meatest

EN, rev. 01 FW ver. 1.014

M151 Current Calibrator

remote control manual

Content

1. Connection Setup

The calibrator can be integrated into automated calibration systems (ATS) and generally controlled from remote computer via RS232, or GPIB (IEEE488) interface. When controlled remotely, maximum ratings of calibrator's output signals as well as all other specifications are the same as in manual mode.

Only one interface can be used for communication at any given time. To establish connection between the calibrator and computer, select prefered interface in MENU->Interface->Active bus (RS232 is active by default) and follow instructions from relevant subchapter below.

1.1. RS232 Connection

Connect the calibrator to your remote controller (PC) using either:

- 9-pin D-SUB, 1:1 male/female RS232 cable

And set the remote controller as follows:

- COM port — see available COM ports in Windows Device Manager
USB appears as serial converter "USB Serial Port (COMxx)"
- Baudrate — 9600 (can be changed to 1200 – 115200 in calibrator Menu->Interface)
- Data bits — 8
- Stop bits — 1
- Parity — None
- Handshake (Xon/Xoff) — Off

1.2. GPIB Connection

Set GPIB address of controlled device in your remote controller (PC) to match value in calibrator's MENU->Interface->GPIB Address (2 by default).

Connect the calibrator to your remote controller (PC) using IEEE488 cable. When using converters, make sure you use National Instruments converters as other types (e.g. Agilent) might not work properly. Once connected, the calibrator automatically executes IEEE488 bus commands SH1, AH1, T5, L3, RL1, and DC1 (in that order). Calibrator also recognizes following general commands:

- DCL Device Clear – resets device to its basic state
- SDC Selected Device Clear – resets device to its basic state
- LLO Local Lock Out – puts device into Remote mode and locks keyboard
- GTL Go To Local – returns device from Remote mode and unlocks keyboard

2. Executing Commands

Most commands can be executed in Remote mode only. In Remote mode, device blocks manual entry interfaces (keyboard, touchscreen) to prevent unintended manipulation during remote control session. GPIB interface switches device into and out of Remote mode automatically. Other interfaces (RS232, USB and Ethernet) need to use following commands:

- SYSTem:REMote Puts device into Remote mode and locks keyboard except Local button
- SYSTem:RWLock Puts device into Remote mode and locks entire keyboard
- SYSTem:LOCal Returns device from Remote mode and unlocks keyboard

2.1. Command Syntax

All commands listed in this chapter are made of KEYWORDS and PARAMETERS.

One or more KEYWORDS divided by colons (:) make up name of a command. When typing a command, keywords in square brackets ([ ]) can be omitted as well as lowercase letters and leading colons. For example ":OUTPut:STATe ON", ":OUTPut ON" and "OUTP 1" are considered the same.

Different PARAMETER types describe what kind of data is expected. For more details on parameter types, see table below. In command reference parameters are indicated by sharp brackets (< >) and separated by commas (,). Square brackets ([ ]) indicate non-mandatory parameters, vertical lines (|) indicate parameter alternatives.

Multiple commands can be executed at once if separated by semicolon (;). In any case, command (or set of commands) has to end with line feed (LF) terminator in order to be executed. With exception of GPIB interface, the device will also accept carriage return (CR) and combination of both (CRLF) as terminators. Inversely, device will always terminate its reply with CRLF or just LF in case of GPIB. The communication may look like this:

1. Controller> „RES 100;OUTP ON"+LF (ASCII code 10)
2. Device> „1.000000E+02 OHM"+LF
3. Device> „1"+LF

Used parameter types

Type	Description
<DNPD>	Decimal Numeric Program Data, this format is used to express decimal number with or without the exponent.
<CPD>	Character Program Data. Usually, it represents a group of alternative character parameters. E.g. {SERial|GPIB|USB|LAN}.
<SPD>	String Program Data (quoted string). This type of parameter is similar to CPD, but allows transmission of more ISO characters.
<BOOL>	Boolean Program Data. This type of parameter has only two states 0 and 1. Parameter can take form of integer value (0 or 1), or character alias (ON or OFF). Device always returns integer value (0 or 1).
<UNIT>	Unit parameter works in conjunction with DNPD parameter and specifies unit of DNPD (numeric) value. Unit must be selected from predefined ones. If UNIT part is omitted, default one is used. Query always returns actual unit.
<N>	Numeric suffix – unique identifier with NR1 format (an integer without a decimal point).
?	A flag indicating a request for the value of the parameter specified by the command. No other parameter than the question mark can be used.
(?)	A flag indicating a request for the parameter specified by the command. This command permits a value to be set as well as requested.

3. Status Reporting

3.1. Status Data Structure-Register Model

Device meets standard protocol according to the standard IEEE488.2. The protocol can be used for checking of error and status behavior of the device. It enables single-wire transmitting of SRQ command. The conditions on which SRQ signal (local control request) is sent can be set with commands *STB?, *SRE?, *SRE, *ESR?, *ESE?, *ESE a *CLS.

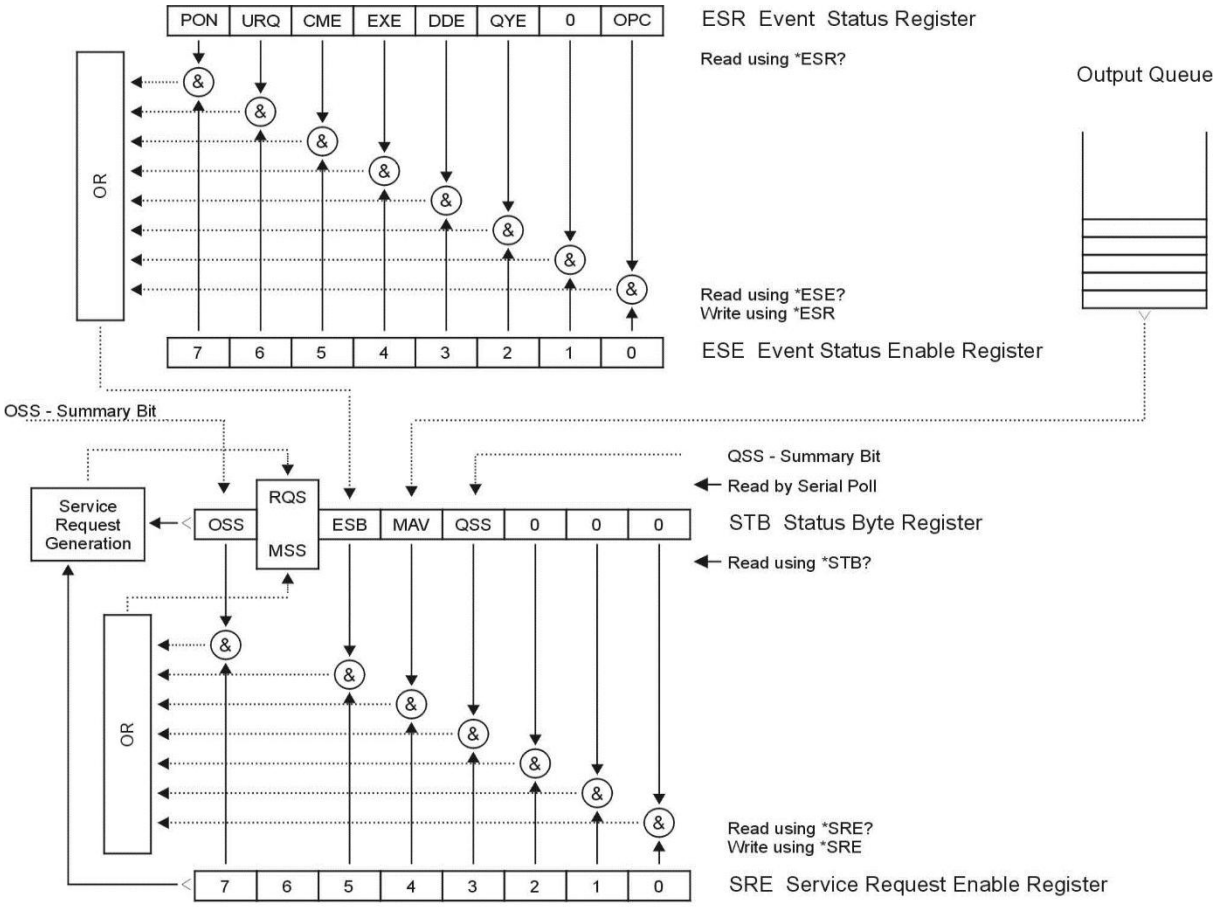

Figure 1 Status register overview

Status data structure contains following registers:

- STB – Status Byte Register
- SRE – Service Request Enable Register
- ESR – Event Status Register
- ESE – Event Status Enable Register
- Output Queue

3.2. Standard Status Data Structures

STB Status Byte Register

STB is main register where information from other status registers and from output queue is collected. Value of STB register is reset after switching on the calibrator or after sending command *CLS. This command reset the STB register except bit MAV, which remains set if the output queue is not empty. STB register value can be read via serial message or through general query *STB?.

Bit configuration of Status Byte Register:

OSS Operation Summary Status, bit 7. SCPI-defined. The OSS bit is set to 1 when the data in the OSR (Operation Status Register) contains one or more enabled bits which are true.

RQS Request Service, bit 6. The bit is read as a part of status byte only when serial message is sent.

MSS Master Summary Status, bit 6. The MSS bit is set to 1 whenever bits ESB or MAV are 1 and enabled (1) in the SRE. This bit can be read using the *STB? command. His value is derived from STB and SRE status.

ESB Event Summary Bit, bit 5. His value is derived from STB and SRE status. The ESB bit is set to 1 when one or more enabled ESR bits are set to 1.

MAV Message Available, bit 4. The MAV bit is set to 1 whenever data is available in the IEEE488 Output Queue (the response on query is ready).

QSS Questionable Summary Status, bit 3. SCPI-defined. The QSS bit is set to 1 when the data in the QSR (Questionable Status Register) contains one or more enabled bits which are true.

SRE Service Request Enable Register

The Service Request Enable Register suppresses or allows the STB bits. “0” value of a SRE bit means, that the bit does not influence value of MSS bit. Value of any unmasks STB bit results in setting of the MSS bit to the level “1”. SRE bit 6 is not influenced and its value is “0”. The SRE register value can be set via the command *SRE followed by mask register value (0 – 191). The register can be read with the command *SRE?. The register is automatically resets after switching the calibrator on. The register is not reset by the command *CLS.

ESR Event Status Register

Every bit of the EventStatusRegister corresponds to one event. Bit is set when the event is changed and it remains set also when the event passed. The ESR is cleared when the power is turned on (except bit PON which is set), and every time it is read via command *ESR? Or cleared with *CLS.
Bit configuration of Event Status Register:

PON Power On, bit 7. This event bit indicates that an off-to-on transition has occurred in the device's power supply.

URQ User Request, bit 6. Bit is not used and it is always "0".

CME Command Error, bit 5. This event bit indicates that an incorrectly formed command or query has been detected by the instrument.

EXE Execution Error, bit 4. This event bit indicates that the received command cannot be executed, owing to the device state or the command parameter being out of limits.

DDE Device Dependent Error, bit 3. This event bit indicates that an error has occurred which is neither a Command Error, a Query Error, nor an Execution Error. A Device-specific Error is any executed device operation that did not properly complete due to some condition, such as overload.

QYE Query Error, bit 2. The bit is set if the calibrator is addressed as talker and output queue is empty or if control unit did not pick up response before sending next query.

OPC Operation Complete, bit 0. This event bit is generated in response to the *OPC command. It indicates that the device has completed all selected pending operations.

ESE Event Status Enable Register

The Event Status Enable Register allows one or more events in the Event Status Register to be reflected in the ESB summary-message bit. This register is defined for 8 bits, each corresponding to the bits in the Event Status Register. The Event Status Enable Register is read with the common query *ESE?. Data is returned as a binary-weighted value. The Event Status Enable Register is written to by the common command, *ESE. Sending the *ESE common command followed by a zero clears the ESE. The Event Status Enable Register is cleared upon power-on.
It suppresses or allows bits in ESR register. Value „0" of a bit of ESE register suppresses influence of appropriate bit of ESR register on value of sum bit of ESB status register. Setting of any unmask bit of ESR register results in setting of ESB status register. ESE register value can be modified by command *ESE followed by value of mask register (integer in range 0 – 255). Reading of the register can be performed with command *ESE?. The register is automatically reset after switching on. The register is not reset with *CLS command.

Operation Status Register

Not used in the calibrator.

Questionable Status Register

Not used in the calibrator.

Output Queue

The Output Queue stores response messages until they are read from control unit. If there is at minimum one sign in the output queue, MAV register (message available) is set. The Output Queue is cleared upon power-on and after reading all signs from output queue.

Error Queue

The Error Queue stores error messages. They are placed in a “first in, first out” queue.

The queue is read destructively using the query command “SYSTem:ERRor?” to obtain a code number and error message. The query “SYSTem:ERRor?” can be used to read errors in the queue until it is empty, when the message “0, No Error” will be returned.

3.3. Error Messages

SCPI Remote control errors

Code	Name	Description
-430	Deadlocked.	Remote interface error. A command was received which generates too much data to fit in the output buffer and the output buffer is full. Command execution continues but all data are lost.
-420	Unterminated.	Remote interface error. The calibrator was addressed to talk but a command has not been received which sends data to the output buffer.
-410	Interrupted.	Remote interface error. A command was received which sends data to the output buffer, but the output buffer contained data from a previous command. The output buffer is cleared when power has been off, or after reset command has been executed.
-363	Input buffer overrun	Remote interface error.
-220	Invalid parameter	Remote interface error. An invalid character string was received. Check to see if you have enclosed the character string in single or double quotes and that the string contains valid ASCII character.
-140	Character data	Remote interface error. Received command doesn’t contain valid character parameter.
-120	Numeric data	Remote interface error. Received command doesn’t contain valid numeric parameter.
-110	Command header	Remote interface error. Received command isn’t valid.
501	Eeprom write.	Eeprom write failed.
502	Eeprom read.	Eeprom read failed.
503	Eeprom error.	Eeprom data lost. Check calibration data.
701,704	Output overload	Output signal over specified limits. Decrease signal level or decrease the load.
703	High temperature	Calibrator power stage overheated. Disconnect external load. Ambient temperature is too high or forced ventilation holes are blocked.
705	Input overload	Input signal over specified limits. Decrease signal level.
706	Current output overload	Current output terminals overloaded. Decrease signal level or decrease the load.
722	Unexpected crossing.	Error in internal communication.
721	Unknown function.	Error in internal communication.
730 - 731	Calibrator not ready	Error in internal communication.
732	Internal cpu RESET	Calibrator will be restarted.
743	Interface receive	Error in internal communication.
745	Internal CPU timeout	Error in internal communication.
746 - 748	Slave error	Error in internal communication.

4. M151 SCPI Command Reference

OUTPut subsystem

This subsystem enables to activate or deactivate the calibrator output.

Keyword	**Parameters**
OUTPut	
[:STATe](?)	<CPD> { ON | OFF }
:LOWCurrent (?)	<CPD> { FLOat | GROund }
:CURCoil(?)	<CPD> { OFF | X25 | USER }
:USER(?)	<DNPD>
:SYNChronization(?)	<CPD> { INT | LINE |EXT }
:LOCKed?	<CPD> { 0 | 1 }

OUTP[:STAT](?) <CPD> { ON | OFF }

This command activates or deactivates output terminals.

- ON - activates the output
- OFF - deactivates the output

If query is sent, calibrator returns ON if the output is active or OFF if it is inactive

Example: OUTP ON - activates the output
OUTP? - the calibrator returns ON or OFF

OUTP:LOWC(?) <CPD> { FLO | GRO }

This command connects or disconnects the Lo terminal of current output to/from GND terminal.

- FLOat ungrounds current output
- GROund grounds current output

If query is sent, calibrator returns GRO when the output is grounded or FLO when floating.

Example:

OUTP:LOWC GRO
OUTP:LOWC? Calibrator returns FLO or GRO

Warning!

Lo output terminal and COM input terminal are grounded in real TransConductance mode regardless of this item's setting.

OUTP:CURC(?) <CPD> { OFF | X25 | USER }

This command activates or deactivates the use of 25-turns or user current coil.

- OFF - deactivates the current coil
- X25 - activates the 25 turns current coil
- USER - activates the user current coil

If query is sent, calibrator returns USER if the user coil is activated X25 if the 25-turn coil is activated or OFF if the current coil is deactivated.

Example: OUTP:CURC X25 activates 25-turn current coil
OUTP:CURC? Calibrator returns OFF, USER or X25

OUTP:CURC:USER(?) <DNPD>

This command sets the number of the turns.
Příkaz nastavuje hodnotu uživatelské závitové cívky.
<DNPD>
Parameter represents the number of the turns.
Example: OUTP:CURC:USER 50 sets the 50-turn current coil

OUTP:SYNC(?) <CPD> { INT | LINE | EXT }

This command sets mode of output frequency synchronization.

- INT – internal synchronization
- LINE – synchronization to power supply frequency
- EXT – synchronization to external signal

If query is sent, calibrator returns INT, LINE or EXT according to selected synchronization mode.
Example: OUTP:SYNC INT selects frequency synchronization to internal reference
OUTP:SYNC? Calibrator returns INT, LINE or EXT

OUTP:SYNC:LOCK? <CPD> { 0 | 1 }

This query returns the actual state of frequency synchronization.

- 0 – output frequnecy is not locked (it is not possible to activate the output terminals)
- 1 – output frequency is locked to reference signal

Example: OUTP:SYNC:LOCK? calibrator returns 1 or 0

MEASure a CONFigure subsystem

This subsystem allows controling the internal multimeter.

Keyword	**Parameters**
MEASure?	
CONFigure(?)	<CPD> { VOLTage | CURRent }

MEAS?

This query returns value measured by the multimeter.
Calibrator returns the set values of amplitude and frequency using standard exponential format. Example: 7.456 V / 50.1 Hz is returned as 7.456000e+000,5.010000e+001.
Example: MEAS? Calibrator returns measured value

CONF(?) <CPD> { VOLT | CURR}

This command sets the function of internal multimeter.
If query is sent, calibrator returns VOLT or CURR according to selected multimeter function.
Example: CONF VOLT sets the multimeter to function voltage

SOURce subsystem

This subsystem allows controlling the individual functions of the calibrator.

Keyword	**Parameters**
[SOURce]	
:MODE? { CAC | CDC | AMAC | AMDC | TAMP }	
:CAC	
:CURRent(?)	<DNPD>
:FREQuency(?)	<DNPD >
:CDC	
:CURRent(?)	<DNPD>
:AMAC	
:CURRent?	<DNPD>
:FREQuency(?)	<DNPD >
:AMDC	
:CURRent?	<DNPD>
:TAMP	
:RANGe?	<CPD> {0.3 | 1 | 2 | 5 | 10 | 30 | 60 | 120}
:GNU(?)	<DNPD >
:GNI(?)	<DNPD >
:STEP(?)	<DNPD >

[SOUR]:MODE?

This command returns selected mode of the calibrator {CAC | CDC | AMAC | AMDC | TAMP}.

- CAC – Current AC mode
- CDC – Current DC mode
- AMAC – Current AC amplifier mode
- AMDC – Current DC amplifier mode
- TAMP – Real TransConductance amplifier mode

Example:

MODE?

[SOUR]:CAC:CURR(?) <DNPD>

This command sets the output current value in Current AC mode.

<DNPD>

Parameter represents the value of the output current expressed in A.

If query is sent, calibrator returns the value of output current using standard exponential format. Example: 11.012 A is returned as 1.101200e+001.

Example:

Current 23.05A:

CAC:CURR 23.05

Note:

Command switches the calibrator in Current AC mode.

[SOUR]:CAC:FREQ(?) <DNPD>

This command sets the frequency in Current AC mode.
<DNPD>
Parameter represents the value of frequency expressed in Hz.
If query is sent, calibrator returns the value of frequency using standard exponential format.
Example: 50Hz is returned as 5.000000e+001.
Example:
Frequency 60Hz:

CAC:FREQ 60

Notes:
Command switches the calibrator in Current AC mode.

[SOUR]:CDC:CURR(?) <DNPD>

This command sets the output current value in Current DC mode.
<DNPD>
Parameter represents the value of the output current expressed in A.
If query is sent, calibrator returns the value of output voltage using standard exponential format. Example: 11.012 A is returned as 1.101200e+001.
Example:
Current 23.05A:

CDC:CURR 23.05

Note:
Command switches the calibrator in Current DC mode.

[SOUR]:AMAC:CURR(?) <DNPD>

This command sets the output current value in AMAC mode.
<DNPD>
Parameter represents the value of the output current expressed in A.
If query is sent, calibrator returns the value of output current using standard exponential format. Example: 11.012 A is returned as 1.101200e+001.
Example:
Current 23.05A:

AMAC:CURR 23.05

Note:
Command switches the calibrator in AMAC mode.

[SOUR]:AMAC:FREQ(?) <DNPD>

This command sets the frequency in AMAC mode.
<DNPD>
Parameter represents the value of frequency expressed in Hz.
If query is sent, calibrator returns the value of frequency using standard exponential format.
Example: 50Hz is returned as 5.000000e+001.
Example:
Frequency 60Hz:

AMAC:FREQ 60

Notes:
Command switches the calibrator in AMAC mode.

[SOUR]:AMDC:CURR(?) <DNPD>

This command sets the output current value in AMDC mode.
<DNPD>
Parameter represents the value of the output current expressed in A.
If query is sent, calibrator returns the value of output current using standard exponential format. Example: 11.012 A is returned as 1.101200e+001.
Example:
Current 23.05A:

AMDC:CURR 23.05

Note:
Command switches the calibrator in AMDC mode.

[SOUR]:TAMP:RANG(?) <CPD> {0.3 | 1 | 2 | 5 | 10 | 30 | 60 | 120}

This command sets the range of real TransConductance amplifier.
<CPD>
Parameter represents the range. Ranges are from 0.3A up to 120A.
If query is sent, calibrator returns the selected range. Example: 10A range is returned as 10.
Example:
Range 10A:

TAMP:RANG 10

Note:
Command switches the calibrator in Real TransConductance amplifier mode.

[SOUR]:GNU(?) <DNPD>

This command sets the gain value in transconductance amplifier mode (AMAC and AMDC).
<DNPD>
Parameter represents the value of the gain expressed in A/V.
If query is sent, calibrator returns the value of output current using standard exponential format. Example: 5.123 A/V is returned as 5.123000e+000.
Example:
Gain 8.05 A/V:

GNU 8.05

[SOUR]:GNI(?) <DNPD>

This command sets the gain value in current amplifier mode (AMAC and AMDC).
<DNPD>
Parameter represents the value of the gain expressed in A/A.
If query is sent, calibrator returns the value of output current using standard exponential format. Example: 612 A/A is returned as 6.120000e+002.
Example:
Gain 600 A/A:

GNI 600

[SOUR]:STEP(?) <DNPD>

This command sets the output current step value in AMAC and AMDC mode.
<DNPD>
Parameter represents the value of the output current step expressed in A.
If query is sent, calibrator returns the value of output current using standard exponential format. Example: 0.5 A is returned as 5.000000e-001.
Example:
Step 0.5 A:

```
STEP 0.5
```

SYSTem subsystem

The subsystem enables to control various functions from the MENU.

```
SYSTem
    :DATE(?)                    <DNPD>,<DNPD>,<DNPD>
    :TIME(?)                    <DNPD>,<DNPD>,<DNPD>
    :ERRor?
    :REMote
    :RWLock
    :LOCal
```

SYST:DATE(?) <DNPD>,<DNPD>,<DNPD>

This command sets system date of the calibrator.
<DNPD>,<DNPD>,<DNPD>
Represents date in format YYYY, MM, DD.
If query is sent, calibrator returns current value of system date in format YYYY,MM,DD.
where YYYY = year (2000..2099)
MM = month (01..12)
DD = day (01..31)

SYST:TIME(?) <DNPD>,<DNPD>,<DNPD>

This command sets time of the calibrator.
<DNPD>,<DNPD>,<DNPD>
Represents time in format HH,MM,SS.
If query is sent, calibrator returns current value of system time in format HH,MM,SS.
where HH = hour (00..23)
MM = minute (00..59)
SS = second (00..59)

SYST:ERR?

Query the multimeter's error queue. Detected errors are placed in the queue. This query returns the first error from the queue. The queue is type "first in, first out". The error is deleted after reading.
The response is in the form of 'String Program Data', and consists of two elements: a code number and error message.
If all errors are read calibrator returns message 0,"No Error".
In case of overflow the error queue the last error in the queue is –350,"Queue overflow". Also the earliest errors remain in the queue, and the most recent error is discarded.
Example:

```
SYST:ERR?
```

SYST:REM

This command places the calibrator in the remote mode for RS232 operation. All keys on the front panel, except the LOCAL key, are disabled.
Note:
It is not possible to send or receive data over RS232 when the calibrator is not in remote mode.

SYST:RWL

This command places the calibrator in the remote mode for RS232 operation. All keys on the front panel, including the LOCAL key, are disabled.

SYST:LOC

This command returns the calibrator in the local mode. This command is for RS232.

STATus subsystem

This subsystem is used to enable bits in the Operation and Questionable Event registers. The Operation and Questionable: Event, Enable and Condition registers can be interrogated to determine their state.

```
STATus
    :OPERational
        :EVENt?
        :ENABle(?)          <DNPD>
        :CONDition?
    :QUEStionable
        :EVENt?
        :ENABle(?)          <DNPD>
        :CONDition?
    :PRESet
```

STAT:OPER:EVEN?

This query returns the content of Operational Data Event register. It is a decimal value which corresponds to the binary-weighted sum of all bits set in the register. Register is cleared after this query.
Example:

```
STAT:OPER:EVEN?
```

STAT:OPER:ENAB? <DNPD>

This command enables bits in the Operational Data Enable register. Selected bits are summarized at bit 7 (OSS) of the IEEE488.2 Status Byte register.
<DNPD>
Parameter represents required value of Operational Data Enable register.
If query is sent, calibrator returns the value of the registr as a decimal number. Example: 64 is returned as 64.
Example:
Set bit 1 in the Operational Data Enable register (other bits are 0):

```
STAT:OPER:ENAB 2
```

STAT:OPER:COND?

This query returns the content of Operational Condition register. It is a decimal value which corresponds to the binary-weighted sum of all bits in the register. Register is not cleared after this query. The response to the query therefore represents an instantaneous 'Snapshot' of the register state, at the time that the query was accepted.
Example:

STAT:OPER:COND?

STAT:QUES:EVEN?

This query returns the content of Questionable Data Event register. It is a decimal value which corresponds to the binary-weighted sum of all bits set in the register. Register is cleared after this query.
Example:

STAT:QUES:EVEN?

STAT:QUES:ENAB? <DNPD>

This command enables bits in the Questionable Data Enable register. Selected bits are summarized at bit 3 (QSS) of the IEEE488.2 Status Byte register.
<DNPD>
Parameter represents required value of Questionable Data Enable register.
If query is sent, calibrator returns the value of the registr as a decimal number. Example: 64 is returned as 64.
Example:
Set bit 1 in the Questionable Data Enable register (other bits are 0):

STAT:QUES:ENAB 2

STAT:QUES:COND?

This query returns the content of Questionable Condition register. It is a decimal value which corresponds to the binary-weighted sum of all bits in the register. Register is not cleared after this query. The response to the query therefore represents an instantaneous 'Snapshot' of the register state, at the time that the query was accepted.
Example:

STAT:QUES:COND?

STAT:PRES

This command clears all bits in the Operation Data Enable register and in the Questionable Data Enable register.
Example:

STAT:PRES

IEEE488.2 Common Commands

I/D (instrument identification)

***IDN?**

This command returns the identification of the manufacturer, model, serial number and firmware revision.
The reply is formatted as follows:
MEATEST,M151,510001,1.22

Operation complete

***OPC**

This command sets the OPC bit in the ESR (Event Status Register) when all pending operations are complete.

Operation complete?

***OPC?**

This command returns “1” to the output queue after all pending operations are complete.

Wait-to-Continue command

***WAI**

Prevents the instrument from executing any further commands or queries until all previous remote commands have been executed.

Reset

***RST**

This command resets the calibrator to its initial status.

Test operation

***TST?**

This command launches an internal self-test. Return the self-test result (“0” for pass or “1” for fail).

Status byte reading

***STB?**

This query returns number in range 0 to 255 with information about content of register STB, which carries the MSS bit status.

Service Request Enable

***SRE <value>**

This command sets condition of the Service Request Enable register. Since bit 6 is not used, the maximum value is 191.

Service Request Enable reading

***SRE?**

This query returns the Service Request Enable Register number.

Event Status Register reading

***ESR?**

This query returns the contents of the Event Status Register and clears the register.

Event Status Enable setting

***ESE <value>**

This command programs the Event Status Enable register bits. Parameter “value” is number in range 0 – 255.

Event Status Enable reading

***ESE?**

This query returns the Event Status Enable register.

Clear status

***CLS**

This command clears the Event Status Register and the Status Byte Register except the MAV bit and output queue. Output line is not reset.

OUTPut subsystem

This subsystem enables to activate or deactivate the calibrator output.