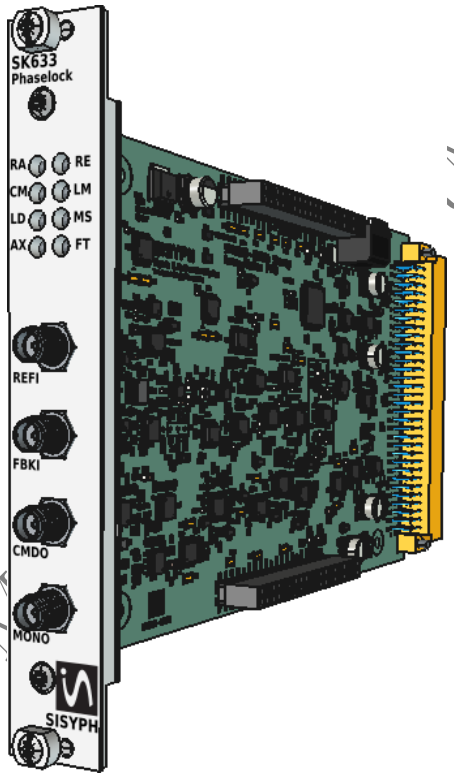
Programming Guide

SK633 *Sancy*

Low-Noise Phase-Frequency Detector

SK-Series Modular Instrument

PRELIMINARY INFORMATION



SISYPH
Signals & Systems for Physics

www.sisyph.com

General Information

Important Notice

Information in this document is subject to change without notice.
Copyright © SISYPH, 2026. All rights reserved.

Signals and Systems for Physics
206 Rue Alfred Kastler
17310 Saint-Georges-d'Oléron
FRANCE
Phone (+33) 781 547 391
www.sisyph.com

Scope

This document is a practical guide for operating the SK633 *Low-Noise Phase-Frequency Detector* via its serial interface. It provides clear, step-by-step instructions to help users :

- Connect the SK633 module to a host system and establish serial communication.
- Send commands and interpret responses to control the instrument.
- Configure module settings, including operating modes and parameters.
- Perform basic diagnostics, monitor status, and troubleshoot common issues.
- Integrate the module into laboratory setups or embedded systems with minimal setup time.

The guide focuses on hands-on usage, with examples and illustrations to make it easy to operate the SK633 safely and efficiently.

Document Identifier

This document is identified as SK633-SU03-R26A and was generated on November 8, 2026.

PRELIMINARY INFORMATION

Contents

General Information **2**
Important Notice . . . 2
Scope . . . 2

1 Introduction **4**
1.1 Power-on Configuration . . . 4
1.2 Buffers . . . 4
1.3 Command Syntax . . . 4
1.4 Examples . . . 5
1.5 Data Streaming . . . 5

2 Status Model **7**
2.1 Introduction . . . 7
2.2 Registers . . . 8

3 List of Commands **18**

4 Index of Commands **67**

PRELIMINARY INFORMATION

1 Introduction

Remote operation of the SK633 is through a simple command language documented in this chapter. Both set and query forms of most commands are supported, allowing the user complete control of the module from a remote computer.

1.1 Power-on Configuration

The settings for the serial interface are 9600 baud with no parity and no hardware flow control, and local echo disabled (`CONS 0`). Most of the instrument settings are stored in non-volatile memory and can be retrieved using the appropriate commands. At power-on the instrument returns to the state noted in the command descriptions. Reset values (see `*RST` command) of parameters are shown in **boldface**.

1.2 Buffers

The instrument stores incoming bytes from the host interface in a 128-byte input buffer. Characters accumulate in the input buffer until a command terminator (either `<CR>` or `<LF>`) is received, at which point the message is parsed and executed. Query responses from the instrument are sent when they are ready without any flow control or output buffering. The input buffer is automatically flushed upon detecting an overflow, and an error is recorded in the EVTS status register.

1.3 Command Syntax

The four-letter mnemonic (shown in CAPS) in each command sequence specifies the command. The rest of the sequence consists of parameters. The command parser accepts only uppercase mnemonics.
Commands may take either set or query form, depending on whether the `?` character follows the mnemonic. *Set-only* commands are listed without the `?`, *query-only* commands show the `?` after the mnemonic, and commands whose *query* takes an optional argument are marked with a `[?]`. Parameters shown in `{ }` and `[ ]` are not always required. Parameters in `{ }` are only required to set a value, and should be omitted for queries. Parameters in `[ ]` are optional in both set and query commands. Parameters listed without any surrounding characters are always required. Do not send `( )` or `{ }` or `[ ]` as part of the command. Multiple parameters are separated by commas. Multiple commands may be sent on one command line by separating them with semicolons `;` so long as the input buffer does not overflow. Commands are terminated by either `<CR>` or `<LF>` characters. Null commands and whitespace are ignored. Execution of the command does not begin until the command terminator is received.
Fixed-point parameters may be entered with fewer decimal places than the specified resolution, including no decimal places. The parser automatically scales the supplied value to the required representation. More decimal places than specified are not accepted and result in a parser error. For example, for a fixed-point parameter with two decimal places, `-9.99`, `-9.9`, and `-9` are valid, while `-9.999` is illegal.

The following table summarizes the notation used in the command descriptions:

PRELIMINARY INFORMATION

Symbol	Definition
b	boolean
i, *m*, *n*	unsigned integers
u, *v*	signed integers
x, *y*	fixed-point decimal values
(?)	required for queries – illegal for set commands
p	parameter always required
{*p*}	required parameter for set commands – illegal for queries
[*p*]	optional parameter for both set and query forms

Table 1: Command argument notation and usage

1.4 Examples

Each command is provided with a simple example illustrating its usage. In these examples, all data sent by the host computer to the instrument are set in `straight teletype font`, while responses received by the host computer from the instrument are set as *`slanted teletype font`*. The usage examples vary with respect to set/query, optional parameters, and token formats. These examples are not exhaustive, but are intended to provide a convenient starting point for user programming.

1.5 Data Streaming

The streaming facility was originally developed to support firmware development and diagnostic activities. It is now available through the user interface as a general-purpose monitoring feature.
The instrument can periodically stream selected values through the communications port. Only values associated with commands designated as streamable can be included in the stream.
Whether a parameter or register can be streamed is indicated in the descriptor of the corresponding command.
Streaming is controlled by the `STME` and `STMN` commands. `STME` enables or disables data streaming, while `STMN` sets the number of samples to be streamed.
A streamable value is added to the output by appending `+` to the query form of its command and removed from the output by appending `-`: `INSS ? +` adds the INSS register to the stream, while `INSS ? -` removes it.
For example, the temperature measurement may be added to the stream with: `TMPM ? +` and subsequently removed with `TMPM ? -`.
Selecting a value that is already enabled for streaming has no effect. Likewise, requesting the removal of a value that is not currently selected has no effect.
Adding or removing a streamed value changes the composition of the output and resets the sample counter. The first sample of the new stream is therefore numbered 1, independently of the previous counter value.
The set of selected streamed values is independent of the `STME` state. Disabling and subsequently re-enabling streaming therefore preserves the current selection. All streaming selections are cleared at power-on and by an instrument reset.
Streamed data are produced with an approximate period of 1 s and formatted as comma-separated values (CSV). The leftmost field is always the sample number. The remaining fields contain the currently selected streamed values.
The first output row contains the column headers corresponding to the fields that follow. The header row is emitted before the sample rows and is not itself a numbered sample. The order of the streamed columns is

PRELIMINARY INFORMATION

fixed by the firmware and does not depend on the order in which values are selected for streaming. It follows the position of the corresponding command or register in the firmware lookup tables.
For directly addressable parameters and measurements, the command mnemonic is used as the corresponding column header. The same principle applies to directly addressable registers, for which the register name is used as the column header.
For example, a stream containing the temperature measurement obtained from `TMPM` and the INSS status register may have the following form:

N	*TMPM*	*INSS*
1	*24.3*	*0b00000100*
2	*24.3*	*0b00000100*
3	*24.4*	*0b00000000*

The representation of each streamed value depends on its type. Values may be output as integers, fixed-point decimal values, or binary values (`0b` prefix). A physical measurement such as the value returned by `TMPM` may be represented as a fixed-point decimal value, whereas a status register such as INSS is represented in binary form so that the individual status bits remain directly visible.
Control commands require a different treatment when several controls share the same underlying control register. Such commands may address only one bit or one bit field of this register. When one of these controls is selected for streaming, the complete underlying control register is streamed rather than only the individual control bit or field. The control-register name is then used as the column header.
For example, if a streamable control command operates on a bit contained in the CTLA control register, the streamed values may have the following form:

N	*CTLA*
1	*0b00000100*
2	*0b00000100*
3	*0b00000000*

This gives access to the complete state of a shared control register, including bits that may be associated with other control commands. Binary representation is therefore particularly useful for shared control registers and status registers. For commands associated with a single bit of a control or flag register, the complete register is streamed. The command descriptor therefore specifies *(header, pos, mask)*, where *header* is the streamed register column, *pos* is the zero-indexed bit position, and $mask = 2^{pos}$ is the corresponding decimal bit mask. For example, `(CTLA, 4, 16)` identifies bit 4 of the streamed `CTLA` register.
Streaming a status register is non-destructive. The periodic read performed by the streaming mechanism does not clear or otherwise alter the register contents. This differs from explicitly issuing commands that act on the same register.
For example, if a sticky bit in the OVLS status register is set, successive streamed values continue to report that bit as set. If `OVLS?` or `*CLS` is then issued, the register is affected according to the normal command semantics, and the next streamed row reflects the resulting register contents. This behavior can be useful for diagnostic purposes, since it allows the evolution of a status condition to be observed continuously while retaining explicit control over when the associated latched information is cleared.
All commands remain available while streaming is active. A normal query response is transmitted when the command is executed and may therefore appear between two consecutive streamed rows. Such command responses can disrupt the CSV structure and should be taken into account by software recording or parsing the streamed data.

PRELIMINARY INFORMATION

2 Status Model

2.1 Introduction

A complete block diagram of the status register array is provided in the documentation accompanying the instrument. It is also accessible online on the corresponding product page. There are four categories of registers in this model :

Last Event Registers These four read-only registers (LINS, LCMD, LURQ and LEXE) store the last event that they monitor. A query command (i) returns the last registered event since the previous query and (ii) clears the register's content.

Condition Registers These read-only registers correspond to the real-time condition of some underlying physical properties being monitored. Queries return the latest value of the property, and have no other effect. Condition registers' flags correspond one-to-one with the bits of the related Status Register (see below). Condition register names end with C (*e.g.* INS**C**).

Status Registers These read-only registers record the occurrence of defined events. If the event occurs, the corresponding status bit is set to 1. Upon querying a status register, any set bits within it are cleared. These bits are sometimes known as sticky bits since, once set, they can only be cleared by reading their value. Status register names end with S (*e.g.* INS**S**).

Enable Registers These read/write registers define a bitwise mask for their corresponding status register. If any bit position is set in a status register while the same bit position is also set in the enable register, then the corresponding summary bit is set in either the Event Summary (EVTS) or Master Summary register (MSTS). Enable registers' flags correspond one-to-one with the bits of the related Status Register (see above). Enable register names end with E (*e.g.* INS**E**).

PRELIMINARY INFORMATION

2.2 Registers

This section provides descriptions of status registers.

Last Event Registers	LEXE	Last Execution Error
	LINS	Last Instrument Error
	LCMD	Last Command Error
	LURQ	Last User Request
Status Registers	INSS	Instrument Status Register
	OVLS	Overload Status Register
	EVTS	Event Status Summary Register
	COMS	Communications Status Register
	MSTS	Master Summary Status Register
Condition Registers	INSC	Instrument Condition Register
	OVLC	Overload Condition Register
Enable Registers	INSE	Instrument Enable Register
	OVLE	Overload Enable Register
	EVTE	Event Summary Enable Register
	COME	Communications Enable Register
	MSTE	Master Summary Enable Register

Table 2: Status-model registers by type

PRELIMINARY INFORMATION

2.2.1 Instrument Status Register (INSS)

The Instrument Status register consists of eight event flags. These flags are set by the corresponding event, and cleared only by reading or with the `*CLS` command ("sticky bits"). Querying the single bit i with the command `INSS? n` where the bit-mask $n = 2^i$ will only clear the bit i. For instance, issuing the command `INSS? 1` will clear the bit 0 only.

Weight $n = 2^i$	Bit i	Flag	Description
128	7	RFU	Undefined (read 0)
64	6	CRG	Command voltage out of range (bit set)
32	5	ERG	Error voltage out of range (bit set)
16	4	FRG	Frequency difference out of range (bit set)
8	3	IPW	Local voltage regulation fault detected (bit set)
4	2	LKD	Lock-detect asserted (bit set)
2	1	IKS	Internal 10-MHz clock source used (bit set)
1	0	PUV	PWGD line not asserted; under-voltage detected externally (bit set)

Table 3: Bit definitions for the INSS status register and the corresponding INSC condition and INSE enable registers.

2.2.2 Instrument Enable Register (INSE)

Each bit in the INSE register corresponds one-to-one with a bit in the INSS register, and acts as a bitwise logical-AND of the flags to generate the INS flag in the Master Summary Status (MSTS) register. The INSE register is set and queried with the `INSE` command and cleared at power-on.

2.2.3 Instrument Condition Register (INSC)

Each bit in the INSC register corresponds one-to-one with a bit in the INSS register. The bits in the INSC register reflect the real-time values of their corresponding signals. Reading the entire register, or individual bits within it, does not affect the value of INSC. This register is queried with the `INSC` command. All bits read 0 at power-on, until the corresponding signals are first evaluated.

PRELIMINARY INFORMATION

2.2.4 Overload Status Register (OVLS)

The Overload Status register consists of eight event flags, which record overrange and overload conditions, that is, signals or physical quantities found outside their allowed limits. These flags are set by the corresponding event, and cleared only by reading or with the `*CLS` command ("sticky bits"). Querying the single bit i with the command `OVLS? n` where the bit-mask $n = 2^i$ will only clear the bit i. For instance, issuing the command `OVLS? 1` will clear the bit 0 only. The real-time state of the monitored signals is available in the OVLC register.

Weight $n = 2^i$	**Bit** i	**Flag**	**Description**
128	7	RFU	Undefined (read 0)
64	6	RFU	Undefined (read 0)
32	5	RFU	Undefined (read 0)
16	4	CLN	Negative command voltage limiter engaged (bit set)
8	3	CLP	Positive command voltage limiter engaged (bit set)
4	2	FBK	Feedback input frequency outside the allowed range (bit set)
2	1	REF	Reference input frequency outside outside the allowed range (bit set)
1	0	PCB	PCB temperature outside the allowed range (bit set)

Table 4: Bit definitions for the OVLS status register and the corresponding OVLC condition and OVLE enable registers.

2.2.5 Overload Enable Register (OVLE)

Each bit in the OVLE register corresponds one-to-one with a bit in the OVLS register, and acts as a bitwise logical-AND of the flags to generate the OVL flag in the Master Summary Status (MSTS) register. The OVLE register is set and queried with the `OVLE` command and cleared at power-on.

2.2.6 Overload Condition Register (OVLC)

Each bit in the OVLC register corresponds one-to-one with a bit in the OVLS register. The bits in the OVLC register reflect the real-time values of their corresponding signals. Reading the entire register, or individual bits within it, does not affect the value of OVLC. This register is queried with the `OVLC` command. All bits read 0 at power-on, until the corresponding signals are first evaluated.

PRELIMINARY INFORMATION

2.2.7 Master Summary Status Register (MSTS)

The Master Summary Status register is the top-level summary of the status model. It consists of eight flags that summarize the state of all the other status registers. A summary flag is set when at least one bit is set in both the corresponding status register and its enable register. The MSS flag (bit 0) is the master summary: it is set when any other summary flag is set and enabled in the MSTE register. When a bit set in MSTS is unmasked by the MSTE register, the STATUS signal is asserted on the related connector.

Weight $n = 2^i$	**Bit** i	**Flag**	**Description**
128	7	OVL	Overload Status Summary
64	6	INS	Instrument Status Summary
32	5	RFU	Undefined (read 0)
16	4	RFU	Undefined (read 0)
8	3	RFU	Undefined (read 0)
4	2	EVT	Event Status Summary
2	1	COM	Communications Status Summary
1	0	MSS	Master Summary Status

Table 5: Bit definitions for the MSTS status register and the corresponding MSTE enable register.

2.2.8 Master Summary Enable Register (MSTE)

Each bit in the MSTE register corresponds one-to-one with a bit in the Master Summary Status (MSTS) register, and acts as a bitwise logical-AND to generate the MSS flag. Bit 0 of MSTE is undefined—setting it has no effect, and reading it always returns 0. This register is set and queried with the `MSTE` command and cleared at power-on.

PRELIMINARY INFORMATION

2.2.9 Event Summary Status Register (EVTS)

The Event Summary Status register consists of eight event flags, which record general events of the instrument, such as power-on, operation completion, buffer clearing, and command, execution or instrument errors. These flags are all "sticky bits" that are set by the corresponding event, and cleared only by reading or with the `*CLS` command. Querying the single bit i with the command `EVTS? n` where the bit-mask $n = 2^i$ will only clear the bit i. For instance, issuing the command `EVTS? 1` will clear the bit 0 only.

Weight $n = 2^i$	**Bit** i	**Flag**	**Description**
128	7	INS	Instrument error detected
64	6	URQ	User request
32	5	TXQ	TX buffer cleared
16	4	RXQ	RX buffer cleared
8	3	EXE	Command cannot execute
4	2	CMD	Parser-detected error
2	1	OPC	Operation complete. Set by the *OPC command
1	0	PON	Set at power-on

Table 6: Bit definitions for the EVTS status register and the corresponding EVTE enable register.

2.2.10 Event Summary Enable Register (EVTE)

Each bit in the EVTE register corresponds one-to-one with a bit in the EVTS register, and acts as a bitwise logical-AND of the flags to generate the EVT flag in the Master Summary Status (MSTS) register. The EVTE register is set and queried with the `EVTE` command and cleared at power-on.

PRELIMINARY INFORMATION

2.2.11 Communications Status Register (COMS)

This status register and related ones are currently unused. They are retained as placeholders for future communications status reporting.

Weight	Bit	Flag	Description
$n = 2^i$	i		
128	7	RFU	Undefined (read 0)
64	6	RFU	Undefined (read 0)
32	5	RFU	Undefined (read 0)
16	4	RFU	Undefined (read 0)
8	3	RFU	Undefined (read 0)
4	2	RFU	Undefined (read 0)
2	1	RFU	Undefined (read 0)
1	0	RFU	Undefined (read 0)

Table 7: Bit definitions for the COMS status register and the corresponding COME enable register.

2.2.12 Communications Enable Register (COME)

Each bit in the COME register corresponds one-to-one with a bit in the COMS register, and acts as a bitwise logical-AND of the flags to generate the COM flag in the Master Summary Status (MSTS) register. The COME register is set and queried with the `COME` command and cleared at power-on.

PRELIMINARY INFORMATION

2.2.13 Last Execution Error Register (LEXE)

The last error detected during command execution is stored in the LEXE register. Its content is cleared after it is read. The code of the last event can be retrieved with the `LEXE?` command. When an event is recorded, the corresponding bit (EXE) in the Event Summary Status register (EVTS) is set.

Value	Description
0	No execution error since the last `LEXE?`
1	Invalid parameter
2	Argument value out of range
3	Execution caused parameters to be adapted or clamped
4	Command not executed because it conflicts with the current operation
5	No change occurred upon executing the command
6	Operation aborted due to a fault condition

Table 8: Event codes for the LEXE last-event register

PRELIMINARY INFORMATION

2.2.14 Last Command Error Register (LCMD)

The last error detected during command parsing is stored in the LCMD register. Its content is cleared after it is read. The code of the last event can be retrieved with the `LCMD?` command. When an event is recorded, the corresponding bit (CMD) in the Event Summary Status register (EVTS) is set.

Value	Description
0	No command error since the last `LCMD?`
1	Illegal (unknown) command
2	Illegal query
3	Illegal set (read-only command)
4	Extra parameter
5	Missing parameter
6	Null command

Table 9: Event codes for the LCMD last-event register

PRELIMINARY INFORMATION

2.2.15 Last Instrument Error Register (LINS)

The last error detected during instrument operation is stored in the LINS register. Its content is cleared after it is read. The code of the last event can be retrieved with the `LINS?` command. When an event is recorded, the corresponding bit (INS) in the Event Summary Status register (EVTS) is set.

Value	Description
0	No instrument error since the last `LINS?`
1	On-chip ADC error
2	Invalid hardware state detected
3	Some parameters have been adapted or clamped
4	Some features have been disabled
5	External ADC error
6	External DAC error

Table 10: Event codes for the LINS last-event register

PRELIMINARY INFORMATION

2.2.16 Last User Request Register (LURQ)

The last user request is stored in the LURQ register. Since these requests are related to a specific instrument, the list may be empty when the user interface is only the remote control port. Its content is cleared after it is read. The code of the last event can be retrieved with the `LURQ?` command. When an event is recorded, the corresponding bit (URQ) in the Event Summary Status register (EVTS) is set.

Value	Description
0	No user request since the last `LURQ?`

Table 11: Event codes for the LURQ last-event register

PRELIMINARY INFORMATION

3 List of Commands

This section provides syntax and operational descriptions for remote commands.

Instrument Setting	OFSV	Offset
	LIMP	Positive Limit
	LIMN	Negative Limit
	INTF	Integrator Frequency
Instrument Configuration	CMDE	Command Output State
	XERE	External Error Input State
	INVE	Error Inverter State
	INTE	Integrator State
	OFSE	Offset State
	CMDS	Command Selection
Instrument Monitoring	VEAV	Average Error Voltage
	VEPP	Positive Peak Error Voltage
	VENP	Negative Peak Error Voltage
	VCAV	Average Command Voltage
	VCPP	Positive Peak Command Voltage
	VCNP	Negative Peak Command Voltage
	CRGU	Command Voltage Range Upper Bound
	CRGL	Command Voltage Range Lower Bound
	FREF	Reference Input Frequency
	FFBK	Feedback Input Frequency
	MONS	Monitoring Output Selection
	STMN	Number of Measurements to Stream
	STME	Data Streaming Enable
	TPCB	Board Temperature
Status Reporting	INSS	Instrument Status Register
	INSE	Instrument Enable Register
	INSC	Instrument Condition Register
	OVLS	Overload Status Register
	OVLE	Overload Enable Register
	OVLC	Overload Condition Register
	EVTS	Event Status Summary Register
	EVTE	Event Summary Enable Register
	COMS	Communications Status Register
	COME	Communications Enable Register
	MSTS	Master Summary Status Register
	MSTE	Master Summary Enable Register
	*CLS	Clear Status Registers
Interface	LEXE	Last Execution Error
	LINS	Last Instrument Error
	LCMD	Last Command Error
	LURQ	Last User Request
	*IDN	Identify
	*OPC	Operation Complete
	*RST	Reset
	CONS	Console Mode
	TERM	Response Termination
Memory	*RCL	Recall Settings
	*SAV	Save Current Settings

Table 12: Commands by group

PRELIMINARY INFORMATION

OFSV (Offset)

Group	Instrument Setting
Action	Set/Query
Syntax	`OFSV (?) {y}`
Description	Set (query) the command offset {to y}, in volts, with a resolution of 10 millivolts.
Allowed values	$-12.00 \leq y \leq 12.00$
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0.00**
Streamable (header)	Yes (`OFSV`)
Related commands	`OFSE` See command index
Example	`OFSV 10.00; OFSV ?` *`10.00`*

PRELIMINARY INFORMATION

LIMP (Positive Limit)

Group	Instrument Setting
Action	Set/Query
Syntax	`LIMP (?) {y}`
Description	Set (query) the positive limit for the command output signal {to y}, in volts, with a resolution of 10 millivolts.
Allowed values	$0.01 \leq y \leq$ **12.00**
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**12.00**
Streamable (header)	Yes (`LIMP`)
Related commands	`LIMN` See command index
Example	`LIMP 9.99; LIMP ?` *`9.99`*

PRELIMINARY INFORMATION

LIMN (Negative Limit)

Group	Instrument Setting
Action	Set/Query
Syntax	`LIMN (?) {y}`
Description	Set (query) the negative limit for the command output signal {to y}, in volts, with a resolution of 10 millivolts.
Allowed values	$-\mathbf{12.00} \leq y \leq -0.01$
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	$-\mathbf{12.00}$
Streamable (header)	Yes (`LIMN`)
Related commands	`LIMP` See command index
Example	`LIMN -9.99; LIMN ?` *`-9.99`*

PRELIMINARY INFORMATION

INTF (Integrator Frequency)

Group	Instrument Setting
Action	Set/Query
Syntax	`INTF (?) {m}`
Description	Set (query) the corner frequency for the integrator of the main loop filter {to m}. The loop filter is implemented as a Proportional-Integral compensator whose corner frequencies range from 100 Hz to 300 kHz in a 1-3-10 sequence. The proportional gain is fixed to unity.
Allowed values	$m \in \{\mathbf{1}, 2, 3, 4, 5, 6, 7, 8\}$ **1** $\longleftrightarrow$ FC1 : 100 Hz 2 $\longleftrightarrow$ FC2 : 300 Hz 3 $\longleftrightarrow$ FC3 : 1 kHz 4 $\longleftrightarrow$ FC4 : 3 kHz 5 $\longleftrightarrow$ FC5 : 10 kHz 6 $\longleftrightarrow$ FC6 : 30 kHz 7 $\longleftrightarrow$ FC7 : 100 kHz 8 $\longleftrightarrow$ FC8 : 300 kHz
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**1**
Streamable (header)	Yes (`INTF`)
Related commands	`INTE` See command index
Example	`INTF 1`

PRELIMINARY INFORMATION

CMDE (Command Output State)

Group	Instrument Configuration
Action	Set/Query
Syntax	`CMDE (?) {b}`
Description	Set (query) the command output state {to b}. When $b = 1$ (resp. 0), the command output signal is enabled (resp. disabled). When disabled, the command output is tied to 0 V; when enabled, it is provided by the selected loop filter and the offset DAC.
Allowed values	$b \in \{\mathbf{0}, 1\}$ **0** $\longleftrightarrow$ OFF : command output disabled 1 $\longleftrightarrow$ ON : command output enabled
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0**
Streamable (header, pos, mask)	Yes (`CTLA`, 0, 1)
Related commands	`CMDS` See command index
Example	`CMDE 1; CMDE ?` *1*

PRELIMINARY INFORMATION

XERE (External Error Input State)

Group	Instrument Configuration
Action	Set/Query
Syntax	`XERE (?) {b}`
Description	Set (query) the external error input state {to b}. When $b = 1$ (resp. 0), the external error input is enabled (resp. disabled). When disabled, the error is provided by the Phase-Frequency Detector, as required for normal operation; when enabled the voltage applied to the XERI input is routed to the Spurious filter, providing a user-controlled excitation.
Allowed values	$b \in \{\mathbf{0}, 1\}$ **0** $\longleftrightarrow$ OFF : external error input disabled 1 $\longleftrightarrow$ ON : external error input enabled
Power-on value	Restored from non-volatile memory (*cf.* `*SAV`).
Reset value (after `*RST`)	**0**
Streamable (header, pos, mask)	Yes (`CTLA`, 1, 2)
Related commands	`CMDS` See command index
Example	`XERE 1; XERE ?` *1*

PRELIMINARY INFORMATION

INVE (Error Inverter State)

Group	Instrument Configuration
Action	Set/Query
Syntax	`INVE (?) {b}`
Description	Set (query) the phase error inverter state {to b}. When $b = 1$ (resp. 0), the polarity inverter is enabled (resp. disabled). Switching between these two states may be required to set the correct loop sign that ensures stable operation.
Allowed values	$b \in \{\mathbf{0}, 1\}$ **0** $\longleftrightarrow$ OFF : error inverter disabled 1 $\longleftrightarrow$ ON : error inverter enabled
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0**
Streamable (header, pos, mask)	Yes (`CTLA`, 2, 4)
Related commands	`CMDS` See command index
Example	`INVE 1; INVE ?` *1*

PRELIMINARY INFORMATION

INTE (Integrator State)

Group	Instrument Configuration
Action	Set/Query
Syntax	`INTE (?) {b}`
Description	Set (query) the integrator state {to b}. When $b = 1$ (resp. 0), the integrator is enabled (resp. disabled). When disabled, the current integrator frequency is not altered; the integrator circuitry is bypassed and the main loop filter behaves as a unity-gain link.
Allowed values	$b \in \{\mathbf{0}, 1\}$ **0** $\longleftrightarrow$ OFF : Integrator disabled 1 $\longleftrightarrow$ ON : Integrator enabled
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0**
Streamable (header, pos, mask)	Yes (`CTLA`, 3, 8)
Related commands	`INTF` See command index
Example	`INTE 1; INTE ?` *1*

PRELIMINARY INFORMATION

OFSE (Offset State)

Group	Instrument Configuration
Action	Set/Query
Syntax	`OFSE (?) {b}`
Description	Set (query) the offset state {to b}. When $b = 1$ (resp. 0), the offset is enabled (resp. disabled). The offset allows the plant to be tuned or biased close to the locking point. It can also be used to remove the mean value from the loop filter output.
Allowed values	$b \in \{\mathbf{0}, 1\}$ **0** $\longleftrightarrow$ OFF : command offset disabled 1 $\longleftrightarrow$ ON : command offset enabled
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0**
Streamable (header, pos, mask)	Yes (`CTLA`, 4, 16)
Related commands	`OFSV` See command index
Example	`OFSE 1; OFSE ?` *1*

PRELIMINARY INFORMATION

CMDS (Command Selection)

Group	Instrument Configuration
Action	Set/Query
Syntax	`CMDS (?) {m}`
Description	Set (query) which loop filter signal is used for the command output {to m}. Issuing CMDS with the required argument selects the loop filter signal from either the main or optional loop filter. Open-loop operation is also provided by setting the source to 0 V; in this case the offset voltage remains available.
Allowed values	$m \in \{\mathbf{0}, 1, 2\}$ **0** $\longleftrightarrow$ GND : ground voltage (0 V) 1 $\longleftrightarrow$ MLF : main loop filter 2 $\longleftrightarrow$ OLF : optional loop filter
Power-on value	Restored from non-volatile memory (*cf.* `*SAV`).
Reset value (after `*RST`)	**0**
Streamable (header)	Yes (`CMDS`)
Related commands	See command index
Example	`CMDS 1`

PRELIMINARY INFORMATION

VEAV (Average Error Voltage)

Group	Instrument Monitoring
Action	Query-only
Syntax	`VEAV ?`
Description	Return to the host computer the last measurement of the average phase error voltage, in V, with a resolution of 1 mV.
Allowed values	$0.000 \leq x \leq 0.000$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`VEAV`)
Related commands	`VEPP`, `VENP` See command index
Example	`VEAV ?` *`-0.007`*

PRELIMINARY INFORMATION

VEPP (Positive Peak Error Voltage)

Group	Instrument Monitoring
Action	Query-only
Syntax	`VEPP ?`
Description	Return to the host computer the last measurement of the peak (positive) phase error voltage, in V, with a resolution of 1 mV.
Allowed values	$0.000 \leq x \leq 0.000$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`VEPP`)
Related commands	`VEAV`, `VENP` See command index
Example	`VEPP ?` *`0.221`*

PRELIMINARY INFORMATION

VENP (Negative Peak Error Voltage)

Group	Instrument Monitoring
Action	Query-only
Syntax	`VENP ?`
Description	Return to the host computer the last measurement of the peak (negative) phase error voltage, in V, with a resolution of 1 mV.
Allowed values	$0.000 \leq x \leq 0.000$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`VENP`)
Related commands	`VEAV`, `VEPP` See command index
Example	`VENP ?` *`-0.011`*

PRELIMINARY INFORMATION

VCAV (Average Command Voltage)

Group	Instrument Monitoring
Action	Query-only
Syntax	`VCAV ?`
Description	Return to the host computer the last measurement of the average command voltage, in V, with a resolution of 1 mV.
Allowed values	$0.000 \leq x \leq 0.000$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`VCAV`)
Related commands	`VCPP`, `VCNP` See command index
Example	`VCAV ?` *8.017*

PRELIMINARY INFORMATION

VCPP (Positive Peak Command Voltage)

Group	Instrument Monitoring
Action	Query-only
Syntax	`VCPP ?`
Description	Return to the host computer the last measurement of the peak (positive) command voltage, in V, with a resolution of 1 mV.
Allowed values	$0.000 \leq x \leq 0.000$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`VCPP`)
Related commands	`VCAV`, `VCNP` See command index
Example	`VCPP ?` *3.545*

PRELIMINARY INFORMATION

VCNP (Negative Peak Command Voltage)

Group	Instrument Monitoring
Action	Query-only
Syntax	`VCNP ?`
Description	Return to the host computer the last measurement of the peak (negative) command voltage, in V, with a resolution of 1 mV.
Allowed values	$0.000 \leq x \leq 0.000$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`VCNP`)
Related commands	`VCAV`, `VCPP` See command index
Example	`VCNP ?` *`-5.021`*

PRELIMINARY INFORMATION

CRGU (Command Voltage Range Upper Bound)

Group	Instrument Monitoring
Action	Set/Query
Syntax	`CRGU (?) {y}`
Description	Set (query) the upper voltage bound for the command window comparator, in V, with a resolution of 1 mV.
Allowed values	$-12.100 \leq y \leq 12.100$
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0.000**
Streamable (header)	Yes (`CRGU`)
Related commands	`CRGL` See command index
Example	`CRGU 10.0; CRGU ?` *`10.000`*

PRELIMINARY INFORMATION

CRGL (Command Voltage Range Lower Bound)

Group	Instrument Monitoring
Action	Set/Query
Syntax	`CRGL (?) {y}`
Description	Set (query) the lower voltage bound for the command window comparator, in V, with a resolution of 1 mV.
Allowed values	$-12.100 \leq y \leq 12.100$
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0.000**
Streamable (header)	Yes (`CRGL`)
Related commands	`CRGU` See command index
Example	`CRGL -10.0; CRGL ?` *`-10.000`*

PRELIMINARY INFORMATION

FREF (Reference Input Frequency)

Group	Instrument Monitoring
Action	Query-only
Syntax	`FREF ?`
Description	Return to the host computer the last measurement of the reference input frequency, in megahertz, with a resolution of 0.1 MHz.
Allowed values	$8.1 \leq x \leq 1024.0$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`FREF`)
Related commands	`FFBK` See command index
Example	`FREF ?` *`99.9`*

PRELIMINARY INFORMATION

FFBK (Feedback Input Frequency)

Group	Instrument Monitoring
Action	Query-only
Syntax	`FFBK ?`
Description	Return to the host computer the last measurement of the feedback input frequency, in megahertz, with a resolution of 0.1 MHz.
Allowed values	$8.1 \leq x \leq 1024.0$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`FFBK`)
Related commands	`FREF` See command index
Example	`FFBK?` *`500.1`*

PRELIMINARY INFORMATION

MONS (Monitoring Output Selection)

Group	Instrument Monitoring
Action	Set/Query
Syntax	`MONS (?) {m}`
Description	Set (query) the signal source for the Monitoring output {to m}. Issuing MONS with the required argument selects which signal is routed to the MONO connector.
Allowed values	$m \in \{\mathbf{0}, 1, 2, 3, 4, 5, 6\}$ **0** ⟷ GND : ground voltage (0V) 1 ⟷ CMD : command voltage 2 ⟷ EFB : phase error – full bandwidth 3 ⟷ EAC : phase error – AC-coupled with voltage gain 4 ⟷ ELF : phase error – low frequency spectrum 5 ⟷ VLD : voltage lock detector signal 6 ⟷ STS : /STATUS signal
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0**
Streamable (header)	Yes (`MONS`)
Related commands	See command index
Example	`MONS 1`

PRELIMINARY INFORMATION

STMN (Number of Measurements to Stream)

Group	Instrument Monitoring
Action	Set/Query
Syntax	`STMN (?) {m}`
Description	Set (query) the number of measurements to stream out {to m}. When $m = 0$, measurements will be output indefinitely, until streaming is disabled with STME. If $m > 0$, data streaming will stop once m measurements have been output. Measurements are streamed out at a constant rate of approximately one measurement per second.
Allowed values	$\mathbf{0} \leq m \leq 65535$
Power-on value	Restored from non-volatile memory (*cf* `*SAV`).
Reset value (after `*RST`)	**0**
Streamable	No
Related commands	`STME` See command index
Example	`STMN 1000; STMN ?` *1000*

PRELIMINARY INFORMATION

STME (Data Streaming Enable)

Group	Instrument Monitoring
Action	Set/Query
Syntax	`STME (?) {b}`
Description	Set (query) the state of the data streaming functionality {to b}. If $b = 0$ (resp. 1), data streaming is disabled (resp. enabled). When data streaming is enabled, the number of measurements that will be streamed out is specified using the STMN command.
Allowed values	$b \in \{\mathbf{0}, 1\}$ **0** $\longleftrightarrow$ OFF : data streaming disabled. 1 $\longleftrightarrow$ ON : data streaming enabled.
Power-on value	0
Reset value (after `*RST`)	**0**
Streamable	No
Related commands	`STMN` See command index
Example	`STME 1; STME ?` *1*

PRELIMINARY INFORMATION

TPCB (Board Temperature)

Group	Instrument Monitoring
Action	Query-only
Syntax	`TPCB ?`
Description	Return to the host computer the last measurement of the printed circuit board temperature, in °C, with a resolution of 0.1 °C.
Allowed values	$0.0 \leq x \leq 0.0$
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable (header)	Yes (`TPCB`)
Related commands	`MONS` See command index
Example	`TPCB ?` *`25.6`*

PRELIMINARY INFORMATION

INSS (Instrument Status Register)

Group	Status Reporting
Action	Query-only
Syntax	`INSS ? [n]`
Description	Read the Instrument Status register [with bit-mask n].
Allowed values	$0 \leq n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`INSS`)
Related commands	`INSE`, `INSC` See command index
Example	`INSS ?`
	4

PRELIMINARY INFORMATION

INSE (Instrument Enable Register)

Group	Status Reporting
Action	Set/Query
Syntax	`INSE (?) [n] {m}`
Description	Set (query) the Instrument Enable register [with bit-mask n] {to bit-mask m}. Each bit in the INSE register corresponds one-to-one with a bit in the INSS register, and acts as a bitwise AND of the INSS flags to generate the INS flag in the Master Summary Status (MSTS) register.
Allowed values	$0 \leq m, n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`INSE`)
Related commands	`INSS`, `INSC` See command index
Example	`INSE 2`

PRELIMINARY INFORMATION

INSC (Instrument Condition Register)

Group	Status Reporting
Action	Query-only
Syntax	`INSC ? [n]`
Description	Read the Instrument Condition register [with bit-mask n]. The values of the bits in the INSC condition register are determined by the current (real-time) condition of the events defined in the INSS status register. Reading the condition register does not affect the register.
Allowed values	$0 \leq n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`INSC`)
Related commands	`INSS`, `INSE` See command index
Example	`INSC ? 2`

PRELIMINARY INFORMATION

OVLS (Overload Status Register)

Group	Status Reporting
Action	Query-only
Syntax	`OVLS ? [n]`
Description	Read the Overload Status register [with bit-mask n].
Allowed values	$0 \leq n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`OVLS`)
Related commands	`OVLE`, `OVLC` See command index
Example	`OVLS ?`
	4

PRELIMINARY INFORMATION

OVLE (Overload Enable Register)

Group	Status Reporting
Action	Set/Query
Syntax	`OVLE (?) [n] {m}`
Description	Set (query) the Overload Enable register [with bit-mask n] {to bit-mask m}. Each bit in the OVLE register corresponds one-to-one with a bit in the OVLS register, and acts as a bitwise AND of the OVLS flags to generate the OVL flag in the Master Summary Status (MSTS) register.
Allowed values	$0 \leq m, n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`OVLE`)
Related commands	`OVLS`, `OVLC` See command index
Example	`OVLE 2`

PRELIMINARY INFORMATION

OVLC (Overload Condition Register)

Group	Status Reporting
Action	Query-only
Syntax	`OVLC ? [n]`
Description	Read the Overload Condition register [with bit-mask n]. The values of the bits in the OVLC condition register are determined by the current (real-time) condition of the events defined in the OVLS status register. Reading the condition register does not affect the register.
Allowed values	$0 \leq n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`OVLC`)
Related commands	`OVLS`, `OVLE` See command index
Example	`OVLC ? 2`

PRELIMINARY INFORMATION

EVTS (Event Status Summary Register)

Group	Status Reporting
Action	Query-only
Syntax	`EVTS ? [n]`
Description	Read the Event Status Summary register [with bit-mask n].
Allowed values	$0 \leq n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`EVTS`)
Related commands	`EVTE` See command index
Example	`EVTS ?` *4*

PRELIMINARY INFORMATION

EVTE (Event Summary Enable Register)

Group	Status Reporting	
Action	Set/Query	
Syntax	`EVTE (?) [n] {m}`	
Description	Set (query) the Event Summary Enable register [with bit-mask n] {to bit-mask m}. Each bit in the EVTE register corresponds one-to-one with a bit in the EVTS register, and acts as a bitwise AND of the EVTS flags to generate the EVT flag in the Master Summary Status (MSTS) register. This register is set and queried with the EVTE command and cleared at power-on.	
Allowed values	$0 \leq m, n \leq 255$	
Power-on value	0	
Reset value (after `*RST`)	Not affected by `*RST`.	
Streamable (header)	Yes (`EVTE`)	
Related commands	`EVTS`	See command index
Example	`EVTE 2`	

PRELIMINARY INFORMATION

COMS (Communications Status Register)

Group	Status Reporting
Action	Query-only
Syntax	`COMS ? [n]`
Description	Read the Communications Status register [with bit-mask n].
Allowed values	$0 \leq n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable	No
Related commands	`COME` See command index
Example	`COMS ?`
	4

PRELIMINARY INFORMATION

COME (Communications Enable Register)

Group	Status Reporting	
Action	Set/Query	
Syntax	`COME (?) [n] {m}`	
Description	Set (query) the Communications Enable register [with bit-mask n] {to bit-mask m}. Each bit in the COME register corresponds one-to-one with a bit in the COMS register, and acts as a bitwise AND of the COMS flags to generate the COM flag in the Master Summary Status (MSTS) register. This register is set and queried with the COME command and cleared at power-on.	
Allowed values	$0 \leq m, n \leq 255$	
Power-on value	0	
Reset value (after `*RST`)	Not affected by `*RST`.	
Streamable	No	
Related commands	`COMS`	See command index
Example	`COME 2`	

PRELIMINARY INFORMATION

MSTS (Master Summary Status Register)

Group	Status Reporting
Action	Query-only
Syntax	`MSTS ? [n]`
Description	Return the Master Summary Status register [with bit-mask n]. The execution of the MSTS? query – without the optional bit-mask [n] – always causes the STATUS signal to be de-asserted. Note that `MSTS? n` will not clear the STATUS, even if the bit $i \mid n = 2^i$ is the only bit currently asserting the STATUS signal.
Allowed values	$0 \leq n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable	No
Related commands	`MSTE` See command index
Example	`MSTS?; MSTS? 128;` *129* *128*

PRELIMINARY INFORMATION

MSTE (Master Summary Enable Register)

Group	Status Reporting
Action	Set/Query
Syntax	`MSTE (?) [n] {m}`
Description	Set (query) the Master Summary Enable register [with bit-mask n] {to bit-mask m}. The set-form command will clear the bits outside the bit-mask. Each bit in the MSTE register corresponds one-to-one with a bit in the MSTS register, and acts as a bitwise AND of the MSTS flags to generate the MSS flag. Bit 0 of the MSTE is undefined—setting it has no effect, and reading it always returns 0. This register is set and queried with the MSTE command and cleared at power-on.
Allowed values	$0 \leq m, n \leq 255$
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable	No
Related commands	`MSTS` See command index
Example	`MSTE 128; MSTE?` *128*

PRELIMINARY INFORMATION

*CLS (Clear Status Registers)

Group	Status Reporting
Action	Set-only
Syntax	`*CLS`
Description	Immediately clear all status registers. This command clears the following registers: LEXE, LCMD, LINS, LURQ, INSS, OVLS, COMS, and EVTS.
Allowed values	Not applicable.
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable	No
Related commands	See command index
Example	`*CLS`

PRELIMINARY INFORMATION

LEXE (Last Execution Error)

Group	Interface	
Action	Query-only	
Syntax	`LEXE ?`	
Description	Return the last execution error. The LEXE register holds the last error detected during the execution of a command. The related error code can be retrieved with the command LEXE?. When such an error is detected, the corresponding bit in the Event Summary Status register is set (bit EXE in EVTS).	
Allowed values	See LEXE register.	
Power-on value	0	
Reset value (after `*RST`)	Not affected by `*RST`.	
Streamable (header)	Yes (`LEXE`)	
Related commands	`LURQ`, `LCMD`, `LINS`	See command index
Example	`CONS 2;LEXE?;LEXE?` *1* *0*	

PRELIMINARY INFORMATION

LINS (Last Instrument Error)

Group	Interface
Action	Query-only
Syntax	`LINS ?`
Description	Return the last instrument error. `LINS?` returns the unique code number associated with this error.
Allowed values	See LINS register.
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`LINS`)
Related commands	`LURQ`, `LEXE`, `LCMD` See command index
Example	`LINS?` *0*

PRELIMINARY INFORMATION

LCMD (Last Command Error)

Group	Interface
Action	Query-only
Syntax	`LCMD ?`
Description	Return the last command error.
	`LCMD?` returns the unique code number associated with this error.
Allowed values	See LCMD register.
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`LCMD`)
Related commands	`LURQ`, `LEXE`, `LINS` See command index
Example	`*RST?; LCMD?`
	2

PRELIMINARY INFORMATION

LURQ (Last User Request)

Group	Interface
Action	Query-only
Syntax	`LURQ ?`
Description	Return the last user request. `LURQ?` returns the unique code number associated with this event.
Allowed values	See LURQ register.
Power-on value	0
Reset value (after `*RST`)	Not affected by `*RST`.
Streamable (header)	Yes (`LURQ`)
Related commands	`LINS`, `LEXE`, `LCMD` See command index
Example	`LURQ?` *0*

PRELIMINARY INFORMATION

*IDN (Identify)

Group	Interface
Action	Query-only
Syntax	`*IDN ?`
Description	Read the device identification string. This string is formatted as: Signals and Systems for Physics, model <model>, hw <hardware revision>, fw <firmware revision>, s/n <serial number>.
Allowed values	Not applicable.
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable	No
Related commands	See command index
Example	`*IDN` *`Signals and Systems for Physics, model SK633, hw R26A, fw R26A, s/n 123456.`*

PRELIMINARY INFORMATION

*OPC (Operation Complete)

Group	Interface
Action	Set/Query
Syntax	`*OPC (?)`
Description	Set the OPC flag in the EVTS register. The query form *OPC? returns 1 once all pending operations are complete, without modifying the EVTS register..
Allowed values	Not applicable.
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable	No
Related commands	See command index
Example	`*OPC?` *1*

PRELIMINARY INFORMATION

*RST (Reset)

Group	Interface
Action	Set-only
Syntax	`*RST`
Description	Reset the instrument to its default configuration. When a parameter is affected by the *RST command, its value is reset according to the information given by the Reset value field within the related command section. Although status registers are unaffected by *RST, the content of some condition registers may change upon resetting the instrument.
Allowed values	Not applicable.
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable	No
Related commands	See command index
Example	`*RST`

PRELIMINARY INFORMATION

CONS (Console Mode)

Group	Interface
Action	Set/Query
Syntax	`CONS (?) {b}`
Description	Set (query) the Console mode {to b}. `CONS 1` causes each character received to be echoed back to the host computer.
Allowed values	$b \in \{\mathbf{0}, 1\}$ **0** $\longleftrightarrow$ OFF : console mode disabled. 1 $\longleftrightarrow$ ON : console mode enabled.
Power-on value	0
Reset value (after `*RST`)	**0**
Streamable	No
Related commands	See command index
Example	`CONS 1`

PRELIMINARY INFORMATION

TERM (Response Termination)

Group	Interface
Action	Set/Query
Syntax	`TERM (?) {m}`
Description	Set (query) the termination sequence mode {to m}. The termination sequence is appended to all query responses sent by the instrument. It consists of the ASCII character CR (carriage return), LF (line feed), or both.
Allowed values	$m \in \{1,\ 2,\ \mathbf{3},\ 4\}$ $1 \longleftrightarrow$ CR : carriage return appended $2 \longleftrightarrow$ LF : line-feed appended $\mathbf{3} \longleftrightarrow$ CRLF : CR + LF appended $4 \longleftrightarrow$ NONE : no character appended
Power-on value	3
Reset value (after `*RST`)	**3**
Streamable	No
Related commands	See command index
Example	`TERM ?` *3*

PRELIMINARY INFORMATION

*RCL (Recall Settings)

Group	Memory
Action	Set-only
Syntax	`*RCL`
Description	Recall the settings stored in the non-volatile memory.
Allowed values	Not applicable.
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable	No
Related commands	`*RST`, `*SAV`
Example	`*RCL`

See command index

PRELIMINARY INFORMATION

*SAV (Save Current Settings)

Group	Memory
Action	Set-only
Syntax	`*SAV`
Description	Save the current settings in the non-volatile memory.
Allowed values	Not applicable.
Power-on value	Not applicable.
Reset value (after `*RST`)	Not applicable.
Streamable	No
Related commands	`*RST`, `*RCL`
Example	`*SAV`

See command index

PRELIMINARY INFORMATION

4 Index of Commands

Instrument Configuration

- CMDE (Command Output State), 23
- CMDS (Command Selection), 28
- INTE (Integrator State), 26
- INVE (Error Inverter State), 25
- OFSE (Offset State), 27
- XERE (External Error Input State), 24

Instrument Monitoring

- CRGL (Command Voltage Range Lower Bound), 36
- CRGU (Command Voltage Range Upper Bound), 35
- FFBK (Feedback Input Frequency), 38
- FREF (Reference Input Frequency), 37
- MONS (Monitoring Output Selection), 39
- STME (Data Streaming Enable), 41
- STMN (Number of Measurements to Stream), 40
- TPCB (Board Temperature), 42
- VCAV (Average Command Voltage), 32
- VCNP (Negative Peak Command Voltage), 34
- VCPP (Positive Peak Command Voltage), 33
- VEAV (Average Error Voltage), 29
- VENP (Negative Peak Error Voltage), 31
- VEPP (Positive Peak Error Voltage), 30

Instrument Setting

- INTF (Integrator Frequency), 22
- LIMN (Negative Limit), 21
- LIMP (Positive Limit), 20
- OFSV (Offset), 19

Interface

- *IDN (Identify), 60
- *OPC (Operation Complete), 61
- *RST (Reset), 62
- CONS (Console Mode), 63
- LCMD (Last Command Error), 58
- LEXE (Last Execution Error), 56
- LINS (Last Instrument Error), 57
- LURQ (Last User Request), 59
- TERM (Response Termination), 64

Memory

- *RCL (Recall Settings), 65
- *SAV (Save Current Settings), 66

Status Reporting

- *CLS (Clear Status Registers), 55
- COME (Communications Enable Register), 52
- COMS (Communications Status Register), 51
- EVTE (Event Summary Enable Register), 50
- EVTS (Event Status Summary Register), 49
- INSC (Instrument Condition Register), 45
- INSE (Instrument Enable Register), 44
- INSS (Instrument Status Register), 43
- MSTE (Master Summary Enable Register), 54
- MSTS (Master Summary Status Register), 53
- OVLC (Overload Condition Register), 48
- OVLE (Overload Enable Register), 47
- OVLS (Overload Status Register), 46

PRELIMINARY INFORMATION