LOAD CELL SYSTEMS

User Manual

version 1.01

COMMUNICATIONS PROTOCOLS

TLK - TLKWF

Web: https://loadcellsys.com
Email: sales@loadcellsys.com
Phone: 607-426-1467

KEY TO SYMBOLS

Below are the symbols used in the manual to draw the reader's attention:

Caution! High Voltage.

Caution! This operation must be performed by skilled workers.

Read the following indications carefully.

Further information.

TABLE OF CONTENTS

CONTINUOUS FAST WEIGHT TRANSMISSION PROTOCOL

This protocol allows the continuous transmission of the weight at high update frequencies. Up to 200 strings per second are transmitted with a minimum transmission rate of 38400 baud (TLKWF only: in case of WiFi communication the baud rate is set by the instrument).
Following communication modes availables (see **SERIAL COMMUNICATION SETTINGS** section in instrument manual):

- *nOd t*: communication compatible with TX RS485 instruments;
- *nOd td*: communication compatible with TD RS485 instruments.

If *nOd t* is set, the following string is transmitted to PC/PLC:
xxxxxxCRLF

where: **xxxxxx**.....6 characters of gross weight (48 ÷ 57 ASCII)
CR.....1 character return to the start (13 ASCII)
LF.....1 character on new line (10 ASCII)

In case of negative weight, the first character from the left of the weight characters takes on the value "-" (minus sign - ASCII 45).
In case of error or alarm, the 6 characters of the weight are substituted by the messages found in the table of the ALARMS section (see the instrument manual).

If *nOd td* is set, the following string is transmitted to PC/PLC:
&TzzzzzzPzzzzzz\ckckCR

where: **&**.....1 initial string character (38 ASCII)
T.....1 character of gross weight identification
P.....1 character of gross weight identification
zzzzzz.....6 characters of gross weight (48 ÷ 57 ASCII)
****.....1 character of separation (92 ASCII)
ckck.....2 ASCII control characters or calculated considering the characters included between "&" and "\" excluded. The control value is obtained executing the XOR operation (exclusive OR) for the 8 bit ASCII codes of the characters considered. Therefore, a character expressed in hexadecimal is obtained with 2 numbers that may assume values from "0" to "9" and from "A" to "F". "**ckck**" is the ASCII code of the two hexadecimal digits
CR.....1 character of end string (13 ASCII)

In case of negative weight, the first character from the left of the weight characters takes on the value "-" (minus sign - ASCII 45).
In case of error or alarm, the 6 characters of the gross weight are substituted by the messages found in the table of the ALARMS section (see the instrument manual).

CONTINUOUS WEIGHT TRANSMISSION TO REMOTE DISPLAYS PROTOCOL

This protocol allows the continuous weight transmission to remote displays. The communication string is transmitted 10 times per second.
Following communication modes availables (see **SERIAL COMMUNICATION SETTINGS** section in instrument manual):

- *rI P*: communication with RIP5/20/60, RIP50SHA, RIPLED series remote displays; the remote display shows the net weight or gross weight according to its settings
- *HdrI P*: communication with RIP6100, RIP675, RIP6125C series remote displays; the remote display shows the net weight or gross weight according to its settings
- *HdrI Pn*: communication with RIP6100, RIP675, RIP6125C series remote displays

The instrument sends the following string to the remote display:
&NxxxxxxLyyyyyy\ckckCR

where: **&**.....1 initial string character (38 ASCII)
N.....1 character of net weight identification (78 ASCII)
xxxxxx.....6 characters of net weight or PEAK if present (48 ÷ 57 ASCII)
L.....1 character of gross weight identification (76 ASCII)
yyyyyy.....6 characters of gross weight (48 ÷ 57 ASCII)
\.....1 character of separation (92 ASCII)
ckck.....2 ASCII checksum characters calculated considering the characters between "**&**" and "****" excluded. The checksum value is obtained from the calculation of XOR (exclusive OR) of the 8-bit ASCII codes of the characters considered. This obtains a character expressed in hexadecimals with two digits that can have the values from "0" to "9" and from "A" to "F". "**ckck**" is the ASCII code of the two hexadecimal digits
CR.....1 character of end string (13 ASCII)

In case of negative weight, the first character from the left of the weight characters takes on the value "-" (minus sign - ASCII 45).
If *HdrI P* has been set, the decimal point at the position shown on the instrument's display can also be transmitted. In this case, if the value exceeds 5 digits, only the 5 most significant digits are transmitted, while if the value is negative, no more than the 4 most significant digits are transmitted. In both cases, however, the decimal point shifts consistently with the value to display.
If *HdrI Pn* has been set, in addition to what stated in *HdrI P* protocol, the instrument transmits the prompt *nEt* every 4 seconds in the gross weight field, if on the instrument, it has been carried out a net operation (see **SEMI-AUTOMATIC TARE (NET/GROSS)** section in instrument manual).
In case of weight value is under -99999, the minus sign "-" is sent alternated with the most significant figure.
In case of error or alarm, the 6 characters of the gross weight and net weight are substituted by the messages found in the table of the ALARMS section (see the instrument manual).

ASCII BIDIRECTIONAL PROTOCOL

The instrument replies to the requests sent from a PC/PLC.
It is possible to set a waiting time for the instrument before it transmits a response (see *dELAY* parameter in the **SERIAL COMMUNICATION SETTINGS** section in the instrument manual).
Following communication modes availables (see **SERIAL COMMUNICATION SETTINGS** section in instrument manual):

- *nOdU60*: communication compatible with instruments series W60000, WL60 Base, WT60 Base, TLA600 Base
- *nOd td*: communication compatible with TD RS485 instruments

Captions:

$.....Beginning of a request string (36 ASCII)
& or **&&** .....Beginning of a response string (38 ASCII)
aa.....2 characters of instrument address (48 ÷ 57 ASCII)
!.....1 character to indicate the correct reception (33 ASCII)
?.....1 character to indicate a reception error (63 ASCII)
#.....1 character to indicate an error in the command execution (23 ASCII)
ckck.....2 ASCII characters of Check-Sum (for further information, see **CHECK-SUM CALCULATION** section)
CR.....1 character for string end (13 ASCII)
****.....1 character of separation (92 ASCII)

1. SETPOINT PROGRAMMING

The PC transmits the following ASCII string: **$aaFffckckCR**

where: **F**.....selection command of the class to be programmed
ff.....number of the setpoint class (fixed at 01)

Possible instrument responses:

- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**
- **ff** exceeds the maximum allowable: **&aa#\ckckCR**

Example: to select the class no. 01 to program for the instrument no. 01, the PC must transmit the following command: **$01F0146(Cr)**.

1.1. SETTING SETPOINT VALUES

The PC transmits the following ASCII string: **$aaxxxxxxyckckCR**

where: **xxxxxx**.....6 characters for the setpoint value (48 ÷ 57 ASCII)
y = A .....set the value in the setpoint 1
y = B .....set the value in the setpoint 2
y = C .....set the value in the setpoint 3
y = D .....set the value in the setpoint 4

Possible instrument responses:
- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**
- **ff** parameter exceeds the maximum allowable: **&aa#\ckckCR**

Example: to set 500 in the setpoint no. 4, the PC must transmit the following command:
$01000500D70(Cr).

1.2. SETPOINT STORAGE IN EEPROM MEMORY

The setpoint are stored in the RAM memory and lost upon instrument power off. It is necessary to send a special command to save them permanently in the EEPROM memory. Please note that the writing number allowed in the EEPROM memory is limited (about 100000).

The PC transmits the following ASCII string: **$aaMEMckckCR**

Possible instrument responses:
- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**

2. READING WEIGHT, SETPOINT AND PEAK (IF PRESENT) FROM PC

The PC transmits the following ASCII string: **$aajckckCR**

where: **j** = a.....to read setpoint 1
j = b.....to read setpoint 2
j = c.....to read setpoint 3
j = d.....to read setpoint 4
j = t.....to read gross weight
j = n.....to read net weight
j = p .....to read the gross weight peak if the ASCII parameter is set as MODBUS; if, instead, the ASCII parameter is set on MOd tA the gross weight will be read. **To read the points, set the FS_tEO parameter equal to 50000**

Possible instrument responses:
- correct reception: **&aaxxxxxxj\ckckCR**
- incorrect reception: **&&aa?\ckckCR**
- In case of peak not configured: **&aa#CR**

where: **xxxxxx**.....6 characters of the required weight value

Notes: in case of negative weight, the first character from the left of the weight characters takes on the value "-" (minus sign - ASCII 45). In case of weight value is under -99999, the minus sign "-" is sent alternated with the most significant figure.

Error messages:
in case of an instrument alarm for exceeding 110% of the full scale or 9 divisions above the value of the parameter MASS, the instrument sends the string:
&aassO-Lst\ckck

in case of faulty connection of the load cells or of another alarm, the instrument sends:
&aassO-Fst\ckck

where: **s**.....1 separator character (32 ASCII – space)

Generally refer to the **ALARMS** section (see the instrument manual).

3. SEMI-AUTOMATIC ZERO (WEIGHT ZERO-SETTING FOR SMALL VARIATIONS)

The PC transmits the following ASCII string: **$aaZEROckckCR**

Possible instrument responses:

- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**
- the current weight is over the maximum resettable value: **&aa#CR**

4. SWITCHING FROM GROSS TO NET WEIGHT

The PC transmits the following ASCII string: **$aaNETckckCR**

Possible instrument responses:

- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**

5. SWITCHING FROM NET TO GROSS WEIGHT

The PC transmits the following ASCII string: **$aaGROSSckckCR**

Possible instrument responses:

- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**

6. READING OF DECIMALS AND DIVISION NUMBER

The PC transmits the following ASCII string: **$aaDckckCR**

Possible instrument responses:

- correct reception: **&aaxy\ckckCR**
- incorrect reception: **&&aa?\ckckCR**

where: **x**.....number of decimals
y = 3.....for division value = 1
y = 4.....for division value = 2
y = 5.....for division value = 5
y = 6.....for division value = 10
y = 7.....for division value = 20
y = 8.....for division value = 50
y = 9.....for division value = 100

7. TARE ZERO-SETTING

The PC transmits the following ASCII string: **$aazckckCR**

where: **z**.....command of weight zero-setting (122 ASCII)

Possible instrument responses:

- correct reception: **&aaxxxxxxt\ckckCR**
- incorrect reception: **&&aa?\ckckCR**
- the gross weight is not displayed on the instrument: **&aa#CR**

where: **xxxxxx**.....6 characters to indicate the required weight value
t.....character to indicate the weight (116 ASCII)

Example: zeroing the weight of the instrument with address 2
For the calibration you have to make sure that the system is unloaded or that the instrument measures a signal equal to the mV in the same condition:
query: **$02z78(Cr)**
response: **&02000000t\76(Cr)**

If the zeroing works correctly the instrument sends the zeroed weight value ("000000").

The calibration values are stored permanently in the EEPROM memory and the number of allowed writings is limited (about 100000).

8. REAL CALIBRATION (WITH SAMPLE WEIGHT)

After the tare zero-setting, this function allow the operator to check the calibration obtained by using sample weights and correct automatically any change between the displayed value and the actual one.

Load onto the weighing system a sample weight, which must be at least 50% of the Full Scale, or make so that that the instrument measures a corresponding mV signal.

The PC transmits the following ASCII string: **$aasxxxxxxckckCR**

where **: s**.....calibration command (115 ASCII)
xxxxxx.....6 characters to indicate the value of sample weight

Possible instrument responses:
- correct reception: **&aaxxxxxxt\ckckCR**
- incorrect reception or full scale equal to zero: **&&aa?\ckckCR**

where: **t**.....character of gross weight identification (116 ASCII)
xxxxxx.....6 characters to indicate the value of current weight

In case of correct reception, the read value has to be equal to the sample weight.

Example: calibration of the instrument no. 1 with a sample weight of 20000 kg:
query: **$01s02000070(Cr)**
response: **&01020000t\77(Cr)**

In case of correct calibration, the read value has to be “020000”.

9. KEYPAD LOCK (BLOCK THE ACCESS TO THE INSTRUMENT)

The PC transmits the following ASCII string: **$aaKEYckckCR**

Possible instrument responses:
- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**

10. KEYPAD UNLOCK

The PC transmits the following ASCII string: **$aaFREckckCR**

Possible instrument responses:
- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**

11. DISPLAY AND KEYPAD LOCK

The PC transmits the following ASCII string: **$aaKDISckckCR**

Possible instrument responses:

- correct reception: **&&aa!\ckckCR**
- incorrect reception: **&&aa?\ckckCR**

12. CHECK-SUM CALCULATION

The two ASCII characters (**ckck**) are the representation of a hexadecimal digit in ASCII characters. The check digit is calculated by executing the operation of XOR (exclusive OR) of 8-bit ASCII codes of only the string underlined.

The procedure to perform the calculation of check-sum is the following:

- Consider only the string characters highlighted with underlining
- Calculate the exclusive OR (XOR) of 8-bit ASCII codes of the characters

Example:

character	decimal ASCII code	hexadecimal ASCII code	binary ASCII code
0	48	30	00110000
1	49	31	00110001
t	116	74	01110100
XOR =	117	75	01110101

- The result of the XOR operation expressed in hexadecimal notation is made up of 2 hexadecimal digit (that is, numbers from 0 to 9 and/or letters from A to F). In this case the hexadecimal code is 0x75.
- The checksum is made up of the 2 characters that represent the result of the XOR operation in hexadecimal notation (in our example the character "7" and the character "5").

MODBUS-RTU PROTOCOL

The MODBUS-RTU protocol allows the management of the reading and writing of the following registries according to the specifications found on the reference document for this **Modicon PI-MBUS-300** standard**.**

To select the MODBUS-RTU communication see **SERIAL COMMUNICATION SETTINGS** section in instrument manual.

Check if the Master MODBUS-RTU in use (or the development tool) requires the disclosure of registers based on 40001 or 0. In the first case the registers numbering corresponds to the one in the table; in the second case the register must be determined as the value in the table minus 40001. E.g.: the register 40028 shall be reported as 27 (= 40028-40001).

Certain data, when specifically indicated, will be written directly in the EEPROM type memory. This memory has a limited number of writing operations (100000), therefore it is necessary to pay particular attention to not execute useless operations on said locations. The instrument in any case makes sure that no writing occurs if the value to be memorised is equal to the value in memory.

The numerical data found below are expressed in decimal notation; if the prefix 0x is entered the notation will be hexadecimal.

MODBUS-RTU DATA FORMAT

The data received and transmitted by way of the MODBUS-RTU protocol have the following characteristics:

- 1 start bit
- 8 bit of data, *least significant bit* sent first
- Settable parity bit
- Settable stop bit

FUNCTIONS SUPPORTED IN MODBUS

Among the commands available in the MODBUS-RTU protocol, only the following are utilised for management of communication with the instruments; other commands could be incorrectly interpreted and generate errors or blocks of the system:

FUNCTIONS	DESCRIPTION
03 (0x03)	READ HOLDING REGISTER (READ PROGRAMMABLE REGISTERS)
16 (0x10)	PRESET MULTIPLE REGISTERS (WRITE MULTIPLE REGISTERS)

Interrogation frequency is linked to the communication speed set (the instrument stands by for at least 3 bytes before starting calculations an eventual response to the interrogation query). The dELAY parameter present in the **SERIAL COMMUNICATION SETTING** section in the instrument manual, allows the instrument to respond with a further delay and this directly influences the number of interrogations possible in the unit of time.
For additional information on this protocol refer to the general technical specifications PI_MBUS_300.

In general queries and answers toward and from one slave instrument are composed as follows:

FUNCTION 3: Read holding registers (READ PROGRAMMABLE REGISTERS)

QUERY

Address	Function	1st register address	No. registers	2 byte
A	0x03	0x0000	0x0002	CRC

Tot. byte = 8

RESPONSE

Address	Function	No. bytes	1st register	2nd register	2 byte
A	0x03	0x04	0x0064	0x00C8	CRC

Tot. byte = 3+2*No. registers+2

where: No. registers ..... number of Modbus registers to write beginning from the address no. 1
No. byte ..... number of bytes of the following data

FUNCTION 16: Preset multiple registers (WRITE MULTIPLE REGISTERS)

QUERY

Address	Function	1st reg. add.	No. reg.	No. bytes	Val.reg.1	Val.reg.2	2 byte
A	0x10	0x0000	0x0002	0x04	0x0000	0x0000	CRC

Tot. byte = 7+2*No. registers+2

RESPONSE

Address	Function	1st reg. address	No. reg.	2 byte
A	0x10	0x0000	0x0002	CRC

Tot. byte = 8

where: No. registers ..... number of Modbus registers to read beginning from the address no. 1
No. byte ..... number of bytes of the following data
Val.reg.1 ..... contents of the register beginning from the first

The response contains the number of registers modified beginning from the address no. 1.

COMMUNICATION ERROR MANAGEMENT

The communication strings are controlled by way of the CRC (Cyclical Redundancy Check).
In case of communication error the slave will not respond with any string. The master must consider a time-out for reception of the answer. If it does not receive an answer it deduces that there has been a communication error.

In the case of the string received correctly but not executable, the slave responds with an EXCEPTIONAL RESPONSE. The "Function" field is transmitted with the msb at 1.

EXCEPTIONAL RESPONSE

Address	Function	Code	2 byte
A	Funct + 0x80		CRC

CODE	DESCRIPTION
1	ILLEGAL FUNCTION (the function is not valid or is not supported)
2	ILLEGAL DATA ADDRESS (the specified data address is not available)
3	ILLEGAL DATA VALUE (the data received has an invalid value)

LIST OF AVAILABLE REGISTERS

The MODBUS-RTU protocol implemented on this instrument can manage a maximum of 32 registers read and written in a single query or response.

R.....the register may only be read
W.....the register may only be written
R/W .....the register may be both read and written
H.....high half of the DOUBLE WORD containing the number
L .....low half of the DOUBLE WORD containing the number

Register	Description	Saving in EEPROM	Access
40001	Firmware version	-	R
40002	Instrumento type	-	R
40003	Year of manufacture	-	R
40004	Serial number	-	R
40005	Program type	-	R
40006	COMMAND REGISTER	NO	R/W
40007	STATUS REGISTER	-	R
40008	GROSS WEIGHT H	-	R
40009	GROSS WEIGHT L	-	R
40010	NET WEIGHT H	-	R
40011	NET WEIGHT L	-	R
40012	PEAK WEIGHT H	-	R
40013	PEAK WEIGHT L	-	R
40014	Divisions and Units of measure	-	R
40015	Coefficient H	-	R
40016	Coefficient L	-	R
40017	INPUTS	-	R
40018	OUTPUTS	NO	R/W
40019	SETPOINT 1 H	Only after command 99 of the Command Register	R/W
40020	SETPOINT 1 L		R/W
40021	SETPOINT 2 H		R/W
40022	SETPOINT 2 L		R/W
40023	SETPOINT 3 H		R/W
40024	SETPOINT 3 L		R/W
40025	SETPOINT 4 H		R/W
40026	SETPOINT 4 L		R/W
40039	HYSTERESIS 1 H		R/W
40040	HYSTERESIS 1 L		R/W
40041	HYSTERESIS 2 H		R/W
40042	HYSTERESIS 2 L		R/W
40043	HYSTERESIS 3 H		R/W
40044	HYSTERESIS 3 L		R/W
40045	HYSTERESIS 4 H		R/W
40046	HYSTERESIS 4 L		R/W

40065	Sample weight for instrument calibration H	Use with command 101 of the Command Register	R/W
40066	Sample weight for instrument calibration L		R/W
40073	Preset Tare H	Use with command 130 of the Command Register	R/W
40074	Preset Tare L		R/W

Setpoint and hysteresis values are lost upon instrument power off; to save them in EEPROM, so that they are maintained upon instrument power on, it is necessary to send the command 99 “Save data in EEPROM” of the Command Register.

SPECIAL REGISTERS

STATUS REGISTER (40007)

Bit 0	Load cell error
Bit 1	AD convertor malfunction
Bit 2	Maximum weight exceeded by 9 divisions
Bit 3	Gross weight higher than 110% of full scale
Bit 4	Gross weight beyond 999999 or less than -999999
Bit 5	Net weight beyond 999999 or less than -999999
Bit 6	
Bit 7	Gross weight negative sign
Bit 8	Net weight negative sign
Bit 9	Peak weight negative sign
Bit 10	Net display mode
Bit 11	Weight stability
Bit 12	Weight within ±¼ of a division around ZERO
Bit 13	
Bit 14	
Bit 15	

INPUTS AND OUTPUTS REGISTERS

INPUTS REGISTER (40017)
(read only)

Bit 0	INPUT 1 status
Bit 1	INPUT 2 status
Bit 2	
Bit 3	
Bit 4	
Bit 5	
Bit 6	
Bit 7	
Bit 8	
Bit 9	
Bit 10	
Bit 11	
Bit 12	
Bit 13	
Bit 14	
Bit 15	

OUTPUTS REGISTER (40018)
(read only)

Bit 0	OUTPUT 1 status
Bit 1	OUTPUT 2 status
Bit 2	OUTPUT 3 status
Bit 3	OUTPUT 4 status
Bit 4	
Bit 5	
Bit 6	
Bit 7	
Bit 8	
Bit 9	
Bit 10	
Bit 11	
Bit 12	
Bit 13	
Bit 14	
Bit 15	

The output status can be read at any time but can be set (written) only if the output has been set as *PLC* (see **OUTPUTS AND INPUTS CONFIGURATION** section); otherwise, the outputs will be managed according to the current weight status with respect to the relevant setpoint.

DIVISION AND UNITS OF MEASURE REGISTER (40014)

This register contains the current setting of the divisions (parameter dIUI5) and of the units of measure (parameter Unit).

H Byte	L Byte
Unit of measure	Division

Use this register together with the Coefficient registers to calculate the value displayed by the instrument.

Least significant byte (L Byte)

Division value	Divisor	Decimals
0	100	0
1	50	0
2	20	0
3	10	0
4	5	0
5	2	0
6	1	0
7	0.5	1
8	0.2	1
9	0.1	1
10	0.05	2
11	0.02	2
12	0.01	2
13	0.005	3
14	0.002	3
15	0.001	3
16	0.0005	4
17	0.0002	4
18	0.0001	4

Most significant byte (H Byte)

Unit of measure value	Unit of measure description	Coefficient effect on the read gross weight
0	Kilograms	No effect
1	Grams	No effect
2	Tons	No effect
3	Pounds	No effect
4	Newton	Multiplies
5	Litres	Divides
6	Bar	Multiplies
7	Atmospheres	Multiplies
8	Pieces	Divides
9	Newton Metres	Multiplies
10	Kilogram Metres	Multiplies
11	Other	Multiplies

POSSIBLE COMMANDS TO BE SENT TO THE COMMAND REGISTER (40006)

0	No command	**1**	
6		**7**	SEMI-AUTOMATIC TARE enabling (net weight displaying)
8	SEMI-AUTOMATIC ZERO	**9**	SEMI-AUTOMATIC TARE disabling (gross weight displaying)
20		**21**	Keypad lock
22	Keypad and display unlock	**23**	Keypad and display lock
98		**99**	Save data in EEPROM
100	TARE WEIGHT ZERO SETTING for calibration	**101**	Sample weight storage for calibration
130	Preset Tare enabling	**131**	

REAL CALIBRATION (WITH SAMPLE WEIGHTS)

The instrument calibration can be changed via MODBUS.
To set correctly the sample weight, consider the value of the Division register (40014); e.g.: if you want to set the value to 100 kg and the division value is 0.001, set the register value to 100000 (100/0.001 = 100000).
To carry out this procedure, the system must be unloaded and the weight value display reset to zero with the command 100 of the Command Register. Then, a sample weight must be placed on the system and the correct weight value must be sent to the registers 40065-40066; to save this value, send the command 101 of the Command Register. If the operation is successfully completed, the two sample weight registers are set to zero.

USE OF SETPOINT

Setpoint management is done by reading and writing registers 40019 – 40028.

Warning: the new values of the setpoint are active immediately after the writing.

COMMUNICATION EXAMPLES

The numerical data below are expressed in hexadecimal notation with prefix h.

EXAMPLE 1
Command for multiple writing of registers (command 16, h10 hexadecimal).
Assuming that we wish to write the value 0 to the register 40019 and the value 2000 to the register 40020, the string to generate must be:

h01 h10 h00 h12 h00 h02 h04 h00 h00 h07 hD0 h70 hD6

The instrument will respond with the string:

h01 h10 h00 h12 h00 h02 hE1 hCD

Query field name	hex	Response field name	hex
Instrument address	**h01**	Instrument address	**h01**
Function	**h10**	Function	**h10**
Address of the first register H	**h00**	Address of the first register H	**h00**
Address of the first register L	**h12**	Address of the first register L	**h12**
Number of registers to send H	**h00**	Number of registers H	**h00**
Number of registers to send L	**h02**	Number of registers L	**h02**
Byte count	**h04**	CRC16 L	**hE1**
Datum 1 H	**h00**	CRC16 H	**hCD**
Datum 1 L	**h00**		
Datum 2 H	**h07**		
Datum 2 L	**hD0**		
CRC16 L	**h70**		
CRC16 H	**hD6**		

EXAMPLE 2
Command for multiple writing of registers (command 16, h10 hexadecimal).
Assuming that we wish to write the two setpoint values on the instrument, at 2000 and 3000 respectively, the string must be sent:

h01 h10 h00 h12 h00 h04 h08 h00 h00 h07 hD0 h00 h00 h0B hB8 h49 h65

The instrument will respond with the string:

h01 h10 h00 h12 h00 h04 h61 hCF

Query field name	hex	Response field name	hex
Instrument address	**h01**	Instrument address	**h01**
Fuction	**h10**	Function	**h10**
Address of the first register H	**h00**	Address of the first register H	**h00**
Address of the first register L	**h12**	Address of the first register L	**h12**
Number of registers H	**h00**	Number of registers H	**h00**
Number of registers L	**h04**	Number of registers L	**h04**
Byte count	**h08**	CRC16 L	**h61**
Datum 1 H	**h00**	CRC16 H	**hCF**
Datum 1 L	**h00**		
Datum 2 H	**h07**		
Datum 2 L	**hD0**		
Datum 3 H	**h00**		
Datum 3 L	**h00**		
Datum 4 H	**h0B**		
Datum 4 L	**hB8**		
CRC16 L	**h49**		
CRC16 H	**h65**		

EXAMPLE 3
Multiple commands reading for registers (command 3, h03 hexadecimal).
Assuming that we wish to read the gross weight value (in the example 4000) and net weight value (in the example 3000), reading from address 40008 to address 40011 must be performed by sending the following string:

h01 h03 h00 h07 h00 h04 hF5 hC8

The instrument will respond with the string:

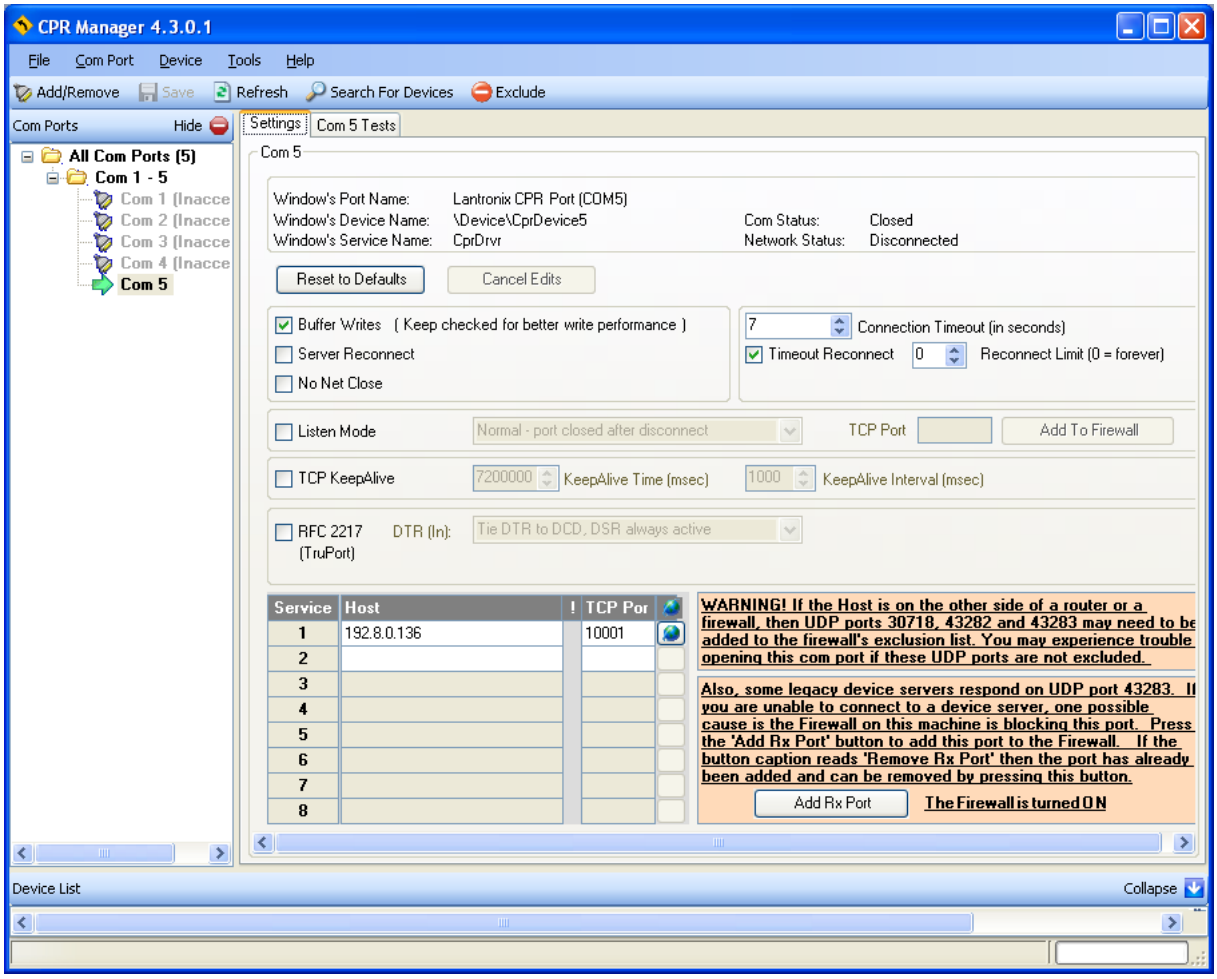
h01 h03 h08 h00 h00 h0F hA0 h00 h00 h0B hB8 h12 h73

Query field name	hex	Response field name	hex
Instrumenta address	**h01**	Instrument address	**h01**
Function	**h03**	Function	**h03**
Address of the first register H	**h00**	Byte Count	**h08**
Address of the first register L	**h07**	Datum 1 H	**h00**
Number of registers H	**h00**	Datum 1 L	**h00**
Number of registers L	**h04**	Datum 2 H	**h0F**
CRC16 L	**hF5**	Datum 2 L	**hA0**
CRC16 H	**hC8**	Datum 3 H	**h00**
		Datum 3 L	**h00**
		Datum 4 H	**h0B**
		Datum 4 L	**hB8**
		CRC16 L	**h12**
		CRC16 H	**h73**

For additional examples regarding the generation of correct control characters (CRC16) refer to the manual **Modicon PI-MBUS-300.**

PC SETUP

A PC can be connected, by a virtual serial port, to the instrument via ethernet TCP/IP.
To install the virtual COM port, use the CPR Manager included in the supply: run file *CPR.exe* on CD, add a serial port, set an IP address (host) and a TCP port (10001), then save.



Use the just created virtual COM port to communicate with the instrument, using the protocol selected on it.
Alternatively connect to the instrument using a socket (e.g.: Winsock) on port 10001.

OUTPUTS AND INPUTS CONFIGURATION

MENU + ESC → OUt-1 n:

OUTPUTS

The outputs are set by default as follows: OPEn / SEt / GrOSS / POSnEG / OFF.

Possible operation modes:

- OPEn **(normally open)**: the relay is de-energised and the contact is open when the weight is lower than the programmed setpoint value; it closes when the weight is higher than or equal to the programmed setpoint value.
- CLOSE **(normally closed)**: the relay is energised and the contact is closed when the weight is lower than the programmed setpoint value; it opens when the weight is higher than or equal to the programmed setpoint value.

- SEt: the contact will switch on the basis of weight, according to setpoint (see **SETPOINT PROGRAMMING** section in the instrument manual).
- PLC: the contact will not switch on the basis of weight, but is controlled by remote protocol commands.
- StAbLE: relay switching occurs when the weight is stable.

If the operation mode SEt is selected, the following options are also active:

- GrOSS: the contact will switch on the basis of gross weight.
- nEt: the contact will switch on the basis of net weight (If the net function is not active, the contact will switch on the basis of gross weight).

- POSnEG: relay switching occurs for both positive and negative weight values.
- POS: relay switching occurs for positive weight values only.
- nEG: relay switching occurs for negative weight values only.

By confirming with ENTER the setpoint operation can be set to the value '0':

- OFF: relay switching will not occur if the setpoint value is '0'.
- On:
    - Setpoint = 0 and relay switching = POSnEG, relay switching occurs when the weight is 0; the relay will switch again when the weight is different from zero, taking hysteresis into account (both for positive and for negative weights).
    - Setpoint = 0 and relay switching = POS, relay switching occurs for a weight higher than or equal to 0, the relay will switch again for values below 0, taking hysteresis into account.
    - Setpoint = 0 and relay switching = nEG**,** relay switching occurs for a weight lower than or equal to 0, the relay will switch again for values above 0, taking hysteresis into account.

INPUTS

Default: input 1 = 2Er0 input 2 = nE-L0

Possible operation modes:

- nE-L0 (NET/GROSS): by closing this input for no more than one second, it's making an operation of SEMI-AUTOMATIC TARE and the display will show the net weight. To display the gross weight again, hold the NET/GROSS input closed for 3 seconds.
- 2Er0: by closing the input for no more than one second, the weight is set to zero (see **WEIGHT ZERO-SETTING FOR SMALL VARIATIONS (SEMI-AUTOMATIC ZERO)** section in the instrument manual).
- PEAH: keeping the input closed the maximum weight value reached remains on display. Opening the input the current weight is displayed.
- PLC: closing the input no operation is performed, the input status may however be read remotely by way of the communication protocol.
- C0nt1 n: closing the input for max one second the weight is transmitted over the serial connection according to the fast continuous transmission protocol only once (**only if C0nt1 n is set in the item SEr1 AL**).
- C0EFF: when the input is closed the weight is displayed based on the set coefficient (see setting of the units of measure and coefficient), otherwise the weight is displayed.
- Pr1 ntr: when the input is closed the data are sent for printing if in the communication protocol of either serial port the parameter Pr1 ntr is set.

Web: https://loadcellsys.com
Email: sales@loadcellsys.com
Phone: 607-426-1467