TECHMIZE by TONGHUI

CR

SCPI & MODBUS COMMAND REFERENCE

# TH2683

## Insulation Resistance / Leakage Current Meter — TH2683A / TH2683B

Companion document to the TH2683 Series Operation Manual. Full remote-control command set for both the SCPI (RS232C/RS485/USBTMC/USBVCOM) and MODBUS (RS485) protocols, organized by subsystem.

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

# TH2683 SCPI & MODBUS Command Reference

## Insulation Resistance / Leakage Current Meter

This document lists the full remote-control command set for the TH2683A/TH2683B. It is the companion reference to the *TH2683 Series Operation Manual*, which covers front-panel operation, the physical remote interfaces (RS232C/RS485/USBTMC/USBVCOM/HANDLER) and general operating procedures. SCPI is available over all four remote interfaces; MODBUS is available only over RS485 (Operation Manual §6.1–6.2).


| Section | |
|---|---|
| **1. Conventions & Reference Tables** | 1 page |
| **2. IEEE 488.2 Common Commands & DISPlay Subsystem** | 1 page |
| **3. FUNCtion Subsystem** | 2 pages |
| **4. DISCharge & TRIGger Subsystem** | 1 page |
| **5. FETCh Subsystem & Measurement Workflow** | 1 page |
| **6. COMParator Subsystem** | 2 pages |
| **7. SYSTem Subsystem** | 1 page |
| **8. Mass MEMory Subsystem** | 1 page |
| **9. MODBUS Protocol** | 5 pages |


*SCPI commands conform to the SCPI/IEEE 488.2 standard; the capitalized portion of each keyword is the mandatory short form, the lower-case portion optional (e.g.* `FUNCtion:RANGe` *may be sent as* `FUNC:RANG`*). The instrument does not distinguish upper/lower case on received commands. 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. §9.4 cross-references each MODBUS register to its SCPI equivalent.*

SECTION 1

# Conventions & Reference Tables

## 1.1 SCPI syntax conventions

- Commands are ASCII, hierarchical (tree-structured); related commands share a root, forming a subsystem (e.g. all `FUNC:...` commands).
- Within a keyword, the capitalized portion is the mandatory short form; the lower-case portion is optional long-form text (e.g. `FUNCtion:CZERo` may be sent as `FUNC:CZER` or in full). The instrument does not distinguish upper/lower case on received commands.
- A query form ends in `?` and returns the current value or result; the same root without `?`, followed by a parameter, sets the value or triggers the action.
- NR1 = integer (e.g. `123`); NR2 = fixed-point decimal (e.g. `12.3`); NR3 = scientific/floating-point (e.g. `12.3E+5`). A command accepting "NR1, NR2 or NR3" accepts any of the three formats for that value.
- NL = newline (decimal 10, hex 0x0A), used to terminate each command line sent over RS232C/RS485/USBVCOM. ^END denotes the IEEE 488 bus EOI (end-of-identify) signal appended to USBTMC query responses.
- ON and OFF may each also be sent as the equivalent integer: `1` (ASCII 49) for ON, `0` (ASCII 48) for OFF.

## 1.2 Remote interfaces & protocol selection

The TH2683A/B accepts SCPI commands over RS232C, RS485, USBTMC and USBVCOM. MODBUS is available only over RS485, as an alternative to SCPI on that interface (never simultaneously). RS232C: 9600–115200 baud (selectable), no parity, 8 data bits, 1 stop bit; SCPI commands are LF-terminated and limited to 2 KB per command string. RS485 adds a bus-address prefix to every command (§1.3) to support multi-drop addressing, and carries either SCPI or MODBUS depending on the instrument's configured protocol.

## 1.3 RS485 addressed command format

```
ADDRESS + @ + SCPI command
```

Where **ADDRESS** is the instrument's bus address (1–32; 0 is the broadcast/common address) and **@** separates the address from the SCPI command. This framing applies to RS485 only; RS232C, USBTMC and USBVCOM commands are sent without an address prefix.

## 1.4 TH2683A / TH2683B model difference

TH2683A and TH2683B share an identical command set and MODBUS register map. The only functional difference is the maximum DC test-output voltage reachable via `FUNCtion:OVOLtage` (§3.1) and MODBUS register 0x05: **TH2683A** accepts 1–1000 V, **TH2683B** accepts 1–500 V. All other commands, ranges and registers are identical between the two models.

SECTION 2

# IEEE 488.2 Common Commands & DISPlay Subsystem

## 2.1 IEEE 488.2 common commands

| Command | Details |
|---|---|
| `*RST` | Reset the instrument. Example: `WrtCmd("*RST")`. |
| `*TRG` | Trigger a measurement and send the result to the instrument's output buffer. The trigger source must first be set to bus (`TRIG:SOUR BUS`, §4.2) before `*TRG` is used. |
| `*IDN?` | Query the instrument identification. Returns three comma-separated fields: `<manufacturer>,<model>,<firmware><NL^END>` — manufacturer (Tonghui), model (e.g. `TH2683A` or `TH2683B`) and firmware version (e.g. `Version1.0.0`). Example: `WrtCmd("*IDN?")`. |

## 2.2 DISPlay subsystem

Sets or queries the instrument's active display page.

| Command | Details |
|---|---|
| `DISP:PAGE?`<br>`DISP:PAGE <page>` | Query/switch the active display page. `page`: `MEASurement` (Measurement Display), `MSETup` (Measurement Setup), `LTABle` (Pickout/Bin-Table Setup), `SYSTem` (System Setup), `FLISt` (internal file list). The query returns the same token as a string. Example: `WrtCmd("DISP:PAGE MEAS")` → switches to the Measurement Display page. |

> **NOTE**
> It is recommended to issue all setting-changing commands (DISPlay, FUNCtion, COMParator, SYSTem, Mass MEMory) while the instrument is in the discharged state ( `DISCharge[:GO]` , §4.1); changing settings during a test or while charged may produce incorrect results.

SECTION 3

# FUNCtion Subsystem

Controls the Measurement Setup page: output voltage, measurement mode and speed, charge/wait/measurement/discharge times, averaging count, leakage-current range and auto-range, measurement-display on/off, input impedance (MIRE) and open-circuit zero adjustment.

## 3.1 Output voltage & zero adjustment

| Command | Details |
|---|---|
| `FUNC:OVOL?`<br>`FUNC:OVOLtage <value>` | DC test output voltage. `value`: any real value within the instrument's output-voltage range — **TH2683A: 1–1000 V**, **TH2683B: 1–500 V** (§1.4). Query returns the current value. Example: `WrtCmd("FUNCtion:OVOL 12.50")` → sets 12.50 V. |
| `FUNC:CZER:ON`<br>`FUNC:CZER OFF`<br>`FUNC:CZER?` | `ON` (or `1`) performs an open-circuit zero adjustment; if it succeeds, the zero baseline used for measurement is taken from this adjustment, otherwise the factory zero baseline is used. `OFF` (or `0`) cancels a successful user zero adjustment and reverts to the factory zero baseline. The query (`FUNCtion:CZERo?`) returns the result of the last zero-adjustment attempt as `SUCCess` or `FAILed`. Example: `WrtCmd(":FUNC:CZER:ON")`. |

## 3.2 Measurement mode, speed & display

| Command | Details |
|---|---|
| `FUNC:MMOD?`<br>`FUNC:MMODe <SINGle\|CONTinuous>` | Measurement mode: `SINGle` or `CONTinuous`. Query returns the same token. Example: `WrtCmd("FUNCtion:MMOD SINGle")`. |
| `FUNC:MSP?`<br>`FUNC:MSPeed <FAST\|SLOW>` | Measurement speed: `FAST` or `SLOW`. Query returns the same token. Example: `WrtCmd("FUNCtion:MSP SLOW")`. |
| `FUNC:CCH?`<br>`FUNC:CCHeck <ON\|OFF>` | Contact-check function on/off. Query returns `ON` or `OFF`. Example: `WrtCmd("FUNC:CCH:ON")`. |
| `FUNC:MDIS?`<br>`FUNC:MDISplay <ON\|OFF>` | Measurement-result display on/off (Measurement Display page). Query returns `ON` or `OFF`. Example: `WrtCmd("FUNC:MDIS:ON")`. |

## 3.3 Timing parameters

All four timing parameters accept a real value in the range 0–999.0 s, resolved to 0.1 s.

| Command | Details |
|---|---|
| `FUNC:CTIM?`<br>`FUNC:CTIMe <value>` | Charge time, 0–999.0 s. Example: `WrtCmd("FUNCtion:CTIM 12.5")` → 12.5 s. |
| `FUNC:WTIM?`<br>`FUNC:WTIMe <value>` | Wait time, 0–999.0 s. Example: `WrtCmd("FUNCtion:WTIMe 12.5")`. |
| `FUNC:MTIM?`<br>`FUNC:MTIMe <value>` | Measurement time, 0–999.0 s. Example: `WrtCmd("FUNCtion:MTIMe 12.5")`. |
| `FUNC:DTIM?`<br>`FUNC:DTIMe <value>` | Timed-discharge time, 0–999.0 s. Example: `WrtCmd("FUNCtion:DTIMe 12.5")`. |

## 3.4 Averaging, range & input impedance

| Command | Details |
|---|---|
| `FUNC:AVER?`<br>`FUNC:AVERage <value>` | Averaging count, integer 1–999. Example: `WrtCmd("FUNCtion:AVERage12")` → 12 readings averaged. |
| `FUNC:RANG?`<br>`FUNC:RANGe <value>` | Leakage-current range (manual selection). `value`: `1mA`, `100uA`, `10uA`, `1uA`, `100nA`, `10nA`. Example: `WrtCmd("FUNC:RANG 1mA")`. |
| `FUNC:RANG:AUTO?`<br>`FUNC:RANG:AUTO <ON\|OFF>` | Auto-ranging on/off. Query returns `ON` or `OFF`. Example: `WrtCmd("FUNC:RANG:AUTO ON")`. |
| `FUNC:MIRE?`<br>`FUNC:MIREsistance <AUTO\|10k\|1M>` | Input impedance for the current leakage-current range: `AUTO`, `10k` (10 kΩ) or `1M` (1 MΩ). Query returns `auto`, `10k` or `1M`. |

SECTION 4

# DISCharge & TRIGger Subsystem

## 4.1 DISCharge subsystem

| Command | Details |
|---|---|
| `DISCharge[:GO]` | Starts discharging the instrument/device under test. No query form. |

## 4.2 TRIGger subsystem

Sets the trigger source and issues an immediate (software) trigger.

| Command | Details |
|---|---|
| `TRIG[:IMMediate]` | Trigger one measurement immediately. Example: `WrtCmd("TRIG")`. |
| `TRIG:SOUR?`<br>`TRIG:SOURce <EXTernal\|BUS\|HOLD>` | Trigger source: `EXTernal` — triggered by the HANDLER interface; `BUS` — triggered by the RS232/RS485 or USB (DEVICE) interface, via `TRIG` or `*TRG`; `HOLD` — triggered by the front-panel TEST key. Query returns `EXT`, `BUS` or `HOLD`. Example: `WrtCmd("TRIG:SOUR BUS")`. |

> **NOTE**
> `*TRG` (§2.1) requires the trigger source to already be set to `BUS`; sending `*TRG` with any other trigger source has no defined effect.

SECTION 5

# FETCh Subsystem & Measurement Workflow

## 5.1 FETCh commands

| Command | Details |
|---|---|
| `FETC[:IMP]?` | Send the last measurement result to the output buffer.<br><br>**With pickout (COMParator) ON:** returns `<insulation resistance>,<leakage current>,<bin>,<comparison result>,<out-of-limit>`. `bin`: comparison result — `0` pass on bin 1, `1` pass on bin 2, `2` pass on bin 3, `3` fail on all three bins. `out-of-limit`: `0` under-range, `1` within range, `2` over-range.<br><br>**With pickout OFF:** returns `<insulation resistance>,<leakage current>,<out-of-limit>` (same out-of-limit coding).<br><br>Example: `WrtCmd("TRIG:SOUR BUS"); WrtCmd("TRIG"); WrtCmd("FETC?")`. |
| `FETC:AUTO?`<br>`FETC:AUTO <ON\|OFF>` | Automatic result return after each measurement (`ON`/`1`) or not (`OFF`/`0`). Example: `WrtCmd("FETC:AUTO ON")`. |
| `FETC:SMON:VOLT?` | Query the currently monitored voltage value. |

### 5.1.1 Out-of-limit range table (per current range)

The following ranges apply to the out-of-limit comparison when pickout is ON (range 6 has no lower-limit comparison):

| Range | Comparison range |
|---|---|
| Range 1 | 95 µA – 1.05 mA |
| Range 2 | 9.5 µA – 105 µA |
| Range 3 | 0.95 µA – 10.5 µA |
| Range 4 | 95 nA – 1.05 µA |
| Range 5 | 9.5 nA – 105 nA |
| Range 6 | – 10.5 nA (no lower limit) |

## 5.2 Measurement-result retrieval workflow

1. **No automatic return** — set `TRIG:SOUR BUS` (skip if already BUS), send `TRIG`, then `FETC?`. Repeat the trigger and fetch for each new reading.
2. **Automatic return** — set `TRIG:SOUR BUS` (skip if already BUS), set `FETC:AUTO ON` (skip if already ON), then send `TRIG`; the result is returned automatically. Repeat `TRIG` for each new reading.

SECTION 6

# COMParator Subsystem

Controls the 3-bin pickout (sorting) function on the Pickout/Bin-Table Setup page: enable, sorted item, per-bin limits, beeper, bin-number display, limit-value display, output result type and pulse width, and number of active bins.

## 6.1 Enable & sorted item

| Command | Details |
|---|---|
| `COMP:FUNC?`<br>`COMParator:FUNCtion <ON\|OFF>` | Pickout function on/off. Query returns `<NR1>` (0/1). Example: `WrtCmd("COMP:FUNC ON")`. |
| `COMP:ITEM?`<br>`COMParator:ITEM <CURRent\|RESistance>` | Sorted item: leakage current or insulation resistance. Query returns `CURRent` or `RESistance`. Example: `WrtCmd("COMP:ITEM CURR")`. |

## 6.2 Per-bin limits

| Command | Details |
|---|---|
| `COMP:CURR:BIN<n> <low>,<high>`<br>`COMP:CURR:BIN<n>?` | Current-item upper/lower limit for bin `n` (NR1, 1–3). `low`/`high`: NR1, NR2 or NR3; unit defaults to amperes (A). Query returns `<NR3>,<NR3>`. Example: `WrtCmd("COMP:CURR:BIN1 12.00n,50.00n")`. |
| `COMP:RES:BIN<n> <low>,<high>`<br>`COMP:RES:BIN<n>?` | Resistance-item upper/lower limit for bin `n` (NR1, 1–3). `low`/`high`: NR1, NR2 or NR3; unit defaults to ohms (Ω). Query returns `<NR3>,<NR3>`. |

## 6.3 Beeper, bin display & limit display

| Command | Details |
|---|---|
| `COMP:BEEP?`<br>`COMP:BEEPer`<br>`<BONe\|BTWo\|BTHRee\|NG\|OFF>` | Pickout-result beeper: `BONe` beeps on bin 1 pass, `BTWo` on bin 2 pass, `BTHRee` on bin 3 pass, `NG` beeps when all three bins fail, `OFF` disables the beeper. Query returns the same token. Example: `WrtCmd("COMP:BDIS ON")` (bin display, see next row). |
| `COMP:BDIS?`<br>`COMP:BDISplay <ON\|OFF>` | Bin-number display on/off. Query returns `<NR1>`. Example: `WrtCmd("COMP:BDIS ON")`. |
| `COMP:PLIM?`<br>`COMP:PLIMitvalue <ON\|OFF>` | Pickout limit-value display on/off. Query returns `<NR1>`. Example: `WrtCmd("COMP:PLIM ON")`. |

## 6.4 Output result type, pulse width & bin count

| Command | Details |
|---|---|
| `COMP:ORES?`<br>`COMParator:ORESult <LEVel\|PULSe>` | Pickout output-result type on the HANDLER interface: `LEVel` (level) or `PULSe` (pulse). Query returns the same token. Example: `WrtCmd("COMP:ORES LEV")`. |
| `COMP:PWID?`<br>`COMParator:PWIDth <value>` | HANDLER pickout output pulse width, integer 1–25 ms (applies when `COMP:ORES` is `PULSe`). Example: `WrtCmd("COMParator:PWID 10")` → 10 ms. Note that `:PWIDth` belongs to the `COMParator` subsystem, not `FUNCtion`. |
| `COMP:PBNO?`<br>`COMParator:PBNO <OBIN\|TBIN\|THBIN>` | Number of active pickout bins: `OBIN` (1 bin), `TBIN` (2 bins), `THBIN` (3 bins). Query returns `OBIN`, `TBINs` or `THBins`. |

SECTION 7

# SYSTem Subsystem

System-level settings: key/touch beeper, firmware version query, instrument status query, HANDLER interface power source and RS485 bus address.

| Command | Details |
|---|---|
| `SYST:BEEP?`<br>`SYST:BEEP <ON\|OFF>` | System beeper on/off. Query returns `0` or `1`. Example: `WrtCmd("SYST:BEEP 0")` → beeper off. |
| `SYST:VERS?` | Query the instrument's firmware version, e.g. `Ver 1.00`. |
| `SYST:STAT?` | Query the instrument's current status. Returns `DISCharging` (discharging), `TESTing` (test in progress) or `test complete` (test finished). |
| `SYST:HPOW?`<br>`SYST:HPOWer <INTernal\|EXTernal>` | HANDLER interface power source. Query returns `INTernal` (internal supply) or `EXTernal` (external supply). Example: `WrtCmd("SYSTem:HPOWer INTernal")`. |
| `SYST:BADD?`<br>`SYST:BADDr <value>` | RS485/RS232 bus address, integer 1–32. Query returns the current address. Example: `WrtCmd("SYST:BADDr1")` → sets bus address 1. |

SECTION 8

# Mass MEMory Subsystem

Saves and loads complete instrument-setting files to/from internal storage (20 record slots).

| Command | Details |
|---|---|
| `MMEMory:LOAD:STATe <value>` | Load a previously saved settings file. `value`: NR1 file-record number, 1–20. Example: `WrtCmd("MMEM:LOAD:STAT 1")`. |
| `MMEMory:STORe:STATe <value>,<name>` | Save the instrument's current settings to a file. `value`: NR1 file-record number, 1–20. `name`: file name, no `.STA` extension needed, max. 15 characters. Example: `WrtCmd("MMEM:STOR:STAT 1,ABC")`. |

> **NOTE**
> No query form is documented for either `MMEMory` command.

SECTION 9

# MODBUS Protocol

The TH2683A/B supports MODBUS as an alternative to SCPI, over the RS485 interface only (§1.2). Every setting reachable by a SCPI command in Sections 2–8 is also reachable as a MODBUS register; §9.4 gives the full address cross-reference.

## 9.1 Write frame (function code 0x10)

**Send (host → instrument):**

| Byte 0 | 1 | 2–3 | 4–5 | 6 | 7…7+n−1 | last−1…last |
|---|---|---|---|---|---|---|
| **Instrument address** | Function code `0x10` | Register address (hi,lo) | Register count (hi,lo) | Byte count | Data byte 1…n | CRC (lo,hi) |

**Response (instrument → host):**

| Byte 0 | 1 | 2–3 | 4–5 | 6–7 |
|---|---|---|---|---|
| **Instrument address** | Function code `0x10` | Register address (hi,lo) | Register count (hi,lo) | CRC (lo,hi) |

*Worked example — set output voltage to 2.5 V (register 0x05, a 4-byte float value needing 2 registers):*

```
Send: 08 10 00 05 00 02 04 40 20 00 00 09 06
Response: 08 10 00 05 00 02 xx xx
```

## 9.2 Read frame (function code 0x03)

**Send (host → instrument):**

| Byte 0 | 1 | 2–3 | 4–5 | 6–7 |
|---|---|---|---|---|
| **Instrument address** | Function code `0x03` | Register address (hi,lo) | Register count (hi,lo) | CRC (lo,hi) |

**Response (instrument → host):**

| Byte 0 | 1 | 2 | 3…3+n−1 | last−1…last |
|---|---|---|---|---|
| **Instrument address** | Function code `0x03` | Byte count | Data byte 1…n | CRC (lo,hi) |

*Worked example — query insulation resistance, leakage current, etc. (register 0x1E; returns 2 float values + 1 U16 value = 10 data bytes, 5 registers):*

```
Send: 08 03 00 1E 00 05 E5 56
Response: 08 03 0A 56 B5 E6 21 42 C8 00 00 00 02 00 E5
```

(In this example the insulation-resistance value decodes from data bytes `56 B5 E6 21`.)

## 9.3 Field reference

- **Instrument address** — the instrument's bus address, set via `SYST:BADDr` (§7) or the System Setup page, range 1–32.
- **Register address** — the storage address of the data inside the instrument (addresses listed in §9.4).
- **Register count** — number of 16-bit registers involved: 1 register per 2-byte (U16) value, 2 registers per 4-byte (float) value.
- **Byte count** — total data bytes transferred.
- **Data bytes** — a float value is IEEE-754 single precision; a U16 value is a 16-bit integer.
- **CRC** — a 2-byte CRC check code, transmitted low byte then high byte.

## 9.4 Register map

Register addresses are given in hex, as used in the frame's address field. Read registers use function code `0x03`; write registers use function code `0x10`. Where a SCPI equivalent exists, see that command's own entry (Sections 2–8) for value ranges and units.

### 9.4.1 Read registers (function code 0x03)

| Address | Function | Return value | Equivalent SCPI |
|---|---|---|---|
| **0x01** | System beeper state | 0: off · 1: on | `SYST:BEEP?` |
| **0x02** | Firmware version | String | `SYST:VERS?` |
| **0x03** | Instrument current status | 0: testing · 1: discharging | `SYST:STAT?` |
| **0x04** | HANDLER interface power type | 0: internal · 1: external | `SYST:HPOW?` |
| **0x05** | Current display page | 0–9 | `DISP:PAGE?` |
| **0x06** | Zero-adjustment result | 0: failed · 1: succeeded | `FUNC:CZER?` |
| **0x07** | Current output-voltage setting | Float | `FUNC:OVOL?` |
| **0x08** | Measurement mode | 0: single · 1: continuous | `FUNC:MMOD?` |
| **0x09** | Measurement speed | 0: fast · 1: slow | `FUNC:MSP?` |
| **0x0A** | Contact-check state | 0: on · 1: off | `FUNC:CCH?` |
| **0x0B** | Charge time | Float | `FUNC:CTIM?` |
| **0x0C** | Wait time | Float | `FUNC:WTIM?` |
| **0x0D** | Measurement (test) time | Float | `FUNC:MTIM?` |
| **0x0E** | Timed discharge time | Float | `FUNC:DTIM?` |
| **0x0F** | Averaging count | U16 | `FUNC:AVER?` |
| **0x10** | Range setting state | 0: auto · 1: locked | `FUNC:RANG:AUTO?` |

| Address | Function | Return value | Equivalent SCPI |
|---|---|---|---|
| **0x11** | Measurement-display control variable | 0: on · 1: off | `FUNC:MDIS?` |
| **0x12** | Input impedance (MIRE) | 0: Auto · 1: 10k · 2: 1M | `FUNC:MIRE?` |
| **0x13** | Trigger mode | 0: INT · 1: EXT · 2: BUS | `TRIG:SOUR?` |
| **0x14** | Pickout function state | 0: on · 1: off | `COMP:FUNC?` |
| **0x15** | Pickout item | 0: current · 1: resistance | `COMP:ITEM?` |
| **0x16** | 3-bin current upper/lower limits | 6 floats | `COMP:CURR:BIN<n>?` (bins 1–3) |
| **0x17** | 3-bin resistance upper/lower limits | 6 floats | `COMP:RES:BIN<n>?` (bins 1–3) |
| **0x18** | Beeper-to-pickout-result mapping | 1 / 2 / 3 / 4 (coded) | `COMP:BEEP?` |
| **0x19** | Bin-number display | 0: on · 1: off | `COMP:BDIS?` |
| **0x1A** | Pickout limit-value display | 0: on · 1: off | `COMP:PLIM?` |
| **0x1B** | Output result type | 0: level · 1: pulse | `COMP:ORES?` |
| **0x1C** | Pulse width | U16 | `COMP:PWID?` |
| **0x1D** | Active pickout bin count | 1 / 2 / 3 (coded) | `COMP:PBNO?` |
| **0x1E** | Last-test insulation resistance, leakage current, etc. | Pickout ON: 5 params (2 float + 3 U16); OFF: 3 params | `FETC[:IMP]?` |
| **0x1F** | Monitored (detection) voltage | Float | `FETC:SMON:VOLT?` |
| **0x20** | Instrument basic information | String | `*IDN?` |

### 9.4.2 Write registers (function code 0x10)

| Address | Function | Parameter(s) | Equivalent SCPI |
|---|---|---|---|
| **0x01** | Set system beeper | 1 param: 0 off · 1 on | `SYST:BEEP` |
| **0x02** | Set HANDLER interface power type | 1 param: 0 internal · 1 external | `SYST:HPOW` |
| **0x03** | Set current display page | 1 param: 0–9 | `DISP:PAGE` |
| **0x04** | Zero-adjust / cancel open-circuit zero | 1 param: 1 = zero-adjust | `FUNC:CZER` |
| **0x05** | Set output voltage | 1 param: float | `FUNC:OVOL` |
| **0x06** | Set measurement mode | 1 param: 0 single · 1 continuous | `FUNC:MMOD` |
| **0x07** | Set measurement speed | 1 param: 0 fast · 1 slow | `FUNC:MSP` |
| **0x08** | Set contact-check state | 1 param: 0 on · 1 off | `FUNC:CCH` |
| **0x09** | Set charge time | 1 param: float | `FUNC:CTIM` |
| **0x0A** | Set wait time | 1 param: float | `FUNC:WTIM` |
| **0x0B** | Set test time | 1 param: float | `FUNC:MTIM` |
| **0x0C** | Set timed discharge time | 1 param: float | `FUNC:DTIM` |
| **0x0D** | Set averaging count | 1 param: int | `FUNC:AVER` |
| **0x0E** | Set range mode | 1 param: 0 auto · 1 locked | `FUNC:RANG:AUTO` |
| **0x0F** | Locked range selection | 1 param: 0 1mA · 1 100µA · 2 10µA · 3 1µA · 4 100nA · 5 10nA | `FUNC:RANG` |
| **0x10** | Set measurement-display control variable | 1 param: 0 on · 1 off | `FUNC:MDIS` |

| Address | Function | Parameter(s) | Equivalent SCPI |
|---|---|---|---|
| **0x11** | Set input impedance (MIRE) | 1 param: 0 Auto · 1 10k · 2 1M | `FUNC:MIRE` |
| **0x12** | Discharge | 1 param: 1 = discharge | `DISCharge[:GO]` |
| **0x13** | Trigger | 1 param: 1 trigger · 0 no effect | `TRIG[:IMMediate]` |
| **0x14** | Set trigger mode | 1 param: 0 INT · 1 EXT · 2 BUS | `TRIG:SOUR` |
| **0x15** | Set pickout function state | 1 param: 0 on · 1 off | `COMP:FUNC` |
| **0x16** | Set pickout item | 1 param: 0 current · 1 resistance | `COMP:ITEM` |
| **0x17** | Set 3-bin current limits | 6 params: float | `COMP:CURR:BIN<n>` |
| **0x18** | Set 3-bin resistance limits | 6 params: float | `COMP:RES:BIN<n>` |
| **0x19** | Set beeper-to-pickout-result mapping | 1 param: coded 0/1/2/3/4 | `COMP:BEEP` |
| **0x1A** | Set bin-number display | 1 param: 0 on · 1 off | `COMP:BDIS` |
| **0x1B** | Set pickout limit-value state | 1 param: 0 on · 1 off | `COMP:PLIM` |
| **0x1C** | Set output result type | 1 param: 0 level · 1 pulse | `COMP:ORES` |
| **0x1D** | Set pulse width | 1 param: U16, 1–25 | `COMP:PWID` |
| **0x1E** | Set active pickout bin count | 1 param: coded 1/2/3 | `COMP:PBNO` |
| **0x1F** | Load settings file | 1 param: U16 | `MMEMory:LOAD:STATe` |
| **0x20** | Save current settings to file | 2 params: record no. (U16), file name (string) | `MMEMory:STORe:STATe` |
| **0x23** | Set automatic measurement-data return | 1 param: 0 off · 1 on | `FETC:AUTO` |

### 9.4.3 Error codes

| Error | Meaning |
|---|---|
| Error 1 | Instrument-address error |
| Error 2 | Function-code error |
| Error 3 | Checksum (CRC) error |
| Error 4 | Command (register) address error |

> **NOTE**
>
> Write register addresses 0x21 and 0x22 are reserved (unused) — the register map continues from 0x20 to 0x23, with no function assigned to 0x21/0x22.

## Contact, support & related documents

This command reference is the companion to the *TH2683 Series Operation Manual* (front-panel operation, remote-interface wiring, HANDLER interface) and the *TH2683 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