



Cloud Scales

CloudScale Communication Protocol (CSCP)

English manual v.1.0 (2026-1)

INDEX

1.	Structure of commands and responses.....	4
1.1.	Command structure.....	4
1.2.	Response format	5
1.2.1.	Response with weight value	5
1.2.2.	Response without weight value.....	5
1.2.3.	Error message	6
1.2.4.	General error message	6
2.	Commands	7
2.1.	@ – cancel command execution.....	9
2.2.	ADC – read ADC points	9
2.3.	C – cancel command execution	9
2.4.	D – show text on display.....	9
2.5.	DW – display weight	10
2.6.	E01 – current error code and description	10
2.7.	I0 – get list of implemented commands	11
2.8.	I1 – query implemented levels and versions	12
2.9.	I2 – device model and scale capacity.....	12
2.10.	I3 – hardware and software versions	13
2.11.	I4 – serial number	13
2.12.	I5 – Firmware checksum.....	13
2.13.	T – tare.....	14
2.14.	TA – query or set tare value	15
2.15.	TAC – clear tare value	16
2.16.	TI – tare immediately.....	16
2.17.	S – get stable weight.....	17
2.18.	SI – get weight immediately	17
2.19.	SIR – start continuous transmission of net weight	18
2.20.	SIX1 – get all weighing data	19
2.21.	SX – get STABLE GROSS, NET, TARE	20
2.22.	SXI – get GROSS, NET, TARE IMMEDIATELY	21
2.23.	SXIR – REPEATED GROSS, NET, TARE	21
2.24.	SNS – get or set active scale (channel)	22
2.25.	UPD -update interval of continuous transmission.....	22
2.26.	QASI1 – query all scale information	23
2.27.	FT – get/set filter	24

2.28.	FC – enable/disable high resolution filter.....	24
2.29.	RST – reset scale	24
3.	Scale and applications parameters.....	25
3.1.	RSD – get all parameters	25
3.2.	<parameter>=? – get one parameter value	25
3.3.	SSD – set parameter(s)	26
3.4.	IS – initialize scale	26
3.5.	GCAL – get calibration data	26
3.6.	CAL – SCALE calibration	27
3.7.	DCAL – DELETE CALIBRATION	28
3.8.	Metrological data – scale parameters	28
4.	Applications.....	30
4.1.	M25 – get available applications	30
4.2.	M26 – query current application & set active application	30
4.3.	PMI – get target and tolerance.....	31
4.4.	Common application parameters.....	32
5.	Counting application	33
5.1.	Counting application parameters	33
5.2.	PW – query or set piece weight.....	33
5.3.	REF – calculate average piece weight.....	34
5.4.	PCS – get number of pieces	34
5.5.	PCSX – get number of pieces and weighing data	35
5.6.	PMC – set target and tolerance in counting.....	36
6.	Check weighing (+/-) application.....	38
6.1.	PMW – set target and tolerances or high/low limit	38
6.2.	CWX – get tolerance status and weighing data	40
7.	Input and outputs.....	41
7.1.	Inputs.....	41
7.1.1.	SIF – get or set input configuration parameters	41
7.1.2.	SIN – get status of input(s).....	42
7.1.3.	DIC – get counter value.....	43
7.1.4.	SIC – set counter value.....	43
7.1.5.	CIC – clear counter value	44
7.2.	Outputs.....	44
7.2.1.	SOT - get or set output state.....	44
7.2.2.	SOF - get or set output function	46
7.2.3.	IIO - initialize inputs and outputs.....	47

1. STRUCTURE OF COMMANDS AND RESPONSES

Commands for CSCP are composed of ASCII characters. Commands are composed of parameters and values.

␣	Symbol represent space.
<MandatoryParam>	Triangle brackets indicate that this is mandatory command parameter.
[OptionalParam]	Square brackets indicate that this is optional parameter and can be omitted.
CRLF	Each command sent must have string terminator CR – carriage return (ASCII 13) and LF – line feed (ASCII 10). Also each response will have string terminator CRLF.
↓	Data sent to scale.
↑	Data received from scale.

Please note that commands and parameters value are case sensitive.

1.1. COMMAND STRUCTURE

General command structure:

Command	[␣]	[Parameters...]	CR	LF
1-n characters		0-n characters		

Examples:

Command without parameters:

↓	S	Request stable weight.
↑	S_S_____100.00_g	Scale response with stable weight of 100 g

Command with parameters:

↓	TA_100_g	Set tare value to 100 g.
↑	TA_A_____100.00_g	Scale response with settled tare value.

1.2. RESPONSE FORMAT

Scale always respond to command with answer which can be:

- Response with weight value
- Response without weight value
- Error message

Note: Weight values are always returned as 10 characters with leading spaces.

1.2.1. RESPONSE WITH WEIGHT VALUE

General response with weight value:

Command ID		Status		Weight value		Unit	CR	LF
1-n characters		1 character		10 characters		1-n characters		

Example:

↓	S	Request stable weight.
↑	S_S_____100.00_g	Scale response with stable weight of 100 g

1.2.2. RESPONSE WITHOUT WEIGHT VALUE

General response without weight value:

Command ID		Status		Parameters	CR	LF
1-n characters		1 character		0-n characters		

Example:

↓	I4	Request scale serial number.
↑	I4 A "ABC2KL "	Scale response with serial number in ""

1.2.3. ERROR MESSAGE

In general error message has the following structure:

Command ID	␣	Status	CR	LF
1-n characters		1 character		

Status	Description
+	Scale gross value or parameter value exceed scale maximum range (scale maximum + 9 divisions).
-	Scale gross value or parameter value is under scale zero range (more than 20 divisions in minus).
I	Internal error – scale executing another command, command timeout etc.
L	Logical error – for example tare value exceed scale maximum.

Example:

↓	S	Request stable weight.
↑	S_I	Scale did not stabilise in defined timeout window (default 5 seconds).
↓	SI	Request weight immediately.
↑	SI_+	Scale gross value exceed scale maximum.

1.2.4. GENERAL ERROR MESSAGE

In case you send to scale non-existing command to scale you will receive ES response.

↓	S0	Non-existing command
↑	ES	Syntax error or non-existing command received.

2. COMMANDS

In this chapter you will find detailed description of each command together with examples.

Command	Description
@	Terminates current command execution
ADC	Get current ADC points
C	Cancel all commands
CAL	Calibrate point.
CIC	Clear counter value
CWX	Get tolerance status and weighing data in check weighing mode
D	Write text to display
DCAL	Delete calibration point.
DIC	Get counter input value
DW	Switch to weight
E01	Status of error and description if any
FC	Enabled/disable high resolution filter
FT	Set filter value
GCAL	Get calibration data.
I0	List of implemented commands
I1	Level information and versions
I2	Scale model and capacity
I3	Hardware and software version
I4	Serial number
I5	Firmware checksum
IS	Initialize scale
IIO	Initialize digital inputs and outputs
M25	Get list of available applications
M26	Get or set application
PCS	Get number of pieces
PCSX	Get number of pieces and weighing data
PMC	Set target and tolerance in counting application
PMI	Get target and tolerances
PMW	Set target and tolerances in check weighing mode
QAS11	Query all scale information
RSD	Read scale data – get all parameters
REF	Calculate average piece weight
RST	Reset scale
S	Get stable weight

SI	Get weight immediately regardless of stability
SIC	Set counter input value
SIF	Get or set input configuration parameters
SIN	Get input status
SIR	Start continuous transmission of net weight regardless of stability status
SX	Get current stable gross, net and tare
SXI	Get current gross, net and tare value immediately
SXIR	Get current gross, net and tare value immediately and repeat (continuous transmission)
SIX1	Get all weighing data stability, gross, net, tare, range etc. with one command
SNS	Get or set current scale (channel)
SOF	Get or set output function
SOT	Get or set output state
SSD	Set scale data - parameters
T	Tare the scale
TA	Query or set preset tare value
TAC	Clear tare weight value
TI	Tare scale immediately regardless of stability
Z	Zero the scale
ZI	Zero the scale immediately regardless of stability status

2.1. @ – CANCEL COMMAND EXECUTION

Command	
@	Command cancel current command on scale.
Responses	
I4_A_<"SerialNumber">	Canceling current command execution has been started.

2.2. ADC – READ ADC POINTS

Command	
ADC	Request for current ADC value.
Responses	
ADC_A_866585	Current ADC value is 866585

2.3. C – CANCEL COMMAND EXECUTION

Command	
C	Command cancel current command on scale.
Responses	
C_A	Canceling current command execution has been started.

2.4. D – SHOW TEXT ON DISPLAY

Command	
D_<"Text">	Display Text on the scale display.
Responses	
D_A	Command executed successfully .
D_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).
D_L	Command received but not executable (syntax error or scale without display).

2.5. DW – DISPLAY WEIGHT

Command	
DW	Switch main display to show weight.
Responses	
DW_A	Command executed successfully .
DW_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).

2.6. E01 – CURRENT ERROR CODE AND DESCRIPTION

Command	
E01	Query current scale state.
Responses	
E01_<ErrorCode>_<Description>	Current error code and description.
E01_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).

Error code	Description
1	No error
96	No calibration.
95	No metrological data (max1 and d1 are not set).
94	ADC error (no load cell, sense error, etc.).
93	Sense error – reference lines error detected (not connected, short cut, etc.).

2.7. I0 – GET LIST OF IMPLEMENTED COMMANDS

Command	
I0	Query all available commands.
Responses	
I0_B_0_"I1" I0_B_0_"I2" ... I0_B_<Level>_"Command">	List of all available commands in format I0 B <Level> <"Command">
I0_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).

Parameters	Description
<Level>	0 – basic commands without parameters 1 – commands with mandatory/optional parameters 2 – extended commands 3 – application specific commands
<"Command">	CSCP command

2.8. I1 – QUERY IMPLEMENTED LEVELS AND VERSIONS

Command	
I1	Query of CSCP levels.
Responses	
I1_A_<"Levels">_<"V0">_["V1"]_["V2"]_["V3"]	Available levels and corresponding versions.
I1_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).
Examples of responses	
I1_A_"012"_2.30_2.22_2.33"	Level 0-2 are implemented and corresponding versions are provided.

Parameters	Description
<Level>	0 – basic commands without parameters 1 – commands with mandatory/optional parameters 2 – extended commands 3 – application specific commands
<"V0">...<"V3">	CSCP corresponding versions.

2.9. I2 – DEVICE MODEL AND SCALE CAPACITY

Command	
I2	Query of model name and capacity.
Responses	
I2_A_<"ModelName Capacity Unit">	Device mode, capacity and unit returned.
I2_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).
Examples of responses	
I2_A_"CS-NEUTRON 1000.0 g"	Model name CS-NEUTRON, capacity 1000 g.

2.10. I3 – HARDWARE AND SOFTWARE VERSIONS

Command	
I3	Query hardware and software versions.
Responses	
I3_A_<"HardwareVersion FirmwareVersion CSCPVersion">	Hardware versions represent PCB version. Firmware version is version of scale firmware version. CSCP is CloudScale Communication Protocol version.
I3_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).
Examples of responses	
I3_A_"1 1.0.0 1.0.0"	Hardware version 1, Firmware 1.0.0., CSCP version 1.0.0.

2.11. I4 – SERIAL NUMBER

Command	
I4	Query scale serial number.
Responses	
I4_A_<"SerialNumber">	Serial number of scale.
I4_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).
Examples of responses	
I4_A_"N7W2KL"	Hardware version 1, Firmware 1.0.0., CSCP version 1.0.0.

2.12. I5 – FIRMWARE CHECKSUM

Command	
I5	Query firmware check sum.
Responses	
I5_A_<"Checksum">	Firmware checksum.
I5_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).
Examples of responses	
I5 A "357940E...F723"	Firmware checksum is 357940E...F723

2.13. T – TARE

This command will execute scale taring as soon as next stable weight is detected.

Command	
T	Set current gross as tare value when weight is stable.
Responses	
T_S_<TareValue>_<Unit>	Tare executed successfully. Command returns tare value.
T_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command, or timeout occurred as scale did not reach stability in specified command window).
T_+	Tare value exceeds the maximum capacity.
T_-	Tare value is less than 0.
Examples of responses	
T_S_____100.00_g	Scale response with stable tare value 100 g
T_S_____1152.05_kg	Scale response with stable tare value 1152.05 kg

2.14. TA – QUERY OR SET TARE VALUE

This command will get currently set tare value or set tare value.

Command		
TA		Query current tare value.
TA_<TareValue>_<Unit>		Preset tare value.
Responses		
T_A_<TareValue>_<Unit>		Tare executed successfully with stable scale .
T_I		Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command, or timeout occurred as scale did not reach stability in specified command window).
T_L		Command received but not executable (syntax error or tare value over max or less than 0, wrong unit etc.)
Examples		
↓	TA	Query current tare value
↑	T_A_____100.00_g	Current tare value is 100 g
↓	TA_____150.11_g	Set tare value to 150.11 g
↑	T_A_____150.11_g	Response from scale that tare value is set to 150.11 g

Note:

Tare value will be automatically rounded to range division. For instance, if scale has division of 0.5 g and you send T_A_____150.11_g it will set 150 g as tare value which will be given in response. If you send T_A_____150.75_g it will set 151 g.

2.15. TAC – CLEAR TARE VALUE

This command will delete current tare value regardless of tare type.

Command	
TAC	Clear tare value
Responses	
TAC_A	Tare value has been cleared (set to 0).
TAC_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command).

2.16. TI – TARE IMMEDIATELY

This command will execute scale taring regardless of scale stability.

Command	
TI	Set current gross as tare value regardless of stability.
Responses	
T_S_<TareValue>_<Unit>	Tare executed successfully with stable scale .
T_D_<TareValue>_<Unit>	Tare executed successfully with unstable scale .
T_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command, or timeout occurred as scale did not reach stability in specified command window).
T_+	Tare value exceeds the maximum capacity.
T_-	Tare value is less than 0.
Examples of responses	
T_S_100.00_g	Scale response with stable tare value 100 g
T_D_1152.05_kg	Scale response with unstable tare value 1152.05 kg

2.17. S – GET STABLE WEIGHT

Returns the current **net weight only when the scale is stable**.

If the scale is not stable when the command is received, the scale **waits until stability is achieved or until the command timeout expires**.

If stability is not reached within the configured **command timeout window**, a **timeout error** is returned.

Command	
S	Returns the net weight only when stable; waits for stability up to the command timeout.
Responses	
S_S_<NetWeight>_<Unit>	Scale responded with stable net weight value.
S_I	Command received but cannot be executed at this time or timeout occurred (the scale is busy processing another task or command, or timeout occurred as scale did not reach stability in specified command window).
S_+	Scale is overloaded (weight exceeds the maximum capacity).
S_-	Scale is underloaded (weight is below the minimum measurable range).
Examples of responses	
S_S_____100.00_g	Scale response with stable net weight 100 g
S_S_____1152.05_kg	Scale response with stable net weight 1152.05 kg

2.18. SI – GET WEIGHT IMMEDIATELY

SI command returns the current **net weight** immediately, **without waiting for scale stability**.

The response reflects the scale status at the moment the command is processed.

Command	
SI	Commands returns current net weight regardless of scale stability status. This means that weight is returned immediately.
Responses	
S_S_<NetWeight>_<Unit>	Scale responded with stable net weight value.
S_D_<NetWeight>_<Unit>	Scale responded with unstable net weight value.
S_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
S_+	Scale is overloaded (weight exceeds the maximum capacity).
S_-	Scale is underloaded (weight is below the minimum measurable range).
Examples of responses	
S_S_____100.00_g	Scale response with stable net weight 100 g
S_D_____1152.05_kg	Scale response with unstable net weight 1152.05 kg

2.19. SIR – START CONTINUOUS TRANSMISSION OF NET WEIGHT

SIR command returns the current **net weight** immediately, **without waiting for scale stability**. As soon as next weight is processed scale sends new weighing data.

Command	
SIR	Command to start continuous transmission of stability indicator and net weight.
Responses	
S_S_<NetWeight>_<Unit>	Scale data with stable net weight value.
S_D_<NetWeight>_<Unit>	Scale data with unstable net weight value.
S_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
S_+	Scale is overloaded (weight exceeds the maximum capacity).
S_-	Scale is underloaded (weight is below the minimum measurable range).
Example of continuous transmission	
↓	SIR
↑	S_S_100.00_g
↑	S_D_102.04_g
↑	S_D_104.52_g
↑	S_D_105.02_g
↑	S_S_105.02_g

Note:

To **cancel continuous transmission** send command @, C, SI, S.

2.20. SIX1 – GET ALL WEIGHING DATA

SIX1 command returns the **all current weighing data** such as **stability, tare type, gross weight, net weight, tare value** immediately, **without waiting for scale stability**.

Command	
SIX1	Query all current scale data
Responses	
SIX1_A_<Stability>_0_<ZeroRange>_N_R_0_0_0_<Range>_<TareType>_<Gross>_<Net>_<Tare>_<Unit>	Scale data with stable net weight value.
SIX1_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
Example communication	
↓	SIX1
↑	SIX1_S_0_Z_N_R_0_0_0_1_N_____0.0_____0.0_____0.0_g
↓	SIX1
↑	SIX1_S_0_Z_N_R_0_0_0_1_N_____1000.0_____1000.0_____0.0_g
↓	TA_____100.0_g
↑	TA_A_____100.0_g
↓	SIX1
↑	SIX1_S_0_Z_N_R_0_0_0_1_N_____1000.0_____900.0_____100.0_g
↓	SIX1
↑	SIX1_D_0_Z_N_R_0_0_0_1_N_____1042.0_____942.0_____100.0_g

Parameters	Description
<Stability>	S – stable weight D – unstable weight (dynamic) + - overload - - underload
<ZeroRange>	Z – if gross +/- ¼ d N – gross not in the range of +/- ¼ d
<Range>	1...3 range in which current gross value is
<TareType>	N – no tare M – measured tare (user pressed tare button or sent T command) P – preset tare (tare entered for example using TA command)
<Gross>	Gross weight
<Net>	Net weight
<Tare>	Tare
<Unit>	Unit

2.21. SX – GET STABLE GROSS, NET, TARE

SX command returns the **current weighing data - stability gross weight, net weight, tare value** when weight is **stable**.

Command	
SX	Query current scale data
Responses	
SX_<status>_<Gross>_<Unit>_<Net>_<Unit>_<Tare>_<Unit>	Scale data with stable net weight value.
SX_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
SX_+	Scale is overloaded (weight exceeds the maximum capacity).
SX_-	Scale is underloaded (weight is below the minimum measurable range).
Example communication	
↓	SX
↑	SX_S_.....0.0_g.....0.0_g.....0.0_g
↓	SX
↑	SX_S_.....1045.0_g.....945.0_g.....100.0_g

2.22. SXI – GET GROSS, NET, TARE IMMEDIATELY

SXI is simplified SIX1 command that returns the **current weighing data - stability gross weight, net weight, tare value** immediately, **without waiting for scale stability**.

Command	
SXI	Query current scale data
Responses	
SX_<status>_<Gross>_<Unit>_<Net>_<Unit>_<Tare>_<Unit>	Scale data with stable net weight value.
SXI_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
SX_+	Scale is overloaded (weight exceeds the maximum capacity).
SX_-	Scale is underloaded (weight is below the minimum measurable range).
Example communication	
↓	SXI
↑	SX_S_.....0.0_g.....0.0_g.....0.0_g
↓	SXI
↑	SX_D_.....1042.0_g.....942.0_g.....100.0_g

2.23. SXIR – REPEATED GROSS, NET, TARE

SXIR will enable continuous sending of SXI command. To disable continuous sending of SXI command send C or @ command.

Command	
SXIR	Query current scale data
Responses	
SX_<status>_<Gross>_<Unit>_<Net>_<Unit>_<Tare>_<Unit>	Scale data with stable net weight value.
Example communication	
↑	SX_S_.....0.0_g.....0.0_g.....0.0_g
↑	SX_D_.....1022.0_g.....922.0_g.....100.0_g
↑	SX_D_.....1042.0_g.....942.0_g.....100.0_g

2.24. SNS – GET OR SET ACTIVE SCALE (CHANNEL)

Command	
SNS_[scale]	Command to get or set active channel/scale. If scale parameter is omitted then you will receive in response current channel otherwise you will receive success status of changing active scale. Scale parameter can be 1-4.
Responses	
SNS_1	Current scale (channel) is 1.
SNS_A	Scale change has been successful.
SNS_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
SNS_L	Command received but not executable (syntax error or wrong channel – it must be in range 1-4).
Example of continuous transmission	
↓	SNS
↑	S_1
↓	SNS_2
↑	SNS_A

2.25. UPD -UPDATE INTERVAL OF CONTINUOS TRANSMISSION

UPD command is used to set delay between transmissions in commands like SIR, SXIR that activate continuous mode.

Note: this is the delay between sending data after they have been processed. Since data are sent to the scale communication interface this means that there will be additional delay especially if using wireless connection like WIFI or Bluetooth. Same delay is guaranteed only on CPU level and therefore data will not come always with the same delay depending on interface used. Wired interfaces tend to have more constant delay then wireless ones.

Command	
UPD	Get current delay between transmissions.
UPD_[delay_ms]	Set delay between transmission in milliseconds.
Responses	
UPA_A_[delay_ms]	Current scale (channel) is 1.
Example of continuous transmission	
↓	UPD
↑	UPD_A_100

↓	UPD_A_50
↑	UPD_A_50

2.26. QASI1 – QUERY ALL SCALE INFORMATION

QASI1 command reads all the **current scale data** such as **channel, range, stability, gross weight, net weight, tare value, high resolution, adc point, millivolts** etc. immediately, **without waiting for scale stability**.

Command	
QASI1	Query all scale information
Responses	
QASI_<Channel>_<Range>_<Stability>_<Gross>_<Net>_<Tare>_<HighResolution>_<ADCPoints>_<Milivolts>_<Unit>	Scale data with stable net weight value.
QASI_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
Example communication	
↓	QASI1
↑	QASI1_1_1_3_____0.0_____0.0_____0.0_____0.002_____8517221_____0.5989_g
↓	QASI1
↑	QASI1_1_1_3_____1000.00_____100.00_____900.00_____999.9999_____8698107_____1.4411_g

Parameters	Description
<Channel>	Current scale channel 1-4
<Range>	1...3 range in which current gross value is
<Stability>	S – stable weight D – unstable weight (dynamic) + - overload - - underload
<Gross>	Gross weight
<Net>	Net weight
<Tare>	Tare
<HighResolution>	High resolution gross weigh with two additional decimals
<ADCPoints>	Signal from load cell expressed in points
<Milivolts>	Signal from load cell expressed in millivolts
<Unit>	Unit

2.27. FT – GET/SET FILTER

FT command is used to get or set filter on currently active scale. If you want to set filter for different scale either switch to different scale by using SNS command or use SSD command and parameter nFT.

Command	
FT=?	Query current filter value.
FT=<value>	Set filter value – range 1-1023 – 1 no filter, 1023 maximum filter
Responses	
FT_A_<value>	Response with filter value

2.28. FC – ENABLE/DISABLE HIGH RESOLUTION FILTER

FC command is used to enable/disable high resolution filter on currently active scale. If you want to set filter for different scale either switch to different scale by using SNS command or use SSD command and parameter nFC.

Command	
FC=?	Query current high resolution filter value.
FC=<value>	Enable (1) or disable (0) high resolution filter.
Responses	
FC_A_<value>	Response with high resolution filter value.

2.29. RST – RESET SCALE

Command	
RST	Reset scale.
Responses	
I4_A_"serialNo"	Response with high resolution filter value.

3. SCALE AND APPLICATIONS PARAMETERS

Command RSD and SSD give you possibility to read all scale and applications parameters.

Each parameter has unique name and to which a corresponding value can be set. List of all available parameters are listed below in table with description, default value and range value.

Parameters are always in key=value pairs. For example 1D1=0.1 where 1D1 is parameter and 0.1 is it's value. Parameters are separated with semicolon (;).

3.1. RSD – GET ALL PARAMETERS

Get all scale and applications parameters. This is very usefully not just to get current values but also you can store them as backup of your scale configuration. To restore values you just simple replace first character R with S and send it to scale.

Command	
RSD	Query all parameters.
Response	
RSD;<param1=value>;<param2=value>;...	All parameters.
Example	
↓	RSD
↑	RSD;1D1=0.1;1D2=;1D3=;1M1=1000;1M2=;1M3=;1U=g;1ZT;...

3.2. <PARAMETER>=? – GET ONE PARAMETER VALUE

If you need to retrieve only one parameter value you can use this command.

Command	
<parameter>=?	Query parameter value.
Response	
<parameter>=<value>	Parameter and it's value.
Example	
↓	1D1=?
↑	1D1=0.1

3.3. SSD – SET PARAMETER(S)

This command is used to set scale and applications parameters. Please note that after changing parameter values in storage they are not applied immediately. To apply new values you must either reboot scale or execute command IS to apply new settings.

Command	
SSD; [parameter1=value]; [parameter2=value2];...	Set on or multiple parameters values.
Response	
SSD; [parameter1=@1]; [parameter2=@1];...	Successfully set application parameter(s). SSD command does not return parameter value but it return success status which can be: @1 – parameter changed successfully @-1 – parameter does not exists @-2 – not allowed ASCII value or character @-3 – not valid scale parameter

3.4. IS – INITIALIZE SCALE

Command	
IS	Initialize scale with new parameters
Response	
IS_A	Scale initialization finished. Note: scale initialization will take few seconds to finish.

3.5. GCAL – GET CALIBRATION DATA

Command	
GCAL=<scaleNo>	Get calibration data of selected scale. If <scaleNo> parameter is set to 0 you will receive calibration data of all scales.
Response	
GCAL=<sZP=value>, <sxW=value>, <sxA=value>, <sxF=value>,...	Calibration parameters: <ul style="list-style-type: none"> • s – is scale number • x – is consecutive number • ZP – zero points • W – load • A – ADC points • F - factor

3.6. CAL – SCALE CALIBRATION

Command	
CAL=<point>,<load>,[requireStable]	Calibrate selected point with selected value.
Response	
CAL=@<status>,<channel>,<repNo>,<point>,<load>,<adcPoints>,<factor>	Scale initialization finished.
Example of calibrations	
↓	CAL=0,0
↑	CAL=@S,1,0,0,0.00,0,0 CAL=@C,1,1,0,0.00,8517076.76,0 CAL=@C,1,2,0,0.00,8517075.26,0 CAL=@F,1,2,0,0.00,8517075.98,0
↓	CAL=1,500
↑	CAL=@S,1,0,1,500.00,0,0 CAL=@C,1,1,1,500.00,8607489.80,0 CAL=@C,1,2,1,500.00,8607488.14,0 CAL=@F,1,2,1,500.00,8607487.56,180.82401

Parameters	Description
<point>	Calibration point number 0-8 (0 is zero calibration).
<load>	Weight that is placed on the scale load receptor. In case point is zero load must be 0. Load must be less or equal to scale maximum.
[requireStable]	By default stable weight during calibration process is mandatory otherwise calibration will not finish and error will be returned. In case you would like to store calibration even, if stability is not achieved you must set [requireStable] to 0 and calibration will be stored regardless of weight stability.
<status>	Status represent current status of calibration with @ followed by status character which can be: S - calibration started C – calibration of point in progress F – calibration finished E – calibration error (unstable weight, ADC points less than previous point etc.)
<channel>	Channel represent scale on which calibration is being executed – 1-4.
<repNo>	Calibration requires stable weight therefore there will be up to 10 repetitions. If stability is not reached in 10 repetitions calibration will fail unless require stable parameter was set to 0. If stability is achieved before 10 th repetition calibration will finish sooner.

<adcPoints>	Signal from load cell expressed in points
<factor>	Factor that was calculated for this point and provided load.

3.7. DCAL – DELETE CALIBRATION

Command	
DCAL=<point>	Delete calibration point 0-8. If point is 99 it will delete all calibration points.
Response	
DCAL=A	Calibration point deleted successfully.
DCAL=E	Deletion of calibration point failed (wrong point).

3.8. METROLOGICAL DATA – SCALE PARAMETERS

Metrological and calibration data are stored in parameters described in the table below.

Note: you can manually send any parameter to scale via any connection interface to which you set scale. Important is that you follow proper structure and start the command with SSD. See example below the table.

Parameters	Descriptions	Default	Data	Example
sN	Channel/scale name – max 8 characters		char	
sDx	Division – D1-D3		Float	
sMx	Maximum – M1-M3		Float	
sRx	No. of ranges – 1-3		UInt8	
sxA	Point x ADC points		UInt32	
sxW	Point x weight		Float	
sxF	Point x factor		Float	
sU	Unit	kg	Char[2]	
sFT	Filter: 1-1023 (1- no filter, 1023 max)	0	UInt16	
sFC	High resolution filter: 0-disabled, 1-enabled	0	UInt8	
sDS	Division for test stability	0	UInt8	
sZT	Zero tracking – 0-99 (0 disabled)	0	UInt8	
sZR	Zero range in percentage - 0 means zeroing is disabled	0	UInt8	
sOZ	On zero -executed zeroing in range – percentage 0-50% - 0 means on zero is disabled	0	UInt8	

sCW	Command timeout window – value how long to wait for stability in zeroing, taring, getting stable weight etc. in seconds 1,2,3,...	5	Uint16	
OSW	Stability window – how long the weight must be stable in order to set status to stable - in milliseconds	500	Uint16	SW=1000

s – means scale number 1-4

x – is consecutive number 1-n

Example:

SSD;1D1=1;1M1=1000;1U=kg

Will set up scale parameters of scale 1 with division of first range to 1 kg and capacity of first range to 10000 kg. Unit will be set to kg.

4. APPLICATIONS

Applications have specific functionalities and parameters related to them. They are described in the following chapters.

Application common commands are:

- M25 – query available applications
- M26 – query current application or set specific application

4.1. M25 – GET AVAILABLE APPLICATIONS

Command	
M25	Command to get request available applications.
Response	
M25_B_0_ "Weighing"	List of available applications.
M25_B_1_ "Counting"	
M25_B_2_ "Check weighing"	
M25_B_3_ "Dosing"	
...	

4.2. M26 – QUERY CURRENT APPLICATION & SET ACTIVE APPLICATION

Command	
M26	Query currently active application
M26_1	Set application 1 – Counting.
Response	
M26_A_1	Current active application is 1 - Counting
M26_A	Successfully set application.
M26_L	Command received but not executable (syntax error or wrong or non-existing application number).

4.3. PMI – GET TARGET AND TOLERANCE

PMI command can be used in counting and check weighing applications.

Command	
PMI	Command to get checkweighing parameters.
Responses	
PMC_<Application>_<TolMode>_<Unit>_<V1>_<V2>_<V3>_<T_Tare>_<A_APW>_<NO "ArticleID">_<NA "ArticleDescription">	Command executed successfully. Please note: parameters in [] are optional.
PMC_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
PMC_L	Command received but not executable (syntax error or wrong parameter value).
Examples	
↓	PMI
↑	PMI_W_ABS_g_10.00_20.00_T_0.0_NO_ "3185" _NA_ "Screw M5, inox metric"
↓	PMI
↑	PMI_C_REL_g_100.0_0.0_2.0_T_115.3_A_0.500000

Parameters	Description
<Application>	W – weighing C – counting
<TolMode>	ABS – high and low limit REL – relative limits based on target and tolerance plus and minus PER – percentage mode (only available in checkweighing) where tolerances are calculate based on percentage of target
<Unit>	Unit
<V1>	Target if TolMode=ABS and Low limit if TolMode=REL
<V2>	Tolerance minus if TolMode=ABS/PER and High limit if TolMode=REL
[V3]	Tolerance plus if TolMode=ABS/PER
[T_Tare]	Tare value
[A_APW]	Average piece weight
[NO "ArticledID"]	Article id
[NA "ArticleDescription"]	Article description

4.4. COMMON APPLICATION PARAMETERS

Parameter	Description	Default value	Range
nLT	LT represent lock/unlocked tare. By default tare is locked which means that also after scale is unloaded tare will stayed stored in scale memory. If you set LT=0 tare will be automatically cleared once you unload the scale (return in zero range is required).	1	0-1

*** n represents number of scale 1-4**

5. COUNTING APPLICATION

Counting application is calculating no. of pieces based on the provided or referenced Average Piece Weight (APW).

5.1. COUNTING APPLICATION PARAMETERS

Parameters related to counting application are:

Parameter	Description	Default value	Range
nACD	Counting decimals – when calculating number of pieces you can set to use also decimal values not just integer values. Counting decimals are used also in check mode.	0	0...10
nACAD	APW decimals – set on how many decimals APW is calculated.	No. of decimals in first division + 2	0...10

* n represents number of scale 1-4

To change or get counting application parameters you must use SAP and GAP command.

5.2. PW – QUERY OR SET PIECE WEIGHT

Command	
PW_[apw]_[unit]	Command to get or set piece weight value. If apw and unit param's are omitted then current piece weight value is returned.
Responses	
PW_A_<apw>_<unit>	Command executed successfully. Response includes also average piece weight value.
PW_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
PW_L	Command received but not executable (syntax error or wrong parameter value).
Examples	
↓	PW
↑	PW_A_0.005001_g
↓	PW_A_0.005_g
↑	PW_A_0.005000_g

5.3. REF – CALCULATE AVERAGE PIECE WEIGHT

Command	
REF_<noOfPieces>	Command to execute calculation of APW. You must provide no. of pieces currently placed on the scale load receptor. Please note that APW calculation can take up to 10 seconds. Net weight can not be equal to zero.
Responses	
REF_A_<apw>_<unit>	Command executed successfully. Response includes also average piece weight value.
REF_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
REF_L	Command received but not executable (syntax error or wrong parameter value, net weight is zero, unstable conditions – could not perform reference).
Examples	
↓	REF_100000
↑	REF_A_0.004997_g

5.4. PCS – GET NUMBER OF PIECES

Command	
PCS	Command to get current number of pieces.
Responses	
PCS_<Status>_<noOfPieces>	Response with current number of pieces and scale status.
PCS_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).

Parameters	Description
<Status>	S – stable weight D – unstable weight (dynamic) + - overload - - underload
<noOfPieces>	Current number of pieces.

5.5. PCSX – GET NUMBER OF PIECES AND WEIGHING DATA

Command	
PCSX	Command to get current number of pieces and weighing data.
Responses	
PCS<Status>_<TolStatus>_<noOfPieces> _<APW>_<Gross>_<Net>_<Tare>_ <Unit>_[NO "ArticleID"] _[NA "ArticleDescription"]	Response with current number of pieces, scale and tolerance status and all weighing data. If set also article ID and description are returned. See command PMC.
PCSX_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
Examples	
↓	PCSX
↑	PCSX_S_NA_____100030___0.004997_____499.85_____49 9.85_____0.00_g
↓	PCSX
↑	PCSX_S_OK_____100000___0.004997_____499.70_____49 9.70_____0.00_g

Parameters	Description
<Status>	S – stable weight D – unstable weight (dynamic) + - overload - - underload
<TolStatus>	NA – not applicable – target/tolerances are not set OK – in tolerance LO – under tolerance HI – over tolerance
<Gross>	Gross weight
<Net>	Net weight
<Tare>	Tare
<APW>	Current average piece weight
<Unit>	Unit

5.6. PMC – SET TARGET AND TOLERANCE IN COUNTING

With PMC command you can activate check program where you can set target and tolerances (relative mode) or lower and higher limit (absolute mode) to get tolerance status based on the number of present pieces on scale. To get current values use PMI command.

Negative check counting

You can use negative check counting in relative mode where you set target in minus. In that case please note that upper and lower limit will be swapped in order to correctly determine tolerance status.

Command		
<pre>PMC_REL<Target>_<TolMinus>_<TolPlus>_[Unit]_ [A_APW]_[T_Tare]_[NO "ArticleID"] _[NA "ArticleDescription"] PMC_ABS_<LowLimit>_<HighLimit>_[Unit]_[A_APW] _[T_Tare]_[NO "ArticleID"]_[NA "ArticleDescription"]</pre>	<p>Command to set checkweighing parameters. There are two modes available:</p> <p>REL – relative mode with target and tolerance minus/plus</p> <p>ABS – absolute mode with low and high limit.</p> <p>If set PCSX command will return tolerance status according to parameters values.</p> <p>If output function are set it will activate/deactivate outputs according to set function and according to parameters.</p> <p>Please note:</p> <p>Parameters in [] are optional. If you do not provide unit the tare and APW will not be set even, if they are provided.</p>	
Responses		
PMC_A		Command executed successfully.
PMC_I		Command received but cannot be executed at this time (the scale is busy processing another task or command).
PMC_L		Command received but not executable (syntax error, wrong parameter value or counting application not activated – see command M25/M26).
Examples		
↓	PMC_ABS_10_11	Lower limit=10 pcs Upper limit=11 pcs
↑	PMC_A	
↓	PMC_REL_10_0_1_U_kg_A_0.003_T_0.513	Target=10 pcs Tolerance minus=0 pcs

		Tolerance plus=1 pcs APW=0.003 kg Preset tare=0.513 kg
↑	PMC_A	

Parameters	Description
<Target>	Target used in absolute mode. If no. of pieces is equal or higher than lower limit and lower or equal than upper limit tolerance status is OK.
<TolMinus>	Tolerance minus is used to set lower limit and is calculated as <Target>-<TolMinus>. When no. of pieces is lower than low limit tolerance status is LO.
<TolPlus>	Tolerance plus is used to set upper limit and is calculated as <Target>+<TolPlus>. When no. of pieces is higher than upper limit tolerance status is HI.
<LowLimit> <HighLimit>	<LowLimit> Lower limit in relative mode. If no. of pieces is lower than tolerance status is LO. <HighLimit> Upper limit in relative mode. If no. of pieces is higher than upper limit tolerance status is HI. If no. of pieces is equal or higher than lower limit and lower or equal than upper limit tolerance status is OK.
[Unit]	Unit
[A_APW]	Average piece weight – value used to calculate no. of pieces.
[T_Tare]	Preset tare value
[NO "ArticleID"]	Article ID – to be used to determine which article is being counted when using PCSX command.
[NA "ArticleDescription"]	Article description

6. CHECK WEIGHING (+/-) APPLICATION

6.1. PMW – SET TARGET AND TOLERANCES OR HIGH/LOW LIMIT

With PMW command you can activate check weighing program where you can set target and tolerances (relative mode) or lower and higher limit (absolute mode) or provide tolerance in percentage related to target value in unit of measure. To get current values use PMI command.

Negative check weighing

You can use negative check weighing in relative mode where you set target in minus. In that case please note that upper and lower limit will be swapped in order to correctly determine tolerance status.

Command	
<pre>PMW_REL_<Target>_<TolMinus>_<TolPlus>_<Unit>_ [T_Tare]_[NO "ArticleID"]_[NA "ArticleDescription"]</pre>	<p>Command to set checkweighing parameters. There are three modes available:</p> <p>REL – relative mode with target and tolerance minus/plus</p> <p>ABS – absolute mode with low and high limit.</p> <p>PER – percentage mode where you set target in unit and tolerances in percentage.</p> <p>If set PCSX command will return tolerance status according to parameters values.</p> <p>If output function are set it will activate/deactivate outputs according to set function and according to parameters.</p> <p>Please note: parameters in [] are optional while parameters in <> are required.</p>
<pre>PMC_ABS_<LowLimit>_<HighLimit>_<Unit>_[T_Tare]_[NO "ArticleID"]_[NA "ArticleDescription"]</pre>	
<pre>PMW_PERC_<Target>_<TolMinus>_<TolPlus>_<Unit> _[T_Tare]_[NO "ArticleID"]_[NA "ArticleDescription"]</pre>	
Responses	
PMW_A	Command executed successfully.
PMW_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
PMW_L	Command received but not executable (syntax error, wrong parameter value or checkweighing application not activated – see command M25/M26).
Examples	
↓	<pre>PMW_PER_16.3_10.2_10.9_lb</pre> <p>Target=16.3 lb Tolerance minus=10.2% (1,663 lb) Tolerance plus=10.9% (1,777 lb)</p>

↑	PMW_A	
↓	PMW_ABS_10.2_10.4_kg	Lower limit=10.2 kg (under) Upper limit=10.4 kg (over)
↑	PMW_A	
↓	PMW_REL_100_4.1_4.5_kg	Target=100 kg Tolerance minus=4.1 kg (under<95.9 kg) Tolerance plus=4.5 kg (over>104.5 kg)
↑	PMW_A	
↓	PMW_PER_100_1.1_2_kg	Target=100 kg Tolerance minus=1.1 % (under<98.9 kg) Tolerance plus=2% (over>102 kg)
↑	PMW_A	

Parameters	Description
<Target>	Target used in absolute mode. If no. of pieces is equal or higher than lower limit and lower or equal than upper limit tolerance status is OK.
<TolMinus>	Tolerance minus is used to set lower limit and is calculated as <Target>-<TolMinus>. When no. of net is lower than low limit tolerance status is LO.
<TolPlus>	Tolerance plus is used to set upper limit and is calculated as <Target>+<TolPlus>. When net is higher than upper limit tolerance status is HI.
<LowLimit> <HighLimit>	<LowLimit> Lower limit in relative mode. If no. of pieces is lower than tolerance status is LO. <HighLimit> Upper limit in relative mode. If net is higher than upper limit tolerance status is HI. If net is equal or higher than lower limit and lower or equal than upper limit tolerance status is OK.
[Unit]	Unit
[T_Tare]	Preset tare value
[NO "ArticleID"]	Article ID – to be used to determine which article is being checkweighed when using CWX command.
[NA "ArticleDescription"]	Article description - to be used to get article description when using CWX command.

6.2. CWX – GET TOLERANCE STATUS AND WEIGHING DATA

Command	
CWX	Command to get current check weighing status and weighing dat.
Responses	
PCS_<Status>_<TolStatus>_<Gross>_<Net>_<Tare>_<Unit>_[NO "ArticleID"] _[NA "ArticleDescription"]	Response with tolerance status and all weighing data. If set also article ID and description are returned. See command PMW.
CWX_I	Command received but cannot be executed at this time (the scale is busy processing another task or command).
Examples	
↓	CWX
↑	CWX_S_NA_____499.85_____499.85_____0.00_g
↓	CWX
↑	CWX_S_OK_____499.70_____499.70_____0.00_g

Parameters	Description
<Status>	S – stable weight D – unstable weight (dynamic) + - overload - - underload
<TolStatus>	NA – not applicable – target/tolerances are not set OK – in tolerance LO – under tolerance HI – over tolerance
<Gross>	Gross weight
<Net>	Net weight
<Tare>	Tare
<Unit>	Unit

7. INPUT AND OUTPUTS

7.1. INPUTS

In order to use digital inputs you must first configure them as by default all inputs are disabled.

Each digital input has two modes:

- On/off
- Counter

In on/off mode digital input acts as switch. If function is set it will also trigger function execution (e.g. zero scale, tare scale, etc) otherwise you can use it as you wish by monitoring it's status.

In counter mode digital input will increase counter value each time the state of input is high. Counter value is also stored so even after module reboot it is still available. Off-course you have possibilities to set and reset counter value.

7.1.1. SIF – GET OR SET INPUT CONFIGURATION PARAMETERS

SIF command is used to setup input mode which can be on/off (classic input mode) or counter.

Command	
SIF,<inputNo>=99	Command to get configuration of selected input. If <inputNo> is 00 you will get configuration of all inputs separated by semicolon (;).
SIF,<inputNo>=<M_mode>,[D_debounce],[G_glitch],[F_function]	Command to set input configuration. If <inputNo> is 00 you can set configuration of all inputs by providing parameters for each input separated by semicolon (;). First configuration is for first input, second for second etc.
SIF,00=<M_mode>,[D_debounce],[G_glitch],[F_function];[M_mode],[D_debounce],[G_glitch],[F_function];...	
Responses	
RIF,<inputNo>=<M_mode>,<D_debounce>,<G_glitch>,<F_function>	Configuration of one input
RIF,00=<M_mode>,<D_debounce>,<G_glitch>,<F_function>;<M_mode>,<D_debounce>,<G_glitch>,<F_function>;...	Configuration of all inputs.

Examples	
↓	SIF,2=99
↑	RIF,2=M_1,D_5,G_0,F_0
↓	SIF,2=D_20,F_1
↑	RIF,2=M_1,D_20,G_0,F_1

Parameters	Description
<inputNo>	Input number. Depending on the CloudScale you are using it is value from 1...n depending on the number of available inputs. Please note that 00 has special function to retrieve all inputs settings.
<M_mode>	Digital input mode: 0 – disabled 1 – on/off 2 – counter Please note that this parameter is mandatory when changing input settings.
[D_debounce]	Filters switch bounce. Input must stay stable for this time before the state is accepted. To be used only when input mode is on/off. Typical value: 5-100 ms.
[G_glitch]	Ignores short electrical noise pulses. Pulses shorter than this time are not counted. To be used only when input mode is counter. Typical value: 1-20 μs.
[F_function]	Digital input function (only available in on/off mode): 0 – disabled 1 – zero scale 2 – tare scale

7.1.2. SIN – GET STATUS OF INPUT(S)

To use digital input you have to set it's function first to on/off mode.

Command	
SIN, <inputNo>	Command to get status of input.
SIN, 00	If <inputNo> is 00 you will get status of all inputs.
Responses	
RIN, <inputNo>=<iS>	<iS> can be 0 (input not activated) or 1 (input activated).
RIN, 00=<iS1>, <iS2>, ..	If <inputNo> is 00 response includes status of all inputs in ordered by input number from 1 to n. If value is -1 it means that digital input is disabled. If value starts with # it means that this is the value of counter.
Examples	
↓	SIN,2
↑	RIN, 2=1

↓	SIN,00
↑	RIN,00=0,1,1,1,1,0

7.1.3. DIC – GET COUNTER VALUE

Command	
DIC,<inputNo> DIC,00	Command to get current counter value at selected input. If <inputNo> is 00 you will get values of all counters.
Responses	
RIC,<inputNo>=<cv>	<iS> can be 0 (input not activated) or 1 (input activated).
RIC,00=<#cv1>,<#cv2>,. .	If <inputNo> is 00 response includes values of all counters in ordered by input number from 1 to n. Please note that response includes also inputs that are in on/off mode or disabled in this case value is -1 which means that inputs is not in counter mode.
Examples	
↓	DIC,2
↑	RIC, 2=13243

7.1.4. SIC – SET COUNTER VALUE

Command	
SIC,<inputNo>=<value>	Command to set selected counter value.
Responses	
RIC,<inputNo>=<value>	
Examples	
↓	SIC,4=500
↑	RIC, 4=500

7.1.5. CIC – CLEAR COUNTER VALUE

Command	
CIC, <inputNo>=<value>	Command to clear selected counter value.
CIC, 00	Command to clear all counters values.
Responses	
RIC, <inputNo>=<value>	
Examples	
↓	CIC,4
↑	RIC, 4=0
↓	CIC, 00
↑	RIC, 00=-1, -1, -1, #0, -1, -1

7.2. OUTPUTS

Digital outputs enables you to control external peripherals such as lights, buzzers, relays, etc.. You can manually control outputs via commands, or you can set function and it will set output automatically according to selected function.

7.2.1. SOT - GET OR SET OUTPUT STATE

Command	
SOT, <outputNo>=<oS>	Command to get or set output state related to <oS> parameter: <ul style="list-style-type: none"> - 99 - you will receive output state, - 0 – you will disable/deactivate output so the state is low - 1 – you will enable/activate output so the state is high <p>In case you set output no. to 00 you can set all outputs at once. Please note that order is from 1...n</p> <p>This command will return you all outputs states.</p>
SOT, 00=<oS1>, <oS2>, <oS3>, ...	
SOT, 00=99	
Responses	

ROT, <outputNo>=<oS>	Response for one output.
ROT, 00=<oS1>, <oS2>, <oS3>, ...	Response for multiple outputs.
Examples	
↓	SOT, 4=99
↑	ROT, 4=0
↓	SOT, 4=1
↑	ROT, 4=1
↓	SOT, 00
↑	ROT, 00=0, 1, 0, 1, 0, 0
↓	SOT, 00=0, 0
↑	ROT, 00=0, 0, 0, 1, 0, 0

Parameters	Description
<outputNo>	Output number. Depending on the CloudScale you are using it is value from 1...n depending on the number of available outputs. Please note that 00 has special function to retrieve or set all outputs states.
<oS>	Output state: <ul style="list-style-type: none"> - 0 – low (disabled) - 1 – high (enabled) - 99 – special value to get current state of output

7.2.2. SOF - GET OR SET OUTPUT FUNCTION

Command	
SOF, <outputNo>=<oF>	Command to get or set output function related to <oF> parameter: <ul style="list-style-type: none"> - 99 - you will receive current output function, - 0...98 – you will set output function
SOF, 00=<oF1>, <oF2>, <oF3>, ...	In case you set output no. to 00 you can set all outputs functions at once. Please note that order is from 1...n
SOT, 00=99	This command will return you all outputs functions.
Responses	
ROF, <outputNo>=<oF>	Response for one output.
ROF, 00=<oF1>, <oF2>, <oF3>, ...	Response for multiple outputs.
Examples	
↓	SOF, 4=99
↑	ROF, 4=0
↓	SOF, 4=1
↑	ROF, 4=1
↓	SOF, 00
↑	ROF, 00=2, 3, 0, 1, 0, 0
↓	SOF, 00=1, 2, 3
↑	ROF, 00=1, 2, 3, 1, 0, 0

Parameters	Description
<outputNo>	Output number. Depending on the CloudScale you are using it is value from 1...n depending on the number of available outputs. Please note that 00 has special function to retrieve or set all outputs states.
<oF>	Output function: <ul style="list-style-type: none"> - 0 – disabled - 1 – tolerance ok - 2 – tolerance low - 3 – tolerance high - 99 – special value to get current state of output

7.2.3. IIO - INITIALIZE INPUTS AND OUTPUTS

Command	
IIO	After changing inputs and outputs you must perform initialization or scale restart to apply changes.
Responses	
IIO_A	