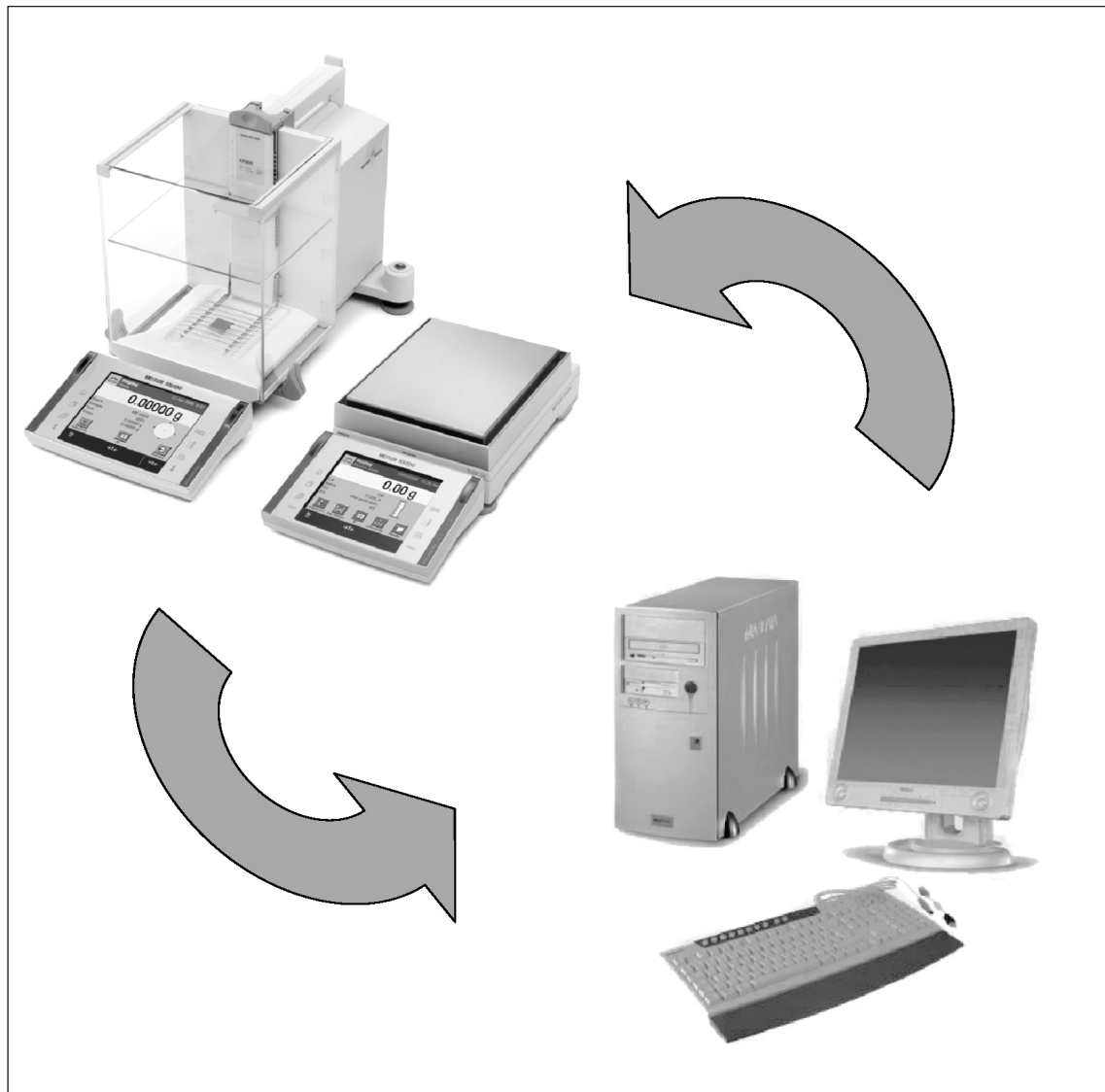


Standard Interface Command Set for Excellence Balances



METTLER TOLEDO

Table of Contents

1	Quick Reference: Commands Listed Alphabetically	3
2	Introduction	5
2.1	Command Formats	6
2.1.1	Language Conventions	6
2.1.2	Balance Response Formats	7
2.1.3	Tips for Programmer	11
3	Commands and Balance Responses	14
4	Commands Listed by Tasks.....	204
4.1	Weighing, Zeroing, Taring	204
4.1.1	Setup	204
4.2	General Setup	205
4.2.1	Printer	205
4.2.2	Applications.....	205
4.2.3	Options	206
4.3	Adjustment/Test.....	206
4.4	Balance Information	206
4.5	Fast Data Communication	207
5	Example: Formula weighing application	208
6	What if...?	209

1 Quick Reference: Commands Listed Alphabetically

@ – Cancel	Level 0	14
A01 – Percent Weighing: Query/Set Reference in %	Level 3	15
A02 – Query/Set IDs of the Weighing Application	Level 3	16
A03 – Query/Set ID-Names of the Weighing Application	Level 3	17
A06 – Dynamic Weighing: Query/Set Dynamic Behavior	Level 3	19
A07 – Dynamic Weighing: Query/Set Start Mode	Level 3	20
A08 – Dynamic Weighing: Query/Set AutoTare	Level 3	21
A09 – Dynamic Weighing: Query/Set Data Acquisition	Level 3	22
A10 – Weighing: Query/Set Nominal, +Tolerance, -Tolerance	Level 3	23
A30 – Internal loads	Level 3	24
B00 – FastHost: List Commands	FastHost	25
B01 – FastHost: Send Individual Value	FastHost	26
B02 – FastHost: Start/Stop Continuous Mode	FastHost	27
B03 – FastHost: Query Time Interval of Value Counter	FastHost	28
B04 – FastHost: Query/Set Format Specification	FastHost	29
B05 – FastHost: Query/Set Stability Criteria	FastHost	32
B06 – FastHost: Query/Set Downsampling Factor	FastHost	33
B07 – FastHost: Query/Set Switch-On Mode	FastHost	34
B08 – FastHost: Zero with FastHost Stability Criteria	FastHost	35
C0 – Query/Set Adjustment Settings	Level 2	36
C1 – Start Adjustment According to Current Settings	Level 2	38
C2 – Start Adjustment with External Weight	Level 2	39
C3 – Start Adjustment with Internal Weight	Level 2	40
COM – Query/Set Communication Parameters of the built-in Serial Interface	Level 2	41
COPT – Query/Set Interface Option Parameters	Level 2	43
D – Display: Write Text to Display	Level 1	49
DAT – Query/Set Date	Level 2	50
DW – Display: Show Weight	Level 1	51
I0 – List All Implemented MT-SICS Commands	Level 0	52
I1 – Query MT-SICS Level and MT-SICS Versions	Level 0	53
I2 – Query Balance Data (Type and Capacity)	Level 0	54
I3 – Query Balance Software Version and Type Definition Number	Level 0	55
I4 – Query Serial Number	Level 0	56
I5 – Query SW-Identification Number	Level 0	57
I10 – Query Balance ID	Level 2	58
I11 – Query Balance Type	Level 2	59
I14 – Query Balance Information	Level 2	60
I15 – Query "Power On" Time	Level 2	62
I16 – Query Date of Next Service	Level 2	63
I17 – MinWeigh: Query Date of Next Test	Level 2	64
I18 – MinWeigh: Query Method	Level 2	65
I19 – MinWeigh: Query Limits	Level 2	66
I20 – Query MinWeigh Parameter	Level 2	67
I28 – Query Level Sensor	Level 2	68
K – Keys: Set Configuration	Level 1	69
M01 – Query/Set Weighing Mode	Level 2	71
M02 – Query/Set Environment	Level 2	72
M03 – Query/Set AutoZero	Level 2	73
M04 – Query/Set SmartSens Functions	Level 2	74
M05 – Query User Profile	Level 2	76
M06 – Query/Set Current User Profile	Level 2	77
M07 – Query/Set AutoDoor	Level 2	78
M08 – Display: Query/Set Brightness	Level 2	79
M09 – Display: Query/Set Contrast	Level 2	80
M10 – Beeper: Query/Set Display Mode	Level 2	81
M11 – Beeper: Query/Set Volume	Level 2	82
M12 – Beeper: Beep	Level 2	83
M13 – Query/Set Touch Function	Level 2	84
M14 – List Available Languages	Level 2	85
M15 – Query/Set Language	Level 2	87
M16 – Query/Set Standby Mode	Level 2	88
M17 – ProFACT: Query/Set single Time Criteria	Level 2	89
M18 – ProFACT/FACT: Query/Set Temperature Criterion	Level 2	91
M19 – Query/Set Adjustment Weight	Level 2	92
M20 – Query/Set Test Weight	Level 2	93

M21 – Query/Set Unit	Level 2	94
M22 – Query/Set Custom Unit Definitions	Level 2	96
M23 – Query/Set Readability, 1d/xd	Level 2	97
M24 – Query/Set Print Key Function	Level 2	98
M25 – List Applications	Level 2	99
M26 – Query/Set Current Application	Level 2	100
M27 – Query Adjustment History	Level 2	101
M29 – Query/Set Value Release	Level 2	102
M32 – ProFACT: Query/Set Time Criteria	Level 2	103
M33 – ProFACT: Query/Set Weekday	Level 2	104
M34 – MinWeigh: Query/Set Method	Level 2	105
M35 – Query/Set Zeroing Mode at Start-Up	Level 2	106
M36 – Query/Set LevelControl Setup	Level 2	107
M37 – Query/Set Draft Shield Door Opening Width	Level 2	108
M39 – SmartTrac: Query/Set Graphic	Level 2	109
M40 – Query/Set Inner Draft Shield Door Opening Width	Level 2	110
M43 – Custom Unit activate/deactivate	Level 2	111
M47 – Read and write frequently changed test weight settings	Level 2	112
M48 – Read and write infrequently changed test weight settings	Level 2	114
M50 – GWP Test Sequence	Level 2	117
M51 – GWP Method EC	Level 2	121
M52 – GWP RP1 and RPT1	Level 2	123
M53 – GWP Service	Level 2	125
M54 – GWP Weight Tolerances	Level 2	128
M55 – GWP Task	Level 2	132
M56 – GWP Task State	Level 2	136
M57 – GWP System State	Level 2	139
M58 – GWP History Export	Level 2	141
M64 – GWP Test Sequence Version	Level 2	158
M66 – Certified test weight settings	Level 2	160
M95 – Read and write test / adjust weight index to be used for TST2 and C2 commands	Level 2	162
P100 – Printer Interface: Send Text	Level 2	165
P101 – Printer Interface: Send Weight Value	Level 2	166
P102 – Printer Interface: Send Weight Value Immediately	Level 2	167
PW – Piece Counting: Query/Set Piece Weight	Level 3	168
PWR – Power On/Off	Level 2	169
S – Send Stable Weight Value	Level 0	170
SI – Send Weight Value Immediately	Level 0	171
SIR – Send Weight Value Immediately and Repeat	Level 0	172
SIRU – Send Weight Value with Currently Displayed Unit Immediately and Repeat	Level 2	173
SIS – Send NetWeight Value with Actual Unit and Weighing Status	Level 2	174
SIU – Send Weight Value with Currently Displayed Unit Immediately	Level 2	176
SIUM – Send Weight Value with Currently Displayed Unit and MinWeigh Information Immediately	Level 2	177
SMO – Dynamic Weighing: Cancel all SMx Commands	Level 3	178
SM1 – Dynamic Weighing: Start Immediately and Send the Result	Level 3	179
SM2 – Dynamic Weighing: Start After a Minimum Load Is Exceeded and Send Result	Level 3	180
SM3 – Dynamic Weighing: Start After a Minimum Load Is Exceeded, Send Result and Repeat	Level 3	181
SM4 – Dynamic Weighing: Query/Set Time Interval	Level 3	182
SNR – Send Stable Weight Value and Repeat on Weight Change	Level 2	183
SNRU – Send Stable Weight Value with Currently Displayed Unit and Repeat on Weight Change	Level 2	184
SR – Send Weight Value on Weight Change (Send and Repeat)	Level 1	186
SRU – Send Stable Weight Value with Currently Displayed Unit on Weight Change	Level 2	187
ST – Send Stable Weight Value on Pressing F (Transfer) Key	Level 2	188
SU – Send Stable Weight Value with Currently Displayed Unit	Level 2	189
SUM – Send Stable Weight Value With Currently Displayed Unit and MinWeigh Information	Level 2	190
T – Tare	Level 1	191
TA – Query/Preset Tare Weight Value	Level 1	192
TAC – Clear Tare Value	Level 1	193
TI – Tare Immediately	Level 1	194
TIM – Query/Set Time	Level 2	195
TST0 – Query/Set Test Function Settings	Level 2	196
TST1 – Start Test Function According to Current Settings	Level 2	197
TST2 – Start Test Function with External Weight	Level 2	198
TST3 – Start Test Function with Internal Weight	Level 2	199
UPD – Query/Set Update Rate of the Host Interface	Level 2	200
WS – Query/Set Position of Draft Shield Doors	Level 2	201
Z – Zero	Level 0	202
ZI – Zero Immediately	Level 0	203

2 Introduction

In weight measurements the demands on the readability and maximum capacity of balances and scales range from less than one microgram up to several hundred tons. To meet these and other requirements, METTLER TOLEDO offers an extensive range of balances and scales. Many of the balances and scales used have to be capable of integration in a complex computer or data acquisition system. To enable you to integrate balances in your system in a simple manner and utilize their capabilities to the full, most balance functions are also available as appropriate commands via the data interface.

Standardization of the commands

All new METTLER TOLEDO balances launched on the market support the standardized command set "METTLER TOLEDO Standard Interface Command Set" (MT-SICS), which is divided into 4 levels, depending on the functionality of the balance:

- MT-SICS level 0: Basic Command set, e.g. weighing cell.
- MT-SICS level 1: Elementary command set, i.e. balances without integrated applications.
- MT-SICS level 2: Extended command set specific for a balance family, e.g. for the Excellence balance line.
- MT-SICS level 3: Application-specific command set, e.g. MT-SICS for piece counting or percent weighing.

A particular distinguishing feature of this concept is that the commands combined in MT-SICS level 0 and 1 are identical for all balances. Both the simplest weighing balance and a fully expanded weighing work station recognize the commands of MT-SICS level 0 and 1.

What do the commands of MT-SICS level 0 and 1 offer?

You can use the commands of MT-SICS level 0 and 1 to perform the following operations via the interface:

- Request weighing results.
- Tare the balance and preset the tare weight.
- Zero the balance.
- Identify MT-SICS implementation.
- Identify the balance.
- Cancel Commands.
- Control the display.
- Control the keys for operation of the balance.

The commands of MT-SICS level 2 and 3

You can naturally use the data interface to exploit all functions available with your current balance or application. These additional functions are collected in the commands of MT-SICS level 2 and 3. When creating your software application, please note that whereas the commands of MT-SICS level 2 have been specially tailored to your balance family, the commands of MT-SICS level 3 apply to your weighing application and can not be supported by every METTLER TOLEDO balance. In the enclosure with this manual, you can see what commands are supported by your balance and application.

Version number of the MT-SICS

Each level of the MT-SICS has its own version number which can be requested with the command I1 from level 0.

This manual describes:

- MT-SICS level 0, version 2.3x
- MT-SICS level 1, version 2.2x
- MT-SICS level 2 for Excellence balances
- MT-SICS level 3 for Excellence balances

You can use the command **I1** via the interface to request the MT-SICS level and MT-SICS versions implemented on your balance. Please make sure that the versions implemented on your balance agree with those listed above.

Additional documentation on data interface

Settings of the interface such as baud rate, number of data bits, parity, handshake protocols and connector pin assignment are described in the operating instructions of the optional interface and the peripheral instrument or cable in question.

Data Exchange with the Balance

Each command received by the balance via the data interface is acknowledged by a response of the balance to the transmitter. Commands and balance responses are data strings with a fixed format, and will be described in detail in chapter 3.

The existing commands that are available can be called up using the IO command. Please take note that some of the commands work only via the built-in RS-232 interface.

2.1 Command Formats

Commands sent to the balance comprise one or more characters of the ASCII character set. Here, the following must be noted:

- Enter commands only in uppercase. Nevertheless, units have to be capitalized properly.
- The possible parameters of the command must be separated from one another and from the command name by a space (ASCII 32 dec., in this manual represented as `_`).
- The possible input for "text" is a sequence of characters of the 8-bit ASCII character set from 32 dec. to 255 dec.
- Each command must be closed by CR LF (ASCII 13 dec., 10 dec.).
The characters CR LF, which can be inputted using the Enter or Return key of most entry keypads, are not listed in this description, but it is essential they be included for communication with the balance.

2.1.1 Language Conventions

Throughout this manual, the following conventions are used for command and balance response syntax:

- Triangle brackets (`< >`) indicate that you must specify a value for the enclosed parameter. The parameter's name is written in italic typeface. The brackets are not sent with the command string.
- Square brackets (`[]`) indicate that the enclosed expression is optional and can be omitted. The brackets are not sent with the command string.
- Intervals or ranges are represented using the "dot-dot" notation (`a..b`) indicating the set of numbers from *a* to *b* including *a* and *b*.
- In Examples, commands sent to the balance are set as \downarrow ; balance responses from the balance are set as \uparrow .

Example

Command to balance which writes Hello into the balance display:

\downarrow	D_ "Hello"	The quotation marks " " must be inserted in the entry
\uparrow	D_A	Command executed successfully

The command terminator CR LF is not shown.

2.1.2 Balance Response Formats

All responses sent by the balance to the transmitter to acknowledge the received command have one of the following formats:

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

2.1.2.1 Format of Balance Responses with Weight Value

A general description of the balance response with weight value is the following.

<ID>	␣	<Status>	␣	<WeightValue>	␣	<Unit>	CR	LF
1-2 characters		1 character		10 characters		1-5 characters		

Name	Type	Values	Meaning
<ID>	String		Balance response identification, refers to the invoking command
␣	Blank		Space (ASCII 32 dec.)
<Status>	Character	S	S table weight value
		M	Stable weight value, but below minimal weight (SIUM and SUM only)
		D	Unstable ("D" for D ynamic) weight value
		N	Unstable weight value, below minimal weight (SIUM and SUM only)
<WeightValue>	Float		Weighing result; shown as a number with 10 characters (after a blank/space!), including decimal point, and minus sign (–) directly in front of the first digit if the value is negative. The weight value appears right aligned. Preceding zeros are not shown except for the zero to the left of the decimal point. With METTLER TOLEDO DeltaRange balances, outside the fine range the last decimal place is shown as a space.
<Unit>	String		Weight unit as actually set under host unit
CR	Byte		Carriage return (ASCII 13 dec.)
LF	Byte		Line feed (ASCII 10 dec.)

Examples

Balance response with stable weight value of 0.256 g:

↓	S	Request a stable weight value
↑	S␣S␣␣␣␣␣␣␣␣␣␣␣␣0.256␣g	

Balance response with stable weight value outside the fine range:

↓	S	Request a stable weight value
↑	S␣S␣␣␣␣4875.2␣g	

2.1.2.2 Format of Balance Responses Without Weight Value

A general description of the balance response without weight value is the following:

<ID>	␣	<Status>	␣	Parameters...	CR	LF
1-5 characters		1 character				

Name	Type	Values	Meaning
<ID>	String		Balance response identification, refers to the invoking command
␣	Blank		Space (ASCII 32 dec.)
<Status>	Character	A	Command executed successfully
		B	Command not yet terminated, additional responses following
Parameters...			Command-dependent response code
CR	Byte		Carriage return (ASCII 13 dec.)
LF	Byte		Line feed (ASCII 10 dec.)

Examples

Set the update rate to 20 weight values per second:

↓	UPD_20	
↑	UPD_A	Command executed successfully

Query the actual update rate:

↓	UPD	
↑	UPD_A_18.311	Update rate is set to 18.311 values per second

2.1.2.3 Error Messages

Command-specific Error Messages

A general description of the balance response without weight value is the following:

<ID>	␣	<Status>	CR	LF
1-5 characters		1 character		

Name	Type	Values	Meaning
<ID>	String		Balance response identification, refers to the invoking command
␣	Blank		Space (ASCII 32 dec.)
<Status>	Character	+	Balance is in overload range (weighing range exceeded)
		-	Balance is in underload range (e.g. weighing pan is not in place)
		L	Logical error (e.g. parameter not allowed)
		I	Internal error (e.g. Balance not ready yet)
CR	Byte		Carriage return (ASCII 13 dec.)
LF	Byte		Line feed (ASCII 10 dec.)

Examples

Trial to set the update rate to 20 weight values per second:

↓	UPD_290	Update rate accidentally set to 290
↑	UPD_L	Command not executed successfully; parameters is outside valid range

Response while balance is in overload range:

↓	SI	Request a weight value immediately
↑	S_+	overload; no weight value available

General Error Messages

There are three different error messages:

<ID>	CR	LF
2 characters		

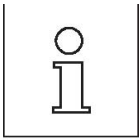
Name	Type	Values	Meaning
<ID>	String	ES	Syntax error: The balance has not recognized the received command
		ET	Transmission error: The balance has received a "faulty" command, e.g. owing to a parity error or interface break
		EL	Logical error: The balance can not execute the received command
CR	Byte		Carriage return (ASCII 13 dec.)
LF	Byte		Line feed (ASCII 10 dec.)

Example

Trial to set the update rate to 20 weight values per second:

↓	upd_20	UPD accidentally written in lowercase
↑	ES	Syntax error; upd not recognized as a command

2.1.3 Tips for Programmer



This reference manual covers the MT-SICS commands for Excellence and Excellence Plus balances. As the balances can differ based on model and software version, not all the MT-SICS level 2 and 3 commands are usable on every model. We therefore recommend using the **IO** command to get an overview of all commands that are supported by a particular balance. Commands that are listed with the **IO** command, but not described in this booklet, are not available for the user.

Planning the use of MT-SICS Commands

Investigations of various applications have shown that the vast majority of all system solutions can be handled with the commands of MT-SICS level 0 and 1. This means for you: if you restrict yourself to the commands of MT-SICS level 0 and 1, you can expand your system with additional balances from METTLER TOLEDO without having to change your application programs.

Setup

Use the same setup during configuration and later use: If you intend to use the bridge without the terminal, the configuration has to be done without terminal as well. Due to the system's architecture, the storage behavioral of configurations is different whether the terminal is attached to the bridge or not: With a terminal attached, configuration is stored in the terminal's memory; without a terminal attached, the bridge's memory is used. Removing a terminal after configuration means to remove the configuration and activation the bridge's (default) configuration. Adding a terminal after configuration means overriding the configuration with the one stored within the terminal.

Command and Balance Response

You can improve the dependability of your application software by having your program evaluate the response of the balance to a command. The response is the acknowledgement that the balance has received the command.

Cancel

To be able to start from a determined state, when establishing the communication between balance and system, you should send a cancel command (@) to the balance. When the balance or system is switched on or off, faulty characters can be received or sent.

Parameter Values After Switching Balance On/Off

The commands of the standard command are saved on the permanent memory of the balance. This means that all values changed via the interface are saved when the balance is switched off.

Several Commands in Succession

If several commands are sent in succession without waiting for the corresponding balance responses, it is possible that the balance confuses the sequence of command processing or ignores entire commands.

METTLER TOLEDO DeltaRange Balances

If the fine range of DeltaRange balances has been exceeded at the time of transmission, the balance sends a weight value as balance response in which the tenth character is a space.

Update Rate and Timeout

The update rate for repeated commands and the duration of the timeout (time-limit function) depend on the balance type; see technical data of the balance in question.

Carriage Return, Line Feed

Depending on the platform, CR LF is not just a "new line" (Java: "newLine()" or C/C++ "\n"):

Platform	'New Line'
DOS/Windows	CR LF
Macintosh	CR
Unix	LF

Nevertheless, all commands have to be closed by a CR LF (dec: 13, 10; hex: 0D, 0A).

Quotation Marks ""

Quotation marks included in the command must always be entered. If a quotation mark is located within the string, it may be escaped by a backslash (\):

↓	D_"place 4\"filter!"	
↑	D_A	Balance display: place 4" filter!

Weight Unit of Weight Value – Host Unit

It is always essential to consider the weight unit that is to be used to display weighing results. Depending on where the results are output, the balances offer the possibility of selecting a particular unit (**M21**). This enables the displayed unit and info unit to be shown on the terminal. Host unit is used to output the weighing results via an interface (host) on the basis of MT-SICS commands. The weight values and the displayed unit can only be output by means of the "S*U" commands.

Digit [d]

A digit refers to the smallest numerical increment a balance can display – this is also referred to as the balance's readability. E.g. a XP205 has five decimal places; its digit is 0.01 mg. The digit is sometimes used as a generic unit.

Behavior with Automatic Door Control

Various commands implicitly support motor-operated draft shield doors (where used). This means that motor-operated draft shield doors can be activated whenever a stable weight value has to be recorded in the shortest possible time. Activating automatic door control results in the following:

- Condition of doors saved
- Doors closed
- Commands executed
- Previously saved condition of doors reestablished.

The most important applications for the behavior described above are:

- Zeroing, taring
- Adjustment and testing procedures, as well as FACT
- S* commands
- Print commands
- Others

Other definitions

- The command is executed even when automatic door control is not activated.
- Even if the doors cannot be closed (clamping protection), the command is executed and the previously saved condition of the doors reestablished.
- If a command in progress is interrupted (Timeout, Cancel, etc.), the "old" door condition is reestablished.
- If the "old" door condition cannot be reestablished, the doors either remain where they are or are closed (this may follow a few attempts to attain the old condition).

Binary Coded Multiple Selections

Some parameters that allow multiple selections are binary coded: Each possible selection is represented by one bit, the corresponding parameter equals to the decimal interpretation.

Selection 8	Selection 7	Selection 6	Selection 5	Selection 4	Selection 3	Selection 2	Selection 1	Parameter
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0..255
$2^7 = 256$	$2^6 = 128$	$2^5 = 64$	$2^4 = 32$	$2^3 = 16$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$	$\sum_{i=1}^8 Selection_i \cdot 2^{i-1}$

Responses may easily be interpreted by converting the decimal number to binary again.

FastHost Commands

This section deals with a special set of commands that are implemented in the weighing platform independently of the other MT-SICS commands. Because the interface is addressed directly, these commands support an extremely high update rate. The weight values that are output can also be individually formatted and assigned additional information such as stability and time.

Essentially, these commands are suitable for special applications with embedded systems and in-process applications such as dispensing.

If implemented, FastHost commands will demonstrate the following properties:

- They will not appear in the command listing of the command **I0** (request with **B00**)
- In the case of a weighing platform with a connected terminal, the commands will only be available on the permanent RS interface via a host.
- In the case of a stand-alone platform without any additional options, the commands will be available on the permanent RS232 interface along with the usual host commands.
- In the case of a stand-alone bridge with an additional option, the commands will only be available on the optional interface along with the usual host commands.
- Most commands will be permanently saved. To establish a default condition, the following command sequence is recommended:
 - B02_0
 - B04_"S_%S%_W:10%_U%"
 - B05_0
 - B06_1
 - B07_0

3 Commands and Balance Responses

@ – Cancel

Level 0

Description

@ can be used to achieve the same effect as disconnecting and reconnecting the power supply, which empties the volatile memories. The purpose of this command is to initiate a command sequence.

Syntax

Command

@	Resets the balance to the condition found after switching on, but without a zero setting being performed.
---	---

Balance Response

I4_A_ "<SNR>"	Serial number is emitted; the balance is ready for operation.
---------------	---



Comments

- All commands awaiting responses are cancelled.
- Key control is set to the default setting [K_1](#).
- Unlike other balances, the tare memory is not reset to zero for XP/XS/XA balances.
- The cancel command is always executed.
- If the balance is on standby, it is switched on.
- The emitted serial number corresponds to the serial number of the terminal (if one is present). See also [I4](#).

Example

↓	@	Cancel
↑	I4_A_ "1114350697"	Balance is "reset", its serial number is 1114350697

A01 – Percent Weighing: Query/Set Reference in %**Level 3****Description**

Use this command to set or query the reference value for percent weighing. For querying to take place, a reference value must have been saved beforehand (A01 or function key  or .

Syntax**Commands**

A01	Query of the reference for the percent weighing application.
A01_<Reference>	Set the reference for the percent weighing application.

Balance Responses

A01_A_<Reference>	Reference for the percent weighing application is set.
A01_B A01_A	Start to set the reference (waiting for stable weight) Command understood and executed successfully.
A01_I	Command understood but currently not executable.
A01_L	Command understood but not executable (e.g. percent weighing application is not active or parameter is incorrect) or no reference value present.
A01_E	Setting reference aborted (not stable, over- or underload, abortkey,...).

Parameter

Name	Type	Values	Meaning
<Reference>	Float	(0)..100	Reference for the percent weighing application in %; must be greater than zero.

Comments

- This command can only be used when the application "percent weighing" is started. For details on available applications and how to activate them see also [M25](#) and [M26](#).
- Use the **S*U** commands for percent weighing. Otherwise, the results will be displayed in the set unit unless the **host unit** is changed to % using [M21](#).

Example

↓	A01_100.00	Set the reference for percent weighing to 100.00%
↑	A01_B	Reference is set, waiting for stable weight
↑	A01_A	Command executed successfully

A02 – Query/Set IDs of the Weighing Application

Level 3

Description

You can use A02 to assign a value or text to sample IDs, or query the current value or text.

Syntax

Commands

A02	Query the IDs of the Weighing Application.
A02_<No>_<ID>	Set the ID of the Weighing Application.

Balance Responses

A02_B_<No>_<ID>	Query the ID's of the Weighing Application.
A02_B_<No>_<ID>	
A02_A_<No>_<ID>	
A02_A	Command understood and executed successfully.
A02_I	Command understood but currently not executable.
A02_L	Command understood but not executable (e.g. weighing application is not active or parameter is incorrect).

Parameters

Name	Type	Values	Meaning
<No>	Integer	1..n	Number of Weighing ID
<ID>	String		String of Weighing ID (max. 24 char.)

Comment

- This command only applies to the "Weighing" application. For details on available applications and how to activate them see also [M25](#) and [M26](#).

Example

↓	A02	Query the IDs of the Weighing Application
↑	A02_B_1_"12345"	ID1 is 12345
↑	A02_B_2_"67890"	ID2 is 67890
↑	A02_A_3_" "	No ID3 is set
↓	A02_1_"98765"	Set the ID1 to 98765
↑	A02_A	ID1 is set

A03 – Query/Set ID-Names of the Weighing Application

Level 3

Description

You can use A03 to assign an individual name to sample IDs, or query the current name.

Syntax

Commands

Query:

A03	Query the ID-Names of the Weighing Application.
A03_<No>_<ID>"	Set the ID-Name of the Weighing Application.

Balance Responses

A03_B_<No>_<ID>" A03_B_<No>_<ID>" A03_A_<No>_<ID>"	All existing ID-Names of the Weighing Application.
A03_A	Command understood and executed successfully.
A03_I	Command understood but currently not executable.
A03_L	Command understood but not executable (e.g. weighing application is not active or parameter is incorrect).

Query:

A03_<No>	Query of specific ID.
----------	-----------------------

Balance Responses

A03_A_<No>_<ID>"	Specific ID-Name of the Weighing Application.
------------------	---

Parameters

Name	Type	Values	Meaning
<No>	Integer	1..n	Number of Weighing ID-Name
<ID>	String		String of Weighing ID-Name (max. 24 char.)

Comment

- This command applies to the "Weighing" application. For details on available applications and how to activate them see also [M25](#) and [M26](#).

Examples

↓	A03	Query the IDs-Name of the Weighing Application
↑	A03_B_1_ "Batch"	Name of ID1 is "Batch"
	A03_B_2_ "Lot "	Name of ID2 is "Lot"
	A03_A_3_ " "	No ID3-Name is set

↓	A03_2	Query the second ID-Name of the Weighing Application
↑	A03_A_2_ "Lot "	Name of second ID is "Lot"

↓	A03_1_ "Batch"	Set the ID1-Name to "Batch"
↑	A03_A	Name of ID1 is set

A06 – Dynamic Weighing: Query/Set Dynamic Behavior

Level 3

Description

You can use A06 to set the dynamic behavior of the weighing material, or query the current setting.

Syntax

Commands

A06	Query of the weighing filter for the dynamic weighing application.
A06_<DynamicBehavior>	Set the weighing filter for the dynamic weighing application.

Balance Responses

A06_A_<DynamicBehavior>	Set the dynamic behavior.
A06_A	Command understood and executed successfully.
A06_I	Command understood but currently not executable, application is not active.
A06_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect).

Parameter

Name	Type	Values	Meaning
<DynamicBehavior>	Integer	1	Stable, suitable for relatively stable weighing objects
		2	Standard, suitable for normal weighing objects
		3	Unstable, suitable for unstable weighing objects

Comment

- This command can only be used when the application "dynamic weighing" is started. For details on available applications and how to activate them see also [M25](#) and [M26](#).

Example

↓	A06_2	Set the dynamic behavior to "standard"
↑	A06_A	Dynamic weighing filter is set

A07 – Dynamic Weighing: Query/Set Start Mode

Level 3

Description

Settings for automatically starting dynamic weighing. A measuring cycle can be started automatically if a specified minimum weight is exceeded.

Syntax

Commands

A07	Query of the start mode for the dynamic weighing application.
A07_<StartMode>	Set the AutoStart for the dynamic weighing application.

Balance Responses

A07_A_<StartMode>	Set the dynamic weighing start mode.
A07_A	Command understood and executed successfully.
A07_I	Command understood but currently not executable.
A07_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect).

Parameter

Name	Type	Values	Meaning
<StartMode>	Integer	0	Step procedure
		1	AutoStart on
		2	Quick procedure

Comments

- This command can only be used when the application "dynamic weighing" is started. For details on available applications and how to activate them see also [M25](#) and [M26](#).
- If the "AutoTare" function is also activated, the display is automatically reset to zero on completion of each successful weighing operation.
- Refer to Weighing Manual, Chapter "Dynamic Weighing".

Example

↓	A07_1	Set the dynamic weighing with AutoStart
↑	A07_A	Dynamic weighing AutoStart is activated

A08 – Dynamic Weighing: Query/Set AutoTare

Level 3

Description

Querying or setting the AutoTare function. As soon as you have removed the weighing sample in the "Dynamic Weighing" application, the display is automatically reset to zero and the balance is then ready for the next sample to be weighed.

Syntax

Commands

A08	Query of the AutoTare function for the dynamic weighing application.
A08_<AutoTare>	Set the AutoTare function for the dynamic weighing application.

Balance Responses

A08_A_<AutoTare>	Set the dynamic weighing AutoTare function.
A08_A	Command understood and executed successfully.
A08_I	Command understood but currently not executable.
A08_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect).

Parameter

Name	Type	Values	Meaning
<AutoTare>	Integer	0	AutoTare off, no zeroing after weighing
		1	AutoTare on, automatic zeroing after weighing

Comments

- This command can only be used when the application "dynamic weighing" is started. For details on available applications and how to activate them see also [M25](#) and [M26](#).
- The Dynamic Weighing AutoTare function has the same effect as zeroing. No tare values are imported into the tare memory.

Example

↓	A08_1	Set the dynamic weighing with AutoTare
↑	A08_A	Dynamic weighing AutoTare is activated

A09 – Dynamic Weighing: Query/Set Data Acquisition

Level 3

Description

Use A09 to set the type of data acquisition or query the current setting. Depending on the behavior of the weighing sample, it can also be used to optimize the speed at which the results are transferred.

Syntax

Commands

A09	Query of the data acquisition for the dynamic weighing application.
A09_<DataAcq>	Set the data acquisition for the dynamic weighing application.

Balance Responses

A09_A_<DataAcq>	Set the data acquisition for the dynamic weighing application.
A09_A	Command understood and executed successfully.
A09_I	Command understood but currently not executable.
A09_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect).

Parameter

Name	Type	Values	Meaning
<DataAcq>	Integer	0	Data acquisition setting: Dynamic behavior
		1	Data acquisition setting: Time interval

Comments

- This command can only be used when the application "dynamic weighing" is started. For details on available applications and how to activate them see also [M25](#) and [M26](#).
- Use [A06](#) to set the filters for "Dynamic behavior" in accordance with the dynamic behavior of the weighing sample.
- The time interval can be set with [SM4](#).
- Refer to Weighing Manual, Chapter "Dynamic Weighing"

Example

↓	A09_1	Set the data acquisition for the dynamic weighing application to "time interval"
↑	A09_A	Data acquisition for the dynamic weighing application has been set

A10 – Weighing: Query/Set Nominal, +Tolerance, -Tolerance

Level 3

Description

You can use A10 to enter the nominal values, inc. +/- tolerances, or query the current values. As soon as you have specified the values, the SmartTrac changes and displays the graphic weighing-in aid.

Syntax

Commands

A10	Query of the nominal value, + tolerance, - tolerance.
A10_<No>_<Value>_<Unit>	Set the nominal value, + tolerance, - tolerance.

Balance Responses

A10_B_0_<Value>_<Unit> A10_B_1_<Value>_<Unit> A10_A_2_<Value>_<Unit>	Query of the nominal value, + tolerance, - tolerance.
A10_A	Command understood and executed successfully.
A10_I	Command understood but currently not executable.
A10_L	Command understood but not executable.

Parameters

Name	Type	Values	Meaning
<No>	Integer	0	Nominal value
		1	+ tolerance
		2	- tolerance
<Value>	Float		Nominal value
<Unit>			Weight unit, % with +/- tolerances possible

Comments

- The values will be output differently depending on the application. For details on available applications and how to activate them see also [M25](#) and [M26](#).
- Specified nominal and tolerance values must be reset manually:
A10_0_0_g
A10_1_2.5_%
A10_2_2.5_%
- As soon as you have specified the values, the SmartTrac switches to the graphic weighing-in aid.
- Weight and percentage values are rounded, as is the case with values entered manually.

Examples

↓	A10	Query of the nominal value, + tolerance, - tolerance
↑	A10_B_0_100.12_g	Current setting is nominal value 100.12 g, + tolerance is 5.25 g and - tolerance is 7.6%
↑	A10_B_1_5.25_g	
↑	A10_A_2_7.6_%	
↓	A10_0_100.12_g	Set the nominal value to 100.12 g
↑	A10_A	Nominal value is set

A30 – Internal loads

Level 3

Description

You can use A30 to request status of internal loads. This command is used to inquire how many internal weights are available in the balance and its status.

Syntax

Command

A30	Query of quantity and status of the internal loads.
-----	---

Balance Responses

A30_A_Qty_Stat	Qty represents number of internal weights present. Stat is status of internal weights.
----------------	--

Parameters

Name	Type	Values	Meaning
<Qty>			Number of internal loads
<Stat>	Integer	0	No load placed
		1	Total load placed
		2	First partial load placed
		3	Second partial load placed
		8	Error
		9	Not determined (not in defined end position)

Comments

- The number of internal loads depends on the balance model.
- This command is similar to XP2300 command.

Examples

↓	A30	Query of quantity and status of the internal loads.
↑	A30_A_1_0	There is only one internal load which is currently not placed.

Control of internal loads

↓	A30_LOAD	Place an internal load Load Internal load Load=0 No load Load=1 Total load Load=2 First partial load Load=3 Second partial load
↑	A10_A	The load is placed.

B00 – FastHost: List Commands

FastHost

Description

Returns all implemented FastHost commands for the connected balance.

Syntax

Command

B00	Query of lists of FastHost commands.
-----	--------------------------------------

Balance Responses

B00_B_ "<FastHostCommand>"	1 st FastHost command.
B00_B_
B00_A_ "<FastHostCommand>"	last FastHost command.
B00_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<FastHostCommand>	String		FastHost command

Example

↓	B00	Query of lists of FastHost commands
↑	B00_B_ "B00"	Command B00 is available
↑	B00_B_ "B01"	Command B01 is available
↑	B00_B_ "B02"	Command B02 is available
↑	B00_B_ "B03"	Command B03 is available
↑	B00_B_ "B04"	Command B04 is available
↑	B00_B_ "B05"	Command B05 is available
↑	B00_B_ "B06"	Command B06 is available
↑	B00_B_ "B07"	Command B07 is available
↑	B00_A_ "B08"	Command B08 is available

B01 – FastHost: Send Individual Value

FastHost

Description

Outputs the data and weight values that have been individually formatted using **B04**. The command behavior is similar to that of **SI**.

Syntax

Command

B01	Send individual FastHost value.
-----	---------------------------------

Balance Responses

<output>	As per format specification (cf B04).
B01_L	Command understood but not executable (incorrect parameter).

Example

↓	B01	Send individual FastHost value
↑	S_123.45_g	As per current format specification (B04 , "%S%W:10%U%") FastHost value is a stable ("S") value of 123.45 g

B02 – FastHost: Start/Stop Continuous Mode

FastHost

Description

Continually outputs the data and weight values that have been individually formatted using [B04](#), and repeats this procedure until the function is switched off.

Syntax

Command

B02_<ContinuousMode>	Start/stop continuous mode.
----------------------	-----------------------------

Balance Responses

<output>	Continuous as per sampling reduction (cf B06) and format specification (cf B04).
B02_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<ContinuousMode>	Integer	0	Stop (Default)
		1	Start

Comment

- Switch off: Must be switched off with B02 0 once the task is complete.

Example

↓	B02_1	Query of Start continuous mode
↑	D_123.45_g	Continuous and as per current format specification
↑	S_123.54_g	(B04_ "%S%_ %W:10%_ %U%") defined

B03 – FastHost: Query Time Interval of Value Counter

FastHost

Description

B03 can be used to query the value-counter time interval. A time axis for the weighing results can be calculated on the basis of this time interval and the counter, while also taking into account the downsampling factor.

The value of the time interval is determined by the transmission speed.

Syntax

Command

B03	Query of time interval of the value counter.
-----	--

Balance Responses

B03_A_<TimeInterval>	Time interval the value counter in seconds.
B03_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<TimeInterval>	Float		Time interval the value counter in seconds

Example

↓	B03	Query of time interval of value counter
↑	B03_A_0.010923	Time interval is 10.923 ms

B04 – FastHost: Query/Set Format Specification

FastHost

Description

B04 can be used to individually format the output for B01 and B02. As well as the structure of the output string, additional information and comments on the weight value can be provided.

Syntax

Commands

B04	Query of the FastHost format specification.
B04_ "<Format>"	Set the FastHost format specification.

Balance Responses

B04_A_ "<Format>"	Format (string).
B04_A	FastHost format specification is set.
B04_L	Command understood but not executable (incorrect parameter).
B04_I	Parameter not permanently saved.

Parameter

Name	Type	Values	Meaning
<Format>	String	Combination of Format specifiers listed in the table below; separated by blanks	Specification of the output format for B01 and B02

Format specifiers	Information	Options
%W[:[l][:p]]%	Weight value (net value)	l Length: Total number of characters of the weight value including decimal point and decimal places (if any). Formatting is right-justified as long as the defined number of places for the output display is sufficient.
%A[:[l][:p]]%	Absolute weight value referring to fabrication zero point	p Decimal places: Number of decimal places for the weight value. Note: The maximum possible number of decimal places is limited to the resolution of the balance (Default). Output is always as full range with rounding (DeltaRange is not supported).
%U%	Unit (as string)	No options, unit is fixed to gram
%S[:s:i]%	Stability information (stable/unstable)	s Indicator for a stable weight value (default S) i Indicator for an unstable weight value (default D). Note: Only 1 character allowed. The stability information is only used with B01 and B02.

%Q[:[l]][:p]]%	<p>Signal width as indicator for the measured value stability in grams.</p> <p>Note: The signal width can be used for a user specific detection stable/ unstable with B05.</p>	<p>l Length: Total number of characters of the stability indication including decimal point and decimal places (if any). Formatting is right-justified as long as the defined number of places for the output display is sufficient.</p> <p>p Decimal places: Number of decimal places for the weight value.</p> <p>Note: Default: 10:0.</p>
%C[:m]%	<p>Value counter</p> <p>Note: The measured value counter arises with fix counting rate and can be used as a Time information</p>	<p>m Maximal counter value: supplies the measured value counter modulo indicated value m.</p>
%T[:[l]][:p]]%	<p>Temperature level of the measuring cell.</p> <p>Note: The temperature value in the cell isn't identical to the balancing ambient temperature and can not be used instead.</p>	<p>l Length: Total number of characters of the temperature value including decimal point and decimal places (if any). Formatting is right-justified as long as the defined number of places for the output display is sufficient.</p> <p>p Decimal places: Number of decimal places for the weight value (default: 3).</p> <p>Note: Maximum number of decimal places is limited to 3.</p>
<tex>	<p>Constant Text</p>	<p>Note: The characters % and " need to be escaped when they occur in static text:</p> <p>%% Percent sign as constant text at the start of format or after a <Blank></p> <p>\ " Double quotation mark as a text component</p>
_	<p>Information delimitation</p>	

Comments

- The format specification is permanently stored under user data.
- Only applies to the **B01** and **B02** commands.
- With overload the results %W% and %A% the value 99999999 are given.
With underload the results %W% and %A% the value -99999999 are given.
- %C% concerns an internal value counter of signal processing. This is initialized after each PowerOn of the weighbridge with 0. With the maximum count 4294967295 (4Byte unsigned long) an overflow occurs and the counter starts again with 0. With Lotus/ Magellan balances with a time interval of measured value counter of 10.923 ms this overflow takes place on the 543rd day after PowerOn. With a "modulo-1000" formatting, the overflow counter runs only to 999 and then returns to 0 and starts again.

Examples

↓	B04_"S_%S%_W:11:3%_U%_C:100%"	Set the FastHost format specification
↑	B04_A	FastHost format specification is set
↓	B01	Request a weight value
↑	S_D_____1234.563_g_23	B01/B02 response (weight right-aligned)

↓	B04_"Weight=_%W:2%_U%"	Set the FastHost format specification
↑	B04_A	FastHost format specification is set
↓	B01	Request a weight value
↑	Weight=_1234.56_g	B01/B02 response (weight left-aligned)

↓	B04_"%%W%*_%A%=_%WA%"	Set the FastHost format specification
↑	B04_A	FastHost format specification is set
↓	B01	Request a weight value
↑	%W%*_%A%=_%WA%	B01/B02 supply this response

↓	B04_"\"%W%\"_is_%W%"	Set the FastHost format specification
↑	B04_A	FastHost format specification is set
↓	B01	Request a weight value
↑	"%W%"_is_161.5435	B01/B02 supply this response:

B05 – FastHost: Query/Set Stability Criteria**FastHost****Description**

Use B05 to define a stability criterion for the weighing signal in grams/time interval.

Syntax**Commands**

B05	Query of the FastHost stability criteria.
B05_<Width>	Set the FastHost stability criteria in grams.

Balance Responses

B05_A_<Width>	Current FastHost stability criteria.
B05_A	FastHost stability criteria is set.
B05_L	Command understood but not executable (incorrect parameter).
B05_I	Parameter not permanently saved.

Parameter

Name	Type	Values	Meaning
<Width>	Float	See comments	FastHost signal width as stable criteria in grams

Comments

- The condition of stability is permanently stored.
- Only the signal width can be adjusted to determine the FastHost stability by means of B05. The observation time for the determination of the signal width is given by the balance system. This can be different, depending on the adjustment of the balance parameters and on the type.
- If the value of the FastHost signal width (B05) is more than 1000 times smaller than the smallest resolution of the balance, then the internal stability information of the balancing system will be sent to the stability marking, i.e. same behavior concerning stability information as with the S-commands.
- The default value of the balance according to its type definition is set using B05 0.
- The value has up to (number of decimal places balance +3) decimal places. The size of the value can be determined locally using B04 and the parameter %Q%, and typically moves within the range of approx. ½ digit/time interval.

Example

↓	B05	Query of the FastHost stability criteria
↑	B05_A_0.0006	Width of signal for stability is 0.0006 g

B06 – FastHost: Query/Set Downsampling Factor

FastHost

Description

Use B06 to specify that only every nth value should be output within the context of repetitive continuous sending.

Syntax

Commands

B06	Query of the FastHost downsampling factor.
B06_<Rate>	Set the FastHost downsampling factor.

Balance Responses

B06_A_<Rate>	Current FastHost downsampling factor.
B06_A	FastHost downsampling factor is set.
B06_L	Command understood but not executable (incorrect parameter).
B06_I	Parameter not permanently saved.

Parameter

Name	Type	Values	Meaning
<Rate>	Integer	1	Give all values (max. rate)
		2	Give every second value
		3	Give every third value
	
		65,535	Send every 65,535 th sample only

Comments

- Settings are permanently stored.
- In the case of repetitive sending, the time between outputs is calculated using (value B06 x value B03).

Example

↓	B06	Query of the FastHost downsampling factor
↑	B06_A_4	With continuous sending every fourth value will be given

B07 – FastHost: Query/Set Switch-On Mode

FastHost

Description

You can use B07 to specify that repetitive continuous sending (as in B02) should start automatically as soon as the balance is switched on.

Syntax

Commands

B07	Query of the FastHost switch-on mode.
B07_<Mode>	Set the FastHost switch-on mode.

Balance Responses

B07_A_<Mode>	Current FastHost switch-on mode.
B07_A	FastHost switch-on mode is set.
B07_L	Command understood but not executable (incorrect parameter).
B07_I	Parameter not permanently saved.

Parameter

Name	Type	Values	Meaning
<Mode>	Integer	0	Normal start
		1	Automatic send after switch-on is activated

Comments

- The switch-on mode is permanently stored.

Example

↓	B07	Query of the FastHost switch-on mode
↑	B07_A_1	Automatic send after switch-on is activated

B08 – FastHost: Zero with FastHost Stability Criteria

FastHost

Description

Use B08 to define the stability criterion for the FastHost zero, in a similar way to B05 for the weight value.

Syntax

Command

B08	Set to zero with FastHost stability criteria.
-----	---

Balance Responses

B08_A	Set to zero with FastHost stability criteria successfully completed.
B08_-	Command stopped due to overload.
B08_+	Command stopped due to underload.
B08_I	Command not available, stopped or timed-out.

Comments

- The switch-on mode is permanently stored.
- Only affects B01 and B02.
- If a very large criterion has been selected (\geq weighing capacity), the corresponding command is ZI.
- If a very small criterion has been selected, this will result in a timeout error.
- With B08 0, the stability criterion corresponds to the default weight value and to command Z.

Example

↓	B08	Set to zero with FastHost stability criteria
↑	B07_A_1	Set to zero with FastHost stability criteria successfully completed

C0 – Query/Set Adjustment Settings

Level 2

Description

C0 can be used to define the type of adjustment. Other commands are required to actually trigger the adjustment procedure and define the adjusted weights.

Syntax

Commands

C0	Query of the current adjustment setting.
C0_<Mode>_<Weight>	Set the adjustment setting.

Balance Responses

C0_A_<Mode>_<Weight>_<Value>_<Unit>	Weight value and unit specify the value of the weight for an external calibration requested from the user via the display (see command C1). The unit corresponds to the factory setting of host unit , e.g. gram (g) with standard balances or carat (ct) with carat balances respectively. With internal adjustment, neither weight value nor unit appears.
C0_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
C0_A	Command understood and executed successfully.
C0_L	Command understood but not executable (incorrect parameter; certified version of the balance).

Parameters

Name	Type	Values	Meaning
<Mode>	Integer	0	Mode = Manual The adjustment can only be triggered manually. A change in the ambient conditions has no influence on the initiation of the calibration procedure.
		1	Mode = Auto, status display AutoCal or Cal not activated When a considerable change in the ambient conditions is determined, the status display AutoCal or Cal will be activated; this means the balance will ask for adjustment.
		2	Mode = Auto, status display "AutoCal" or "Cal" flashes The sensors built into the balance have determined a considerable change in the ambient conditions. The balance requests an adjustment or at least a test (see TSTx commands).
<Weight>	Integer	0	Internal weight (factory setting)
		1	External weight

Comments

- Setting *<Mode>* = 1 and *<Weight>* = 0 corresponds to the menu setting "ProFACT" / "FACT" under "Adjust/Test".
- The value of the external weight can be changed in the menu of the balance under "Adjust/Test " (see operating instructions) or with **M19**.
- Use **C1** to start an adjustment defined with C0.
- **C2** is independent of C0.
- C0 must be reset manually; @ has no effect.

Examples

↓	C0	Query of the current status and setting of the adjustment
↑	C0_A_2_1_ "100.000_g"	Current setting of mode is "Auto". The ambient conditions of the balance have changed so much that the balance requests an adjustment (<i><Mode></i> = 2) with the external weight (<i><Weight></i> = 1). The adjustment is initiated with the command C1 and requires a weight of 100.000 g
↓	C0_0_1	Set adjustment setting to manual and external
↑	C0_A	Adjustment setting set

C1 – Start Adjustment According to Current Settings

Level 2

Description

C1 is used to carry out adjustment in accordance with the C0 settings.

Syntax

Command

C1	Start the adjustment according to the current setting C0.
----	---

First Balance Responses

C1_B	The adjustment procedure has been started. Wait for second response (see comment).
C1_I	Command understood but currently not executable (balance is currently executing another command). No second response follows.
C1_L	Command understood but not executable (e.g. certified version of the balance). No second response follows.

Further Balance Responses

C1_ "<Weight>_<Unit>"	Weight request with external adjustment.
C1_A	Command understood and executed successfully.
C1_I	Command understood but currently not executable (e.g. stability not attained or wrong weight loaded) .

Comments

- Commands sent to the balance during the adjustment operation are not processed and responded to in the appropriate manner until the adjustment is at an end.
- Use @ to abort a running adjustment.

Example

↓	C1	Start the adjustment according to the current setting
↑	C1_B	Adjustment operation started
↑	C1_ "_____0.00_g"	Prompt to unload the balance
↑	C1_ " _2000.00_g"	Prompt to load the adjustment weight of 2000.00 g
↑	C1_ "_____0.00_g"	Prompt to unload the balance
↑	C1_A	Adjustment completed successfully

C2 – Start Adjustment with External Weight

Level 2

Description

Regardless of the **C0** setting, C2 carries out external adjustment with the reference weight defined in **M19**.

Syntax

Command

C2	Start the external adjustment. Query of the current weight used by means of the C0 command.
----	--

First Balance Responses

C2_B	The adjustment procedure has been started.
C2_I	Command understood but currently not executable (balance is currently executing another command). No second response follows.
C2_L	Command understood but not executable (e.g. adjustment with an external weight is not admissible, certified version of the balance). No second response follows.

Further Balance Responses

C2_ "<Weight>"	Prompt to unload or load the balance.
C2_A	Command understood and executed successfully.
C2_I	Command understood but currently not executable (e.g. stability not attained or wrong weight loaded).

Comments

- Commands sent to the balance during the adjustment operation are not processed and responded to in the appropriate manner until the adjustment is at an end.
- Use @ to abort a running adjustment.
- For XS/XP only: If a test weight is defined, this test weight is used instead of the weight defined in the command **M19**.

Example

↓	C2	Start the external adjustment
↑	C2_B	Adjustment operation started
↑	C2_ "_____0.00_g"	Prompt to unload the balance
↑	C2_ "_____2000.00_g"	Prompt to load adjustment weight 2000.00 g
↑	C2_ "_____0.00_g"	Prompt to unload the balance
↑	C2_A	Adjustment completed successfully

C3 – Start Adjustment with Internal Weight

Level 2

Description

You can use C3 to start an internal adjustment procedure.

Syntax

Command

C3	Start the internal adjustment.
----	--------------------------------

First Balance Responses

C3_B	The adjustment procedure has been started. Wait for second response.
C3_I	Adjustment can not be performed at present as another operation is taking place. No second response follows.
C3_L	Adjustment operation not possible (e.g. no internal weight). No second response follows.

Further Balance Responses

C3_A	Adjustment has been completed successfully.
C3_I	The adjustment was aborted as, e.g. stability not attained or the procedure was aborted with the C key.

Comments

- Commands sent to the balance during the adjustment operation are not processed and responded to in the appropriate manner until the adjustment is at an end.
- Use @ to abort a running adjustment.
- For XS/XP only: If ProFACT Advance is defined and activated, the internal adjustment is started by following the ProFACT Advance settings.

Example

↓	C3	Start the internal adjustment
↑	C3_B	Adjustment operation started
↑	C3_A	Adjustment completed successfully

COM – Query/Set Communication Parameters of the built-in Interface

Serial Level 2

Description

You can use this command to define the connection parameters of the permanently integrated RS232 interface for a stand-alone weighing platform.

Syntax

Commands

COM	Query of the existing interface settings.
COM_<Port>_<Baud>_<Bit>_<HS>	Set parameters of the specified interface to desired values.

Balance Responses

COM_A_<Port>_<Baud>_<Bit>_<HS>	Current communication parameters.
COM_A	Command executed successfully.
COM_I	Command understood but not executable (e.g. update rate is too high for the selected baud rate, see comments).
COM_L	Command understood but not executable (e.g. parameter incorrect).

Parameters

Name	Type	Values	Meaning
<Port>	Integer	0	Built-in RS232 interface
<Baud>	Integer	0	110 baud
		1	300 baud
		2	600 baud
		3	1200 baud
		4	2400 baud
		5	4800 baud
		6	9600 baud (factory setting)
		7	19200 baud
<Bit>	Integer		Bits Parity Stop bits
		0	7 Even 1
		1	7 Odd 1
		2	7 None 1
		3	8 None 1 (factory setting)
		4	7 Even 2
		5	7 Odd 2
		6	7 None 2
7	8 None 2		

<HS>	Integer	0	No handshake (factory setting)
		1	Software handshake (Xoff – Xon controlled protocol)
		2	Hardware handshake (CTS – RTS controlled protocol)

Comments

- Command only available without a connected terminal.
- If an option is present in the system, the host is automatically assigned to that interface and the COM command is not available anymore.
- The answer is returned with the current settings, the settings are changed afterwards.
- No values other than those specified must be used; otherwise, uncontrollable settings may result.
- When adjusting the values, the connection parameters of the connected communication partner must also be adjusted. Otherwise, it will not be possible to establish any further communication.

Examples

↓	COM	Send current settings for interface parameters for all present interfaces
↑	COM_B_0_6_3_0	RS-232 is set to 9600 baud, 8 bits, no parity, 1 stop bit, no handshake

↓	COM_0_8_3_0	Setting the parameters for the serial interface to 38400 baud, 8 data bits, no parity, 1 stop bit, no handshake
↑	COM_A	Parameters successfully set to desired values

COPT – Query/Set Interface Option Parameters

Level 2

Description

You can use this command to define the additional commands for configuring the interface options in the stand-alone weighing module.

Syntax

Commands

COPT	Query configurations for options.
COPT_<function>"_<id>"_<typ>"_<val>"	Set configurations for options.
COPT_"start"	Starts configuration. Host connection is suspended.
COPT_"end"	Ends the configuration. Host connection is resumed.
COPT_"get"_<id>"	Read a datum.
COPT_"set"_<id>"_<typ>"_<val>"	Sets configurations for options.

Balance Responses

COPT_A_<response>"	Command executed successfully.
COPT_I	Command understood but currently not executable (e.g. no option inserted, "start" command not executed, COPT already active).
COPT_L	Command understood but currently not executable (e.g. parameter incorrect).

Parameters for all options

Name	Type	Values	Meaning
<function>	String	start get set end	Starts configuration Read a datum Set a datum Ends the configuration
<id>	String	System.Infos.DeviceName System.Infos.SWNumber System.Infos.SWVersion	DeviceName (read only) SWNumber (read only) SWVersion (read only)
<typ>	String	Max 20 characters	Type of the datum
<val>	String		Value of the datum

<id>	String	System.Infos.Status	System status (read only)
<typ>	String	i4	Type of the datum
<val>	String	-5 -4 -3 -2 -1 0	Lost DHCP lease (only Ethernet option) Busy Wrong/not configured Starting up Out-of-order Ready

Parameters for R232 option

Name	Type	Values	Meaning
<id>	String	Connection.ConnectionList.1.Baudrate	Baudrate
<typ>	String	i4	Type of the datum
<val>	String	2 3 4 5 6 7 8	600 1200 2400 4800 9600 19200 38400

<id>	String	Connection.ConnectionList.1.Parity	Parity
<typ>	String	i4	Type of the datum
<val>	String	0 1 2 3	7Bit/None 7Bit/Even 7Bit/Odd 8Bit/None

<id>	String	Connection.ConnectionList.1.Handshake	Handshake
<typ>	String	i4	Type of the datum
<val>	String	0 1 2	None XON/XOFF RTS/CTS

<id>	String	Connection.ConnectionList.1.EndOfLine	End of Line
<typ>	String	i4	Type of the datum
<val>	String	1 2 3	<CR><LF> <CR> <LF>

<id>	String	Connection.ConnectionList.1.CharSet	Char Set
<typ>	String	i4	Type of the datum
<val>	String	0 1	ANSI/WIN IBM/DOS

Parameters for Ethernet option

<id>	String	System.Infos.SerialNumber	Serial number (read only)
<typ>	String	string	
<val>	String	Max 10 characters	

<id>	String	Interface.DHCP	DHCP
<typ>	String	i4	Type of the datum
<val>	String	0 1	Off On

<id>	String	Interface.IPAdress	IP Address
<typ>	String	string	Type of the datum
<val>	String	Max 15 characters	

<id>	String	Interface.SubnetMask	End of Line
<typ>	String	string	Type of the datum
<val>	String	Max 15 characters	

<id>	String	Interface.StandardGateway	Standard Gateway
<typ>	String	string	Type of the datum
<val>	String	Max 15 characters	

<id>	String	Interface.DomainNameServer	Domain Name Server
<typ>	String	string	Type of the datum
<val>	String	Max 15 characters	

<id>	String	Interface.HostName	Host Name
<typ>	String	string	Type of the datum
<val>	String	Max 41 characters	

<id>	String	Interface.Apply	This function has to be called for changes in parameters with an identifier starting with "Interface" to take effect. This command can be called to make the previous interface parameters permanent.
<typ>	String		Two quotation marks with no space between.
<val>	String		Two quotation marks with no space between.

<id>	String	Connection.ConnectionList.1.Client	Connection to Client
<typ>	String	i4	Type of the datum
<val>	String	0 1	Off On

<id>	String	Connection.ConnectionList.1.ClientAddress	Client Address
<typ>	String	string	Type of the datum
<val>	String	Max 40 characters	

<id>	String	Connection.ConnectionList.1.ClientPort	Client Port
<typ>	String	string	Type of the datum
<val>	String	1...65535	Increment 1

<id>	String	Connection.ConnectionList.1.Server	Server
<typ>	String	i4	Type of the datum
<val>	String	0 1	Off On

<id>	String	Connection.ConnectionList.1.ServerPort	Server Port
<typ>	String	i4	Type of the datum
<val>	String	1...65535	Increment 1

<id>	String	Connection.ConnectionList.1.EndOfLine	End of Line
<typ>	String	i4	Type of the datum
<val>	String	1 2 3	<CR><LF> <CR> <LF>

<id>	String	Devices.Host.MuxID	Mux ID
<typ>	String	string	Type of the datum
<val>	String	Max 6 characters	

Comments

- This command is only available if an interface option is present.
- If no interface option is present, balance response will be ES.

Examples

Example for all options

↓	COPT_"get_"_ "System.Infos.DeviceName"	Get device name
↑	COPT_A_"Value='Ethernet Option' "	Device name = Ethernet Option
↓	COPT_"get_"_ "System.Infos.Status"	Get system status
↑	COPT_A_"Value='0' "	System is ready

Example for RS232 option

↓	COPT_"start"	Starts configuration. Host
---	--------------	----------------------------

		connection is suspended.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"set"_"Connection.ConnectionList.1.Baudrate"_"i4"_"5"	Set baud rate to 4800 baud.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"set"_"Connection.ConnectionList.1.Parity"_"i4"_"1"	Set format to 7 Bit odd parity.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"end"	Ends the configuration. Host connection is resumed.
↑	COPT_A_ " "	Command executed successfully.
↓	COPT_"get"_"System.Infos.Status"	Get system status
↑	COPT_A_"Value='0' "	System is ready
↓	COPT_"start"	Starts configuration. Host connection is suspended.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"set"_"Interface.DHCP"_"i4"_"0"	Set DHCP off.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"set"_"Interface.IPAddress"_"string"_"172.24.113.7"	Set IP address to 172.24.113.7.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"set"_"Interface.SubnetMask"_"string"_"255.255.248.0"	Set subnet mask to 255.255.248.0.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"set"_"Interface.StandardGateway"_"string"_"172.24.112.1"	Set standard gateway to 172.24.112.1.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"set"_"Interface.Apply"_" " "	Apply interface parameters.
↑	COPT_ " "	Command executed successfully.
↓	COPT_"set"_"Connection.ConnectionList.1.ServerPort"_"i4"_"8001"	Set Server port to 8001.
↑	COPT_ " "	Command executed successfully.

↓	COPT_"set"_"Connection.ConnectionList.1.Server"_" "i4"_"1"	Set server visible to client.
↑	COPT_" "	Command executed successfully.
↓	COPT_"end"	Ends the configuration. Host connection is resumed.
↑	COPT_A_" "	Command executed successfully.
↓	COPT_"get"_"System.Infos.Status"	Get system status
↑	COPT_A_"Value=`0`"	System is ready

D – Display: Write Text to Display

Level 1

Description

Use **D** to write text to the balance display.

Syntax

Command

D_ "<DisplayText>"	Write text into the balance display.
--------------------	--------------------------------------

Balance Responses

D_A	Command understood and executed successfully: Text appears left-aligned in the balance display marked by a symbol, e.g. *.
D_I	Command understood but currently not executable.
D_L	Command understood but not executable (incorrect parameter or balance with no display) .

Parameter

Name	Type	Values	Meaning
<DisplayText>	String		Text on the balance display

Comments

- A symbol in the display, e.g. * indicates that the balance is not displaying a weight value.
- The maximum number of characters of "text" visible in the display depends on the balance type. If the maximum number of characters is exceeded, the text disappears on the right side.
- [Quotation marks](#) can be displayed as indicated in chapter 2.1.3.

Examples

↓	D_ "HALLO"	Write "HALLO" into the balance display
↑	D_A	The full text HALLO appears in the balance display
↓	D_ " "	Clear the balance display
↑	D_A	Balance display cleared, marked by a symbol, e. g. *

See also

[DW – Display: Show Weight](#): Show weight value on the display instead of a text

DAT – Query/Set Date

Level 2

Description

Set/Query the balance system date.

Syntax

Commands

DAT	Query of the current date of the balance.
DAT_<dd>_<mm>_<yyyy>	Set the date of the balance.

Balance Responses

DAT_A_<dd>_<mm>_<yyyy>	Current date of the balance.
DAT_A	Command understood and executed successfully.
DAT_I	Command understood but currently not executable (balance is currently executing another command).
DAT_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<dd>	Integer	01..31	Day
<mm>	Integer	01..12	Month
<yyyy>	Integer	2000..2099	Year

Comment

- The set date is retained even after the Cancel-command @.

Example

↓	DAT	Query of the current date of the balance
↑	DAT_A_01_04_2011	Current date of the balance is 1 st April 2011

DW – Display: Show Weight**Level 1****Description**

Writes the current weight value to the balance display using the set unit. This command is used to reset the display after using the **D** command.

Syntax**Command**

DW	Switch the main display to weight mode.
----	---

Balance Responses

DW_A	Command understood and executed successfully: Main display shows the current weight value.
DW_I	Command understood but currently not executable.

Comment

- DW resets the balance display following a **D** command.

IO – List All Implemented MT-SICS Commands

Level 0

Description

The `IO` command lists all commands implemented in the present software. Thus, there is no need of the supplement sheet delivered with the previous versions of this manual.

All level 0 commands are listed in alphabetical order before all commands of level 1 etc.

Syntax

Command

<code>IO</code>	Send list of all implemented MT-SICS commands.
-----------------	--

Balance Responses

<code>IO_B_<Level>_<Command></code>	Number of the MT-SICS level where the command belongs to
<code>IO_B_<Level>_<Command></code>	2nd (next) command implemented.
<code>IO_B...</code>	...
<code>IO_A_<Level>_<Command></code>	Last command implemented.
<code>IO_I</code>	Command understood but currently not executable (balance is currently executing another command) .

Parameters

Name	Type	Values	Meaning
<code><Level></code>	Integer	0	MT-SICS level 0 (Basic set)
		1	MT-SICS level 1 (Elementary commands)
		2	MT-SICS level 2 (Extended command list)
		3	MT-SICS level 3 (Application specific command set)
<code><Command></code>	String		MT-SICS command

Comments

- If a terminal and a weighing platform are being used, the command list of the terminal is output. If only a platform is being used, the command list of the platform is output.
- If you are only working with a weighing platform, the command set depends on whether an additional option is being used:
 - With option → RS232 permanent; only a limited command set is available.
 - Without option → RS232 permanent = host, normal weighing platform command set.
- If `IO` lists commands that cannot be found in the manual, these are reserved commands "for internal use" or "for future use", and should not be used or altered in any way.

Example

↓	<code>IO</code>	Send list of commands
↑	<code>IO_B_0_"IO"</code>	Level 0 command <code>IO</code> implemented
↑	<code>IO_B...</code>	...
↑	<code>IO_B_0_"@"</code>	Level 0 command <code>@</code> (cancel) implemented
↑	<code>IO_B_1_"D"</code>	Level 1 command <code>D</code> implemented
↑	<code>IO_B...</code>	...
↑	<code>IO_A_3_SM4</code>	Level 3 command <code>SM4</code> implemented

I1 – Query MT-SICS Level and MT-SICS Versions

Level 0

Description

Query MT-SICS level and versions.

Syntax

Command

I1	Query of MT-SICS level and MT-SICS versions.
----	--

Balance Responses

I1_A_"<Level>"_"<V0>"_"<V1>"_"<V2>"_"<V3>"	Current MT-SICS level and MT-SICS versions.
I1_I	Command understood but currently not executable.

Parameters

Name	Type	Values	Meaning
<Level>	String	0	MT-SICS level 0
		01	MT-SICS level 0 and 1
		012	MT-SICS level 0, 1 and 2
		03	MT-SICS level 0 and 3
		013	MT-SICS level 0, 1 and 3
		0123	MT-SICS level 0, 1, 2, and 3
		3	Application device with MT-SICS level 3
<V0>..<V3>	String		MT-SICS versions of the related level (0 to 3)

Comment

- The command [I14](#) provides more comprehensive and detailed information.

I2 – Query Balance Data (Type and Capacity)

Level 0

Description

Use I2 to query the balance data (type and capacity), including the weighing capacity. The response is output as a whole string.

Syntax

Command

I2	Query of the balance data.
----	----------------------------

Balance Responses

I2_A_ "<Type>_<Capacity>_<Unit>"	Balance type and capacity.
I2_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring) .

Comments

- With DeltaRange balances, the last decimal place is available only in the fine range.
- The number of characters of "text" depends on the balance type and capacity.

Example

↓	I2	Query of the balance data
↑	I2_A_ "XP205_ Excellence_ Plus_220.00900_g"	Balance type and capacity

I3 – Query Balance Software Version and Type Definition Number

Level 0

Description

Provides the software version and the type definition number.

Syntax

Command

I3	Query of the balance Software version and type definition number.
----	---

Balance Responses

I3_A_ "<Software>_<TDNR>"	Balance Software version and type definition number.
I3_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring) .

Comments

- Only the software version of the terminal software is issued.
- If no terminal is present, the bridge software is issued instead.
- More detailed information is available with [I14](#).

Example

↓	I3	Query of the Software version number(s) and type definition number
↑	I3_A_ "4.10_10.28.0.493.142"	4.10: Software version number 10.28.0.493.142 Type definition number

I4 – Query Serial Number

Level 0

Description

Use I4 to query the serial number of the balance. In the case of Excellence balances, the serial number of the terminal is output.

Syntax

Command

I4	Query of the serial number.
----	-----------------------------

Balance Responses

I4_A_ "<SNR>"	Serial number.
I4_I	Command not understood, not executable at present Command understood but currently not executable (balance is currently executing another command, e.g. initial zero setting).

Comments

- The serial number agrees with that on the model plate and is different for every balance.
- The serial number can be used, for example, as a device address in a network solution.
- The balance response to I4 appears unsolicited after switching on and after the cancel command (@).
- Only the serial number of the terminal is issued.
- If no terminal is present, the serial number of the bridge is issued instead.
- More detailed information is available with I14.

Example

↓	I4	Query of the serial number
↑	I4_A_"0123456789"	0123456789: Serial number

I5 – Query SW-Identification Number

Level 0

Description

Provides the software identification number.

Syntax

Command

I5	Query of the SW-identification number.
----	--

Balance Responses

I5_A_ "<SWID>"	SW-identification number with index.
I5_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring) .

Comments

- The SW-Identification number is unique for every Software. It consists of a 8 digit number and an alphabetic character as an index
- Only the software identification number of the terminal is issued.
- If no terminal is present, the software identification number of the bridge is issued instead.
- More detailed information is available with [I14](#).

Example

↓	I5	Query of the SW-identification number
↑	I5_A_ "01234567A"	01234567A: SW-identification number with index

I10 – Query Balance ID**Level 2****Description**

You can use I10 to query the balance ID or define it. This allows an individual name to be assigned to a balance.

Syntax**Commands**

I10	Query of the current balance identification.
I10_ "<ID>"	Set the balance identification.

Balance Responses

I10_A_ "<ID>"	Current identification the balance.
I10_A	Balance identification has been set.
I10_I	Balance identification can not be set at present as another operation is taking place.
I10_L	Command not executed as the name is too long (max. 20 characters).

Comments

- A sequence of maximum 20 alphanumeric characters are possible as <ID>.
- The set balance identification is retained even after the cancel command @.

Example

↓	I10	Query of the current balance identification
↑	I10_A_ "My_Balance"	Current balance identification is "My Balance"

I11 – Query Balance Type

Level 2

Description

You can use I11 to obtain the precise type designation of a balance.

Syntax

Command

I11	Query of the current balance type.
-----	------------------------------------

Balance Responses

I11_A_ "<Type>"	Current balance type.
I11_I	Balance type can not be transferred at present as another operation is taking place.

Comments

- A sequence of maximum 20 alphanumeric characters is possible as <Type>.
- The following abbreviations used in type designations are relevant to MT-SICS:
 - DR = Delta Range.
 - DU = Dual Range.
 - /M, /A = Approved balance.

Example

↓	I11	Query of the current balance type
↑	I11_A_ "XS603SDR"	The balance is an XS603SDR

I14 – Query Balance Information

Level 2

Description

You can use I14 to query detailed information about the balance. All components – including optional accessories – are taken into account and the associated data is output.

Syntax

Command

I14_<No>	Query of the current balance information.
----------	---

Balance Responses

I14_A_<No>_<Index>_<Info>"	Current balance information.
I14_I	Command understood but currently not executable.
I14_L	Command understood but not executable (incorrect parameter) .

Parameters

Name	Type	Values	Meaning
<No>	Integer	0	Instrument configuration
		1	Instrument description
		2	SW-identification number
		3	SW version
		4	Serial number
		5	TDNR number
<Index>	Integer		Index of instrument module
<Info>	String	<Bridge>	Weighing bridge information corresponding to <No>
		<Terminal>	Balance terminal information corresponding to <No>
		<Option>	Balance option information corresponding to <No>
		<Balance>	Balance information corresponding to <No>
		<Printer>	Printer information corresponding to <No>
		<Second Display>	Second Display information corresponding to <No>

Comments

- The response to the query of instrument configuration can comprise one or more lines (compact balances, bridges with/without terminal etc.)
- The description of an option is the language-independent product name, e.g. "RS232-Option".
- If there are several modules of the same kind, the descriptions have an appendix, comprising of a hyphen and a number. Examples: <Option-1>, <Option-2>.
- Not all options contain a serial number or a TDNR number.

Example

↓	I14_0	Query of the current balance information
↑	I14_B_0_1_"Bridge"	Bridge
↑	I14_B_0_2_"Terminal"	Terminal
↑	I14_A_0_3_"Option"	Option
↓	I14_1	Query of the current instrument descriptions
↑	I14_B_1_1_"X205T"	Bridge is a X205T Example continued on next page
↑	I14_B_1_2_"PAT"	Excellence Plus Terminal
↑	I14_A_1_3_"RS232_Option"	RS232 Option
↓	I14_2	Query of the current SW-identification numbers
↑	I14_B_2_1_"11670123A"	SW-identification number of the bridge is 11680123A
↑	I14_B_2_2_"11670456B"	SW-identification number of the terminal is 11680456B
↑	I14_A_2_3_"11670789B"	SW-identification number of the option is 11680789B
↓	I14_3	Query of the current software versions
↑	I14_B_3_1_"4.23"	Version of the bridge software is 4.23
↑	I14_B_3_2_"4.10"	Version of the terminal software is 4.10
↑	I14_A_3_3_"1.01"	Version of the RS232 option software is 1.01
↓	I14_4	Query of the serial numbers
↑	I14_B_4_1_"0123456789"	Serial number of the bridge
↑	I14_B_4_2_"1234567890"	Serial number of the terminal
↑	I14_A_4_3_"2345678901"	Serial number of the RS232 option
↓	I14_5	Query of the type definition numbers
↑	I14_B_5_1_"1.2.3.4.5"	type definition number of the bridge
↑	I14_B_5_2_"1.2.3.4.5"	type definition number of the terminal
↑	I14_A_5_3_"1.2.3.4.5"	type definition number of the RS232 option

I15 – Query "Power On" Time

Level 2

Description

Use I15 to query the "Power On" time of the balance.

Syntax

Command

I15	Query "Power On" Time.
-----	------------------------

Balance Responses

I15_A_ "<minutes>"	Time in minutes since "Power On", accuracy +/-5%.
I15_I	Power On Time can not be transferred at present as another operation is taking place.

Example

↓	I15	Query "Power On" Time
↑	I15_A_123014	Power on of balance was approx.123014 minutes ago.

I16 – Query Date of Next Service

Level 2

Description

You can use I16 to query the date when the balance is next due to be serviced.

Syntax

Command

I16	Query Date of Next Service.
-----	-----------------------------

Balance Responses

I16_A_<dd>_<mm>_<yyyy>	Current Date of Next Service.
I16_I	Date of Next Service can not be transferred at present as another operation is taking place.

Parameters

Name	Type	Values	Meaning
<dd>	Integer	01..31	Day
<mm>	Integer	01..12	Month
<yyyy>	Integer	2000..2099	Year

Example

↓	I16	Query Date of Next Service
↑	I16_A_19_07_2003	Date of Next Service is July 19, 2003

I17 – MinWeigh: Query Date of Next Test**Level 2****Description**

You can use I17 to query the date when the balance's next MinWeigh test is due to be performed.

Syntax**Command**

I17	Query of the current next Date of MinWeigh Test.
-----	--

Balance Responses

I11_A_<dd>_<mm>_<yyyy>	Current next Date of MinWeigh Test.
I11_I	Next Date of MinWeigh Test can not be transferred at present as another operation is taking place.

Parameters

Name	Type	Values	Meaning
<dd>	Integer	01..31	Day
<mm>	Integer	01..12	Month
<yyyy>	Integer	2000..2099	Year

Comments

- MinWeigh can only be activated by a service technician.
- For additional information on MinWeigh (Minimum Weight), see the Operating Instructions for the balance.

Example

↓	I17	Query of the current next Date of MinWeigh Test
↑	I17_A_19_07_2003	Date of Next MinWeigh Test is July 19, 2003

I18 – MinWeigh: Query Method

Level 2

Description

You can use I18 to query the methods defined for determining MinWeigh.

Syntax

Command

I18	Query MinWeigh Methods.
-----	-------------------------

Balance Responses

I18_A_<No>_<Method>_<Parameter>	Current set MinWeigh Methods.
I18_I	MinWeigh Method can not be transferred at present as another operation is taking place.

Parameters

Name	Type	Values	Meaning
<No>	Integer	1 . . n	Number of defined method
<Method>	String		Name of method: The length is restricted up to 20 characters
<Parameter>	String		Test parameter: The length is restricted up to 20 characters

Comments

- MinWeigh can only be activated by a service technician.
- For additional information on MinWeigh (Minimum Weight), see the Operating Instructions for the balance.

Example

↓	I18	Query MinWeigh Methods
↑	I18_B_1_"USP"_3sd,_0.1%"	1 st Method is a USP-Method with 3 x sd and ≤ 0.1% error
↑	I18_B_2_"SOP"_2sd,_1%"	2 nd Method is a SOP-Method with 2 x sd and ≤ 1% error
↑	I18_A_3_"_"	3 rd Method is not defined

I19 – MinWeigh: Query Limits

Level 2

Description

Use I19 to query the tares defined for the MinWeigh methods, along with their Minimum Weight limits.

Syntax

Command

I19	Query of the current MinWeigh limits.
-----	---------------------------------------

Balance Responses

I19_A_<MNo>_<No>_<MinWeigh>_<Tare>	Current MinWeigh limits.
I19_I	MinWeigh limits can not be transferred at present as another operation is taking place.

Parameters

Name	Type	Values	Meaning
<MNo>	Integer	1..n	Number of defined method
<No>	Integer	0..n	Number of defined pair of limits
<MinWeigh>	Float		Determined Minimum Weight value depending on following Tare weight
<Tare>	Float		Tare weight used for MinWeigh determination

Comments

- MinWeigh can only be activated by a service technician.
- For additional information on MinWeigh (Minimum Weight), see the Operation Instructions for the balance.

Example

↓	I20	Query of the current MinWeigh Limits
↑	I19_B_1_0_0.0100_50.0000	Method 1, value pair 0, MinWeigh is 10 mg with tare 50 g
↑	I19_B_1_1_0.0200_120.0000	Method 1, value pair 1, MinWeigh is 20 mg with tare 120 g
↑	I19_B_1_2_0.1000_500.0000	Method 1, value pair 2, MinWeigh is 100 mg with tare 500 g
↑	I19_B_2_0_0.0110_51.0000	Method 2, value pair 0, MinWeigh is 11 mg with tare 51 g
↑	I19_B_2_1_0.0210_121.0000	Method 2, value pair 1, MinWeigh is 21 mg with tare 121 g
↑	I19_B_2_2_0.1010_501.0000	Method 2, value pair 2, MinWeigh is 101 mg with tare 501 g
↑	I19_B_3_0_0.0120_52.0000	Method 3, value pair 0, MinWeigh is 12 mg with tare 52 g
↑	I19_B_3_1_0.0220_122.0000	Method 3, value pair 1, MinWeigh is 22 mg with tare 122 g
↑	I19_A_3_2_0.1020_502.0000	Method 3, value pair 2, MinWeigh is 102 mg with tare 502 g

I20 – Query MinWeigh Parameter

Level 2

Description

Use I20 to query the balance parameter settings for the MinWeigh methods.

Syntax

Command

I20	Query of the current MinWeigh Parameter.
-----	--

Balance Responses

I20_A_<Method>_<Index>_<Parameter>	Current balance type.
I20_I	Balance type can not be transferred at present as another operation is taking place.

Parameters

Name	Type	Values	Meaning
<Method>	Integer	0..n	Number of defined method
<Index>	Integer	0	Weighing mode, see M01
		1	Environment, see M02
		2	Measured value release, see M29
		3	AutoZero, see M03
<Parameter>	Integer		Parameter for the corresponding weight setting

Comment

- The parameters correspond to the values provided for the corresponding MT-SICS commands:
 - Weighing mode, see [M01](#).
 - Environment, see [M02](#).
 - Measured value release, see [M29](#).
 - AutoZero, see [M03](#).

Example

↓	I20	Query of the current balance type
↑	I20_B_1_0_0	Method 1, Weighing mode "Universal"
↑	I20_B_1_1_2	Method 1, Environment "Standard"
↑	I20_B_1_2_1	Method 1, Measured value release "Quick"
↑	I20_B_1_3_1	Method 1, AutoZero "On"
↑	I20_B_2_0_1	Method 2, Weighing mode "Dispensing"
↑	I20_B_2_1_3	Method 2, Environment "Unstable"
↑	I20_B_2_2_0	Method 2, Measured value release "Very quick"
↑	I20_A_2_3_0	Method 2, AutoZero "Off"

I28 – Query Level Sensor

Level 2

Description

You can use I28 to determine whether or not the balance is level.

Syntax

Command

I28	Query of the level sensor.
-----	----------------------------

Balance Responses

I28_A_<Status>	Current state of the level sensor.
I28_I	Level sensor can not be transferred at present as another operation is taking place or there is a failure at the level sensor.

Parameter

Name	Type	Values	Meaning
<Status>	Integer	0	Level
		1	Not level

Comments

- Also supported by stand-alone platforms. During stand-alone operation, the backlighting for the level sensor is not activated.
- If no level sensor is present, an "ES" response is generated.

Example

↓	I28	Query of the current state of the level sensor
↑	I28_A_1	Weighing bridge is out of level

K – Keys: Set Configuration

Level 1

Description

With the κ command, the behavior of the terminal keys may be configured: first, the K command controls whether a key invokes its corresponding function or not and second, it configures whether an indication of which key has been pressed or released is sent to the host interface or not.

Using this functionality, an application running on a connected system (e.g. a PC or PLC) may make use of the balance terminal to interact with the balance operator.

Syntax

Commands

$K_{<Mode>}$	Set configuration.
--------------	--------------------

Balance Responses

$K_A [_{<FuncNr>}]$	Command understood and executed successfully. Mode 4: Function with $<FuncNr>$ was invoked by pressing the corresponding key and executed successfully.
$K_I [_{<FuncNr>}]$	Command understood but currently not executable (balance is actually in menu or input mode). Mode 4: Function with $<FuncNr>$ by pressing the corresponding key, but it could not be successfully executed (e.g. calibration was aborted by user or a negative value was tared).
K_L	Command understood but not executable (incorrect or no parameter).

Additional Responses in Mode 3:

$K_{<Event>}__{<KeyNr>}$	Key $<KeyNr>$ has issued an $<Event>$.
--------------------------	---

Additional Responses in Mode 4:

$K_B_{<FuncNr>}$	Function with $<FuncNr>$ was invoked and started; the execution needs time to complete.
------------------	---

Parameters

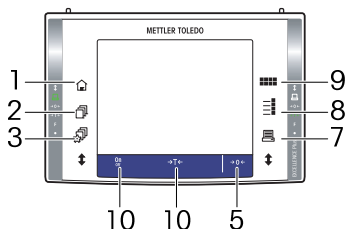
Name	Type	Values	Meaning
$<Mode>$	Integer	1	Functions are executed, no indications are sent (factory setting)
		2	Functions are not executed, no indications are sent
		3	Functions are not executed, indications are sent
		4	Functions are executed, indications are sent
$<Event>$	Char	R	Key was pressed and held around 2 seconds
		C	Key was released (after being pressed shortly or for 2 second)
$<FuncNr>$	Integer	0	Adjustment
		1	Tare
		2	Zero
		3	Data transfer to printing device
		4..6	Reserved for future use
		7	Test

<KeyNr>	Integer	1	Home
		2	User profile (XP balances only)
		3	Settings (XP balances only)
		4	reserved
		5	Zero
		6	reserved
		7	Transfer
		8	Configure actual applications
		9	Applications
		10	Tare On/Off

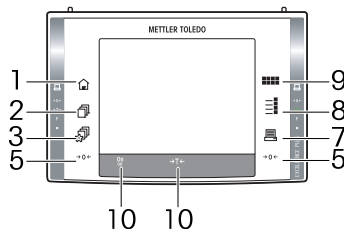
Comments

- There is no key number assigned to the door keys; therefore, no response is invoked upon pressing one of these keys.
- K_1 is the factory setting (default value).
- K_1 active after balance switched on and after the cancel command @.
- K_2 door function is not disabled.
- Only one κ mode is active at one time.
- The mapping of the key numbers on the different terminals are displayed below:

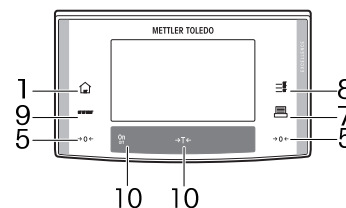
XP Analytical Balances



XP Precision Balances



XS/XA Balances



Example

When a code with a long press is sent, new key commands will not be accepted.

↓	K_4	Set mode 4: when a key is pressed, execute the corresponding function and send the function number as a response
↑	K_A	Command executed successfully
↑	K_B_1	The taring function has been started → taring active
↑	K_A_1	Taring completed successfully
↑	K_B_1	The taring function has been started → taring active
↑	K_I_1	Taring not completed successfully, taring aborted (e.g. tried to tare a negative value)

M01 – Query/Set Weighing Mode

Level 2

Description

You can use M01 to set the weighing mode or query the current setting.

Syntax

Commands

M01	Query of the current weighing mode.
M01_<WeighingMode>	Set the weighing mode.

Balance Responses

M01_A_<WeighingMode>	Current weighing mode.
M01_A	Command understood and executed successfully.
M01_I	Command understood but currently not executable.
M01_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<WeighingMode>	Integer	0	Normal weighing/Universal
		1	Dosing
		2	Sensor mode
		3	Check weighing

Comment

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M01_0	Set the weighing mode to normal
↑	M01_A	Weighing mode is set

M02 – Query/Set Environment

Level 2

Description

You can use M02 to adjust the balance so that it is optimized for the local ambient conditions, or to query the current value.

Syntax

Commands

M02	Query of the current environment.
M02_<Environment>	Set the environment.

Balance Responses

M02_A_<Environment>	Current environment.
M02_A	Command understood and executed successfully.
M02_I	Command understood but currently not executable.
M02_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<Environment>	Integer	0	Very stable
		1	Stable
		2	Standard
		3	Unstable
		4	Very unstable

Comments

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).
- Not all balances offer the complete range of settings. If a setting is made that is not supported by the balance, an error message is issued (M02_L).

Example

↓	M02_3	Set the environment to unstable
↑	M02_A	Environment is set

M03 – Query/Set AutoZero

Level 2

Description

Use M03 to switch the AutoZero function on or off and query its current status.

Syntax

Commands

M03	Query of the current AutoZero function.
M03_<AutoZero>	Set the AutoZero function.

Balance Responses

M03_A_<AutoZero>	Current AutoZero function
M03_A	Command understood and executed successfully.
M03_I	Command understood but currently not executable.
M03_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<AutoZero>	Integer	0	AutoZero is switched off
		1	AutoZero is switched on

Comment

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M03_1	Switch on the AutoZero function
↑	M03_A	AutoZero function is activated

M04 – Query/Set SmartSens Functions

Level 2

Description

You can use M04 to assign functions to the contact-free sensors (SmartSens) in the top left and right-hand corners of the terminal and/or the external sensors available as an option (ErgoSens), or to call up the function settings.

Syntax

Commands

M04	Query of the current SmartSens functions.
M04_<No>_<Function>	Set the SmartSens function.

Balance Responses

M04_B_<No>_<Function> M04_B... M04_A_<No>_<Function>	Current setting of the first SmartSens function. ... Current setting of the last SmartSens or ErgoSens function.
M04_A	Command understood and executed successfully.
M04_I	Command understood but currently not executable.
M04_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<No>	Integer	0, 1	left SmartSens, right SmartSens
		2, 3	ErgoSense 1 (Aux 1), ErgoSense 2 (Aux 2)
<Function>	Integer	0	Off
		1	Door
		2	Left Door
		3	Right Door
		4	Zero
		5	Tare
		6	Print
		7	ID1
		8	ID2
		9	ID3
		10	ID4
		11	ID5
		12	ID6
		13	Header
		14	Footer
15	PreTare		

Comment

- The parameter settings will be saved and the only way to reset the default values will be via MT-SICS or by means of a balance master reset (not @).

Examples

↓	M04_0_2	Set the left SmartSens to 2 nd function (e.g. door)
↑	M04_A	Left SmartSens function set

↓	M04	Query of the current SmartSens functions
↑	M04_B_0_0	Left SmartSens: No function
↑	M04_B_1_4	Right SmartSens: Zero
↑	M04_B_2_0	ErgoSens Aux 1: No function
↑	M04_A_3_0	ErgoSens Aux 2: No function

M05 – Query User Profile

Level 2

Description

You can use M05 to query the user profile name.

Syntax

Command

M05	Query of the available user profiles.
-----	---------------------------------------

Balance Responses

M05_B_<No>_ "<Profile>"	First user profile.
M05_B...	...
M05_A_<No>_ "<Profile>"	Last user profile.
M05_I	Command understood but currently not executable.
M05_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<No>	Integer	0.. max. user profiles	User profile number
<Profile>	String		User profile name

Example

↓	M05	Query of the available user profiles
↑	M05_B_0_ "Home"	Default user profile is called Home
↑	M05_B_1_ "Julie Evans"	1 st user profile is called Julie Evans
↑	M05_B_2_ "Method 1"	2 nd user profile is called Method 1
↑	M05_B_3_ "Statistics 1"	3 rd user profile is called Statistics 1
↑	M05_B_4_ "User 4"	4 th user profile is called User 4
↑	M05_B_5_ "User 5"	5 th user profile is called User 5
↑	M05_B_6_ "User 6"	6 th user profile is called User 6
↑	M05_A_7_ "User 7"	7 th user profile is called User 7

M06 – Query/Set Current User Profile

Level 2

Description

You can use M06 to define which user profile the balance should use, or to query the current profile setting.

Syntax

Commands

M06	Query of the current user profile.
M06_<No>	Set the user profile.

Balance Responses

M06_A_<No>	Current user profile.
M06_A	Command understood and executed successfully.
M06_I	Command understood but currently not executable.
M06_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<No>	Integer	0.. max. user profiles	User profile number

Comment

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M06_3	Activate the 3 rd user profile
↑	M06_A_3	3 rd user profile is set

M07 – Query/Set AutoDoor

Level 2

Description

You can use M07 to set the AutoDoor function or query the current setting. The autodoor function automatically opens and closes the doors of the glass draft shield as soon as this is necessary.

Syntax

Commands

M07	Query of the current AutoDoor setting.
M07_<AutoDoor>	Set the AutoDoor function.

Balance Responses

M07_A_<AutoDoor>	Current AutoDoor setting.
M07_A	Command understood and executed successfully.
M07_I	Command understood but currently not executable.
M07_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<AutoDoor>	Integer	0	AutoDoor function switched off
		1	AutoDoor function switched on

Comments

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).
- For further details, see the section titled "Door function" in the "User Settings" chapter of the Manual for the balance.

Example

↓	M07_1	AutoDoor function switched on
↑	M07_A	AutoDoor function is set.

M08 – Display: Query/Set Brightness

Level 2

Description

You can use M08 to set the brightness of the terminal display or query the current setting.

Syntax

Commands

M08	Query of the current display brightness.
M08_<Brightness>	Set the display brightness.

Balance Responses

M08_A_<Brightness>	Current display brightness.
M08_A	Command understood and executed successfully.
M08_I	Command understood but currently not executable.
M08_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<Brightness>	Integer	20..100	Display brightness in %

Comments

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).
- Technically speaking, it is possible to enter values with decimal places, but the balance will omit these and will only store the integer value.

Examples

↓	M08_55	Set the display brightness to 55%
↑	M08_A	Display brightness is set
↓	M08	Query of the current display brightness
↑	M08_A_60	Current display brightness is set to 60%

M09 – Display: Query/Set Contrast

Level 2

Description

You can use M09 to set the contrast of the terminal display or query the current setting.

Syntax

Commands

M09	Query of the current display contrast.
M09_<Contrast>	Set the display contrast.

Balance Responses

M09_A_<Contrast>	Current display contrast.
M09_A	Command understood and executed successfully.
M09_I	Command understood but currently not executable.
M09_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<Contrast>	Integer	0..100	Display contrast in %

Comments

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).
- Technically speaking, it is possible to enter values with decimal places, but the balance will omit these and will only store the integer value.

Example

↓	M09	Query of the current display contrast
↑	M09_A_60	Current display contrast is set to 60%

M10 – Beeper: Query/Set Display Mode

Level 2

Description

You can use M10 to set or query the display mode of XS balances.

Syntax

Commands

M10	Query the actual display mode.
M10_<DisplayMode>	Set the display mode.

Balance Responses

M10_A_<DisplayMode>	Current display mode.
M10_A	Command understood and executed successfully.
M10_I	Command understood but currently not executable.
M10_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<DisplayMode>	Integer	1	First mode: Display weight and SmartTrac
		2	Second mode: Display weight
		3	Third mode: Display weight, info fields and SmartTrac

Comments

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).
- It is also possible to change the display mode by means of the display function key on the terminal. Please refer to the Operating Instructions for Excellence XS Balances.
- This command is only supported by XS balances.

Example

↓	M10	Query of the current display mode
↑	M10_A_3	Currently, the terminal displays weight and SmartTrac

M11 – Beeper: Query/Set Volume

Level 2

Description

You can use M11 to set the volume of the terminal beeper or query the current setting.

Syntax

Commands

M11	Query of the current beeper volume.
M11_<BeeperVolume>	Set the beeper volume.

Balance Responses

M11_A_<BeeperVolume>	Current beeper volume.
M11_A	Command understood and executed successfully.
M11_I	Command understood but currently not executable.
M11_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<BeeperVolume>	Integer	0..100	Beeper volume in %

Comments

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).
- Technically speaking, it is possible to enter values with decimal places, but the balance will omit these and will only store the integer value.

Example

↓	M11	Query of the current beeper volume
↑	M11_A_50	Current beeper volume is set to 50%

M12 – Beeper: Beep

Level 2

Description

This command triggers an acoustic beep signal.

Syntax

Command

M12_<BeeperVariant>	Trigger beep.
---------------------	---------------

Balance Responses

M12_A	Command understood and executed successfully.
M12_I	Command understood but currently not executable.
M12_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<BeeperVariant>	Integer	0	Beep variant 1 (e.g. 1x beep)
		1	Beep variant 2 (e.g. high sound)
		2	Beep variant 3 (e.g. deep sound)

Comment

- You can set the volume of the beeper using [M11](#). However, this setting will not necessarily work on all balances.

Example

↓	M12_1	Trigger a beep variant 2
↑	M12_A	Beep has been triggered

M13 – Query/Set Touch Function

Level 2

Description

Use M13 to switch the display's touch function on and off, or query the current setting. The function keys and terminal keys (see **K** command) are not affected.

Syntax

Commands

M13	Query of the current touch function.
M13_<TouchFunction>	Set the touch function.

Balance Responses

M13_A_<TouchFunction>	Current touch function.
M13_A	Command understood and executed successfully.
M13_I	Command understood but currently not executable.
M13_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<TouchFunction>	Integer	0	Touch function is reduced on function keys and keys beside of screen
		1	Touch function of screen is switched on

Comments

- For additional information on the terminal display, please see the Manuals for the balances.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Examples

↓	M13_1	Switch on the touch function
↑	M13_A	Touch function is activated

M14 – List Available Languages

Level 2

Description

Lists all languages available on the balance.

Syntax

Command

M14	Query of the available languages.
-----	-----------------------------------

Balance Responses

M14_B_<No>"<Language>"	First language.
M14_B...	...
M14_A_<No>"<Language>"	Last language.
M14_I	Command understood but currently not executable.
M14_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<No>	Integer	0..max	Language number
<Language>	Integer		Language name
		Values	Meaning
		0	English
		1	German
		2	French
		3	Spanish
		4	Italian
		5	Russian
		6	Katakana
		7	Reserved
		8	Polish
		9	Reserved
		10	Czech
		11	Hungarian
		12..15	Reserved
		16	Chinese
		17	Japanese

Example

↓	M14	Query of the available languages
↑	M14_B_0_"English"	No 0 language is English
↑	M14_B_1_"Deutsch"	No 1 language is Deutsch
↑	M14_B_2_"Français"	No 2 language is Français
↑	M14_B_3_"Español"	No 3 language is Español
↑	M14_B_4_"Italiano"	No 4 language is Italiano
↑	M14_B_5_"Russian"	No 5 language is Russian
↑	M14_B_6_"Katakana"	No 6 language is Katakana
↑	M14_B_8_"Polski"	No 8 language is Polski
↑	M14_B_10_"Cestina"	No 10 language is Cestina
↑	M14_A_11_"Magyar"	No 11 language is Magyar
↑	M14_A_16_"Chinese"	No 11 language is Chinese
↑	M14_A_17_"Japanese"	No 11 language is Japanese

M15 – Query/Set Language

Level 2

Description

Use M15 to set the required language for the terminal dialog boxes or to query the current language setting.

Syntax

Commands

M15	Query of the current language.
M15_<No>	Set the language.

Balance Responses

M15_A_<No>	Current language.
M15_A	Command understood and executed successfully.
M15_I	Command understood but currently not executable.
M15_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<No>	Integer	0.. max. languages	Language number

Comments

- Language number: Number of the language according to the available languages (see command [M14](#)).
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Examples

↓	M15_3	Set the language number 3
↑	M15_A	Language 3 is set

↓	M15	Query of the current language
↑	M15_A_0	Language is set to the default language

See also

[M14 – List Available Languages](#)

M16 – Query/Set Standby Mode

Level 2

Description

You can use M16 to set the delay for switching to standby mode, or to query the current setting.

Syntax

Commands

M16	Query of the current standby mode.
M16_<StandbyMode>	Set the standby mode.

Balance Responses

M16_A_<StandbyMode>	Current standby mode.
M16_A	Command understood and executed successfully.
M16_I	Command understood but currently not executable.
M16_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<StandbyMode>	Integer	0	Standby mode switched off
		3	Standby 30 min.
		4	Standby 60 min.
		5	Standby 120 min.
		6	Standby 240 min.

Comments

- The balance switches to standby mode if it is not operated within 15 min. Operating the balance includes pressing a key, significant changes in weight, item counter > 0 or interface commands.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Examples

↓	M16_4	Set the standby mode to 60 min.
↑	M16_A	Standby mode is set

↓	M16	Query of the current standby mode
↑	M16_A_0	Current standby mode is switched off

M17 – ProFACT: Query/Set single Time Criteria

Level 2

Description

Use M17 to set the time and days when a proFACT adjustment should be executed automatically, or to query the current setting.

Syntax

Commands

M17	Query of the current proFACT time criteria.
M17_<hh>_<mm>_<ss>_<WeekDay>	Set the proFACT time criteria.

Balance Responses

M17_A_<hh>_<mm>_<ss>_<WeekDay>	Current proFACT time criteria.
M17_A	Command understood and executed successfully.
M17_I	Command understood but currently not executable.
M17_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<hh>	Integer	00..23	Hours
<mm>	Integer	00..59	Minutes
<ss>	Integer	00..59	Seconds
<WeekDay>	Integer	0	Time criteria is switched off
		2 ⁰ = 1	Monday
		2 ¹ = 2	Tuesday
		2 ² = 4	Wednesday
		2 ³ = 8	Thursday
		2 ⁴ = 16	Friday
		2 ⁵ = 32	Saturday
		2 ⁶ = 64	Sunday

Comments

- The days of the week are written in [binary code](#). Combinations of different days are expressed as the sum of the individual days.
- Only one time criterion can be set using M17; all other times are deactivated. [M32](#) must be used if you wish to set several different times.
- If two or more times are set ([M32](#) command), resulting in an M17 query, an M17_I response is generated.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Examples

↓	M17_12_00_00_5	Set the proFACT time criteria to Monday and Wednesday (5 = 1 + 4) at 12:00 h
↑	M17_A	ProFACT time criteria is set:
↓	M17	Query of the current proFACT time criteria
↑	M17_A_12_00_00_127	The balance will currently be adjusted every day (127 = 1 + 2 + 4 + 8 + 16 + 32 + 64) at 12:00 h

M18 – ProFACT/FACT: Query/Set Temperature Criterion

Level 2

Description

Use M18 to set the temperature criterion for triggering a proFACT adjustment, or to query the current value.

Syntax

Commands

M18	Query of the current proFACT/FACT temperature criterion.
M18_<TemperatureCriterion>	Set the proFACT/FACT temperature criterion.

Balance Responses

M18_A_<TemperatureCriterion>	Current standby mode.
M18_A	Command understood and executed successfully.
M18_I	Command understood but currently not executable.
M18_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<TemperatureCriterion>	Integer	0	Temperature criterion is switched off
		1	1 st setting is activated
		2	2 nd setting is activated
		3	3 rd setting is activated
		4	4 th setting is activated

Comments

- Temperature difference (Δ temp.) is defined as the criterion. The balance is automatically adjusted if the temperature of the balance changes by the defined temperature difference.
- The settings depend on the balance model and must, therefore, be checked in relation to the balance concerned. For additional information, see Operating Instructions>System settings>proFACT fully automatic adjustment function.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M18_1	Set the proFACT/FACT temperature criterion to the 1 st setting
↑	M18_A	1 st setting is activated

M19 – Query/Set Adjustment Weight

Level 2

Description

You can use M19 to set your external adjustment weight, or to query the current weight setting.

Syntax

Commands

M19	Query of the current adjustment weight.
M19_<Value>_<Unit>	Set the adjustment weight.

Balance Responses

M19_A_<Value>_<Unit>	Current adjustment weight.
M19_A	Command understood and executed successfully.
M19_I	Command understood but currently not executable.
M19_L	Command understood but not executable (incorrect parameter) or adjustment weight is too low.

Parameters

Name	Type	Values	Meaning
<Value>	Float	> 1% of capacity	Value of the adjustment weight
<Unit>	String		Weight unit of the adjustment weight = defined unit of the balance

Comments

- The adjustment weight must be entered in the defined unit of the balance. This unit can be found by entering a query command (M19 without arguments).
- The reference weight should be greater than 10% of the balance capacity; it must be greater than 1% of the capacity.
- Use **C2** to begin the adjustment procedure with the set weight.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).
For XS/XP only: If **M19** and **M20** set commands are used, adjust and test weight indices are set to 0. See **M95** for details.

Example

↓	M19	Query of the current adjustment weight
↑	M19_A_100.123_g	Current adjustment weight is 100.123 g

M20 – Query/Set Test Weight

Level 2

Description

You can use M20 to define your external test weight or query the currently weight setting.

Syntax

Commands

M20	Query of the current external test weight.
M20_<Value>_<Unit>	Set the external test weight.

Balance Responses

M20_A_<Value>_<Unit>	Current external test weight.
M20_A	Command understood and executed successfully.
M20_I	Command understood but currently not executable.
M20_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<Value>	Float	> 10 digits	Value of the external test weight
<Unit>	String		Weight unit of the external test weight = defined unit of the balance

Comments

- The adjustment weight must be entered in the defined unit of the balance. This unit can be found by entering a query command (M20 without arguments).
- Use **TST2** to begin the test procedure with the set weight.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

For XS/XP only: If **M19** and **M20** set commands are used, adjust and test weight indices are set to 0. See **M95** for details.

Example

↓	M20	Query of the current external test weight
↑	M20_A_100.123_g	Current external test weight is 100.123 g

M21 – Query/Set Unit

Level 2

Description

Use M21 to set the required weighing unit or query the current setting.

Syntax

Commands

M21	Query of the current unit.
M21_<Des>_<Unit>	Set the unit(s).

Balance Responses

M21_B_<Des>_<Unit> M21_B... M21_A_<Des>_<Unit>	Current first unit. ... Current last unit.
M21_A	Command understood and executed successfully.
M21_I	Command understood but currently not executable.
M21_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<Des>	Integer	0	Host unit, to MT-SICS (volatile Memory)
		1	Display unit, used on the terminal screen
		2	Info unit, used in the info field on the terminal screen
<Unit>	Integer	0	Gram g
		1	Kilogram kg
		2	reserved
		3	Milligram mg
		4	Microgram µg
		5	Carat ct
		6	reserved
		7	Pound avdp lb
		8	Ounce avdp oz
		9	Ounce troy ozt
		10	Grain GN
		11	Pennyweight dwt
		12	Momme mom
		13	Mesghal msg
		14	Tael Hongkong tlh
15	Tael Singapore tls		
16	Tael Taiwan tlt		
17	reserved		
18	Tola tola		

		19	Baht	baht	
		25	no unit	--	
		26	Piece	PCS	available with application "Counting"
		27	Percent	%	available with application "Percent"
		28	Custom unit 1	cu1	available if custom unit 1 is switched on (M22)
		29	Custom unit 2	cu2	available if custom unit 2 is switched on (M22)

Comments

- All **s*** commands (except **s*U**) are given in **Host unit** according to the definition of the MT-SICS. Only weight units are accepted as Host unit.
- It is not possible to use "no unit" for the displayed unit.
- In the event of a power failure, the host unit is lost and, following a restart, the weighing unit is displayed as "g".
- The units and/or their notation may be different in older software versions.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Examples

↓	M21_0_1	Set the unit to 1 kg
↑	M21_A	Unit is set

↓	M21	Query of the current unit
↑	M21_B_0_0 M21_B_1_3 M21_A_2_5	Current host unit is "g" Current display unit is "mg" Current info unit is "carat"

M22 – Query/Set Custom Unit Definitions

Level 2

Description

You can use M22 to set your own custom unit or query the currently defined custom unit.

Syntax

Commands

M22	Query of the current custom unit definitions.
M22_<No>_<Formula>_<Factor>_<Unit>_<Rounding>	Set the custom unit(s).

Balance Responses

M22_B_<No>_<Formula>_<Factor>_<Unit>_<Rounding> M22_B... M22_A_<No>_<Formula>_<Factor>_<Unit>_<Rounding>	Current first custom unit. ... Current last custom unit.
M22_A	Command understood and executed successfully.
M22_I	Command understood but currently not executable.
M22_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<No>	Integer	0.. max. units	Number of custom unit
<Formula>	Integer	0	(net weight) x factor
		1	factor/(net weight)
<Factor>	Float		Factor
<Unit>	String		Unit name (max. 4 characters)
<Rounding>	Float		Rounding step

Comments

- To query or define a custom unit, it must be switched on (M21).
- The parameter settings will be saved and the only way to reset the default values will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M22	Query of the current custom unit definitions
↑	M22_B_1_0_15.5_ "sfr" _0.05	Current first custom unit is (net weight) x 15.5 sfr, rounded to 0.05
	M22_A_2_1_25.4_ "h1" _0.1	Current second custom unit is 25.4/(net weight) h1, rounded to 0.1

M23 – Query/Set Readability, 1d/xd

Level 2

Description

Use M23 to set how many **digits** of the weighing result should be displayed or output and whether the value should be rounded, or to query the current value setting.

Syntax

Commands

M23	Query of the current readability.
M23_<Readability>	Set the readability.

Balance Responses

M23_A_<Readability>	Current readability.
M23_A	Command understood and executed successfully.
M23_I	Command understood but currently not executable.
M23_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<Readability>	Integer	0	1 d
		1	10 d
		2	100 d
		3	1000 d
		4	2 d
		5	5 d

Comments

- It is the balance model that determines which parameters can be changed.
- The readability is specified in **digits [d]** – this is the smallest increment a balance may display.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M23_1	Set the readability to 10 d
↑	M23_A	Readability is set

M24 – Query/Set Print Key Function

Level 2

Description

Use M24 to set the print key function or query the current setting.

Syntax

Commands

M24	Query of the current print key function.
M24_<PrintKeyFunction>	Set the print key function.

Balance Responses

M24_A_<PrintKeyFunction>	Current print key function.
M24_A	Command understood and executed successfully.
M24_I	Command understood but currently not executable.
M24_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<PrintKeyFunction>	Integer	0	Print stable weight
		1	Print weight immediately
		2	No print function

Comment

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M24_1	Set the print key function to "print weight immediately"
↑	M24_A	Print key function is set

M25 – List Applications

Level 2

Description

Use M25 to list all the applications available on the balance.

Syntax

Command

M25	Query of the available applications.
-----	--------------------------------------

Balance Responses

M25_B_<No>_ "<Application>"	First application.
M25_B...	...
M25_A_<No>_ "<Application>"	Last application.
M25_I	Command understood but currently not executable.
M25_L	Command understood but not executable.

Parameters

Name	Type	Values	Meaning
<No>	Integer	0.. max. applications	Application number
<Application>	String		Application name

Comment

- It is the balance model that determines which parameters can be used.

Example

↓	M25	Query of the available applications
↑	M25_B_0_"Weighing"	Default application is Weighing
↑	M25_B_1_"Piececounting"	Application 1 is Counting
↑	M25_B_2_"Percent"	Application 2 is Percent
↑	M25_B_3_"Formulation"	Application 3 is Formulation
↑	M25_B_7_"Density"	Application 7 is Density
↑	M25_B_9_"Diff.weighing"	Application 9 is DiffWeigh
↑	M25_B_11_"Statistic"	Application 11 is Statistic
↑	M25_A_17_"LabX Client"	Application 17 is LabX Client

M26 – Query/Set Current Application

Level 2

Description

Use M26 to select the required application or query the current selection.

Syntax

Commands

M26	Query of the current application selection.
M26_<No>	Set the application number.

Balance Responses

M26_A_<No>	Current application selection.
M26_A	Command understood and executed successfully.
M26_I	Command understood but currently not executable.
M26_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<No>	Integer	0.. max. applications	Application number

Comment

- Application number: Number of the application according to the application list (command [M25](#)).

Example

↓	M26_3	Set the application number 3
↑	M26_A	Application 3 is set

See also

[M25 – List Applications](#)

M27 – Query Adjustment History

Level 2

Description

Use M27 to call up the adjustment history.

Syntax

Command

M27	Query of the adjustment history.
-----	----------------------------------

Balance Responses

M27_B_<No>_<DD>_<MM>_<YYYY>_<hh>_<mm>_<Mode>_ " <Wgt> " M27_B... M27_A_<No>_<DD>_<MM>_<YYYY>_<hh>_<mm>_<Mode>_ " <Wgt> "	1 st adjustment entry. ... last adjustment entry.
M27_I	Command understood but currently not executable.
M27_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<No>	Integer	0.. max. entries	Number of the adjustment entry
<DD>	Integer	01..31	Date, day
<MM>	Integer	01..12	Date, month
<YYYY>	Integer		Date, year
<hh>	Integer	0..23	Time, hour
<mm>	Integer	0..59	Time, minute
<Mode>	Integer	0	Internal adjustment
		1	External adjustment
<Wgt>	Float		Weight of the adjustment weight used

Example

↓	M27	Query of the adjustment history
↑	M27_B_1_01_01_2011_08_26_0_ " "	1 st adjustment, performed at 01.01.2011, 08:26 h, internal adjustment
↑	M27_B_2_14_12_2010_14_30_1_ "200.1234_g"	2 nd adjustment, performed at 14.12.2010, 14.30 h, external adjustment, weight 200.1234 g
↑	M27_A_3_14_12_2010_8_26_1_ "200.1234_g"	3 rd adjustment, performed at 14.12.2010, 08:26 h, external adjustment, weight 200.1234 g

M29 – Query/Set Value Release

Level 2

Description

Use M29 to define the weight value release or query the current setting.

Syntax

Commands

M29	Query of the current value release setting.
M29_<ValueRelease>	Set the value release.

Balance Responses

M29_A_<ValueRelease>	Current value release.
M29_A	Command understood and executed successfully.
M29_I	Command understood but currently not executable.
M29_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<ValueRelease>	Integer	0	Very fast
		1	Fast
		2	Reliable and fast
		3	Reliable
		4	Very reliable

Comments

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).
- Not all balances offer the complete range of settings. If a setting is made that is not supported by the balance, an error message is issued (M29_L).

Example

↓	M29_3	Set the value release to "reliable"
↑	M29_A	Value release is set

M32 – ProFACT: Query/Set Time Criteria

Level 2

Description

You can use M32 to set several times at which an automatic proFACT adjustment procedure should be carried out, or query the current settings. The days of the week are defined with M33.

Syntax

Commands

M32	Query of the current proFACT time criteria.
M32_<No>_<hh>_<mm>_<status>	Set the proFACT time criteria.

Balance Responses

M32_B_<No>_<hh>_<mm>_<status> ... M32_A_<No>_<hh>_<mm>_<status>	Current proFACT time criteria.
M32_A	Command understood and executed successfully.
M32_I	Command understood but currently not executable.
M32_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<No>	Integer	0..3	FACT time index
<hh>	Integer	00..23	Hours
<mm>	Integer	00..59	Minutes
<status>	Integer	0	Time deactivated (off)
		1	Time activated (on)

Comments

- Only 1 time criterion can be set using M17; all other times are permanently deactivated. M32 and M33 must be used if you wish to set several different times.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Examples

↓	M32_2_12_00_1	FACT time 2 set to 12:00 and activated (on).
↑	M32_A	ProFACT time criteria is set:

↓	M32	Query of the current proFACT time criteria
↑	M32_B_1_09_00_1	The balance will currently be adjusted at 9:00 h, 12:00 and 15:00 h
↑	M32_B_2_12_00_1	
↑	M32_A_3_15_00_1	

M33 – ProFACT: Query/Set Weekday

Level 2

Description

You can use M33 to set the days of the week on which a proFACT adjustment procedure should be carried out, or to query the current settings. The times for each are defined using M32.

Syntax

Commands

M33	Query of the current proFACT Weekday.
M33_<Weekday>	Set the proFACT Weekday.

Balance Responses

M33_A_<Weekday>	Current proFACT Weekday.
M33_A	Command understood and executed successfully.
M33_I	Command understood but currently not executable.
M33_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<WeekDay>	Integer	0	Time criteria is switched off
		2 ⁰ = 1	Monday
		2 ¹ = 2	Tuesday
		2 ² = 4	Wednesday
		2 ³ = 8	Thursday
		2 ⁴ = 16	Friday
		2 ⁵ = 32	Saturday
		2 ⁶ = 64	Sunday

Comments

- The days of the week are written in [binary code](#). Combinations of different days are expressed as the sum of the individual days.
- Only 1 time criterion can be set using [M17](#); all other times are deactivated. [M32](#) and M33 must be used if you wish to set several different times.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M33_5	Time adjustments are made on Mondays and Wednesdays (5 = 1 + 4)
↑	M33_A	ProFACT Weekday is set.

M34 – MinWeigh: Query/Set Method

Level 2

Description

Use M34 to select the MinWeigh method you wish to work with, or query the currently set MinWeigh method.

Syntax

Commands

M34	Query of the current MinWeigh Method.
M34_<Method>	Set the MinWeigh Method.

Balance Responses

M34_A_<Method>	Current MinWeigh Method.
M34_A	Command understood and executed successfully.
M34_I	Command understood but currently not executable.
M34_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<Method>	Integer	0	MinWeigh deactivated
		1	Method 1 activated
		2	Method 2 activated
		3	Method 3 activated

Comments

- MinWeigh can only be activated by a service technician.
- For additional information on Minimum Weight (MinWeigh), see the Operating Instructions of the balance.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M34_1	Set the MinWeigh Method to 1.
↑	M34_A	MinWeigh Method 1 is set.

M35 – Query/Set Zeroing Mode at Start-Up

Level 2

Description

You can use M35 to save the last zero. Following a power failure, the balance will resume operation with the saved zero. In normal mode (M35_0), the balance specifies a new zero reference point at start-up as soon as a stable condition has been achieved.

Syntax

Commands

M35	Query of the current Zeroing Mode at Start-Up.
M35_<Mode>	Set the Zeroing Mode at Start-Up.

Balance Responses

M35_A_<Mode>	Current Zeroing Mode at Start-Up.
M35_A	Command understood and executed successfully.
M35_I	Command understood but currently not executable.
M35_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<Mode>	Integer	0	Normal mode
		1	Start with saved zero or save last zero

Comments

- If the mode is set to 1 when the balance is started up, the fail-safe, saved zero is used.
- For certification reasons, this command may only be executed on normal balances. Certifiable balances do not have this function.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M35_1	Save the last zero and use it at following start-up
↑	M35_A	Start-up Zeroing Mode is set

M36 – Query/Set LevelControl Setup

Level 2

Description

Use M36 to define the LevelControl and its responses, or query the current settings.

Syntax

Commands

M36	Query of the current LevelControl settings.
M36_<Status>_<Text>_<Sound>	Set the LevelControl.

Balance Responses

M36_A_<Status>_<Text>_<Sound>	Current settings of the LevelControl.
M36_A	Command understood and executed successfully.
M36_I	Command understood but currently not executable.
M36_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<Status>	Integer	0	LevelControl switched off
		1	LevelControl switched on
<Text>	Integer	0	Warning text: Off
		1	Warning text: Once
		2	Warning text: Repeat
<Sound>	Integer	0	Warning beep: Off
		1	Warning beep: Once
		2	Warning beep: Repeat

Comments

- This command is not supported by weighing platforms without terminal.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M36	Query of the current LevelControl settings
↑	M36_A_1_1_1	LevelControl is switched on with warning text once and beep once

See Also

[I28 – Query Level Sensor](#)

M37 – Query/Set Draft Shield Door Opening Width

Level 2

Description

Use M37 to set the draft shield door opening width or query the current setting.

Syntax

Commands

M37	Query of the current draft shield door opening width.
M37_<DoorOpening>	Set the draft shield door opening width.

Balance Responses

M37_A_<DoorOpening>	Current setting of the draft shield door opening width.
M37_A	Command understood and executed successfully.
M37_I	Command understood but currently not executable.
M37_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<DoorOpening>	Integer	25	Door opening 25%
		50	Door opening 50%
		75	Door opening 75%
		100	Door open

Comments

- Only the values available in the set-up of the balance are accepted (25%, 50%, 75%, and 100%).
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M37	Query of the current draft shield door opening width
↑	M37_A_75	Door opening width is set to 75%

See Also

[WS – Query/Set Position of Draft Shield Doors](#)

M39 – SmartTrac: Query/Set Graphic

Level 2

Description

You can use M39 to set the type of SmartTrac graphic (used weighing range graphic) or query the current setting.

Syntax

Command

M39	Query of the current SmartTrac Graphic.
M39_<SmartTrac>	Set the SmartTrac Graphic.

Balance Responses

M39_A_<SmartTrac>	Current setting of the SmartTrac Graphic.
M39_A	Command understood and executed successfully.
M39_I	Command understood but currently not executable.
M39_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning (Nominal =0)	Meaning (Nominal > 0)
<SmartTrac>	Integer	0	No SmartTrac	Weighing-in graphic
		1	Round SmartTrac	Round weighing in SmartTrac
		2	SmartTrac bar	Weighing-in SmartTrac bar
		3	SmartTrac measuring beaker	SmartTrac crosshairs

Comments

- If the application contains a nominal value that is > 0, the used weighing range graphics mentioned above are automatically displayed as weighing-in graphics listed in the left-most column.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M39_2	Set the SmartTrac bar
↑	M39_A	SmartTrac bar has been set

M40 – Query/Set Inner Draft Shield Door Opening Width

Level 2

Description

Use M37 to set the inner draft shield door opening width or query the current setting.

Syntax

Commands

M40	Query of the current inner draft shield door opening width.
M40_<DoorOpening>	Set the inner draft shield door opening width.

Balance Responses

M40_A_<DoorOpening>	Current setting of the inner draft shield door opening width.
M40_A	Command understood and executed successfully.
M40_I	Command understood but currently not executable.
M40_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<DoorOpening>	Integer	25	Door opening 25%
		50	Door opening 50%
		75	Door opening 75%
		100	Door open

Comments

- Only the values available in the set-up of the balance are expected (25%, 50%, 75%, and 100%)
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or by means of a balance master reset (not @).

Example

↓	M40	Query of the current inner draft shield door opening width
↑	M40_A_50	Door opening width is set to 50%

See Also

[WS – Query/Set Position of Draft Shield Doors](#)

M43 – Custom Unit activate/deactivate

Level 2

Description

Use M43 to activate or deactivate Custom Units (Custom Unit1, Custom Unit2)

Syntax

Commands

Query

M43	Query of the current custom unit setting.
-----	---

Balance Responses

M43_B_<CustomUnitNumber>_<Value> M43_A_<CustomUnitNumber>_<Value>	Current settings of Custom Unit1 and Custom Unit2
--	---

Write

M43_<CustomUnitNumber>_<Value>	Write new custom unit.
--------------------------------	------------------------

Balance Responses

M43_A	Command understood and executed successfully.
M43_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<CustomUnitNumber>	Integer	1	Custom Unit1
		2	Custom Unit2
<Value>	Integer	0	Deactivate custom unit
		1	Activate custom unit

Comments

- Dependency: M21 - Query/Set Units (Host-, Display- and Info-Unit)
- M22 - Query/Set Custom Unit Definitions (Formula, Factor, Unit, Rounding)
- Custom Units cannot be fully defined or managed via Host.
- This command is available only in XP and XS balances.
- Only XA balances have Custom Units.

Examples

↓	M43	Query of current custom unit settings
↑	M43_B_1_1 M43_A_2_0	Custom Unit1 is on Custom Unit2 is off

↓	M43_1_0	Custom Unit1 is deactivated
↑	M43_A	Command understood and executed successfully

M47 – Read and write frequently changed test weight settings

Level 2

Description

This command is necessary to read and write the frequently changed test weight settings, such as actual weight and next calibration date – from/to the balance.

Syntax

Commands

Request the whole list of entries:

M47	Query of the current test weight settings.
-----	--

Balance Responses

<pre>M47_B_<TestWeightNr>_ <"ActualWeight">_<"Unit">_ <NextCalDay>_<NextCalMonth>_ <NextCalYear> ... M47_A_<TestWeightNr>_ <"ActualWeight">_<"Unit">_ <NextCalDay>_<NextCalMonth>_ <NextCalYear></pre>	Current test weight settings.
--	-------------------------------

Request for specific test weight settings from TestWeightNr 'X' only

M47_<TestWeightNr>	Query of the specific test weight setting.
--------------------	--

Balance Responses

<pre>M47_A_<TestWeightNr>_ <"ActualWeight">_<"Unit">_ <NextCalDay>_<NextCalMonth>_ <NextCalYear></pre>	Command understood and executed successfully.
--	---

Write

<pre>M47_<TestWeightNr>_ <"ActualWeight">_<"Unit">_ <NextCalDay>_<NextCalMonth>_ <NextCalYear></pre>	Write new test weight settings for the specific test weight.
--	--

Balance Responses

M47_A	Command understood and executed successfully.
M47_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<TestWeightNr >	Integer	1..12	Number of the test weight.
<ActualWeight>	String	Max 10 chars	Actual weight of the test weight.
<Unit>	String	Max 2 chars	Actual unit of the test weight.
<NextCalDay>	Integer	1..31	Day of the next calibration date.
<NextCalMonth>	Integer	1..12	Month of the next calibration date.
<NextCalYear>	Integer	2000..2099	Year of the next calibration date.

Comments

- These initial values are set by the GWP software on the balance.
 The parameter "TestWeightNr" corresponds with the "TestWeightNr" of M48 command.
 To write the infrequently changed parameters, the command "M48" is used.
 The following conditions must be met before a test weight is considered valid: if name is defined (max 20 characters), if weight value is defined (more than 0), and if unit is valid.
 This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M47	Query of the list for all test weight settings.
↑	M47_B_1_"100.0"_"g"_"12_10_2010 M47_B_2_"9.9999"_"g"_"19_08_2010 M47_B_3_"20.0001"_"g"_"10_12_2009 M47_B_4_"0"_"mg"_"12_09_2011 M47_B_5_"0"_"g"_"31_12_2099 M47_B_6_"0"_"g"_"31_12_2099 M47_B_7_"0"_"g"_"31_12_2099 M47_B_8_"0"_"g"_"31_12_2099 M47_B_9_"0"_"g"_"31_12_2099 M47_B_10_"0"_"g"_"31_12_2099 M47_B_11_"0"_"g"_"31_12_2099 M47_A_12_"0"_"g"_"31_12_2099	The first three test weight settings are defined correctly, the fourth weight is not completely defined (weight value is still 0) and the rest is not defined at all.
↓	M47_1	The parameters of the first test weight are requested.
↑	M47_A_1_"100.0"_"g"_"12_10_2010	The requested test weight has an actual value of 100 grams and the next recalibration is on the 12. October 2010.
↓	M47_1_"20.0"_"g"_"20_10_2011	Parameters of the first test weight are changed.
↑	M47_A	The test weight's actual weight is set to 20 grams and the next recalibration date to the 20. October 2011.

M48 – Read and write infrequently changed test weight settings

Level 2

Description

This command is necessary to read and write the infrequently changed test weight settings, such as actual weight and next calibration date – from/to the balance.

Syntax

Commands

Query:

M48	Query of the infrequently used test weight settings.
-----	--

Balance Responses

<pre>M48_B_<TestWeightNr>_ <"WeightName">_<"WeightID">_ <"WeightClass">_ <"WeightCertificate">_ <"WeightSetNo"> M48_B... M48_A_<TestWeightNr>_ <"WeightName">_<"WeightID">_ <"WeightClass">_ <"WeightCertificate">_ <"WeightSetNo"></pre>	Infrequently used test weight settings.
---	---

Query:

M48_<TestWeightNr>	Query of the specific infrequently used test weight setting.
--------------------	--

Balance Responses

<pre>M48_A_<TestWeightNr>_ <"WeightName">_<"WeightID">_ <"WeightClass">_ <"WeightCertificate">_ <"WeightSetNo"></pre>	Command understood and executed successfully.
---	---

Write

<pre>M48_<TestWeightNr>_ <"WeightName">_<"WeightID">_ <"WeightClass">_ <"WeightCertificate">_ <"WeightSetNo"></pre>	Write new infrequently used test weight settings for the specific test weight.
---	--

Balance Responses

M48_A	Command understood and executed successfully.
M48_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<TestWeightNr >	Integer	1..12	Number of the test weight.
<WeightName>	String	Max 20 chars	Name of the test weight.
<WeightID>	String	Max 20 chars	ID of the test weight.
<WeightClass>	String	See Comments	Class of the test weight.
<WeightCertificate>	String	Max 20 chars	Certificate of test weight.
<WeightSetNo>	String	Max 20 chars	Set number of test weight.

Comments

The parameter "TestWeightNr" corresponds with the "TestWeightNr" of M47 command.

Examples for Weight classes: E1, E2, F1, F2, M1, M2, M3, ASTM1, ASTM2, ASTM3, ASTM4, ASTM5, ASTM6, ASTM7.

The following conditions must be met before a test weight is considered valid: if name is defined (max 20 characters), if weight value is defined (more than 0), and if unit is valid.

See also M47 command.

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M48	Query of the list for all infrequently used test weight settings
↑	<pre>M48_B_1_"50gQK"_"798012"_"E1"_"1231"_"4551" M48_B_2_"55gQK"_"798013"_"E1"_"1232"_"4552" M48_B_3_"60gQK"_"798014"_"E1"_"1233"_"4553" M48_B_4_"Test/Adj. Weight 4"_"_"E1"_"_"_" M48_B_5_"Test/Adj. Weight 5"_"_"E1"_"_"_" M48_B_6_"Test/Adj. Weight 6"_"_"E1"_"_"_" M48_B_7_"Test/Adj. Weight 7"_"_"E1"_"_"_" M48_B_8_"Test/Adj. Weight 8"_"_"E1"_"_"_" M48_B_9_"Test/Adj. Weight 9"_"_"E1"_"_"_" M48_B_10_"Test/Adj. Weight 10"_"_"E1"_"_"_" M48_B_11_"Test/Adj. Weight 11"_"_"E1"_"_"_" M48_A_12_"Test/Adj. Weight 12"_"_"E1"_"_"_"</pre>	The first three test weight settings that are infrequently used are defined correctly, and the rest is not defined at all.

↓	M48_1	The infrequently used parameters of the first test weight are requested.
↑	M48_A_1_"50gQK"_"798012"_"E1"_"5467"_"4556"	The actual test weight name of the requested test weight is 50gQK, the weight ID is 798012, the weight class is E1, the weight certificate is 5467 and the weight set number is 4556.
↓	M48_3_"100gQK"_"10988"_"F1"_"5991"_"4111"	Parameters of the third test weight are changed.
↑	M48_A	Command understood and executed successfully.

M50 – GWP Test Sequence

Level 2

Description

Use M50 command to read and write all parameters of a test sequence. The method parameters are not transmitted with this command. See comments about this command.

Syntax

Commands

Request the whole list of entries:

M50	Query of the test sequences.
-----	------------------------------

Balance Responses

<pre>M50_B_<TestSeqID>_<"Name">_ <PrepInstr>_<Method>_ <ActionIfFail>_<WarnMsg>_ <WarnInt>_<NoOfWarn>_ <Attempts>_<InstrIfFail>_ <"UnblockCode">_<EntryInHistory> ... M50_A_<TestSeqID>_<"Name">_ <PrepInstr>_<Method>_ <ActionIfFail>_<WarnMsg>_ <WarnInt>_<NoOfWarn>_ <Attempts>_<InstrIfFail>_ <"UnblockCode">_<EntryInHistory></pre>	Current settings of test sequences.
--	-------------------------------------

Request the parameters from test sequence ID 'X' only:

M50_<TestSeqID>	Query of the specific test sequence.
-----------------	--------------------------------------

Balance Responses

<pre>M50_A_<TestSeqID>_<"Name">_ <PrepInstr>_<Method>_ <ActionIfFail>_<WarnMsg>_ <WarnInt>_<NoOfWarn>_ <Attempts>_<InstrIfFail>_ <"UnblockCode">_<EntryInHistory></pre>	Command understood and executed successfully.
---	---

Write

<pre>M50_<TestSeqID>_<"Name">_ <PrepInstr>_<Method>_ <ActionIfFail>_<WarnMsg>_ <WarnInt>_<NoOfWarn>_ <Attempts>_<InstrIfFail>_ <"UnblockCode">_<EntryInHistory></pre>	Write new test sequence settings.
---	-----------------------------------

Balance Responses

M50_A_Validity	Command understood and executed successfully.
M50_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<TestSeqID>	Integer	1..12	ID of the test sequence.
<Name>	String	Max 20 chars	Name of the test sequence that can be freely defined.
<PrepInstr>	Integer	0 1	Defines if a preparation dialog is displayed and what kind of dialog it is. No dialog is displayed Standard dialog is displayed
<Method>	Integer	0..13	The method that is executed with this test sequence.
<ActionIfFail>	Integer	0 1 2	Defines the action that is executed, when the test sequence fails. No action Warning (a defined number of warnings are displayed to redo the test sequence) Attempts (a defined number of trials is allowed to redo the test sequence)
<WarnMsg>	Integer	0 1	Defines whether a warning dialog is displayed and what kind of dialog it is. Standard dialog is displayed Advanced dialog is displayed
<WarnInt>	Integer	0 1..1000	Defines the time (in hours) between two warning dialogs. Deactivated (this means that another action if failure than "Warning" is selected). Number of hours between two warnings.
<NoOfWarn>	Integer	0 1..1000	Defines how many warnings are displayed before the test sequence is blocked. Deactivated (this means that another action if failure than "Warning" is selected). Number of warnings.
<Attempts>	Integer	0 1 2 3 4	The maximum allowed number of attempts. Deactivated (this means that another action if failure than "Attempts" is selected). One attempt Two attempts Three attempts Attempts until test sequence is passed
<InstrIfFail>			Defines if an instruction dialog is displayed when the test

	Integer	0 1	sequence fails. No dialog is displayed Standard dialog is displayed
<UnblockCode>	String	Max 20 chars	The code that can be used to unblock the test sequence.
<EntryInHistory>	Integer	0 1	Defines if an entry is added to the GWP history, whenever this test sequence is executed. No entry is added to the history An entry is added to the history
<Validity>	Integer	0 1	Indicates whether the written test sequence is valid or not. Test sequence is invalid Test sequence is valid (the method must be valid that the test sequence can be valid)

Comments

These initial values are set by the GWP Software on the balance. The initial test sequence name is language-dependent.

Query: M50_1

Response: M50_A_1_"Test Sequence 1"_0_0_0_0_0_0_0_0_0_0_"Z"_0

Depending on which method the test sequence has assigned, the method must be defined with specific parameters. To set these parameters, the following commands are needed:

MT-SICS: M51 – GWP Eccentricity
M52 – GWP RP1, RPT1 and IRP1
M53 – GWP Service
M54 – GWP Weight Tolerances

The following parameters must be defined before a test sequence is considered valid: name (max. 20 characters), method type, method parameters, action if failure, number of action, number of warning, and interval.

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M50	Query of list of all test sequences.
↑	M50_B_1_"Sensitivity"_1_6_1_0_24_2_0_1_"12345"_1 M50_B_2_"Test Sequence 2"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_3_"Test Sequence 3"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_4_"Test Sequence 4"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_5_"Test Sequence 5"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_6_"Test Sequence 6"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_7_"Test Sequence 7"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_8_"Test Sequence 8"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_9_"Test Sequence 9"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_10_"Test Sequence 10"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_B_11_"Test Sequence 11"_0_0_0_0_0_0_0_0_0_0_"Z"_0 M50_A_12_"Test Sequence 12"_0_0_0_0_0_0_0_0_0_0_"Z"_0	The test sequence with ID 1 has already been configured, all other commands still have their initial values.
↓	M50_1	Query of the test sequence with the ID 1.
↑	M50_B_1_"Sensitivity"_1_5_1_0_24_2_0_1_"12345"_1	The first test sequence is called Sensitivity. Its method is SE1 and every time this method is

		<p>executed, a standard preparation dialog is displayed. If the test sequence fails, a warning dialog is displayed for a period of 24 hours, and it asks the user to redo the test sequence. This warning is displayed two times before the test sequence is blocked. If the test sequence is blocked, it can be unblocked by using the password "12345".</p>
--	--	---

↓	M50_1_"EC Test"_1_1_2_0_0_0_3_0_"1234"_1	Parameters of the third test sequence are changed.
↑	M50_A_1	The response indicates that the written test sequence is valid.

M51 – GWP Method EC**Level 2****Description**

This command is used for reading and writing the tolerances of the GWP method EC (Eccentricity). This command contains the tolerances for the eccentricity and the ID of the test sequence that uses this method. The test weight and its tolerances are transmitted using the command GWP Weight Tolerances. It is not required to request all EC parameters at a time. Therefore, it is only possible to read the EC parameters of a specific test sequence. When the command to request is used without the test sequence ID parameter, the command is invalid and will result in 'M51 L'.

Syntax**Commands****Query:**

M51_<TestSeqID>	Request the text from test sequence 'TestSeqID' only.
-----------------	---

Balance Responses

M51_A_<TestSeqID>_<"ECT1">_<"ECT1Unit">_<"ECT1Name">_<"ECT2">_<"ECT2Unit">_<"ECT2Name">	Current test sequence ID settings.
---	------------------------------------

Write

M51_<TestSeqID>_<"ECT1">_<"ECT1Unit">_<"ECT1Name">_<"ECT2">_<"ECT2Unit">_<"ECT2Name">	Write new test sequence settings.
---	-----------------------------------

Balance Responses

M51_A_Validity	Command understood and executed successfully.
M51_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<TestSeqID>	Integer	1..12	ID of the test sequence.
<ECT1>	String	Max 10 chars	Warn tolerance EC T1 for the eccentricity deviation.
<ECT1Unit>	String	Max 2 chars	Unit for the tolerance EC T1.
<ECT1Name>	String	Max 20 chars	Name of the tolerance EC T1.
<ECT2>	String	Max 10 chars	Warn tolerance EC T2 for the eccentricity deviation and its unit.
<ECT2Unit>	String	Max 2 chars	Unit for the tolerance EC T2
<ECT2Name>	String	Max 20	Name of the tolerance EC T2.

		chars	
<Validity>	Integer	0 1	Displays if the written parameters are valid or not (e.g. ECT1Name not empty). The written parameters are invalid. The written parameters are valid.

Comments

These initial values are set by the GWP software on the balance. The initial values for the tolerance names (ECT1Name and ECT2Name) are language dependent.

Query: M51_1

Response: M51_A_1_"0.001"_"g"_"Warning Limit"_"0.001"_"g"_"
"Control Limit"

If you wish to check the validity of the test sequence where this EC method was assigned, the command GWP Test Sequence can be used.

If the EC parameters are requested from an undefined test sequence or from a test sequence that has another method other than EC, the response will be 'M51 I'.

For LabTec Excellence, the data type of the text is ASCII due to limitations in handling UTF-8.

To check if the testsequence is valid, the command GWP Test Sequence is required.

MT-SICS: M50 – GWP Test Sequence

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M51_1	The EC parameters of the first test sequence are requested.
↑	M51_A_1_"1.0"_"mg"_"Warning Tolerance"_"4.0"_"mg"_" "Control Tolerance"	The deviation tolerances are 1 and 4 mg and their tolerance names are Warning- and Control Tolerance.
↓	M51_1_"0.002"_"g"_"Warn Limit"_"0.004"_"g"_" "Control Limit"	The EC parameters of the first test sequence are written.
↑	M51_A_1	The response indicates that the written tolerances are valid.

M52 – GWP RP1 and RPT1

Level 2

Description

This command is used for reading and writing the tolerances of the three GWP methods RP1 (Repeatability with one test weight) and RPT1 (Repeatability with a tare- and a test weight). These two methods have been summarized into one command because they both use the same parameters. This command contains the tolerances for the standard deviation and the ID of the test sequence that uses this method. The weights and their tolerances are transmitted using the command GWP Weight Tolerances. It is not required to request all RP parameters at a time. Therefore, it is only possible to read the RP parameters of a specific test sequence. When the command to request is used without the test sequence ID parameter, the command is invalid and will result in 'M52 L'.

Syntax

Commands

Query:

M52_<TestSeqID>	Query of repeatability parameters from test sequence ID 'X'.
-----------------	--

Balance Responses

M52_A_<TestSeqID>_<Method>_<NoOfRep>_<CheckLevelling>_<"ST1">_<"ST1Unit">_<"ST1Name">_<"ST2">_<"ST2Unit">_<"ST2Name">	Current test sequence ID settings.
---	------------------------------------

Write

M52_<TestSeqID>_<Method>_<NoOfRep>_<CheckLevelling>_<"ST1">_<"ST1Unit">_<"ST1Name">_<"ST2">_<"ST2Unit">_<"ST2Name">	Write new test sequence settings.
---	-----------------------------------

Balance Responses

M52_A_Validity	Command understood and executed successfully.
M52_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<TestSeqID>	Integer	1..12	The ID of the test sequence where these parameters have to be assigned.
<Method>	Integer	2 3	The method that is selected for the test sequence. Method RP1 Method RPT1
<NoOfRep>	Integer	2..15	Number of repetitions that are defined for this method 1.
<CheckLevelling>	Integer	0	Defines whether the level state is checked before the method is performed. Don not check the level state.

		1	Check the level state.
<ST1>	String	Max 10 chars	Tolerance s T1 for the standard deviation.
<ST1Unit>	String	Max 2 chars	Unit of the s T1 tolerance.
<ST1Name>	String	Max 20 chars	Name of the s T1 tolerance.
<ST2>	String	Max 10 chars	Tolerance s T2 for the standard deviation.
<ST2Unit>	String	Max 2 chars	Unit of the s T2 tolerance.
<ST2Name>	String	Max 20 chars	Name of the s T2 tolerance.
<Validity>	Integer	0 1	Indicates whether the written tolerances and number of repetitions are valid or not. The written tolerances and number of repetitions are invalid. The written tolerances and number of repetitions are valid.

Comments

These initial values are set by the GWP software on the balance. The initial values of the tolerance names (ST1Name and ST2Name) are language dependent.

Query: M52_1

Response: M52_A_1_2_10_0_0_0_0_0_0_0.000"_"g_"Warn Limit"_"0.000"_"g"_"Control Limit"

If you wish to check if the method is valid after setting these tolerances, the validity of the test sequence can be checked with the command GWP Test Sequence.

If the RP1 and RPT1 parameters are requested from an undefined test sequence or from a test sequence that has another method other than RP1 and RPT1, the response will be 'M52 L'.

Dependencies: MT-SICS: M50 – GWP Test Sequence

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M52_1	Query of standard deviation tolerances of the first test sequence.
↑	M52_A_1_2_4_0_0_0.001"_"g"_"Warn Limit"_"0.002"_"g"_"Control Limit"	The received data displays that this method has 4 repetitions and tolerances of 0.001 and 0.002 gram. It is not obvious what method type it is, but this can be found out by requesting the test sequence parameters of the same test sequence.
↓	M52_3_2_5_1_2.0"_"mg"_"Warn Limit"_"4.0"_"mg"_"Control Limit"	The parameters of a repeatability method are attached to the third test sequence. This repeatability method has 5 repetitions and tolerances of 2.0 and 4.0 mg. Before the method is started, the level state must be checked.
↑	M52_A_1	The written parameters are valid.

M53 – GWP Service

Level 2

Description

This command is used to transmit all parameters of the method Service. These parameters contain the early warning alert and the status that are monitored by the service method. It is not required to request all Service parameters at a time. Therefore, it is only possible to read the Service parameters of a specific test sequence. When the command to request is used without the test sequence ID parameter, the command is invalid and will result in 'M53 L'.

Syntax

Commands

Query:

M53_<TestSeqID>	Query of Service parameters from the test sequence ID 'X' only.
-----------------	---

Balance Responses

M53_A_<TestSeqID>_<Status>_<EarlyWarning>	Current settings of Service from test sequence ID.
---	--

Write

M53_<TestSeqID>_<Status>_<EarlyWarning>	Write new Service settings.
---	-----------------------------

Balance Responses

M53_A_Validity	Command understood and executed successfully.
M53_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<TestSeqID>	Integer	1..12	ID of the test sequence where this method is assigned.
<Status>	Binary	0 1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 ...	Defines which dates are monitored by this method. Status is not monitored Battery Change Date Next Service Date Next MinWeigh Date All Weight Calibration Dates Task 1 Task 2 Task 3 Task 4 Task 5 Task 6 Task 7 Task 8 Task 9 Task 10 Task 11 Task 12
<EarlyWarning>	Integer	1..365	The early warning alert defines how many days in advance the method warns.
<Validity>	Integer	0 1	Displays if the written parameters are valid or not. The written parameters are invalid. The written parameters are valid.

Comments

These initial values are set by the GWP software on the balance. As standard there are no objects selected for being monitored and the early warning alert is 7 days.

Query: M53_1
Response: M53_A_1_0_7

To check if the test sequence where this service method was assigned is valid, the command GWP Test Sequence has to be used.

If the Service parameters are requested from an undefined test sequence or from a test sequence that has another method than Service, the response will be 'M53 I'.

Dependencies: MT-SICS: M50 – GWP Test Sequence

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M53_2	Query of service parameters of the second test sequence.
↑	M53_A_2_27_5	<p>The method warns 5 days before a date expires. To find out which dates are monitored, the status parameter has to be analysed. In this example, the value is 27. The next lower bit significance from the different dates is 16 and applies to Task 1. This means that the execution date of Task 1 is monitored. The next step is to subtract the bit significance 16 from the value 27, so the new value is 11. The next lower bit significance is 8, which means that all weight calibration dates are monitored.</p> <p>Continue with these steps until the value of the status parameter is zero, then you have all dates that are monitored.</p>
↓	M53_1_13_7	<p>The service method has to be assigned to the first test sequence, the early warning time should be 7 days and the monitored dates should be Battery Change, MinWeigh and Weight Calibration.</p> <p>To get the value for the status parameter, the significances of all monitored dates have to be added. In this example the bit significances are 1 (Battery Change), 4 (MinWeigh) and 8 (Weight Calibration). These three significances added, result in the value 13 that must be written as status parameter.</p>
↑	M53_A_1	The response indicates that the written parameters are valid.

M54 – GWP Weight Tolerances

Level 2

Description

Use M54 command to read and write the test weight ID, the tolerance parameters and in addition, you can decide in which test sequence at which position (test weight or tare weight) the parameters should be saved. It is not required to request all weights and tolerances at a time. Therefore, it is only possible to read them from a specific test sequence or a specific test sequence and position. When the command to request is used without the test sequence ID parameter, the command is invalid and will result in 'M54 L.

Syntax

Commands

Query:

M54_<TestSeqID>	Query of Weight ID and tolerances from a specific test sequence ID.
-----------------	---

Balance Responses

<pre>M54_B_<TestSeqID>_<Method>_ <Position>_<TestweightID>_"T1"_ "T1Unit"_ "T1Name"_ "T2"_ "T2Unit"_ "T2Name " ... M54_A_<TestSeqID>_<Method>_ <Position>_<TestweightID>_"T1"_ "T1Unit"_ "T1Name"_ "T2"_ "T2Unit"_ "T2Name "</pre>	Current settings of the Weight ID and the tolerances from a specific test sequence ID.
--	--

or

Query:

M54_<TestSeqID>_<Position>	Query of Weight ID and tolerances from a specific test sequence ID and position.
----------------------------	--

Balance Responses

<pre>M54_A_<TestSeqID>_<Method>_ <Position>_<TestweightID>_"T1"_ "T1Unit"_ "T1Name"_ "T2"_ "T2Unit"_ "T2Name "</pre>	Current settings of the Weight ID and the tolerances from a specific test sequence ID and position.
--	---

Write

<pre>M54_<TestSeqID>_<Method>_ <Position>_<TestweightID>_"T1"_ "T1Unit"_ "T1Name"_ "T2"_ "T2Unit"_ "T2Name "</pre>	Write new settings of the Weight ID and the tolerances from a specific test sequence ID and position.
--	---

Balance Responses

M54_A_Validity	Command understood and executed successfully.
M54_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<TestSeqID>	Integer	1..12	The ID of the test sequence that has assigned the method with these tolerances.
<Method>	Integer	1 2 3 5 7 8 9 11	The method that is selected for the test sequence. EC Method RP1 Method RPT1 Method SE1 Method SE2 Method SET1 Method SET2 Method SERVICE Method
<Position>	Integer	0 1 2	Defines which tolerances are selected. Weight ID and tolerances of the tare weight (only available if the method uses a tare weight). Weight ID and tolerances of the first test weight. Weight ID and tolerances of the second test weight (only available if the method uses more than one test weight).
<TestweightID>	Integer	0 1..12	The ID of the test weight that is used in the method. No test weight selected. Test weight numbers defined.
<T1>	String	Max 10 chars	Warning tolerance T1.
<T1Unit>	String	Max 2 chars	Unit of the warning tolerance T1.
<T1Name>	String	Max 20 chars	Name of the T1 tolerance.
<T2>	String	Max 10 chars	Control tolerance T2.
<T2Unit>	String	Max 2 chars	Unit of the control tolerance T2.
<T2Name>	String	Max 20 chars	Name of the tolerance T2.
<Validity>	Integer	0 1	Displays if the selected test weight and the tolerances are valid or not. The test weight or the corresponding tolerances are invalid. The test weight and the corresponding tolerances are valid.

Comments

These initial values are set from the GWP software on the balance. The initial names of the tolerances (T1Name and T2Name) are language-dependent.

Query: M54_1_0
 Response: M54_A_1_0_0_0_0_0_0"0.001"_ "g"_ "Warn Limit"_ "0.001"_ "g"_ "Control Limit"

After writing the tolerance parameters, it is recommended to check if the test sequence is valid.

If the M54 parameters are requested from an undefined test sequence or from a test sequence that has no M54 parameters, the response will be 'M54 I'.

For LabTec Excellence, the data type of the text is ASCII due to limitations in handling UTF-8.

Dependencies: If the selected test weight is invalid, it has to be configured correctly before the tolerances can be valid. To do this, the commands M47 and M48 can be used. If it is desired to check the validity of the test sequence after setting these tolerance parameters, the command GWP Test Sequence can be used.

MT-SICS: M47 – Read and write frequently changed test weight settings
 M48 – Read and write infrequently changed test weight settings
 M50 – GWP test sequence

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M54_1_0	Query of tolerances on the first position (tare weight tolerances) of the test sequence.
↑	M54_L	The response displays that the first test sequence has no tare weight tolerances. This means that the test sequence uses a method without tare weight.

↓	M54_1_1	Query of tolerances of the first test weight of the test sequence.
↑	M54_A_1_5_1_2_0_0_0"0.001"_ "g"_ "Warn Limit"_ "0.002"_ "g"_ "Control Limit"	The number 5 after the test sequence ID indicates that these tolerances pertain to SE1 method (for a list of all methods see GWP Intro). SE1 only uses one test weight, so no more tolerances need to be requested. The test weight that is used for this SE1 method is the weight with ID 2.

↓	M54_2_1	Query of tolerances of the second test sequence.
↑	M54_A_2_6_1_1_0_0_0"0.001"_ "g"_ "Warn Limit"_ "0.002"_ "g"_ "Control Limit"	The number 6 after the test sequence ID indicates that these tolerances pertain to SE2 method (for a list of all methods see GWP Intro). SE2 uses two test weights, so the tolerances for the second test weight need to be requested also. The first test weight that is used for this SE2 method is the weight with ID 1.

↓	M54_2_2	Because the upper example uses the method SE2, the tolerances for the second test weight of the method have to be requested. Therefore, the position in the Query is set to 2.
↑	M54_A_2_6_2_2_0_0_0"2.0"_ "mg"_ "Warn Limit"_ "4.0"_ "mg"_ "Control Limit"	The second test weight that is used for this SE2 method is the weight with ID 2. The tolerances for this test weight differ from those of the first test weight. They are now 2 and 4 mg.
↓	M54_1	Query of all test weights and tolerances of the first test

		sequence.
↑	M54_B_2_8_1_3_ "4.0" "mg" _ "Warn Limit" "8.0" "mg" _ "Control Limit" M54_A_2_8_2_5_ "2.0" "mg" _ "Warn Limit" "4.0" "mg" _ "Control Limit"	The first test weight of this SET1 method has the ID 3 and tolerances of 4 and 8 mg. The second test weight has the ID 5 and tolerances of 2 and 4 mg.
↓	M54_1_8_0_1_ "100.0" "%" _ "Warn Limit" "100.0" "%" _ "Control Limit"	Tolerances for the tare weight of a SET1 method are written. Method SET1 has the ID 8. It uses the weight with ID 1. Tolerances of the tare weight are set to 100%, so the measured weight of the tare weight does not affect the test result.
↑	M54_A_1	The response indicates that the written tolerances are valid.

M55 – GWP Task

Level 2

Description

Use M55 to read and write all parameters of a task, such as the ID of the used test sequence or the starting method.

Syntax

Commands

Query:

M55	Query of the whole list of entries.
-----	-------------------------------------

Balance Responses

<pre>M55_B_<TaskID>_<TestSeqID>_ <StartingMethod>_<FirstDay>_ <FirstMonth>_<FirstYear>_ <StartHour>_<StartMinute>_ <Interval>_<DefDays>_<DefUsers> ... M55_A_<TaskID>_<TestSeqID>_ <StartingMethod>_<FirstDay>_ <FirstMonth>_<FirstYear>_ <StartHour>_<StartMinute>_ <Interval>_<DefDays>_<DefUsers></pre>	Current settings of the of the list of entries.
--	---

or

Query:

M55_<TaskID>	Request the text from TaskID 'X' only.
--------------	--

Balance Responses

<pre>M55_A_<TaskID>_<TestSeqID>_ <StartingMethod>_<FirstDay>_ <FirstMonth>_<FirstYear>_ <StartHour>_<StartMinute>_ <Interval>_<DefDays>_<DefUsers></pre>	Current settings of specific task.
--	------------------------------------

Write

<pre>M55_<TaskID>_<TestSeqID>_ <StartingMethod>_<FirstDay>_ <FirstMonth>_<FirstYear>_ <StartHour>_<StartMinute>_ <Interval>_<DefDays>_<DefUsers></pre>	Write new settings of specific task.
--	--------------------------------------

Balance Responses

M55_A_Validity	Command understood and executed successfully.
M55_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<TaskID>	Integer	1..12	ID of the task.
<TestSeqID>	Integer	0..12 0	ID of the test sequence that is assigned to this task. No test sequence assigned.
<StartingMethod>	Integer	0 1 2 3	Defines how the task is started. Manual On User Change Interval On Power On
<FirstDay>	Integer	1..31	The first day that the task is executed (only effective if starting method is interval, e.g. 05 (Format: dd)).
<FirstMonth>	Integer	1..12	The first month that the task is executed (only effective if starting method is interval), e.g. 06 for June.
<FirstYear>	Integer	2000.. 2099	The first year that the task is executed (only effective if starting method is interval), e.g. 2009 (Format: yyyy).
<StartHour>	Integer	0..23	Hour at which the task is executed (only effective if starting method is interval), e.g. 08 (Format: hh).
<StartMinute>	Integer	0..59	Minute at which the task is executed (only effective if starting method is interval), e.g. 00 (Format: mm).
<Interval>	String	1..720	The interval in days in which the task is executed (only effective if starting method is interval).
<DefDays>	Binary	0 1 2 4 8 16 32 64 ... 127	Defines on which days the task can be executed. No defined day Monday Tuesday Wednesday Thursday Friday Saturday Sunday All days are active

<DefUsers>	Binary	0 1 2 4 8 16 32 64 128 ... 255	Defines which users are allowed to execute the task. User is deactivated Home User 1 User 2 User 3 User 4 User 5 User 6 User 7 All users are active
<Validity>	Integer	0 1	Displays if the written task is valid or not. The task can, for example, be invalid when the assigned test sequence is not valid. Task is invalid (when the task is invalid, the task is automatically set to disabled). Task is valid (it is necessary that the assigned test sequence is valid, otherwise the task can not be valid).

Comments

These initial values are set by the GWP software on the balance.

Query: M55_1

Response: M55_A_1_0_0_31_12_2099_08_00_1_127_255

After the parameters have been set (it does not matter if the task is valid or not), the task will be disabled automatically. If the assigned test sequence is invalid, the ID is set to 0.

Dependencies: To enable or disable a task, the command GWP Task State is needed and a task can only be valid when the assigned test sequence is also valid.

MT-SICS: M50 – GWP Test Sequence

M56 – GWP Task State

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M55	All tasks and their parameters are read from the balance.
↑	M55_B_1_3_2_02_11_2009_09_00_31_1_31 M55_B_2_2_1_31_12_2099_08_00_1_127_255 M55_B_3_0_0_31_12_2099_08_00_1_127_255 M55_B_4_0_0_31_12_2099_08_00_1_127_255 M55_B_5_0_0_31_12_2099_08_00_1_127_255 M55_B_6_0_0_31_12_2099_08_00_1_127_255 M55_B_7_0_0_31_12_2099_08_00_1_127_255 M55_B_8_0_0_31_12_2099_08_00_1_127_255 M55_B_9_0_0_31_12_2099_08_00_1_127_255 M55_B_10_0_0_31_12_2099_08_00_1_127_255 M55_B_11_0_0_31_12_2099_08_00_1_127_255 M55_A_12_0_0_31_12_2099_08_00_1_127_255	Only two tasks are defined. The first task is an interval task that starts every 31 days at 9 o'clock. The first execution date of this task is on 02.11.2009 and it can only be executed by the first user from Monday until Friday.

↓	M55_1	Query of the parameters of the first task.
↑	M55_A_1_1_1_31_12_2099_08_00_1_127_31	The first task has assigned the test sequence with the ID one. The task is executed on user change. Therefore, the first start date and the start time are not important. The task can be executed by every user on the days Monday until Friday.

↓	M55_2_1_2_31_12_2099_06_00_1_127_255	An interval task is written on the balance. The task is executed every day at 6:00. It can be executed by every user.
↑	M55_A_1	The response says that the task is valid. This indicates that all parameters and the assigned test sequence are valid.

M56 – GWP Task State

Level 2

Description

Use M56 command to read the state of a GWP task. This includes, for example, if the task is activated and when the next execution date is. Furthermore, the command allows to write the active state of a task. Only the task state can be set with this command, the other parameters are read only.

Syntax

Commands

Query:

M56	Query of the whole list of entries.
-----	-------------------------------------

Balance Responses

M56_B_<TaskID>_<ActiveState>_<ToDo>_<Mode>_<Day>_<Month>_<Year>_<Hour>_<Minute>_<Trials> ... M56_A_<TaskID>_<ActiveState>_<ToDo>_<Mode>_<Day>_<Month>_<Year>_<Hour>_<Minute>_<Trials>	Current settings of the list of entries.
---	--

or

Query:

M56_<TaskID>	Request the text from Task ID 'X' only.
--------------	---

Balance Responses

M56_A_<TaskID>_<ActiveState>_<ToDo>_<Mode>_<Day>_<Month>_<Year>_<Hour>_<Minute>_<Trials>	Current settings of the Task ID 'X'.
--	--------------------------------------

Write

M56_<TaskID>_<ActiveState>_	Write new settings of the Weight ID and the tolerances from a specific test sequence ID and position.
-----------------------------	---

Balance Responses

M56_A_ActiveState	Command understood and executed successfully.
M56_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<TaskID>	Integer	1..12	ID of the task.
<ActiveState>	Integer	0 1	Indicates whether the task is enabled or not. Task is disabled. Task is enabled.
<ToDo>	Integer	0	Defines in which Todo-State the task is. is not executed

		1 2	waits to be executed is being executed at the moment
<Mode>	Integer	0 1	The mode setting of the task (warning mode, when the task failed or was aborted). Task is in normal mode. Task is in warning mode.
<Day>	Integer	1..31	The day when the task is executed the next time or when the next warning is displayed, e.g. 05 (Format: dd).
<Month>	String	1..12	The month when the task is executed the next time or when the next warning is displayed, e.g. 06 for June.
<Year>	Integer	2000.. 2099	The year when the task is executed the next time or when the next warning is displayed, e.g. 2009 (Format: yyyy).
<Hour>	Integer	0..23	The hour when the task is executed the next time or when the next warning is displayed, e.g. 08 (Format: hh).
<Minute>	Integer	0..59	The minute when the task is executed the next time or when the next warning is displayed, e.g. 00 (Format: mm).
<Trials>	Integer	0..1000	The number of trials (warnings) that were displayed.

Comments

These initial values are set by the GWP Software on the balance:

Query: M56_1

Response: M56_A_1_0_0_0_31_12_2099_08_00_0

When a task can not be set to be enabled, it is possible that the task is not valid. In order to check this, the command GWP Task is necessary.

MT-SICS: M55 – GWP Task

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M56	Query of the state of all twelve tasks.
↑	M56_B_1_1_1_2_0_21_09_2009_08_00_0 M56_B_2_1_1_1_21_09_2009_08_00_2 M56_B_3_1_0_0_22_09_2009_16_00_0 M56_B_4_0_0_0_31_12_2099_08_00_0 M56_B_5_0_0_0_31_12_2099_08_00_0 M56_B_6_0_0_0_31_12_2099_08_00_0 M56_B_7_0_0_0_31_12_2099_08_00_0 M56_B_8_0_0_0_31_12_2099_08_00_0 M56_B_9_0_0_0_31_12_2099_08_00_0 M56_B_10_0_0_0_31_12_2099_08_00_0 M56_B_11_0_0_0_31_12_2099_08_00_0 M56_A_12_0_0_0_31_12_2099_08_00_0	The tasks 1, 2 and 3 are enabled, all others are disabled. The task with ID 1 is being executed, while the second task is waiting to be executed. The second task is in warning mode, indicating that the task has failed at least once and the number of trials suggests that the task already failed twice.
↓	M56_4	With this query command, the state of the fourth task is read from the balance.
↑	M56_A_4_1_0_1_01_10_2009_06_00_1	The fourth task is enabled and its next execution date is 01.10.2009 at 06:00. The task already failed once and therefore is in warning mode.

↓	M56_1_0	This way, the active state of a task can be changed.
↑	M56_A_0	The first task is set to disabled.

↓	M56_1_1	The first task is set to enabled.
↑	M56_A_0	Although the task should be enabled, the response is that the task is disabled. This happens when the task that should be enabled is not valid, for example because it has no test sequence assigned. To find out more about the possible error, read the task parameters using the command GWP Task.

M57 – GWP System State

Level 2

Description

Use M57 command to request the GWP system state. It provides information whether the system is blocked and if there are pending tasks to be executed.

Syntax

Commands

Query:

M57	Query of the GWP system state.
-----	--------------------------------

Balance Responses

M57_A_<GWPState>_<RunningTaskID>_<WaitingTasks>	Current state of GWP system.
---	------------------------------

Parameters

Name	Type	Values	Meaning
<GWPState>	Integer	0 1 2 3 4	Displays the current state of the system, for example that the system is in standby. System is in standby System is on Task is running Warm up is running System is blocked
<RunningTaskID>	Integer	0 1..12	ID of the task that is running. No task is running ID of the running Task
<WaitingTasks>	Integer	1..12	Number of tasks that are waiting to be executed.

Comments

These initial values are set by the GWP software on the balance. There are two possibilities, because the system could be in standby or it could be on.

Query: M57
Response: M57_A_0_0_0

or

Query: M57
Response: M57_A_1_0_0

The system state can not be changed with this command. It can only be requested.

This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M57	Query of the GWP system state.
↑	M57_A_4_1_2	The first parameter indicates that the system is blocked by a task with the ID 1. There are 2 other tasks that are waiting to be executed.

↓	M57	Query of the GWP system state.
↑	M57_A_3_0_1	The system is warming up and one task is waiting to be executed after the warm up.

M58 – GWP History Export

Level 2

Description

Use M58 command to export the history from the balance to a computer. All history entries of a specific method can be read at one time. Therefore, the request must contain the method type that should be exported. It is not possible to read all history entries at one time, because every entry with another method has a different amount of parameters. It would be impossible to export all these entries in one readable table. When the command to request is used without the method parameter, the response will list how many entries of which method are available. Methods that have no entries in the history will not be listed.

Syntax

Commands

Query:

M58	Query of the number of history entries per method.
-----	--

Balance Responses

M58_B_<Index>_<Method>_<NoOfEntries>	Current number of history entries per method.
--------------------------------------	---

or

Query:

M58_<Method>	Query of all history entries with method type "Method".
--------------	---

Balance Responses

M58_B_<Index>_<Method>_<TestSeqName">_<VersionNo">_<StartTime">_<EndDate">_<EndTime">_<Attempt">_<Summary"> M58_B_<Method Specific> ... M58_B_<Index>_<Method>_<TestSeqName">_<VersionNo">_<StartTime">_<EndDate">_<EndTime">_<Attempt">_<Summary"> M58_A_<Method Specific>	Current history entries with method type "Method".
---	--

Common Parameters to All Methods

Name	Type	Values	Meaning
<Index>	String	1..120	Each history entry has its own index, so it is easier to see which Common- and Specific parameters belong to the same history entry.
<Method>	Integer	1 2 3	Requested method type. EC Method RP1 Method RPT1 Method

		5 6 7 8 9 11	SE1 Method SE2 Method SE3 Method SET1 Method SET2 Method SERVICE Method
<NoOfEntries>	Integer	1..120	Number of history entries that exist from a method.
<TestSeqName>	String	Max 20 chars	Name of the test sequence.
<VersionNo>	String	Max 3 chars	Version number of the test sequence.
<StartDate>	String	Max 15 chars	Date when the test sequence was started. The date format is assumed from the balance's date format.
<StartTime>	String	Max 10 chars	Time when the test sequence was started. The time format is assumed from the balance's time format.
<EndDate>	String	Max 15 chars	Date when the test sequence was finished. The date format is assumed from the balance's date format.
<EndTime>	String	Max 10 chars	Time when the test sequence was finished. The time format is assumed from the balance's time format.
<Attempt>	String	Max 3 chars	Number of times that the test sequence was executed.
<Summary>	String	Max 30 chars	Summary of the test sequence result. The summary text is language dependent.

Comments

When the history is initialized, it is empty. When the number of entries are requested and no entry exists, the balance response is 'L'.

Query: M58
Response: M58_L

This command is available only in XP and XS balances and is not supported in XA balances.

Common to all methods:

The odd lines always contain the common parameters that are the same for every history entry, no matter which method is requested. The even lines contain the method specific parameters (see "Method Specific" on next page). The order of the history entries is defined as following: the newest entry is sent first and the oldest as last.

Example

↓	M58	Query of the number of history entries for each method.
↑	M58_B_1_5_4 M58_B_2_3_9 M58_B_3_2_6 M58_B_4_7_4 M58_A_5_11_2	There are entries for five methods. These methods are SE1, RPT1, RP1, SE3 and SERVICE. Each entry has its own index (1..5) and the last number in each line displays how many entries there are for the corresponding method.

Method Specific Parameters

The method specific parameters are sent in the even lines after the common parameters. Depending on which method is requested, the response sends the corresponding method specific parameters.

EC Method

Syntax

Commands

Query:

M58_1	Query of all history entries with method type EC.
-------	---

Balance Responses

<pre>M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_B_<Index>_<Method>_ "Temp"_"LevelState"_"P1"_" "P1Unit"_"P2"_"P2Unit"_"P3"_" "P3Unit"_"P4"_"P4Unit"_"P5"_" "P5Unit" ... M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_A_<Index>_<Method>_ "Temp"_"LevelState"_"P1"_" "P1Unit"_"P2"_"P2Unit"_"P3"_" "P3Unit"_"P4"_"P4Unit"_"P5"_" "P5Unit"</pre>	Current history entries with method type EC.
--	--

EC Method Parameters

Name	Type	Values	Meaning
<Index>	String	1..120	See 'Common to All Methods' for description.
<Method>	Integer	1..11	Requested method type. See 'Common to All Methods' for list of available methods. For a list of available methods see GWP Intro.
<Temp>	String	Max 5 chars	Temperature during the test sequence execution (in °C).
<LevelState>	String		Indicates the level state of the balance during the test. For a list of all available level states, see GWP Intro.
<P1>	String	Max 15 chars	Value of the center weight.
<P1Unit>	String	Max 2 chars	Unit of the center weight. For a list of all available units, please see GWP Intro.
<P2>	String	Max 15 chars	Eccentricity at P2 (left front).
<P2Unit>	String	Max 2 chars	Unit of the eccentricity at P2 (left front). For a list of all available units, please see GWP Intro.

RP1 Method

Syntax

Commands

Query:

M58_2	Query of all history entries with method type RP1.
-------	--

Balance Responses

<p>M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_B_<Index>_<Method>_"Temp" "LevelState" "ActualRepetitions" "Repeatability" "RepUnit_"_ "MeanValue" "MeanValueUnit" ... M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_A_<Index>_<Method>_"Temp" "LevelState" "ActualRepetitions" "Repeatability" "RepUnit_"_ "MeanValue" "MeanValueUnit"</p>	Current history entries with method type RP1.
--	---

RP1 Method Parameters

Name	Type	Values	Meaning
<Index>	String	1..120	See 'Common to All Methods' for description.
<Method>	Integer	1..11	Requested method type. See 'Common to All Methods' for list of available methods. For a list of available methods see GWP Intro.
<Temp>	String	Max 5 chars	Temperature during the test sequence execution (in °C).
<LevelState>	String		Indicates the level state of the balance during the test. For a list of all available level states, see GWP Intro.
<ActualRepetitions>	String	Max 2 chars	Number of repetitions that have been performed.
<Repeatability>	String	Max 15 chars	Repeatability of the balance.
<RepUnit>	String	Max 2 chars	Unit of the repeatability. For a list of all available units, please see GWP Intro.
<MeanValue>	String	Max 15 chars	Mean value of the measurements

<MeanValueUnit>	String	Max 2 chars	Unit of the mean value. For a list of all available units, please see GWP Intro.
-----------------	--------	----------------	---

Example

↓	M58_2	Query of all entries with the RP1 method.
↑	<pre> M58_B_1_2_2_Rep. Test_"2_" "26.10.2009_"08:00_" "26.10.2009_"08:04_"1_" "Passed With Warning" M58_B_1_2_2_23.2_"Levelled"_" "4_"0.00005_"g"_" "100.00020_"g" M58_B_2_2_2_Monthly Rep._"5"_" "23.10.2009_"06:00_" "23.10.2009_"06:10_"1_" "Passed" M58_B_2_2_2_23.2_"Levelled"_" "8_"0.00004_"g"_" "100.00010_"g" M58_B_3_2_2_Monthly Rep._"5"_" "23.09.2009_"06:00_" "23.09.2009_"06:01_"1_" "Passed" M58_A_3_2_2_23.8_"Levelled"_" "4_"0.00003_"g"_" "100.00009_"g" </pre>	<p>There are three history entries for the RP1 method. Two are from the test sequence "Monthly Rep." and the third from "Rep. Test".</p>

RPT1 Method

Syntax

Commands

Query:

M58_3	Query of all history entries with method type RPT1.
-------	---

Balance Responses

<pre>M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_B_<Index>_<Method>_"Temp"_ "LevelState"_ "ActualRepetitions"_ "Repeatability"_ "RepUnit"_ "MeanValue"_ "MeanValueUnit" ... M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_A_<Index>_<Method>_"Temp"_ "LevelState"_ "ActualRepetitions"_ "Repeatability"_ "RepUnit"_ "MeanValue"_ "MeanValueUnit"</pre>	Current history entries with method type RPT1.
--	--

RPT1 Method Parameters

Name	Type	Values	Meaning
<Index>	String	1..120	See 'Common to All Methods' for description.
<Method>	Integer	1..11	Requested method type. See 'Common to All Methods' for list of available methods. For a list of available methods see GWP Intro.
<Temp>	String	Max 5 chars	Temperature during the test sequence execution (in °C).
<LevelState>	String		Indicates the level state of the balance during the test. For a list of all available level states, see GWP Intro.
<ActualRepetitions>	String	Max 2 chars	Number of repetitions that have been performed.
<Repeatability>	String	Max 15 chars	Actual repeatability of the balance.
<RepUnit>	String	Max 2 chars	Unit of the repeatability. For a list of all available units, please see GWP Intro.
<MeanValue>	String	Max 15 chars	Mean value of the measurements

<MeanValueUnit>	String	Max 2 chars	Unit of the mean value. For a list of all available units, please see GWP Intro.
-----------------	--------	----------------	---

Example

↓	M58_3	Query of all entries with the RPT1 method.
↑	<pre>M58_B_1_3_"RPT1Test"_"2"_" "26.10.2009"_"08:00"_" "26.10.2009"_"08:04"_"1"_" "Passed With Warning" M58_B_1_3_"23.2"_"Levelled"_" "4"_"0.00003"_"g"_" "50.00060"_"g" M58_B_2_3_"Rep. with Tare"_" "5"_"23.10.2009"_"06:00"_" "23.10.2009"_"06:10"_"1"_" "Passed" M58_A_2_3_"23.8"_"Levelled"_" "8"_"0.00002"_"g"_" "50.00040"_"g"</pre>	<p>There are two entries from different test sequences called "RPT1Test" and "Rep. with Tare".</p>

SE1 Method

Syntax

Commands

Query:

M58_5	Query of all history entries with method type SE1.
-------	--

Balance Responses

<p>M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_B_<Index>_<Method>_"Temp"_ "ActualWeight"_ "Unit" ... M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_A_<Index>_<Method>_"Temp"_ "ActualWeight"_ "Unit"</p>	Current history entries with method type SE1.
--	---

SE1 Method Parameters

Name	Type	Values	Meaning
<Index>	String	1..120	See 'Common to All Methods' for description.
<Method>	Integer	1..11	Requested method type. See 'Common to All Methods' for list of available methods. For a list of available methods see GWP Intro.
<Temp>	String	Max 5 chars	Temperature during the test sequence execution (in °C).
<LevelState>	String		Indicates the level state of the balance during the test. For a list of all available level states, see GWP Intro.
<ActualWeight>	String	Max 15 chars	Actual weight that was measured.
<Unit>	String	Max 2 chars	Unit of the actual weight. For a list of all available units, please see GWP Intro.

Example

↓	M58_5	Query of all entries with the SE1 method.
↑	<pre> M58_B_1_5_"Sensitivity"_2_" 26.10.2009_"08:11_" 26.10.2009_"08:12_"2_" Passed" M58_B_1_5_"23.2_"Levelled_" 20.0001_"g" M58_B_2_5_"Sensitivity"_2_" 26.10.2009_"08:00_" 26.10.2009_"08:10_"1_" Failed" M58_A_2_5_"23.2_"Levelled_" 20.00012_"g" </pre>	<p>There are two entries from the test sequence called "Sensitivity". On the first time, the method failed. However, on the second time, it passed.</p>

SE2 Method

Syntax

Commands

Query:

M58_6	Query of all history entries with method type SE2.
-------	--

Balance Responses

<pre>M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_B_<Index>_<Method>_"Temp"_ "LevelState_"_<ActualWeight1">_ "Weight1Unit_"_<ActualWeight2">_ "Weight2Unit" ... M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_A_<Index>_<Method>_"Temp"_ "LevelState_"_<ActualWeight1">_ "Weight1Unit_"_<ActualWeight2">_ "Weight2Unit"</pre>	Current history entries with method type SE2.
--	---

SE2 Method Parameters

Name	Type	Values	Meaning
<Index>	String	1..120	See 'Common to All Methods' for description.
<Method>	Integer	1..11	Requested method type. See 'Common to All Methods' for list of available methods. For a list of available methods see GWP Intro.
<Temp>	String	Max 5 chars	Temperature during the test sequence execution (in °C).
<LevelState>	String		Indicates the level state of the balance during the test. For a list of all available level states, see GWP Intro.
<ActualWeight1>	String	Max 15 chars	Actual weight of the first test weight that was measured.
<Weight1Unit>	String	Max 2 chars	Unit of the first weight. For a list of all available units, please see GWP Intro.
<ActualWeight2>	String	Max 15 chars	Actual weight of the second test weight that was measured.
<Weight2Unit>	String	Max 2 chars	Unit of the second weight. For a list of all available units, please see GWP Intro.

Example

↓	M58_6	Query of all entries with the SE2 method.
↑	<pre> M58_B_1_6_"SE2 Test_"_2_" 27.10.2009_"_08:00_" 27.10.2009_"_08:05_"_1_" Passed" M58_B_1_6_"23.2_"_Levelled_" 20.0001_"_g_"_99.9999_"_g" M58_A_2_6_"SE2 Test_"_2_" 26.10.2009_"_08:00_" 23.10.2009_"_08:10_"_1_" Passed with Warning" </pre>	There are two entries from the test sequence called "SE2 Test".

SET1 Method

Syntax

Commands

Query:

M58_8	Query of all history entries with method type SET1.
-------	---

Balance Responses

<p>M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_B_<Index>_<Method>_"Temp"_ "LevelState"_ "TareWeight"_ "TareUnit"_ "ActualWeight"_ "WeightUnit" ... M58_B_<Common Parameters (see 'Common Parameters to all Methods')> M58_A_<Index>_<Method>_"Temp"_ "LevelState"_ "TareWeight"_ "TareUnit"_ "ActualWeight"_ "WeightUnit"</p>	Current history entries with method type SET1.
--	--

SET1 Method Parameters

Name	Type	Values	Meaning
<Index>	String	1..120	See 'Common to All Methods' for description.
<Method>	Integer	1..11	Requested method type. See 'Common to All Methods' for list of available methods. For a list of available methods see GWP Intro.
<Temp>	String	Max 5 chars	Temperature during the test sequence execution (in °C).
<LevelState>	String		Indicates the level state of the balance during the test. For a list of all available level states, see GWP Intro.
<TareWeight>	String	Max 15 chars	Value of the tare weight.
<TareUnit>	String	Max 2 chars	Unit of the tare weight. For a list of all available units, please see GWP Intro.
<ActualWeight>	String	Max 15 chars	Actual weight of the test weight.
<WeightUnit>	String	Max 2 chars	Unit of the test weight. For a list of all available units, please see GWP Intro.

Example

↓	M58_8	Query of all entries with the SET1 method.
↑	<pre> M58_B_1_8_"Sensitivity + Tare"_ "11"_ "30.10.2009"_ "08:00"_ "30.10.2009"_ "08:02"_ "2"_ "Passed" M58_B_1_8_"23.2"_ "Levelled"_ "200.0001"_ "g"_ "19.9998"_ "g" M58_B_2_8_"Sensitivity + Tare"_ "11"_ "29.10.2009"_ "08:00"_ "29.10.2009"_ "08:01"_ "1"_ "Aborted" M58_A_2_8_"23.2"_ "Levelled"_ "_" " " " " " " </pre>	The method was aborted the first time, so the weight values are undefined. However, it was successfully executed a day later.

SET2 Method

Syntax

Commands

Query:

M58_9	Query of all history entries with method type SET2.
-------	---

Balance Responses

<p>M58_B_<Common Parameters (see 'Common Parameters to all Methods')></p> <p>M58_B_<Index>_<Method>_"Temp"_ "LevelState"_ "TareWeight"_ "TareUnit"_ "ActualWeight1"_ "Weight1Unit"_ "ActualWeight2"_ "Weight2Unit"</p> <p>...</p> <p>M58_B_<Common Parameters (see 'Common Parameters to all Methods')></p> <p>M58_A_<Index>_<Method>_"Temp"_ "LevelState"_ "TareWeight"_ "TareUnit"_ "ActualWeight1"_ "Weight1Unit"_ "ActualWeight2"_ "Weight2Unit"</p>	Current history entries with method type SET2.
--	--

SET2 Method Parameters

Name	Type	Values	Meaning
<Index>	String	1..120	See 'Common to All Methods' for description.
<Method>	Integer	1..11	Requested method type. See 'Common to All Methods' for list of available methods. For a list of available methods see GWP Intro.
<Temp>	String	Max 5 chars	Temperature during the test sequence execution (in °C).
<LevelState>	String		Indicates the level state of the balance during the test. For a list of all available level states, see GWP Intro.
<TareWeight>	String	Max 15 chars	Value of the tare weight.
<TareUnit>	String	Max 2 chars	Unit of the tare weight. For a list of all available units, please see GWP Intro.
<ActualWeight1>	String	Max 15 chars	Actual weight of the first test weight.
<Weight1Unit>	String	Max 2 chars	Unit of the first test weight. For a list of all available units, please see GWP Intro.
<ActualWeight2>	String	Max 15 chars	Actual weight of the second test weight.

<Weight2Unit>	String	Max 2 chars	Unit of the second test weight. For a list of all available units, please see GWP Intro.
---------------	--------	----------------	---

Example

↓	M58_9	Query of all entries with the SET2 method.
↑	<pre> M58_B_1_9_"Sensitivity + Tare"_ "11"_"30.10.2009"_"08:00"_" "30.10.2009"_"08:03"_"2"_" "Passed" M58_B_1_9_"23.2"_"Levelled"_" "200.0001"_"g"_"19.9998"_"g"_" "10.0000"_"g" M58_B_2_9_"Sensitivity + Tare"_ "11"_"29.10.2009"_"08:00"_" "29.10.2009"_"08:01"_"1"_" "Aborted" M58_A_2_9_"23.2"_"Levelled"_" ""_"""_"""_"""_"""_" </pre>	<p>The method was aborted the first time, so the weight values are undefined. However, it was successfully executed a day later.</p>

SERVICE Method

Syntax

Commands

Query:

M58_11	Query of all history entries with method type SERVICE.
--------	--

Balance Responses

M58_B_<Common Parameters (see 'Common Parameters to all Methods')> ... M58_A_<Common Parameters (see 'Common Parameters to all Methods')>	Current history entries with method type SERVICE. Please note that the SERVICE method has no method specific parameters.
---	--

Example

↓	M58_11	Query of all entries with the SERVICE method.
↑	M58_B_1_11_"Service"_"2"_" "15.10.2009"_"15:00"_" "15.10.2009"_"15:05"_"1"_" "Passed" M58_B_2_11_"Service"_"2"_" "08.10.2009"_"15:00"_" "08.10.2009"_"15:15"_"1"_" "Passed" M58_A_3_11_"Service"_"2"_" "01.10.2009"_"15:00"_" "01.10.2009"_"15:01"_"1"_" "Aborted"	The SERVICE method was executed three times. The first time it was aborted, but in the following two weeks it was executed correctly.

M64 – GWP Test Sequence Version

Level 2

Description

Use M64 command to request the current version of a GWP test sequence. It is not possible to write a GWP test sequence version because the version of a test sequence is generated by the terminal software automatically, even if a change has been done.

Syntax

Commands

Query:

M64	Query of the version of all test sequences.
-----	---

Balance Responses

M64_B_<TestsequenceID>_<VersionNo> ... M64_A_<TestsequenceID>_<VersionNo>	Current version of all test sequences.
---	--

or

Query:

M64_TestsequenceID	Query for a version of a specific test sequence.
--------------------	--

Balance Responses

M64_A_<TestsequenceID>_<VersionNo>	Current version of the specific test sequence.
------------------------------------	--

Parameters

Name	Type	Values	Meaning
<TestsequenceID>	Integer	1..12	Test Sequence ID
<VersionNo>	Integer		Test sequence version

Comments

When a test sequence has not yet been modified, the version will be 0.
 This command is available only in XP and XS balances and is not supported in XA balances.

Examples

↓	M64	Query of all test sequence versions.
↑	M64_B_1_5 M64_B_2_3 M64_B_3_7 M64_B_4_3 M64_B_5_0 M64_B_6_0 M64_B_7_0 M64_B_8_0 M64_B_9_0 M64_B_10_0 M64_B_11_0 M64_A_12_0	The version number of all test sequences are shown.
↓	M64_1	Query of the version number of one test sequence.
↑	M64_A_1_5	The system is warming up and one task is waiting to be executed after the warm up.

M66 – Certified test weight settings

Level 2

Description

Use M66 command to write the certified test weight settings. It is used primarily for the Matrix-Code of the weight certificate of Mettler-Toledo. It allows to directly import the settings of a certified weight from the certificate into the balance and thus eliminates any typing errors.

Please note that for other applications, the command M47 and M48 would be the better choice.

Syntax

Commands

Query:

M66	Query of the data from one weight only.
-----	---

Balance Responses

M66_A_ "WeightID"_"WeightClass"_"WeightCertificate"_"ActualWeight"_"Unit"_"<NextCalDay>"_<NextCalMonth>"_<NextCalYear>	Current data from one weight only.
--	------------------------------------

Write:

M66_"WeightID"_"WeightClass"_"WeightCertificate"_"ActualWeight"_"Unit"_"<NextCalDay>"_<NextCalMonth>"_<NextCalYear>	Write data of one weight only.
---	--------------------------------

Balance Responses

M66_A	Command understood and executed successfully.
M66_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<WeightID>	String	Max 20 chars	ID of the test weight
<WeightClass>	String	Max 20 chars	Class of the test weight
<WeightCertificate>	String	Max 20 chars	Certificate of test weight
<ActualWeight>	String	Max 10 chars	actual weight of the test weight
<Unit>	String	Max 2 chars	Unit of the actual weight For a list of all available units, please see GWP Intro.
<NextCalDay>	Integer	1..31	Day of the next calibration date, e.g. 05 (Format: dd).
<NextCalMonth>	String	1..12	Month of the next calibration date, e.g. 11 (Format:

			mm).
<NextCalYear>	Integer	2000.. 2099	Year of the next calibration date, e.g. 2009 (Format: yyyy).

Comments

Query of whole list of entries is not possible. Use M47 and M48 to get information about all specific tests. The initial values are set by the software on the balance.

This command is available only in XP and XS balances and is not supported in XA balances.

Examples for Weight classes: E1, E2, F1, F2, M1, M2, M3, ASTM1, ASTM2, ASTM3, ASTM4, ASTM5, ASTM6, ASTM7

Product specific implementation:

Please note that this command has a product specific implementation.

E.g. XP/XS balances have 12 Test/Adj Weight. In order to successfully inquire or write certified test weight settings, it is necessary that a specific test weight menu should be opened first from the 12 test weight menus.

Examples

↓	M66	
↑	M66_A_"A-0926748"_"E1"_" "MT-089987"_"99.99807"_"g"_" 21_07_210	The query was uniquely defined for the balance, the balance responds with the inquired data.

↓	M66	
↑	M66_I	The device is not ready to read the test/adj. weight settings. (e.g. there are more than one Test / Adj. Weight available, therefore the query could not be answered. Please see product specific implementation).

↓	M66_"A-0926748"_"E1"_" "MT-089987"_"99.99807"_"g"_" 21_07_210	Write data on the balance.
↑	M66_A	The received data are valid and has been stored on the balance.

↓	M66_"A-0926748"_"E1"_" "MT-089987"_"99.99807"_"g"_" 21_07_210	Write data on the balance.
↑	M66_I	The device is not ready to read the test/adj. weight settings. (e.g. there are more than one Test / Adj. Weight available, therefore the query could not be answered. Please see product specific implementation).

M95 – Read and write test / adjust weight index to be used for TST2 and C2 commands

Level 2

Description

Use M95 command to read and write the test weight index and the adjust weight index to be utilized for TST2 and C2 commands.

TST2 and C2 always utilize the latest used test weight or adjust weight, regardless if the test or adjustment routine was called by MT-SICS TST2, C2 or called manually by a function key.

The M95 command makes it possible to read the index of the last used test weight and adjust weight. With the aid of M95, it is also possible to define the index for the next execution of TST2 or C2.

Each manually executed external test or external adjustment and each set of test or adjust weight value by M19/M20 updates the index automatically.

Syntax

Commands

Query:

Request the whole list of entries:

M95	Query of the entire list of entries.
-----	--------------------------------------

Balance Responses

M95_B_Index_WeightIndex	Current test weight index.
M95_A_Index_WeightIndex	Current adjust weight index.

Request the text from index 'X' only:

M95_Index	Query of the specific adjust or test weight index.
-----------	--

Balance Responses

M95_A_Index_WeightIndex	Current adjust or test weight index.
-------------------------	--------------------------------------

Write:

M95_Index_WeightIndex	Write new adjust or test weight index.
-----------------------	--

Balance Responses

M95_A	Command understood and executed successfully.
M95_L	Command understood but not executable (incorrect parameter).

Initial Values

In a new balance where no weight has been defined in the database and no weight value has been set by M19 and M20, the TST2 and C2 will be executed with the factory settings for the M19 and M20. Therefore, a M95 request will return the following values:

Query:

M95	Query of the entire list of entries.
-----	--------------------------------------

Balance Responses

M95_B_0_0	Test weight index value refers to the factory settings of M20.
M95_A_1_0	Adjust weight index value refers to the factory settings of M19.

Command specific error response (see Section 2.1.2.3 Error Messages for general error responses):

Write:

M95_1_3	Set the adjust weight index to 3.
---------	-----------------------------------

Balance Responses

M95_I	The weight 3 from the internal database is not valid and can not be selected at this moment.
-------	--

Comments

If M19 and M20 set commands are used, test and adjust weight indices are set to 0.

This command is not supported on XA balances.

Dependencies: MT-SICS: M19 – Query/Set Adjustment Weight
 M20 – Query/Set Test Weight
 TST2 – Start Test Function with External Weight
 C2 – Start Adjustment with External Weight
 M47 – Read and write frequently changed test weight settings
 M48 – Read and write infrequently changed test weight settings
 M66 – Certified test weight settings

Examples

↓	M95	Query of both test and adjust weight indices
↑	M95_B_0_5	The 5 th defined weight from the internal database is used for TST2.
↑	M95_A_1_1	The 1 st defined weight from the internal database is used for C2.

↓	M95_0	Query of test weight indices
↑	M95_A_0_5	The 5 th defined weight from the internal database is used for TST2.
↓	M95_0_3	Set the 3rd defined weight from the internal database for TST2.

↑	M95_A	Command understood and executed successfully.
↓	M95_0_0	Set the value defined by M20 or the default test weight value for TST2.
↑	M95_A	Command understood and executed successfully.
↓	M95_1_0	Set the value defined by M19 or the default adjust weight value for C2.
↑	M95_A	Command understood and executed successfully.

P100 – Printer Interface: Send Text

Level 2

Description

You can use P100 to send text to the printer.

Syntax

Command

P100_ "<PrintOutText>"	Send text to the printer.
------------------------	---------------------------

Balance Responses

P100_A	Command understood and executed successfully.
P100_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
P100_L	Command understood but not executable (e.g. printer is not switched on, not configured or not connected).

Parameter

Name	Type	Values	Meaning
<PrintOutText>	String		Print out "text" on the printer (24 characters/line)

Comments

- A sequence of maximum 24 characters is admissible as text.
- The command P100_ "<Text>" has no influence on the function of the printer as a record printer.
- The Printer has to be connected, switched on and configured.

Example

↓	P100_ "Hallo"	Print out "Hallo" on the printer
↑	P100_A	Print out has been started

P101 – Printer Interface: Send Weight Value

Level 2

Description

Use P101 to send a stable weight value to the printer. This corresponds to the command **S**.

Syntax

Command

P101	Send stable weight value to the printer, corresponds to the command S .
------	--

Balance Responses

P101_A	Command understood and executed successfully.
P101_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
P101_L	Command understood but not executable (e.g. printer is not connected).

Comments

- The command P101 has no influence on the function of the printer as a record printer.
- The draft shield closes with this command, when the "Door function" is set on "Automatic". It opens after sending a stable weight.
- The Printer has to be connected, switched on and configured.

Example

↓	P101	Send stable weight value to the printer
↑	P101_A	Print out has been started

P102 – Printer Interface: Send Weight Value Immediately

Level 2

Description

Use P102 to send the weight value to the printer immediately. This corresponds to the command **SI**.

Syntax

Command

P102	Send weight value to the printer immediately, corresponds to the command SI .
------	--

Balance Responses

P102_A	Command understood and executed successfully.
P102_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
P102_L	Command understood but not executable (e.g. printer is not connected).

Comments

- The command P102 has no influence on the function of the printer as a record printer.
- The draft shield does not close with this command.
- The Printer has to be connected, switched on and configured.




Example

↓	P102	Send weight value to the printer immediately
↑	P102_A	Print out has been started

PW – Piece Counting: Query/Set Piece Weight

Level 3

Description

Use **PW** to set the reference weight of 1 piece, which you can then use for the piece counting application. You can also use **PW** to query the reference weight that you have set using , ,  or **PW**.

Syntax

Commands

PW	Query of the piece weight for the piece counting application.
PW_<PieceWeightValue>_<Unit>	Set the piece weight for the according preset value. The unit should correspond to the unit actually set under host unit .

Balance Responses

PW_A_<PieceWeightValue>_<Unit>	Current piece weight value in unit actually set under host unit .
PW_A	Command understood and executed successfully.
PW_I	Command understood but currently not executable (balance is currently executing another command).
PW_L	Command understood but not executable (e.g. piece counting application is not active or parameter is incorrect).

Comments

- This command can only be used with the application "piece counting". For details on available applications and how to activate them see also [M25](#) and [M26](#).
- The range of the piece weight value is specified to the balance type.
- If a reference weight has been defined, the display unit automatically changes to PCS and can be queried with **S*U**.

Example

↓	PW_20.00_g	Set the piece weight of the piece counting application to 20.00 g
↑	PW_A	Piece weight value is set

PWR – Power On/Off

Level 2

Description

Use `PWR` to switch the balance on or off. When it is switched off, standby mode is activated.

Syntax

Command

<code>PWR_<OnOff></code>	Switch the balance on or off.
--------------------------------	-------------------------------

Balance Responses

<code>PWR_A</code>	Balance has been switched off successfully.
<code>PWR_A</code> <code>I4_A_ " <SNR> "</code>	Balance with the serial number "SNR" has been switched on successfully (see also I4 command).
<code>PWR_I</code>	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
<code>PWR_L</code>	Command understood but not executable.

Parameter

Name	Type	Values	Meaning
<code><OnOff></code>	Integer	0	Set the balance to standby mode
		1	Switch the balance on

Comment

- The balance response to [I4](#) appears unsolicited after switching the balance on.

S – Send Stable Weight Value

Level 0

Description

Use S to send a stable weight value, along with the [host unit](#), from the balance to the connected communication partner via the interface.

Syntax

Command

S	Send the current stable net weight value.
---	---

Balance Responses

S_S_<WeightValue>_<Unit>	Current stable weight value in unit actually set under host unit .
S_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring, or timeout as stability was not reached).
S_+	Balance in overload range.
S_-	Balance in underload range.

Comments

- The duration of the timeout depends on the balance type.
- To send the stable weight value in actually displayed unit, see [SU](#) command.
- The draft shield closes with this command, when the "Door function" is set on "Automatic". It opens after sending a stable weight.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	S	Send a stable weight value
↑	S_S_100.00_g	The current, stable ("S") weight value is 100.00 g

SI – Send Weight Value Immediately

Level 0

Description

Use **SI** to immediately send the current weight value, along with the **host unit**, from the balance to the connected communication partner via the interface.

Syntax

Command

SI	Send the current net weight value, irrespective of balance stability.
----	---

Balance Responses

S_S_<WeightValue>_<Unit>	Stable weight value in unit actually set under host unit .
S_D_<WeightValue>_<Unit>	Non-stable (dynamic) weight value in unit actually set under host unit .
S_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
S_+	Balance in overload range.
S_-	Balance in underload range.

Comments

- The balance response to the command **SI** is the last internal weight value (stable or dynamic) before receipt of the command **SI**.
- To send weight value immediately in actually displayed unit, see **SIU** command.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SI	Send current weight value
↑	S_D_129.07_g	The weight value is unstable (dynamic, "D") and is currently 129.07 g

SIR – Send Weight Value Immediately and Repeat

Level 0

Description

Use **SIR** to immediately send the current weight value, along with the [host unit](#), from the balance to the connected communication partner via the interface, but this time on a continuous basis.

Syntax

Command

SIR	Send the net weight values repeatedly, irrespective of balance stability.
-----	---

Balance Responses

S_S_<WeightValue>_<Unit>	Stable weight value in unit actually set under host unit .
S_D_<WeightValue>_<Unit>	Non-stable (dynamic) weight value in unit actually set under host unit .
S_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
S_+	Balance in overload range.
S_-	Balance in underload range.

Comments

- **SIR** is overwritten by the commands **S**, **SI**, **SR**, **@** and hardware break and hence cancelled.
- The number of weight values per second can be configured using **UPD**.
- To send weight value in actually displayed unit, see **SIRU** command.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SIR	Send current weight values at intervals
↑	S_D_129.07_g	The balance sends stable ("S") or unstable ("D") weight values at intervals
↑	S_D_129.08_g	
↑	S_S_129.09_g	
↑	S_S_129.09_g	
↑	S_D_114.87_g	
↑	S_...	

SIRU – Send Weight Value with Currently Displayed Unit Immediately and Repeat

Level 2

Description

Use `SIRU` to immediately send the current weight value, along with the displayed unit, from the balance to the connected communication partner via the interface and on a continuous basis.

Syntax

Command

SIRU	Send the current net weight value with currently displayed unit, irrespective of balance stability and repeat.
------	--

Balance Responses

<code>S_S_<WeightValue>_<Unit></code>	Stable weight value in currently displayed unit.
<code>S_D_<WeightValue>_<Unit></code>	Non-stable (dynamic) weight value in currently displayed unit.
<code>S_I</code>	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
<code>S_+</code>	Balance in overload range.
<code>S_-</code>	Balance in underload range.

Comment

- As the `SIR` command, but with currently displayed unit.
- The number of weight values per second can be configured using `UPD`.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SIRU	Query of the current weight value with currently displayed unit
↑	S_D_12.34_lb	Non-stable (dynamic) weight value of 12.34 lb
↑	S_D_12.44_lb	Non-stable (dynamic) weight value of 12.44 lb
↑	S_D_12.43_lb	Non-stable (dynamic) weight value of 12.43 lb

SIS – Send NetWeight Value with Actual Unit and Weighing Status

Level 2

Description

Use SIS to immediately send the current net weight value to the connected communication partner via the interface, along with the **host unit** and other information regarding the weighing status.

Syntax

Command

SIS	Send the current net weight value with actual unit and weighing status.
-----	---

Balance Responses

SIS_A_<Status>_ "<NetWeight>"_<Unit1>_<Readability>_<Step>_<Approv>_<Info>	At status 0 to 3.
SIS_A_<Status>_ "<Error>"	At status 4 to 6.
SIS_I	Command understood but currently not executable.

Parameters

Name	Type	Values	Meaning
<Status>	Integer	0	Stable weight value
		1	Dynamic weight value
		2	Stable inaccurate weight (MinWeigh)
		3	Dynamic inaccurate weight (MinWeigh)
		4	Overload
		5	Underload
		6	Error, not valid
<Netweight>	Float		Net weight value
<Unit1>	Integer	0	Gram g
		1	Kilogram kg
		2	reserved
		3	Milligram mg
		4	Microgram µg
		5	Carat ct
		6	reserved
		7	Pound avdp lb
		8	Ounce avdp oz
		9	Ounce troy ozt
		10	Grain GN
		11	Pennyweight dwt
		12	Momme mom
		13	Mesghal msg
		14	Tael Hongkong tlh
		15	Tael Singapore tls
		16	Tael Taiwan tlt
17	reserved		

		18	Tola	tola
		20	Baht	baht
<Readability>	Integer	0..6	Amount of decimal places	
<Step>	Integer	1	"1" step	
		2	"2" step	
		5	"5" step	
		10	"10" step	
		20	"20" step	
		50	"50" step	
		100	"100" step	
<Approv>	Integer	0	Standard balance, Not approved	
		1	e = d	
		10	e = 10 d	
		100	e = 100 d	
		-1	Unapproved with * in display	
<Info>	Integer	0	Without tare	
		1	Net with weighed tare	
		2	Net with stored tare	

Comment

- Can not be used with custom unit, piece counting (PCS) or percent weighing (%).
- This command has no effect on the other **s*** commands.
- The units and/or their notation may be different in older software versions.
- Relates to the host output interfaces. The weight unit is the **host unit**, not the displayed unit.
- Also supplies a weigh value for zeroing, adjusting and taring, and in the menu.

Examples

↓	SIS	Query of the current weight value with actual host unit and weighing status
↑	SIS_A_0_"100.00"_0_2_1_10_0	100.0(0) g

↓	SIS	Query of the current weight value
↑	SIS_A_1_"10.0"_5_2_50_0_2	10.0 ct, carat value, with step 50, in coarse range, with stored tare and unstable

↓	SIS	Query of the current weight value
↑	SIS_A_6_"Error7"	Error, not valid

↓	SIS	Query of the current weight value
↑	SIS_A_4_" "	Overload

SIU – Send Weight Value with Currently Displayed Unit Immediately

Level 2

Description

Use **SIU** to immediately send the current weight value, along with the displayed unit, from the balance to the connected communication partner via the interface.

Syntax

Command

SIU	Send the current net weight value with currently displayed unit, irrespective of balance stability.
-----	---

Balance Responses

S_S_<WeightValue>_<Unit>	Stable weight value in currently displayed unit.
S_D_<WeightValue>_<Unit>	Non-stable (dynamic) weight value in currently displayed unit.
S_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
S_+	Balance in overload range.
S_-	Balance in underload range.

Comment

- As the **SI** command, but with currently displayed unit.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SIU	Query of the current weight value with currently displayed unit
↑	S_D_12.34_lb	Non-stable (dynamic) weight value is 12.34 lb

SIUM – Send Weight Value with Currently Displayed Unit and MinWeigh Information Immediately Level 2

Description

Use `SIUM` to immediately send the current weight value, along with the displayed unit and MinWeigh information, from the balance to the connected communication partner via the interface.

Syntax

Command

<code>SIUM</code>	Send the current net weight value with currently displayed unit and MinWeigh Information, irrespective of balance stability.
-------------------	--

Balance Responses

<code>S_<Status>_<WeightValue>_<Unit></code>	Weight value in currently displayed unit.
<code>S_I</code>	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
<code>S_+</code>	Balance in overload range.
<code>S_-</code>	Balance in underload range.

Parameters

Name	Type	Values	Meaning
<code><Status></code>	String	S	Stable, net >= MinWeigh limit
		D	Dynamic, net >= MW limit
		M	Stable, net < MinWeigh limit
		N	Dynamic, net < MW limit
<code><WeightValue></code>	Float		Weight value
<code><Unit></code>	String		Currently displayed unit

Comment

- As the `SI` command, but with currently displayed unit and MinWeigh information.
- If the MinWeigh function is switched off, or is not available on the balance, it corresponds to the command `SIU`.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	<code>SIUM</code>	Query of the current weight value with currently displayed unit
↑	<code>S_D_123.34_mg</code>	Dynamic net weight displayed, greater than MW limit
↓	<code>SIUM</code>	Query of the current weight value with currently displayed unit
↑	<code>S_M_123.34_mg</code>	Stable net weight displayed, less than MinWeigh limit
↓	<code>SIUM</code>	Query of the current weight value with currently displayed unit
↑	<code>S_N_123.34_mg</code>	Dynamic net weight displayed, less than MW limit

SM0 – Dynamic Weighing: Cancel all SMx Commands

Level 3

Description

You can use SM0 to cancel any SMx commands that are in progress.

Syntax

Command

SM0	Cancel all SMx commands (except SM4).
-----	---------------------------------------

Balance Responses

SM0_A	Command understood and executed successfully.
SM0_I	Command understood but currently not executable (balance is currently executing another command).
SM0_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect).

Comments

- This command can only be used with the application "dynamic weighing". For details on available applications and how to activate them see also [M25](#) and [M26](#).
- Can only be executed if no weight is being applied respectively the "dynamic weighing" window has disappeared.

SM1 – Dynamic Weighing: Start Immediately and Send the Result

Level 3

Description

Use SM1 to start dynamic weighing immediately. The result is transferred via the interface once the weighing time has elapsed.

Syntax

Command

SM1	Start dynamic weighing immediately and transfer the result.
-----	---

First Balance Responses

SM1_A	Dynamic weighing has been started, wait for second response. During the weighing operation, e.g. until the second response, no further commands can be executed.
SM1_I	Command understood but currently not executable (balance is currently executing another command). No second response follows.
SM1_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect). No second response follows.

Further Balance Responses

SM_*_<WeightValue>_<Unit>	Transfer of the result completed successfully. Weight value corresponds to the result of the measurement cycle. The unit corresponds to the current weight unit in the display.
SM_+	Abort, overload during the measurement cycle.
SM_-	Abort, underload during the measurement cycle.
SM_I	The dynamic weighing has been aborted, e.g. with the "C" button.

Comments

- This command can only be used with the application "dynamic weighing". For details on available applications and how to activate them see also [M25](#) and [M26](#).
- The SM1 command does not test any minimum load.
- Set AutoStart off (see command [A07](#)), set AutoTare off (see command [A08](#)).
- The balance does not perform stability or plausibility checks for the start.
- The balance must be loaded, weighing starts immediately after SM1.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SM1	Start a dynamic weighing immediately and transfer the result
↑	SM1_A	Command understood, result follows
↑	SM_*_#####23.76_g	Result of the dynamic weighing is 23.76 g

SM2 – Dynamic Weighing: Start After a Minimum Load Is Exceeded and Send Result

Level 3

Description

Use SM2 to start dynamic weighing if the applied weight exceeds the specified minimum load. The result is transferred via the interface once the weighing time has elapsed.

Syntax

Command

SM2	Start a dynamic weighing automatically after the defined minimum load is exceeded and transfer the result (once).
-----	---

First Balance Responses

SM2_A	Dynamic weighing has been started, wait for second response. During the weighing operation, e.g. until the second response, no further commands can be executed.
SM2_I	Command understood but currently not executable (balance is currently executing another command). No second response follows.
SM2_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect). No second response follows.

Further Balance Responses

SM_*_<WeightValue>_<Unit>	Transfer of the result completed successfully. Weight value corresponds to the result of the measurement cycle. The unit corresponds to the current weight unit in the display.
SM_+	Abort, overload during the measurement cycle.
SM_-	Abort, underload during the measurement cycle.
SM_I	The dynamic weighing has been aborted, e.g. with the "C" button.

Comments

- This command can only be used with the application "dynamic weighing". For details on available applications and how to activate them see also [M25](#) and [M26](#).
- Set AutoStart on (see command [A07](#)).
- The minimum load can be defined in the dynamic weighing application setup, in the menu AutoStart.
- The single start standby is cancelled by the [SM0](#) and [@](#) commands before start of the weighing.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SM2	Start a dynamic weighing after the defined minimum load is exceeded and transfer the result
↑	SM2_A	Command understood, result follows
↑	SM_*_#####24.30_g	Result of the dynamic weighing is 24.30 g

SM3 – Dynamic Weighing: Start After a Minimum Load Is Exceeded, Send Result and Repeat

Level 3

Description

Use SM3 to start several dynamic weighing procedures in succession if the applied weights exceed the specified minimum load. The results are transferred via the interface once the weighing time has elapsed.

Syntax

Command

SM3	Start dynamic weighing automatically.
-----	---------------------------------------

First Balance Responses

SM3_A	Dynamic weighing has been started, wait for second response. During the weighing operation, no further commands can be executed.
SM3_I	Command understood but currently not executable (balance is currently executing another command). No second response follows.
SM3_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect). No second response follows.

Further Balance Responses

SM_*_<WeightValue>_<Unit>	Transfer of the result completed successfully. Weight value corresponds to the result of the measurement cycle. The unit corresponds to the current weight unit in the display. Further results follow when the start condition is fulfilled again.
SM_+	Abort, overload during the measurement cycle.
SM_-	Abort, underload during the measurement cycle.
SM_I	The dynamic weighing has been aborted, e.g. with the "C" button.

Comments

- This command can only be used with the application "dynamic weighing". For details on available applications and how to activate them see also [M25](#) and [M26](#).
- Set AutoStart on (see command [A07](#)).
- The minimum load can be defined in the dynamic weighing application setup in the menu AutoStart.
- The recurring establishment of the start standby is cancelled by the [SM0](#), [SM1](#), [SM2](#) and [@](#) commands.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SM3	Start dynamic weighing when weight drops below and afterwards above the defined minimum load, transfer the result and repeat the process.
↑	SM3_A	Command understood, results follow
↑	SM_*_#####25.83_g	Result of the 1 st dynamic weighing is 25.83 g
↑	SM_*_#####22.91_g	Result of the 2 nd dynamic weighing is 22.91 g
↑	SM_*_#####24.05_g	Result of the 3 rd dynamic weighing is 24.05 g

SM4 – Dynamic Weighing: Query/Set Time Interval

Level 3

Description

Use SM4 to set the time interval for a dynamic weighing procedure, or query the current time setting.

Syntax

Commands

SM4	Query of the time interval for the dynamic weighing application.
SM4_<DynWeighTimeInterval>	Set the time interval for the dynamic weighing application.

Balance Responses

SM4_A_<DynWeighTimeInterval>	Current time interval for the data acquisition of the dynamic weighing.
SM4_A	Command understood and executed successfully; time interval set.
SM4_I	Command understood but currently not executable (balance is currently executing another command).
SM4_L	Command understood but not executable (e.g. dynamic weighing application is not active or parameter is incorrect).

Parameter

Name	Type	Values	Meaning
<DynWeighTimeInterval>	Integer	0..99	Time interval in seconds

Comments

- This command can only be used with the application "dynamic weighing". For details on available applications and how to activate them see also [M25](#) and [M26](#).
- The time interval preset by the SM4 command is automatically set to the actual value indicated in the balance's display when the time interval parameter is selected manually.
- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS, manually in the application settings, or by means of a balance master reset (not @ or [SM0](#)).
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SM4_10	Set the time interval for the data acquisition of the dynamic weighing to 10 seconds
↑	SM4_A	Current time interval is set to 10 seconds

SNR – Send Stable Weight Value and Repeat on Weight Change

Level 2

Description

Use SNR to send the current stable weight value following a predefined minimum change in weight and on a continuous basis. The weight value is sent, along with the [host unit](#), from the balance to the connected communication partner via the interface.

Syntax

Command

SNR [<PresetValue>_ <Unit>]	Send the current stable weight value and repeat after each deflection greater or equal to the preset value (see comment).
--	---

Balance Responses

S_ S_ <WeightValue>_ <Unit>	Current stable weight value (1 st value).
S_ S_ <WeightValue>_ <Unit>	Next stable weight value after preset deflection (2 nd value).
...	...
S_ I	Command understood but currently not executable (balance is currently executing another command, e.g. taring, or timeout as stability was not reached).
S_ L	Command understood but not executable (incorrect parameter).
S_ +	Balance in overload range.
S_ -	Balance in underload range.

Parameters

Name	Type	Values	Meaning
<PresetValue>	Float	1 d .. capacity	Preset minimum deflection load
<Unit>	String		Weight unit

Comment

- The preset value is optional. If no value is defined, the deflection depends on balance readability as follows:

Readability	Min. Deflection
0.001 mg	0.001 g
0.01 mg	0.01 g
0.1 mg	0.1 g
0.001 g	1 g
0.01 g	1 g
0.1 g	1 g
1 g	5 g

- SNR is overwritten by the commands [S](#), [SI](#), [SIR](#), [@](#) and hardware break and hence cancelled.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SNR_50_g	Send the current stable weight value and repeat after each deflection greater or equal to the preset value of 50 g
↑	S_ S_ 12.34_g	1 st weight value is 12.34 g
↑	S_ S_ 67.89_g	2 nd weight value is 67.89 g

SNRU – Send Stable Weight Value with Currently Displayed Unit and Repeat on Weight Change

Level 2

Description

Use **SNRU** to send the current stable weight value following a predefined minimum change in weight and on a continuous basis. The weight value is sent, along with the displayed unit, from the balance to the connected communication partner via the interface.

Syntax

Command

SNRU [_␣ <PresetValue>_<Unit>]	Send the current stable weight value with the currently displayed unit and repeat after each deflection greater or equal to the preset value (see comment).
---	---

Balance Responses

S_S_<WeightValue>_<Unit> S_S_<WeightValue>_<Unit> ...	Current stable weight value (1 st value). Next stable weight value after preset deflection (2 nd value). ...
S_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring, or timeout as stability was not reached).
S_L	Command understood but not executable (incorrect parameter).
S_+	Balance in overload range.
S_-	Balance in underload range.

Parameter

Name	Type	Values	Meaning
<PresetValue>	Float	1 d .. capacity	Preset minimum deflection load
<Unit>	String		Weight unit

Comment

- As the **SNR** command, but with current displayed unit.
- The preset value is optional. If no value is defined, the deflection depends on balance readability as follows:

Readability	Min. Deflection
0.001 mg	0.001 g
0.01 mg	0.01 g
0.1 mg	0.1 g
0.001 g	1 g
0.01 g	1 g
0.1 g	1 g
1 g	5 g

- SNRU** is overwritten by the commands **S**, **SI**, **SIR**, **@** and hardware break and hence cancelled.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SNRU_50_g	Send the current stable weight value with the currently displayed unit and repeat after each deflection greater or equal to the preset value of 50 g
↑	S_S_12.34_g	1 st weight value is 12.34 g
↑	S_S_67.89_g	2 nd weight value is 67.89 g

SR – Send Weight Value on Weight Change (Send and Repeat)

Level 1

Description

Use **SR** to send the current weight values following a predefined minimum change in weight and on a continuous basis. The weight value is sent, along with the **host unit**, from the balance to the connected communication partner via the interface.

Syntax

Command

<code>SR [_<PresetValue>_<Unit>]</code>	Send the current stable weight value and then continuously after every weight change greater or equal to the preset value a non-stable (dynamic) value followed by the next stable value, range = 1 d to maximal capacity. If no preset value is entered, the weight change must be at least 12.5% of the last stable weight value, minimum = 30 d .
---	---

Balance Responses

<code>S_S_<WeightValue>_<Unit></code>	Current, stable weight value in unit actually set as host unit , 1 st weight change.
<code>S_D_<WeightValue>_<Unit></code>	Dynamic weight value in unit actually set as host unit .
<code>S_S_<WeightValue>_<Unit></code>	Next stable weight value in unit actually set as host unit .
<code>S_I</code>	Command understood but currently not executable (balance is currently executing another command, e.g. zero setting, or timeout as stability was not reached).
<code>S_L</code>	Command understood but not executable (incorrect parameter).
<code>S_+</code>	Balance in overload range.
<code>S_-</code>	Balance in underload range.

Comments

- **SR** is overwritten by the commands **S**, **SI**, **SIR**, **@** and hardware break and hence cancelled.
- In contrast to **SR**, **SNR** only sends stable weight values.
- If, following a non-stable (dynamic) weight value, stability has not been reached within the timeout interval, the response **S_I** is sent and then a non-stable weight value. Timeout then starts again from the beginning.
- The preset value can be entered in any by the balance accepted unit.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	<code>SR_10.00_g</code>	Send the current stable weight value followed by every load change of 10 g
↑	<code>S_S_100.00_g</code>	Balance stable
↑	<code>S_D_115.23_g</code>	100.00 g loaded
↑	<code>S_S_200.00_g</code>	Balance again stable

See also

[S – Send Stable Weight Value](#)

[SI – Send Weight Value Immediately](#)

[SIR – Send Weight Value Immediately and Repeat](#)

[SNR – Send Stable Weight Value and Repeat on Weight Change](#)

SRU – Send Stable Weight Value with Currently Displayed Unit on Weight Change

Level 2

Description

Use **SRU** to send the current weight values following a predefined minimum change in weight and on a continuous basis. The weight value is sent, along with the currently displayed unit, from the balance to the connected communication partner via the interface.

Syntax

Command

<code>SRU [_<WeightValue>_<Unit>]</code>	Send the current stable weight value with the currently displayed unit and then continuously after every weight change greater or equal to the preset value a non-stable (dynamic) value followed by the next stable value, range = 1 d to maximal capacity. If no preset value is entered, the weight change must be at least 12.5% of the last stable weight value, minimum = 30 d .
--	---

Balance Responses

<code>S_S_<WeightValue>_<Unit></code>	Current, stable weight value with the currently displayed unit until 1 st weight change.
<code>S_D_<WeightValue>_<Unit></code>	Non-stable (dynamic) weight value with the currently displayed unit.
<code>S_I</code>	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
<code>S_+</code>	Balance in overload range.
<code>S_-</code>	Balance in underload range.

Comment

- As the **SR** command, but with currently displayed unit.
- **SRU** is overwritten by the commands **S**, **SI**, **SIR**, **@** and hardware break and hence cancelled.
- In contrast to **SR**, **SNRU** only sends stable weight values.
- If, following a non-stable (dynamic) weight value, stability has not been reached within the timeout interval, the response **S_I** is sent and then a non-stable weight value. Timeout then starts again from the beginning.
- The preset value can be entered in any by the balance accepted unit.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).


Example

↓	<code>SRU</code>	Send the current stable weight value followed by every default load change with current display unit
↑	<code>S_S_12.34_lb</code>	1 st weight value is stable and 12.34 lb
↑	<code>S_D_13.88_lb</code>	2 nd weight value is non-stable and 13.88 lb
↑	<code>S_S_15.01_lb</code>	3 rd weight value is stable and 15.01 lb

ST – Send Stable Weight Value on Pressing (Transfer) Key


Level 2

Description



Use `ST` to send the current stable weight value when the transfer key () is pressed. The value is sent, along with the currently displayed unit, from the balance to the connected communication partner via the interface.

Syntax

Commands

<code>ST</code>	Query of the current status of the <code>ST</code> function.
<code>ST_1</code>	Sent the current stable net weight value with Display unit each time when the transfer key () is pressed.
<code>ST_0</code>	Stop sending weight value when Print key is pressed.


Balance Responses

<code>ST_A_0</code>	Function inactive, no weight value is sent when the transfer key () is pressed.
<code>ST_A_1</code>	Function active, weight value is sent each time when the transfer key () is pressed.
<code>ST_A</code>	Command understood and executed successfully.
<code>ST_I</code>	Command understood but currently not executable (balance is currently executing another command).
<code>ST_L</code>	Command understood but not executable (incorrect parameter; certified version of the balance).

Comments

- `ST_0` is the factory setting (default value).
- `ST` function is not active after switching on and after reset command.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Examples

↓	<code>ST_1</code>	Activate <code>ST</code> function
↑	<code>ST_A</code>	Command executed.
↑	<code>S_S_123.456_g</code>	When transfer key () pressed: current net weight is 123.456 g

SU – Send Stable Weight Value with Currently Displayed Unit

Level 2

Description

Use **SU** to send a stable weight value, along with the currently displayed unit, from the balance to the connected communication partner via the interface.

Syntax

Command

SU	Send the current stable net weight value with the currently displayed unit.
----	---

Balance Responses

S_S_<WeightValue>_<Unit>	Current stable weight value with the currently displayed unit.
S_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring, or timeout as stability was not reached).
S_+	Balance in overload range.
S_-	Balance in underload range.

Comments

- As the **s** command, but with currently displayed unit.
- The [draft shield](#) closes with this command, when the "Door function" is set on "Automatic". It opens after sending a stable weight.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	SU	Send the stable weight value with the currently displayed unit
↑	S_S_00000012.34_lb	The current, stable weight value is 12.34 lb

SUM – Send Stable Weight Value With Currently Displayed Unit and MinWeigh Information Level 2

Description

Use `SUM` to send the current stable weight value, along with the currently displayed unit and the MinWeigh information, from the balance to the connected communication partner via the interface.

Syntax

Command

<code>SUM</code>	Send the current stable net weight value with currently displayed unit and MinWeigh Information.
------------------	--

Balance Responses

<code>S_<Status>_<WeightValue>_<Unit></code>	Weight value in currently displayed unit.
<code>S_I</code>	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
<code>S_+</code>	Balance in overload range.
<code>S_-</code>	Balance in underload range.

Parameters

Name	Type	Values	Meaning
<code><Status></code>	String	S M	Stable, >= MinWeigh limit Stable, < MinWeigh limit
<code><WeightValue></code>	Float		Weight value
<code><Unit></code>	String		Display-Unit

Comment

- As the `S` command, but with currently displayed unit and MinWeigh information.
- If a weight other than the net weight is displayed, only the "S" index and the stable weight value displayed are output on the interface.
- If the MinWeigh function is switched off or not available on the balance, the corresponding command is `SU`.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	<code>SUM</code>	Query of the current weight value with currently displayed unit
↑	<code>S_M_123.34_mg</code>	Stable weight displayed, less than MinWeigh limit
↓	<code>SUM</code>	Query of the current weight value with currently displayed unit
↑	<code>S_S_123.34_mg</code>	Stable weight displayed, greater than MinWeigh limit

T – Tare

Level 1

Description

Use **T** to tare the balance. The next stable weight value will be saved in the tare memory.

Syntax

Command

T	Tare, i.e. store the next stable weight value as a new tare weight value.
---	---

Balance Responses

T_S_<TareWeightValue>_<Unit>	Taring successfully performed. The tare weight value returned corresponds to the weight change on the balance in the unit actually set under host unit since the last zero setting.
T_I	Command understood but currently not executable (balance is currently executing another command, e.g. zero setting, or timeout as stability was not reached).
T_+	Upper limit of taring range exceeded.
T_-	Lower limit of taring range exceeded.

Comments

- The tare memory is overwritten by the new tare weight value.
- The duration of the timeout depends on the balance type.
- Clearing tare memory: See command [TAC](#).
- The draft shield closes with this command, when the "Door function" is set on "Automatic". It opens after sending a stable weight.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	T	Tare
↑	T_S_100.00_g	The balance is tared and has a value of 100.00 g in the tare memory

TA – Query/Preset Tare Weight Value

Level 1

Description

Use TA to query the current tare value or preset a known tare value.

Syntax

Command

TA	Query of the current tare weight value.
TA_<TarePresetValue>_<Unit>	Preset of a tare value.

Balance Responses

TA_A_<TareWeightValue>_<Unit>	Query current tare weight value in tare memory, in unit actually set under host unit .
TA_I	Command understood but currently not executable (balance is currently executing another command, e.g. zero setting, or timeout as stability was not reached).
TA_L	Command understood but not executable (incorrect parameter).

Comments

- The tare memory will be overwritten by the preset tare weight value.
- The inputted tare value will be automatically rounded by the balance to the current readability.
- The taring range is specified to the balance type.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	TA_100.00_g	Preset a tare weight of 100 g
↑	TA_A_00000100.00_g	The balance has a value of 100.00 g in the tare memory

See also

[TAC – Clear Tare Value](#)

TAC – Clear Tare Value

Level 1

Description

Use TAC to clear the tare memory.

Syntax

Command

TAC	Clear tare value.
-----	-------------------

Balance Responses

TAC_A	Tare value cleared, 0 is in the tare memory.
TAC_I	Command understood but currently not executable (balance is currently executing another command, e.g. zero setting).

TI – Tare Immediately

Level 1

Description

Use **TI** to tare the balance immediately and independently of balance stability.

Syntax

Command

TI	Tare immediately, i.e. store the current weight value, which can be stable or non stable (dynamic), as tare weight value.
----	---

Balance Responses

TI_S_<TareWeightValue>_<Unit>	Taring performed, stable tare value. The new tare value corresponds to the weight change on the balance since the last zero setting.
TI_D_<TareWeightValue>_<Unit>	Taring performed, non-stable (dynamic) tare value.
TI_I	Command understood but currently not executable (balance is currently executing another command, e.g. zero setting).
TI_L	Command understood but not executable (e.g. certified version of the balance).
TI_+	Upper limit of taring range exceeded.
TI_-	Lower limit of taring range exceeded.

Comments

- The tare memory will be overwritten by the new tare weight value.
- After a non-stable (dynamic) stored tare weight value, a stable weight value can be determined. However, the absolute value of the stable weight value determined in this manner is not accurate.
- The stored tare weight value is sent in the unit actually set under [host unit](#).
- This command is not supported by approved balances.
- The taring range is specified to the balance type.
- The weight value is formatted as a right aligned string with 10 characters including the decimal point. For details, please refer to [Format of Balance Responses with Weight Value](#).

Example

↓	TI	Tare immediately
↑	TI_D_117.57_g	The tare memory holds a non-stable (dynamic) weight value

See also

[TAC – Clear Tare Value](#)

TIM – Query/Set Time

Level 2

Description

Set the system time of the balance or query the current setting.

Syntax

Commands

TIM	Query of the current time of the balance.
TIM_<hh>_<mm>_<ss>	Set the time of the balance.

Balance Responses

TIM_A_<hh>_<mm>_<ss>	Current time of the balance.
TIM_A	Command understood and executed successfully.
TIM_I	Command understood but currently not executable (balance is currently executing another command).
TIM_L	Command understood but not executable (incorrect parameter, e.g. 22_67_25) or no clock is built in.

Parameters

Name	Type	Values	Meaning
<hh>	Integer	00..23	Hours
<mm>	Integer	00..59	Minutes
<ss>	Integer	00..59	Seconds

Comment

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS or the balance system settings (not @ or a master reset).

Example

↓	TIM	Query of the current time of the balance
↑	TIM_A_09_56_11	The current time of the balance is 9 hours, 56 minutes and 11 seconds

TST0 – Query/Set Test Function Settings

Level 2

Description

Use TST0 to query the current setting for testing the balance, or to specify the type of testing (internal or external).

Syntax

Commands

TST0	Query of the setting for the test function.
TST0_<Test>	Set the test configuration of the balance.

Balance Responses

TST0_A_<Test>_<TestWeightValue>_<Unit>"	Current setting for the test function.
TST0_A	Command understood and executed successfully.
TIM_I	Command understood but currently not executable (balance is currently executing another command).
TIM_L	Command understood but not executable (incorrect parameter).

Parameters

Name	Type	Values	Meaning
<Test>	Integer	0	Test with internal weight
		1	Test with external weight
<WeightValue>	Float		Weight of the external weight currently set that is requested in the test from the balance user via the display
<Unit>	String		Unit of the external weight currently set

Comments

- The current value of the external weight can be seen in the menu under "Test" (see operating instructions).
- The value of the external weight is set in the menu under "Test" or with **M20**.
- With an internal test, no weight value appears.
- Use @ to abort a running test.
- For additional information on testing the adjustment, see the Manual for the balance.

Example

↓	TST0	Query of the current setting for the test and the value of the external test weight
↑	TST0_A_1_2000.0_g	The current setting corresponds to the test with an external weight. For a test initiated with the TST2 command (see below), an external weight of 2000.0 g is needed.

TST1 – Start Test Function According to Current Settings

Level 2

Description

Use TST1 to start the balance test function using the preset parameter settings.

Syntax

Commands

TST1	Start test function in the current setting (TST0, M20).
------	---

First Balance Responses

TST1_B	The test procedure has been started. Wait for next response (see comment).
TST1_I	Command understood but currently not executable (balance is currently executing another command). No second response follows.
TST1_L	Command understood but not executable (incorrect parameter). No second response follows.

Further Balance Responses

TST1_<WeightValue>_<Unit>	Prompt to unload and load the balance (only with external weight).
TST1_A_<WeightValue>_<Unit>	Test procedure completed successfully. Weight value with unit corresponds to the deviation from the specified value displayed after the test. No unit is specified if the test has been performed with the internal weight.
TST1_I	The test procedure has been aborted as, e.g. stability was not attained or wrong weights were loaded.

Comment

- Commands sent to the balance during the test procedure are not processed and responded to in the appropriate manner until the test procedure is at an end.
- Use @ to abort a running test.
- For additional information on testing the adjustment, see the Manual for the balance.

Example

↓	TST1	Start test function in the current setting
↑	TST1_B	The test procedure has been started
↑	TST1_ "0.00000_g"	Clear weighing pan
↑	TST1_ "100.00000_g"	Load 100 g external weight
↑	TST1_ "0.00000_g"	Unload weight
↑	TST1_A_ "0.00020_g"	Test completed, current difference is 0.00020 g

TST2 – Start Test Function with External Weight

Level 2

Description

Use TST2 to start the balance test function using external test weights.

Syntax

Commands

TST2	Start test function with external weight.
------	---

First Balance Responses

TST2_B	The test procedure has been started. Wait for next response (see comment).
TST2_I	Command understood but currently not executable (balance is currently executing another command). No second response follows.
TST2_L	Command understood but not executable (incorrect parameter). No second response follows.

Further Balance Responses

TST2_ "<WeightValue>_<Unit>"	Prompt to unload and load the balance.
TST2_A_ "<WeightValue>_<Unit>"	Test procedure completed successfully. Weight value with unit corresponds to the deviation from the specified value displayed in the top line after the test.
TST2_I	The test procedure has been aborted as, e.g. stability was not attained or wrong weights were loaded.

Comment

- Commands sent to the balance during the test procedure are not processed and responded to in the appropriate manner until the test procedure is at an end.
- The value of the external weight is set in the menu under "Test" or with **M20**.
- Use @ to abort a running test.
- For additional information on testing the adjustment, see the Manual for the balance.

Example

↓	TST2	Start test with external weight
↑	TST2_B	The test procedure has been started
↑	TST2_ "____0.00_g"	Prompt to unload the balance
↑	TST2_ "_200.00_g"	Prompt to load the test weight
↑	TST2_ "____0.00_g"	Prompt to unload the balance
↑	TST2_A_ "____0.01_g"	External test completed successfully

TST3 – Start Test Function with Internal Weight

Level 2

Description

Use TST3 to start the balance test function using internal test weights.

Syntax

Commands

TST3	Start test function with built-in weight.
------	---

First Balance Responses

TST3_B	The test procedure has been started. Wait for next response (see comment).
TST3_I	Command understood but currently not executable (balance is currently executing another command). No second response follows.
TST3_L	Command understood but not executable (incorrect parameter). No second response follows.

Further Balance Responses

TST3_A_ "<WeightValue>"	Test procedure completed successfully. Weight value corresponds to the deviation from the specified value displayed after the test.
TST3_I	The test procedure has been aborted as, e.g. stability was not attained or wrong weights were loaded.

Comment

- The commands received immediately after the first response are not processed and responded to in the appropriate manner until after the second response.
- Use @ to abort a running test.
- For additional information on testing the adjustment, see the Manual for the balance.

Example

↓	TST3	Start test with internal weight
↑	TST3_B	The test procedure has been started
↑	TST3_A_ "_____0.01"	Test with internal weight completed successfully. The difference to the specified value is 0.01.

UPD – Query/Set Update Rate of the Host Interface

Level 2

Description

Use UPD to set the update rate of the host interface or query the current setting.

Syntax

Commands

UPD	Query of the update rate of the host interface.
UPD_<UpdateRate>	Set the update rate of the host interface.

Balance Responses

UPD_A_<UpdateRate>	Current setting of the update rate of the host interface.
UPD_A	Command understood and executed successfully.
UPD_I	Command understood but currently not executable (balance is currently executing another command).
UPD_L	Command understood but not executable (incorrect parameter).

Parameter

Name	Type	Values	Meaning
<UpdateRate>	Float	1..100	Update rate in values per second Terminal: 1..23, stand-alone bridge: 1..100

Comment

- The parameter setting will be saved and the only way to reset the default value will be via MT-SICS (not @ or a master reset).
- Use UPD to configure the update rate of **SIR** and **SIRU**.
- The balance can not realize every arbitrary update rate. The specified update rate is therefore rounded to the next realizable update rate. Use UPD without parameter to query the actually configured update rate.
- If an update rate greater than 23 is specified for a stand-alone bridge, an update rate less than 23 must be specified before re-attaching a terminal. Otherwise, unpredictable behavior may occur.

Example

↓	UPD	Query of the update rate of the host interface
↑	UPD_A_20.2	The update rate of the interface is 20.2 values per second

WS – Query/Set Position of Draft Shield Doors

Level 2

Description

You can use `ws` to query the position of the draft shield doors. However, you can also use it to open or close the doors.

Syntax

Commands

<code>WS</code>	Query of the position of the draft shield door(s).
<code>WS_<DoorPosition></code>	Set the position of the draft shield door(s).

Balance Responses

<code>WS_A</code>	Command understood and executed successfully.
<code>WS_I</code>	Command understood but currently not executable (balance is currently executing another command).
<code>WS_L</code>	Command understood but not executable (incorrect parameter).

Comments

- The user must ensure that the doors are in the correct position. If the doors are blocked when closing, then they return to their original position. The position can be monitored by a query command.
- If the balance has an additional inner draft shield, when the position is queried the "interim" position is output until both of the draft shield units are in the required end position. A draft shield drive error will be indicated by status value = 8 (Error). It is not possible to control either of the draft shield units individually.

The draft shield doors in most models are driven by only a single drive mechanism, the left and the right door can be anchored mechanically. With these models, all draft shield doors that are anchored can be opened with the command "Open left door" or "Open right door".

Parameter

Name	Type	Values	Meaning
<code><DoorPosition></code>	Integer	0	Door(s) closed
		1	Door(s) open (right)
		2	Door(s) open (left)
		8	Error
		9	Intermediate

Example

↓	<code>WS_1</code>	Open the draft shield door(s)
↑	<code>WS_A</code>	Draft shield door(s) opened

See Also

[M37 – Query/Set Draft Shield Door Opening Width](#)

[M40 – Query/Set Inner Draft Shield Door Opening Width](#)

Z – Zero

Level 0

Description

Use **z** to set a new zero; all weight values (including the tare weight) will be measured relative to this zero. After zeroing has taken place, the following values apply: tare weight = 0; net weight (= gross weight) = 0.

Syntax

Command

z	Zero the balance.
----------	-------------------

Balance Responses

Z_A	Zero setting successfully performed. Gross, net and tare = 0.
Z_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring, or timeout as stability was not reached).
Z_+	Upper limit of zero setting range exceeded.
Z_-	Lower limit of zero setting range exceeded.

Comments

- The tare memory is cleared after zero setting.
- The zero point determined during switching on is not influenced by this command, the measurement ranges remain unchanged.
- The duration of the timeout depends on the balance type.
- The draft shield closes with this command, when the "Door function" is set on "Automatic". It opens after sending a stable weight.

Example

↓	z	Zero
↑	Z_A	Zero setting performed

ZI – Zero Immediately

Level 0

Description

Use **ZI** to set a new zero immediately, regardless of balance stability. All weight values (including the tare weight) will be measured relative to this zero. After zeroing has taken place, the following values apply: tare weight = 0; net weight (= gross weight) = 0.

Syntax

Command

ZI	Zero the balance immediately regardless the stability of balance.
----	---

Balance Responses

ZI_D	Re-zero performed under non-stable (dynamic) conditions.
ZI_S	Re-zero performed under stable conditions.
ZI_I	Command understood but currently not executable (balance is currently executing another command, e.g. taring).
ZI_-	Upper limit of zero setting range exceeded.
ZI_-	Lower limit of zero setting range exceeded.

Comments


- The tare memory is cleared after zero setting.
- This command is not supported by approved balances.
- The zero point determined during switching on is not influenced by this command, the measurement ranges remain unchanged.

Example

↓	ZI	Zero immediately
↑	ZI_D	Re-zero performed under non-stable (dynamic) conditions

4 Commands Listed by Tasks

4.1 Weighing, Zeroing, Taring

Weighing	S	Send Stable Weight Value
	SI	Send Weight Value Immediately
	SIR	Send Weight Value Immediately and Repeat
	SR	Send Weight Value on Weight Change (Send and Repeat)
	SIS	Send Weight Value with Actual Unit and Weighing Status
	SIU	Send Weight Value with Currently Displayed Unit Immediately
	SIUM	Send Weight Value With Currently Displayed Unit and MinWeigh Information Immediately
	SIRU	Send Weight Value with Currently Displayed Unit Immediately and Repeat
	SNR	Send Stable Weight Value and Repeat on Weight Change
	SNRU	Send Stable Weight Value with Currently Displayed Unit and Repeat on Weight Change
	SRU	Send Stable Weight Value with Currently Displayed Unit on Weight Change
	ST	Send Stable Weight Value on Pressing  (Transfer) Key
	SU	Send Stable Weight Value with Currently Displayed Unit
	SUM	Send Stable Weight Value With Currently Displayed Unit and MinWeigh Information See also commands specially used with MinWeigh
Zeroing	Z	Zero
	ZI	Zero Immediately
	M35	Query/Set Zeroing Mode at Start-Up
Taring	T	Tare
	TA	Query/Preset Tare Weight Value
	TAC	Clear Tare Value
	TI	Tare Immediately
	A08	Dynamic Weighing: Query/Set AutoTare
4.1.1 Setup		
Status	I0	List All Implemented MT-SICS Commands
	B00	FastHost: List Commands
	M25	List Applications
	DAT	Query/Set Date
	TIM	Query/Set Time
	PWR	Power On/Off
Cancel, Reset	@	Cancel
	SM0	Dynamic Weighing: Cancel all SMx Commands
	DW	Display: Show Weight
Weighing filter setup	M01	Query/Set Weighing Mode
	M02	Query/Set Environment
	M29	Query/Set Value Release
	M03	Query/Set AutoZero
	I20	Query MinWeigh Parameter

Terminal, Display	D	Display: Write Text to Display
	DW	Display: Show Weight
	K	Keys: Query/Set Configuration
	M04	Query/Set SmartSens Functions
	M08	Display: Query/Set Brightness
	M09	Display: Query/Set Contrast
	M11	Beeper: Query/Set Volume
	M12	Beeper: Beep
	M13	Query/Set Touch Function
	M14	List Available Languages
	M15	Query/Set Language
	M16	Query/Set Standby Mode
	M23	Query/Set Readability, 1d/xd
	M24	Query/Set Print Key Function
PWR	Power On/Off	
Door	M37	Query/Set Draft Shield Door Opening Width
	M40	Query/Set Inner Draft Shield Door Opening Width
	WS	Query/Set Position of Draft Shield Doors
	M07	Query/Set AutoDoor
4.2 General Setup		
4.2.1 Printer		
Printer	M24	Query/Set Print Key Function
	P100	Printer Interface: Send Text
	P101	Printer Interface: Send Weight Value
	P102	Printer Interface: Send Weight Value Immediately
4.2.2 Applications		
MinWeigh	I17	MinWeigh: Query Date of Next Test
	I18	MinWeigh: Query Method
	I19	MinWeigh: Query Limits
	I20	Query MinWeigh Parameter
	SUM	Send Stable Weight Value With Currently Displayed Unit and MinWeigh Information
	SIUM	Send Weight Value with Currently Displayed Unit and MinWeigh Information Immediately
	M34	MinWeigh: Query/Set Method
Percent Weighing	A01	Percent Weighing: Query/Set Reference in %
	M25	List Applications
	M26	Query/Set Current Application
	SU	Send Stable Weight Value with Currently Displayed Unit
Piece Counting	PW	Piece Counting: Query/Set Piece Weight
	M25	List Applications
	M26	Query/Set Current Application
	SU	Send Stable Weight Value with Currently Displayed Unit
Dynamic Weighing	A06	Dynamic Weighing: Query/Set Filter
	A07	Dynamic Weighing: Query/Set AutoStart

A08	Dynamic Weighing: Query/Set AutoTare
A09	Dynamic Weighing: Query/Set Data Acquisition
SM0	Dynamic Weighing: Cancel all SMx Commands
SM1	Dynamic Weighing: Start Immediately and Send the Result
SM2	Dynamic Weighing: Start After a Minimum Load Is Exceeded and Send Result
SM3	Dynamic Weighing: Start After a Minimum Load Is Exceeded, Send Result and Repeat
SM4	Dynamic Weighing: Query/Set Time Interval
M25	List Applications
M26	Query/Set Current Application

Weighing to a nominal value	A10	Weighing: Query/Set Nominal, +Tolerance, -Tolerance
------------------------------------	------------	---

4.2.3 Options

Data Interface	COM	Query/Set Serial Interface Communication Parameters
	UPD	Query/Set Update Rate of the Host Interface

4.3 Adjustment/Test

Adjustment	C0	Query/Set Adjustment Settings
	C1	Start Adjustment According to Current Settings
	C2	Start Adjustment with External Weight
	C3	Start Adjustment with Internal Weight
	M19	Query/Set Adjustment Weight
	M17	ProFACT: Query/Set single Time Criteria
	M18	ProFACT/FACT: Query/Set Temperature Criterion (Δ temp.)
	M32	ProFACT: Query/Set Time Criteria
	M33	ProFACT: Query/Set Weekday
	M27	Query Adjustment History

Test Function	TST0	Query/Set Test Function Settings
	TST1	Start Test Function According to Current Settings
	TST2	Start Test Function with External Weight
	TST3	Start Test Function with Internal Weight
	M20	Query/Set Test Weight

4.4 Balance Information

Balance Information	I0	List All Implemented MT-SICS Commands
	B00	FastHost: List Commands
	I1	Query MT-SICS Level and MT-SICS Versions
	I2	Query Balance Data (Type and Capacity)
	I3	Query Balance Software Version and Type Definition Number
	I4	Query Serial Number
	I5	Query SW-Identification Number
	I10	Query Balance ID
	I11	Query Balance Type
	I14	Query Balance Information
	M36	Query/Set LevelControl

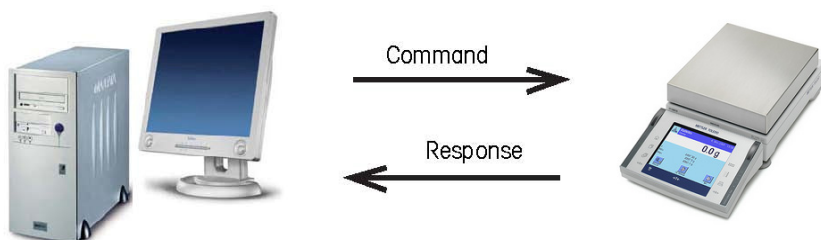
4.5 Fast Data Communication

Fast Host Commands	B00	FastHost: List Commands
	B01	FastHost: Send Individual Value
	B02	FastHost: Start/Stop Continuous Mode
	B03	FastHost: Query Time Interval of Value Counter
	B04	FastHost: Query/Set Format Specification
	B05	FastHost: Query/Set Stability Criteria
	B06	FastHost: Query/Set Downsampling Factor
	B07	FastHost: Query/Set Switch-On Mode
	B08	FastHost: Zero with FastHost Stability Criteria

5 Example: Formula weighing application

The following simple formula weighing application shows the data interchange between the computer with the formula weighing program and the balance.

A substance ($S = 121 \text{ g}$) comprising components $C1 = 100 \text{ g}$ and $C2 = 21 \text{ g}$ needs to be weighed into a beaker. If too much or too little of the first component is weighed in, the target weight of the second component should be adjusted so that the ratio of the two components remains the same. The user is guided by the balance display and acknowledges his actions with the tare key. In the end, the total weight of the substance is displayed.



↓	@	Cancel any running command
↑	I4_A_"1114350697"	Balance is reset. The serial number is 1114350697
↓	K_3	Disable key function and report each keystroke
↑	K_A	Command understood and executed successfully
↓	D_"BEAKER"	Write "BEAKER" into the balance display to prompt loading the beaker
↑	D_A	"BEAKER" appears in the balance display
↑	K_C_10	Acknowledges that the tare key has been pressed
↓	T	Tare weight on the balance
↑	T_S_70.0000_g	The beaker weighs 70.0000 g
↓	D_"C1_100g"	Write "C1 100g" into the balance display to prompt adding component C1 = 100 g
↑	D_A	"C1 100g" appears in the balance display
↑	K_C_10	Acknowledges that the tare key has been pressed
↓	S	Get actual weight of component 1
↑	S_S_105.0000_g	Target weight of component 1 missed by 5 g → 5% to much
↓	T	Tare weight on the balance
↑	T_S_175.0000_g	Contents of the tare memory, now corresponds to gross weight
↓	D_"C2_22.5g"	Write "C2 22.5g" into the balance display to prompt adding component C2 = 21 g + 5% = 22.5 g
↑	D_A	"C2 22.5g" appears in the balance display
↑	K_C_10	Acknowledges that the tare key has been pressed
↓	TA_70_g	Preset the tare weight of the beaker (70 g)
↑	TA_70.0000_g	The balance acknowledges the tare weight of 70 g
↓	DW	Switch balance display to show the weight again
↑	DW_A	127.5000 g appears in the balance display

6 What if...?

Tips from actual practice when the communication between the system (computer) and the balance does not function.

Establishing the communication

Switch the balance off with the corresponding "Off" key and then on again with the "On" key.

The balance must now send identification string **I4**, e.g. **I4_A_ "0123456789"**.

If this is not the case, check the following points.

Connection

For bidirectional communication, at least three connecting lines are needed:

- Data line from the balance (TxD signal with RS232 interface).
- Data line to the balance (RxD signal with RS232 interface).
- Signal ground line (SG with RS232 interface).

Make sure that all these connections are in order. Check the connector pin assignment of the connection cables.

Interface parameters

For the transmission to function properly, the settings of the following parameters must match at both the computer and the balance:

- Baud rate (send/receive rate)
- Number of data bits
- Parity bit

Check the settings at both devices.

Handshake

For control of the transmission, in part separate connection lines are used (CTS/DTR). If these lines are missing or wrongly connected, the computer or balance can not send or receive data.

Check whether the balance is prevented from transmitting by handshake lines (CTS or DTR). Set the parameter "protocol" for the balance and the peripheral device to "No Handshake" or "none". The handshake lines now have no influence on the communication.

Characters are not displayed correctly

In order to display ASCII characters >127 dec., ensure that 8-bit communication is taking place.

GWP® – Good Weighing Practice™

The global weighing guideline GWP® reduces risks associated with your weighing processes and helps to

- choose the appropriate balance
- reduce costs by optimizing testing procedures
- comply with the most common regulatory requirements

► www.mt.com/GWP

www.mt.com/excellence

For more information

Mettler-Toledo AG Laboratory & Weighing Technologies

CH-8606 Greifensee, Switzerland

Tel. +41 (0)44 944 22 11

Fax +41 (0)44 944 30 60

www.mt.com

Subject to technical changes

© Mettler-Toledo AG 04/2011

11780711D 2.12

