CR

SCPI COMMAND REFERENCE

TH2883 Series

Impulse Winding Tester — TH2883-1 / TH2883-5 / TH2883-10

Companion document to the TH2883 Series Operation Manual. Full remote-control SCPI command set over RS232C, USB (USBTMC/USBCDC) and LAN, organized by subsystem.

Techmize Electronic GmbH · Ismaning, Germany
Rev. 2026-10

TH2883 SCPI Command Reference

Impulse Winding Tester

This document lists the full remote-control command set for the TH2883-1/-5/-10. It is the companion reference to the *TH2883 Series Operation Manual*, which covers front-panel operation, the physical remote interfaces (RS232C/USB/LAN) and general operating procedures. SCPI is available over all three remote interfaces; the instrument does not support MODBUS.

Section	
1. Conventions & Reference Tables	1 page
2. Common Commands & DISPlay Subsystem	1 page
3. IVOLTage & SRATE Subsystem	1 page
4. COMParator Subsystem	3 pages
5. TRIGger, STATistic & WADJust Subsystem	1 page
6. SWAVE Subsystem	1 page
7. FETCh? Subsystem & Waveform Data Format	2 pages
8. MEASure, ABORt & MMEMory Subsystem	1 page
9. Error Messages	1 page

SCPI commands are tree-structured (max. 3 levels, colon-separated) and case-insensitive. The capitalized portion of each keyword is the mandatory short form; the lower-case portion is optional (e.g. `IVOLTage:VOLTage` *may be sent as* `IVOLT:VOLT`*). A semicolon separates multiple commands at the same tree level;* `;:` *restarts the next command from the tree root. All setting commands return the string* `"1"` *(success) or* `"0"` *(failure). Numeric formats: NR1 = integer (e.g.* `123`*), NR2 = fixed point (e.g.* `12.3`*), NR3 = floating point (e.g.* `12.3E+5`*); NL (newline, decimal 10) terminates each command line, and ^END denotes the IEEE 488 bus EOI signal on a query response.*

SECTION 1

Conventions & Reference Tables

1.1 Command types

The instrument accepts two categories of command: IEEE 488.2 common commands and SCPI (Standard Commands for Programmable Instruments) commands. Common commands are defined by the IEEE 488.2-1987 standard and apply generically across instrument types; the TH2883 series implements only a subset of these (§2.1). SCPI commands are tree-structured, up to three levels deep; the top level of each tree is called a subsystem command. A command is only valid once its subsystem (and any intermediate level) has been selected; levels are separated by a colon.

1.2 Syntax rules

- Commands are case-insensitive, including units (e.g. `DISP:PAGE MEAS` = `disp:page meas` = `DiSp:PAGeMEas`).
- No space is permitted immediately before or after a colon.
- A space separates a command from its parameter: everything before the space is the command, everything after is the parameter (e.g. in `COMP:AREA ON`, `AREA` is the command and `ON` is its parameter).
- Some commands take no parameter (e.g. the trigger command `TRIG`, the standard-waveform-selection command `SWAVE:CHOose`).
- A command may be sent abbreviated or spelled out in full; the mandatory abbreviated portion is given in capitals throughout this document (e.g. `COMParator:AREAsize ON` = `COMP:AREA ON`).
- A command immediately followed by a question mark (`?`) performs a query of that command's current value or result (e.g. `COMP:AREA?`).

1.3 Multiple commands in one line

- A semicolon (`;`) separates multiple commands at the same subsystem level within one command line (e.g. `COMP:AREA:STATe ON;RANGe 0,960` — `STATe ON` and `RANGe 0,960` are both sub-commands of `AREA`).
- A semicolon immediately followed by a colon (`;:`) restarts the next command from the top of the command tree (e.g. `COMP:AREA:STATe ON;:COMP:AREA:RANGe 0,960`).
- Common commands may be inserted at the start, middle or end of a command line, separated by semicolons, without affecting the tree-level relationships of the surrounding commands (e.g. `COMP:AREA ON;*TRG;DIFF ON`).

1.4 Abbreviation rule

- If the full-length keyword or parameter is 4 characters or fewer, the abbreviation is identical to the full form.
- If the full-length form is longer than 4 characters: when the 4th character is a vowel, the abbreviation is the first 3 characters; otherwise, the abbreviation is the first 4 characters. Example: `TIME` → `TIME`; `TRIGger` → `TRIG`; `DELete` → `DEL`; `FREQuency` → `FREQ`.
- For a keyword or parameter made up of two or more words, the full-length form is the first word's initial letter plus the complete last word; the abbreviation is the first word's initial letter plus the (possibly abbreviated) last word. Example: `Mass MEMory` → full form `MMEMory`, abbreviation `MMEM`; `Impulse VOLTage` → full form `IVOLTage`, abbreviation `IVOLT`.

1.5 Symbol conventions

Symbol	Meaning
:	Colon — moves to the next level of the command tree.
;	Semicolon — separates multiple commands on one line (§1.3).
*	Asterisk — marks the command that follows as a common (IEEE 488.2) command.
?	Question mark — marks a query.

`,`	Comma — separates multiple parameters.
(space)	Separates a command from its parameter (§1.2).
`" "`	Quotation marks — content inside is taken literally and not otherwise parsed (used for file names).
`NR1` / `NR2` / `NR3`	Integer (e.g. `123`) / fixed-point decimal (e.g. `12.3`) / floating-point (e.g. `12.3E+5`).
`NL`	Newline, decimal 10 — terminates a command string.
`^END`	IEEE 488 bus EOI (end-of-identify) signal, appended to a query response.
`<>` / `[ ]`	Angle brackets enclose a program parameter; square brackets enclose an optional item.

> **NOTE**
>
> Every setting command returns the string `"1"` once it has successfully updated the instrument's parameter and the instrument is ready for the next command, or `"0"` if the update failed (also followed by readiness for the next command).

SECTION 2

Common Commands & DISPlay Subsystem

2.1 IEEE 488.2 common commands

The TH2883 series implements only the following common commands.

Command	Details
`*RST`	Reset the instrument. Example: `WrtCmd("*RST")`.
`*TRG?`	Trigger a measurement and write the measured waveform data to the output buffer — equivalent to `TRIG` followed by `FETCh:TWAVE?`. Used only to trigger a device-under-test measurement (not a standard-waveform acquisition, §6). Valid only on the Measurement Display page; ignored on other pages and while a test is already in progress. Example: `WrtCmd("*TRG?")`.
`*IDN?`	Query the instrument identification. Returns `<product>,<version><NL^END>`, where `product` is the model designation (e.g. `TH2883-5 Impulse Winding Tester`) and `version` is the firmware version. Example: `WrtCmd("*IDN?")`.

2.2 DISPlay subsystem

Controls the active display page, waveform display mode, and the grid and secondary-display areas of the Measurement Display page.

Command	Details
`DISP:PAGE?` `DISP:PAGE <page>`	Query/switch the active display page. `page`: `MEASurement`, `MSETup`, `IFILE` (internal file list), `EFILE` (external file list), `SSETup` (system setup), `ISETup` (interface setup), `ABOut`. Ignored while a measurement is in progress. The query returns one of `MEAS DISP`, `MEAS SETUP`, `SYSTEM SETUP`, `INTFILE`, `EXT FILE`, `IO SETUP`, `ABOUT`. Example: `WrtCmd("DISP:PAGE MEAS")`.
`DISP:WAVE?` `DISP:WAVE <AON|STD|TEST|AOFF>`	Waveform display mode: `AON` shows both the standard and the test (measured) waveform; `STD` shows only the standard waveform; `TEST` shows only the test waveform; `AOFF` shows neither. Query returns `ALLON`, `ALLOFF`, `ONLY STDWAVE` or `ONLY TESTWAVE`. Example: `WrtCmd("DISP:WAVE AON")`.
`DISP:GRID?` `DISP:GRID <ON|OFF>`	Grid overlay on the Measurement Display waveform area. Query returns `ON` or `OFF`. Example: `WrtCmd("DISP:GRID ON")`.
`DISP:SECD?` `DISP:SECDisplay <ON|OFF>`	Secondary display area on/off. Query returns `ON` or `OFF`. Example: `WrtCmd("DISP:SECD ON")`.

SECTION 3

IVOLTage & SRATE Subsystem

IVOLTage controls the parameters of the applied impulse: test voltage, destructive (breakdown) voltage sweep, test and demagnetizing pulse counts, voltage adjustment, and the inter-measurement delay used with internal triggering. SRATE sets the waveform sampling rate.

> **VOLTAGE RANGE DEPENDS ON THE MODEL**
>
> TH2883-1: 30–1200 V, 5 V step · TH2883-5: 100–5000 V, 10 V step · TH2883-10: 500–10000 V, 20 V step. Program `IVOLTage:VOLTage` and `IVOLTage:BVOLTage` within the connected unit's own range (see also datasheet and Operation Manual §6).

3.1 Test voltage & destructive-test sweep

Command	Details
`IVOLT:VOLT?` `IVOLTage:VOLTage <value>`	Impulse test-voltage setting. `value`: NR1, NR2 or NR3, optionally with a `KV` or `V` suffix; valid range is the connected model's own rated impulse-voltage range (see note above) — TH2883-1: 30–1200 V, 5 V step; TH2883-5: 100–5000 V, 10 V step; TH2883-10: 500–10000 V, 20 V step. Example (TH2883-5): `WrtCmd("IVOLT:VOLT 1000V")` → 1000 V.
`IVOLT:BVOLT?` `IVOLTage:BVOLTage <start,end,step%>`	Destructive (breakdown) test voltage sweep, used to find a suitable test voltage (see Operation Manual for the destructive-test procedure). `start`/`end`: NR1, NR2 or NR3, optionally with a `KV` or `V` suffix, within the connected model's own rated impulse-voltage range (see note above); `start` must be less than `end`. `step%`: NR1, 1–50, the percentage step from `start` toward `end`. Example (TH2883-5): `WrtCmd("IVOLT:BVOLT 1000V,3000V,4")` → sweeps 1000 V to 3000 V in 4% steps.
`IVOLT:TIMP?` `IVOLTage:TIMPulse <number>`	Number of test (impulse) pulses per measurement. `number`: NR1, 1–32. Example: `WrtCmd("IVOLT:TIMP 3")`.
`IVOLT:EIMP?` `IVOLTage:EIMPulse <number>`	Number of demagnetizing pulses. `number`: NR1, 1–16. Example: `WrtCmd("IVOLT:EIMP 3")`.
`IVOLT:VADJ?` `IVOLTage:VADJust <ON|OFF>`	Voltage-adjustment function on/off. Query returns `ON` or `OFF`. Example: `WrtCmd("IVOLT:VADJ ON")`.
`IVOLT:DTIM?` `IVOLTage:DTIME <value>`	Inter-measurement delay used with internal triggering (§5.1, `TRIGger:SOURce INTernal`). `value`: NR1, NR2 or NR3, 0.1–99.9 s. Example: `WrtCmd("IVOLT:DTIM 50.1s")` → 50.1 s.

3.2 SRATE subsystem

Command	Details
`SRATE[:RATE]?` `SRATE[:RATE] <value>`	Waveform sampling rate. `value`: one of 200, 100, 50, 25, 12.5, 6.25, 3.12, 1.56 (MSa/s), given with an `m` or `msa/s` suffix. Example: `WrtCmd("SRATE:RATE 100m")` or `WrtCmd("SRATE:RATE 50msa/s")`.

SECTION 4

COMParator Subsystem

Controls the comparator (pass/fail judgment) function, including its four independent comparison methods — area, area-difference, corona and phase-difference — against the acquired standard waveform. Each method's destructive-test (breakdown-sweep) counterpart is addressed by inserting `:BDV` immediately after `COMParator` in the command root; its settings are otherwise configured identically.

4.1 Comparator enable

Command	Details
`COMP[:STATe]?` `COMParator[:STATe] <ON|OFF>`	Comparator (pass/fail judgment) function on/off. `ON`/`OFF` may also be sent as the integer `1`/`0` (this applies to every ON/OFF command in this section). Query returns `ON` or `OFF`. Example: `WrtCmd("COMP ON")`.

4.2 Area comparison

Command	Details

`COMP:AREA[:STATe]?`
`COMParator:AREAsize[:STATe]`
`<ON|OFF>`
(destructive test:
`COMParator:BDV:AREAsize[:STATe]`)
	Area-comparison method on/off. Query returns `ON` or `OFF`. Example: `WrtCmd("COMP:AREA ON")`; destructive test: `WrtCmd("COMP:BDV:AREA ON")`.

`COMP:AREA:RANG?`
`COMParator:AREAsize:RANGe`
`<start,end>`
	Comparison window, as a start/end point pair along the waveform. `start`/`end`: NR1, 0–6000, no suffix; `end` must not be less than `start`. Example: `WrtCmd("COMP:AREA:RANG 0,960")`.

`COMP:AREA:DIFF?`
`COMParator:AREAsize:DIFFerence`
`<value>`
	Allowed area-comparison deviation. `value`: NR1, NR2 or NR3, optionally with a `%` suffix. Example: `WrtCmd("COMP:AREA:DIFF 2.5")` or `WrtCmd("COMP:AREA:DIFF 2.5%")` → 2.5%.

4.3 Area-difference comparison

Command	Details

`COMP:DIFF[:STATe]?`
`COMParator:DIFFzone[:STATe]`
`<ON|OFF>`
	Area-difference comparison method on/off. Example: `WrtCmd("COMP:DIFF ON")`.

`COMP:DIFF:RANG?`
`COMParator:DIFFzone:RANGe`
`<start,end>`
	Comparison window. `start`/`end`: NR1, 0–6000, no suffix; `end` must not be less than `start`. Example: `WrtCmd("COMP:DIFF:RANG 0,960")`.

`COMP:DIFF:DIFF?`
`COMParator:DIFFzone:DIFFerence`
`<value>`
	Allowed area-difference deviation. `value`: NR1, NR2 or NR3, optionally with a `%` suffix. Example: `WrtCmd("COMP:DIFF:DIFF 2.5")` → 2.5%.

4.4 Corona comparison

Command	Details
`COMP:CORO[:STATe]?` `COMParator:CORONA[:STATe] <ON|OFF>`	Corona-discharge comparison method on/off. Example: `WrtCmd("COMP:CORONA ON")`.

`COMP:CORO:RANG?`
`COMParator:CORONA:RANGe`
`<start,end>`
	Comparison window. `start`/`end`: NR1, 0–6000, no suffix; `end` must not be less than `start`. Example: `WrtCmd("COMP:CORONA:RANG 100,200")`.

`COMP:CORO:DIFF?`
`COMParator:CORONA:DIFFerence`
`<value>`
	Allowed corona-level deviation. `value`: NR1, 0–999, no suffix. Example: `WrtCmd("COMP:CORO:DIFF 20")`.

4.5 Phase-difference comparison

Command	Details

`COMP:PHAS[:STATe]?`
`COMParator:PHASediff[:STATe]`
`<ON|OFF>`
	Phase-difference comparison method on/off. Example: `WrtCmd("COMP:PHAS ON")`.

`COMP:PHAS:POS?`
`COMParator:PHASediff:POSition`
`<value>`
	Which zero-crossing of the damped waveform the phase-difference comparison is measured at. `value`: NR1, 2–10, no suffix. Example: `WrtCmd("COMP:PHAS:POS 3")` → 3rd zero-crossing.

`COMP:PHAS:DIFF?`
`COMParator:PHASediff:DIFFerence`
`<value>`
	Allowed phase-difference deviation. `value`: NR1, NR2 or NR3, optionally with a `%` suffix. Example: `WrtCmd("COMP:PHAS:DIFF 2.5")` → 2.5%.

> **NOTE**
>
> For every comparison method above, each destructive-test (`BDV`) counterpart command follows the same syntax with `:BDV:` inserted immediately after `COMParator` (e.g. `COMParator:BDV:CORONA:RANGe`) and is not listed separately in this table.

4.6 Comparator result retrieval

The comparator does not have its own result-query command; the overall pass/fail result and the per-method comparison values are retrieved through `FETCh:CRESult?` in the FETCh? subsystem (§7.2).

4.7 Typical workflow

1. Acquire and select a standard waveform (SWAVE subsystem, §6).
2. Enable the comparator (`COMP ON`) and the comparison method(s) appropriate to the test objective (§4.2–4.5; see Operation Manual for guidance on selecting a method per application).
3. Set each enabled method's comparison window and allowed deviation.
4. Trigger a measurement (TRIGger subsystem, §5.1) and retrieve the result with `FETCh:CRESult?`.

SECTION 5

TRIGger, STATistic & WADJust Subsystem

5.1 TRIGger subsystem

Command	Details
`TRIG[:IMMediate]`	Trigger one device-under-test measurement. Valid only on the Measurement Display page; ignored on other pages and while a test is already in progress. Used only for device-under-test measurements — to trigger a standard-waveform acquisition, use `SWAVE:TRIGger` (§6). Example: `WrtCmd("TRIG")`.
`TRIG:SOUR?` `TRIGger:SOURce <MAN|EXTernal|INTernal|BUS>`	Trigger source. `MAN`: front-panel START key or foot switch; `EXTernal`: HANDLER interface; `INTernal`: instrument free-runs once started by the front-panel START key; `BUS`: remote interface (`TRIG` or `*TRG?`). Query returns `MAN`, `EXTERNAL`, `INTERNAL` or `BUS`. Example: `WrtCmd("TRIG:SOUR BUS")`.

5.2 STATistic subsystem

Controls the pass/fail statistics counter.

Command	Details
`STAT[:STATe]?` `STATistic[:STATe] <ON|OFF>`	Statistics function on/off. Query returns `ON` or `OFF`. Example: `WrtCmd("STAT ON")`.
`STATistic:CLEAr`	Clear the accumulated statistics data. No query form.
`STATistic:SAVE`	Save the accumulated statistics data to a file. No query form.

The statistics result values themselves are retrieved with `FETCh:STATistic?` (§7.1).

5.3 WADJust subsystem

Controls the time-base zoom and left/right scroll of the displayed waveform.

Command	Details
`WADJust[:MOVE] <RIGHt|LEFt>`	Scroll the displayed waveform left or right (only meaningful once the waveform has been zoomed in). No query form. Example: `WrtCmd("WADJust:MOVE RIGHt")`.
`WADJust[:EXTend]?` `WADJust[:EXTend] <value>`	Waveform time-base zoom factor. `value`: NR1, or `min` for the fully zoomed-out view showing the complete waveform. Query returns `<NR1>`. Example: `WrtCmd("WADJust:EXTend min")`.

SECTION 6

SWAVE (Standard WAVE) Subsystem

Controls acquisition and selection of the standard (reference) waveform that the comparator methods measure against: sampling mode, triggering a standard-waveform acquisition, waveform selection, and loading waveform or adjustment-voltage data.

Command	Details
`SWAVE:SMODe?` `SWAVE:SMODe <SCYCLe|OCYCLe|OSAMPle>`	Standard-waveform sampling mode. `SCYCLe`: continuous (repeating) acquisition cycle; `OCYCLe`: single acquisition cycle; `OSAMPle`: single-point sampling. Query returns `SEQ CYCLE`, `ONE CYCLE` or `ONE SAMPLE`. Example: `WrtCmd("SWAVE:SMODe OSAMPle")`.
`SWAVE:TRIGger[:IMMediate]`	Trigger one standard-waveform acquisition. Valid only on the Measurement Display page and only when the trigger source is `BUS`; ignored on other pages, with any other trigger source, and while a test is in progress. Example: `WrtCmd("SWAVE:TRIG")`.
`SWAVE:CHOose`	Select (accept) the currently acquired standard waveform as the active reference. No query form. Example: `WrtCmd("SWAVE:CHO")`.
`SWAVE:LOAD <data>`	Load standard-waveform data supplied by the host. `data`: the waveform data block, in the format described in §7.3. No query form. Example: `WrtCmd("SWAVE:LOAD 78F6...98")`.
`SWAVE:LOADADJVOLT <value>`	Load an adjustment voltage associated with the standard waveform. No query form. Example: `WrtCmd("SWAVE:LOADADJVOLT 1350V")`.
`SWAVE:ADJVOLT?`	Query the adjustment voltage. Returns `<NR1><NL^END>`.

> **NOTE**
> `SWAVE:TRIGger`, `SWAVE:LOAD`, `SWAVE:LOADADJVOLT` and `SWAVE:ADJVOLT?` are valid only on the Measurement Display page and are ignored elsewhere. `SWAVE:ADJVOLT?` additionally requires the sampling mode to be `SCYCLe` or `OCYCLe`; in `OCYCLe` mode it is valid only once the acquisition has completed and the instrument is in the selection state.

SECTION 7

FETCh? Subsystem

Retrieves waveform data, comparator results, statistics data, and the peak-voltage, frequency and time measurement results from the most recent test.

7.1 Measurement & statistics results

Command	Details
`FETC:VOLT?`	Peak voltage of the most recent measurement. Returns `<NR1><NL^END>`, in volts (V).
`FETC:FREQ?`	Frequency result within the range set by `MEASure:FREQuency` (§8.1). Returns `<NR3><NL^END>`, in hertz (Hz). Returns `9.9E37` if the configured range has zero width (start equal to end).
`FETC:TIME?`	Time result within the range set by `MEASure:TIME` (§8.1). Returns `<NR3><NL^END>`, in seconds (s).
`FETC:STATistic?`	Current statistics-counter data. Returns ten comma-separated NR1 values: total tests and total passes, then (total tests, passes) for each of the area, area-difference, corona and phase-difference comparison methods, in that order. The pass rate for each method, and overall, can be calculated from these counts.

7.2 Comparator result

Command	Details
`FETC:CRESult?`	Comparator result of the most recent device-under-test measurement. Three possible return formats: (1) if the comparator is off, or on with no comparison method enabled, returns `<NR1><NL^END>` with NR1 = 2; (2) if no measurement has yet been made, returns `<NR1><NL^END>` with NR1 = 3; (3) otherwise, once the comparator is on with at least one method enabled and a measurement has completed, returns `<NR1,NR3,NR3,NR1,NR3><NL^END>` — the first value is the overall result (1 = pass, 0 = fail), followed by the four per-method results in the order area, area-difference, corona, phase-difference. A method that is not enabled returns `9.9E37` (area, area-difference, phase-difference) or `9999` (corona) in its position.

7.3 Waveform data

Command	Details
`FETC:SWAVE?`	Output the currently acquired/selected standard waveform data (6000 points, format below). Example: `WrtCmd("FETC:SWAVE?")`.
`FETC:TWAVE?`	Output the most recent test (device-under-test) waveform data (6000 points, format below). Example: `WrtCmd("FETC:TWAVE?")`.
`FETC:CWAVE?`	Output the currently displayed corona waveform data.

If neither a new nor a completed measurement's data is yet available when `FETC:SWAVE?` or `FETC:TWAVE?` is queried, the response is deferred until the next measurement completes.

Waveform data transfer format

Waveform data is transmitted as an ASCII string. The standard waveform is transmitted as 6000 points; the test (measured) waveform is transmitted as 1200 points:

```
<DATA[1][0]><DATA[1][1]><DATA[2][0]><DATA[2][1]>…<DATA[n][0]><DATA[n][1]><NL^END>
```

Each waveform data point is transmitted as two characters, representing the high and low nibble (in that order) of the point's hexadecimal value (point range: 0–255). The host should convert these as needed (e.g. to decimal) before using them to render the waveform. If the requested waveform data does not exist, the instrument returns only the terminator `<NL^END>`.

> **WARNING**
>
> After sending a waveform-data query, the host must be ready to receive immediately, to avoid data loss.

SECTION 8

MEASure, ABORt & MMEMory Subsystem

8.1 MEASure subsystem

Sets the measurement range used for the voltage, frequency and time results retrieved via `FETCh` (§7.1). Frequency and time share the same underlying range, so changing one changes the other.

Command	Details
`MEAS:VOLT?` `MEASure:VOLTage <upper,lower>`	Voltage measurement range. `upper`/`lower`: NR1, 1–248, no suffix; `lower` must not be less than `upper`. Example: `WrtCmd("MEAS:VOLT 1,199")`.
`MEAS:FREQ?` `MEASure:FREQuency <start,end>`	Frequency measurement range. `start`/`end`: NR1, 0–598, no suffix; `end` must not be less than `start`. Example: `WrtCmd("MEAS:FREQ 100,200")`.
`MEAS:TIME?` `MEASure:TIME <start,end>`	Time measurement range. `start`/`end`: NR1, 0–598, no suffix; `end` must not be less than `start`. Example: `WrtCmd("MEAS:TIME 100,200")`.

8.2 ABORt subsystem

Command	Details
`ABORt`	Abort the measurement currently in progress. No query form. Example: `WrtCmd("ABOR")`.

8.3 MMEMory subsystem

Saves, loads and deletes complete instrument-setting files held in the instrument's internal file storage (100 record slots). A loaded file also restores the display page that was active when it was saved.

Command	Details
`MMEMory:LOAD:STATe <record>`	Load a previously saved settings file. `record`: NR1, 1–100. Example: `WrtCmd("MMEM:LOAD:STAT 1")`. No query form.
`MMEMory:STORe:STATe` `<record,"name">`	Save the instrument's current settings to a file. `record`: NR1, 1–100. `"name"`: file name, up to 10 ASCII characters, enclosed in quotation marks; if omitted, the instrument assigns the default name `<Unnamed>`. A name sent over the remote interface is taken literally (quoted), so characters not enterable from the front panel (certain symbols, lower-case letters) can be used. Example: `WrtCmd('MMEM:STOR:STAT 1,"#TH2883*"')`. No query form.
`MMEMory:DELete:STATe <record>`	Delete a stored settings file. `record`: NR1, 1–100. Example: `WrtCmd("MMEM:DEL:STAT 1")`. No query form. Deletion is not confirmed by the instrument before it occurs.

If the referenced file record does not exist (`LOAD` / `DELete`), or the record number is outside 1–100 (any of the three commands), the instrument reports `File not exist!` or `Out of file range!` respectively (§9).

SECTION 9

Error Messages

The instrument parses and executes each command string as it is received. If an error is encountered, the instrument displays an error message (in the LCD's message area) and stops parsing the remainder of that command string; content after the error is discarded. If the message indicates a command or trigger was merely ignored, the instrument continues normal parsing of subsequent command strings.

Error message	Meaning
`Unknown message!`	Unrecognized command. Example: `TRG` should be `TRIG`; `DISP:PAG MEAS` should be `DISP:PAGE MEAS`.
`Data out of range!`	Parameter value outside the valid range. Example: a `COMParator:DIFFzone:RANGe <start,end>` where `start > end`.
`Error parameter!`	Unrecognized parameter for the given command. Example: `TRIG:SOUR INTER` — `INTER` is not a supported trigger mode.
`Error unit suffix!`	Unit suffix not valid for that parameter. Example: `IVOLT:VOLT 200us` — `us` is not a supported voltage unit.
`Data too long!`	Parameter exceeds its maximum length. Example: a file name over 12 characters, or a numeric parameter over 10 characters.
`File not exist!`	The referenced file record does not exist (`MMEMory:LOAD/DELete`).
`Out of file range!`	File record number outside its valid range (1–100). Example: `WrtCmd('MMEM:STOR:STAT 105,"#TH2883*"')` — 105 exceeds the maximum of 100.
`Trigger ignores!`	A trigger was ignored, typically because a test was already in progress.
`Command ignores!`	A command was ignored because it is not valid in the instrument's current state. Example: `DISP:PAGE MSET` sent while a test is in progress.
`File already exist!`	The file name given to `MMEMory:STORe:STATe` is already in use.
`Error syntax!`	Malformed command syntax. Example: `WrtCmd('MMEM:STOR:STAT 105,"#TH2883*"')` with the file name not enclosed in quotation marks.

CAUTION — VERIFY BEFORE PRODUCTION USE

Users should verify command behaviour against the firmware version installed on their instrument. Please report any discrepancy to Techmize.

Contact, support & related documents

This command reference is the companion to the *TH2883 Series Operation Manual* (front-panel operation, remote-interface wiring, HANDLER interface) and the *TH2883 Series Datasheet* (specifications). For questions about any command's exact behaviour, or to report a discrepancy against your instrument's firmware, contact Techmize using the details below.

Techmize Electronic GmbH

Office 125–126, Regus
Carl-Zeiss-Ring 15a, 85737 Ismaning,
Germany
An international subsidiary of Changzhou
Tonghui Electronic Co., Ltd. (China)

Contact

Mobile: +49 171 4981898
Email: dx.han@tonghui.com.cn
Web: en.tonghui.com.cn

Registration

Managing Director: Haohua Zhao
Register Court: Amtsgericht München
HRB 312530 · VAT ID DE463170824