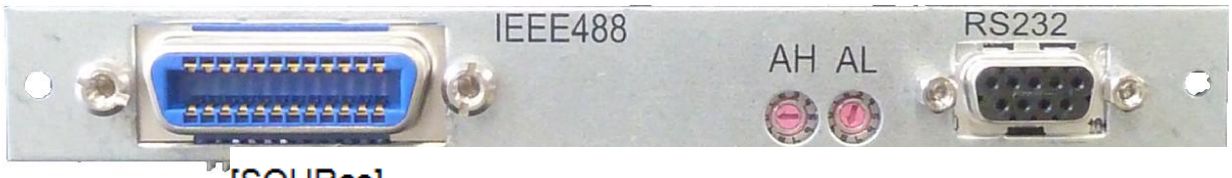
TC.IEEE (GPIB) Manual

Option



[SOURce]
:VOLTage
[:LEVel]
[:IMMediate]
[:AMPLitude]{<voltage>|MINimum|MAXimum}
:CURRent
[:LEVel]
[:IMMediate]
[:AMPLitude]{<current>|MINimum|MAXimum}

DO6130.0029 V02.62

Regatron AG
Kirchstrasse 11
CH-9400 Rorschach
Tel +41 71 846 67 67
Fax +41 71 846 67 77
www.regatron.ch
topcon@regatron.ch

Distributor-Aufkleber

The actual version of the TC.IEEE(GPIB) Manual can be found on the internet page www.regatron.com.

Please refer to the latest edition of our Conditions of Delivery.

Manufacturer

Manufacturer	
Regatron AG	Tel. +41 71 846 67 67
Kirchstrasse 11	Fax +41 71 846 67 77
9400 Rorschach	www.regatron.com
Switzerland	topcon@regatron.ch

Tab. 1

Manual

Overview	
Manual	**TC.IEEE (GPIB) Manual;** DO6130.0029 V02.62
Valid of versions starting from:	
TopCon Main firmware	Up to version V4.11.45
GPIB option Card firmware	Up to version V0.08

Tab. 2 Subject to technical changes.

TABLE OF CONTENT

1. Information

1.1. Safety notes

Before using the option TC.CANOPEN the operating manual of the TopCon power supply unit must be read. The safety notes in the operating manual must be observed and the necessary measures must be taken.

1.2. Used Pictograms and Signal Words

Important notes are marked with the following symbols throughout this operating manual:

Hazard and warning information	
Pictogram	**Meaning**
 **DANGER**	For an immediate hazard that will result in serious injuries or fatality.
 **Warning**	For an immediate hazard that can result in serious injuries or fatality.
 **CAUTION**	For a possibly hazardous situation that can result in serious injuries or fatality.
CAUTION	For a possibly hazardous situation that could result in damage to the product or another item in its surroundings.

Tab. 3 Basic hazard and warning information

Further warning and hazard information	
Pictogram	**Meaning**
	DANGER, WARNING or CAUTION due to electrical power

Tab. 4 Symbols included in the table can be used for more specific depiction of warning information from Tab. 3 "Basic hazard and warning information".

Instructions	
Pictogram	**Meaning**
	Important information

Tab. 5 Mandatory signs that are important for the operation of the device or the software

General notes	
Pictogram	**Meaning**
	Tip, for working efficiently with the device

Tab. 6 Additional information, so that you can find possibly important information quickly.

2. Specification and attributes

2.1. Scope

TCGPIB option allows for controlling TopCon units by an IEEE488 compatible bus.

The SCPI-compatible instruction set provides a fast and uncomplicated integration of TopCon units in a test and evaluation environment.

.

2.2. SCPI instruction set

The used SCPI instruction set refers to SCPI regulation Nr. 1999.0 and to Standard IEC 60488-2(E): 2004 and IEEE488-2(E):1992.

Note the following remarks:

- Maximum length of a <program message>(IEEE488.2) is 64 bytes
- Maximum length of a <program message unit>(IEEE488.2) is 64 bytes
- Maximum number of <program message unit>inside a<program message> is 8
- A maximum of 64 errors can be stored inside the Error/Event Queue
- All instructions are computed sequentially, no overlap

Extended programming

For further information about extended commands and the necessary address programming, refer to the Low-Level Protocol“-Manual.
The actual version can be found on the internet page www.regatron.com.

3. Configuration

3.1. Installation

Option TCGPIB is installed completely by the manufacturer. Later upgrade of a TopCon unit with TCGPIB is possible, in this case the unit has to be shipped to the manufacturer.

3.2. Set GPIB address

Prior to unit power-on the GPIP address has to be set by means of two rotary switches AH and AL on the back of the case..

Modul-ID = 16 * AH + AL	Value range: 1...30 A set value > 30 is corrected to 30, A set value = 0 is corrected to 1

Example 1:

- The address of unit shall be 25.

↳ Set AH to value 2,
set AL to value 5.

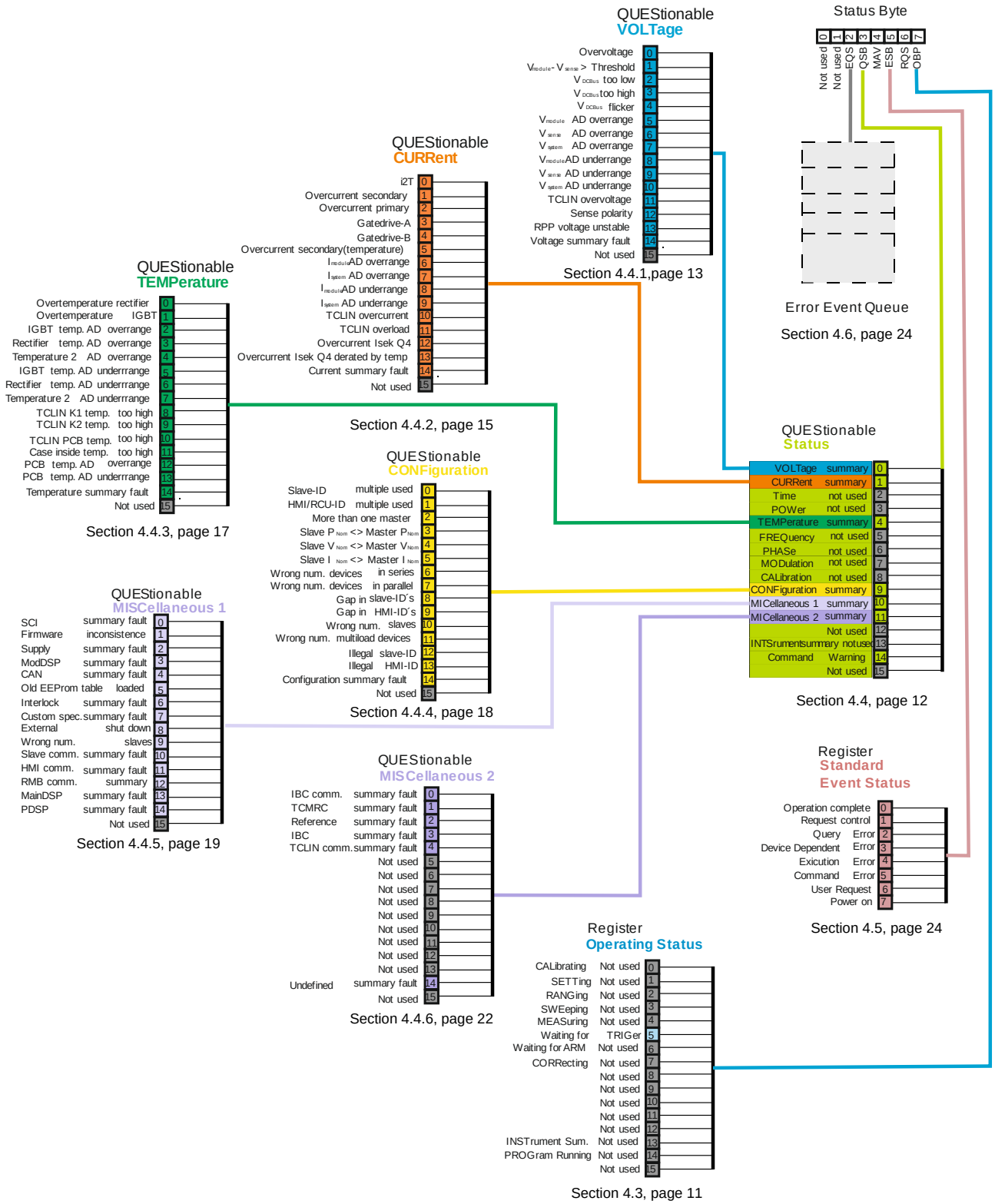
Example 2:

- The two address switches are at AH = 4 and AL = 7
(This is beyond the allowed range)

↳ Internally the value will be corrected to 30, so the unit is addressed as Nr 30.

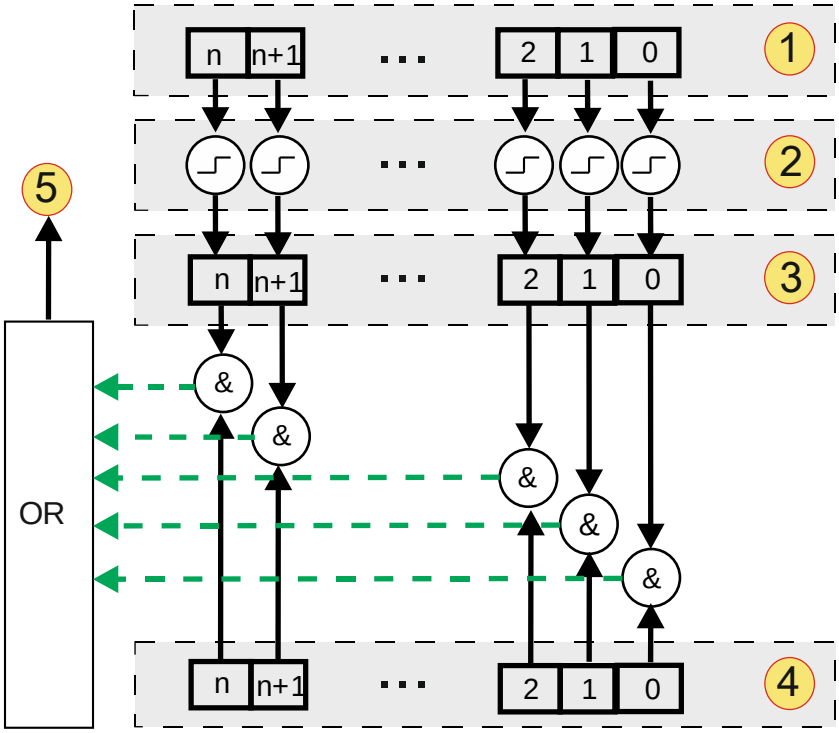
4. Status Register

4.1. Overview



4.2. Structure of Status Register

All status registers with the exception “Error Event Queue” and the “Status Byte” consist of a “condition register”**-1-**, an “event register” **-3-** an “enable register” **-4-** and a “transition filter” **-2-**. Their function is described below.



1	**“Condition Register”,** read only (r) A “Condition Register” reflects the state of the associated TopCon unit. The condition bits inside the register are set automatically. Reading of the register does not change the register content.
2	**“Transition Filter”** The “transition filter” determines, whether raising or falling slopes of “condition registers” are enabled to set respective bits inside a “event register”.
3	**“Event Register”** An “event register” stores events, which are related to bits in the “condition register”. Every change of the condition bits is stored in the “event register”. Note, that reading-out of the event register will clear the register itself. A “CLS” instruction clears all “event register”.
4	**“Event Enable Register”**, readable and writeable (rw) The content of the “event enable register” enables or disables the transition of the respective event bits towards the superposed system structure. Each bit can be enabled or disabled from being transmitted to the system. “Enable registers” are subject to reading or writing, “CLS” or “RST” instruction do not change the contents.

Tab. 7

4.3. “OPERation Status Register”

The “operation status register” reflects all TopCon unit states which are valid during normal operation.

Bit	Description
0	**CALibrating**[1]
1	**SETTing**[1]
2	**RANGing**[1]
3	**SWEeping**[1]
4	**MEASuring**[1]
5	**Waiting für TRIGger** The TopCon device is waiting for a trigger event.
6	**Waiting für ARM**[1]
7	**CORRecting**
8	---[1]
9	---[1]
10	---[1]
11	---[1]
12	---[1]
13	**INSTrument summary**
14	**PROGram running**
15	---[1]

Tab. 8 [1] not used → default value : 0;

4.4. QUEStionable: “Status Register”

The “Questionable Status Register” contains 6 sub-registers. These represent states, error conditions and events inside the system.

Definition of bits:

Bit	Description
0	**VOLTage** Summary of the sub-register VOLTage see section 4.4.1, page 13.
1	**CURRent** Summary of the sub-register CURRent see section 4.4.2, page 15.
2	**Time**[1]
3	**POWer**[1]
4	**TEMPerature summary** Summary of sub-register TEMPerature see section 4.4.3, page 17.
5	**FREQuency**[1]
6	**PHASe**[1]
7	**MODulation**[1]
8	**CALibration**[1]
9	**CONFIguration summary** Summary of the sub-register CONFIguration see section 4.4.4, page 18.
10	**MICellaneous 1 summary** Summary of sub-register MICellaneous 1 see section 4.4.5, page 19.
11	**MICellaneous 2 summary**[1] Summary of sub-register MICellaneous 2 see section 4.4.6, page 22
12	---[1]
13	**INSTrument summary**[1]
14	**Command Warning**[1]
15	---[1]

Tab. 9 [1] not used → default value : 0

For further information refer to TopCon manual.

4.4.1. QUEStionable: “VOLTage Subregister”

Collects all voltage-related errors.

Bit	Description
0	**Overvoltage,** Output overvoltage (30)
1	**V_{Modul} – V_{sense} > Threshold** Voltage differenz between module and sense part exceeded (31)
2	**V_{DCBus} too low ,** Intermediate DC voltage too low (49)
3	**V_{DCBus} too high**[1] Intermediate DC voltage too high (4A)
4	**V_{DCBus} flicker** Unexpected intermediate voltage ripple (4D)
5	**V_{Modul} AD overrange** Module voltage exceeds AD range (84)
6	**V_{Sense} AD overrange** Sense voltage exceeds AD range (86)
7	**V_{System} AD overrange** System voltage exceeds AD range (87)
8	**V_{Modul} AD underrange** Modul voltage below AD range (A4)
9	**V_{Sense} AD underrrange** Sense voltage below AD range (A6)
10	**V_{System} AD underrrange** System voltage below AD range (A7)
11	**TCLIN overvoltage** TC.LIN overvoltage at output (32)
12	**Sense polarity** Negative sense voltage measured (reverse polarity) (33)
13	**RPP voltage unstable** Unstable RPP voltage (34)

The table is continued on the next page. (xy) : Error codes

	Description
14	**Voltage summary fault** Bit collection for: → Intermediate voltage: - exceeds the AD range (89) - AC voltage exceeds the AD range (8B) - below the AD range(A9) - AC voltage below the AD range (AB) → IBC overvoltage: - Uout (KC) - Uclamp (KD) - Intermediate circuit (KE) → IBC intermediate voltage: - too low (L0) - too high (L1) → IBC mains frequency: - too low (L2) - too high (L3) → IBC mains voltage: - too high (L4) - too low (L5) → IBC power factor - too low (L8) → IBC Phase sequence error (LD)
15	---[1]

Tab. 10 [1] not used → default value : 0; (xy) : Error codes

4.4.2. QUEStionable: “CURRent Subregister”

Collects errors related to current levels.

Bit	Description
0	**I^2t** I^2t-monitoring calls for an error (20)
1	**Overcurrent secondary** DC-output overcurrent (user limited) (21)
2	**Overcurrent primary** Primary overcurrent (22)
3	**Gatedrive-A** Gatedriver-A error (23)
4	**Gatedrive-B** Gatedriver-B error (24)
5	**Overcurrent secondary (temperature)** DC output overcurrent (temperature derated value voided) (25)
6	**I_{Module} AD overrange** Module current exceeds AD range (85)
7	**I_{System} AD overrange** System current exceeds AD range (88)
8	**I_{Module} AD underrange** Module current above AD range (A5)
9	**I_{System} AD underrrange** System current above AD range (A8)
10	**TCLIN overcurrent** TC.LIN output overcurrent (26)
11	**TCLIN overload** TC.LIN overload (27)
12	**Overcurrent Isek Q4** Output Q4 overcurrent (user limited) (29)
13	**Overcurrent Isek Q4 (level derated by temperature)** Output Q4 overcurrent (current limited by temperature) (2A)

The table is continued on the next page. (xy) : Error codes

Bit	Description
14	**Current summary fault** Bit collection for: → Primary current: - exceeds AD range (8A) - below AD range (AA) → IBC Gatedrive: - Gatedrive-1 error (K0) - Gatedrive-2 error (K1) - Gatedrive-3 error (K2) → IBC overcurrent: - Isek (K4) - IL (K5) - Iout (K6) - Isys (K7) - short circuit Isek (K8) - in phase L1, L2 or L3 (LB) - in phase L1 (N0) - in phase L2 (N1) - in phase L3 (N2) → IBC IGBT error (L9) → IBC DC offset of the phase currents to high (N3) → Arc detection limit exceeded (28)
15	---[1]

Tab. 11 [1] not used → default value : 0; (xy) : Error codes

4.4.3. QUEStionable: ”TEMPerature Subregister”

Collects errors in relation to temperature.

Bit	Description
0	**Overtemperature rectifier** Rectifier temperature too high (50)
1	**Overtemperature IGBT** IGBT temperature too hight (51)
2	**IGBT temperature AD overrange** IGBT exceeds AD range (94)
3	**Rectifier temperature AD overrange** Rectifier temperature exceeds AD range (95)
4	**Temperature 2 AD overrange** Temperature 2 exceeds AD range (96)
5	**IGBT temperature AD underrange** IGBT temperature below AD range (B4)
6	**Rectifier temperature AD underrange** Rectifier temperature below AD range (B5)
7	**Temperature 2 AD underrange** Temperature 2 below AD range (B6)
8	**TCLIN K1 temperature too high** TC.LIN output stage temperature K1 to hight (52)
9	**TCLIN K2 temperature too high** TC.LIN output stage temperature K2 too high (53)
10	**TCLIN PCB temperature too high** TC.LIN PCB temperature too high (54)
11	**Case inside temperature too high** Device inside temperature too high (55)
12	**PCB temperature AD overrange** PCB temperature exceeds AD range (97)
13	**PCB temperature AD underrange** PCB temperature below AD range (B7)
14	**Temperature summary fault** → IBC heat sink temperature too high (G4) → IBC PCB temperature too high (G5) → IBC heat sink sensor temperature not connected (G6) → IBC inverter heat sink temperature sensor not connected (G7) → IBC temperature of the regeneration units heat sink too high (LA)
15	---[1]

Tab. 12 [1] not used → default value : 0; (xy) : Error codes

4.4.4. QUEStionable: “CONFiguration Subregister”

Collects errors in relation to system configuration.

Bit	Description
0	**Slave-ID multiple used** Slave device ID is not unique (D0)
1	**HMI/RCU-ID multiple used** HMI/RCU ID is not unique (D1)
2	**More than one master** More than one master in the system (D2)
3	**Slave P_{Nom}<> Master P_{Nom}** Nominal power mismatches from master and slave unit (D3)
4	**Slave V_{Nom} <> Master V_{Nom}** Nominal voltage mismatches from master and slave unit (D4)
5	**Slave I_{Nom} <> Master I_{Nom}** Nominal current mismatches from master and slave unit (D5)
6	**Wrong num. devices in series** Number of series units do not match the setting (D6)
7	**Wrong num. devices in parallel** Number of parallel units do not match the setting (D7)
8	**Gap in slave-ID’s** Non consistent numbering of slave ID`s (D8)
9	**Gap in HMI-ID’s** Non consistent numbering of HMI ID’s (D9)
10	**Wrong num. Slaves** Number of the slave units do not match the setting (DA)
11	**Wrong num. multiload devices** Number of the Multiload units do not match the setting (DB)
12	**Illegal slave-ID** Non valid slave-ID (outside of the valid range) (DC)
13	**Illegal HMI-ID** Non valid HMI-ID (outside of the valid range) (DD)
14	**Configuration summary fault** → TC.LIN device: - Non valid TC.LIN-ID (DE) - TC.LIN-ID not unique (DF) - TC.LIN is not activated (E1) - Nominal TC.LIN voltage is not consistent (E2) → Slave: Not all slaves in series are able to provide the Q4-mode (E5) → Sense: Sense in series or with RMB is not allowed (F0)
15	---[1]

Tab. 13 [1] not used default value : 0; (xy) : Error codes

4.4.5. QUEStionable: "MISCellaneous1 Subregister"

Collect errors not represented in the above subregisters.

Bit	Description
0	**SCI summary fault** General errors in SCI communication as: → SCI Checksum error (15), (X5) → SCI parity error (16), (X6) → SCI Overrun error (17), (X7) → SCI Framing error (18), (X8) → SCI Break error (19), (X9) → SCI Timeout within as Talk-Frame (1F) (XF) → RS232 Watchdog error (6C)
1	**Firmware inconsistence** Wrong firmware loaded stands for : →Wrong PDSP version (11) → Wrong modulator software version (7D) → TC.LIN CAN protocol conflict with master (C4) → CAN Protocol version conflict between the bus members (C8) → SW version conflict with the master (C9) → Slave CAN protocol version conflict with master (CA) → HMI CAN protocol conflict with master (CB) → Incompatible PLD version (E3) → Incompatible IBC Version (E4)
2	**Supply summary fault** General errors of the internal power supply as: → Supply too low: ± 5 V (43); ± 15 V (45), (47); ± 24 V (4B); → Supply too high: ± 5 V (44); ± 15 V (46), (48); ± 24 V (4C); → TC.LIN supply too high: ± 5 V (40); ± 15 V (4E); → TC.LIN supply too low: ± 5 V (41); ± 15 V (4F); → IBC supply too high: + 5 V (H5); + 15 V (H3); +24 V (H1); → IBC supply too low: + 5 V (H4); + 15 V (H2); +24 V (H0); → AD supply overrange: + 5 V (90); ± 15 V (91), (92); + 24 V (93); → AD supply underrange: + 5 V (B0); ± 15 V (B1), (B2); + 24 V (B3);
3	**ModDSP summary fault** → Modulator : - Checksum error (detected) (70); - Unknown ID/ Command (76); - Communication failed (7C); - Modulator communication too slow (75) - Modulator Scope Buffers: overflow at reading (7B) - From Modulator received: Unknown state bit (7F) → MainDSP (detected): - Checksum error (detected) (71); - Unknown ID/ Command (77); → SPI-Register: - Transmit- and receive-Register full (73), (74) → General errors: - VzGain too low (78); - IprimGain too low (79); - Manual start not allowed while errors exists (7A)

The table is continued on the next page. (xy) : Error codes

Bit	Description
4	**CAN summary fault** Errors in CAN communication. Master bit for : → CAN controller state: „Bus-Off“ (60); „Error passive“ (61) → CAN Controller Bits: WDIF Bit (62) ; AAIF Bit (63); RMLIF Bit (64) → CAN Kommunikation: - Unknown CAN mailbox (X0) - CAN TX queue full (X3); - CAN RX queue full (X4);
5	**Old EEprom table loaded** Old EEprom table loaded
6	Interlock fault Errors in interlock circuit. Master bit for: → General Interlock errors: - interlock circuit is disconnected (F2) - Safety relay is open (F4) - Interlock 0-level missed (F5) - Interlock circuit connected, but safety relay is open (F6) - No enable signal (F7) → IBC Interlock: - Is disconnected (LF) - Is disconnected (M0) - Safety relay is open (M1) - Interlock circuit is connected, but the safety relay is open (M2)
7	**Custom specific summary fault** Errors in the custom specific function. Master bit for: → Error in the customer specific function (2F) → Internal parameters not set in the ReGen system (E0) → ReGen error (FC) → Switch bridge error (FD)
8	**External shutdown** Modulator PWM external inhibit (F3)
9	**Wrong option code** Wrong software option key code (F1)
10	**Slave communication summary fault** Errors in the slave communication. Master bit for: → CLF: - No CLF received (C0) or invalid (C1) → EOL: - NO EOL received (C2) or invalid (C3) → Slave device: - Non complete EOL received (C3) - No answer from slave (67) - Slave receive no data from master (69) →Master device: incomplete numbers of subframes received by master(C5)
11	**HMI communication summary fault** Errors in the HMI communication. Master bit for: → HMI/RCU: - does not respond (65) - No CFL received (CC) or invalid (CD) - NO EOL received (CE) or invalid (CF) → Master: - Incomplete number of RFL Subframes from HMI/RCU received (C7)

The table is continued on the next page. (xy) : Error codes

Bit	Description
12	**RMB communication summary fault** Error in the RMB communication. Master bit for: → RMB is not connected (68)
13	**Main DSP summary fault** Errors in Main DSP. Master bit for: → State error: - Invalid system state (00) - Invalid module state (01) - ordered state not available (08) - RUN not available, Thyristor is switched on (09) - Overflow: actual value FIFO (multi-unit) full (FF) → EEprom error:- at writing in the EEprom table (04) - No valid EEprom table available (07) → General errors:- Flash timeout error (05) - invalid Interrupt Routine call (0E) - Na active controller defined (0A) - Calculation overflow (02) - Internal debug error (0D) - Power-Up after a Watchdog Reset (W0) - PWM shutdown by an unknown source (WF) → ADC-error: - Sequence error (06) - Timeout error (0B) - DMA Interrupt missing (0C)
14	**PDSP summary fault** Errors in the peripheral. Master bit for: → CAN Bus: - Overflow error in the RX and TX queue (13) (66) - Unknown state bit (1B) → PDSP: - General error (12) - Overflow error: Too much packets received (14) - Checksum error (10) - Communication failed (1E) - Unknown packet received (1C) - Packet from a non-initialized mailbox received (1D) → SCI: - Unknown state bit (1A)
15	---[1]

Tab. 14 [1] not used → default value : 0; (xy) : Error codes

4.4.6. QUEStionable: “MISCellaneous 2 Subregister”

This register is not used in present system.

Bit	Description
0	**IBC comm. summary fault** Error in IBC communication. Master bit for: → IBC reception error (6D) → IBC transmission error (6E) → IBC Talk timeout (6F) → IBC communication Watchdog (J0) → IBC communication SPI error (J1) → IBC LVDS error (J2) → Talk error in IBC communication (X1)
1	**MRC summary fault** Error in responding with TC.MRC. Master bit for: → A MRC Rack has not switched in time (FA) → A MRC Rack has an error (FB)
2	**Reference summary fault** Error in responding with the analogue set values. Mater bit for: → Voltage set value exceeds AD range (80) → Current set value exceeds AD range (81) → Power set value exceeds AD range (82) → Resistance set value exceeds AD range (83) → Voltage set value below AD range (A0) → Current set value below AD range (A1) → Power set value below AD range (A2) → Resistance set value below AD range (A3)
3	**IBC summary fault** Internal IBC error. Mater bit for: → IBC Power-Up after Watchdog reset (G0) → IBC Power-Up after software reset (G1) → IBC EEprom queue overflow (G2) → IBC PLL error (L6) → IBC Timeout while switching on to mains (L7) → IBC self-check (LC) → IBC inverter error (LE)
4	**TCLIN comm. summary fault** Errors in TC.LIN communication. Master bit for: → TC.LIN don’t responds (6A) → TC.LIN CAN error (6B) → TC.LIN does not login to CAN Bus (C6)

The table is continued on the next page. (xy) : Error codes

Bit	Description
5	---[1]
6	---[1]
7	---[1]
8	---[1]
9	---[1]
10	---[1]
11	---[1]
12	---[1]
13	---[1]
14	**Undefined summary fault** Master bit for all actual undefined errors. The error number has to be read out of the device via interface RS-232
15	---[1]

Tab. 15 [1] not used → Default value : 0

4.5. "Standard Event Status Register"

This register collects a number of events throughout the system.

Note that all bits refer also to messages in the Error Event Queue.

Bit	Description
0	**Operation Complete (OPC)** Set if an *OPC instruction was received and all related operations are set. The message –800 „Operation complete" is loaded in the Error Event Queue.
1	**Request Control (RQC)** [1]
2	**Query Error (QYE)** Set if attempted to read from an empty output buffer.
3	**Device Dependent Error (DDE)** Set if an unit specific error occurs
4	**Execution Error (EXE)** An instruction fails because of a specific state of the unit. A parameter is out of range.
5	**Command Error (CME)** The parser detected a syntax error or an invalid instruction.
6	**User Request (URQ)** [1]
7	**Power ON (PON)** [1]

Tab. 16 [1] not used → Default value : 0

4.6. "Error Event Queue"

In order to collect any error messages from the system, the "Error Event Queue" works as a FIFO (First in – first out). If the available storage is fully written, then the oldest error message will be overwritten by error … 350 "Queue overflow"

In the case all errors are being read out from the memory, any further query will be prompted by a"no errors" message.

The Queue will be cleared in the following cases:

- reception of a *CLS instruction
- read-out of the last valid error message

This is a complete description of all possible errors.

4.6.1. Errors related to a command instruction (Command Error)

Error-code	Description
…100	**Command error,** general errors No further details available.
…104	**Data type error,** wrong parameter type The command expects another parameter type. I.e. a string was received instead of an numerical expression.
…115	**Unexpected number of parameters,** wrong number of parameter The command expects another number of parameter.
…120	**Numeric data error,** numerical error A numerical error was encountered in one or more parameters.
…130	**Suffix error,** No further details are available
…131	**Invalid suffix,** invalid suffix Not in accordance with IEEE488.2
…171	**Invalid expression,** invalid instruction Not to be recognized by parser.

Tab. 17

4.6.2. Errors related to execution (Execution Error)

Error-code	Description
…211	**Trigger ignored** A GET or *TRG instruction was received. Due to the actual system state, this instruction have not been processed.
…213	**Init ignored** A trigger initialisation command was received. Due to the actual system state, this instruction have not been processed.
…220	**Parameter error** Instruction not processed due to parameter problem.
…222	**Data out of range** Parameter value is out of range
…240	**Hardware error** An instruction was not proceed due to a communication problem .

Tab. 18

4.6.3. Device-Specific Error

Error-code	Description
…300	**Device-specific error** General unit specific error
…350	**Queue overflow** An error message had to be written into“Error Queue’, but the memory was full.

Tab. 19

4.6.4. Query Error

Error-code	Description
…410	**Query INTERRUPTED** See IEEE488.2, 6.3.2.3 Example: A new query was received while the preceding is still active.
…420	**Query UNTERMINATED** See IEEE488.2, 6.3.2.2 Example: The unit was addressed as “talker”, but no query was posed at all.

Tab. 20

4.6.5. Operation complete event

Error-code	Description
…800	**Operation complete** The unit has worked out all instructions, sync protocol enabled by *OPC. bit 0 of „Standard Event Status Register“ is set.

Tab. 21

4.7. "Status Byte"

This byte reflects the state of several internal state structures according to IEEE488.1. The byte can be read either by "Serial Poll" or by the query instruction *STB?. The latter method clears whether the byte itself nor MSS nor RQS.

Bit	Description
0	**Reserviert**[1]
1	**Reserviert**[1]
2	**Error Event Queue** Set if one or more error messages are inside the error queue.
3	**Questionable Status Summary** Set if a bit in the „Questionable register" is set.
4	**Message Available (MAV)** Set if unit is ready to send an answer. Not set if output queue is empty.
5	**Standard Event Status Summary** Set if a bit in „Standard Event Status" is set.
6	**Request Service (RQS) bzw. Master Status Summary (MSS)** This bit indicates the presence of a „Service Request'.
7	**Operation Status Summary** Set if a bit in „Operation Status" is set and the appropriate bit in the „Operation Status Enable Register" is set too.

Tab. 22 [1] not used → Default value : 0

4.7.1. Master Summary Status (MSS)

MSS (Master Status Summary) is set if a bit in the „Status Byte" was set and therefore the according bit in the „Service Request Enable Register" is set too.

MSS is the answer to a „*STB?" query in bit 6. MSS is not sent following a „Serial Poll answer'.

4.7.2. Request Service (RQS)

Bit 6 (RQS) is set if a bit of the status byte is set (and therefore the according bit in the „Service Request Enable Register').

Note that the SRQ signal is activated on the GPIB bus in this case.

4.8. Status Register Commands

The instructions described below allow to access to the Register of the Status structure.

4.8.1. Setting the Status Register

The following instructions configure the Status Register.

Command	Description
STATus:PRESet	---

Tab. 23

All bits of “Event Enable Register“ are set to a predefined value, see table.

PRESet clears no “Event Register“ nor the “Event Queue”. Settings of “Event Enable Register“ may change superposed data.

Settings according to SCPI status structures cause bits of “Event Enable Register“ to be reset.

All unit-specific status structures however will set all bits of the “Event Enable Register“ to “1”.

The“Service Request Enable Register“ as also the “Parallel Poll Enable Register“ remain unchanged.

Register	Filter/Enable	
Operational Status	Enable Register Positive Transition Filer	Value: “0“ Value: “1“ (permanent)
Questionable Status	Enable Register Positive Transition Filer	Value: “0“ Value: “1“ (permanent)
Alle anderen	Enable Register Positive Transition Filer	Value: “0“ Value: “1“ (permanent)

Tab. 24

4.8.2. Clear Status Register

Command	Notices
***CLS**	Clears the Event Register, the Status Byte, the Standard Event Status as also the Error Event Queue.

Tab. 25

4.8.3. Reading and setting of Event Enable Register (Standard Event Status Register)

The Event Enable Register as part of the Standard Status Register is read by the following commands:

Command	Notices
***ESE <SEE value>**	<SRE value> Value range: 0 … 255
***ESE?**	Answer is a decimal number Value range: 0 … 255

Tab. 26

Example:

- Input: *ESE #H18

↳ Sets bits 3 and 4 of Event Enable Register

- Input: *ESE 127

↳ Sets bits 0 … 6

4.8.4. Readout of Standard Event Status Register

The readout of this register will clear the register content!

Command	Notices
ESR?	Answer is a decimal number Value range: 0 … 255

Tab. 27

4.8.5. Readout and setting of "Service Request Enable Register"

*SRE will set the "Service Request Enable Register".

This determines, which bits of the "Status byte" will call for a "Service Request" at the controller. *SRE? Is the corresponding read instruction.

Command	Notices
***SRE <SRE value>**	<SRE value> Value range: 0 … 255
***SRE?**	Answer is a decimal number Value range: 0 … 255

Tab. 28

Example:

- Input: *SRE 128

↳ Sets bit 7 of "Service Request Enable Register".
As a result the Operation Status Summary bit is enabled to send a Service Request.

4.8.6. Read out of "Status Byte"

*STB? Don't clears the "Status Byte" content while reading.

A "Serial Poll" call also reads the "Status Byte" but with the following difference:

*STB? Causes sending of a MSS message, while "Serial Poll" send a RQS message.

Command	Notices
***STB?**	Answer is a decimal number Value range: 0 … 255

Tab. 29

4.8.7. Readout and setting of "Parallel Poll Enable Register"

The "Parallel Poll Enable Register" is a 16 bit register.

The "Parallel Poll Enable Register" is a 16-bit register. The lower 8 bits of "Parallel Poll Enable Register" determine, which bit of the "Status Byte" is enabled to change the „ist" (Individual Status). Each of the lower 8 bits is logically AND-ed with a corresponding bit of the „Status Byte. The resulting bits are OR-ed and form the „ist" Status. The upper 8 bits remain unused at present.

Command	Notices
***PRE <PRE value>**	<PRE value> Value range: 0 … 32767
***SRE?**	Answer is a decimal number Value range: 0 … 32767

Tab. 30

Example:

- Input: *PRE 8

↳ Set bit 3 of “Parallel Poll Enable Register”.
A set bit in the Questionable Status Summary Bit causes setting of “ist“ Status.

4.8.8. Reading of Individual Status “ist”

Readout of “ist“ (Individual Status) Status will not cause a “Parallel Poll”.

Command	Notices
IST?	Answer is binary Value range: „0“ or „1“

Tab. 31

4.8.9. Reading of “Error Event Queue”

The “Error Event Queue” is a FIFO. The oldest value will be read first. After reading, the value will be cleared.

Command	Notices
SYSTem:ERRor[:NEXT]?	The answer contains an error number and an error comment; both separated by a comma. All possible error numbers are listed in chapter “Error Event Queue”.

Tab. 32

4.8.10. Reading of "Condition Registers"

This command applies also to all "Questionable Status Structures" and to "Operation Status Structure".

Command	Notices
STATus:QUEStionable:CONDition?	Answer is a decimal number Value range: 0 … 32767

Tab. 33

4.8.11. Reading of "Event Registers"

This command applies also to all "Questionable Status Structures" and to "Operation Status Structure".

Command	Notices
STATus:QUEStionable[:EVENt]?	Answer is a decimal number Value range: 0 … 32767

Tab. 34

4.8.12. Reading of "Event Enable Registers"

This command applies also to all "Questionable Status Structures" and to "Operation Status Structure".

Command	Notices
STATus:QUEStionable:ENABle?	Answer is a decimal number Value range: 0 … 32767
STATus:QUEStionable:ENABle <status enable>	<status enable> input Value range: 0...32767

Tab. 35

5. Operational commands

For normal operation, only a few commands are necessary.

5.1. Output switching on and off

CAUTION

Danger of life by the output voltage!

- Depending on the device type the output voltage can be very dangerous!

⇒ Protect the output terminals from accidental touch.
⇒ Pay attention to the safety instructions in the TopCon manual and follow the instructions.

5.1.1. Set " VOLTAGE ON" and "OFF"

OUTPut sets or resets the system output. To test the output state , the OUTPut? Is used instead.

Command	Notices
OUTPut[:STATe] {ON|OFF|0|1}	ON bzw. "1": Switch on the output. OFF bzw. "0": Switch off the output.
OUTPut[:STATe]?	Answer "0": for switched off of output. Answer "1": for switched on of output.

Tab. 36

Example:

- Input: OUTPut ON

↳ The output is switched on.

- Eingabe OUTPut 0

↳ The output is switched off.

5.2. Set values (direct access)

All set values are treated internally with a resolution of 1/4000.
All minor differencies are rounded.

5.2.1. Setting and readout of Voltage set value

Set the voltage set value and readout the set value as follows::

Commande
[SOURce:]VOLTage[:LEVel][:IMMediate][:AMPLitude] <voltage>|MINimum|MAXimum}
[SOURce:]VOLTage[:LEVel][:IMMediate][:AMPLitude]?

Tab. 37

Parameter	Notices
<voltage>	voltage value within the range of the TopCon unit(s) used. Without a dimensional designator, Volts are assumed. Possible dimensions are [mV], [V] oder [kV]. (Not case sensitive)
MINimum	Set the smallest possible voltage. Default value: 0V
MAXimum	Set the maximum voltage value Default value: Nominal voltage

Tab. 38

Beispiele:

- Input: VOLTage 50V

↳ Set 50 V

- Input: VOLT 0.23kV

↳ Set 230 V.

- Input: SOUR:VOLT MAX

↳ Set maximum possible voltage value.

- VOLT?

↳ Returns the actual system voltage [V]

5.2.2. Setting and readout of current set value

Setting of current set value corresponds to that of voltage set value.

Command
[[SOURce:]CURRent[:LEVel][:IMMediate][:AMPLitude] {<current>|MINimum|MAXimum}
[SOURce:]CURRent[:LEVel][:IMMediate][:AMPLitude]?

Tab. 39

Parameter	Notices
<current>	current value within the range of the TopCon unit(s) used. Without a dimensional designator, amperes are assumed. Possible dimensions are [mA], [A] oder [kA]. (Not case sensitive).
MINimum	The minimum possible current will be set. Default value: 0 A
MAXimum	The maximum possible current will be set. Default value: Nominal current

Tab. 40

Examples:

- Input: CURRent 100A

↳ A set value of 100 A will be set.

- Input: CURR 0.153kA

↳ A set value of 153 A will be set.

- Input: SOUR:CURR MAX

↳ A set value of maximum possible current will be set.

- Input: CURR?

↳ Returns the actual current set value [A].

5.2.3. Setting and readout of Power set value

This setting corresponds to the settings of voltage and current.

Command
[SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude] {<power>|MINimum|MAXimum}
[SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]?

Tab. 41

Parameter	Notices
<power>	is a power value within the range of the TopCon unit(s) used. Without a dimensional designator, Watts are assumed. Possible dimensions are [W] oder [kW]. (Not case sensitive).

Tab. 42

The return value following a query is given in Watts [W] .

5.2.4. Setting and readout of Resistance

This setting corresponds to the settings of voltage and current.

Commande
[SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude] {<resist>|MINimum|MAXimum}
[SOURce:]RESistance[:LEVel][:IMMediate][:AMPLitude]?

Tab. 43

Parameter	Notices
<resist>	is a value in the range of 0...1Ω. (Extended value possible by adapting the ‚TopCon grid file). Without a dimensional designator, Ohms [Ω] are assumed. Accepted dimensions are [µΩ], [Ω] und [kΩ]. „UR“ resp. „UOHM“ stands for [µΩ]; „R“ resp.. „OHM“ for [Ω]; and „KR“ resp. „KOHM“ for [kΩ]. (Not case sensitive).

Tab. 44

The return value following a query is given in Ohms [Ω] .

5.3. Changing the limits for overvoltage and overcurrent

5.3.1. Setting and readout of overvoltage limit

The limit for an overvoltage detection is set and read as follows:

Commands
[SOURce:]VOLTage:PROTection[:OVER][:LEVel] {<voltage>|MINimum|MAXimum}
[SOURce:]VOLTage:PROTection[:OVER][:LEVel]?

Tab. 45

Parameter	Notices
<voltage>	Is a voltage level up to 110% of the nominal TopCon voltage. Without any dimensional designator Volts are assumed. Possible dimensional designators are [mV], [V] or [kV]. (Not case sensitive).
MINimum	Set the smallest possible voltage level. Default: 0 V
MAXimum	Set the maximum possible voltage level. Default: 110 % of nominal voltage

Tab. 46

Beispiele:

- Input: VOLTage:PROTection 50V

↳ Overvoltage limit is set to 50 V.

- Input: VOLT:PROT 0.1kV

↳ Overvoltage limit is set to 100 V.

- Input: SOUR:VOLT:PROT MAX

↳ Overvoltage limit is set to maximum (110 % of nominal voltage).

- Input: VOLT:PROT?

↳ Returns the actual value of overvoltage limit [V].

5.3.2. Setting and readout of overcurrent limit

The limit for an overcurrent detection is set and read as follows:

Command
[SOURce:]CURRent:PROTection[:OVER][:LEVel] {<current>|MINimum|MAXimum}
[SOURce:]CURRent:PROTection[:OVER][:LEVel]?

Tab. 47

Parameter	Notices
<current>	is a current level up to 110% of the nominal TopCon current. Without any dimensional designator Amperes are assumed. Possible dimensional designators are [mA], [A] or [kA]. (Not case sensitive).
MINimum	Set the smallest possible overcurrent level. Default value: 0 A
MAXimum	Set the maximum possible overcurrent level. Default value: 110 % of nominal current

Tab. 48

Examples:

- Input: CURRent:PROTection 75A

↳ Overcurrent limit is set to 75 A.

- Input: CURR:PROT 0.01kA

↳ Overcurrent limit is set to 10 A.

- Input: SOUR:CURR:PROT MAX

↳ Overcurrent limit is set to maximum overcurrent level (110 % of nominal current).

- Input: CURR:PROT?

↳ Returns the value of actual overcurrent level [A].

5.4. Measuring of voltage, current and power within the system

5.4.1. Readout of the actual system voltage

Command	Notices
MEASure[:SCALar]:VOLTage[:DC]? [<expected_value>[,<resolution>]]	The parameters <expected_value> und <resolution> are not processed and need not to be given. By compatibility reasons, both parameters may be defined by a voltage value, a MINimum, MAXimum or DEFault value without further processing.

Tab. 49

All voltage measurements are returned in [V]

Examples:

- Input: MEASure:SCALar:VOLTage? DEF,DEF

↳ DEF und DEF will be ignored.

- Input: MEAS:VOLT?

↳ Both queries will return the actual system voltage in [V]

5.4.2. Readout of the actual system current

Command	Notices
MEASure[:SCALar]:CURRent[:DC]? [<expected_value>[,<resolution>]]	The parameters <expected_value> und <resolution> are not processed and need not to be given. By compatibility reasons, both parameters may be defined by a current value, a MINimum, MAXimum or DEFault value without further processing.

Tab. 50

All measured current values are returned in [A].

Examples:

- Input: MEASure:SCALar:CURRent? MAX,DEF

↳ MAX und DEF will be ignored.

- Input: MEAS:VOLT?

↳ Both queries will return the actual system current in [A]

5.4.3. Readout of the actual system power

Command	Notices
MEASure[:SCALar]:POWer[:DC]? [<expected_value>[,<resolution>]]	The parameters <expected_value> und <resolution> are not processed and need not to be given. By compatibility reasons, both parameters may be defined by a power value, a MINimum, MAXimum or DEFault value without further processing.

Tab. 51

All measured power values are returned in [W].

Examples:

- Input: MEASure:SCALar:POWer? MAX,DEF

↳ MAX und DEF will be ignored.

- Input: MEAS:VPOW?

↳ Both queries will return the actual system power in [W]

5.5. Trigger functions

SCPI allows for the trigger function to different trigger signal sources:

- BUS : Trigger via GPIB bus
- IMMediate: direct triggering

Initialisation of as trigger sequence may be single or continuous mode.

5.5.1. Selection of the trigger source

Command
TRIGger[:SEQuence]:SOURce {BUS|IMMediate}
TRIGger[:SEQuence]:SOURce?

Tab. 52

Parameter	Notices
BUS	Triggering via GBIB bus by a “GET“ message or by a “*TRG“ command.
IMMediate	Triggering follows immediately after initialisation

Tab. 53

Selection of the trigger source will not initiate the trigger facility. This is done only by additional initialising of the trigger sequence.

5.5.2. Initialising of a trigger sequence

A trigger sequence may be initialized either once or continually. At continually mode the unit is awaiting the next trigger sequence directly after receiving a trigger input. At this point, bit 5 of “Operation Status” (Waiting for TRIGgering) is set.

After initial turn-on of the system, this bit is reset, disabling the continous trig mode.

Command	Notices
INITiate:CONTinous {ON|OFF}	ON / OFF mode of initialisation
INITiate:CONTinous?	Readout of mode (Answer is „0“ or „1“)
INITiate[:IMMediate]	Immediate initialisation

Tab. 54

Example: Continuous Initialisation

- Input: INIT:CONT ON

↳ Enable continuous initialisation.
The system is waiting for a trigger event.

Example: Single Initialisation

- Input: INIT:IMM

↳ The system is waiting for a single trigger event.

5.5.3. Single, immediate triggering

The trigger source can be set for an immediate single triggering, even if the source was set to BUS previously.

Command	
TRIGger[:SEQuence]:IMMediate	

Tab. 55

Example:

- Input: TRIG:IMM

↳ Activates a single trigger event, if the trigger module was initialized previously.

5.5.4. Setting of set values after triggering

The set values for voltage, current, power and ‘internal resistance’ may be set by means of a trigger sequence.

The appropriate commands are:

Commande
[SOURce]:VOLTage[:LEVel]:TRIGgered[:AMPLitude] {<voltage>|MAXimum|MINimum
[SOURce]:CURRent[:LEVel]:TRIGgered[:AMPLitude] {<current>|MAXimum|MINimum}
[SOURce]:POWer[:LEVel]:TRIGgered[:AMPLitude] {<power>|MAXimum|MINimum}
[SOURce]:RESistance[:LEVel]:TRIGgered[:AMPLitude] {<resist>|MAXimum|MINimum}

Tab. 56 All parameters match the normal set value settings.

The setting of the current set value has the first priority followed by the voltage, power and the internal resistance.

5.5.5. Triggering via GPIB bus

After selecting “BUS” as trigger source and initializing of the trigger sequence, the triggering event may be realizied by the commands:

Command	Notices
***TRG**	Trigger common command according IEEE488.2
„GET“	IEEE488.1 command “Group Executive Trigger”

Tab. 57

5.5.6. Example for a complete trigger sequence

The following profile is given:

- The system delivers 50 V
  Current set value 10 A
  max. power 10 kW
- At a trigger signal “GET” all units of the bus system have to increase the voltage to 60 V maximum power 5 kW.

Command string (short syntax form):

- VOLT 50V;CURR 10A;POW 10KW (normal set value setting)
- TRIG:SOUR BUS
- VOLT:TRIG 60V;:POW:TRIG 5KW
- INIT (trigger initialisation)
- „GET“ (IEEE488.1 Command)

↳ The trigger event starts immediately

6. Special commands

6.1. Register access

The following commands allow for a direct access to registers, which are not implemented in the normal GPIB instruction set.

Special commands are used only for very specialised measures. Some care has to be taken in order to not disturb or mismatch unit-relevant internal settings.(TopCon controller parameter settings or similar). Please contact the manufacturer for details.

6.1.1. Writing a register

Command
TOPCon:REGister:WRITe <register>,<value>

Tab. 58

Parameter	Notices	
	Until version[1] V4.19.99	From Version[1] V4.20.00
<register>	Value range: 16 Bit Value range which is representing the address of the register.	Value range: 32 Bit
<value>	Is the value which will write in the register, The effective range and the interpretation of the sign depends also on the respective register. Value range: 0 … 65535	

Tab. 59 [1]Depending of the device version.

Example:

- Input: TOPC:REG:WRIT #H5140,100

↳ Sets the proportional gain of the TopCon voltage controller to 100.

- Input: TOPC:REG:WRIT #H30251D,61536

↳ Sets the current limit of a TC.GSS device in the Q4 operation to value -4000 (65536-4000=61536)

6.1.2. Reading a register

Command
TOPCon:REGister:READ? <register>

Tab. 60

Parameter	Notices	
	Until version[1] V4.19.99	From Version[1] V4.20.00
<register>	Value range: 16 Bit Value range which is representing the address of the register	Value range: 32 Bit
	Die Answer is a decimal number. The interpretation of the sign depends also on the respective register. Value range: 0 … 65535	

Tab. 61 [1]Depending of the device version.

Example: (Writing and reading back)

- Input: TOPC:REG:WRIT #H5140,100

↳ Sets the proportional gain of the TopCon voltage controller to 100.

- TOPC:REG:READ? #H5140

↳ Answer: “100” (content of Register 0x5140)

- Input: TOPC:REG:WRIT #H30251D,61536

↳ Sets the current limit of a TC.GSS device in the Q4 operation to the value -4000 (65536-4000=61536)

- Input: TOPC:REG:READ? #30251D

↳ Answer „61536“.
This corresponds to the value of -4000 (61536-65536=-4000), because the register 0x30251D has a signed value.

6.2. Saving of user settings

6.2.1. Saving of data in the system-EEprom

User settings like set values can be saved in the EEprom in order to retain data live even after a power-down.

Command	
***SAV 0**	

Tab. 62

6.3. Synchronisation commands

The system treats all commands in a sequential mode. The following commands are implemented only by compatibility reasons.

Command	
***OPC**	Sets bit 0 (Operation Complete) in the “Standard Event Status Register”.
***OPC?**	Write a “1“ in the output buffer, if all received commands are terminated.
***WAI**	Wait until all previous commands have been terminated.

Tab. 63

6.4. System commands

6.4.1. Readout of identification

The indetification string can be readout by the following command:

Command	Notices
***IDN?**	The Answer has the following format: „Regatron AG,TopCon Quadro,xxxxyyzzz,Vm,ss,rr“ The firmware version may also appear in customer specific firmware in Format: Vmm,ss,rr. Xxxxyyzzz: Serial number of unit, Vm,ss,rr : Version of the firmware.

Tab. 64

6.4.2. Readout of the SCPI functionality

The following command returns information about the actual SPI functional state:

Command	Notices
SYSTem:CAPability?	Answer: „(DCSUPPLY WITH(MEASURE&TRIGGER))“

Tab. 65

6.4.3. Readout of SCPI version

The SCPI version describes the actual standard of implementation

Command	Notices
SYSTem:VERSion?	Return format: „YYYY.V“

Tab. 66

7. Appendix A

7.1. SCPI format conventions

7.1.1. Hierarchy of Commands

SCPI commands are based on a hierarchic structure. “Nodes” and “roots” are basic elements of the SCPI language.

An example:

[SOURce]
:VOLTage
[:LEVel]
[:IMMediate]
[:AMPLitude] {<voltage>|MINimum|MAXimum}
:CURRent
[:LEVel]
[:IMMediate]
[:AMPLitude] {<current>|MINimum|MAXimum}

SOURce is a command („node“), fixed on the highest level, the so-called “root” level. VOLTage and CURRent are “nodes” on the next lower level. A double-point [:] separates the levels.

7.1.2. Short and long-form commands

SCPI commands can be written eitherin short or extended (long) form. In the short-form all lower case and [] can be omitted. In the source text upper and lowercase is not distingued.

Example:

- Input:
  SOURce:VOLTage:LEVel:IMMediate:AMPLitude
  {<voltage>|MINimum|MAXimum}

↳ May be written also as follows:
VOLT {<voltage>|MINimum|MAXimum}
or
volt {<voltage>|MINimum|MAXimum}

Since the „root“ command „SOURce“ is written in brackets [ ], it can be omitted. The same applies to the node-commands „LEVel“, „IMMediate“ and „AMPLitude“.

7.1.3. Separation signs

The following signs are used within the syntax and are transmitted through the bus:

Sign	Notices
Double-point ":"	Separation sign between two levels
Space sign " "	Separates parameters from command body
comma ","	Separates parameters among each other if more the one used.
semicolon ";"	Separates commands standing on same level.
Semicolon with double-point ";:"	Commands origination from different levels are separated by this sign combination. Note, that a "double-point" preceding a command resets the SCPI path to "root".

Tab. 67 Diese Zeichen werden an das Gerät gesendet.

Example:

- Input:
  SOUR:VOLT 0.1V;CURR 0.3A

↳ Equals:
SOUR:VOLT 0.1V
SOUR:CURR 0.3A

Example: Semicolon with double-point

- Input:
  SOUR:VOLT 0.1V;:MEAS:VOLT?

The following signs are used for programming purposes and are not transmitted through the bus:

Sign	Notices
Braces "{ }"	The programmer chooses between elements indicated between brackets.
Vertical bar "|"	Separates parameters noted between brackets.
Angle brackets "< >"	A specific value has to be entered.
Square Brackets "[ ]"	Specification of elements in brakcets are obsolete and can be omitted.

Tab. 68 Diese Zeichen werden nicht an das Gerät gesendet.

7.1.4. Parameter Types

The following parameter types are used:

- **Boolean type**
  Binary value “0” or “1”, these values can be also written as “OFF” resp. “ON” alternatively.
- **Discrete type**
  Discrete mnemonic value, limited number of arguments.

  Example: TRIG:SOUR {BUS|IMMediate}

  Note: The answer on a query always is in short form and in upper case..
- **Numerical type**
  Numerical parameters allow for decimal point, signs and scientific notation. Together with a number of commands, discrete parameters are allowed too; like MINimum, MAXimum or DEFault. The unit itself translates these parameters to corresponding values.
  Example for a command with numerical parameter:

  SOUR:VOLT {<voltage>|MIN|MAX}

7.1.5. Queries

A Question mark simply form a query out of a command. Note that nevertheless not all commands allow for a query. Please refer to the command summary for details. The answer of a query is an ASCII string.

Example: (Simple query)

- Query : VOLT?

7.1.6. Termination character

Each ASCII string sent to the unit hat to be terminated by an appropriate character.

Several character are allowed:

- IEEE488.1 EOI (end or identify) message.
- <NL> (New Line)
  <NL> character code 0x0A.
- <CR> (Carriage Return) followed by <NL>
- <NL> followed by EOL message.
- <CR> followed by <NL> and EOL message.

Each answer strin re-sent by the unit is terminated by <NL> followed bac EOL message.

7.2. Summary of commands

The following table show all possible commands which are to be sent to TopCon units via GPIB bus.

The table lists:

- SCPI command in long-form
- A short description to each command
- Query: Ability of each command to form also a query

7.2.1. IEEE488.2 Commands

SCPI-Command	Notices	Query
***CLS**	Clear Status. All data structures cleared	No
***ESE?**	Query of Standard Event Status Enable Register.	No
***ESE <ESE value>**	Setting of Standard Event Status Enable Register.	Yes
***ESR?**	Query of Standard Event Status Register.	No
***IDN?**	Identification Query. ID string is being read	No
***IST?**	Individual Status Query	No
***OPC**	Operation Complete Command .Sets the corresponding bit in Standard Event Status Register, if all previous commands are worked out properly.	No
***OPC?**	Returns a „1“ , if all previous commands are terminated properly.	No
***PRE?**	Query of Parallel Poll Enable Register.	No
***PRE <PRE value>**	Setting of Parallel Poll Enable Register.	Yes
***RST**	Reset. A warm-start sequence is initialised.	No
***SAV 0**	Saves all actual settings in the EEPROM.	No
***SRE?**	Query of Service Request Enable Register.	No
***SRE <SRE value>**	Setting of Service Request Enable Register.	Yes
***STB?**	Query of the Status Byte.	No
***TRG**	Send IEEE488.2 trigger-message through GPIB Bus	No
***TST?**	Unit self-test successful terminated?	No
***WAI**	Wait until No-operation-pending Flag is set.	No

Tab. 69

7.2.2. Measurement commands

SCPI-Command	Notices	Query
[:]MEASure[:SCALar]:CURRent[:DC]? [<expected_value>[,<resolution>]]	Returns the actual current value	No
[:]MEASure[:SCALar]:VOLTage[:DC]? [<expected_value>[,<resolution>]]	Returns the actual voltage value	No
[:]MEASure[:SCALar]:POWer[:DC]? [<expected_value>[,<resolution>]]	Returns the actual output power value	No

Tab. 70

7.2.3. Trigger commands

SCPI-Command	Notices	Query
[:]TRIGger[:SEQuence]:SOURce {BUS|IMMediate}	Defines the trigger source	Yes
[:]TRIGger[:SEQuence]:IMMediate	Triggers the initialized sequence once; the trig source is set to IMMediate once for this event.	No
[:]INITiate:CONTinous {ON|OFF}	Sets the trigger block to continuous initialisation.	Yes
[:]INITiate[:IMMediate]	Initialize a trig sequence	No

Tab. 71

7.2.4. System Commands

SCPI-Command	Notices	Query
[:]SYSTem:CAPability?	Returns the functional settings of the unit	No
[:]SYSTem:ERRor[:NEXT]?	Returns the next error out of the error memory	No
[:]SYSTem:VERSion?	Returns the SCPI version (1999.0).	No

Tab. 72

7.2.5. ON-OFF commands of the output

SCPI-Command	Notices	Query
[:]OUTPut[:STATe] {ON|OFF}	Switches the unit output ON or OFF respectively	Yes

Tab. 73

7.2.6. Set value commands

SCPI-Command	Notices	Query
[[:]SOURce]:CURRent[:LEVel][:IMMediate][:AMPLitude] <current>|MAXimum|MINimum}	Sets immediately the set value for current.	Yes
[[:]SOURce]:VOLTage[:LEVel][:IMMediate][:AMPLitude] {<voltage>|MAXimum|MINimum}	Sets immediately the set value for voltage	Yes
[[:]SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude] {<power>|MAXimum|MINimum}	Sets immediately the set value for output power.	Yes
[[:]SOURce]:RESistance[:LEVel][:IMMediate][:AMPLitude] {<resistance>|MAXimum|MINimum}	Sets immediately the set value for internal resistance simulation	Yes
[[:]SOURce]:CURRent[:LEVel]:TRIGgered[:AMPLitude] {<current>|MAXimum|MINimum}	Sets the set value for current following a trigger event	Yes
[[:]SOURce]:VOLTage[:LEVel]:TRIGgered[:AMPLitude] {<voltage>|MAXimum|MINimum}	Sets the set value for voltage following a trigger event	Yes
[[:]SOURce]:POWer[:LEVel]:TRIGgered[:AMPLitude] {<power>|MAXimum|MINimum}	Sets the set value for output power following a trigger event.	Yes

Tab. 74

7.2.7. Protection set values commands

SCPI-Command	Notices	Query
[[:]SOURce:]CURRent:PROTection[:OVER][:LEVel] {<current>|MAXimum|MINimum}	Sets the overcurrent limit	Yes
[[:]SOURce:]VOLTage:PROTection[:OVER][:LEVel] {<voltage>|MAXimum|MINimum}	Sets the overvoltage limit	Yes

Tab. 75

7.2.8. Status Commands

SCPI-Command	Notices	Query
[:]STATus:OPERation[:EVENt]?	Returns the Event Register of the Status Data structure	No
[:]STATus:OPERation:CONDition?	Returns the Condition Register of the Status Data structure	No
[:]STATus:OPERation:ENABle <status enable>	Sets the Enable Register of the Status Data structure	Yes
[:]STATus:PRESet	Presets all registers of the Status Data structure according to SCPI 1999.0	No
[:]STATus:QUEStionable[:EVENt]?	See Operation Status	No
[:]STATus:QUEStionable:CONDition?	See Operation Status	No
[:]STATus:QUEStionable:ENABle <status enable>	See Operation Status	Yes
[:]STATus:QUEStionable:CURRent[:EVENt]?	See Operation Status	No
[:]STATus:QUEStionable:CURRent:CONDition?	See Operation Status	No
[:]STATus:QUEStionable:CURRent:ENABle <status enable>	See Operation Status	Yes
[:]STATus:QUEStionable:VOLTage[:EVENt]?	See Operation Status	No
[:]STATus:QUEStionable:VOLTage:CONDition?	See Operation Status	No
[:]STATus:QUEStionable:VOLTage:ENABle <status enable>	See Operation Status	Yes
[:]STATus:QUEStionable:TEMPerature[:EVENt]?	See Operation Status	No
[:]STATus:QUEStionable:TEMPerature:CONDition?	See Operation Status	No
[:]STATus:QUEStionable:TEMPerature:ENABle <status enable>	See Operation Status	Yes
[:]STATus:QUEStionable:CONFiguration[:EVENt]?	See Operation Status	No
[:]STATus:QUEStionable:CONFiguration:CONDition?	See Operation Status	No
[:]STATus:QUEStionable:CONFiguration:ENABle <status enable>	See Operation Status	Yes
[:]STATus:QUEStionable:MISCellaneous1[:EVENt]?	See Operation Status	No
[:]STATus:QUEStionable:MISCellaneous1:CONDition?	See Operation Status	No
[:]STATus:QUEStionable:MISCellaneous1:ENABle <status enable>	See Operation Status	Yes
[:]STATus:QUEStionable:MISCellaneous2[:EVENt]?	See Operation Status	No
[:]STATus:QUEStionable:MISCellaneous2:CONDition?	See Operation Status	No
[:]STATus:QUEStionable:MISCellaneous2:ENABle <status enable>	See Operation Status	Yes

Tab. 76

7.2.9. Special commands

SCPI-Command	Notices	Query
[:]TOPCon:REGister:READ? <register>	Reads data from specified TopCon register	No
[:]TOPCon:REGister:WRITe <register>,<value>	Writes data into secified TopCon register	No

Tab. 77

7.2.10. Table of used expressions

Expression	Notices
current	Numerical value in “MA”, “A“ or „KA“. Without unit specification, „A“ is assumed. The range corresponds to TopCon nominal current range.
voltage	Numerical value in “MV”, “V“ or „KV“. Without unit specification, „V“ is assumed. The range corresponds to TopCon nominal voltage range.
power	Numerical value in “W” or „KW“. Without unit specification, „W“ is assumed. The range corresponds to TopCon nominal output power range.
resistance	Numerical value in “R”, „OHM“, „KR“ or “KOHM“. Without unit specification, „OHM“ is assumed. Value range: 0 … 10 Ohm.
MINimum	Minimum possible value
MAXimum	Maximum possible value
register	Numerical value without unit specification. Value range: 0 ... 65535
value	Numerical value without unit specification. Value range: 0.....65535
ESE value	Numerical value without unit specification. Value range: 0 ... 65535
SRE value	Numerical value without unit specification. Value range: 0 ... 65535
PRE value	Numerical value without unit specification. Value range: 0 ... 65535
status enable	Numerical value without unit specification. Value range: 0 ... 65535
Expected_value	Allowed by compatibility reasons, not processed
resolution	Allowed by compatibility reasons, not processed
ON	Boolean value, ON od OFF
OFF	Boolean value, “0” or “1”
IMMediate	Immediate trigger without awaiting an event
BUS	Trigger by IEEE488.2 command *TRG or IEEE488.1 command „GET“

Tab. 78

7.3. Application: Function Generator

7.3.1. Loading Function Sequences by GPIB interface

In order to load a function sequence which is stored on the firmware flash memory

the following GPIB commands can be used (#Hxxxx are hexadecimal numbers)

Procedure

- First, write the Function Sequence number to load (n=1...1000)
  TOPC:REG:WRIT #H5cda, n
- Then perform the Load command:
  TOPC:REG:WRIT #H5cdb, 1
- The status of the load command can be polled:
  TOPC:REG:READ? #H5cdc

↳ Possible results are:
 1: Busy, load command not finished yet
 0: Load command executed without errors
-1: Internal timeout on loading
-3: The requested Function Sequence (n) does not exist

Procedure to programm a Function sequence

- In order to start/stop/pause a Function Sequence use this command: (TopCon has to be VOLTAGE ON already, command: OUTP ON)
  TOPC:REG:WRIT #H5ce7, cmd
  where 'cmd' can be:
  1: stop a Function Sequence
  2: start and/or continue (after a pause) ...
  3: pause ...
  4: restarts the complete curve ...

CAUTION Do not write to undocumented registers!

As these commands all write directly to firmware memory (RAM) be careful to enter the correct register und data values.

Do not write to undocumented registers!

7.4. Controller parameterization with register commands

Access to the different controller parameter variables (PID for voltage, current, power) is provided by the use of the special functions “register read” and “register write”. For read/write operations the register addresses needed are:

7.4.1. Voltage controller

Identifier	Register address [HEX] and range
P gain	**#H5140** Value range[1]: 0 ... 32767
I gain	**#H5141** Value range[1]: 0 ... 32767
D gain	**#H5142** Value range: 0 ... 32767
T1 gain	**#H5151** Value range[1]: 0 ... 32767
FFwd gain	**#H514C** Value range[1]: 0 ... 32767
P adaptive gain	**#H515D** Value range[1]: 0 ... 16384
I adaptive gain	**#H515E** Value range[1]: 0 ... 16384

Tab. 79

[1] Given are the maximum values.
This does not imply that the controller works stable within the whole range.
The stable value range is dependent on the characteristic of the load.

7.4.2. Current controller

Identifier	Register address [HEX] and range
P gain	**#H5143** Value range[1]: 0 ... 32767
I gain	**#H5144** Value range[1]: 0 ... 32767
D gain	**#H5153** Value range[1]: 0 ... 32767
T1 gain	**#H5152** Value range[1]: 0 ... 32767
FFwd gain	**#H514D** Value range[1]: 0 ... 32767
P adaptive gain	**#H515F** Value range[1]: 0...16384
I adaptive gain	**#H5160** Value range[1]: 0...16384

Tab. 80[1]

[1] Given are the maximum values.
This does not imply that the controller works stable within the whole range.
The stable value range is dependent on the characteristic of the load.

7.4.3. Power controller

Identifier	Register address [HEX] and range
P gain	**#H5145** Value range[1]: 0 ... 32767
I gain	**#H5146** Value range[1]: 0 ... 32767

Tab. 81

[1] Given are the maximum values.
This does not imply that the controller works stable within the whole range.
The stable value range is dependent on the characteristic of the load.