

Table 6-8. Command Summary

<u>Command</u>	<u>Function</u>	<u>Page</u>	<u>Command</u>	<u>Function</u>	<u>Page</u>
QCLS	Clear Interface Cmd	24	HTR?	Heater Output Query	32
QESE	Event Status Enable Cmd	24	HTRST?	Heater Status Query	32
QESE?	Event Status Enable Query	24	IEEE	IEEE Interface Parameter Cmd	32
QESR?	Event Status Register Query	24	IEEE?	IEEE Interface Parameter Query	32
QIDN?	Identification Query	24	INCRV	Input Curve Number Cmd	32
QOPC	Operation Complete Cmd	25	INCRV?	Input Curve Number Query	32
QOPC?	Operation Complete Query	25	INTYPE	Input Type Parameter Cmd	33
QRST	Reset Instrument Cmd	25	INTYPE?	Input Type Parameter Query	33
QSRE	Service Request Enable Cmd	25	KEYST?	Keypad Status Query	33
QSRE?	Service Request Enable Query	25	KRDG?	Kelvin Reading Query	33
QSTB?	Status Byte Query	25	LDAT?	Linear Equation Data Query	34
QTST?	Self-Test Query	26	LINEAR	Input Linear Equation Cmd	34
QWAI	Wait-To-Continue Cmd	26	LINEAR?	Input Linear Equation Query	34
ALARM	Input Alarm Parameter Cmd	26	LOCK	Front Panel Keyboard Lock Cmd	34
ALARM?	Input Alarm Parameter Query	26	LOCK?	Front Panel Keyboard Lock Query	35
ALARMST?	Input Alarm Status Query	26	MDAT?	Min/Max Data Query	35
ALMRST	Reset Alarm Status Cmd	27	MNMX	Min/Max Input Function Cmd	35
ANALOG	Analog Output Parameter Cmd	27	MNMX?	Min/Max Input Function Query	35
ANALOG?	Analog Outputs Parameter Query	27	MNMXRST	Min/Max Function Reset Cmd	35
AOUT?	Analog Output Data Query	27	MODE	Set Local/Remote Mode	35
BAUD	RS-232 Baud Rate Cmd	27	MODE?	Query Local/Remote Mode	36
BAUD?	RS-232 Baud Rate Query	27	MOUT	Control Loop MHP Output Cmd	36
BEEP	System Beeper Cmd	28	MOUT?	Control Loop MHP Output Query	36
BEEP?	System Beeper Query	28	PID	Control Loop PID Values Cmd	36
BRIGT	Display Brightness Command	28	PID?	Control Loop PID Values Query	36
BRIGT?	Display Brightness Query	28	RAMP	Control Loop Ramp Cmd	37
CMODE	Control Loop Mode Cmd	28	RAMP?	Control Loop Ramp Query	37
CMODE?	Control Loop Mode Query	28	RAMPST?	Control Loop Ramp Status Query	37
CRDG?	Celsius Reading Query	28	RANGE	Heater Range Cmd	37
CRVDEL	Delete User Curve Cmd	29	RANGE?	Heater Range Query	37
CRVHDR	Curve Header Cmd	29	RDGST?	Input Status Query	38
CRVHDR?	Curve Header Query	29	RELAY	Relay Control Parameter Cmd	38
CRVPT	Curve Data Point Cmd	29	RELAY?	Relay Control Parameter Query	38
CRVPT?	Curve Data Point Query	30	RELAYST?	Relay Status Query	38
CSET	Control Loop Parameter Cmd	30	REV?	Input Firmware Revision Query	38
CSET?	Control Loop Parameter Query	30	SCAL	Generate SoftCal Curve Cmd	39
DFLT	Factory Defaults Cmd	30	SETP	Control Loop Setpoint Cmd	39
DISPFLD	Displayed Field Cmd	30	SETP?	Control Loop Setpoint Query	39
DISPFLD?	Displayed Field Query	31	SRDG?	Sensor Units Reading Query	39
EMUL	330 Emulation Mode Cmd	31	TEMP?	Room-Temp Comp. Temp. Query	40
EMUL?	330 Emulation Mode Query	31	TUNEST?	Control Loop 1 Tuning Query	40
FILTER	Input Filter Parameter Cmd	31	ZONE	Control Loop Zone Table Cmd	40
FILTER?	Input Filter Parameter Query	31	ZONE?	Control Loop Zone Table Query	40

6.3.1 Interface Commands (Alphabetical Listing)

QCLS Clear Interface Command

Input: QCLS [term]

Remarks: Clears the bits in the Status Byte Register and Standard Event Status Register and terminates all pending operations. Clears the interface, but *not* the controller. The related controller command is **QRST**.

QESE Event Status Enable Register Command

Input: QESE <bit weighting>[term]

Format: nnn

Remarks: Each bit is assigned a bit weighting and represents the enable/disable mask of the corresponding event flag bit in the Standard Event Status Register. To enable an event flag bit, send the command **QESE** with the sum of the bit weighting for each desired bit. Refer to Paragraph 6.1.3.2 for a list of event flags.

Example: To enable event flags 0, 3, 4, and 7, send the command **QESE 143[term]**. 143 is the sum of the bit weighting for each bit.

Bit	Bit Weighting	Event Name
0	1	OPC
3	8	DDE
4	16	EXE
7	<u>128</u>	PON
	143	

QESE? Event Status Enable Register Query

Input: QESE? [term]

Returned: <bit weighting>[term]

Format: nnn Refer to Paragraph 6.1.3.2 for a list of event flags.

QESR? Standard Event Status Register Query

Input: QESR? [term]

Returned: <bit weighting>

Format: nnn

Remarks: The integer returned represents the sum of the bit weighting of the event flag bits in the Standard Event Status Register. Refer to Paragraph 6.1.3.2 for a list of event flags.

QIDN? Identification Query

Input: QIDN? [term]

Returned: <manufacturer>,<model>,<serial>,<date>[term]

Format: aaaa,aaaaaaaa,aaaaaa,mmdyy

<manufacture> Manufacturer ID

<model> Instrument model number

<serial> Serial number

<date> Instrument firmware revision date

Example: LSCI ,MODEL332 ,123456 ,020301

QOPC Operation Complete Command

Input: QOPC [**term**]

Remarks: Generates an Operation Complete event in the Event Status Register upon completion of all pending selected device operations. Send it as the last command in a command string.

QOPC? Operation Complete Query

Input: QOPC? [**term**]

Returned: 1[**term**]

Remarks: Places a "1" in the controller output queue upon completion of all pending selected device operations. Send as the last command in a command string. *Not* the same as QOPC.

QRST Reset Instrument Command

Input: QRST [**term**]

Remarks: Sets controller parameters to power-up settings.

QSRE Service Request Enable Register Command

Input: QSRE <bit weighting>[**term**]

Format: nnn

Remarks: Each bit has a bit weighting and represents the enable/disable mask of the corresponding status flag bit in the Status Byte Register. To enable a status flag bit, send the command QSRE with the sum of the bit weighting for each desired bit. Refer to Paragraph 6.1.3.1 for a list of status flags.

Example: To enable status flags 0, 2, 4, and 6, send the command QSRE 89[**term**]. 89 is the sum of the bit weighting for each bit.

<u>Bit</u>	<u>Bit Weighting</u>	<u>Event Name</u>
0	1	New A&B
3	8	Alarm
4	16	Error
6	<u>64</u>	SRQ
	89	

QSRE? Service Request Enable Register Query

Input: QSRE? [**term**]

Returned: <bit weighting>[**term**]

Format: nnn Refer to Paragraph 6.1.3.1 for a list of status flags.

QSTB? Status Byte Query

Input: QSTB? [**term**]

Returned: <bit weighting>[**term**]

Format: nnn

Remarks: Acts like a serial poll, but does not reset the register to all zeros. The integer returned represents the sum of the bit weighting of the status flag bits that are set in the Status Byte Register. Refer to Paragraph 6.1.3.1 for a list of status flags.

QTST? Self-Test Query

Input: QTST? [term]

Returned: <status>[term]

Format: n
<status> 0 = no errors found, 1 = errors found

Remarks: The Model 332 reports status based on test done at power up.

QWAI Wait-to-Continue Command

Input: QWAI [term]

Remarks: This command is not supported in the Model 332.

ALARM Input Alarm Parameter Command

Input: ALARM <input>, <off/on>, <source>, <high value>, <low value>, <deadband>, <latch enable>[term]

Format: a,n,n,±nnnnnn,±nnnnnn,±nnnnnn,n
 <input> Specifies which input to configure: A or B.
 <off/on> Determines whether the instrument checks the alarm for this input, where 0 = off and 1 = on.
 <source> Specifies input data to check. Valid entries: 1 = kelvin, 2 = Celsius, 3 = sensor units, 4 = linear data.
 <high value> Sets the value the source is checked against to activate the high alarm.
 <low value> Sets the value the source is checked against to activate low alarm.
 <deadband> Sets the value that the source must change outside of an alarm condition to deactivate an unlatched alarm.
 <latch enable> Specifies a latched alarm (remains active after alarm condition correction) where 0 = off (no latch) and 1 = on.

Remarks: Configures the alarm parameters for an input.

Example: **ALARM A,0[term]** – Turns off alarm checking for Input A.
ALARM B,1,1,270.0,0,0,1[term] – Turns on alarm checking for input B, activates high alarm if kelvin reading is over 270, and latches the alarm when kelvin reading falls below 270.

ALARM? Input Alarm Parameter Query

Input: ALARM? <input>[term]

Format: a
 <input> A or B

Returned: <off/on>, <source>, <high value>, <low value>, <deadband>, <latch enable> [term]

Format: n,n,±nnnnnn,±nnnnnn,±nnnnnn,n (Refer to command for description)

ALARMST? Input Alarm Status Query

Input: ALARMST? <input>[term]

Format: a
 <input> A or B

Returned: <high state>, <low state>[term]

Format: n,n
 <high state> 0 = Off, 1 = On
 <low state> 0 = Off, 1 = On

ALMRST Reset Alarm Status Command**Input:** **ALMRST** [**term**]**Remarks:** Clears both the high and low status of all alarms, including latching alarms.

ANALOG Analog Output Parameter Command**Input:** **ANALOG** <bipolar enable>, <mode>, <input>, <source>, <high value>, <low value>, <manual value>[**term**]**Format:** n,n,a,n,±nnnnnn,±nnnnnn,±nnnnnn

<bipolar enable> Specifies analog output is 0 = positive output only or 1 = bipolar.

<mode> Specifies data the analog output monitors. Valid entries: 0 = off, 1 = input, 2 = manual, 3 = loop.

<input> Specifies which input to monitor if <mode> = 1.

<source> Specifies input data. Valid entries: 1 = kelvin, 2 = Celsius, 3 = sensor units, 4 = linear equation.

<high value> If <mode> is 1, this parameter represents the data at which the analog output reaches +100% output.

<low value> If <mode> is 1, this parameter represents the data at which the analog output reaches -100% output if bipolar, or 0% output if positive only.

<manual value> If <mode> is 2, this parameter is the output of the analog output.

Example: **ANALOG 0,1,A,1,100.0,0.0[term]** – Sets analog output to monitor Input A kelvin reading with 100.0 K at +100% output (+10.0 V) and 0.0 K at 0% output (0.0 V).

ANALOG? Analog Output Parameter Query**Input:** **ANALOG?** [**term**]**Returned:** <bipolar enable>, <mode>, <input>, <source>, <high value>, <low value>, <manual value>[**term**]**Format:** n,n,a,n,±nnnnnn,±nnnnnn,±nnnnnn (Refer to command for definition)

AOUT? Analog Output Data Query**Input:** **AOUT?** [**term**]**Returned:** <analog output>[**term**]**Format:** ±nnn.n**Remarks:** Returns the percentage of output of the analog output. Most often used for input or loop modes when the output value is set by the instrument. Resolution is 0.5%.

BAUD RS-232 Baud Rate Command**Input:** **BAUD** <bps> [**term**]**Format:** n
<bps> Specifies Baud rate: 0 = 300 Baud, 1 = 1200 Baud, 2 = 9600 Baud.

BAUD? RS-232 Baud Rate Query**Input:** **BAUD?****Returned:** <bps> [**term**]**Format:** n (Refer to command for description)

BEEP Alarm Beeper Command**Input:** **BEEP** <state>[term]**Format:** n
<state> 0 = Off, 1 = On.**Remarks:** Enables or disables system beeper sound when an alarm condition is met.

BEEP? Alarm Beeper Query**Input:** **BEEP?****Returned:** <state>[term]**Format:** n (Refer to command for description)

BRIGT Display Brightness Command**Input:** **BRIGT** <bright>[term]**Format:** n
<bright> 0 = 25%, 1 = 50%, 2 = 75%, 3 = 100%. Default = 2.

BRIGT? Display Brightness Query**Input:** **BRIGT?** [term]**Returned:** <bright>[term]**Format:** n (Refer to command for description)

CMODE Control Loop Mode Command**Input:** **CMODE** <loop>, <mode>[term]**Format:** n,n
<loop> Specifies which loop to configure: 1 or 2.
<mode> Specifies the control mode. Valid entries: 1 = Manual PID, 2 = Zone,
3 = Open Loop, 4 = AutoTune PID, 5 = AutoTune PI, 6 = AutoTune P.**Example:** **CMODE 1,4[term]** – Control Loop 1 uses PID AutoTuning.

CMODE? Control Loop Mode Query**Input:** **CMODE?** <loop>[term]**Format:** n
<loop> Specifies which loop to query: 1 or 2.**Returned:** <mode>[term]**Format:** n (Refer to command for description)

CRDG? Celsius Reading Query**Input:** **CRDG?** <input>[term]**Format:** a
<input> A or B**Returned:** <temp value>[term]**Format:** ±nnnnnn**Remarks:** Also see the RDGST? command.

CRVDEL Curve Delete Command

Input: CRVDEL <curve>[term]

Format: nn
 <curve> Specifies a user curve to delete. Valid entries: 21–41.

Example: CRVDEL 21[term] – Deletes User Curve 21.

CRVHDR Curve Header Command

Input: CRVHDR <curve>, <name>, <SN>, <format>, <limit value>, <coefficient>[term]

Format: nn,aaaaaaaaaaaaaaaa,aaaaaaaaa,n,±nnn.nnn,n
 <curve> Specifies which curve to configure. Valid entries: 21–41.
 <name> Specifies curve name. Limited to 15 characters.
 <SN> Specifies the curve serial number. Limited to 10 characters.
 <format> Specifies the curve data format. Valid entries: 1 = mV/K, 2 = V/K, 3 = Ω/K, 4 = log Ω/K.
 <limit value> Specifies the curve temperature limit in kelvin.
 <coefficient> Specifies the curves temperature coefficient. Valid entries: 1 = negative, 2 = positive.

Remarks: Configures the user curve header.

Example: CRVHDR 21,DT-470,00011134,2,325.0,1[term] – Configures User Curve 21 with a name of DT-470, serial number of 00011134, data format of volts versus kelvin, upper temperature limit of 325 K, and negative coefficient.

CRVHDR? Curve Header Query

Input: CRVHDR? <curve>[term]

Format: nn
 <curve> Valid entries: 1–41.

Returned: <name>, <SN>, <format>, <limit value>, <coefficient>[term]

Format: aaaaaaaaaaaaaaaaa,aaaaaaaaa,n,±nnn.nnn,n (Refer to command for description)

CRVPT Curve Data Point Command

Input: CRVPT <curve>, <index>, <units value>, <temp value>[term]

Format: nn,nnn,±nnnnnnn,±nnnnnnn
 <curve> Specifies which curve to configure. Valid entries: 21–41.
 <index> Specifies the points index in the curve. Valid entries: 1–200.
 <units value> Specifies sensor units for this point to 6 digits.
 <temp value> Specifies the corresponding temperature in kelvin for this point to 6 digits.

Remarks: Configures a user curve data point.

Example: CRVPT 21,2,0.10191,470.000,N[term] – Sets User Curve 21 second data point to 0.10191 sensor units and 470.000 K.

CRVPT? Curve Data Point Query**Input:** CRVPT? <curve>, <index>[term]**Format:** nn,nnn

<curve> Specifies which curve to query: 1–41.

<index> Specifies the points index in the curve: 1–200.

Returned: <units value>, <temp value>[term]**Format:** ±nnnnnnnn,±nnnnnnnn (Refer to command for description)**Remarks:** Returns a standard or user curve data point.

CSET Control Loop Parameter Command**Input:** CSET <loop>, <input>, <units>, <powerup enable>, <current/power>[term]**Format:** n,a,n,n,n

<loop> Specifies which loop to configure: 1 or 2.

<input> Specifies which input to control from: A or B.

<units> Specifies setpoint units. Valid entries: 1 = kelvin, 2 = Celsius, 3 = sensor units.

<powerup enable> Specifies whether the control loop is on or off after power-up, where 0 = powerup enable off and 1 = powerup enable on.

<current/power> Specifies whether the heater output displays in current or power. Valid entries: 1 = current or 2 = power.

Example: CSET 1,A,1,1[term] – Control Loop 1 controls off of Input A with setpoint in kelvin.

CSET? Control Loop Parameter Query**Input:** CSET? <loop>[term]**Format:** n

<loop> Specifies which loop to query: 1 or 2.

Returned: <input>, <units>, <powerup enable>, <current/power>[term]**Format:** a,n,n,n (Refer to command for description)

DFLT Factory Defaults Command**Input:** DFLT 99[term]**Remarks:** Sets all configuration values to factory defaults and resets the instrument. The "99" is included to prevent accidentally setting the unit to defaults.

DISPFLD Displayed Field Command**Input:** DISPFLD <field>, <item>, <source>[term]**Format:** n,n,n

<field> Specifies field to configure: 1–4.

<item> Specifies item to display in the field: 0 = Off, 1 = Input A, 2 = Input B, 3 = Setpoint, 4 = Heater Output, 5 = Heater Bar.

<source> If item is 1 or 2, specifies input data to display. Valid entries: 1 = kelvin, 2 = Celsius, 3 = sensor units, 4 = linear data, 5 = minimum data, and 6 = maximum data.

Example: DISPFLD 2,1,1[term] – Displays kelvin reading for Input A in display field 2.

DISPFLD? Displayed Field Query

Input: **DISPFLD?** <field>[term]
Format: n
 <field> Specifies field to query: 1–4.
Returned: <item>, <source>[term]
Format: n,n (Refer to command for description)

EMUL 330 Emulation Mode Command

Input: **EMUL** <off/on>[term]
Format: n
 <off/on> Specifies whether 330 Emulation Mode is 0 = Off or 1 = On. Default = 0.
Remarks: The 330 Emulation Mode allows the remote interface of the Model 332 to be compatible with Model 330 commands. The 330 Emulation Mode only affects remote operation; front panel operation of the Model 332 is not changed. In 330 Emulation Mode, curve locations are mapped to match Model 330 locations. For example, the DT-500-D Curve, found at curve location 3 in the Model 332, is mapped to location 0 when in 330 mode. This applies to the following remote commands: ACUR, ACUR?, BCUR, BCUR?. The following Model 330 commands are not supported in 330 Emulation Mode: CUID?, CURV, CURV?, ECUR, KCUR, and SCAL.

EMUL? 330 Emulation Mode Query

Input: **EMUL?** [term]
Returned: <off/on >[term]
Format: n (Refer to command for description)

FILTER Input Filter Parameter Command

Input: **FILTER** <input>, <off/on>, <points>, <window>[term]
Format: a,n,nn,nn
 <input> Specifies input to configure: A or B.
 <off/on> Specifies whether the filter function is 0 = Off or 1 = On.
 <points> Specifies how many data points the filtering function uses. Valid range = 2 to 64.
 <window> Specifies what percent of full scale reading limits the filtering function. Reading changes greater than this percentage reset the filter. Valid range = 1 to 10%.
Example: **FILTER B,1,10,2[term]** – Filter input B data through 10 readings with 2% of full scale window.

FILTER? Input Filter Parameter Query

Input: **FILTER?** <input>[term]
Format: a
 <input> Specifies input to query: A or B.
Returned: <off/on >, <points>, <window>[term]
Format: n,nn,nn (Refer to command for description)

HTR? Heater Output Query**Input:** HTR? [term]**Returned:** <heater value>[term]**Format:** +nnn.n

<heater value> Loop 1 heater output in percent (%). Use AOUT? for Loop 2.

HTRST? Heater Status Query**Input:** HTRST? [term]**Returned:** <error code>[term]**Format:** n

<error code> Heater error code: 0 = no error, 1 = heater open load, 2 = heater short.

IEEE IEEE-488 Interface Parameter Command**Input:** IEEE <terminator>, <EOI enable>, <address>[term]**Format:** n,n,nn

<terminator> Specifies the terminator. Valid entries: 0 = <CR><LF>, 1 = <LF><CR>, 2 = <LF>, 3 = no terminator (must have EOI enabled).

<EOI enable> Sets EOI mode: 0 = enabled, 1 = disabled.

<address> Specifies the IEEE address: 1–30. (Address 0 and 31 are reserved.)

Example: IEEE 0,0,4[term] – After receipt of the current terminator, the instrument uses EOI mode, uses <CR><LF> as the new terminator, and responds to address 4.

IEEE? IEEE-488 Interface Parameter Query**Input:** IEEE? [term]**Returned:** <terminator>, <EOI enable>, <address>[term]**Format:** n,n,nn (Refer to command for description)

INCRV Input Curve Number Command**Input:** INCRV <input>, <curve number>[term]**Format:** a,nn

<input> Specifies which input to configure: A or B.

<curve number> Specifies which curve the input uses. If specified curve parameters do not match the input, the curve number defaults to 0. Valid entries: 0 = none, 1–20 = standard curves, 21–41 = user curves.

Remarks: Specifies the curve an input uses for temperature conversion.**Example:** INCRV A,23[term] – Input A uses User Curve 23 for temperature conversion.

INCRV? Input Curve Number Query**Input:** INCRV? <input>[term]**Format:** a

<input> Specifies which input to query: A or B.

Returned: <curve number>[term]**Format:** nn (Refer to command for description)

INTYPE

Input Type Parameter Command

Input: INTYPE <input>, <sensor type>, <compensation>[term]

Format: a,n,n

<input>

Specifies input to configure: A or B.

<sensor type>

Specifies input sensor type. Valid entries:

0 = Silicon Diode

8 = NTC RTD 75mV 75 Ω

1 = GaAlAs Diode

9 = NTC RTD 75mV 750 Ω

2 = Platinum 100/250 Ω

10 = NTC RTD 75mV 7.5 k Ω

3 = Platinum 100/500 Ω

11 = NTC RTD 75mV 75 k Ω

4 = Platinum 1000 Ω

12 = NTC RTD 75mV Auto

5 = NTC RTD 75mV 7.5 k Ω

6 = Thermocouple 25 mV

7 = Thermocouple 50 mV

<compensation> Specifies input compensation where 0 = off and 1 = on. Reversal for thermal EMF compensation if input is resistive, room compensation if input is thermocouple. Always 0 if input is a diode.

Remarks: Sensor type NTC RTD 75mV 7.5k Ω listed twice to maintain compatibility with Model 331 INTYPE command.

Example: INTYPE A,0,0[term] – Sets Input A sensor type to silicon diode.

INTYPE?

Input Type Parameter Query

Input: INTYPE? <input>[term]

Format: a

<input> Specifies input to query: A or B.

Returned: <sensor type>, <compensation>[term]

Format: n,n (Refer to command for description)

KEYST?

Keypad Status Query

Input: KEYST? [term]

Returned: <keypad status>[term]

Format: n 1 = key pressed, 0 = no key pressed.

Remarks: Returns keypad status since the last KEYST?. KEYST? returns 1 after initial power-up.

KRDG?

Kelvin Reading Query

Input: KRDG? <input>[term]

Format: a

<input> Specifies which input to query: A or B.

Returned: <kelvin value>[term]

Format: \pm nnnnnn

Remarks: Also see the RDGST? command.

LDAT? Linear Equation Data Query

Input: **LDAT?** <input>[term]

Format: a
 <input> Specifies which input to query: A or B.

Returned: <linear value>[term]

Format: ±nnnnnn

Remarks: Also see the RDGST? command.

LINEAR Input Linear Equation Parameter Command

Input: **LINEAR** <input>, <equation>, <varM value>, <X source>, <B source>, <varB value>[term]

Format: a,n,±nnnnnn,n,n,±nnnnnn

<input> Specifies input to configure: A or B.

<equation> Specifies linear equation to use.
 Valid entries: 1 = ($y = mx + b$), 2 = ($y = m(x + b)$).

<varM value> Specifies a value for m in the equation.

<X source > Specifies input data to use. Valid entries: 1 = kelvin, 2 = Celsius, 3 = sensor units.

<B source > Specifies what to use for b in the equation. To use a setpoint, set its units to the same type specified in <X source>. Valid entries: 1 = a value, 2 = +SP1, 3 = -SP1, 4 = +SP2, 5 = -SP2.

<varB value> Specifies a value for b in the equation if <B source> is 1.

Example: **LINEAR A,1,1.0,1,3[term]** – The linear data for Input A is calculated from the kelvin reading of the input using the equation: $y = 1.0 * x - SP1$.

LINEAR? Input Linear Equation Parameter Query

Input: **LINEAR?** <input>[term]

Format: a
 <input> Specifies which input to query: A or B.

Returned: <equation>, <varM value>, <X source>, <B source>, <varB value>[term]

Format: n,±nnnnnn,n,n,±nnnnnn (Refer to command for description)

Remarks: Returns input linear equation configuration.

LOCK Front Panel Keyboard Lock Command

Input: **LOCK** <state>, <code>[term]

Format: n,nnn

<state> 0 = Unlocked, 1 = Locked

<code> Specifies lock-out code. Valid entries are 000–999.

Remarks: Locks out all front panel entries except pressing the **Alarm** key to silence alarms. Refer to Paragraph 4.17. Use the CODE command to set the lock code.

Example: **LOCK 1,123[term]** – Enables keypad lock and sets the code to 123.

LOCK? Front Panel Keyboard Lock Query

Input: **LOCK?** [term]

Returned: <state>, <code>[term]

Format: n,nnn (Refer to command for description)

MDAT? Minimum/Maximum Data Query

Input: **MDAT?** <input> [term]

Format: a
 <input> Specifies which input to query: A or B.

Returned: <min value>, <max value>[term]

Format: ±nnnnnn,±nnnnnn

Remarks: Returns the minimum and maximum input data. Also see the RDGST? command.

MNMX Minimum and Maximum Input Function Parameter Command

Input: **MNMX** <input>, <source> [term]

Format: a,n
 <input> Specifies input to configure: A or B.
 <source> Specifies input data to process through max/min. Valid entries: 1 = kelvin,
 2 = Celsius, 3 = sensor units, 4 = linear data.

Example: **MNMX B,3[term]** – Input B min/max function is on and processes data from the input sensor units reading.

MNMX? Minimum and Maximum Input Function Parameter Query

Input: **MNMX?** <input> [term]

Format: a
 <input> Specifies which input to query: A or B.

Returned: <source>[term]

Format: n (Refer to command for description)

MNMXRST Minimum and Maximum Function Reset Command

Input: **MNMXRST** [term]

Remarks: Resets the minimum and maximum data for all inputs.

MODE Remote Interface Mode Command

Input: **MODE** <mode> [term]

Format: n
 <mode> 0 = local, 1 = remote, 2 = remote with local lockout.

Example: **MODE 2[term]** – Places the Model 332 into remote mode with local lockout.

MODE? Remote Interface Mode Query

Input: **MODE?** [**term**]

Returned: <mode>[term]

Format: n (Refer to command for description)

MOUT Control Loop Manual Heater Power (MHP) Output Command

Input: **MOUT** <loop>, <value>[term]

Format: n,±nnnnnn[term]

<loop> Specifies loop to configure: 1 or 2.

<value> Specifies value for manual output.

Example: **MOUT 1,22.45**[term] – Control Loop 1 manual heater power output is 22.45%.

MOUT? Control Loop Manual Heater Power (MHP) Output Query

Input: **MOUT?** <loop>[term]

Format: n

<loop> Specifies which loop to query: 1 or 2.

Returned: <value>

Format: ±nnnnnn[term] (Refer to command for description)

PID Control Loop PID Values Command

Input: **PID** <loop>, <P value>, <I value>, <D value>[term]

Format: n,±nnnnnn,±nnnnnn,±nnnnnn

<loop> Specifies loop to configure: 1 or 2.

<P value> The value for control loop Proportional (gain): 0.1 to 1000.

<I value> The value for control loop Integral (reset): 0.1 to 1000.

<D value> The value for control loop Derivative (rate): 0 to 200.

Remarks: Setting resolution is less than 6 digits indicated.

Example: **PID 1,10,50**[term] – Control Loop 1 P is 10 and I is 50.

PID? Control Loop PID Values Query

Input: **PID?** <loop>[term]

Format: n

<loop> Specifies which loop to query: 1 or 2.

Returned: <P value>, <I value>, <D value>[term]

Format: ±nnnnnn,±nnnnnn,±nnnnnn (Refer to command for description)

RAMP Control Setpoint Ramp Parameter Command

Input: **RAMP** <loop>, <off/on>, <rate value>[term]

Format: n,n,±nnnnnn

<loop> Specifies which loop to configure: 1 or 2.

<off/on> Specifies whether ramping is 0 = Off or 1 = On.

<rate value> Specifies setpoint ramp rate in Kelvin per minute from 0.1 to 100. The rate is always positive, but will respond to ramps up or down.

Example: **RAMP 1,1,10.5[term]** – When Control Loop 1 setpoint is changed, ramp the current setpoint to the target setpoint at 10.5 K/minute.

RAMP? Control Setpoint Ramp Parameter Query

Input: **RAMP?** <loop>

Format: n

<loop> Specifies which loop to query: 1 or 2.

Returned: <off/on>, <rate value>[term]

Format: n,±nnnnn (Refer to command for description)

RAMPST? Control Setpoint Ramp Status Query

Input: **RAMPST?** <loop>[term]

Format: n

<loop> Specifies which loop to query: 1 or 2.

Returned: <ramp status>[term]

Format: n

<ramp status> 0 = Not ramping, 1 = Setpoint is ramping.

RANGE Heater Range Command

Input: **RANGE** <range>[term]

Format: n 0 = Off, 1 = Low (0.5 W), 2 = Medium (5 W), 3 = High (50 W)

RANGE? Heater Range Query

Input: **RANGE?** [term]

Returned: <range>[term]

Format: n (Refer to command for description)

RDGST? Input Reading Status Query**Input:** RDGST? <input>[term]**Format:** a
<input> Specifies which input to query: A or B.**Returned:** <status bit weighting>[term]**Format:** nnn**Remarks:** The integer returned represents the sum of the bit weighting of the input status flag bits. A "000" response indicates a valid reading is present.

<u>Bit</u>	<u>Bit Weighting</u>	<u>Status Indicator</u>
0	1	invalid reading
4	16	temp underrange
5	32	temp overrange
6	64	sensor units zero
7	128	sensor units overrange

RELAY Relay Control Parameter Command**Input:** RELAY <relay number>, <mode>, <input alarm>, <alarm type>[term]**Format:** n,n,a,n
<relay number> Specifies which relay to configure: 1 or 2.
<mode> Specifies relay mode. 0 = Off, 1 = On, 2 = Alarms.
<input alarm> Specifies which input alarm activates the relay when the relay is in alarm mode: A or B.
<alarm type> Specifies the input alarm type that activates the relay when the relay is in alarm mode. 0 = Low alarm, 1 = High Alarm, 2 = Both Alarms.**Example:** RELAY 1,2,B,0[term] – Relay 1 activates when Input B low alarm activates.**RELAY?** Relay Control Parameter Query**Input:** RELAY? <relay number>[term]**Format:** n
<relay number> Specifies which relay to query: 1 or 2.**Returned:** n,a,n (Refer to command for description)**RELAYST?** Relay Status Query**Input:** RELAYST? <high/low>**Format:** n
<high/low> Specifies relay type to query: 1 = Low Alarm or 1 = High Alarm.**Returned:** <status>[term]**Format:** n 0 = Off, 1 = On.**REV?** Input Firmware Revision Query**Input:** REV? [term]**Returned:** <revision>[term]**Format:** n.n**Remarks:** Returns the version number of the input firmware installed in the instrument.

SCAL Generate SoftCal Curve Command

Input: **SCAL** <std>, <dest>, <SN>, <T1 value>, <U1 value>, <T2 value>, <U2 value>, <T3 value>, <U3 value>[term]

Format: n,nn,aaaaaaaaa,±nnnnnn,±nnnnnn,±nnnnnn,±nnnnnn,±nnnnnn,±nnnnnn
 <std> Specifies the standard curve to generate a SoftCal from. Valid entries: 1, 6, 7.
 <dest> Specifies the user curve to store the SoftCal curve. Valid entries: 21–41.
 <SN> Specifies the curve serial number. Limited to 10 characters.
 <T1 value> Specifies first temperature point.
 <U1 value> Specifies first sensor units point.
 <T2 value> Specifies second temperature point.
 <U2 value> Specifies second sensor units point.
 <T3 value> Specifies third temperature point.
 <U3 value> Specifies third sensor units point.

Remarks: Generates a SoftCal curve. Refer to Paragraph 5.3.

Example: **SCAL 1,21,1234567890,4.2,1.6260,77.32,1.0205,300.0,0.5189[term]** – Generates a three-point SoftCal curve from standard curve 1 and saves it in user curve 21.

SETP Control Setpoint Command

Input: **SETP** <loop>, <value>[term]

Format: n,±nnnnnn
 <loop> Specifies which loop to configure.
 <value> The value for the setpoint (in whatever units the setpoint is using).

Example: **SETP 1,122.5[term]** – Control Loop 1 setpoint is now 122.5 (based on its units).

SETP? Control Setpoint Query

Input: **SETP?** <loop>[term]

Format: n
 <loop> Specifies which loop to query: 1 or 2.

Returned: <value>[term]

Format: ±nnnnnn

SRDG? Sensor Units Input Reading Query

Input: **SRDG?** <input>[term]

Format: a
 <input> Specifies which input to query: A or B.

Returned: <sensor units value>[term]

Format: ±nnnnnn

Remarks: Also see the RDGST? command.

TEMP? Thermocouple Junction Temperature Query

Input: **TEMP?**

Returned: <junction temperature>[term]

Format: ±nnnnnnnn

Remarks: Temperature is in kelvin.

TUNEST? Control Tuning Status Query

Input: **TUNEST?**

Returned: <tuning status>[term]

Format: n 0 = no active tuning, 1 = active tuning.

ZONE Control Loop Zone Table Parameter Command

Input: **ZONE <loop>, <zone>, <top value>, <P value>, <I value>, <D value>, <mout value>, <range>[term]**

Format: n,nn,±nnnnnnn,±nnnnnnn,±nnnnnnn,±nnnnnnn,±nnnnnnn,n[term]

<loop> Specifies which loop to configure: 1 or 2.

<zone> Specifies which zone in the table to configure. Valid entries are: 1– 10.

<top value> Specifies the top temperature of this zone.

<P value> Specifies the P for this zone: 0.1 to 1000.

<I value> Specifies the I for this zone: 0.1 to 1000.

<D value> Specifies the D for this zone: 0 to 200%.

<mout value> Specifies the manual output for this zone: 0 to 100%.

<range> Specifies the heater range for this zone if <loop> = 1. Valid entries: 0– 3.

Remarks: Configures the control loop zone parameters. Refer to Paragraph 2.9.

Example: **ZONE 1,1,25.0,10,20,0,0,2[term]** – Control Loop 1 zone 1 is valid to 25.0 K with P = 10, I = 20, D = 0, and a heater range of 2.

ZONE? Control Loop Zone Table Parameter Query

Input: **ZONE? <loop>, <zone>[term]**

Format: n,nn

<loop> Specifies which loop to query: 1 or 2.

<zone> Specifies which zone in the table to query. Valid entries: 1– 10.

Returned: <top value>, <P value>, <I value>, <D value>, <mout value>, <range>[term]

Format: ±nnnnnnn,±nnnnnnn,±nnnnnnn,±nnnnnnn,±nnnnnnn,n (Refer to command for description)
