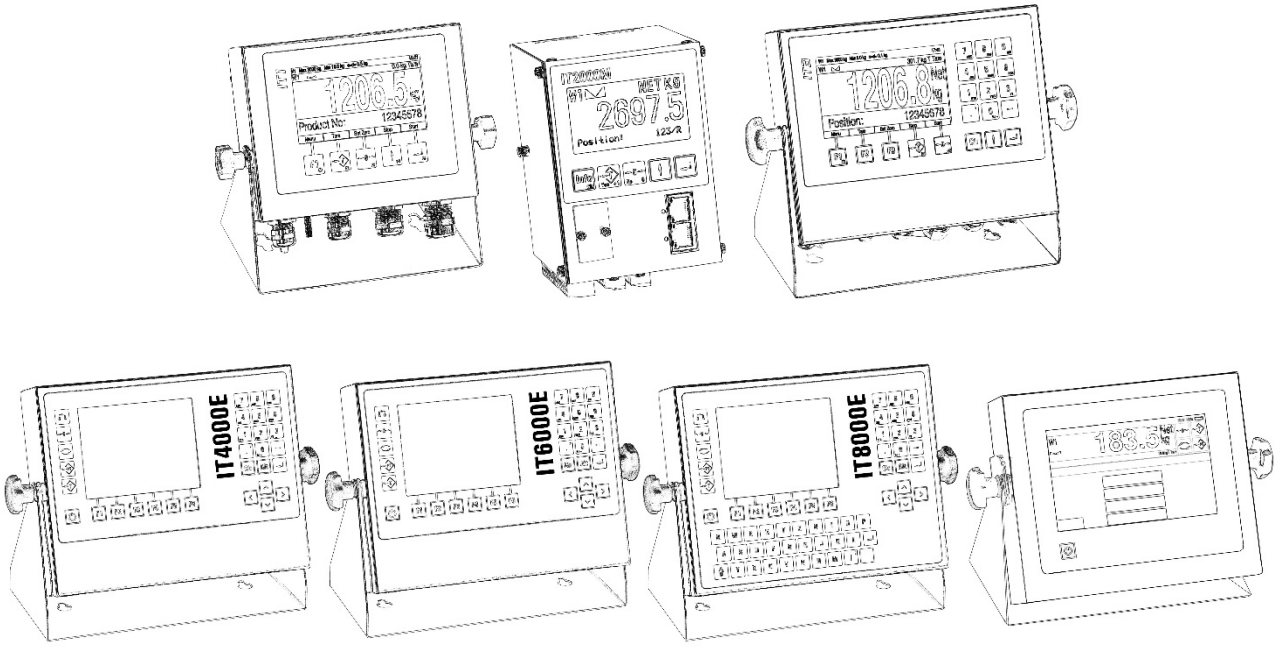
Technical Manual

Continuous Output



June 2024

ST.2309.2023

Rev. 5

Technical Manual 'Continuous Output'

Date: 6/6/2024

File: CONT_OUT_THE.PDF (Translation of the original documentation)

Revision history:	
Rev.	Modification in chapter(s)
5	4

Published By:

Please Note:

While every precaution has been taken in the preparation of this manual, SysTec GmbH assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

The publisher is grateful for any information and/or advice that may contribute to correct errors or omissions in following editions.

Contents

1 Documentation **7**
2 Continuous Output (Cont.out) **7**
3 SysTec Protocol **8**
4 SysTec Remote Protocol **8**
5 Customized Protocol **9**
6 CAS Protocol **12**
7 Flintec Protocol **12**
8 GS Protocol (Gebhardt & Schäfer With Traffic Light Function) **13**
9 MT-SICS Protocol **13**
10 Schauf Protocol **14**
11 SPEC1 Protocol **14**
12 SPEC2 Protocol **15**
13 TOLEDO® Protocol **16**

1 Documentation

This documentation is valid only in conjunction with the respective Technical Manual.

Technical Manual	Bestell-Nr.
IT1	ST.2309.1766
IT1 Ex2/22	ST.2309.1985
IT2000M	ST.2309.1640
IT2000M *BAG*	ST.2309.1750
IT2000M *CONTROL/ONLINE*	ST.2309.2006
IT2000M *FILL*	ST.2309.1809
IT2000M *PROCESS*	ST.2309.2058
IT3	ST.2309.1846
IT3 Ex2/22	ST.2309.1990
IT4000E	ST.2309.1193
IT4000E Ex2/22	ST.2309.1597
IT6000E	ST.2309.1204
IT6000E Ex2/22	ST.2309.1601
IT6000ET	ST.2309.1476
IT8000E	ST.2309.1188
IT8000E Ex2/22	ST.2309.1605
IT8000ET	ST.2309.1472
IT8000Ex2/22	ST.2309.1327

2 Continuous Output (Cont.out)

The interfaces Ethernet (COM0), COM1, COM... can be configured as continuous output, one of several protocols can be chosen in group 'General\Cont.out' of the Service Mode.

Setting of interface parameters is made in group 'Interface' of the Service Mode.

3 SysTec Protocol

The data string consists of 15 ASCII characters plus CR and LF. It includes a status for motion / no motion, the net weight and the unit sign. Characters not used are filled with space characters. The SysTec protocol is not permissible for W&M approved applications.
Example:

'12345678901234567'	Characters:	
	1st char.: Start character	always S
'S 10.98 t CRLF'	2nd char.: Status	␣ (space) = scale settled
'SD 10980 kgCRLF'		D = scale in motion
	3rd–12th char.	weight 10 characters, in the format of the scale calibration
	13th char.	always ␣ (space)
	14th–15th char.	weight unit
	16th–17th char.	CR and LF

4 SysTec Remote Protocol

This data string is used for W&M approved applications to connect an IT1/IT1000 remote display with extended functions using the serial interface.

The operating mode 'Remote Display' must be chosen in the configuration of the IT1000 remote display.

Actuating the tare and zero key at the remote display is transmitted back to the weighing terminal and has the same effect as pressing the corresponding key here.

'1234567890123456'	Characters:	
' 100.0 kgNCRLF'	1st char.	always ␣ (space)
	2nd char.: Status	~ = scale in motion
		° = scale settled and gross weight is in zero range °=Character code (0xAF), corresponds to thecharacter in the proprietary codepage of the IT9000
		␣ (space) = single range scale settled
		1,2,3 = weighing range (multiple-range scale settled)
	3rd–10th char.	weight 8 characters, in the format of the scale calibration
	11th char.	always ␣ (space)
	12th–13th char.	weight unit
	14th char.	N = net weight
		␣ (space) = gross weight
	15th–16th char.	CR and LF

5 Customized Protocol

The data string can be freely defined. In the table below **x** and **y** are wild cards. If the condition is true, the character specified under **x** is transmitted. If the condition is false, the character specified under **y** is transmitted instead.

If you want – for instance – to send the character **~** when the scale is in motion, the corresponding string is **M~:R** (condition = true). If the condition is false, character R is sent instead.

The entry of a colon followed by a character y is optional. In this case, a space character is transmitted if condition = false.

Weights are transmitted as shown on the display including decimal separator.

Non-significant (leading) weight digits are transmitted as space.

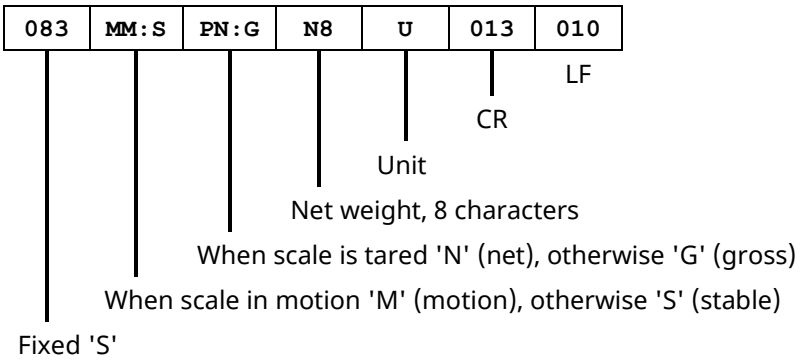
String	Transmission	Example
Wx	Sends the **scale number** with the number of digits specified under **x**, e.g. 6.	**W6**
Mx:y	Transmits the character specified under **x**, if the scale is **in motion**, e.g.: **B,** else transmits the character specified under **y**.	**MB:R**
mx:y	Transmits the character specified under **x**, if the scale is **settled**, e.g.: **R,** else transmits the character specified under **y.**	**mR:~**
Ox:y	Transmits the character specified under **x**, if the scale is **in overload**, e.g.: **U,** else transmits the character specified under **y**	**OU:N**
ox:y	Transmits the character specified under **x**, if the scale is **not in overload**, e.g.: **N,** else transmits the character specified under **y**	**oN:U**
Zx:y	Transmits the character specified under **x**, if the scale is **in zero range**, e.g.: **N,** else transmits the character specified under **y**	**ZN:A**
zx:y	Transmits the character specified under **x**, if the scale is **not in zero range**, e.g.: **A,** else transmits the character specified under **y**	**zA:N**
Px:y	Transmits the character specified under **x**, if the scale is **tared**, e.g.: **T,** else transmits the character specified under **y**	**PT:N**
px:y	Transmits the character specified under **x**, if the scale is **not tared**, e.g.: **N,** else transmits the character specified under **y**	**pN:T**
[space]	Transmits a space character	**[space]**
Lx:y	Transmits the character specified under **x**, if the scale is in **underload (gross weight under zero)**, e.g.: **U,** else transmits the character specified under **y**	**LU:[space]**
Ex:y	Transmits the character specified under **x**, if the scale detects a **scale error** e.g.: **E,** else transmits the character specified under **y**	**EE:[space]**
Gx	Transmits the **gross weight** with **x** digits, e.g.: **8.**	**G8**
G+x **GPx**	Transmits the **gross weight** with **x** digits, e.g.: **8,** with a preceding – character when the gross weight is **under zero**, else transmits a + character**.**	**G+8** **GP8**
G-x **GMx**	Transmits the **gross weight** with **x** digits, e.g.: **8,** with a preceding – character when the gross weight is **under zero**, else transmits a [space] character**.**	**G-8** **GM8**
Nx	Transmits the **net weight** with **x** digits, e.g.: **8.**	**N8**
N+x **NPx**	Transmits the **net weight** with **x** digits, e.g.: **8,** with a preceding – character when the net weight is **under zero**, else transmits a + character**.**	**N+8** **NP8**
N-x **NMx**	Transmits the **net weight** with **x** digits, e.g.: **8,** with a preceding – character when the net weight is **under zero**, else transmits a [space] character**.**	**N8** **NM8**
Tx	Transmits the **tare weight** with **x** digits, e.g.: **6.**	**T6**

String	Transmission	Example
Cx	Transmits the **number of pieces** with **x** digits, e.g.: **6**. When the next letter is **U**, either the **net weight** or **number of pieces** will be transmitted (if supported by the respective application, e.g. *BASIC/COUNT E*, IT3 operating mode *COUNT*; not for IT1 operating mode *COUNT*).	**C6**
U	Transmits the **unit of calibration**, e.g.: '**kg**,' '**t** ,' '**g** ,' '**lb**'	**U**
R	Transmits the **weighing range**, space for single range scales	**R**
123	Transmits a **specified character** (3-digit decimal code)	**002=STX**
S	Traffic light control for remote display (if supported by the respective application, e.g. *CONTROL/ONLINE E*, IT3 operating mode *TRUCK*).	**S**

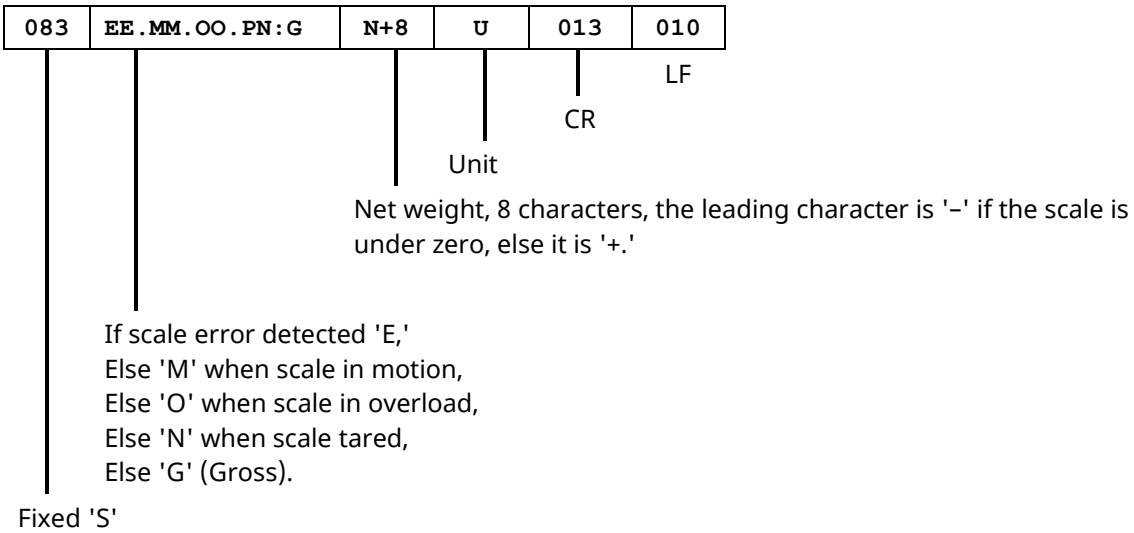
Alternatively, status information can be sent with one character only.

S1x1.S2x2.S3x3....	Transmits the character specified under **x1 if status S1 is active,** else transmits the character specified under **x2 if status S2 is active**, else the character specified under **x3 if status S3 is active**, etc. ..... As status information S1, S2, ... only the queries **M, m, O, o, Z, z, P, p, L, E** are permitted.	**MM.LU.PN:G**

Example for a data string to control a remote display with <u>different</u> status characters for motion / no-motion and gross / net indication.



Example for a data string with <u>1-character</u> status information to control a remote display with error, motion, overload and gross / net indication.



5.1 Extended Standard Protocol

The data string with the option to transmit status information consists of 18 ASCII characters plus CR and LF. It includes information on weighing range, gross/net, no-motion/motion, scale in zero as well as the weight and the unit sign. Characters not used are replaced with space characters.

Example string: '**088087RPN:GMM:SZZ N9 U013010**'

Char.: '123456789012345678901'

1	└────────────────────	Fixed 'X' (ASCII code 088) as start character
2	└────────────────────	Fixed character 'W' (ASCII code 087) as marker for 'weight data string' or 'D' for 'general data' (reserved for future extension)
3	└────────────────────	Weighing range'1,' '2' etc. or space for single-range scale
4	└────────────────────	'N' for net weight, 'G' for gross weight
5	└────────────────────	'M' for scale in motion or 'S' for scale settled
6	└────────────────────	'Z' when scale in zero range, otherwise space character
7	└────────────────────	'S' for traffic light function
8	└──────────┘└────	Weight, 9 characters, right justified, with preceding minus sign and decimal separator, if applicable
17	└────	Fixed space character between weight and unit
18	└┘└────	Weight unit 'kg,' 'g,' 't' or 'lb' (others if applicable), 2 characters, left justified
20	└──	<CR> (carriage return)
21	└─	<LF> (line feed)

Examples:

Char.: '12345678901234567890 1'

'XW NS 1371,5 kg$^{C_R L_F}$'	Net weight, scale settled,
'XW GSZ 0,0 kg$^{C_R L_F}$'	Gross weight, scale in zero and settled,
'XW2GS 21380 t $^{C_R L_F}$'	Gross weight, scale in second weighing range and settled,
'XW NM -1,35 kg$^{C_R L_F}$'	Net weight, scale in motion and weight negative.

6 CAS Protocol

The data string to transmit the weight consists of 20 ASCII characters plus CR and LF. Example:

'12345678901234567890'	Characters:		
'ST,NT,1X,- 10.95 kg$^{CR}_{LF}$'	1st–2nd char.: Status		ST = scale stable
			US = scale in motion
			OL = overload
	3rd char.		separator (comma)
	4th–5th char.		NT = net (scale tared)
			GS = gross
	6th char.		separator (comma)
	7th char.		always 1
	8th char.: Status bits		'76543210' (Bit) '11000110'
		Bit 7	always 1
		Bit 6	0 = scale in motion
			1 = scale stable (settled)
		Bit 5–3	always 0
		Bit 2	0 = scale tared
			1 = scale not tared
		Bit 1	0 = net weight
			1 = gross weight
		Bit 0	always 0
	9th char.		separator (comma)
	10th char.		— = weight negative
			␣ (space) = weight positive
	11th–17th char.		weight 7 characters, in format of scale calibration
	18th char.		separator = ␣ (space)
	19th–20th char.		weight unit
	21st–22nd char.		CR and LF

7 Flintec Protocol

The data string to connect a Flintec remote display consists of 1 start character (@), 7 ASCII characters for the net weight plus CR. Example:

'123456789'	Characters:	
'@ 10.98^{CR}'	1st char.	always @ (Hex 40)
	9th char.	always CR (Hex D)
		non-significant characters are filled with space (Hex 20).

8 GS Protocol (Gebhardt & Schäfer With Traffic Light Function)

The data string to control the remote display consists of 15 ASCII characters plus CR and LF. It contains a status character for motion / no-motion, the color of the traffic light, the net weight and the unit sign. Characters not used are filled with space characters. Example:

'12345678901234567'	Characters:	
	1st char.: Start character	always S
'S 0 10.98 t $^{C_R L_F}$**'**	2nd char.: Status	␣ (space) = scale settled
'SD 0 10980 kg$^{C_R L_F}$**'**		D = scale in motion
	3rd–5th char.	always ␣ (space)
	6th char.: Traffic light	0 = off
		1 = red
		2 = green
		3 = red + green
	7th–12th char.	weight, 6 characters, in format of scale calibration
	13th char.	always ␣ (space)
	14th–15th char.	weight unit
	16th–17th char.	CR and LF

9 MT-SICS Protocol

The data string to control a remote display consists of 17 ASCII characters plus CR and LF. It contains the motion / no-motion status, the net weight and the unit sign. Characters not used are filled with space characters. Example:

'12345678901234567'	Characters:	
	1st char.: Start character	always S
	2nd char.	always ␣ (space)
'S S 10.98 t $^{C_R L_F}$**'**	3rd char.: Status	S = scale settled
'S D 10980 kg$^{C_R L_F}$**'**		D = scale in motion
	4th char.	always ␣ (space)
	5th–14th char.	weight, 10 characters, in format of scale calibration
	15th char.	always ␣ (space)
	16th–17th char.	weight unit
	18th–19th char.	CR and LF

In case of error:

'S +$^{C_R L_F}$**'**	Scale in overload
'S -$^{C_R L_F}$**'**	Scale in underload
'S I$^{C_R L_F}$**'**	Other error

10 Schauf Protocol

The data string to control a Schauf remote display consists of 9 ASCII characters. Example:

`' 1 23456789'`	Characters:	
`'<ESC>!"100.0CR'`	1st char.	always [ESC] (Hex 1B)
	2nd char.	always ! character [33] (Hex 21)
	3rd char.	always ␣ (space) [32] (Hex 20)
	4th–8th char.	weight, 5 characters, in format of scale calibration
	9th char.	always CR character [13] (Hex 0D)

11 SPEC1 Protocol

The data string to control a remote display consists of 22 characters plus CR and LF. It contains net weight, tare weight, unit code and status information. Characters not used are filled with space characters. Example:

`'123456789012345678901234'`
`'sx 12345678 10000000 B0CRLF'`

Characters:		
1st char.		always <STX> Hex 02
2nd char.		always ␣ (space)
3rd–10th char.		net weight, 8 characters, no decimal separator
11th char.		always ␣ (space)
12th–19th char.		tare weight, 8 characters, no decimal separator
20th char.		always ␣ (space)

21st char.: Status	ASCII	Scale tared	Scale in zero range	Scale settled
0 = no	'1'	0	0	0
1 = yes	'3'	0	0	1
	'5'	0	1	0
	'7'	0	1	1
	'9'	1	0	0
	'B'	1	0	1
	'D'	1	1	0
	'F'	1	1	1

22nd char.: unit	0 = kg
	1 = g
	3 = t
23rd–24th char.	CR and LF

12 SPEC2 Protocol

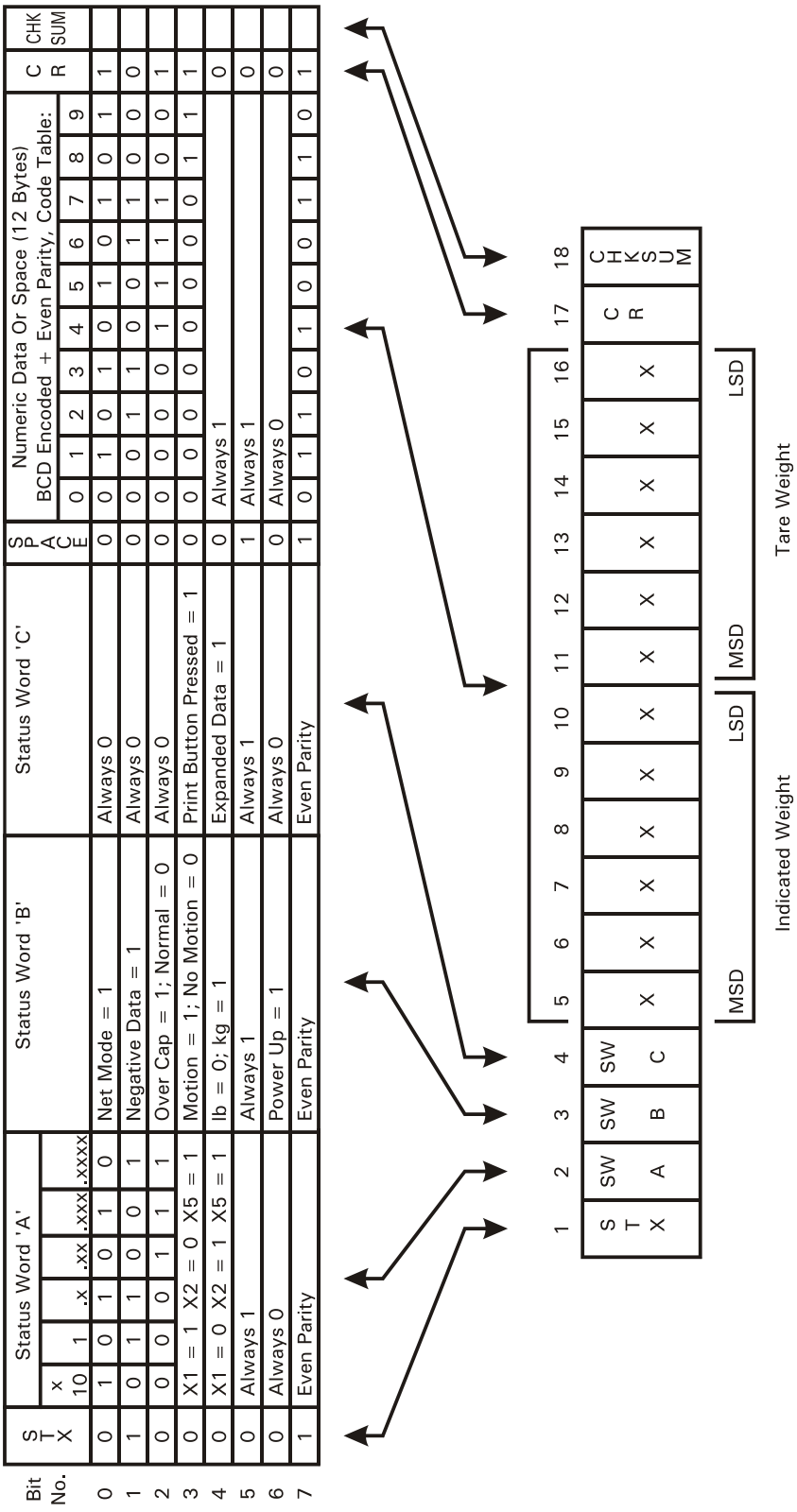
The data string for transmission of weight consists of the 2-character status information, 7 ASCII characters for the weight, 2-character unit sign plus CR and LF. Example:

'12345678901234'	Character:	
'$^{s}_{x}$– 10095KNM$^{C}_{R}$$^{L}_{F}$'	1st char.	always <STX> Hex 02
	2nd char.: Sign	— = weight negative
		␣ (space) = weight positive
	3rd–9th char.	weight 7 characters, in format of scale calibration
	10th char.: Unit	K = kg
		G = g
		T = t
		L = lb
	11th char.: Mode	G = gross weight
		N = net weight
	12th char.: Status	M = scale in motion
		␣ (space) = scale settled
	13th–14th char.	CR and LF

13 TOLEDO® Protocol

The data string emulates the essential functions of the Continuous Mode as it is output by digital indicators of the series TOLEDO® 8132, 8140, 8142, 8520, 8530 etc.

Bit No.	STX	Status Word 'A' x 10	1	.x	.xx	.xxx	.xxxx	Status Word 'B'	Status Word 'C'	SPACE	Numeric Data Or Space (12 Bytes) BCD Encoded + Even Parity, Code Table: 0	1	2	3	4	5	6	7	8	9	CR	CHK SUM
0	0	1	0	1	0	1	0	Net Mode = 1	Always 0	0	0	1	0	1	0	1	0	1	0	1	1	
1	1	0	1	1	0	0	1	Negative Data = 1	Always 0	0	0	0	1	1	0	0	1	1	0	0	0	
2	0	0	0	0	1	1	1	Over Cap = 1; Normal = 0	Always 0	0	0	0	0	0	1	1	1	1	0	0	1	
3	0	X1 = 1	X2 = 0	X5 = 1				Motion = 1; No Motion = 0	Print Button Pressed = 1	0	0	0	0	0	0	0	0	1	1	1	1	
4	0	X1 = 0	X2 = 1	X5 = 1				lb = 0; kg = 1	Expanded Data = 1	0	Always 1										0	
5	0	Always 1						Always 1	Always 1	1	Always 1										0	
6	0	Always 0						Power Up = 1	Always 0	0	Always 0										0	
7	1	Even Parity						Even Parity	Even Parity	1	0	1	1	0	1	0	0	1	1	0	1	



Transmission parameters are: 4800 baud, 7 data bits, even parity, no hardware handshake. Checksum is defined as the 2's complement of the 7 low order bits of the binary sum of the 7 low order bits of all characters preceding the checksum including STX and CR. Bit 8 of the checksum is parity over the 7 low order bits of checksum.