARK [n] Trigger marker control

When using the curve buffer for continuous data acquisition with FIFO readout, it can be useful to be able to identify the time of occurrence of trigger events. The TMARK command allows this. When parameter n is set to 1, trigger events at the front panel **TRIG 1** input connector cause all 32 values of X1, Y1, X2, Y2 to be set to 30001; Freq1 and Freq2 are set to 0. When n is set to 0, the trigger does not "mark" the data set.

The resulting data stream as it is read from the instrument can then be interrogated by user-developed software to see the "illegal" values of 30001, allowing data acquisition to be synchronized with external equipment, while still running uninterrupted.

HC Halt curve acquisition

Halts curve acquisition in progress. It is effective during both single (data acquisition initiated by TD command) and continuous (data acquisition initiated by TDC command) curve acquisitions. The curve may be restarted by means of the TD, TDT or TDC command, as appropriate.

M Curve acquisition status monitor

Causes the lock-in amplifier to respond with four values that provide information concerning data acquisition, as follows:

First value, Curve Acquisition Status: a number with five possible values, defined by the following table:

First Value	Significance
0	No curve activity in progress.
1	Acquisition via TD command in progress and running.
2	Acquisition via TDC command in progress and running.
5	Acquisition via TD command in progress but halted by HC command.
6	Acquisition via TDC command in progress but halted by HC command.

Second value, Number of Sweeps Acquired: This number is incremented each time a TD is completed and each time a full cycle is completed on a TDC acquisition. It is zeroed by the NC command and also whenever a CBD or LEN command is applied without parameters.

Third value, Status Byte: The same as the response to the ST command. The number returned is the decimal equivalent of the status byte and refers to the previously applied command.

Fourth value, Number of Curves Available for Transfer: This number is incremented each time a curve is taken. It is zeroed by the NC command and whenever CBD or LEN is applied without parameters. It is decremented each time curves are read out of the buffer using the DCFIFO or DCBFIFO commands by the number of curves transferred.

DC n Dump acquired curve to computer

This command causes stored curve sets to be dumped via the computer interface in ASCII decimal format.

One curve set only is transferred, as specified by the parameter n, which is the bit number of the required set and which must have been stored by the most recent CBD command. Data is transmitted one curve at a time, starting from the beginning of the buffer. In the case of curves with 32 Values (i.e. curves 0, 1, 4 and 5) the response consists of a total of 32 × (setting of the LEN control) values, ordered as follows (example for curve set 0, with length of 100 curves):

$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 1 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 2 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 3 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 4 in curve set)
.....
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 99 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 100 in curve set)

Each value within a curve is separated from the next by the delimiter that is specified by the DD command - the default delimiter on power-up being a comma (DD 44).

Each curve is separated from the next by a <CR><LF> (carriage return and line feed character pair).

In the case of the other curve sets, the command simply returns (setting of the LEN control) values, again with each one being separated from the next by a <CR><LF> character pair.

The computer program's subroutine which reads the responses to the DC command needs to run a FOR...NEXT loop of appropriate length, storing the responses as required.

Note that when using this command with the GPIB interface the serial poll must be used. After sending the DC command, perform repeated serial polls until bit 7 is set, indicating that the instrument has an output waiting to be read. Then perform repeated reads in a loop, waiting each time until bit 7 is set indicating that a new value is available. The loop should continue until bit 1 is set, indicating that the transfer is completed.

DCFIFO n Dump acquired curve to computer - FIFO mode

This command causes stored data curves to be dumped via the computer interface in ASCII decimal format. Unlike the DC command, it can be used while data acquisition is in progress, thereby making the memory within the instrument essentially function as a first-in, first-out (FIFO) buffer.

The parameter n defines the number of curve sets to transfer, and can be any value between 1 and the present number of curve sets held in the buffer. The latter can be determined by using the M command and reading the fourth value returned by it.

Because of the way that data is organized in the buffer, the DCFIFO command returns all data that has been stored, as defined by the CBD command (unlike the DC command there is no parameter to specify a particular curve set to transfer). The user’s software must therefore parse the data returned by the command and put it into suitable arrays for further processing or storage.

For each curve set transferred, data is transmitted in the following sequence:

$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (X1 values)
$Y1_1$, $Y1_2$, $Y1_3$, $Y1_4$, $Y1_5$, $Y1_6$, $Y1_7$, ..., $Y1_{31}$, $Y1_{32}$ (Y1 values)
FRQ1 (Ref 1 frequency (Hz))
$X2_1$, $X2_2$, $X2_3$, $X2_4$, $X2_5$, $X2_6$, $X2_7$, ..., $X2_{31}$, $X2_{32}$ (X2 values)
$Y2_1$, $Y2_2$, $Y2_3$, $Y2_4$, $Y2_5$, $Y2_6$, $Y2_7$, ..., $Y2_{31}$, $Y2_{32}$ (Y2 values)
$FRQ2_{LSB}$ (Ref 2 frequency bits 0 to 15 (mHz))
$FRQ2_{MSB}$ (Ref 2 frequency bits 0 to 15 (mHz))

subject to the condition that the relevant data curves must have been stored, as specified by the CBD command. Hence, for example, if X1 and Y2 curve sets were stored (CBD = 33) then the responses would be in the following order:

$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 1)
$Y2_1$, $Y2_2$, $Y2_3$, $Y2_4$, $Y2_5$, $Y2_6$, $Y2_7$, ..., $Y2_{31}$, $Y2_{32}$ (curve 1)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 2)

$Y2_1$, $Y2_2$, $Y2_3$, $Y2_4$, $Y2_5$, $Y2_6$, $Y2_7$, ..., $Y2_{31}$, $Y2_{32}$ (curve 2)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 3)
$Y2_1$, $Y2_2$, $Y2_3$, $Y2_4$, $Y2_5$, $Y2_6$, $Y2_7$, ..., $Y2_{31}$, $Y2_{32}$ (curve 3)
..etc.

Each value within a curve is separated from the next by the delimiter that is specified by the DD command - the default delimiter on power-up being a comma (DD 44). Each curve is separated from the next by a <CR><LF> (carriage return and line feed character pair).

The computer program's subroutine which reads the responses to the DCFIFO command needs to run a FOR...NEXT loop of appropriate length, storing the responses as required.

Note that when using this command with the GPIB interface the serial poll must be used. After sending the DCFIFO command, perform repeated serial polls until bit 7 is set, indicating that the instrument has an output waiting to be read. Then perform repeated reads in a loop, waiting each time until bit 7 is set indicating that a new value is available. The loop should continue until bit 1 is set, indicating that the transfer is completed.

Each time a set of curves is transferred, the instrument decrements the number of curves available for transfer, as reported by the M command.

If the controlling software reads data out using the DCFIFO command faster than the instrument is acquiring it, then the maximum acquisition length becomes limited by the software and not by the length of the instrument's memory.

To use this mode of acquisition, issue commands in the following order:

a) Send CBD with a parameter specifying what outputs are to be stored.

b) Send STR with a parameter specifying the data recording interval.

c) Send LEN with a parameter equal to the required total curve sets length. This is the required recording time divided by the recording interval.

d) Start acquisition by sending TD. Triggered mode acquisition using TDT 0 and TDT 1 is also supported.

e) Send the M command and monitor the fourth value returned by it. When this exceeds a suitable threshold value (e.g. 100 curves), send DCFIFO with a parameter equal to this threshold. Read and parse the resulting data from the instrument.

f) Repeat step e) for as long as required, until the acquisition is stopped when the number of curves equals the LEN command parameter, or an HC command is sent to the instrument.

DCB n Dump acquired curve sets to computer in binary format

This command causes a stored curve set to be dumped via the computer interface in binary format, using two bytes per point to transfer each 16-bit 2's complement value, with the MSB transmitted first. The number of data bytes sent is therefore equal to 2 × (current curve length) × Values/Curve.

In order to achieve the maximum transfer rate, no terminators are used within the transmission, although the response is terminated normally at the end.

One curve only is transferred, as specified by the parameter n, which is the bit number of the required curve set and which must have been stored by the most recent CBD command. Data is transmitted one curve at a time, starting from the beginning of the buffer. In the case of curves with 32 Values/Curve (i.e. curves 0, 1, 4 and 5) the response consists of a total of 32 × (setting of the LEN control) values, ordered as follows (example for curve 0, with length of 100 points):

$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$	(curve 1 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$	(curve 2 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$	(curve 3 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$	(curve 4 in curve set)
.....	
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$	(curve 99 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$	(curve 100 in curve set)

In the case of the other curves, the command simply returns (setting of the LEN control) values.

The computer program's subroutine which reads the responses to the DCB command needs to be able to handle the potentially very large data blocks that can be generated. For example, transfer of an output curve set when the curve set length is 1000 points will generate 1000 × 2 × 32 = 64,000 bytes of data.

DCBFIFO n Dump acquired curve to computer - FIFO mode, binary output

This command is similar to the DCFIFO command, except that:

a) Data points are transferred in binary format, using two bytes per point to transfer each 16-bit 2's complement value, with the MSB transmitted first. There are no delimiters between points, curves, or curve sets.

b) Operation when downloading data via the GPIB is a little more complicated, in order to get the fastest possible speeds. Data is transferred in blocks corresponding to eight curve sets, so long as their are still eight or more remaining to transfer. Once there are fewer than eight, then the transfer is for the number remaining.

   For example, suppose CBD = 3, implying that the X1and Y1 output values have been stored, and that 100 curve sets are to be read. In this case, first send the command DCBFIFO 100. Next perform repeated serial polls until bit 7 is set, indicating that the instrument has an output waiting to be read. Then perform a read of (32 x 2 x 2 x 8) bytes (1024 bytes) which will contain the curve sets 1 to 8 of X1 and Y1 curves, in the following order:

$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, \ldots\ldots, X1_{31}, X1_{32}$	(curve 1)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, \ldots\ldots, Y2_{31}, Y2_{32}$	(curve 1)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, \ldots\ldots, X1_{31}, X1_{32}$	(curve 2)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, \ldots\ldots, Y2_{31}, Y2_{32}$	(curve 2)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, \ldots\ldots, X1_{31}, X1_{32}$	(curve 3)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, \ldots\ldots, Y2_{31}, Y2_{32}$	(curve 3)
...	
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, \ldots\ldots, X1_{31}, X1_{32}$	(curve 8)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, \ldots\ldots, Y2_{31}, Y2_{32}$	(curve 8)

Now perform another serial poll; bit 7 will still be set indicating that more data is available. Read this data with another read of 1024 bytes.

Repeat this process, tracking the number of curve sets received, until there are only four left (i.e. after 12 groups). The final read will then need to be of (32 x 2 x 2 x 4) bytes (256 bytes) to complete the transfer. At this point bit 1 in the serial poll byte will is set.

Each time a set of curves is transferred, the instrument decrements the number of curves available for transfer, as reported by the M command.

If the controlling software reads data out using the DCBFIFO command faster than the instrument is acquiring it, then the maximum acquisition length becomes limited by the software and not by the length of the instrument's memory.

To use this mode of acquisition, issue commands in the following order:

a) Send CBD with a parameter specifying what outputs are to be stored.

b) Send STR with a parameter specifying the data recording interval.

c) Send LEN with a parameter equal to the required total curve sets length. This is the required recording time divided by the recording interval.

d) Start acquisition by sending TD. Triggered mode acquisition using TDT 0 and TDT 1 is also supported.

e) Send the M command and monitor the fourth value returned by it. When this exceeds a suitable threshold value (e.g. 100 curves), send DCBFIFO with a parameter equal to this threshold. Read and parse the resulting data from the instrument.

f) Repeat step e) for as long as required, until the acquisition is stopped when the number of curves equals the LEN command parameter, or an HC command is sent to the instrument.

GET [n] Trigger/Monitor Snap Data Acquisition

When sent with a parameter, the GET command functions according to the following table:-

n	function
0	Release hold on data transfer to output buffers within the model 7210
1	Suspend data transfer to output buffers within the model 7210

GET 1 is equivalent to sending the GPIB command Group Execute Trigger.

When sent without a parameter, the response is either 0 or 1, where the value has the following significance:

n	meaning
0	Data is being transferred to output buffers within the model 7210 - normal mode
1	Data transfer to output buffers within the model 7210 suspended, because either a GET 1 command or a GPIB Group Execute Trigger command has been received

In normal operation X1, Y1, X2 and Y2 output data is read from the output buffers at the point at which a command requiring it is received. However, this potentially introduces problems when multiple instruments are in use, since there is a finite delay between sending the relevant commands to, and processing the responses from, each instrument.

The instruments therefore respond to the GPIB group execute trigger which, when sent, triggers all units on a GPIB bus. When a GPIB group execute trigger command is received by the 7210, the data transfer to the instrument's output buffers is suspended, so that the buffers in a group of interconnected units all contain output values sampled at the same point in time.

The user's program should then read the required data sequentially from all the units, and once this has been done, release the data transfer hold within each instrument by sending a GET 0 command to each unit in turn.
Note that when the data transfer within the instrument is suspended, it still continues to operate normally, so there the maximum delay once the hold is removed before the output values are "correct" is only 2 ms.

5.3.07 Computer Interfaces (RS232 and GPIB)

DD [n] Define delimiter control

The value of n, which can be set to 13 or from 32 to 125, determines the ASCII value of the character sent by the lock-in amplifier to separate two numeric values in a two-value response, such as that generated by the XY1 (X and Y first-stage outputs) command. The default setting is a comma (equivalent to sending a DD 44 command).

GP [n] Set/Read GPIB parameters

n	Terminator
0	[CR], test echo disabled
1	[CR], test echo enabled
2	[CR,LF], test echo disabled
3	[CR,LF], test echo enabled
4	no terminator, test echo disabled
5	no terminator, test echo enabled

Note that the GPIB address is set via the rear-panel DIP switches.

RS [n_1 [n_2]] Set/read RS232 interface parameters

n_1	Baud rate (bits per second)
0	75
1	110
2	134.5
3	150
4	300
5	600
6	1200
7	1800
8	2000
9	2400
10	4800
11	9600
12	19200

The lowest five bits in n_2 control the other RS232 parameters according to the following table:

bit number	bit negated	bit asserted
0	data = 7 bits	data = 8 bits
1	no parity bit	1 parity bit
2	even parity	odd parity
3	echo disabled	echo enabled
4	prompt disabled	prompt enabled

ST Report status byte

Causes the lock-in amplifier to respond with the status byte, an integer between 0 and 255, which is the decimal equivalent of a binary number with the following bit-significance:

Bit 0	Command complete
Bit 1	Invalid command
Bit 2	Command parameter error
Bit 3	Reference unlock
Bit 4	Input or Output Overload
Bit 5	GPIB Group Execute Trigger or instrument GET 1 command received - Data output buffer update suspended until GET 0 command is sent.
Bit 6	Asserted SRQ
Bit 7	Data available

If Bit 4 is asserted, the OVL command can be used to determine where the overload is occurring.

NOTE: this command is not normally used in GPIB communications, where the status byte is accessed by performing a serial poll.

MSK [n] Set/read service request mask byte
The value of n sets the SRQ mask byte in the range 0 to 255

5.3.08 Instrument Identification

ID Identification
Causes the lock-in amplifier to respond with the number 7210.

CARDID Signal Board Identification
The response to the CARDID command is 8 numbers, separated by the selected delimiter. Each number indicates the type of signal input mounted the relevant slot, with the slots counted from the left-hand end (i.e. Channel 1). The value of each number has the following significance:

n	Significance
0	No card fitted
1	Wide Bandwidth current mode input (10E6 transimpedance)
2	Voltage Mode input
3	Low Noise current mode input (10E7 transimpedance)

For example a response of 3,3,3,3,3,3,3,3 means that the unit is fitted with 32 channels, each having a low-noise current mode input stage.

SLAVE Read Master/Slave status
The response to this command is a number with the following significance:

n	Significance
0	Unit is Master
1	Unit is Slave

Only one instrument in a group of interconnected units can be the Master.

VER Report firmware version
Causes the lock-in amplifier to respond with the firmware version number.

5.3.09 Auto Default

ADF n Auto Default
The ADF command performs an auto-default operation according to the following table:

n	Significance
0	All instrument settings are returned to their factory default values
1	All instrument settings, with the exception of the communications settings, are returned to their factory default values

NOTE: If the ADF 0 command is used when the communications settings are at values other than their default settings, then communication will be lost.

5.4 Programming Examples

5.4.01 Introduction

This section gives some examples of the commands that need to be sent to the lock-in amplifier for typical experimental situations.

5.4.02 Basic Signal Recovery - Single Reference Mode

In a typical simple experiment, the computer is used to set the instrument controls and then to record the chosen outputs, perhaps as a function of time. At sampling rates of up to a few points per second, there is no need to suspend data transfer to the internal curve buffer. The commands to achieve this, assuming that signals are applied to all 32 channels, would therefore be similar to the following sequence:

REFMODE 0	Single reference mode
REFN1 1	1F detection
ASM 0	Auto-Measure all 32 channels. Note that this could take a long time, up to several minutes, to complete
TC1 0 3	Set time constants to 100 ms, since previous ASM sets the TC to 300 ms

Then the outputs could be read as follows:

X1. 0	Reads 32 X1 channel output values in amps
Y1. 0	Reads 32 Y1 channel output values in amps
FRQ1	Reads reference frequency in hertz

The controlling program would send a new output command each time a new reading were required. Note that with the output filter slope of 12 dB/octave a good "rule of thumb" is to wait for a period of four time-constants after the input signal has changed before recording a new value. Hence in a scanning type experiment, the program should issue the commands to whatever equipment causes the input signal to the lock-in amplifier to change, wait for four time-constants, and then record the required output.

5.4.03 Basic Signal Recovery - Tandem Reference Mode

This is similar to the procedure described in section 5.4.02 above, except that the instrument is now set to the tandem reference mode. This requires that the REF 2 OUT frequency be set correctly and that the OFFSET command be used to remove the effect of any interdemodulator DC offset.

The commands to achieve this, assuming that signals are applied to all 32 channels, would therefore be similar to the following sequence:

REFMODE 1	Tandem reference mode
FRQ2. 1.0E1	Set REF 2 OUT to nominally 10 Hz
ASM 0	Auto-Measure both demodulators for all 32 channels. Note that

	this could take a long time, up to several minutes, to complete.
TC1 0 3	Set time constants to 100 ms, since previous ASM sets the TC to 300 ms changed them
TC2 0 3	Set time constants to 100 ms, since previous ASM will have set the TC as a function of the REF 2 OUT reference frequency.

Then the outputs could be read as follows:

X1. 0	Reads 32 X1 channel output values in amps
Y1. 0	Reads 32 Y1 channel output values in amps
X2. 0	Reads 32 X2 channel output values in amps
Y2. 0	Reads 32 Y2 channel output values in amps
FRQ1	Reads applied reference frequency in hertz
FRQ2.	Reads REF 2 OUT reference frequency in hertz

5.4.04 Tandem Reference Mode with Output Data Sampling Correlation

This is the same as example 5.4.03, except two instruments are now used and the GPIB Group Execute Trigger command is used to synchronize the output data sampling time for both of them.

The commands to achieve this, assuming that signals are applied to all 32 channels, would therefore be similar to the following sequence:

Send to the Master Unit:

REFMODE 1	Tandem reference mode
FRQ2. 1.0E1	Set REF 2 OUT to nominally 10 Hz
ASM 0	Auto-Measure both demodulators for all 32 channels. Note that this could take a long time, up to several minutes, to complete.
TC1 0 3	Set time constants to 100 ms, since previous ASM sets the TC to 300 ms changed them
TC2 0 3	Set time constants to 100 ms, since previous ASM will have set the TC as a function of the REF 2 OUT reference frequency.

and then send to the Slave Unit:

REFMODE 1	Tandem reference mode
ASM 0	Auto-Measure both demodulators for all 32 channels. Note that this could take a long time, up to several minutes, to complete.
TC1 0 3	Set time constants to 100 ms, since previous ASM sets the TC to 300 ms changed them
TC2 0 3	Set time constants to 100 ms, since previous ASM will have set the TC as a function of the REF 2 OUT reference frequency.

Now send a GPIB Group Execute Trigger.

Then read the required outputs from the master and slave units using the following commands:-

X1. 0	Reads 32 X1 channel output values in amps
Y1. 0	Reads 32 Y1 channel output values in amps
X2. 0	Reads 32 X2 channel output values in amps
Y2. 0	Reads 32 Y2 channel output values in amps

Having read all required data, send to the Master Unit:-

GET 0

and to the Slave Unit:-

GET 0

before repeating as required.

Specifications

Measurement Modes

Single-frequency 32 channel dual-phase lock-in amplifier, running with an external reference frequency in the range 20 Hz to 50.5 kHz

Outputs:-
- X1
- Y1

Tandem-operation 32 channel dual-phase lock-in amplifier, running with a first, external reference frequency in the range 20 Hz to 50.5 kHz and generating the second reference frequency by integer division of the first. The range of the of second frequency is 0.001 Hz to 100 Hz, subject to a maximum frequency equal to one half of the applied reference frequency.

Outputs:-
- X1
- Y1

of the modulation at the first reference frequency (e.g. at say 50 kHz),

and
- X2
- Y2

of the first reference frequency by the second. (e.g. the components at say 10 Hz)

Signal Channel

The signal input specifications depend on the model of signal board fitted. Three board types are available:

7210/99 Signal Board - Voltage Mode Inputs

Voltage	
Mode	Virtual Ground
Connector	BNC
Impedance to Ground	0 Ω
Input Impedance	10 MΩ
Input Voltage Noise	< 10 nV/√Hz at 1 kHz
Max Safe Input	± 12.0 V
Frequency Response	20 Hz to 51 kHz
Frequency Response over which following four specifications apply	110 Hz to 51 kHz
Gain Accuracy Overall	± 0.5% to ± 2.0%
Gain Match between Channels	± 1.0% to ± 5.0%
Phase Accuracy Overall	± 2°
Phase Match between Channels	± 1°
Full-scale sensitivity	100 µV to 1 V rms in a 1-3-10 sequence

	(9 settings)

7210/98 Signal Card - Wide Bandwidth Current Mode Inputs

Current Input	
Mode	Virtual Ground
Connector	BNC
Impedance to Ground	0 Ω
Input Impedance	≤ 1 kΩ at 1 kHz to virtual ground
Input Current Noise	< 150 fA/√Hz at 1 kHz
Max Safe Input	± 12.0 V
Frequency Response	20 Hz to 51 kHz
Frequency Response over which following four specifications apply	110 Hz to 51 kHz
Gain Accuracy Overall	± 0.5% to ± 2.0%
Gain Match between Channels	± 1.0% to ± 5.0%
Phase Accuracy Overall	± 2°
Phase Match between Channels	± 1°
Full-scale sensitivity	100 pA to 1 µA rms in a 1-3-10 sequence (9 settings)

7210/97 Signal Card - Low Noise Current Mode Inputs

Current Input	
Mode	Virtual Ground
Connector	BNC
Impedance to Ground	0 Ω
Input Impedance	≤ 1 kΩ at 1 kHz to virtual ground
Input Current Noise	< 50 fA/√Hz at 1 kHz
Max Safe Input	± 12.0 V
Frequency Response	20 Hz to 5 kHz
Frequency Response over which following four specifications apply	110 Hz to 5 kHz
Gain Accuracy Overall	± 0.5% to ± 2.0%
Gain Match between Channels	± 1.0% to ± 5.0%
Phase Accuracy Overall	± 2°
Phase Match between Channels	± 1°
Full-scale sensitivity	10 pA to 100 nA rms in a 1-3-10 sequence (9 settings)

Reference Channel

External Reference Input	
Impedance	1 MΩ//35 pF
Level	250 mV to 2.5 V rms
Connector	BNC
Frequency Range, f1	20 Hz to 50.5 kHz
Lock Acquisition Time	2 seconds max
Reference Phase Shifter	
Set Resolution	10 m°
Orthogonality	90° ± 0.001°
External Reference Frequency Meter Resolution	1 Hz
Reference Output	
Frequency, f2	f1/n, where n, an integer, is calculated by the instrument to give a frequency as close as possible to user- specified value in the
range	0.1 Hz to 100 Hz; maximum frequency also limited to be no greater than f1/2
Amplitude	> 3 V pk-pk square-wave
Impedance	< 200 Ω
Connector	BNC
Harmonic Detection	f and 2f (2f in single -frequency operation only 2f < 50.5 kHz
Tandem Reference Frequency Meter Resolution	0.001 Hz

Demodulator

ADC’s	
Type	12 bit or better
Sampling Rate	
f1 ≥ 200Hz	208 kHz < f_s , 250 kHz
f1 < 200Hz	
	both rates synchronous to external reference (f1) frequency
Single-Frequency Operation	
Time Constants	3 ms to 1 ks in 1-3-10 sequence (12 steps), plus 2 ms
Slope	12 dB/octave
Type	Synchronous digital FIR filters
Harmonic Rejection	> 90 dB
Dynamic Reserve	> 80 dB

Tandem-Frequency Operation
Applying to f1 outputs:-

Time Constants	4 ms to 1 ks in 1-3-10 sequence (12 steps)
Slope	12 dB/octave
Type	Synchronous digital FIR filters

Applying to f2 outputs:-

Time Constants	30 ms to 1 ks in 1-3-10 sequence (11 steps)
Slope	12 dB/octave
Type	Synchronous digital FIR filters
Harmonic Rejection	> 90 dB
Dynamic Reserve	> 80 dB

Data Outputs

The outputs available from the instrument are:-

Single Reference Mode:
X1
Y1

Tandem Mode:
X1
Y1
X2
Y2

for each of 32 channels. Outputs can be read directly on receipt of a command, or stored on receipt of a GPIB trigger or the GET command for later readout. The output values can be read using commands generating binary or ASCII responses.

Interconnections

Up to sixteen instruments can be interconnected to provide up to 512 detection channels. Interconnections are via RG45 multipole connectors. Each instrument has a rear-panel switch to select whether the connectors function as outputs, in which case the unit is the "master", or inputs, when the unit is a "slave".

Indicators

Front-panel LEDs indicate the following conditions:-

Power On - a single LED which is lit when line power is switched on.

Communications Activity - indicates when command is being received and response waiting to be read or being transmitted.

Master/Slave - when lit indicates that the instrument is set to function as a "master" and that its synchronizing signal connectors are outputs

Internal Oscillator - reserved for future expansion.

Reference Unlock - lights when no suitable reference is applied

Signal Channel Overload - a single LED warning of any one of the 32 channels being in overload. It is possible to identify via a computer status command which of the channels is in overload.

General

Computer Interfaces	
Type	GPIB (IEEE-488) and RS232-C
Connectors	Standard GPIB Centronics connector, 9-pin female RS232
Comms Settings	Set by rear-panel DIP switches
Command Set	ASCII commands for all instrument controls and data readout. Binary dump commands for data readout
Power Requirements	
Voltage	100/120/220/240 V AC
Frequency	50 - 60 Hz
Power	200VA max
Dimensions	
Width	446 mm
Height	3U (133.5 mm)
Depth	435 mm
Weight	12.5 kg

Pinouts

Appendix B

B1 RS232 Connector Pinout

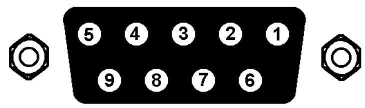

Figure B-1, RS232 and AUX RS232 Connector (Female)

Pin	Function	Description
2	RXD	Data In
3	TXD	Data Out
5	GND	Signal Ground
7	RTS	Request to Send - Always +12 V

All other pins are not connected

B.2 Digital Output Port Connector

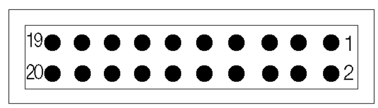

Figure B-2, Digital Output Port Connector

8-bit TTL-compatible output set from the front panel or via the computer interfaces. Each line can drive 3 LSTTL loads. The connector will mate with a 20-pin IDC header plug (not supplied). The pinout is as follows:-

Pin	Function
1	Ground
2	Ground
3	D0
4	Ground
5	D1
6	Ground
7	D2
8	Ground
9	D3
10	Ground
11	D4
12	Ground
13	D5
14	Ground
15	D6
16	Ground

Pin	Function
17	D7
18	Ground
19	+5 V
20	+5 V

D0 = Least Significant Bit
D7 = Most Significant Bit

B.3 Preamplifier Power Connector Pinout

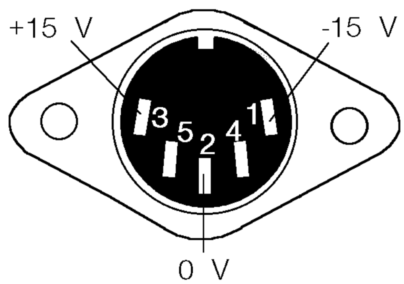

Figure B-3, Preamplifier Power Connector

Pin	Function
1	-15 V
2	Ground
3	+15 V

Pins 4 and 5 are not connected. Shell is shield ground.

Demonstration Programs

Appendix C

C.1 Simple Terminal Emulator

This is a short terminal emulator with minimal facilities, which will run on a PC-compatible computer in a Microsoft GWBASIC or QuickBASIC environment, or can be compiled with a suitable compiler.

```
10 'MINITERM 9-Feb-96
20 CLS : PRINT "Lockin RS232 parameters must be set to 9600 baud, 7 DATA
   bits, 1 stop bit and even parity"
30 PRINT "Hit <ESC> key to exit"
40 OPEN "COM1:9600,E,7,1,CS,DS" FOR RANDOM AS #1
50 '..............................
60 ON ERROR GOTO 180
70 '..............................
100 WHILE (1)
110     B$ = INKEY$
120     IF B$ = CHR$(27) THEN CLOSE #1: ON ERROR GOTO 0: END
130     IF B$ <> "" THEN PRINT #1, B$;
140     LL% = LOC(1)
150     IF LL% > 0 THEN A$ = INPUT$(LL%, #1): PRINT A$;
160 WEND
170 '..............................
180 PRINT "ERROR NO."; ERR: RESUME
```

C.2 RS232 Control Program with Handshakes

RSCOM2.BAS is a user interface program which illustrates the principles of the echo handshake. The program will run on a PC-compatible computer either in a Microsoft GWBASIC or QuickBASIC environment, or in compiled form.

The subroutines in RSCOM2 are recommended for incorporation in the user's own programs.

```
10 'RSCOM2 9-Feb-96
20 CLS : PRINT "Lockin RS232 parameters must be set to 9600 baud, 7 data
   bits, 1 stop bit, even parity"
30 OPEN "COM1:9600,E,7,1,CS,DS" FOR RANDOM AS #1
40 CR$ = CHR$(13)                       ' carriage return
50 '
60 '...main loop.................
70 WHILE 1                              ' infinite loop
80     INPUT "command (00 to exit) "; B$ ' no commas are allowed in B$
90     IF B$ = "00" THEN END
100     B$ = B$ + CR$                   ' append a carriage return
110     GOSUB 180                       ' output the command B$
120     GOSUB 310: PRINT Z$;            ' read and display response
130     IF A$ = "?" THEN GOSUB 410: GOSUB 470 ' if "?" prompt fetch
   STATUS%
140                                      ' and display message
150 WEND                                ' return to start of loop
```

```
160 '
170 '
180 '...output the string B$..............
190 ON ERROR GOTO 510                    ' enable error trapping
200 IF LOC(1) > 0 THEN A$ = INPUT$(LOC(1), #1) ' clear input buffer
210 ON ERROR GOTO 0                      ' disable error trapping
220 FOR J1% = 1 TO LEN(B$)               ' LEN(B$) is number of bytes
230     C$ = MID$(B$, J1%, 1): PRINT #1, C$;  ' send byte
240     WHILE LOC(1) = 0: WEND           ' wait for byte in input buffer
250     A$ = INPUT$(1, #1)                ' read input buffer
260     IF A$ <> C$ THEN PRINT "handshake error"' input byte should be
   echo
270 NEXT J1%                             ' next byte to be sent or
280 RETURN                               ' return if no more bytes
290 '
300 '
310 '....read response..................
320 A$ = "": Z$ = ""
330 WHILE (A$ <> "*" AND A$ <> "?")    ' read until prompt received
340     Z$ = Z$ + A$                     ' append next byte to string
350     WHILE LOC(1) = 0: WEND           ' wait for byte in input buffer
360     A$ = INPUT$(1, #1)                ' read byte from buffer
370 WEND                                 ' next byte to be read
380 RETURN                               ' return if it is a prompt
390 '
400 '
410 '....fetch status byte..............
420 B$ = "ST" + CR$                      ' "ST" is the status command
430 GOSUB 180                            ' output the command
440 GOSUB 310                            ' read response into Z$
450 STATUS% = VAL(Z$)                    ' convert to integer
460 RETURN
470 '....instrument error message.......
480 PRINT "Error prompt, status byte = "; STATUS% ' bits are defined in
   manual
490 PRINT
500 RETURN
510 '....I/O error routine..............
520 RESUME
```

C.3 GPIB User Interface Program

GPCOM.BAS is a user interface program which illustrates the principles of the use of the serial poll status byte to coordinate the command and data transfer.

The program runs under Microsoft GWBASIC or QuickBASIC on a PC-compatible computer fitted with a National Instruments IEEE-488 interface card and the GPIB.COM software installed in the CONFIG.SYS file. The program BIB.M, and the first three lines of GPCOM, are supplied by the card manufacturer and must be the correct version for the particular version of the interface card in use. The interface card may be set up, using the program IBCONF.EXE, to set EOI with the last byte of Write in which case no terminator is required. (Read operations are automatically terminated on EOI which is always sent by the lock-in amplifier). Normally, the options called 'high-speed timing', 'interrupt jumper setting', and 'DMA channel' should all be disabled.

The principles of using the Serial Poll Status Byte to control data transfer, as implemented in the main loop of GPCOM, are recommended for incorporation in the user's own programs.

```
10 'GPCOM 9-Feb-96
20 '....the following three lines and BIB.M are supplied by the.......
30 '....manufacturer of the GPIB card, must be correct version........
40 CLEAR , 60000!: IBINIT1 = 60000!: IBINIT2 = IBINIT1 + 3: BLOAD
   "BIB.M", IBINIT1
50 CALL IBINIT1 (IBFIND, IBTRG, IBCLR, IBPCT, IBSIC, IBLOC, IBPPC, IBBNA,
   IBONL, IBRSC, IBSRE, IBRSV, IBPAD, IBSAD, IBIST, IBDMA, IBEOS, IBTMO,
   IBEOT, IBRDF, IBWRTF, IBTRAP)
60 CALL IBINIT2 (IBGTS, IBCAC, IBWAIT, IBPOKE, IBWRT, IBWRTA, IBCMD,
   IBCMDA, IBRD, IBRDA, IBSTOP, IBRPP, IBRSP, IBDIAG, IBXTRC, IBRDI,
   IBWRTI, IBRDIA, IBWRTIA, IBSTA%, IBERR%, IBCNT%)
70 '.................................................
80 CLS : PRINT "DEVICE MUST BE SET TO CR TERMINATOR"
90 '....assign access code to interface board........
100 BDNAME$ = "GPIB0"
110 CALL IBFIND(BDNAME$, GPIB0%)
120 IF GPIB0% < 0 THEN PRINT "board assignment error": END
130 '....send INTERFACE CLEAR.........................
140 CALL IBSIC(GPIB0%)
150 '....set bus address, assign access code to device..........
160 SUCCESS% = 0
170 WHILE SUCCESS% = 0
180     INPUT "BUS ADDRESS "; A%
190     DEVNAME$ = "DEV" + RIGHT$(STR$(A%), LEN(STR$(A%)) - 1)
200     CALL IBFIND(DEVNAME$, DEV%)    ' assign access code
210     IF DEV% < 0 THEN PRINT "device assignment error": END
220     A$ = CHR$(13): GOSUB 480       ' test: write <CR> to bus
230     IF IBSTA% > 0 THEN SUCCESS% = 1
240     IF (IBSTA% < 0 AND IBERR% = 2) THEN BEEP: PRINT "NO DEVICE AT
   THAT ADDRESS ";
250 WEND
260 '....send SELECTED DEVICE CLEAR...................
270 CALL IBCLR(DEV%)
280 '....set timeout to 1 second......................
290 V% = 11: CALL IBTMO(DEV%, V%)
```

```
300 '....set status print flag.........................
310 INPUT "Display status byte y/n "; R$
320 IF R$ = "Y" OR R$ = "y" THEN DS% = 1 ELSE DS% = 0
330 '....main loop....................................
340 WHILE 1                            ' infinite loop
350     INPUT "command (00 to exit) "; A$
360     IF A$ = "00" THEN END
370     A$ = A$ ' CHR$(13)             ' terminator is <CR>
380     GOSUB 480                      ' write A$ to bus
390     S% = 0                         ' initialize S%
400     WHILE (S% AND 1) = 0           ' while command not complete
410             GOSUB 530              ' serial poll, returns S%
420             IF DS% THEN PRINT "S%= "; S%
430             IF (S% AND 128) THEN GOSUB 500: PRINT B$  ' read bus into
   B$ and print
440     WEND
445           IF (S% AND 4) THEN PRINT "parameter error"
450     IF (S% AND 2) THEN PRINT "invalid command"
460 WEND
470 '....end of main loop.............................
480 '....write string to bus..........................
490 CALL IBWRT(DEV%, A$): RETURN
500 '....read string from bus.........................
510 B$ = SPACE$(32)                          ' B$ is buffer
520 CALL IBRD(DEV%, B$): RETURN
530 '......serial poll................................
540 CALL IBRSP(DEV%, S%): RETURN
```

Cable Diagrams

Appendix **D**

D.1 RS232 Cable Diagrams

Although in normal use the 7210 is controlled via the GPIB interface, it is useful for diagnostic purposes and necessary for firmware upgrades to be able use the RS232 interface as well. In order to connect the instrument to a standard serial port on a computer one of two types of cable is needed. The only difference between them is the number of pins used on the connector which goes to the computer. One has 9 pins and the other 25; both are null-modem (also called modem eliminator) cables in that some of the pins are cross-connected.

Users with reasonable practical skills can easily assemble the required cables from parts which are widely available through computer stores and electronics components suppliers. The required interconnections are given in figures D-1 and D-2.

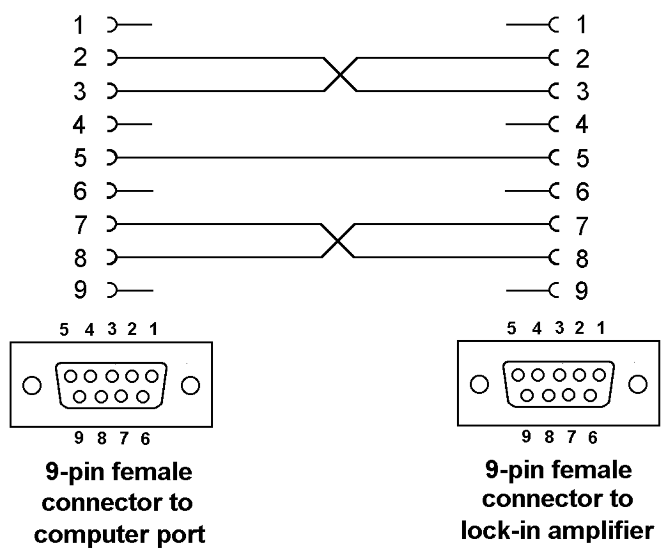

Figure D-1, Interconnecting RS232 Cable Wiring Diagram

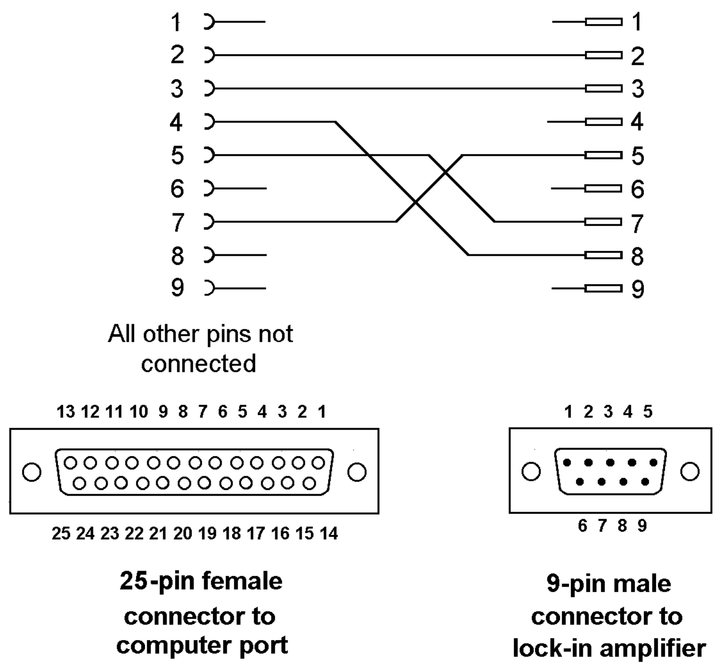

Figure D-2, Interconnecting RS232 Cable Wiring Diagram

Alphabetical Listing of Commands

Appendix E

In the following commands the parameter n_1 is commonly used to signify which channel(s) of the 32 within the instrument will be affected by the command, as follows:-

n_1	Significance
0	All 32 channels are set to the same value
1	Channel 1
2	Channel 2
...	
31	Channel 31
32	Channel 32

For example, AQN 0 will perform an auto-phase operation on all 32 channels, but AQN 5 will perform it only on Channel 5.

Commands that elicit a response where n_1 is equal to 0 generate 32 response values in the order Channel 1 to Channel 32. In the case of the RS232 interface, each response is terminated with a carriage return/line feed pair, while when using the GPIB interface the final character of each response is indicated by the GPIB line EOI being asserted.

ACGAIN n_1 [n_2] AC Gain control

Sets or reads the gain of the signal channel amplifier. Values of n_2 from 0 to 6 can be entered, corresponding to the range 0 dB to 60 dB in 10 dB steps.

ADF n Auto Default

The ADF command performs an auto-default operation according to the following table:

n	Significance
0	All instrument settings are returned to their factory default values
1	All instrument settings, with the exception of the communications settings, are returned to their factory default values

NOTE: If the ADF 0 command is used when the communications settings are at values other than their default settings, then communication will be lost.

AQN1 n_1 Auto-Phase (auto quadrature null)

Main (first stage) demodulator(s)

AQN2 n_1 Auto-Phase (auto quadrature null)

Tandem (second stage) demodulator(s)

AS1 n_1 Perform an Auto-Sensitivity operation

Main (first stage) demodulator(s)

AS2 n_1 Perform an Auto-Sensitivity operation

Tandem (second stage) demodulator(s)

ASM n_1 Perform an Auto-Measure operation

Operates on both the main (first stage) and tandem (second stage) demodulator(s)

Note: The auto-sensitivity and auto-measure operations can take a significant time (in some cases over a minute) per channel to complete, so they should be used with care.

AUTOMATIC n_1 [n_2] AC Gain automatic control

n_2 Status

0 AC Gain is under manual control via the ACGAIN command

1 Automatic AC Gain control is activated, with the gain being adjusted according to the full-scale sensitivity setting

BYTE [n] Digital port output control

The value of n, in the range 0 to 255, determines the bits to be output on the rear panel digital output port. Hence, for example, if BYTE = 0, all outputs are low, and when BYTE = 255, all are high.

BX1 X channel output - Main (first stage) - Binary Mode

This command causes the lock-in amplifier to respond with 64 bytes of data being the X1 demodulator output for all 32 channels. Each value is a sixteen-bit signed integer in the range ±30000, full-scale being ±10000, represented by two bytes in the order High Byte - Low Byte, with the data being transferred in the order Channel 1 to Channel 32.

BY1 Y channel output - Main (first stage) - Binary Mode

This command causes the lock-in amplifier to respond with 64 bytes of data being the Y1 demodulator output for all 32 channels. Each value is a sixteen-bit signed integer in the range ±30000, full-scale being ±10000, represented by two bytes in the order High Byte - Low Byte, with the data being transferred in the order Channel 1 to Channel 32.

BX2 X channel output - Tandem (second stage) - Binary Mode

This command causes the lock-in amplifier to respond with 64 bytes of data being the X2 demodulator output for all 32 channels. Each value is a sixteen-bit signed integer in the range ±30000, full-scale being ±10000, represented by two bytes in the order High Byte - Low Byte, with the data being transferred in the order Channel 1 to Channel 32.

BY2 Y channel output - Tandem (second stage)- Binary Mode

This command causes the lock-in amplifier to respond with 64 bytes of data being the Y2 demodulator output for all 32 channels. Each value is a sixteen-bit signed integer in the range ±30000, full-scale being ±10000, represented by two bytes in the order High Byte - Low Byte, with the data being transferred in the order Channel 1 to Channel 32.

CARDID Signal Board Identification

The response to the CARDID command is 8 numbers, separated by the selected delimiter. Each number indicates the type of signal input mounted the relevant slot, with the slots counted from the left-hand end (i.e. Channel 1). The value of each number has the following significance:

n Significance

0 No card fitted
1 Wide Bandwidth current mode input (10E6 transimpedance)
2 Voltage Mode input
3 Low Noise current mode input (10E7 transimpedance)

For example a response of 3,3,3,3,3,3,3,3 means that the unit is fitted with 32 channels, each having a low-noise current mode input stage.

CBD [n] Curve buffer define

Defines which data outputs are stored in the curve buffer when subsequent TD (take data), TDT (take data triggered) or TDC (take data continuously) commands are issued.

Up to eight curve sets may be stored. Each set either consists of single values per curve, or an array of 32 values representing a full set of 32 output readings defining a curve.

The actual curve sets that will be saved are specified by the CBD parameter, which is an integer between 1 and 247, being the decimal equivalent of an 8-bit binary word. When a given bit is asserted, the corresponding output is selected for storage. When a bit is negated, the output is not stored. The bit function and range for each output are shown in the table below:

Bit	Decimal value	Values/Curve Point	Output and range
0	1	32	X1 Outputs (±10000 FS)
1	2	32	Y1 Outputs (±10000 FS)
2	4	1	Ref 1 frequency (Hz)
3	8	-	Reserved for future use
Tandem Mode Only (i.e. REFMODE = 1)			
4	16	32	X2 Outputs (±10000 FS)
5	32	32	Y2 Outputs (±10000 FS)
6	64	1	Ref 2 frequency bits 0 to 15 (mHz)
7	128	1	Ref 2 frequency bits 16 to 32 (mHz)

Curve sets 6 and 7 store the reference frequency F2 in millihertz. The calculation needed to translate the two 16-bit values to one 32-bit value is:
Reference Frequency F2 = (65536 × value in Curve 7) + (value in Curve 6)

Note that the CBD command directly determines the allowable parameters for the DC and DCB commands. It also interacts with the LEN command and affects the values reported by the M and MAXLEN commands.

No provision is made for storing the instrument's 32 input channel sensitivities when using the curve buffer, so these should not be changed during acquisition since it would not then be possible to correctly interpret the stored X1, Y1, X2 and Y2 values.

DC n Dump acquired curve to computer

This command causes stored curve sets to be dumped via the computer interface in ASCII decimal format.

One curve set only is transferred, as specified by the parameter n, which is the bit number of the required set and which must have been stored by the most recent CBD command. Data is transmitted one curve at a time, starting from the beginning of the buffer. In the case of curves with 32 Values (i.e. curves 0, 1, 4 and 5) the response consists of a total of 32 × (setting of the LEN control) values, ordered as follows (example for curve set 0, with length of 100 curves):

$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 1 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 2 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 3 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 4 in curve set)
.....
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 99 in curve set)
$X1_1$, $X1_2$, $X1_3$, $X1_4$, $X1_5$, $X1_6$, $X1_7$, ..., $X1_{31}$, $X1_{32}$ (curve 100 in curve set)

Each value within a curve is separated from the next by the delimiter that is specified by the DD command - the default delimiter on power-up being a comma (DD 44). Each curve is separated from the next by a <CR><LF> (carriage return and line feed character pair).

In the case of the other curve sets, the command simply returns (setting of the LEN control) values, again with each one being separated from the next by a <CR><LF> character pair.

The computer program's subroutine which reads the responses to the DC command needs to run a FOR...NEXT loop of appropriate length, storing the responses as required.

Note that when using this command with the GPIB interface the serial poll must be used. After sending the DC command, perform repeated serial polls until bit 7 is set, indicating that the instrument has an output waiting to be read. Then perform repeated reads in a loop, waiting each time until bit 7 is set indicating that a new value is available. The loop should continue until bit 1 is set, indicating that the transfer is completed.

DCFIFO n Dump acquired curve to computer - FIFO mode

This command causes stored data curves to be dumped via the computer interface in ASCII decimal format. Unlike the DC command, it can be used while data acquisition is in progress, thereby making the memory within the instrument essentially function as a first-in, first-out (FIFO) buffer.

The parameter n defines the number of curve sets to transfer, and can be any value between 1 and the present number of curve sets held in the buffer. The latter can be determined by using the M command and reading the fourth value returned by it.

Because of the way that data is organized in the buffer, the DCFIFO command returns all data that has been stored, as defined by the CBD command (unlike the DC command there is no parameter to specify a particular curve set to transfer). The user’s software must therefore parse the data returned by the command and put it into suitable arrays for further processing or storage.

For each curve set transferred, data is transmitted in the following sequence:

$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (X1 values)
$Y1_1, Y1_2, Y1_3, Y1_4, Y1_5, Y1_6, Y1_7, ..., Y1_{31}, Y1_{32}$ (Y1 values)
FRQ1 (Ref 1 frequency (Hz))
$X2_1, X2_2, X2_3, X2_4, X2_5, X2_6, X2_7, ..., X2_{31}, X2_{32}$ (X2 values)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, ..., Y2_{31}, Y2_{32}$ (Y2 values)
$FRQ2_{LSB}$ (Ref 2 frequency bits 0 to 15 (mHz))
$FRQ2_{MSB}$ (Ref 2 frequency bits 0 to 15 (mHz))

subject to the condition that the relevant data curves must have been stored, as specified by the CBD command. Hence, for example, if X1 and Y2 curve sets were stored (CBD = 33) then the responses would be in the following order:

$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 1)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, ..., Y2_{31}, Y2_{32}$ (curve 1)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 2)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, ..., Y2_{31}, Y2_{32}$ (curve 2)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 3)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, ..., Y2_{31}, Y2_{32}$ (curve 3)
..etc.

Each value within a curve is separated from the next by the delimiter that is specified by the DD command - the default delimiter on power-up being a comma (DD 44). Each curve is separated from the next by a <CR><LF> (carriage return and line feed character pair).

The computer program's subroutine which reads the responses to the DCFIFO command needs to run a FOR...NEXT loop of appropriate length, storing the responses as required.

Note that when using this command with the GPIB interface the serial poll must be used. After sending the DCFIFO command, perform repeated serial polls until bit 7 is set, indicating that the instrument has an output waiting to be read. Then perform repeated reads in a loop, waiting each time until bit 7 is set indicating that a new value is available. The loop should continue until bit 1 is set, indicating that the transfer is completed.

Each time a set of curves is transferred, the instrument decrements the number of curves available for transfer, as reported by the M command.

DCB n Dump acquired curve sets to computer in binary format

This command causes a stored curve set to be dumped via the computer interface in binary format, using two bytes per point to transfer each 16-bit 2's complement value, with the MSB transmitted first. The number of data bytes sent is therefore equal to 2 × (current curve length) × Values/Curve.

In order to achieve the maximum transfer rate, no terminators are used within the transmission, although the response is terminated normally at the end.

One curve only is transferred, as specified by the parameter n, which is the bit number of the required curve set and which must have been stored by the most recent

CBD command. Data is transmitted one curve at a time, starting from the beginning of the buffer. In the case of curves with 32 Values/Curve (i.e. curves 0, 1, 4 and 5) the response consists of a total of 32 $\times$ (setting of the LEN control) values, ordered as follows (example for curve 0, with length of 100 points):

$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 1 in curve set)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 2 in curve set)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 3 in curve set)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 4 in curve set)
.....
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 99 in curve set)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 100 in curve set)

In the case of the other curves, the command simply returns (setting of the LEN control) values.

The computer program's subroutine which reads the responses to the DCB command needs to be able to handle the potentially very large data blocks that can be generated. For example, transfer of an output curve set when the curve set length is 1000 points will generate 1000 $\times$ 2 $\times$ 32 = 64,000 bytes of data.

DCBFIFO n Dump acquired curve to computer - FIFO mode, binary output

This command is similar to the DCFIFO command, except that:

a) Data points are transferred in binary format, using two bytes per point to transfer each 16-bit 2’s complement value, with the MSB transmitted first. There are no delimiters between points, curves, or curve sets.

b) Operation when downloading data via the GPIB is a little more complicated, in order to get the fastest possible speeds. Data is transferred in blocks corresponding to eight curve sets, so long as their are still eight or more remaining to transfer. Once there are fewer than eight, then the transfer is for the number remaining.

For example, suppose CBD = 3, implying that the X1and Y1 output values have been stored, and that 100 curve sets are to be read. In this case, first send the command DCBFIFO 100. Next perform repeated serial polls until bit 7 is set, indicating that the instrument has an output waiting to be read. Then perform a read of (32 x 2 x 2 x 8) bytes (1024 bytes) which will contain the curve sets 1 to 8 of X1 and Y1 curves, in the following order:

$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 1)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, ..., Y2_{31}, Y2_{32}$ (curve 1)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 2)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, ..., Y2_{31}, Y2_{32}$ (curve 2)
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 3)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, ..., Y2_{31}, Y2_{32}$ (curve 3)
...
$X1_1, X1_2, X1_3, X1_4, X1_5, X1_6, X1_7, ..., X1_{31}, X1_{32}$ (curve 8)
$Y2_1, Y2_2, Y2_3, Y2_4, Y2_5, Y2_6, Y2_7, ..., Y2_{31}, Y2_{32}$ (curve 8)

Now perform another serial poll; bit 7 will still be set indicating that more data is available. Read this data with another read of 1024 bytes.

Repeat this process, tracking the number of curve sets received, until there are only four left (i.e. after 12 groups). The final read will then need to be of (32 x 2 x 2 x 4) bytes (256 bytes) to complete the transfer. At this point bit 1 in the serial poll byte will is set.

Each time a set of curves is transferred, the instrument decrements the number of curves available for transfer, as reported by the M command.

If the controlling software reads data out using the DCBFIFO command faster than the instrument is acquiring it, then the maximum acquisition length becomes limited by the software and not by the length of the instrument's memory.

DD [n] Define delimiter control

The value of n, which can be set to 13 or from 32 to 125, determines the ASCII value of the character sent by the lock-in amplifier to separate two numeric values in a two-value response, such as that generated by the XY1 (X and Y first-stage outputs) command. The default setting is a comma (equivalent to sending a DD 44 command).

FRQ1[.] Reference frequency meter

The FRQ1 command causes the lock-in amplifier to respond with 0 if the main (first stage) reference channel is unlocked, or with the reference input frequency if it is locked.

In both fixed and floating point mode the frequency is in Hz.

FRQ2[.] [n] Set/Read Reference output frequency

The value of n sets or reads the frequency of the REF OUT signal when the instrument is operating in Tandem mode. In fixed point mode, n is in millihertz, and in floating point mode, it is in hertz. Note that because of the finite number of possible frequency divisors, the requested value of n may not be the exact frequency generated. To confirm the exact frequency, send the required value of n as a write command FRQ2[.] n and then read the actual value with a write/read command FRQ2[.]

Once set, the frequency of the REF OUT signal remains at the same integer division of the applied reference frequency as long as the latter does not change by more than ± 100 Hz. Greater changes cause the reference channel to re-lock and the instrument to determine new divisor to give a REF OUT frequency as close as possible to that requested.

GET [n]] Trigger/Monitor Snap Data Acquisition

When sent with a parameter, the GET command functions according to the following table:-

n	function
0	Release hold on data transfer to output buffers within the model 7210
1	Suspend data transfer to output buffers within the model 7210

GET 1 is equivalent to sending the GPIB command Group Execute Trigger.

When sent without a parameter, the response is either 0 or 1, where the value has the following significance:

n	meaning
0	Data is being transferred to output buffers within the model 7210 - normal mode
1	Data transfer to output buffers within the model 7210 suspended, because either a GET 1 command or a GPIB Group Execute Trigger command has been received

In normal operation X1, Y1, X2 and Y2 output data is read from the output buffers at the point at which a command requiring it is received. However, this potentially introduces problems when multiple instruments are in use, since there is a finite delay between sending the relevant commands to, and processing the responses from, each instrument.

The instruments therefore respond to the GPIB group execute trigger which, when sent, triggers all units on a GPIB bus. When a GPIB group execute trigger command is received by the 7210, the data transfer to the instrument's output buffers is suspended, so that the buffers in a group of interconnected units all contain output values sampled at the same point in time.

The user's program should then read the required data sequentially from all the units, and once this has been done, release the data transfer hold within each instrument by sending a GET 0 command to each unit in turn.

Note that when the data transfer within the instrument is suspended, it still continues to operate normally, so there the maximum delay once the hold is removed before the output values are "correct" is only 2 ms.

GP [n]] Set/Read GPIB parameters

n_2	Terminator
0	[CR], test echo disabled
1	[CR], test echo enabled
2	[CR,LF], test echo disabled
3	[CR,LF], test echo enabled
4	no terminator, test echo disabled
5	no terminator, test echo enabled

Note that the GPIB address is set via the rear-panel DIP switches.

HC Halt curve acquisition

Halts curve acquisition in progress. It is effective during both single (data acquisition initiated by TD command) and continuous (data acquisition initiated by TDC command) curve acquisitions. The curve may be restarted by means of the TD, TDT or TDC command, as appropriate.

ID Identification

Causes the lock-in amplifier to respond with the number 7210.

LEN [n] Curve sets length control

The value of n specifies the curve sets buffer length in effect for data acquisition. The maximum value is 2,000,000,000, although this is only useful if using the DCFIFO or DCBFIFO readout commands. In normal operation the curve buffer can hold a maximum of 128,000 curve sets, divided between those to be stored. Some examples of the maximum LEN parameter for this mode are shown below.

CBD	max LEN parameter	
1	4000	(128,000 ÷ 32)
2	4000	(128,000 ÷ 32)
3	2000	(128,000 ÷ 64)
4	100,000	(128,000 ÷ 1)
5	3878	(128,000 ÷ 33)
6	3878	(128,000 ÷ 33)
7	1969	(128,000 ÷ 65)

MAXLEN Read maximum curve length

This command reports the maximum curve sets buffer length corresponding to the current setting of CBD when using normal readout (i.e. not DCFIFO or DCBFIFO commands).

M Curve acquisition status monitor

Causes the lock-in amplifier to respond with four values that provide information concerning data acquisition, as follows:

First value, Curve Acquisition Status: a number with five possible values, defined by the following table:

First Value	Significance
0	No curve activity in progress.
1	Acquisition via TD command in progress and running.
2	Acquisition via TDC command in progress and running.
5	Acquisition via TD command in progress but halted by HC command.
6	Acquisition via TDC command in progress but halted by HC command.

Second value, Number of Sweeps Acquired: This number is incremented each time a TD is completed and each time a full cycle is completed on a TDC acquisition. It is zeroed by the NC command and also whenever a CBD or LEN command is applied without parameters.

Third value, Status Byte: The same as the response to the ST command. The number returned is the decimal equivalent of the status byte and refers to the previously applied command.

Fourth value, Number of Curves Available for Transfer: This number is incremented each time a curve is taken. It is zeroed by the NC command and whenever CBD or LEN is applied without parameters. It is decremented each time curves are read out of the buffer using the DCFIFO or DCBFIFO commands by the number of curves transferred.

MSK [n] Set/read service request mask byte

The value of n sets the SRQ mask byte in the range 0 to 255

NC New curves

Initializes the curve sets storage memory and status variables. All record of previously taken data is removed.

OFFSET [n] Automatically Set/Read Interdemodulator Offset Value

When the instrument is operating in tandem mode, the output of the first stage of the demodulator is a DC level with an AC modulation, at the second reference frequency. In order to allow the (wanted) AC signal to occupy as much as possible of the second demodulator's input dynamic range, it is desirable that this DC level is reduced or removed. The OFFSET command allows this to be done.
When the command is sent without the parameter "n", the instrument automatically calculates the optimum offset for each channel and applies it, with the value remaining in effect until next changed or until the instrument is powered down.

The command also operates, for diagnostic purposes only, so that when sent with a parameter "n" in the range 1 to 32, the present offset setting for the corresponding channel (in arbitrary units) is reported.

OVL Locate overload conditions

Causes the lock-in amplifier to respond with four 8-bit numbers expressed as decimal integers in the range 0 to 255, with each response separated from the next by the defined delimiter character. The first number reports the overload conditions in channels 1 to 8, the second in channels 9 to 16, the third in channels 17 to 24 and the fourth in channels 25 to 32.

Within each number, each bit corresponds to the logical ORing of all the bits in the corresponding channel's overload byte. Hence, for example, if channels 1 and 3 are in overload but all other 32 channels are not, then the response would be:

5,0,0,0

In normal use bit 4 in the ST command or GPIB serial poll status byte can be used to identify that an overload has occurred, and when this happens the OVL command can identify which channel(s) are in overload. Finally the OVR command can be used to determine the nature of the overload in the relevant channel(s).

OVR n_1 Report overload byte

Causes the lock-in amplifier to respond, for the channel specified by n_1, with the overload byte, an integer between 0 and 31, which is the decimal equivalent of a binary number with the following bit-significance:

Bit 0	input overload
Bit 1	X1 channel output overload (> ±300 %FS)
Bit 2	Y1 channel output overload (> ±300 %FS)
Bit 3	X2 channel output overload (> ±300 %FS)
Bit 4	Y2 channel output overload (> ±300 %FS)
Bit 5	not used
Bit 6	not used
Bit 7	not used

Note that if bit 1 is set then bits 3 and 4 are meaningless, since if the X1 output is in overload it will not be feeding any valid signal forward in the second (tandem) demodulator stage.

REFMODE [n] Reference mode selector

n	Mode
0	Single Reference mode
1	Tandem Mode
2	Fast TC Single Reference mode

The selected mode applies to all 32 channels

REFMODE 2 disables the second stage (tandem) demodulators and allows time constants down to 1 ms to be specified, by extending the legal range of the TC command parameter.

REFN1 [n] Reference harmonic mode control

The value of n sets the main (first stage) reference channel to harmonic detection, as follows:-

n	Mode
1	1F detection
2	2F detection (only available when REFMODE = 0)

The selected mode applies to all 32 channels

REFP1[.] n_1 [n_2] Reference phase control

In fixed point mode n_2 sets the phase of the channel(s) specified by n_1 for the main (first stage) demodulator(s) in millidegrees in the range ±360000. In floating point mode n_2 sets the phase of the channel(s) specified by n_1 in degrees

REFP2[.] n_1 [n_2] Reference phase control

In fixed point mode n_2 sets the phase of the channel(s) specified by n_1 for the tandem (second stage) demodulator(s) in millidegrees in the range ±360000.
In floating point mode n_2 sets the phase of the channel(s) specified by n_1 in degrees

RS [n_1 [n_2]] Set/read RS232 interface parameters

n_1	Baud rate (bits per second)
0	75
1	110
2	134.5
3	150

n_1	Baud rate (bits per second)
4	300
5	600
6	1200
7	1800
8	2000
9	2400
10	4800
11	9600
12	19200

The lowest five bits in n_2 control the other RS232 parameters according to the following table:

bit number	bit negated	bit asserted
0	data + parity = 8 bits	data + parity = 9 bits
1	no parity bit	1 parity bit
2	even parity	odd parity
3	echo disabled	echo enabled
4	prompt disabled	prompt enabled

SEN1 n_1 [n_2]
SEN1. n_1 Full-scale sensitivity control

In single reference mode, the value of n_2 sets the overall full-scale sensitivity of the channel(s) specified by n_1 according to the following table:

n_2	full-scale sensitivity		
	Voltage Mode	Wideband Current Mode	Low Noise Current Mode
1	100 µV	100 pA	10 pA
2	300 µV	300 pA	30 pA
3	1 mV	1 nA	100 pA
4	3 mV	3 nA	300 pA
5	10 mV	10 nA	1 nA
6	30 mV	30 nA	3 nA
7	100 mV	100 nA	10 nA
8	300 mV	300 nA	30 nA
9	1 V	1 µA	100 nA

In tandem mode, the value of n_2 sets the full-scale sensitivity of the main (first stage) demodulator(s) for the channel(s) specified by n_1 according to the following table:

n_2	full-scale sensitivity		
	Voltage Mode	Wideband Current Mode	Low Noise Current Mode
1	100 µV	100 pA	10 pA
2	300 µV	300 pA	30 pA
3	1 mV	1 nA	100 pA
4	3 mV	3 nA	300 pA
5	10 mV	10 nA	1 nA
6	30 mV	30 nA	3 nA
7	100 mV	100 nA	10 nA
8	300 mV	300 nA	30 nA
9	1 V	1 µA	100 nA

In either mode, SEN1. n_1 reads the sensitivity of the channel(s) specified by n_1 in floating-point mode.

SEN2 n_1 [n_2]
SEN2. n_1 Full-scale sensitivity control

In tandem mode, the value of n_2 sets the full-scale sensitivity of the second stage demodulator(s) for the channel(s) specified by n_1 according to the following table:

n_2	full-scale sensitivity		
	Voltage Mode	Wideband Current Mode	Low Noise Current Mode
1	100 µV	100 pA	10 pA
2	300 µV	300 pA	30 pA
3	1 mV	1 nA	100 pA
4	3 mV	3 nA	300 pA
5	10 mV	10 nA	1 nA
6	30 mV	30 nA	3 nA
7	100 mV	100 nA	10 nA
8	300 mV	300 nA	30 nA
9	1 V	1 µA	100 nA

SEN2. n_1 reads the sensitivity of the channel(s) specified by n_1 in floating-point mode.

For example, let the instrument have voltage mode inputs, be in tandem mode with an applied signal consisting of a 10 mV rms 50 kHz sinusoidal waveform modulated to 50% at the REF 2 OUT frequency, with the reference phases correctly adjusted and SEN1 for the relevant channel set to 5. In this case the X1 output will report nominally 100%, while the X2 output, if SEN2 for the same channel is also set to 5, will report nominally 50%.

SLAVE Read Master/Slave status

The response to this command is a number with the following significance:

n	Significance
0	Unit is Master
1	Unit is Slave

Only one instrument in a group of interconnected units can be the Master.

ST Report status byte

Causes the lock-in amplifier to respond with the status byte, an integer between 0 and 255, which is the decimal equivalent of a binary number with the following bit-significance:

Bit 0	Command complete
Bit 1	Invalid command
Bit 2	Command parameter error
Bit 3	Reference unlock
Bit 4	Input or Output Overload
Bit 5	GPIB Group Execute Trigger or instrument GET 1 command received - Data output buffer update suspended until GET 0 command is received.
Bit 6	Asserted SRQ
Bit 7	Data available

If Bit 4 is asserted, the OVL command can be used to determine where the overload is occurring.

NOTE: this command is not normally used in GPIB communications, where the status byte is accessed by performing a serial poll.

STR [n] Storage interval control

Sets the time interval between successive curve sets being acquired under the TD or TDC commands. n specifies the time interval in milliseconds. When REFMODE ≠ 3 the resolution is 4 ms, input values being rounded up to a multiple of 4. When REFMODE = 3, the resolution is 2 ms, and input values are rounded up to a multiple of 2 The longest interval that can be specified is 1000000 s corresponding to one point in about 12 days. The shortest interval is 2 ms or 4 ms, depending on the REFMODE setting.

TC1 n_1 [n_2]
TC1. n_1 Main (first stage) filter time constant control

n_2	time constant	
-2	1 ms	(REFMODE = 2 only)
-1	2 ms	(REFMODE = 2 only)
0	4 ms	
1	10 ms	
2	30 ms	
3	100 ms	
4	300 ms	
5	1 s	
6	3 s	
7	10 s	
8	30 s	
9	100 s	
10	300 s	
11	1 ks	

The TC1. n_1 command is only used for reading the time constant, and reports the current setting in seconds. Hence if a TC1 1 2 command were sent, TC1 1 would report 2 and TC1. 1 would report 3.0E-02, i.e. 0.03 s or 30 ms.

TC2 n_1 [n_2]
TC2. n_1 Tandem (second stage) filter time constant control

n_2	time constant
2	30 ms
3	100 ms
4	300 ms
5	1 s
6	3 s
7	10 s
8	30 s
n_2	time constant
9	100 s
10	300 s
11	1 ks

The TC2. n_1 command is only used for reading the time constant, and reports the current setting in seconds. Hence if a TC2 1 3 command were sent, TC2 1 would report 3 and TC2. 1 would report 1.0E-01, i.e. 0.1 s or 100 ms.

TD Take data

Initiates data acquisition. Acquisition starts at the current position in the curve sets buffer and continues at the rate set by the STR command until the buffer is full, or until the HC command is issued.

TDT n Take data triggered

Sets the instrument so that data acquisition will be initiated on receipt of a trigger at the **TRIG IN 1** connector on the front panel. Two triggered modes are possible, as set by the value of n:

n	function
0	One complete set of curves, consisting of the number of curve sets specified by the LEN command parameter, is acquired for each trigger
1	One curve is acquired for each trigger. Hence in order to store a complete set of curves, the number of triggers applied must equal the number of curve sets specified by the LEN command parameter. Note that in this mode the maximum trigger rate is 200 Hz and the storage interval control setting has no effect.

A curve set is stored on the next interrupt after the trigger's rising edge. The minimum trigger pulse high time and the minimum trigger pulse low time are both 2 ms. The minimum pulse period is therefore 4 ms. The trigger has to return low again for a minimum of 2 ms before another trigger can be detected. Although curves may be stored if the minimum pulse high and low times are not adhered to, curves will not be stored on every pulse.

TDC Take data continuously

Initiates data acquisition. Acquisition starts at the current position in the curve buffer and continues at the rate set by the STR command until halted by an HC command. The buffer is circular in the sense that when it has been filled, new data overwrites earlier points.

VER Report firmware version

Causes the lock-in amplifier to respond with the firmware version number.

X1[.] n_1 X channel output - Main (first stage)

In fixed point mode causes the lock-in amplifier to respond with the X1 demodulator output of the channel(s) specified by n_1 in the range ±30000, full-scale being ±10000.

In floating point mode causes the lock-in amplifier to respond with the X1 demodulator output of the channel(s) specified by n_1 in amps.

X2[.] n_1 X channel output - Tandem (second stage)

In fixed point mode causes the lock-in amplifier to respond with the X2 demodulator output of the channel(s) specified by n_1 in the range ±30000, full-scale being ±10000.

In floating point mode causes the lock-in amplifier to respond with the X2 demodulator output of the channel(s) specified by n_1 in amps.

XY1[.] n_1 X, Y channel outputs - Main (first stage)
Equivalent to the compound command X1[.]n_1;Y1[.]n_1

XY2[.] n_1 X, Y channel outputs - Tandem (second stage)
Equivalent to the compound command X2[.]n_1;Y2[.]n_1

Y1[.] n_1 Y channel output - Main (first stage)
In fixed point mode causes the lock-in amplifier to respond with the Y1 demodulator output of the channel(s) specified by n_1 in the range ±30000, full-scale being ±10000. In floating point mode causes the lock-in amplifier to respond with the Y1 demodulator output of the channel(s) specified by n_1 in amps.

Y2[.] n_1 Y channel output - Tandem (second stage)
In fixed point mode causes the lock-in amplifier to respond with the Y2 demodulator output of the channel(s) specified by n_1 in the range ±30000, full-scale being ±10000.

In floating point mode causes the lock-in amplifier to respond with the Y2 demodulator output of the channel(s) specified by n_1 in amps.

Index

WARRANTY

SIGNAL RECOVERY, a part of AMETEK Advanced Measurement Technology, Inc, warrants each instrument of its own manufacture to be free of defects in material and workmanship for a period of ONE year from the date of delivery to the original purchaser. Obligations under this Warranty shall be limited to replacing, repairing or giving credit for the purchase, at our option, of any instruments returned, shipment prepaid, to our Service Department for that purpose, provided prior authorization for such return has been given by an authorized representative of AMETEK Advanced Measurement Technology, Inc.

This Warranty shall not apply to any instrument, which our inspection shall disclose to our satisfaction, to have become defective or unusable due to abuse, mishandling, misuse, accident, alteration, negligence, improper installation, or other causes beyond our control. This Warranty shall not apply to any instrument or component not manufactured by AMETEK Advanced Measurement Technology, Inc. When products manufactured by others are included AMETEK Advanced Measurement Technology, Inc equipment, the original manufacturers Warranty is extended to AMETEK Advanced Measurement Technology, Inc customers. AMETEK Advanced Measurement Technology, Inc reserves the right to make changes in design at any time without incurring any obligation to install same on units previously purchased.

THERE ARE NO WARRANTIES WHICH EXTEND BEYOND THE DESCRIPTION ON THE FACE HEREOF. THIS WARRANTY IS IN LIEU OF, AND EXCLUDES ANY AND ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESSED, IMPLIED OR STATUTORY, INCLUDING MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AS WELL AS ANY AND ALL OTHER OBLIGATIONS OR LIABILITIES OF AMETEK ADVANCED MEASUREMENT TECHNOLOGY, INC, INCLUDING, BUT NOT LIMITED TO, SPECIAL OR CONSEQUENTIAL DAMAGES. NO PERSON, FIRM OR CORPORATION IS AUTHORIZED TOASSUME FOR AMETEK ADVANCED MEASUREMENT TECHNOLOGY, INC ANY ADDITIONAL OBLIGATION OR LIABILITY NOT EXPRESSLY PROVIDED FOR HEREIN EXCEPT IN WRITING DULY EXECUTED BY AN OFFICER OF AMETEK ADVANCED MEASUREMENT TECHNOLOGY, INC.

SHOULD YOUR EQUIPMENT REQUIRE SERVICE

A. Contact your local SIGNAL RECOVERY office, agent, representative or distributor to discuss the problem. In many cases it may be possible to expedite servicing by localizing the problem to a particular unit or cable.

B. We will need the following information, a copy of which should also be attached to any equipment which is returned for service.

1. Model number and serial number of instrument
2. Your name (instrument user)
3. Your address
4. Address to which the instrument should be returned
5. Your telephone number and extension
6. Symptoms (in detail, including control settings)
7. Your purchase order number for repair charges (does not apply to repairs in warranty)
8. Shipping instructions (if you wish to authorize shipment by any method other than normal surface transportation)

C. If you experience any difficulties in obtaining service please contact:

SIGNAL RECOVERY Service
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
E-mail: info@signalrecovery.com

or

SIGNAL RECOVERY Service
AMETEK Advanced Measurement Technology
Unit 1 Armstrong Mall, Southwood Business Park
Farnborough,
GU14 0NR, UNITED KINGDOM

Phone: +44 (0)125 255 6800
Fax: +44 (0)125 255 6899
E-mail: info@signalrecovery.com