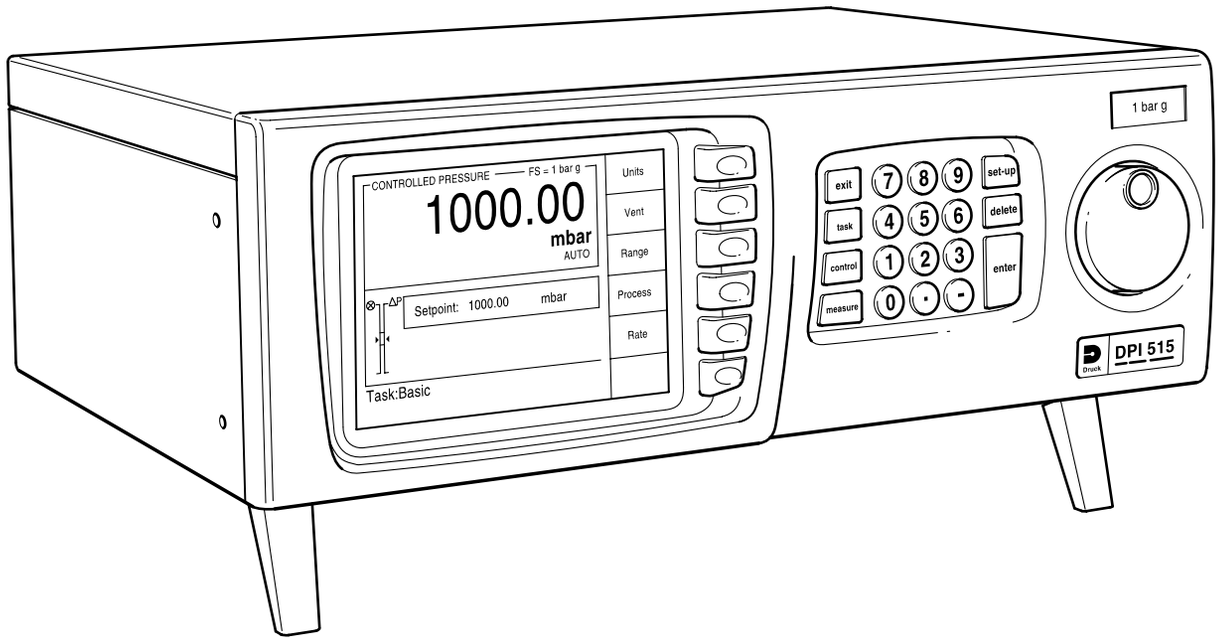
GE Druck

Precision Pressure Controller/Calibrator DPI 515

SCPI User Manual
K257



Druck Limited,
Fir Tree Lane, Groby, Leicester LE6 0FH, England. Tel: (0116) 231 7100 Fax: (0116) 231 7103

K257 Issue No. 2

Introduction

This technical manual provides programming and communication instructions for the Druck DPI 515 Controller/Calibrator compatible with the requirements of a programming technician.

- **Scope**

  This technical manual contains the communications protocol for the operator of this equipment.

- **Software**

  This technical manual applies to software version 1.

Safety

- The manufacturer has designed this product to be safe when operated using the procedures detailed in this manual. Do not use this product for any other purpose than that stated.
- This publication contains operating and safety instructions that must be followed to make sure of safe operation and to maintain the equipment in a safe condition. The safety instructions are either warnings or cautions issued to protect the user and the equipment from injury or damage.
- Use qualified* programming technicians and good engineering practice for all procedures in this publication.

Pressure

Do not apply pressure greater the maximum safe working pressure to the Druck DPI 515 Controller/Calibrator.

Maintenance

The Druck DPI 515 Controller/Calibrator must be maintained using the manufacturer's procedures and should be carried out by authorised service agents or the manufacturer's service departments.

Technical Advice

For technical advice contact the manufacturer or subsidiary.

* A programming technician must have the necessary specialist knowledge of programming, technical knowledge and documentation to carry out the required work on the Druck DPI 515 Controller/Calibrator.

Associated Druck Documents:

K245 DPI 515 User Manual

Associated Documents:

A beginners Guide To SCPI by Barry Eppler, Published by Addison-Wesley Publishing Company Inc. for Hewlett Packard (ISBN 0-201-56350-9)

Table of Contents

Preliminary pages

Table of Contents (contd)

Table of Contents (contd)

List of Illustrations

List of Tables

Abbreviations

The following abbreviations are used in this manual; abbreviations are the same in the singular and plural.

abs	Absolute
ASCII	American Standard Code for Information Interchange
ATE	Automatic test equipment
e.g.	For example
etc.	And so on
Fig.	Figure
ft	Foot
g	Gauge
GPIB	General purpose information bus
i.e.	That is
IEEE 488	Institute of Electrical and Electronic Engineers standard 488 data
m	Metre
max	Maximum
mbar	Millibar
min	Minute or minimum
No.	Number
qa	Quasi-absolute (combination of barometric and gauge sensor readings)
RS232	Serial communications standard
SCM	Sensor calibration module
SCPI	Standard commands for programmable instruments
+ve	Positive
-ve	Negative
°C	Degrees Celsius
°F	Degrees Fahrenheit

Pressure measurement units

The following units are used in this manual

ATM	atmosphere
BAR	bar
CMH_2O	centimetres of water at 20°C
CMHG	centimetres of mercury
FTH_2O	feet of water at 20°C
FTH_2O4	feet of water at 4°C
HPA	hecto Pascals
INH_2O	inches of water at 20°C
INH_2O4	inches of water at 4°C
INH_2O60	inches of water at 60°F
INHG	inches of mercury
KG/CM^2	kilogrammes per square centimetre
KG/M^2	kilogrammes per square metre
KPA	kilo Pascals
LB/FT^2	pounds per square foot
MH_2O	metres of water
MHG	metres of mercury
MMH_2O	millimetres of water
MMHG	millimetres of mercury
MPA	mega Pascals
PA	Pascals
PSI	pounds per square inch
TORR	torr
MBAR	millibar

Code Definitions

The following codes are used in this manual.

CAL	Calibration
CAT	Catalogue
CLS	Clear
COND	Condition
CONF	Configuration
EAV	Error in error queue
ENAB	Enable
EOI	End of input
ERR	Error
ESB	Summary bit from standard event
EVEN	Event
GTL	Go to local
INL	In limit
INP	Input
INST	Instrument
ISOL	Isolation
LEV	Level
LLO	Local lock out
LOG	Logical
MAV	Message available in output queue
MEAS	Measure
MSS	Summary bit after SRQ
OPR	Operation
OUTP	Output
OSB	Summary bit from standard operations status register
PASS	Passive
PRES	Preset
PRES	Pressure
QUE	Queue
QUES	Questionable
RES	Reset
SENS	Sense
SOUR	Source
SRE	Service request enable
SRQ	Service request
STB	Status register query
STAT	State
SYST	System
UNIT	Unit of pressure

Glossary

Terminology

The terminology used in this manual is specific and individual interpretation must not be introduced. The terms are defined as follows:

Adjust	To bring to a more satisfactory state; to manipulate controls, levers, linkages, etc. to return equipment from an out-of-tolerance condition to an in-tolerance condition.
Align	To bring into line; to line up; to bring into precise adjustment, correct relative position or coincidence.
Assemble:	To fit and secure together the several parts of; to make or form by combining parts.
Calibrate:	To determine accuracy, deviation or variation by special measurement or by comparison with a standard.
Check:	Make a comparison of a measure of time, pressure, temperature, resistance, dimension or other quality with a known figure for that measurement.
Disconnect:	To detach the connection between; to separate keyed or matched equipment parts.
Dismantle:	To take apart to the level of the next smaller unit or down to all removable parts.
Examine:	To perform a critical visual observation or check for specific conditions; to test the condition of.
Fit:	Correctly attach one item to another.
Inspect:	Review the work carried out by Specialists to ensure it has been performed satisfactorily.
Install:	To perform operations necessary to properly fit an equipment unit into the next larger assembly or system.
Maintain:	To hold or keep in any particular state or condition especially in a state of efficiency or validity.
Operate:	Make sure that an item or system functions correctly as far as possible without the use of test equipment or reference to measurement.
Readjust:	To adjust again; to move back to a specified condition; to bring back to an in-tolerance condition.

Reconnect:	To rejoin or refasten that which has been separated.
Refit:	Fit an item which has previously been removed.
Remove:	To perform operations necessary to take an equipment unit out of the next larger assembly or system. To take off or eliminate. To take or move away.
Repair:	To restore damaged, worn out or malfunctioning equipment to a serviceable, usable or operable condition.
Replace:	Remove an item and fit a new or a serviced item.
Reset:	To put back into a desired position, adjustment or condition.
Service:	To perform such operations as cleaning, lubricating and replenishing to prepare for use.
Test:	Ascertain by using the appropriate test equipment that a component or system functions correctly.

Pressure unit conversions

Pressure unit	Factor (Pascals)	Pressure unit	Factor (Pascals)
bar	100000	lbf/ft^2	47.8803
lbf/in^2 (psi)	6894.76	inHg	3386.39
mH_2O	9806.65	inH_2O [1]	249.089
mbar	100	ftH_2O [1]	2989.07
kgf/cm^2	98066.5	atm	101325.0
kgf/m^2	9.80665	pdl/ft^2	1.48816
mmHg	133.322	dyn/cm^2	0.1
cmHg	1333.22	hbar	10000000
mHg	133322.0	$tonf/ft^2$ (UK)	107252.0
mm/H_2O [1]	9.80665	$tonf/in^2$ (UK)	15444300
cm/H_2O [1]	98.0665	inH_2O (USA) [2]	248.64135
N/m^2	1	ftH_2O (USA) [2]	2983.6983
hPa	100	kp/mm^2	9806650
kPa	1000	kp/cm^2	98066.5
MPa	1000000	kp/m^2	9.80665
torr	133.322		

Table of pressure units and conversion factors

Unit Conversion

To convert FROM pressure VALUE 1 in pressure UNITS 1
TO pressure VALUE 2 in pressure UNITS 2, calculate as follows:

$$\text{VALUE 2} = \text{VALUE 1} \times \frac{\text{FACTOR 1}}{\text{FACTOR 2}}$$

Note:

The conversion factor for pressure units referenced [1] are calculated for a water temperature of 4°C. Pressure units referenced [2] are calculated for a water temperature of 68°F, these units are normally used in the USA.

1 INTRODUCTION

1.1 General

The IEEE 488 and RS232 interfaces of the DPI 515 Controller/Calibrator provide remote control of the instrument from a suitable computer or controller. The SCPI protocol enables any instrument with a SCPI facility to be controlled using the same commands. The DPI 515 Controller/Calibrator use the full SCPI command set and the defined SCPI syntax.

The following sections describe and define each instrument command used by the DPI 515 Controller/Calibrator. The commands for the aeronautical option and the sensor calibration module option are described and defined in separate sections. Each section contains a quick reference structure of the relevant commands.

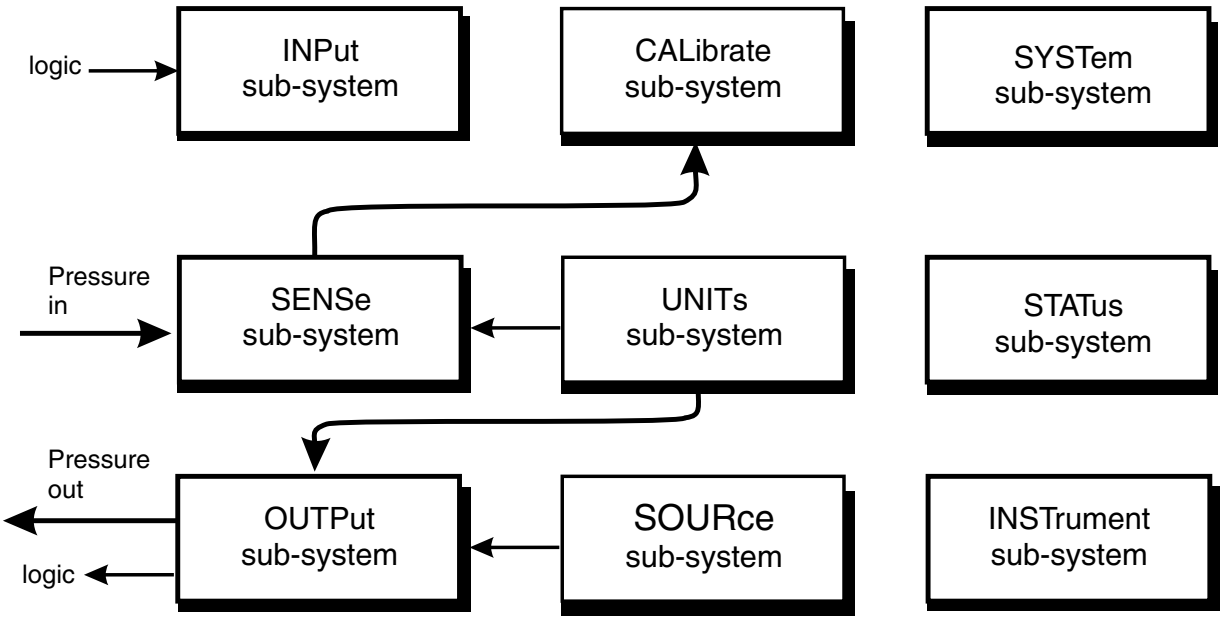


Figure 1-1 System Model

System Model

SCPI starts with a high-level block diagram of the measurement functions of the instrument. Each functional block is broken down into smaller block diagrams. SCPI contains a hierarchy of commands called a subsystem that maps directly to the hierarchy of the block diagram.

1.2 Remote/Local Operation

Any command received over the SCPI interface automatically puts the DPI 515 Controller/Calibrator into remote control mode and disables all front panel keys. Sending the IEEE 488 command GTL (Go To Local) returns the DPI 515 Controller/Calibrator to local control mode. Sending the IEEE 488 command LLO (Local Lock-out) locks the DPI 515 Controller/Calibrator in remote control mode until power-down.

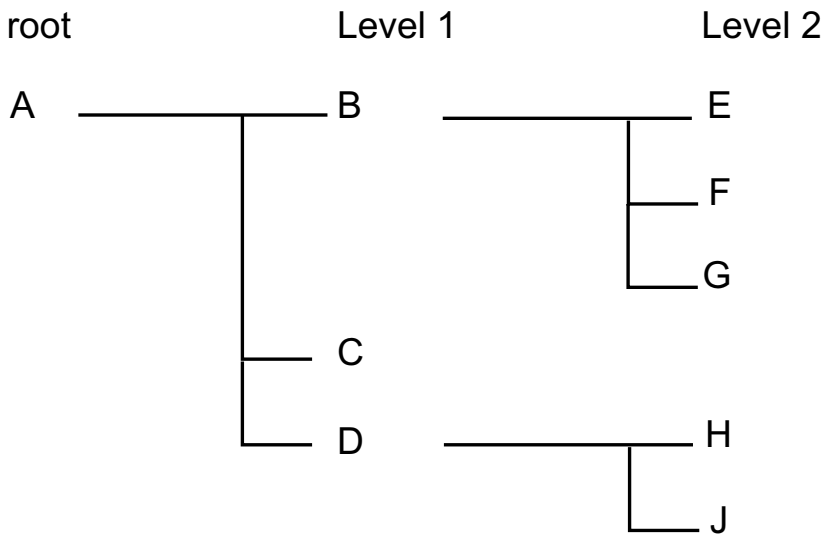
The DPI 515 Controller/Calibrator confirms remote control mode by displaying the message “Under Remote Control”. The display shows the full-scale value in the selected units of measurement and updates pressure readings. The parameters and units, used for the display, only change when the DPI 515 Controller/Calibrator receives a SOURce:PRESsure or UNIT:PRES command (new pressure or units). Any available parameters or units can be used over the SCPI interface.

2 COMMAND STRUCTURE

This section describes the structure of the commands and data sent and received by the DPI 515 Controller/Calibrator.

2.1 Notation

All SCPI commands are based on a hierarchical tree structure consisting of key words and parameters. Associated commands are grouped together under a common node in the hierachy.



In the command tree the command A is the root command. A tree pointer is used to decode the SCPI commands. At power-up the pointer goes to the root command.

2.2 Message Terminators

All SCPI commands are terminated by line feed i.e., either <newline> (ASCII character, decimal 10), EOI for IEEE. After receiving a termination character the tree pointer returns to the root command.

Colon

A colon moves the current path down one level in the command tree, (e.g., the colon in SOURCE:PRESSURE specifies PRESSURE the is one level below SOURCE). When the colon is the first character of the command, it specifies that the next command is a root level command (e.g., :SOURCE specifies that SOURCE is a root level command).

Semicolon

A semicolon separates two commands in the same message without changing the tree pointer.
(e.g., with reference to the tree) :A:B:E;F:G
This equivalent to sending three messages

:A:B:E
:A:B:F
:A:B:G

Commas

If a command requires more than one parameter, separate adjacent parameters by using a comma. A commas does not affect the tree pointer.

(e.g.) :SYSTEM:TIME 10,25,30

To execute a command the full path to the command must be specified

(e.g.) :OUTPut:STATe ON
This turns the pressure controller on.

Note:
There must be a space between the command words and the parameter. In the above example there is a space between :STATe and ON.

SCPI commands are not case sensitive and may have a short form. In this manual, upper case letter identify the short form.

(e.g.) :OUTP is the short form of OUTPUT.

Some nodes can be the default node and these key words are optional when programming the command. The instrument processes the command, with the same effect, with or without the option node. In this manual [] enclose [default notes].

(e.g.) :SOURce[:PRESsure:][:LEVel][:IMMediate][:AMPlitude] 5.0

can be sent as

:SOURce:PRESsure:LEVel 5.0

or

:SOURce 5.0

This sets the set-point to 5.0

2.3 Program Headers

Program headers are keywords that identify a command, instruments accept both upper and lower case characters in a program header. There are two types of program header, common command headers and instrument control headers; each header can be a command or a query.

Common Command and Query Headers

The common command and query program header syntax, specified in IEEE 488.2, are defined as follows:

Command

*<PROGRAM MNEMONIC>

Query

*<PROGRAM MNEMONIC>?

Instrument Control Command and Query Headers

The instrument control command and query program header syntax controls and extracts data from the instrument as follows:

Command

:<MNEMONIC>

:<MNEMONIC> <PARAMETER>

Query

:<MNEMONIC>?

Instrument command headers can have a numeric suffix to identify each of several cases of the same header; the numeric suffix applies to both the long and short forms. All commands headers without a numeric suffix assume the value 1.
e.g.,

:OUTPut:LOGic1:LEVel?

is the same as

:OUTPut:LOGic:LEVel?

Queries

Most SCPI commands can be queried. A query is a command header with an attached question mark character (?). On receiving a query command, the current settings for the command are loaded in the output buffer. A query does not affect the operation or set-up of the instrument.

When the parameter contains enumerated character data, both long form and short form are recognised. Querying the command causes the return of data in the short form.

Querying numeric parameters causes the resulting data to be returned in the units selected by the instrument unless specified otherwise.

2.4 SCPI Data Types

A variety of data types can be sent to the instrument as parameters or sent out from the instrument as response data.

Decimal Numeric Data

All normal decimal expressions are accepted including optional signs, decimal point and scientific notation.

Note:

This includes floating point data.

The following are valid:

123
45.67
-2.6
4.6e-10
.76

A suffix multiplier can be added to the numeric value.

:SOUR 100 m

would set the programmable output to 0.1 units (100m units).

The multipliers supported are:

Mnemonic	Multiplier
A	1e-18
G	1e+9
K	1e+3
M	1e-3
T	1e+12

If a real value is sent to the instrument when an integer is expected, it will be rounded to an integer.

Integer Data

Integer data are whole numbers (containing no decimal places). A query of an integer value returns numbers containing no decimal places.

Note:

Integer values can be specified in binary, octal or hexadecimal formats using the suffix letters (upper or lower case) B, Q and H respectively.

e.g., #B1010 binary representation of 10
#Q71 octal representation of 57
#HFA hexadecimal representation of 250

Hexadecimal digits A-F can be in upper or lower case.

Enumerated Character Program Data
Enumerated characters are used for data that has a finite number of values; enumerated parameters use mnemonics to represent each valid setting. The mnemonics have long and short forms just like command mnemonics.

Example:

:SOURce:PRESsure:SLEW:MODE MAXimum

selects the maximum rate mode.

A query of an enumerated parameter always returns the short form data in upper case.

Example:

:SOURce:PRESsure:SLEW:MODE?

queries rate mode, reply:

MAX

Boolean Program Data
Boolean data can only be one of two conditions; the numbers 1 and 0.

Example:

:OUTPut:STATe 1

A query of boolean data always returns 1 or 0.

String Data
String data can contain any of the ASCII characters. A string must start with a double "quote" (ASCII 34) or a single `quote` (ASCII 39) and end with the same character.

Example:

:SOURCe[:PRESsure]:RANGe ‘2barg’

or

:SOURCe[:PRESsure]:RANGe “2barg”

selects the 2 bar g range.

A query of a string parameter always returns the string in double "quotes".

Intentionally left blank

3 STATUS SYSTEM

The status reporting system informs the external controller that an event has occurred. This information is in the form of a service request (SRQ) using IEEE 488 or an SRQ message using RS232.

The DPI 515 instrument uses status reporting as defined in IEEE 488.2 with the implementation of status registers.

The OPERation and QUEStionable status registers have been implemented To comply with the SCPI protocol. These are registers where the individual bits are summary bits of the status of the instrument. Since the SCPI protocol does not include pressure instruments, bit 10 of both these registers are used as a pressure summary bit. This pressure summary bit is expanded to two, 16 bit registers (Bit 15 is not used and is always zero).

The Questionable and Pressure Questionable registers are not used. The only bit implemented in the Operation status register is bit 10 (summary of the pressure operation status).

A summary bit is the final output of a data structure, it is a single bit that shows the status of one or more related events in the instrument. The basic structure of a summary bit

- Condition register
- Event register
- Enable register
- Logical ANDing of the Event and Enable registers
- Summary bit that summarises the result using OR logic

Condition Register

This register shows the current status of the device. The condition register is constantly updated - the bits in the register are set or reset showing the current condition.

Event Register

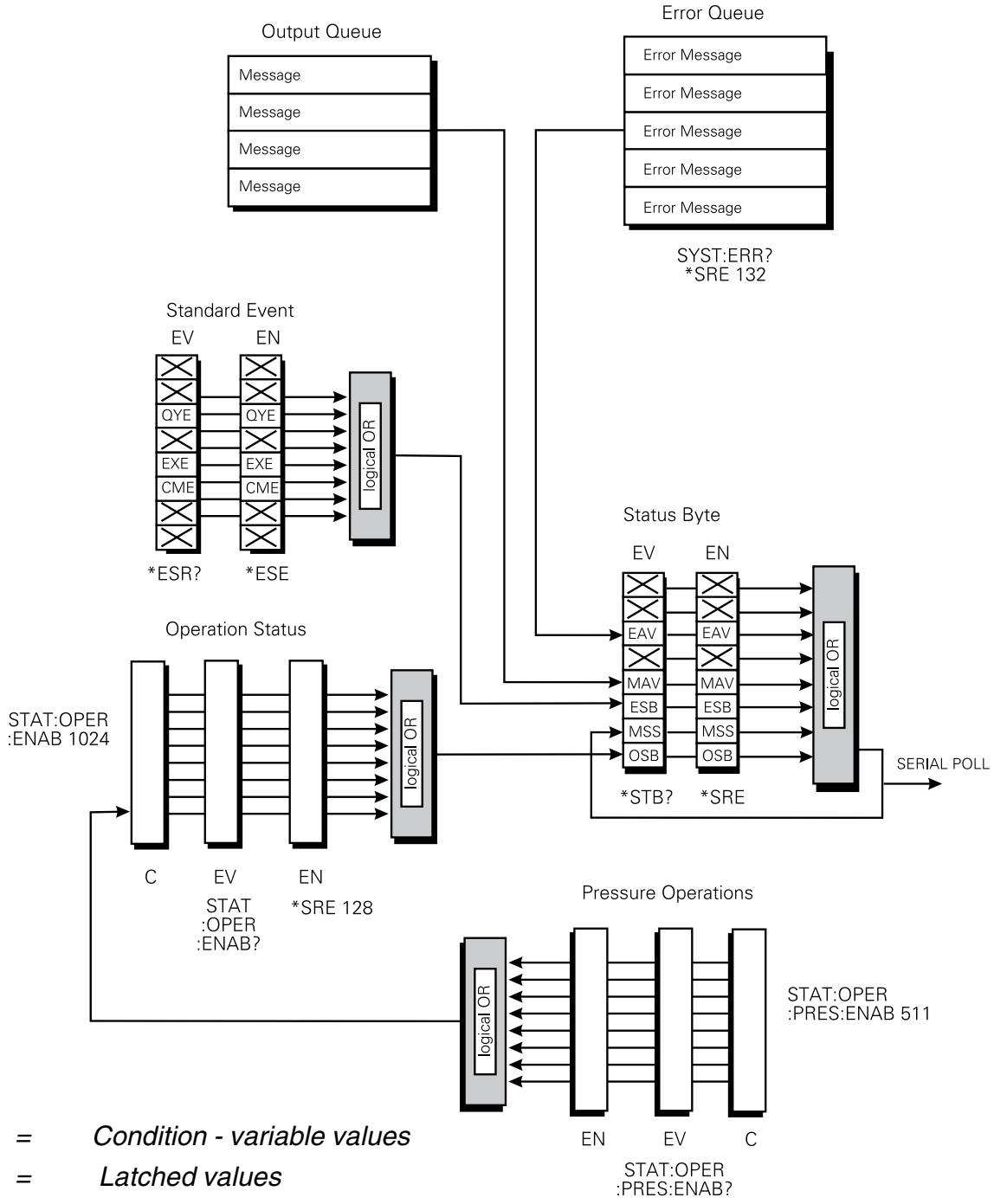
The event register shows an event that occurs in the condition register (a condition bit goes from low to high). This condition change is stored and only reset when the event register is read or the *CLS command sent.

Enable Register

This register allows the results of the event register to pass through to the next cascaded register and enables the user to select the event that should generate the final SRQ event.

The status system implemented in the instrument is shown in the following diagram:

Note: *Initial values of registers are 0, with the queues empty.*



Key:

C = Condition - variable values

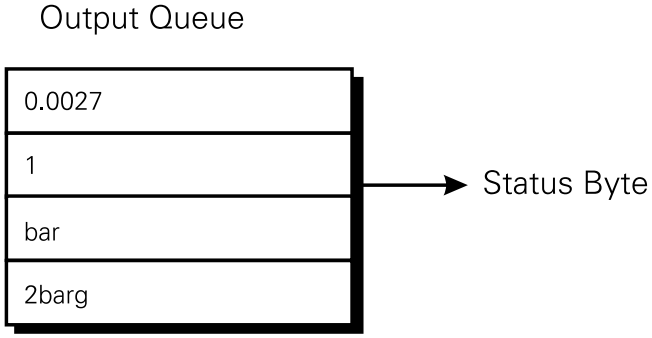
EV = Latched values

EN = Bit mask

Figure 3-1 Status System

3.1 Output queue

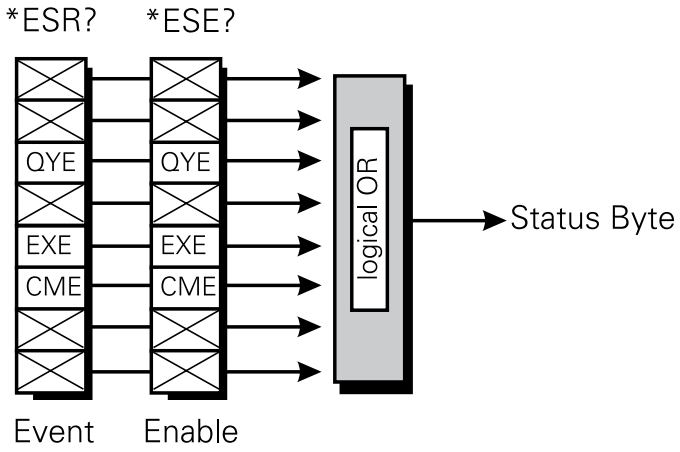
The output queue is a text readable data queue that is read through the IEEE 488 talk command. The queue is cleared by reading all elements in it or by the *CLS command.



Every time a query has been successfully completed, the response, in a text readable format is placed at the end of the output queue. If the MAV bit in the "Status Byte" was previously cleared it will be set. The output queue can contain up to 256 characters. If there is not enough space in the output queue for a new message, the error -350, "Queue overflow" will be placed into the error queue and the most recent output message will be lost.

3.2 Standard event group

The standard event group are 8 bit registers that are read by the IEEE 488 standard commands. The event register is cleared by reading it; the event and enable registers are cleared by the *CLS command.



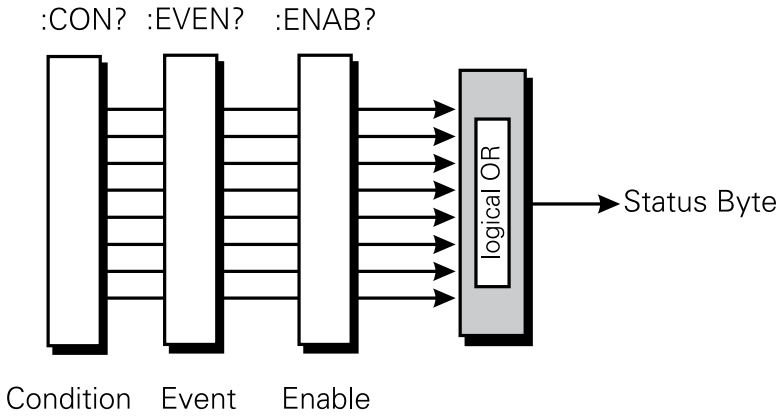
Bits within the standard event condition register are set by system errors and events. In addition to setting the status bits, a text message will be placed in the error queue. The ESB bit in the status byte sets if the associated bit in the event enable register is set. The enable register may be set through the *ESE command so that selected standard events cause the ESB bit to be set. The system events that set each bit are as follows:

Bit	Name	Description	Meaning/data
0	OPC	Not used	Reserved currently returns 0
1	RQC	Not used	Reserved currently returns 0
2	QYE	-400 to -499	Query errors
3	DDE	Not used	Reserved currently returns 0
4	EXE	-200 to -299	Execution errors
5	CME	-100 to -199	Command errors
6	URQ	Not used	Reserved currently returns 0
7	PON	Not used	Reserved currently returns 0

Table 3-1 Standard Event Register

3.3 Operation status group

The operation status group are 16 bit registers that are read by the STAT:OPER commands. The event register is cleared by reading it; the event and enable registers are cleared by the *CLS command.



When a standard operation condition occurs an appropriate bit is set in the condition register (this clears when the condition no longer exists). The bit is then latched in the event register. If the associated bit in the enable register is set, the OPR bit in the status byte sets. The enable register may be set through the STAT:OPER:ENAB command so that only selected standard operation events cause the OPR bit to set.
Problems can occur with some IEEE 488 controllers reading 16 bit unsigned numbers. All registers in this group do not use bit 15. The enable bit cannot be set and when read returns 0. The condition register is defined as follows:

Vent complete

This signal occurs when the controller has been requested to vent and the vent has completed or timed out.

Range change complete

This signal occurs when the controller has been requested to perform a range change and the range change is complete.

In-Limits reached

This signal is set every time the controlled pressure is within the specified limits. The signal is only generated if the pressure has been within limits for a user defined wait time period.

Zero complete

This signal is generated when a manual or timed zero is complete. If the zero times out then this signal is also generated.

Bit (1)	Data (2)	Bit (3)	Data (4)
0	Vent complete	1	Range change complete
2	In-limits reached	3	Zero complete
4	Auto-zero started	5	Fill time, timed-out
6	Reserved - returns 0	7	Reserved - returns 0
8	Switch contacts changed state	9	Reserved - returns 0
10	Reserved - returns 0	11	Reserved - returns 0
12	Reserved - returns 0	13	Reserved - returns 0
14	Reserved - returns 0	15	Reserved - returns 0

Table 3-2 Operation Status Register

Auto zero started

When the controller is in the auto zero mode this signal indicates that the auto zero process has started. Thezero complete signal indicates that the zero process has finished.

Fill timed out

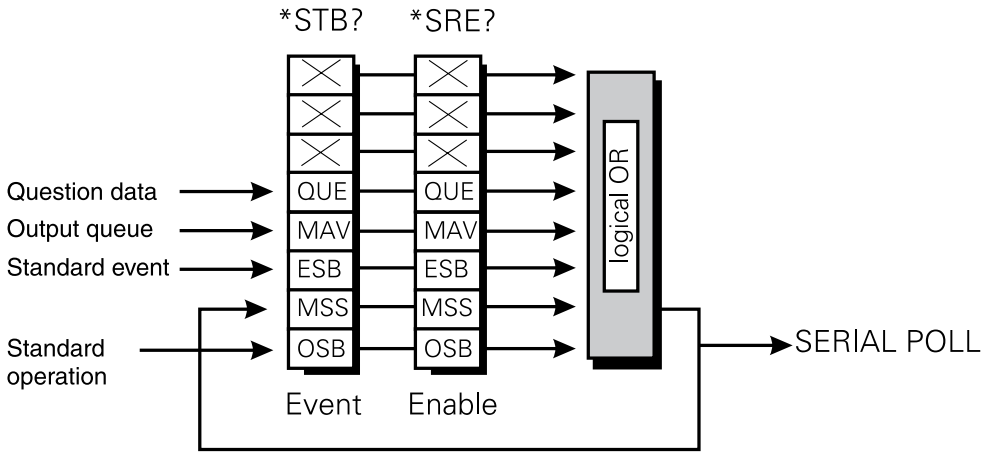
If a set-point has been requested and the set-point cannot be achieved within the fill timeout time, the fill timed out signal is generated.

Switch contacts changed state

Every time the switch contacts used for performing a switch test change state this bit is set.

3.4 Status Byte group

The status byte group are 8 bit registers that are read by the IEEE 488 standard commands. The event register is cleared by reading it; the event and enable registers are cleared by the *CLS command.



Bits within the status byte are a summary of other data structures in the status system. These bits will become set if other parts of the status system indicates that they should do so (i.e., a message in the output queue or error queue or, a condition and enable set in a register pair).
If the associated bit in the status enable register is set, a serial poll is generated and bit 6 is set. The enable register may be set through the *SRE command so that only selected status bits cause a serial poll.

Note: *Bit 6 of the enable register is always set to 0.*

There are some small differences between * STB? and serial polling. Either method can be used to read the state of bits 0-5 and bit 7. The reading method is different for bit 6 when using *STB? and serial poll. In general, use serial polling inside interrupt service routines, not *STB?

Bit 2 - EAV sets when there is an error in the error queue. The :SYST:ERR? command has to be sent to retrieve the error. The error queue buffers a maximum of five errors. When no more errors are available the message “No Error” is returned.

Bit 4 - MAV sets when there is a message available in the output queue.

Bit 5 - ESB sets when a standard event has occurred in the Standard Event Register.

Bit 6 - MSS sets when an SRQ is generated - SRQ sets when both the Status byte and the Service Request Enable register are at logic 1 (AND function).

RS232 Specific
A service request (SRQ) produces the message: :SRQ <value>

where:
<value> = the contents of the status summary byte.

The status system data structure sets each bit as follows:

Bit	Name	Description
0	-	Reserved currently returns 0
1	-	Reserved currently returns 0
2	EAV	Error in error queue
3	-	Reserved currently returns 0
4	MAV	Messages available in output queue
5	ESB	Summary bit from standard event
6	MSS	Summary bit after service request - SRQ
7	OSB	Summary bit from standard operations status

Table 3-3 Status Byte Register

Example commands using the Status Byte and Status Byte Enable registers:

***SRE 16** *Generate an SRQ interrupt when messages are available.*
***SRE?** *Find out what events are enabled to generate SRQ interrupts.*
***STB?** *Read and clear the Status Byte Event register.*

IEEE 488 Specific
Bit 7 - OSB sets when the pressure operations register bit 10 changes state.
The operations register is a 16 bit register only using bit 10. This bit is a summary of the pressure operations register.

Status reporting register structure

To set-up the status reporting system.

1 All status registers should be cleared by the command:
*CLS

2 The Pressure Operations Event register has to be set to enable the Pressure Operations Condition Register to send all the events to be reported; use the command:

:STAT:OPER:PRES:ENAB 511

The enabled events may also be read by the query:

:STAT:OPER:PRES:ENAB?

3 The Operation Status Event register must then be enabled to read bit 10 by the command:

:STAT:OPER:ENAB 1024

The enabled events may also be read by the query:

:STAT:OPER:ENAB?

4 The status request to enable the SRQ must then be set.

To enable only the Operation Status register (OSB) send the command:

*SRE 128

To enable the Operation Status register (OSB) and the Error Queue (EAV) send the command:

*SRE 132

This register may also be read by the query:

*SRE?

An event occurring generates an SRQ, the Status Byte should be queried to find the source of the event.

If bit 2 of the Status Byte Register is set the error queue can be read by the query:

:SYST:ERR?

Keep issuing this query until there are no more errors in the error queue. At this point, bit 2 of the Status Byte Register clears.

If bit 7 of the Status Byte Register is set the Pressure Operations event register can be read by the query:

:STAT:OPER:PRES?

returning the bits of events that have occurred. Reading this register clears it and the associated status bit (bit 7).

At any time the instantaneous status of the pressure system can be read by the query:

:STAT:OPER:PRES:COND?

3.5 Instrument Errors

Any instrument error that occurs, either programming errors or execution errors, is stored in an error queue which is separate from the main output queue. The errors can be read by issuing the following command query:

:SYST:ERR?

The error queue can hold up to five errors. Each time the error queue is queried the instrument responds with the next stored error in the queue. The response consists of an error number followed by a string describing the error. When the error queue is empty the instrument responds with:

0,”No error”

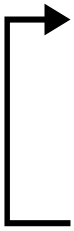
Querying the error queue clears the storage location in the error buffer. If more than five errors occur, before being queried, the ‘Queue overflow;Error queue overflow’ message is placed into the error queue. All subsequent errors are lost until the error queue is cleared.

3.6 Example program

The following program example is written in QUICK BASIC and illustrates the use of various commands and responses.

Program Overview

Initializes the IEEE 488 interface.
Gets instrument identity.
Gets pressure ranges fitted
Sets units as mbar and queries units.
Enters rate of change as maximum rate and queries rate mode.
Enters 1000 mbar set-point and queries set-point.
Sets controller on and queries controller state.
Queries pressure reading time/date (keys a and b changes the query parameter).
Print pressure reading or time/date.
Print time and date.
Wait 1 second.
Goes round loop until q key is pressed.
Sets controller off.
Queries controller state.



See www.DPI515.com/downloads for more examples

Example program

```
OPEN "IEEECTRL" FOR RANDOM AS #1
OPEN "IEEEDATA" FOR RANDOM AS #2

IOCTL #1, "4"
PRINT IOCTL$(1) 'check device address

IOCTL #2, "*IDN?" + CHR$(10)    ' send query instrument type
PRINT IOCTL$(2)                ' read reply of instrument type

IOCTL #2, ":INST:CAT?" + CHR$(10)   ' send query ranges fitted
PRINT IOCTL$(2)                     ' read reply of ranges fitted


IOCTL #2, ":UNIT MBAR" + CHR$(10)   ' select mbar units
IOCTL #2, ":UNIT?" + CHR$(10)       ' send units query
PRINT IOCTL$(2)                     ' read reply of units query

IOCTL #2, ":SOUR:SLEW:MODE MAX" + CHR$(10) ' select max rate
IOCTL #2, ":SOUR:SLEW:MODE?" + CHR$(10)    ' send rate mode query
PRINT IOCTL$(2)                          ' read reply of rate mode query

IOCTL #2, ":SOUR 1000.0" + CHR$(10)   ' setpoint of 1000 mbar
IOCTL #2, ":SOUR?" + CHR$(10)         ' send setpoint query
PRINT IOCTL$(2)                       ' read reply of setpoint query

IOCTL #2, ":OUTP 1" + CHR$(10)     ' turn controller on
IOCTL #2, ":OUTP?" + CHR$(10)     ' send controller state query
PRINT IOCTL$(2)                  ' read reply of controller state query

A$ = ":sens?" + CHR$(10)              ' data to query pressure

DO
        IF INKEY$ = "a" THEN A$ = ":sens?" + CHR$(10)  ' query pressure
        IF INKEY$ = "b" THEN A$ = ":syst:time?;date?" + CHR$(10) '
query time and date"
        IOCTL #2, A$
        PRINT IOCTL$(2)  ' read the device
        SLEEP 1          ' wait for 1 second

LOOP WHILE INKEY$ <> "q"
IOCTL #2, ":OUTP 0" + CHR$(10)   ' turn controller off
IOCTL #2, ":OUTP?" + CHR$(10)    ' send controller state query
PRINT IOCTL$(2)                  ' read reply of controller state query

CLOSE #1: CLOSE #2
END
```

4 COMMAND AND QUERY SUMMARY

The following lists of all the SCPI commands and queries that apply to the instrument.

4.1 Command structure

Some of the commands in the following summary are enabled at specific times and conditions, most can be enabled at any time. The parameters column shows the states, values and data contained in a command. The command structure divides into subsystems as follows:

Command sub-system

:CALibration - calibration commands.

:INPut - reads the logical input.

:INSTrument - instrument specific commands.

:OUTPut - controls the output pressure and logical outputs.

:SENSe - directs the instrument to measure selected parameters.

:SOURce - covers the commands that control the pressure outputs.

. :STATus - instruments state.

:SYSTem - errors and SCPI version.

:UNIT - sets the units for the instrument.

Command	Parameters	Comments
CALibration		
:[PRESSure]		
:ZERO		zero offset
:AUTO		
:AUTO?	<Boolean>	on/off
:VALue	<number>	offset value of absolute range
:TIME	<integer>	
:STATe	<Boolean>	enabled/disabled timed zero
INPut		reads logical input
:LOGic		
[:STATe]?		
INSTrument		lists ranges and details
:CATalog		of transducers fitted
[:LIMits]?		
:SN		
UNIT?		
:VERSion?		
OUTPut		Turns pressure controller
:ISOLation		on/off, controls isolation
[:STATe]	<Boolean>	valves and logical outputs.
:STATe?		
:LOGic		
:LEVel	<Boolean>	
:LEVel?		
[:STATe]	<Boolean>	
[:STATe]?		
:SENSe		configures measurement
[:PRESsure]?		functions, height difference
BARometer?		adjustment. Air or nitrogen.
:CORRection		
:HEAD	<enumerated>,<numeric>	
:HEAD?		
:STATe	<Boolean>	
:STATe?		
:OFFset		
:OFFset?		
:STATe	<number>	
:STATe?		

continued ...

Figure 4-1 Command and Query Summary

Command	Parameters	Comments
:FILTer	<number>	
[:LPASs]		low pass filter
:STEP	<number>	filter step response
:STEP?		
:FREQuency	<number>	filter averaging value
:FREQuency?		
[:STATe]	<Boolean>	on/off
[:STATe]?		
:FILL		time in seconds
:TIME	<integer>	
:TIME?		
:STATe	<Boolean>	on/off
:STATe?		
:RANGe	<string>	sensor range
:RANGe?		
SOURce		
[:PRESsure]	<string>	pressure output details
:EFFort?		
:INLimits	<number>	
:INLimits?		
:TIME	<number>	time in seconds
:TIME?		
[:LEVel]		
[:IMMediate]		
[:AMPlitude]		
:VENT	<number>	vents the user system
:VENT?		
:TIME	<number>	
:TIME?		
:RANGe	<string>	controller range information
:RANGe?		
:LOW?		
:SLEW	<number>	rate in pressure units/ second
:SLEW?		
:MODE	<enumerated>	
:MODE?		
:OVERshoot	<Boolean>	
[:STATe]		
[:STATe]?		

continued ...

Figure 4-1 Command and Query Summary

Command	Parameters	Comments
:STATus		
:OPERation		status register information
:CONDition?		
:ENABle	<integer>	
:ENABLe?		
[:EVENt]?		
:PRESsure		
:CONDition?		
:ENABle	<integer>	
:ENABLe?		
[:EVENt]?		
:SYSTem		
:ERROr?		error queue
:VERSion?		SCPI version
:UNIT		
[:PRESsure]	<name>	
[:PRESsure]?		
:DEFine	<string>,<number>	ASCII (unit name) conversion factor
:DEFine?		
***CLS**		standard SCPI commands
***ESE**	<integer>	
***ESE?**		
***ESR?**		
***GTL**		
***IDN?**		
***LLO**		
***LOC**		
***REM**		
***RST**		
***SRE**	<integer>	
***SRE?**		
***STB?**		

Figure 4-1 Command and Query Summary

CALibration

The CALibration subsystem enables the calibration of the transducers and the rate control system, refer to the user manual for further details.

:CAL:ZERO:AUTO

Command Syntax

:CALibration:ZERO:AUTO <Boolean>

Parameter:

	Boolean	0	-	aborts the zero
		1	-	performs a zero

Short form: :CAL:ZERO:AUTO
Function: Pressure zeroing
Default: 0

Query Syntax

:CALibration:ZERO:AUTO?

Short form: :CAL:ZERO:AUTO?
Function: Query progress of zero
Response:

0 - Zero OK
1 - Zero in progress
2 - Zero timed out
3 - Zero offset outside limits
4 - Zero aborted

Description

This command is used to zero the currently selected sensor and starts a zero sequence. The progress of the zero can be monitored by using the query for this command. A parameter of zero aborts this command function (zero sequence).

:CAL:ZERO:AUTO:VAL

Command Syntax

:CALibration:ZERO:AUTO:VALue <number>

Parameter: number applied pressure

Short form: :CAL:ZERO:AUTO:VAL
Function: Absolute sensor zeroing
Defaults: 0.0
minimum N/A
maximum N/A

Query Syntax

:CALibration:ZERO:AUTO:VALue?

Short form: :CAL:ZERO:AUTO:VAL?
Function: Ask applied value
Response: Number corresponding to applied pressure in the currently selected units.

Description

For absolute sensors a pressure of 0 is very difficult to achieve to perform the zero. Therefore to zero the absolute sensor a near vacuum is applied and measured. This command sets the measured pressure so that a single point compensation can be applied to the reading.

:CAL:ZERO:TIME

Command Syntax

:CALibration:ZERO:TIME <integer>

Parameter:	integer	timed zero time in hours

Short form:	:CAL:ZERO:TIME	
Function:	Timed zero time.	
Defaults:	24	
	minimum	1
	maximum	999

Query Syntax

:CALibration:ZERO:TIME?

Short form:	:CAL:ZERO:TIME?
Function:	Timed zero time-out in hours
Response:	Integer value for the timed zero time-out in hours

Description

The timed zero function allows the instrument to perform periodic zeroes on gauge ranges. The zero is performed when the timed zero time-out period has expired and a set-point of 0 is entered.

:CAL:ZERO:TIME:STAT

Command Syntax

:CALibration:ZERO:TIME:STATe <Boolean>

Parameter	Boolean	0	-	disable timed zero
		1	-	enable timed zero

Short form: :CAL:ZERO:TIME:STATe <Boolean>
Function: Timed zero state
Default: OFF

Query Syntax

:CALibration:ZERO:TIME:STATe?

Short form: :CAL:ZERO:TIME:STAT?
Function: Query state (on or off) of timed zero
Response: 1 (on) 0 (off)

Description

This command is used to enable or disable the timed zero function.

INPut

The INPut subsystem obtains information about the input signal of the instrument. This subsystem contains query only command to get the state of the logical input used for switch testing.

:INP:LOG

Command Syntax

n/a

Parameter:
Short form:
Function:
Default:

Query Syntax

:INPut:LOGic[:STATe]?

Short form: :INP:LOG?
Function: Reads the logic input state
Response: 1 (logic 1) 0 (logic 0)

Description

This query only command is used to read the state of the logical input.

INSTrument

The INSTrument subsystem gets information about the configuration of the instrument and contains query only commands.

:INST:CAT

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:INSTrument:CATalog?

Short form: :INST:CAT?
Function: Query ranges fitted
Response: A list of comma separated strings of ranges fitted.

Description

This command returns a list of ranges fitted to the instrument. The reply is a comma separated list of strings representing each range.

e.g.
“2barg”,”3.5barqa”.

If a barometer is fitted, the string “BAROMETER” is added to the list.

:INST:LIM

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:INSTrument:[LIMits][x]?

Short form: :INST?
Function: Queries the upper and lower full-scale limits of the fitted ranges. The index number x is used to index into the list of available ranges.
Response: A string representing the range, a number representing the upper full-scale and a number representing the lower full-scale.

Description

This query only command return the name of the range as a string and the upper and lower pressure limits in the selected units.

e.g. :INST3?

“2barg”, 2.0, -1.0

:INST:SN

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:INStrument:SN?

Short form:	:INST:SN?
Function:	Used to query the serial number of the instrument. Asks for serial number
Response:	Integer representing serial number

Description

This query only command returns the serial number of the instrument.

:INST:UNIT

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:INSTrument:UNIT[x]?

Short form: :INST:UNIT?
Function: Queries the available units. The number x is the index into the units.
Response: Enumerated type of the unit.

ATM
BAR
CMH2O
CMHG
FTH2O
FTH2O4
HPA
INH2O
INH2O4
INH2O60
INHG
KG/CM2
KG/M2
KPA
LB/FT2
MH2O
MHG
MMH2O
MMHG
MPA
PA
PSI
TORR
MBAR
USER1
USER2
NONE

Description

This command is used to query the available units of pressure measurement in the instrument. The units list depends on the area of use configuration. If a index number is specified which does not contain a unit then NONE is returned.

:INST:VERS

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:INSTrument:VERSion?

Short form: :INST:VERS?
Function: Queries the software versions of the controller.
Response: Returns three strings representing the version numbers of the main controller code, CAN controller and BOOT code.

Description

The main controller code is the part that can be upgraded through the code update utility. The CAN and BOOT code is stored in EEPROM and can only be replaced by replacing the EEPROM.

OUTPut

The OUTPut subsystem turns the pressure controller on/off, controls the state of the isolation valves and the state of the logical outputs.

:OUTP:ISOL:STAT

Command Syntax

:OUTPut:ISOLation[:STATe] <Boolean>

Parameter:	Boolean	0	-	close
		1	-	open

Short form: :OUTP:ISOL
Function: Isolation valve state
Default: OFF - isolation valve closed

Query Syntax

:OUTPut:ISOLation:STATe?

Query :STATe?
Short form: :OUTP:ISOL?
Function: Queries state of isolation valve
Response: 1 (open) 0 (closed)

Description

Opens/Closes the isolation valve of the current range.

:OUTP:LOG:LEV

Command Syntax

:OUTPut:LOGic[x]:LEVel <Boolean>

Parameter:	Boolean	0	-	level low
		1	-	level high

Short form: :OUTP:LOG[x]:LEV
Function: Controls the level of the digital outputs. The index number x specifies the output line. 1 for output 1 and 2 for output 2.
Default: 0

Query Syntax

:OUTPut:LOGic[x]:LEVel?

Short form: :OUTP:LOG[x]:LEV?
Function: Asks for level of output line
Response: 0 - level low
1 - level high

Description

The instrument has two digital output lines, the levels on these lines are controlled using this command.

:OUTP:STAT

Command Syntax

:OUTPut:STATe <Boolean>

Parameter:	Boolean	0	-	turn controller off
		1	-	turns controller on

Short form: :OUTP
Function: Turn the pressure controller on/off
Default: 0

Query Syntax

:OUTPut:STATe?

Short form: :OUTP?
Function: Asks for state of pressure controller
Response: 0 - controller off
1 - controller on

Description

Sets or queries the state of the pressure controller.

SENSe

The SENSe subsystem selects and configures the measurement functions of the instrument.

:SENS:PRES

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SENSe[:PRESsure]?

Short form: :SENS?
Function: This query only command reads the sensor which has been selected by the RANGE command.
Response: A pressure reading in the current units.

Description

Queries the pressure reading for the selected sensor in the selected units. The sensor can be changed see, :SENSe[:PRESsure]:RANGe and :SOURe[:PRESsure]:RANGe commands. Issuing the SOURce command can also change the range.

:SENS:BAR

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SENSe[:PRESsure]:BARometer?

Short form: :SENS:BAR?
Function: Queries the barometric pressure value.
Response: <number> in the selected units of pressure measurement.

Description

Returns the barometric pressure measured by the optional barometric transducer. If the optional barometric transducer is not fitted the response is zero pressure.

:SENS:CORR:HEAD

Command Syntax

:SENSe[:PRESsure]:CORRection:HEAD <enumerated>,<numeric>

Parameters:	enumerated	AIR	- Air used as gas
		NITRogen	- Nitrogen used as gas
	numeric	Height of gas in cm.	
Short form:	:SENS:CORR:HEAD <enumerated>,<numeric>		
Function:	Head correction parameters		
Default:	Enumerated	AIR	
		numeric	0
	Minimum	numeric	-1e10 cm
	Maximum	numeric	1e10 cm

Query Syntax

:SENSe[:PRESsure]:CORRection:HEAD?

Short form:	:SENS:CORR:HEAD?
Function:	Query gas and height of head correction
Response:	AIR/NITRogen and height in cm

Description

A correction must be made if the unit under test is at a different height from the instrument. This command programs the gas used and the height difference.

:SENS:CORR:HEAD:STAT

Command Syntax

:SENSe[:PRESsure]:CORRection:HEAD:STATe <Boolean>

Parameter	Boolean	0	-	Disables head correction
		1	-	Enables head correction

Short form: :SENS:CORR:HEAD:STAT <Boolean>
Function: Enables / disables head correction.
Default: 0

Query Syntax

:SENSe[:PRESsure]:CORRection:HEAD:STATe?

Short form: :SENS:CORR:HEAD:STAT?
Function: Query head correction state
Response: 0 - head correction off
1 - head correction on

Description

This command enables or disables the head correction compensation.

:SENS:CORR:HEAD:OFF

Command Syntax

:SENSe[:PRESsure]:CORRection:HEAD:OFFset

Parameter:	<number> tare offset value		
Short form:	:SENS:OFF		
Function:	Subtracts the offset value from the processed reading.		
Default:	0		
	Minimum	numeric	-1e10 cm
	Maximum	numeric	1e10 cm

Query Syntax

:SENSe[:PRESsure]:CORRection:HEAD:OFFset?

Short form:	:SENS:OFF?
Function:	Asks for the tare value.
Response:	Number corresponding to the tare offset value.

Description

This command must be enabled by SENS:OFF:STAT and sets up the offset value.

:SENS:CORR:HEAD:OFF:STAT

Command Syntax

:SENSe[:PRESsure]:CORRection:HEAD:OFFset:STATe

Parameter:	<Boolean>	0	-	disables tare offset
		1	-	enables tare offset
Short form:	:SENS:OFF:STAT			
Function:	Enables and disables the tare offset function.			

Query Syntax

:SENSe[:PRESsure]:CORRection:HEAD:OFFset:STATe?

Short form:	:SENS:OFF:STAT?
Function:	Asks if tare offset function is on or off.
Response:	1 (on) 0 (off)

Description

This command enables SENS:OFF to be used to set-up the offset value.

:SENS:FILT:STEP

Command Syntax

:SENSe[:PRESsure]:FILTer:[LPASs]:STEP <number>

Parameter: number filter step response value

Short form: :SENS:FILT:STEP
Function: Used to set-up the step response band component of the filter.
Default: 0
minimum 0
maximum 100.0

Query Syntax

:SENSe[:PRESsure]:FILTer:[LPASs]:STEP?

Query :STEP?
Short form: :SENS:FILT:STEP?
Function: Ask for filter step response band parameter.
Response: Number corresponding to filter step response value.

Description

The digital low pass filter has a step response band configured as percentage of full-scale. e.g., defaults to 0.05 %FS. If the reading has changed by more than the configured step response value then the filtering is ignored for that conversion and the pressure goes instantly to the new value.

:SENS:FILT:FREQ

Command Syntax

:SENSe[:PRESsure]:FILTer:[LPASs]:FREQuency <number>

Parameter: number filter averaging value

Short form: :SENS:FILT:FREQ
Function: Used to set up the averaging component of the filter.
Default: 0
minimum 0
maximum 20

Query Syntax

:SENSe[:PRESsure]:FILTer:[LPASs]:FREQuency?

Short form: :SENS:FILT:FREQ?
Function: Ask for filter average parameter.
Response: Number corresponding to filter average value.

Description

A digital low pass filter can be applied to the pressure reading. This is a first order low pass filter, the time constant depends on the value set by this command.

:SENS:FILT:STAT

Command Syntax

:SENSe[:PRESsure]:FILTer[:LPASs]:STATE <Boolean>

Parameter:	Boolean	0	-	Disables low pass filter
		1	-	Enables low pass filter

Short form: :SENS:FILT <Boolean>
Function: Sets low pass filter ON or OFF.
Default: OFF

Query Syntax

:SENSe[:PRESsure]:FILTer[:LPASs]:STATe?

Short form: :SENS:FILT?
Function: Query state (on or off) for the low pass filter
Response: 1 (ON) 0 (OFF)

Description

This command is used to enable or disable the low pass filter for producing a more stable reading. An 'intelligent' filter is implemented so that any noise in the system is filtered while step changes pass straight though the filter.

:SENS:FILL:TIME

Command Syntax

:SENSe[:PRESsure]:FILL:TIME <integer>

Parameter:	integer	fill time in seconds

Short form:	:SENS:FILL:TIME	
Function:	Sets the fill time, time-out period.	
Default:	20 seconds	
	minimum	20 seconds
	maximum	999 seconds

Query Syntax

:SENSe[:PRESsure]:FILL:TIME?

Short form:	:SENS:FILL:TIME?
Function:	Asks for fill time, time-out period.
Response:	Integer corresponding fill time, time-out period in seconds

Description

If fill time, time-out settings are enabled, then the controller automatically switches to measure mode if the controller cannot achieve the set-point within the specified time-out setting.

:SENS:FILL:TIME:STAT

Command Syntax

:SENSe[:PRESsure]:FILL:TIME:STATe <Boolean>

Parameter:	Boolean	0	-	fill time-out disabled
		1	-	fill time-out enabled

Short form: :SENS:FILL:TIME:STAT
Function: Enables / disables the time-outs.
Default: OFF

Query Syntax

STATe?

Short form: :SENS:FILL:TIME:STAT?
Function: Asks if fill time, time-outs are enabled or disabled.
Response: 0 - fill time, time-out disabled
1 - fill time, time-out enabled

Description

If fill time, time-out settings are enabled, then the controller automatically switches to measure mode if the controller cannot achieve the set-point within the specified time-out setting.

:SENS:RANG

Command Syntax

:SENSe[:PRESsure]:RANGe <string>

Parameter: string range information
Short form: :SENS:RANG
Function: Used to select a range to be sensed.

Query Syntax

:SENSe[:PRESsure]:RANGe?

Short form: :SENS:RANG?
Function: Asks for currently sensed range.
Response: String representing selected pressure range.

Description

This command selects the pressure range to be used for returning the pressure reading, also see the SOURCe:RANGe command. Changing the source range also changes the sense range.

e.g. :SENSe:RANGe “2barg”

selects the 2 bar gauge range.

SOURce

The SOURce subsystem controls the pressure output of the instrument.

:SOUR:PRES:EFF

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SOURce[:PRESsure]:EFFort?

Short form: :SOUR:EFF?
Function: This query only command returns the effort the controller does to achieve the set-point.
Response: number representing controller effort.

Description

This query only command returns the effort the controller does to achieve the set-point.

:SOUR:INL

Command Syntax

:SOURce[:PRESsure]:INLimits <number>

Parameter: number in limits value as %full-scale

Short form: :SOUR:INL <number>
Function: The controller has an in-limits set-point indicator. This can generate a service request when the pressure is within limits for a set time period.

Default: 0.01 %FS
minimum 0 % FS
maximum 100 % FS

Query Syntax

:SOURce[:PRESsure]:INLimits?

Short form: :SOUR:INL?
Function: Query in-limits value
Response: Number representing in-limits value as %FS.

Description

Sets the in-limits value.

:SOUR:PRES:INL:TIME

Command Syntax

:SOURce[:PRESsure]:INLimits:TIME <number>

Parameter: number in-limits time in seconds

Short form: :SOUR:INL:TIME <number>
Function: Sets the time that the pressure has to be within limits before generating a service request.

Default: 2 seconds
minimum 2 seconds
maximum 999 seconds

Query Syntax

:SOURce[:PRESsure]:INLimits:TIME?

Short form: :SOUR:INL:TIME?
Function: Query in-limits timers.
Response: Number representing in-limits time in seconds.

Description

Sets the in-limits timer value.

:SOUR:PRES:LEV:IMM:AMPL

Command Syntax

:SOURce[:PRESsure][:LEVel][:IMMediate][:AMPLitude] <number>

Parameter: number Pressure in current units

Short form: SOUR <number>
Function: Set the pressure set-point

Default: 0.0

Query Syntax

:SOURce[:PRESsure][:LEVel][:IMMediate][:AMPLitude]?

Short form :SOUR?
Function: Programmable set-point value
Response: Pressure set-point in current units.

Description

This command sets the pressure set-point. Controllers must be ON before using this command. Use :OUTPut[:STATe] to turn the controller ON.

:SOUR:VENT

Command Syntax

:SOURce[:PRESsure][:LEVel][:IMMediate][:AMPLitude]:VENT <number>

Parameter: integer 0 - abort vent
1 - start vent

Short form: SOUR:VENT <integer>
Function: Vents the user system.

Default: 0

Query Syntax

:SOURce[:PRESsure][:LEVel][:IMMediate][:AMPLitude]:VENT?

Short form: :SOUR:VENT?
Function: Query status of vent

Response:
0 - vent OK
1 - vent in progress
2 - vent timed out
3 - vent outside limits
4 - vent aborted

Description

This command vents the user system; the command should be queried to get the status of the vent.

:SOUR:VENT:TIME

Command Syntax

:SOURce[:PRESsure][:LEVel][:IMMediate][:AMPLitude]:VENT:TIME <number>

Parameter:	integer	vent time-out in seconds
Short form:	SOUR:VENT:TIME <integer>	
Function:	Sets the vent time-out period.	
Default:	20 seconds	
	minimum	20 seconds
	maximum	999 seconds

Query Syntax

:SOURce[:PRESsure][:LEVel][:IMMediate][:AMPLitude]:VENT:TIME?

Short form:	:SOUR:VENT:TIME?
Function:	Query vent time-out
Response:	Vent time-out in seconds

Description

This command sets the time-out period for a system vent. If during a vent, this time-out expires then venting aborts and the vent status sets to `vent timed out', see SOUR:VENT.

:SOUR:RANG

Command Syntax

:SOURCe[:PRESsure]:RANGe <string>

Parameter: string range information

Short form: :SOUR:RANG
Function: Selects the control range.

Query Syntax

:SOURCe[:PRESsure]:RANGe?

Short form: :SOUR:RANGE?
Function: Asks for currently selected controller range.
Response: String representing selected sense range.

Description

This command selects the range to be used for controlling pressure.
e.g. :SOURe:RANGe “2barqa”
selects the 2 bar absolute range; the pressure units are always in bar.

Note: *Absolute ranges are quasi-absolute values, combining barometric and gauge sensor readings.*

:SOUR:RANG:AUTO

Command Syntax

:SOURCe[:PRESsure]:RANGe:AUTO <boolean>

Parameter:	Boolean	0	-	auto ranging disabled
		1	-	auto ranging enabled
Short form:	:SOUR:RANG:AUTO			
Function:	Selects auto ranging.			
Default:	0			

Query Syntax

:SOURCe[:PRESsure]:RANGe:AUTO?

Short form:	:SOUR:RANG:AUTO?		
Function:	Queries auto range selection.		
Response:	0	-	disabled
	1	-	enabled

Description

The manual range changing command :SOUR:PRES:RANGE stays operational but on receipt of a new set-point the range may change because of auto ranging. While auto range changes, any commands sent to the controller stay in a queue until the completion of range changing. When complete, and if enabled, an SRQ will be generated.

Note: *The SCPI auto ranging facility remains independent of the user interface. It is possible for the user interface to be set to manual range while the communications can be set to auto range changing.*

:SOUR:RANG:LOW

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SOURCe[:PRESsure]:RANGe:LOW?

Short form: :SOUR:RANGE:LOW?
Function: This query only command returns the lowest set-point pressure the controlled range can accept.
Response: String representing lowest pressure acceptable.

Description

Once a control range is selected this command can be used to check the lowest set-point available for that range.
e.g. a reply of:- :SOURe:RANGe:LOW “-1barg”
indicates that the current range can accept a minimum set-point of -1 bar; the pressure units are always in bar.

:SOUR:SLEW

Command Syntax

:SOURCE[:PRESsure]:SLEW <number>

Parameter: number rate in pressure units / second
Short form: SOUR:SLEW <number>
Function: Selects the pressure rate used when value rate is selected.
Default: 100

Query Syntax

:SOURCE[:PRESsure]:SLEW?

Short form: :SOUR:SLEW?
Function: Query rate value
Response: Rate value in selected units / second

Description

When the controller rate is selected as value, this command is used to set the controllers rate in selected units / second.

:SOUR:SLEW:MODE

Command Syntax

:SOURCE[:PRESsure]:SLEW:MODE <enumerated>

Parameter:	enumerated	MAXimum - maximum rate VALue - user selected rate

Short form: SOUR:SLEW:MODE <enumerated>
Function: Select the rate the controller should use to achieve set-point.
Default: MAXimum

Query Syntax

:SOURCE[:PRESsure]:SLEW:MODE?

Short form: :SOUR:SLEW:MODE?
Function: Query rate mode
Response: MAX for maximum rate
VAL for user defined rate

Description

The controller can operate in two rate modes - maximum and value. In maximum rate the controller tries to achieve set-point as quickly as possible. In value mode the controller achieves the set-point at a user selected rate.

:SOUR:SLEW:OVER

Command Syntax

:SOURCE[:PRESsure]:SLEW:OVERshoot[:STATe] <Boolean>

Parameter:	Boolean	0	-	overshoot not allowed
		1	-	overshoot allowed

Short form: SOUR:SLEW:OVER <Boolean>
Function: Selects pressure overshoot to `allowed' or `not allowed'.
Default: 1 - overshoot allowed

Query Syntax

:SOURCE[:PRESsure]:SLEW:OVERshoot[:STATe]?

Short form: :SOUR:SLEW:OVER?
Function: Query overshoot state
Response: 0 - overshoot not allowed
1 - overshoot allowed

Description

The controller can reach the set-point in one of two modes:

Overshoot `not allowed', the controller changes the pressure to near the set-point. The rate of pressure change slows when approaching the set-point to avoid overshoot.

Overshoot `allowed', the controller achieves set-point as fast as possible and, when approaching the set-point, may overshoot or undershoot.

STATus

The STATus subsystem supports the OPERation status register as defined in SCPI protocol.

:STAT:OPER:COND

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:STATus:OPERation:CONDition?

Short form: :STAT:OPER:COND?
Function: Query condition register
Response: Contents of condition register

Description
Returns the contents of the 16 bit condition register, see section on status reporting.

:STAT:OPER:ENAB

Command Syntax

:STATus:OPERation:ENABle <integer>

Parameter:	integer	16 bit value to set enable bits

Short form:	STAT:OPER:ENAB <integer>
Function:	Controls the status operation enable register.
Default:	0
	minimum 0
	maximum 32767

Query Syntax

:STATus:OPERation:ENABle?

Short form:	:STAT:OPER:ENAB?
Function:	Query enable register
Response:	16 bit value of enable register.

Description

Controls the bits that pass through the status reporting system, see status reporting section.

:STAT:OPER:EVEN

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:STATus:OPERation:[EVENt]?

Short form: :STAT:OPER?
Function: Query event register
Response: 16 bit value of event register.

Description

Reads contents of event register, see status reporting section.

:STAT:OPER:PRES:COND

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:STATus:PRESsure:OPERation:CONDition?

Short form: :STAT:OPER:PRES:COND?
Function: Reads the contents of the pressure condition register.
Response: Contents of pressure condition register.

Description

Returns the contents of the 16 bit pressure condition register, see section on status reporting.

:STAT:OPER:PRES:ENAB

Command Syntax

:STATus:OPERation:PRESsure:ENABle <integer>

Parameter: integer 16 bit value to set pressure enable bits

Short form: STAT:OPER:PRES:ENAB <integer>
Function: Controls the pressure status operation enable register.
Default: 0
minimum 0
maximum 32767

Query Syntax

:STATus:OPERation:PRESsure:ENABle?

Short form: :STAT:OPER:PRES:ENAB?
Function: Query enable register.
Response: 16 bit value of pressure enable register.

Description

Controls the bits that pass through the status reporting system, see status reporting section.

:STAT:OPER:PRES:EVEN

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:STATus:OPERation:PRESsure:[:EVENt]?

Short form: :STAT:OPER:PRES?
Function: Reads contents of pressure event register
Response: 16 bit value of pressure event register.

Description

Reads contents of pressure event register, see status reporting section.

SYSTem

The SYSTem subsystem consists of general purpose commands.

:SYST:ERR

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SYStem:ERRor?

Query :ERRor?
Short form: :SYST:ERR?
Function: Gets next error from the error queue

Response: The follow list of errors are available

-102, “Syntax error”
-104, “Data type error”
-108, “Parameter not allowed”
-109, “Missing parameter”
-110, “Command Header Error”
-111, “Header Separator Error”
-112, “Program mnemonic too long”
-113, “Undefined header”
-114, “Header suffix out of range”
-120, “Numeric data error”
-121, “Invalid character in number”
-123, “Exponent too large”
-124, “Too many digits”
-128, “Numeric data not allowed”
-130, “Suffix error”
-131, “Invalid suffix”
-134, “Suffix too long”
-138, “Suffix not allowed”
-140, “Character data error”
-141, “Invalid character data”
-144, “Character data too long”
-148, “Character data not allowed”
-150, “String data error”
-151, “Invalid string data”
-158, “String data not allowed”
-200, “Execution error”
-201, “Invalid while in local”
-202, “Settings lost due to rtl”
-220, “Parameter error”
-222, “Data out of range”
-223, “Too much data”
-224, “Illegal parameter value”
-310, “System error”
-350, “Queue overflow”
-400, “Query error”
201 , “Query only”
202 , “No query allowed”
203 , “Parameter(s) not expected”
207 , “Enumerated value not in union”
208 , “Illegal number of parameters”
210 , “Run out of memory handle”
211 , “Unit not matched”
212 , “Unit not required”

Description

This command queries the error queue which holds up to five errors. The instrument returns the message “No error” when no more errors are in the queue.

:SYST:VERS

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SYStem:VERSion?

Short form: :SYST:VERS?
Function: Returns the SCPI version.
Response: version number

Description

Replies with the SCPI version number.

UNIT

The UNIT sub-system configures the instrument's pressure measurement units.

:UNIT:PRES

Command Syntax

:UNIT[:PRESsure] <name>

Parameter: <name>

ATM
BAR
CMH2O
CMHG
FTH2O
FTH2O4
HPA
INH2O
INH2O4
INH2O60
INHG
KG/CM2
KG/M2
KPA
LB/FT2
MH2O
MHG
MMH2O
MMHG
MPA
PA
PSI
TORR
MBAR
USER1
USER2

Short form :UNIT <name>
Function: Selects pressure units

Query Syntax

:UNIT[:PRESsure]?

Short form: :UNIT?
Function: Query what pressure units are selected
Response: name as above

Description

This command selects the current pressure units; USER1 and USER2 are the user defined units.

:UNIT:DEF

Command Syntax

:UNIT[:PRESsure]:DEFine[x] <string>,<number>

Parameter:	string	ASCII representation of unit name.
	number	Conversion factor from Pascals to required units

Short form: :UNIT:DEF <string>,<number>

Function: There are two user defined units on the instrument. This command defines the name and conversion factor to use.

minimum	string	“”
	number	0.0
maximum	string	8 characters
	number	1.0e10

Default:	string	N/A
	number	1000.0

Query Syntax

:UNIT[:PRESsure]:DEFine?

Short form: :UNIT:DEF?

Function: Query conversion factor

Response: A string representing name and number corresponding to the conversion factor.

Description

This command defines the name and conversion factor for the special units. This conversion factor is from Pascals to the required units. An index of 1 is for USER1 unit and an index of 2 is for USER2 unit.

4.2 Standard Commands

The commands identified with * are SCPI standard commands.

*CLS

Command Syntax

***CLS**

Parameter: none
Short form *CLS
Function: This command clears the status registers in the status reporting system.

Query Syntax

n/a
Parameter:
Short form:
Function:

Description

Clears all event and condition register, see status reporting section.

*ESE

Command Syntax

*ESE <integer>

Parameter:	integer 8 bit value of enable mask
Short form:	*ESE <integer>
Function:	Sets the Standard Event Status enable register.
	minimum 0
	maximum 255
Default:	0

Query Syntax

*ESE?

Short form:	*ESE?
Function:	Query Standard Event Status Enable register.
Response:	8 bit integer of contents of Standard Event Status Enable register.

Description

See status reporting section.

*ESR

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

***ESR?**

Short form: *ESR?
Function: Queries the Standard Event Status Register
Response: 8 bit integer of contents of Standard Event Status register.

Description

See status reporting section.

:GTL

Command Syntax

:GTL

Parameter: none
Short form :GTL
Function: Go to local

Query Syntax

n/a
Parameter:
Short form:
Function:

Description

Takes the instrument out of local lockout mode; the key-pad on the instrument becomes active.

*IDN?

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

***IDN?**

Short form:	*IDN?
Function:	Queries the identification of the instrument.
Response:	A comma separated list containing manufacture, model, serial number and software version.

Description

Return identification

e.g., *IDN Druck,DPI515C,1234,01.00.00

:LLO

Command Syntax

:LLO

Parameter: none
Short form :LLO
Function: Local lockout

Query Syntax

n/a

Parameter:
Short form:
Function:

Description

Disables the instrument key-pad, the instrument cannot be put into local mode.

:LOC

Command Syntax

:LOC

Parameter: none
Short form :LOC
Function: Local mode

Query Syntax

n/a

Parameter:
Short form:
Function:

Description

Puts the instrument into local mode. The instrument will no longer be in remote mode and can be operated from the front panel.

:REM

Command Syntax

:REM

Parameter: none
Short form :REM
Function: Remote mode

Query Syntax

n/a

Parameter:
Short form:
Function:

Description

Puts the instrument into remote mode. The instrument will be in remote mode and cannot be operated from the front panel.

*RST

Command Syntax

***RST**

Parameter: none

Short form: *RST
Function: This command resets the operating parameters (units) to default values.

Query Syntax

n/a
Parameter:
Short form:
Function:

Description

Clears SCPI registers.

*SRE

Command Syntax

*SRE <integer>

Parameter:	integer	8 bit value of enable mask
Short form:	*SRE <integer>	
Function:	Sets the Service Request Enable register.	
	minimum	0
	maximum	255
Default:	0	

Query Syntax

*SRE?

Short form:	*SRE?
Function:	Query Service Request Enable register.
Response:	8 bit integer of contents of Service Request Enable register.

Description

See status reporting section.

*STB?

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

***STB?**

Short form: *STB?
Function: Queries the Status Register
Response: 8 bit integer of contents of Status register.

Description

See status reporting section.

5 Aeronautical Command Reference

The following lists of all the SCPI commands and queries for the aeronautical facility that apply to the instrument.

5.1 Command structure

Some of the commands in the following summary are enabled at specific times and conditions, most can be enabled at any time. The parameters column shows the states, values and data contained in a command. The command structure divides into sub-systems as follows:

Command sub-system

:SENSe - directs the instrument to measure aeronautical parameters.

:SOURce - directs the instrument to control the aeronautical parameters.

:UNIT - sets the aeronautical units for the instrument.

Command	Parameters	Comments
SENse		
:ALTitude?	<integer>	altitude reading
:SLEW?	<integer>	altitude per second reading
:MACH?	<integer>	Mach number
:SLEW?	<integer>	Mach number per second
:SPEed?	<integer>	airspeed reading
:SLEW?	<integer>	airspeed reading per second
SOURce		
:ALTitude		
[:LEVel]		
[:IMMediate]		
[:AMPLitude]		**altitude set-point**
:SLEW		**rate of climb (vertical speed)**
:MACH		
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<integer>	**Mach set-point**
:REFerence?		
:MODE		
:VALue	<integer>	**Rate of Mach**
:SPEed		
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<integer>	airspeed set-point
:SLEW	<integer>	rate of airspeed
UNIT		
:ALTitude	**<integer>**	**altitude units**
:SPEed	**<integer>**	**airspeed units**

Figure 5-1 Aeronautical Command and Query Summary

SENSe

The SENSe subsystem selects and configures the aeronautical measurement functions of the instrument.

:SENS:ALT

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SENSe:ALTitude?

Short form: :SENS:ALT?
Function: Queries the aeronautical sensor.
Response: Returns the altitude reading when the altitude range is selected.

Description

This query only command returns the aeronautical altitude value. The altitude range must be selected for a response from this command.

:SENS:ALT:SLEW

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SENSe:ALTitude:SLEW?

Short form: :SENS:ALT:SLEW?
Function: Queries the ROC (vertical speed) value.
Response: Returns the altitude rate of climb in units per second

Description

This query only command returns the aeronautical altitude rate of climb (ROC or vertical speed) in units per second.

:SENS:MACH

Command Syntax

n/a

Parameter:
Short form:
Function:
Default:

Query Syntax

:SENSe:MACH?

Short form:	:SENS:MACH?
Function:	Queries the Mach number reading
Response:	Returns the Mach number reading when Mach number range is selected.

Description

This query only command returns the aeronautical Mach number reading. The Mach number range must be selected for a response from this command.

:SENS:MACH:SLEW

Command Syntax

n/a

Parameter:
Short form:
Function:
Default:

Query Syntax

:SENSe:MACH:SLEW?

Short form:	:SENS:MACH:SLEW?
Function:	Queries the rate of change of Mach number
Response:	Returns the rate of change of Mach number in Mach number per second

Description

This query only command returns the Mach number rate of change in Mach number per second.

:SENS:SPE

Command Syntax

n/a

Parameter:
Short form:
Function:
Default:

Query Syntax

:SENSe:SPEed?

Short form: :SENS:SPE?
Function: Queries the airspeed value
Response: Returns airspeed value when airspeed range selected.

Description

This query only command returns the airspeed reading. The airspeed range must be selected for a response from this command.

:SENS:SPE:SLEW

Command Syntax

n/a
Parameter:

Short form:
Function:

Query Syntax

:SENSe:SPEed:SLEW?

Short form: :SENS:SPE:SLEW?
Function: Queries the rate of change of airspeed
Response: Returns the rate of change of airspeed in units per second

Description

This query only command returns the airspeed rate of change in units per second.

SOURce

The SOURce subsystem controls the aeronautical pressure output of the instrument.

CAUTION:
CHANGING TO THE AERONAUTICAL OPTION FROM NORMAL PRESSURE CONTROL CAN CAUSE DAMAGE TO SENSITIVE AERONAUTICAL INSTRUMENTS AND SENSORS. MAKE SURE THE SLEW SETTING IS BELOW THE MAXIMUM LIMIT FOR THE INSTRUMENT UNDER TEST.

:SOUR:ALT

Command Syntax

:SOURCe:ALTitude [:IMMediate][:AMPLitude] <number>

Parameter: number Altitude in current units

Short form: SOUR:ALT <number>
Function: Sets the altitude set-point
Default: 0.0

Query Syntax

:SOURce:ALT?

Short form: :SOUR:ALT?
Function: This queries the altitude set-point value
Response: Altitude set-point in current units

Description

This command sets the altitude set-point in the currently selected units.

:SOUR:ALT:SLEW

Command Syntax

:SOURCe:ALTitude:SLEW <number>

Parameter: number Altitude rate set-point in current units per second

Short form: SOUR:ALT:SLEW <number>
Function: Sets the altitude rate set-point

Default: 0.01

Query Syntax

:SOURce:ALT:SLEW?

Short form: :SOUR:ALT:SLEW?
Function: Queries the altitude rate set-point
Response: Altitude rate set-point in current units per second

Description

This command sets the altitude rate (ROC or vertical speed) set-point.

:SOUR:MACH:IMM:AMPL

Command Syntax

:SOURCe:MACH][:IMMediate][:AMPLitude] <number>

Parameter: number Mach set-point

Short form: SOUR:MACH <number>
Function: Sets the Mach number set-point
Default: 0.0

Query Syntax

:SOURce:MACH?

Short form: :SOUR:MACH?
Function: Queries the set-point Mach number
Response: Mach number set-point

Description

This command sets the Mach number set-point.

:SOUR:MACH:REF

Command Syntax

n/a
Parameter:
Short form:
Function:
Default:

Query Syntax

:SOURce[:MACH]:REFerence?

Short form: :SENS:MACH:REF?
Function: Queries the Mach number reference pressure
Response: Returns the Mach number reference pressure

Description

This query only command returns the Mach number reference pressure in the currently selected pressure units. This value can be a constant value or the barometric pressure (see MODE command).

:SOUR:MACH:REF:MODE

Command Syntax

SOURCe:MACH:REFerence:MODE <enumerated>

Parameter: enumerated type VAL – value specified by user
BARO – barometric pressure

Short form: SOUR:MACH:REF <enumerated>
Function: Sets the source of the Mach number reference pressure.

Default: VAL

Query Syntax

:SOURce:MACH:REFerence?

Short form: :SOUR:MACH:REF?
Function: Queries the Mach number mode

Response: VAL or BARO

Description

This command sets source of the reference pressure used to calculated Mach number. VAL is for a user defined value (see VALue command). If a barometer is fitted BARO selects the barometric pressure reading.

:SOUR:MACH:REF:VAL

Command Syntax

:SOURCe:MACH:REFerence:MODE:VALue <number>

Parameter: number Mach number reference pressure
Short form: SOUR:MACH:REF:VAL <number>
Function: Sets the manually entered Mach number reference pressure.

Default: 1013.25 mbar

Query Syntax

:SOURce:MACH:REFerence:VALue?

Short form: :SOUR:MACH:REF:VAL?
Function: Queries the Mach number reference value
Response: Manual Mach number reference pressure value

Description

If a constant value Mach number reference pressure is selected (see MODE command) then this command is used to change the value used for the Mach number reference pressure.

:SOUR:SPE

Command Syntax

:SOURCe:SPEed[:IMMediate][:AMPLitude] <number>

Parameter: number Airspeed in current units

Short form: SOUR:SPE <number>
Function: Sets the airspeed set-point
Default: 0.0

Query Syntax

:SOURce:SPE?

Short form: :SOUR:SPE?
Function: Queries the airspeed set-point value
Response: Airspeed set-point in current units

Description

This command sets the airspeed set-point in the currently selected units.

:SOUR:SPE:SLEW

Command Syntax

:SOURCe:SPEed:SLEW <number>

Parameter: number Airspeed rate set-point in current units per second

Short form: SOUR:SPE:SLEW <number>
Function: Sets the airspeed rate set-point
Default: 0.01

Query Syntax

:SOURce:SPE:SLEW?

Short form: :SOUR:SPE:SLEW?
Function: Queries the airspeed rate set-point
Response: Airspeed rate set-point in current units per second

Description

This command sets the airspeed rate set-point in the currently selected units per second.

UNIT

The UNIT subsystem configures the instrument's aeronautical measurement units.

:UNIT:ALT

Command Syntax

:UNIT:ALTitude <FT | M>

Parameter:	enumerated	FT – feet M - metres

Short form: :UNIT:ALT <enumerated>
Default: FT
Function: Sets the altitude units

Query Syntax

:UNIT:ALTitude?

Function: Queries the altitude units
Short form: :UNIT:ALT?
Response: Altitude units FT or M

Description

This command sets the altitude units.

:UNIT:SPE

Command Syntax

:UNIT:SPEed <KN | KPH | MPH>

Parameter:	enumerated	KN – knots KPH – kilometres per hour MPH – miles per hour
Short form:	:UNIT:SPE <enumerated>	
Function:	Sets the airspeed units	
Default:	KN	

Query Syntax

:UNIT:SPEed?

Short form:	:UNIT:SPE?
Function:	Queries the airspeed units
Response:	Airspeed units KN, KPH or MPH

Description

This command sets the airspeed units.

6 Sensor Calibration Module Command Reference

The following lists of all the SCM commands and queries that apply to the instrument.

6.1 Command structure

Some of the commands in the following summary are enabled at specific times and conditions, most can be enabled at any time. The parameters column shows the states, values and data contained in a command. The command structure divides into subsystems as follows:

Command subsystem

:SENSe - directs the instrument to measure parameters.

:CALibration - directs the instrument to calibrate the SCM parameters.

Command	Parameters	Comments
SENse		
:SCM?		
:MODE	<enumerated>	<OFF | MA | MVA | MVC | V>
CALibration		
:SCM		
:ZERO	**<integer>**	

Figure 6-1 SCM Command and Query Summary

SENSe

The SENSe subsystem selects and configures the measurement functions of the SCM.

:SENS:SCM

Command Syntax

n/a
Parameter:
Short form:
Function:

Query Syntax

:SENSe:SCM?

Short form: :SENS:SCM?
Function: Queries the SCM reading
Response: Returns the currently selected SCM channel reading.

Description

This query only command returns the SCM reading for the channel selected by the MODE command

:SENS:SCM:MODE

Command Syntax

:SENSe:SCM:MODE <OFF | MA | MVA | MVC | V>

Parameter: SCM channel selection

OFF – SCM turned off
MA – current in mA
MVA – absolute voltage in mV
MVC – compensated voltage in mV
V – voltage in V

Short form: :SENS:SCM:MODE
Function: Selects SCM channel.
Default: OFF

Minimum	N/A
Maximum	N/A

Query Syntax

:SENSe:SCM:MODE?

Short form: :SENS:SCM:MODE?
Function: Queries the selected SCM channel
Response: Returns an enumerated value in the same format as the parameter.

Description

The SCM option consists of a voltmeter and ammeter. This command selected the required SCM function. MA selects the ammeter. MVA selects the millivoltmeter read as an absolute voltage. MVC selects the millivoltmeter but compensates for the 10 V excitation voltage for ratiometric measurements. V selects the voltmeter. OFF turn the SCM option off.

CALibration

The CALibration subsystem calibrates the signal conditioning module.

:CAL:SCM

Command Syntax

:CALibration:SCM:ZERO <integer>

Parameter:

Short form: :CAL:SCM:ZERO
Function: Zeroes the current SCM reading
Parameter: integer 0 – stop zero
1 – start zero

Query Syntax

:CALibration:SCM:ZERO ?

Short form: :CAL:SCM:ZERO?
Function: Queries the state of SCM zero
Response: Integer value representing zero status
0 – Zero complete and OK
1 – Zero offset too large
2 – Still performing zero - Sampling reading

Default: 0
Minimum 0
Maximum 2

Description

The zero function allows any zero drift to be removed from the SCM reading.

7 ERRORS

Negative error numbers are used for standard SCPI errors. Positive error numbers are device specific errors. Following the error number, a message describes the error. An error, when detected, is held in the error queue.
When SYST:ERR? is sent any error in the error queue sets the error bit in the event status register.

Error numbers

Each error detected causes an error number with an error message to be returned as follows:

Error code	Error message	Error code	Error message
-102	-102, Syntax error	-134	-134, Suffix too long
-104	-104, Data type error	-138	-138, Suffix not allowed
-108	-108, Parameter not allowed	-140	-140, Character data error
-109	-109, Missing parameter	-141	-141, Invalid character data
-110	-110, Command Header Error	-144	-144, Character data too long
-111	-111, Header Separator Error	-148	-148, Character data not allowed
-112	-112, Program mnemonic too long	-150	-150, String data error
-113	-113, Undefined header	-151	-151, Invalid string data
-114	-114, Header suffix out of range	-158	-158, String data not allowed
-120	-120, Numeric data error	-160	-160, Block data error
-121	-121, Invalid character in number	-161	-161, Invalid block data
-123	-123, Exponent too large	-168	-168, Block data not allowed
-124	-124, Too many digits	-170	-170, Expression error
-128	-128, Numeric data not allowed	-171	-171, Invalid expression
-130	-130, Suffix error	-178	-178, Expression data not allowed
-131	-131, Invalid suffix		

Table 7-1 Errors -100 to -199

Error code	Error message	Error code	Error message
-200	-200, Execution error	-256	-256, File name not found
-220	-220, Parameter error	-257	-257, File name error
-221	-221, Settings conflict	-260	-260, Expression Error
-222	-222, Data out of range	-261	-261, Math error in expression
-223	-223, Too much data	-280	-280, Program error
-224	-224, Illegal parameter value	-281	-281, Cannot create program
-230	-230, Data corrupt or stale	-282	-282, Illegal program name
-231	-231, Data questionable	-283	-283, Illegal variable name
-240	-240, Hardware error	-284	-284, Program currently running
-241	-241, Hardware missing	-285	-285, Program syntax error
-254	-254, Media full	-286	-286, Program runtime error

Table 7-2 Errors -200 to -299

Error code	Error message
-300	-300, Device-specific error
-310	-310, System error
-311	-311, Memory error
-313	-313, Calibration memory lost
-314	-314, Save/recall memory lost
-315	-315, Configuration memory lost
-350	-350, Queue overflow
-400	-400, Query error

Table 7-3 Errors -300 to -400

Error code	Error message
201	201, Query only
202	202, No query allowed
203	203, Paramerter(s) not expected
208	208, Illegal number of parameters
210	210, Run out of memory handle
211	211, Unit not matched
212	212, Unit not required

Table 7-4 Errors +201 to +212