PENKO Engineering B.V.

Your Partner for Fully Engineered Factory Solutions

Manual:
1020 Supplement Check Weigher Controller

PENKO
an ETC Company

Table of Contents

1 Introduction

This manual is applicable for the following Check Weigher devices:

- 1020 CHK
- 1020 RS232-RS422 CHK
- 1020 Profibus CHK
- 1020 Profinet CHK

To configure and control the Check Weigher, the following options are available:

Full control:

- PENKO Pi Mach II software
- PENKO PDI Client software
- Modbus protocol
- Profibus protocol
- EtherNet/IP protocol
- ASCII protocol
- Profinet protocol

Basic control:

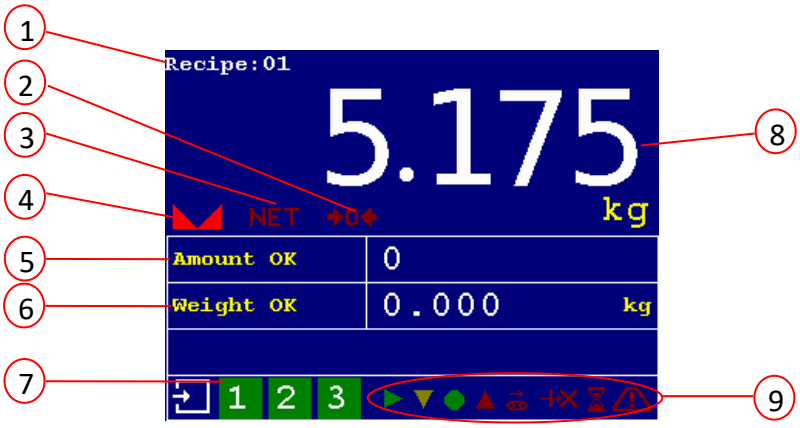
- Fins protocol*
- PENKO TP protocol*

** Register functions not available*

Note:

This manual does not describe the basic functionality of the device. Consult the device manual for this.

2 Indication of display

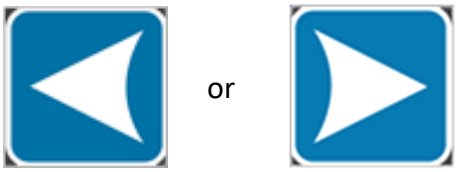


1	Current selected recipe	**6**	Total correct product weight
2	Zero active	**7**	Inputs 1, 2, 3
3	Tare active	**8**	Value
4	Weigher stable	**9**	Status Indications
5	Total correct product amount		

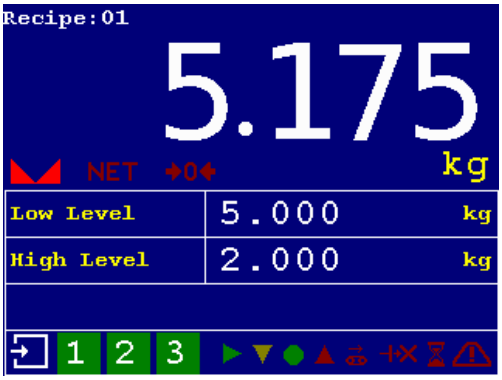
Options for indication 2nd screen

Use the LEFT or RIGHT key to switch between the four main screens.

The 2nd screen shows the flow as the largest indication.



or

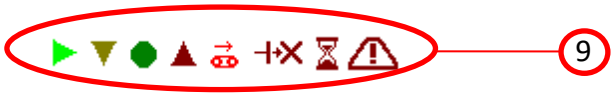


1020 Check Weigher

5a Currently selected low level from recipe

5b Currently selected high level from recipe

Status Indications:




Start


Low – last checked value was too low


OK – last checked value was ok


High – last checked value was too high


Transport – transport belt running


Reject – reject output active


Busy – check weighing in progress


Alarm – alarm output active

3 Configure and control

To configure and control the Check Weigher, the following options are available:

- PENKO configuration software
- Device
- Industrial protocols

3.1 PENKO configuration software

PENKO Pi Mach II and PENKO PDI Client can be downloaded from www.penko.com

USB driver and user manual are included in the download

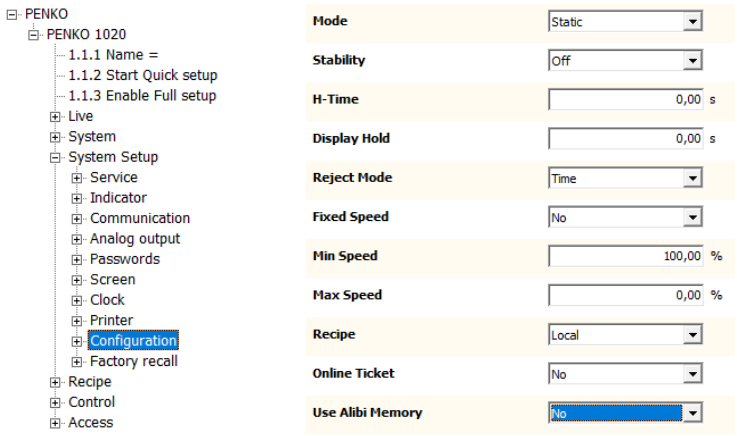
Pi Mach II supports USB and Ethernet connection. PDI Client is USB only.

Consult the manuals on how to install and connect to the device.

In the tree structure of the device, the configuration parameters are found at:

PENKO - PENKO 1020 - System Setup - Configuration

Configuration parameters

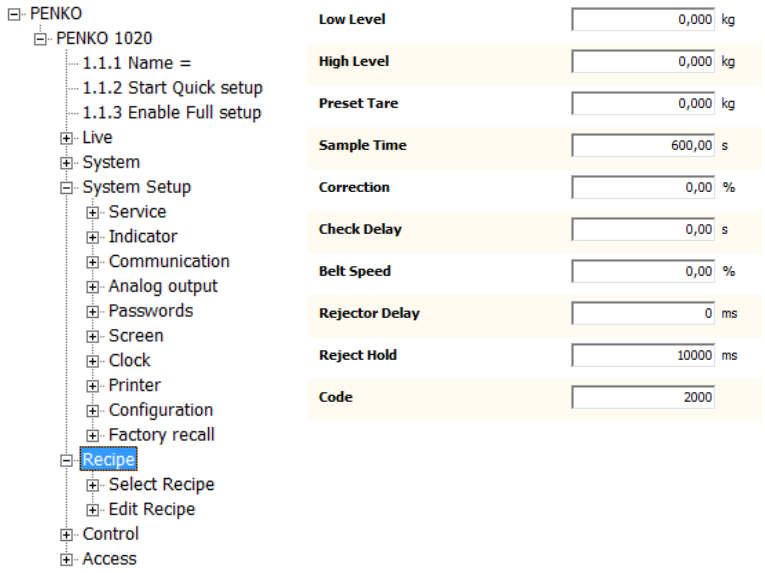


The parameters are explained in chapters parameter

In the tree structure of the device, the recipe parameters are found at:

PENKO - PENKO 1020 - Recipe

Recipe parameters

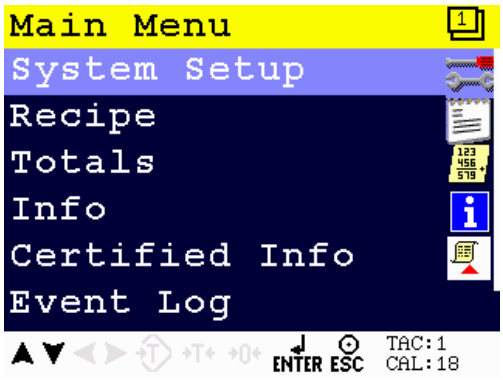


The parameters are explained in chapters parameter

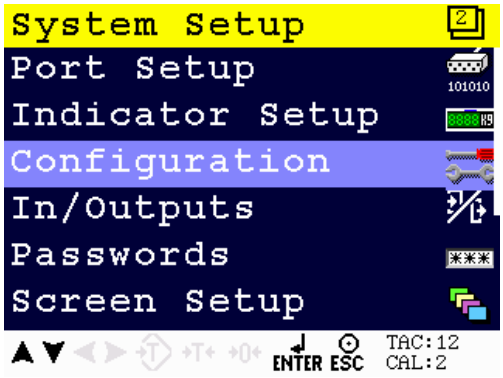
3.2 Device

Configuration

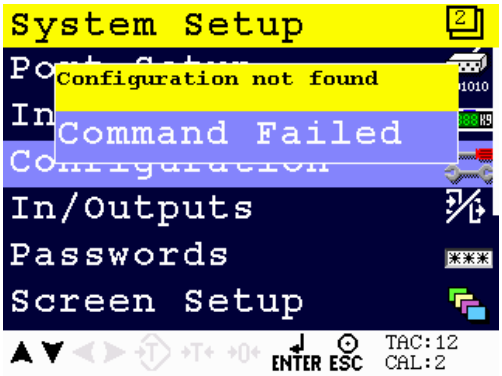
Select **System Setup** from the **Main Menu** and press **Enter**.



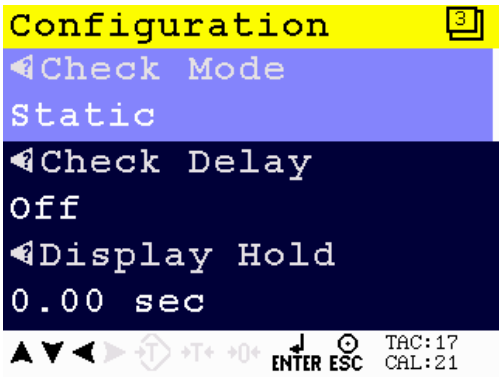
Select **Configuration** from the **System Setup Menu** and press **Enter**.



The following error is visible if no configuration is present.

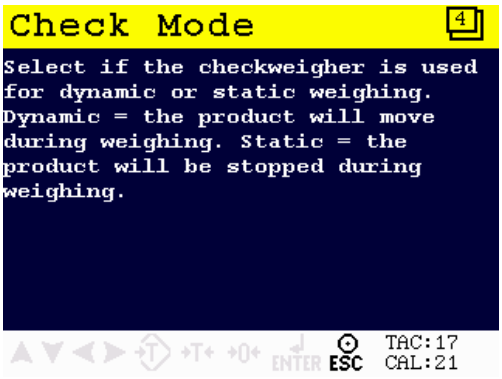


Press **Enter** to start with default values.



When pushing the LEFT key, the help text of the parameter is shown.

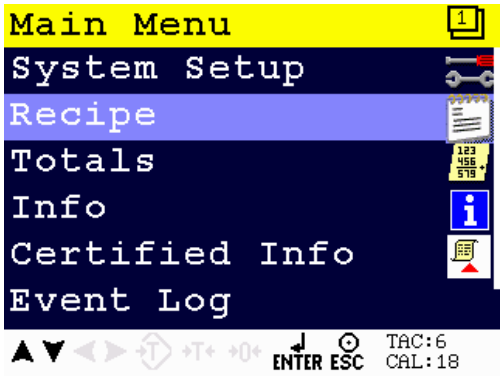
Below the example of a help text for the parameter Check Mode.



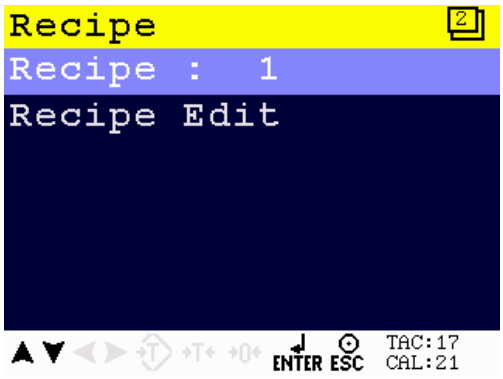
The parameters are explained in chapters parameter

Recipe

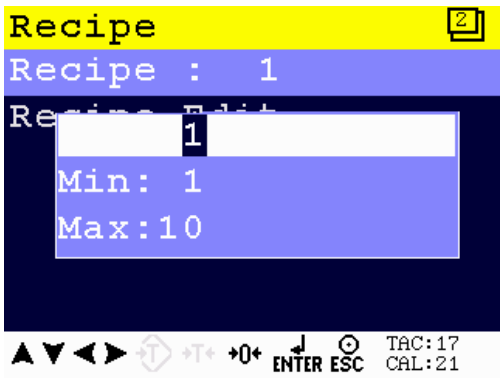
Select **Recipe** from the **Main Menu** and press **Enter**.



Select **Recipe** and press **Enter**.

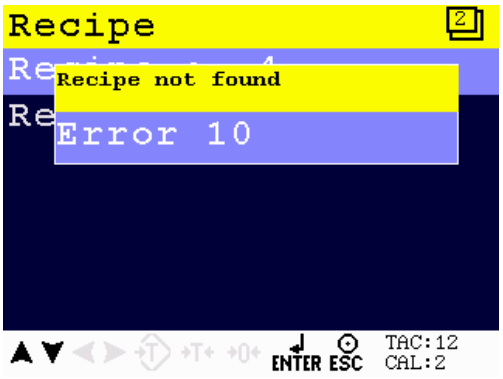


Enter the recipe that you want to edit and press **Enter**.

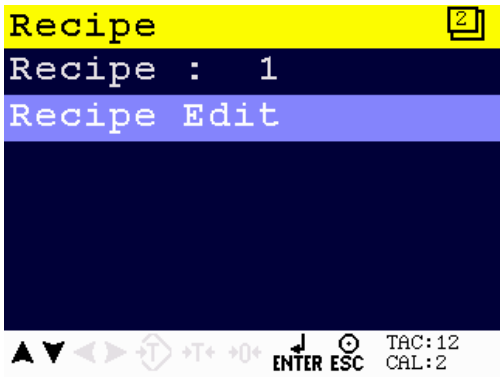


A maximum of 10 recipes can be stored.

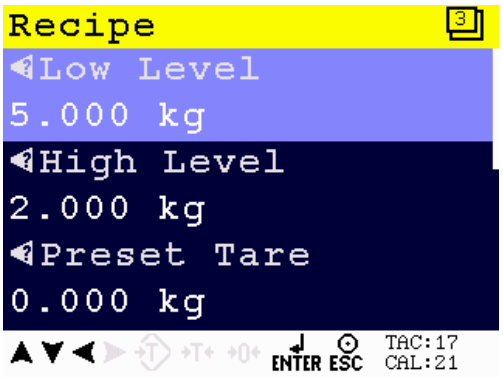
If the selected recipe does not exist, the following error is visible:



To edit current selected recipe parameters, select **Recipe Edit** and press **Enter**.

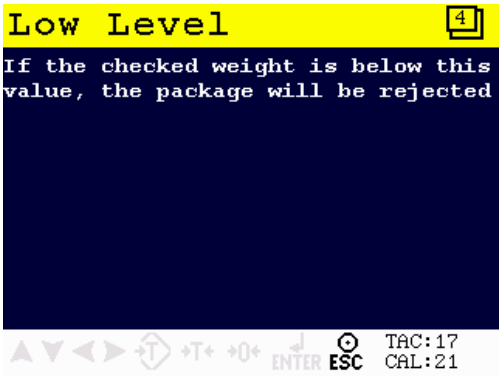


The following screen is visible:



When pushing the LEFT key, the help text of the parameter is accessed.

Below an example of the help text for the parameter Low Level.



The parameters are explained in chapters parameter

4 Parameters

These parameters correspond with the parameters in the tree structure of the device Recipe. When using the industrial protocol register functions, each parameter can be reached using its number.

Some parameters can be reached directly using ASCII, TP protocol, Modbus RTU, Modbus TCP, Fins, Profibus, Profinet or EtherNet/IP.

Note: when the device is rebooted or the recipe is manually changed, all recipe parameters are changed back to the value that were last set manually in the recipe.

4.1 Configuration parameters

No.	Name	Description
1	Check mode	Select if the check weigher is used for dynamic or static weighing. • Dynamic => the product will move during weighing (belt is running). • Static => the product will be stopped during weighing (belt is stopped when check input is high and restarted when check is done).
2	Stability	Select the type of check delay. In static check mode the belt will stop during the check delay. • Off => check starts directly • Stable => check starts after Stable • H-Time => check starts after H-time • H-Time+Stable => check starts after H-time and then Stable • H-Time/Stable => check starts after H-time or Stable • Stable+H-Time => check starts after Stable and then H-Time
3	H-Time	Time to wait for stable weight measurement. **Δ** *Not used if Stability parameter is set to Off or Stable*
4	Display hold	The time to freeze the checked value on the display after the check is done. When a check starts before the display hold time is elapsed, the hold time will be cut off. **Δ** *Not used if Stability parameter is set to Off or Stable*

5	Rejector mode	Select when the packages should be rejected. • Time => reject after a set time (settings are in recipe) • Photocell => reject when passing the photocell (input 3)
6	Fixed speed	Select if the belt speed is fixed or variable. • No => belt speed is variable between set minimum and maximum speed and is controlled by the analog output • Yes => belt speed is fixed (setting is in recipe)
7	Min. speed	Enter the minimum variable belt speed.
8	Max. speed	Enter the maximum variable belt speed.
9	Recipe	Select the used recipe. • Local => use the recipe selected on the device • Remote => use the recipe from a remote device (e.g. PLC)
10	Online ticket	Select if a printer ticket must be printed for each checked product.
11	Use alibi memory	Select if a result must be written to the internal alibi memory.

4.2 Recipe parameters

These parameters correspond with the parameters in the tree structure of the device Recipe. When using the industrial protocol register functions, each parameter can be reached using its number.

Some parameters can be reached directly using ASCII, TP protocol, Modbus RTU, Modbus TCP, Fins, Profibus, Profinet or EtherNet/IP.

Note: when the device is rebooted or the recipe is manually changed, all recipe parameters are changed back to the value that were last set manually in the recipe.

No.	Name	Description
1	Low level	If the checked value is below this value, the package will be rejected.
2	High level	If the checked value is above this value, the package will be rejected.
3	Preset tare	The preset tare will be subtracted from the checked value.
4	Sample time	Duration of the package weight measurement.
5	Correction	Correction factor to correct the fault caused by the dynamic characteristics of the machine. Correction can be set from 0% to 200%. 100% means no correction is used.
6	Check delay	Enter the time between detecting a product and start checking or stopping the belt.
7	Belt speed	The speed of the transport belt in %
8	Rejector delay	Delay time to activate the rejector after detecting a faulty package.
9	Reject hold	Time to hold the rejector output active for a certain time after activating.
10	Batch code	Enter a Batch code for printing reports.

4.3 Live process parameters

When using the industrial protocol register functions, each parameter can be read using its number.

Example: to read the value of low level, Use the function code 701 and value 1.

No.	Name	Description
1	Low level	Get the low level value.
2	High level	Get the high level value.
3	Subtotal std.dev	Get the standard deviation of the subtotal.
4	Subtotal average	Get the average value of the subtotal.
5	Subtotal (weight) ok	Get the weight of the accepted products of the subtotal.
6	Subtotal count ok	Get the number of accepted product of the subtotal.
7	Subtotal count low	Get the number of too low product of the subtotal.
8	Subtotal count high	Get the number of too high product of the subtotal.
9	Subtotal count total	Get the number of all product of the subtotal.
10	Total std.dev	Get the standard deviation of the total batch.
11	Total average	Get the average value of the total batch.
12	total (weight) ok	Get the weight of the accepted products of the total batch.
13	Total count ok	Get the number of accepted product of the total batch.
14	Total count low	Get the number of too low product of the total batch.
15	Total count high	Get the number of too high product of the total batch.
16	total count total	Get the number of all product of the total batch.
17	Alibi no.	Get the number of the Alibi record.

5 Inputs and outputs

The following inputs and outputs are used.

5.1 Inputs

Input	Name	Description
1	Start/Stop	Input must be high to set the program in run mode.
2	Start Sampling	Input to start sampling the package.
3	Reject Sensor	Input to detect the package on the reject position. Input must be placed diagonal over the belt.

5.2 Outputs

Output	Name	Description
1	Rejector	Output to enable the rejector after the package is detected on the reject position.
2	Sampling Busy	Output to enable the busy output. Output is high from detecting the package until sampling is done.
3	Transport Belts	Output to enable the transport belts. In dynamic mode the output is always on if no alarm is generated. In static mode the output is off during sampling and during an indicator alarm.
4	Alarm	Output to enable the indicator alarm. This output is used to activate the indicator alarm. This alarm can be reset by switching off input 1.

5.3 Analog output

Output	Name	Description
Analog out	Belt Speed	Belt Speed will be available from 0.00% to 100.00%.

6 Printer Ticket

Example of the 1020 Printer recipe when 'Ticket' layout is selected.

```
  Programmable header 1

  Programmable header 2

  Programmable header 3

  Programmable header 4

  ------------------------------------
---

  DATE                     07-10-11

  TIME                     05:57.13

  RECIPE                        001

  TICKET NUMBER                 100


  CHECKED NET              00000.00 kg

  ------------------------------------
---

  Programmable footer 1

  Programmable footer 2
```

7 Program basics

This chapter describes a few basics of the Check Weigher program which can be used when starting the program for the first time.

7.1 Static check mode, Reject mode set to Time

With input 1 you can Start (input high) and Stop (input low) the program. The Transport Belt (output 3) will start. With a pulse on Start Sampling (input 2) the checking will start. During checking the Transport Belt (output 3) will stop and Sample Busy (output 2) is on.

When the checking is complete and the weight is ok, the Transport belt (output 3) will start again, Sample Busy (output 2) is off. With a pulse on Start Sampling (input 2) the checking will start again.

If the weight is below Low Level or above High Level the Transport belt (output3) will start again. After the Reject Delay time, the Rejector (output 1) will turn on, and will remain on for the duration of the Reject Hold time. With a pulse on Start Sampling (input 2) the checking will start again.

7.2 Static check mode, Reject mode set to Photocell

With input 1 you can Start (input high) and Stop (input low) the program. The Transport Belt (output 3) will start. With a pulse on Start Sampling (input 2) the checking will start. During checking the Transport Belt (output 3) will stop and Sample Busy (output 2) is on.

When the checking is complete and the weight is ok, the Transport belt (output 3) will start again, Sample Busy (output 2) is off. With a pulse on Start Sampling (input 2) the checking will start again.

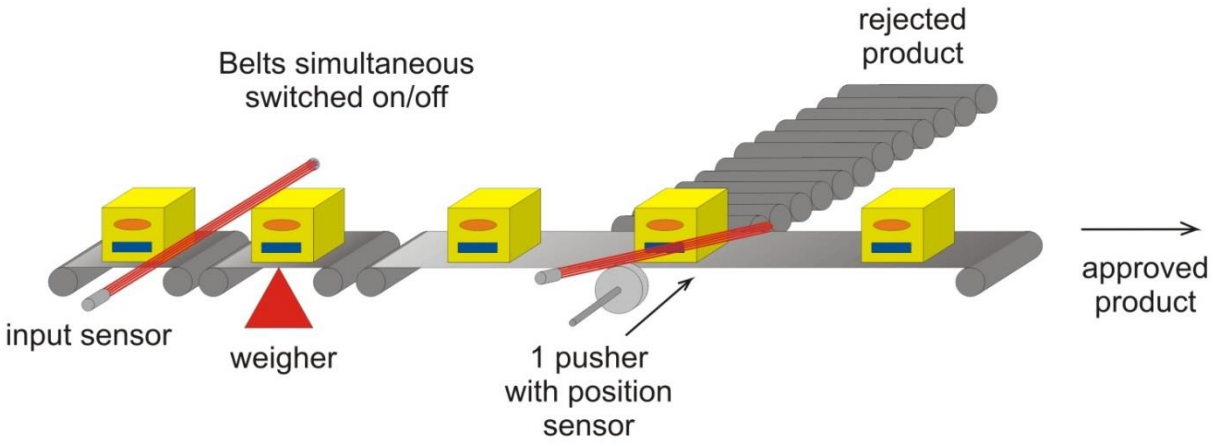
If the weight is below Low Level or above High Level the Transport belt (output3) will start again. When the photocell (connected to input 3) has detected the product, the Rejector (output 1) will turn on, and will remain on as long as input 3 (photocell) in on. With a pulse on Start Sampling (input 2) the checking will start again.

7.3 Dynamic check mode, Reject mode set to Time

With input 1 you can Start (input high) and Stop (input low) the program. The Transport Belt (output 3) will start. With a pulse on Start Sampling (input 2) the checking will start. During the checking the Sample Busy (output 2) is on.

When the checking is complete and the weight is ok. The Transport belt (output 3) will start again, Sample Busy (output 2) is off. With a pulse on Start Sampling (input 2) the checking will start again.

If the weight is below Low Level or above High Level the Transport belt (output3) will start again. After the Reject Delay time, the Rejector (output 1) will turn on, and will remain on for the duration of the Reject Hold time. With a pulse on Start Sampling (input 2) the checking will start again.



8 Default settings

Configuration:

Configuration	Static check mode, Reject mode set to Time	Static check mode, Reject mode set to Photocell	Dynamic check mode, Reject mode set to Time
Check Mode	Static	Static	Dynamic
Stability	Stable + H-Time	Stable + H-Time	Stable + H-Time
H-Time	1.00 sec	1.00 sec	1.00 sec
Display Hold **(Disabled if stability parameter is set to off or stable)**	1.00 sec	1.00 sec	1.00 sec
Rejector Mode	Time	Photocell	Time
Fixed Speed	Yes	Yes	Yes
Min. Speed	0.00%	0.00%	0.00%
Max. Speed	100.00%	100.00%	100.00%
Recipe	Local	Local	Local
Online Ticket	No	No	No
Use Alibi Memory	No	No	No

1020 Check Weigher

To access the DAC setup, select **In/Outputs** from the **System Setup Menu** and press **Enter**. Select **DAC Setup** and press **Enter**. If you don't have the DAC output, **In/Outputs** is not available.

DAC setup	Setting
Indicator	Speed
Min	0.00%
Max	100.00%
Mode	4 – 20 mA

To access the Weigher setup, select **Indicator Setup** from the **System Setup Menu** and press **Enter**. Select **Indicator** and press **Enter,** enter the **TAC code** (the TAC code is visible in the bottom right corner of the LCD screen) and press **Enter**. Select **Weigher** and press **Enter**.

Weigher	Setting
Unit Label	Kg
Step	1
Decimal point	0.00
Operation Mode	Industrial
Max Load	1000.00

To access the Stable Condition setup, select **Indicator Setup** from the **System Setup Menu** and press **Enter**. Select **Indicator** and press **Enter,** enter the **TAC code** (the TAC code is visible in the bottom right corner of the LCD screen) and press **Enter**. Select **Stable Condition** and press **Enter**.

Stable Condition	Setting
Range	0.10 kg
Time	0.50 sec

To access the Stable Condition setup, select **Indicator Setup** from the **System Setup Menu** and press **Enter**. Select **Indicator** and press **Enter,** enter the **TAC code** (the TAC code is visible in the bottom right corner of the LCD screen) and press **Enter**. Select **Filter** and press **Enter**. Select **Digital** and press **Enter**.

Filter Digital	Setting
Digital Filter	Dynamic App.
Cutoff Frequency	1.0 Hz
Frequency	10 Hz

9 Industrial protocols

The PENKO protocols Modbus, Profibus, EtherNet/IP and ASCII have a function set called register functions. These functions allow the user to configure and control the device.

Protocol descriptions can be downloaded from www.penko.com

Consult these on how to connect the device and use the register functions.

	1020	1020 RS232/422	1020 Profibus	1020 Profinet
Modbus TCP	✓	✓	✓	✓
Modbus SERIAL		✓		
Profibus			✓	
EtherNet/IP	✓	✓	✓	✓
ASCII TCP	✓	✓	✓	✓
ASCII SERIAL		✓		
Profinet IO				✓

Note: the FINS and PENKO TP protocol do not support register functions, only basic read and write operations for markers and registers.

9.1 Modbus

Below you will find a list with the data offset to read and write the data. When writing data, don't exceed the length of the data. This will cause a negative effect in the program.

	Name	Access Type	Trigger	READ Offset	Length	Error Handling	WRITE Offset	Length
0	Indicators	Read Input Registers (Function Code 04)	Cyclic, t#100ms	16#0064	50	Keep last value		
1	Inputs	Read Discrete Inputs (Function Code 02)	Cyclic, t#100ms	16#0000	3	Keep last value		
2	Outputs	Read Discrete Inputs (Function Code 02)	Cyclic, t#100ms	16#00C8	4	Keep last value		
3	Markers read	Read Coils (Function Code 01)	Cyclic, t#100ms	16#0190	32	Keep last value		
4	Markers write	Write Multiple Coils (Function Code 15)	Cyclic, t#100ms				16#01B0	8
5	Read Ext. Registers	Read Input Registers (Function Code 04)	Cyclic, t#100ms	16#03E8	20	Keep last value		
6	Write Ext. Registers	Write Multiple Registers (Function Code 16)	Cyclic, t#100ms				16#0410	20
7	Indicator status	Read Discrete Inputs (Function Code 02)	Cyclic, t#100ms	16#0440	15	Keep last value		
8	Control	Write Multiple Coils (Function Code 15)	Cyclic, t#100ms				16#03E8	6

In the lists below the addresses are appointed without the offset. If you use the above list, you can use the lists below as structures.

0) Read Indicators (dint)

Indicator		Address		
		Code	Address	Combined
1	Weight	3x	101	300101
2	Fast gross weight	3x	103	300103
3	Fast net weight	3x	105	300105
4	Display fast gross	3x	107	300107
5	Display fast net	3x	109	300109
6	Tare	3x	111	300111
7	Peak	3x	113	300113
8	Valley	3x	115	300115
9	Hold	3x	117	300117
10	Weight x10	3x	119	300119
11	Fast gross weight x10	3x	121	300121
12	Fast net weight x10	3x	123	300123
13	Display fast gross x10	3x	125	300125
14	Display fast net x10	3x	127	300127
15	Tare x10	3x	129	300129
16	Peak x10	3x	131	300131
17	Valley x10	3x	133	300133
18	Hold x10	3x	135	300135
19	Signal	3x	137	300137
20	Zero	3x	139	300139
21	Checked value gross*10	3x	141	300141
22	Checked value net*10	3x	143	300143

23	Checked value gross	3x	145	300145
24	1020 actual display value	3x	147	300147
25	Checked value net	3x	149	300149

1) Read Inputs (3 bits)

Inputs		**Address**		
		Code	Address	Combined
1	Start/stop program	1x	1	100001
2	Start sampling	1x	2	100002
3	Rejector sensor	1x	3	100003

2) Read Outputs (4 bits)

Outputs		**Address**		
		Code	Address	Combined
1	Rejector	1x	201	100201
2	Sampling busy	1x	202	100202
3	Transport belt	1x	203	100203
4	Alarm	1x	204	100204

3) Read Markers (32 bits)

Markers		**Address**		
		Code	Address	Combined
1	Check low	0x	401	000401
2	Check high	0x	402	000402
3	Check OK	0x	403	000403
4	Check ready	0x	404	000404
5	Sec alive bit	0x	405	000405
6	Sample busy	0x	406	000406
7	Check busy	0x	407	000407
8	Reset color	0x	408	000408
9	Result handled	0x	409	000409
10	Check done	0x	410	000410
11	Display hold	0x	411	000411
12	1020 online	0x	412	000412
13	Not used	0x	413	000413
14	Stop belts	0x	414	000414
15	Reset sub totals	0x	415	000415
16	Reset totals	0x	416	000416

17	IND error	0x	417	000417
18	Not used	0x	418	000418
19	Not used	0x	419	000419
20	Not used	0x	420	000420
21	Not used	0x	421	000421
22	Not used	0x	422	000422
23	Not used	0x	423	000423
24	Not used	0x	424	000424
25	Not used	0x	425	000425
26	Not used	0x	426	000426
27	Not used	0x	427	000427
28	Not used	0x	428	000428
29	Not used	0x	429	000429
30	Not used	0x	430	000430
31	Not used	0x	431	000431
32	Not used	0x	432	000432

4) Write Markers (8 bits)

Markers		**Address**		
		Code	Address	Combined
1	Start / Stop	0x	433	000433
2	Start sampling	0x	434	000434
3	Rejector sensor	0x	435	000435
4	Not used	0x	436	000436
5	Not used	0x	437	000437
6	Not used	0x	438	000438
7	Not used	0x	439	000439
8	Not used	0x	440	000440

5) Read Ext. Registers (dint)

Ext. Registers		Address		
		Code	Address	Combined
1	Net Checked value	3x	1001	301001
2	Net value	3x	1003	301003
3	Internal checked value	3x	1005	301005
4	DAC value	3x	1007	301007
5	Registration	3x	1009	301009
6	Custom code	3x	1011	301011
7	Not used	3x	1013	301013
8	Not used	3x	1015	301015
9	Not used	3x	1017	301017
10	Not used	3x	1019	301019

6) Write Ext. Registers (dint)

Ext. Registers		Address		
		Code	Address	Combined
21	Low level	4x	1041	401041
22	High level	4x	1043	401043
23	Preset tare	4x	1045	401045
24	Sample time	4x	1047	401047
25	Correction	4x	1049	401049
26	Check delay	4x	1041	401051
27	Belt speed	4x	1043	401053
28	Rejector delay	4x	1045	401055
29	Reject hold	4x	1047	401057
30	Code	4x	1049	401059

7) Read Indicator status (16 bits)

Indicator status		Address		
		Code	Address	Combined
1	Hardware overload	1x	1089	101089
2	Maximum load	1x	1090	101090
3	Stable weight	1x	1091	101091
4	Stable range	1x	1092	101092
5	Zero set	1x	1093	101093
6	Center of zero	1x	1094	101094
7	Zero range	1x	1095	101095

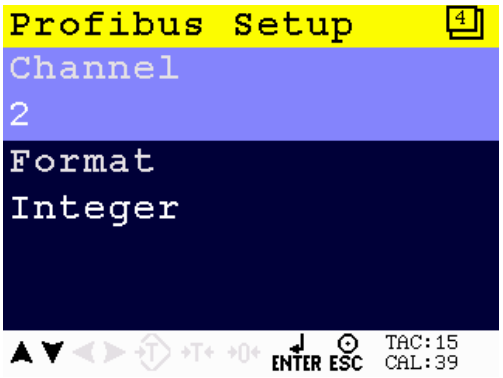
8	Zero track range	1x	1096	101096
9	Tare active	1x	1097	101097
10	Preset tare active	1x	1098	101098
11	New sample available	1x	1099	101099
12	Calibration invalid	1x	1100	101100
13	Calibration enabled	1x	1101	101101
14	Industrial mode	1x	1102	101102
15	Invalid weight	1x	1103	101103
16	Reserved	1x	1104	101104

8) Write Indicator control (6 bits)

Indicator control		**Address**		
		Code	Address	Combined
1	Zero reset	0x	1001	001001
2	Zero set	0x	1002	001002
3	Tare off	0x	1003	001003
4	Tare on	0x	1004	001004
5	Toggle tare	0x	1005	001005
6	Preset tare	0x	1006	001006

9.2 Profibus

First set up the Channel and Format in the Profibus Setup. Press Enter for 3 seconds. Press on System Setup and Port Setup, then press on Profibus Setup. Set up the Channel, Format and press “ESC”. Keep pressing on the “ESC” button to return to the live weight screen.



GSD file data structure

Download the 1020 controller GSD file (PTEC0E02.GSD) from the Penko website www.penko.com/Support/Software/.

Read data structure from the 1020:

Data type	Description	
Double word 32 bit signed integer/float	Read weight value	
Word 16 bit	Read indicator status	Bit 0 = Hardware overload
		Bit 1 = Maximum overload
		Bit 2 = Stable weight
		Bit 3 = Stable range
		Bit 4 = Zero set
		Bit 5 = Center of zero
		Bit 6 = Zero range
		Bit 7 = Zero track range
		Bit 8 = Tare active
		Bit 9 = Preset tare active
		Bit 10 = New sample available
		Bit 11 = Calibration invalid
		Bit 12 = Calibration enabled
		Bit 13 = Industrial mode
		Bit 14 = Invalid weight
		Bit 15 = Reserved

Byte 8 bit	Read command	Bit 0 = Zero reset
		Bit 1 = Zero set
		Bit 2 = Tare off
		Bit 3 = Tare on
		Bit 4 = Reserved
		Bit 5 = Freeze Weight value
		Bit 6 = Indicator channel 2^0
		Bit 7 = Indicator channel 2^1
Byte 8 bit	Read weight select register	Not used
Word 16 bit	Read inputs	Bit 0 = Input 1 Start/stop
		Bit 1 = Input 2 Start sampling
		Bit 2 = Input 3 Rejector sensor
		Bit 3 - 15 = Input 4 – 16 Not used
Word 16 bit	Read outputs	Bit 0 = Output 1 Rejector
		Bit 1 = Output 2 Sampling busy
		Bit 2 = Output 3 Transport belts
		Bit 3 = Output 4 Alarm
		Bit 4 – 15 = Output 5 – 16 Not used
Word 16 bit	Read markers 401 - 416	Bit 0 = Check low
		Bit 1 = Check high
		Bit 2 = Check OK
		Bit 3 = Check Ready
		Bit 4 = Sec alive bit
		Bit 5 = Sample busy
		Bit 6 = Check busy
		Bit 7 = Reset color
		Bit 8 = Result handled
		Bit 9 = Check done
		Bit 10 = Display hold
		Bit 11 = 1020 online
		Bit 12 = Not used
		Bit 13 = Stop belts
		Bit 14 = Reset sub totals
		Bit 15 = Reset totals
Word 16 bit	Read markers 417 - 432	Bit 0 = IND error
		Bit 1 -15 = Not used
Double word 32 bit signed integer	Read register 1	Net checked value (only active when program is started)
Double word 32 bit signed integer	Read register 2	Net value

Double word 32 bit signed integer	Read register 3	Internal check value
Double word 32 bit signed integer	Read register 4	DAC value

Write data structure to the 1020:

Data type	Description	
Byte 8 bit	Write command	Bit 0 = Zero reset
		Bit 1 = Zero set
		Bit 2 = Tare off
		Bit 3 = Tare on
		Bit 4 = Reserved
		Bit 5 = Freeze Weight value
		Bit 6 = Indicator channel 2^0
		Bit 7 = Indicator channel 2^1
Byte 8 bit	Write weight select register	Not used
Word 16 bit	Write markers 969 - 984	Bit 0 = Start / stop program
		Bit 1 = Start sampling
		Bit 2 = Reject sensor
		Bit 3 – 15 = Not used
Word 16 bit	Write markers 985 - 1000	Bit 0 – 15 = Not used
Double word 32 bit signed integer	Write register 85	Low level value from Profibus
Double word 32 bit signed integer	Write register 86	High level value from Profibus
Double word 32 bit signed integer	Write register 87	Preset tare value from Profibus
Double word 32 bit signed integer	Write register 88	Sample time value from Profibus

9.3 EtherNet IP

EDS data structure

Download the 1020 EDS file from the Penko website www.penko.com/Support/Software/.

Control in (884)

Read data structure from the 1020: In the example the instance 0x0374 (884) Control in is used.

Access	Name	Data type	Description
Get	Control In	STRUCT OF	
	Weigher	DINT WEIGHER	Display rate weigher data
		DINT GROSS	Fast Gross weight
		DINT NET	Fast Net weight
		DINT TARE	Active Tare weight
		DINT WEIGHERx10	Display rate weigher data x10
		DINT GROSSx10	Fast Gross weight x10
		DINT NETx10	Fast Net weight x10
		DINT TAREx10	Active Tare weight x10
		WORD FORMAT	Format bits, see Weigher-Format word
		WORD STATUS	Status bits, see Weigher-Status word
	Indicator	ARRAY[20] OF STRUCT OF INDICATOR	Read indicators, default start read at 1
	Register read	ARRAY OF DINT[10]	Registers [10], 1020 controller : Register 1 = Net checked value Register 2 = Net value Register 3 = Internal checked value Register 4 = DAC value Register 5 = Registration Register 6 = Alibi Nr Register 7 = Code Register 8 = Not used Register 9 = Not used Register 10 = Not used
	Markers Input	BYTE ARRAY[4]	Markers 4x8=32 default read at 401-432 Bit 0 = Check low Bit 1 = Check high Bit 2 = Check OK Bit 3 = Check ready Bit 4 = Sec alive bit

	Bit 5 = Sample busy Bit 6 = Check busy Bit 7 = Reset color Bit 8 = Result handled Bit 9 = Check done Bit 10 = Display hold Bit 11 = 1020 online Bit 12 = Not used Bit 13 = Stop belts Bit 14 = Reset sub totals Bit 15 = Reset totals Bit 16 = IND error Bit 17 = Not used Bit 18 = Not used Bit 19 = Not used Bit 20 = Not used Bit 21 = Not used Bit 22 = Not used Bit 23 = Not used Bit 24 = Not used Bit 25 = Not used Bit 26 = Not used Bit 27 = Not used Bit 28 = Not used Bit 29 = Not used Bit 30 = Not used Bit 31 = Not used

Control out (888)

Write data structure to the 1020: In the example the instance 0x0378 (888) Control out is used.

Access	Name	Data type	Description
Set	Control Out	STRUCT OF	
	Weigher Control	ARRAY OF BYTE[2]	Weigher control word, see also Weigher-Control word
	Reserved Control	ARRAY Of BYTE[2]	Set to 0x0000
	Register write	ARRAY OF DINT[10]	Registers [10], 1020 indicator : Register 11 = Low level Register 12 = High level Register 13 = Preset tare Register 14 = Sample time Register 15-20 = Not used
	Markers Output	BYTE ARRAY[4]	Markers 4x8=32 default write at 433-464 Bit 0 = Start/stop program Bit 1 = Start sampling Bit 2 = Reject sensor Bit 3 - 31 = Not used

Weigher-Status word

Bit #	Called	Definition
0	OVERLOAD	Hardware overload/underload detected on loadcell
1	MAXLOAD	Overload detected on loadcell
2	STABLE	Weigher signal is stable
3	STABLE RANGE	Weigher signal is in stable range
4	ZERO SET	Weigher zero is corrected
5	ZERO CENTER	Weigher in center of zero range
6	ZERO RANGE	Weigher is in zero range, zero is possible
7	ZERO TRACK	Weigher signal is in zero tracking range, zero tracking is possible
8	TARE	Weigher tare is active
9	PTARE	Weigher preset tare is active
10	SAMPLE	Used by internal process handling
11	BAD CAL	Calibration is bad, invalid, not available
12	CAL ENABLED	Calibration is enabled, used by internal process handling
13	INDUSTRIAL	If set weigher runs in industrial mode, if reset weigher runs certified operation mode
14	NOT LEVEL	Weigher system in blocking, warming up or scale is not level
15	RESERVED	Reserved mode always 0

Weigher-Control word

Bit #	Called	Definition
0	ZERO_RESET*	Reset the actual zero weight, condition only possible in noncertified mode
1	ZERO_SET*	Activate new zero weight, condition stable signal
2	TARE_OFF*	Switch actual tare weight off
3	TARE_ON*	Activate new tare weight, condition stable signal
4	TARE_TOGGLE*	Toggle the Tare weight on condition stable signal, off condition none
5-16	RESERVED	Reserved bits always 0

*Remark: action on rising edge of bit

Weigher-Format word

Bit number	Description
#15	Signed/unsigned
	0 = Unsigned
	1 = Signed
#14	Zero suppressing
	0 = Nonzero suppressing
	1 = Zero suppressing
#11 - #8	Display step size
	0000 = Step 1
	0001 = Step 2
	0010 = Step 5
	0011 = Step 10
	0100 = Step 20
	0101 = Step 50
	0110 = Step 100
	0111 = Step 200
	1000 = Step 500
	1001 = Step 1000
	1010 = Step 2000
	1011 = Step 5000
#2 - #0	Decimal point position
	000 = 000000
	001 = 00000.0
	010 = 0000.00
	011 = 000.000
	100 = 00.0000
	101 = 0.00000

9.4 Profinet

GSDML data structure

Download the 1020 GSDML file from the Penko website www.penko.com/Support/Software/.

Module	Data type	Provided data (channels)
Weigher Input Module	**Cyclic input data**	
	DInt	Net
	DInt	Gross
	DInt	Tare
	DInt	Preset Tare
	Byte	Status 0 = Weight is valid 1 = Stable weight 2 = Net weight 3 = Center of zero 4 = Zero is set 5 = Floating point 6 = Command is ready 7 = Command is in execution mode
	Byte	Decimal point position in non floating point mode
	Byte	Range, active multiple range/multi interval, 0 is none. i.e. 1 = e1, 2 = e2, etc
Remote Command Module	**Cyclic input data**	
	DInt	Result data
	Byte	Command Result Code
	Bool	Status 0 = Weight is valid 1 = Stable weight 2 = Net weight 3 = Center of zero 4 = Zero is set 5 = Floating point 6 = Command is ready 7 = Command is in execution mode
	Cyclic output data	
	DWord	Command
	DWord	Parameter
	DInt	Exchange

Inputs Outputs Markers Module	**Cyclic input data**	
	DWord	Read inputs 1 - 3: Bit 0 = Start/stop program Bit 1 = Start sampling Bit 2 = Rejector sensor Bit 3 – 32 = Not used
	DWord	Read outputs 1 - 4: Bit 0 = Rejector Bit 1 = Sampling busy Bit 2 = Transport belts Bit 3 = Alarm Bit 4 – 32 = Not used
	DWord	Read markers 401 – 432: Bit 0 = Check low Bit 1 = Check high Bit 2 = Check OK Bit 3 = Check ready Bit 4 = Sec alive bit Bit 5 = Sample busy Bit 6 = Check busy Bit 7 = Reset color Bit 8 = Result handled Bit 9 = Check done Bit 10 = Display hold Bit 11 = 1020 online Bit 12 = Not used Bit 13 = Stop belts Bit 14 = Reset sub totals Bit 15 = Reset totals Bit 16 = IND error Bit 17 = Not used Bit 18 = Not used Bit 19 = Not used Bit 20 = Not used Bit 21 = Not used Bit 22 = Not used Bit 23 = Not used Bit 24 = Not used Bit 25 = Not used Bit 26 = Not used Bit 27 = Not used Bit 28 = Not used Bit 29 = Not used

		Bit 30 = Not used Bit 31 = Not used
	Cyclic output data	
	DWord	Write markers 969 – 1000: Bit 0 = Start/stop program Bit 1 = Start sampling Bit 2 = Rejector sensor Bit 3 - 31 = Not used
Diagnostics Module	**Cyclic input data**	
	DInt	Slave sequence counter, integrated Profinet ASIC
	DInt	Master sequence counter, integrated Main CPU

Recipe read and write

The recipe values can be read or written using the Cyclic output data parameters.

Cyclic output data

DWord	Command
DWord	Parameter
DInt	Exchange

The result data can be read using the Cyclic input data.

Cyclic input data	
DInt	Result data
Byte	Command Result Code

Read recipe

Recipe		Cyclic output data			Cyclic input data	
Nr	Description	Command	Parameter	Exchange	Result data	Command result code
1	Low level	10	0	Not used	Low level value	See list below
2	High level	10	1	Not used	High level value	See list below
3	Preset tare	10	2	Not used	Preset tare value	See list below
4	Sample time	10	3	Not used	Sample time value	See list below
5	Correction	10	4	Not used	Correction value	See list below
6	Check delay	10	5	Not used	Check delay value	See list below
7	Belt speed	10	6	Not used	Belt speed value	See list below
8	Rejector delay	10	7	Not used	Rejector delay value	See list below
9	Reject hold	10	8	Not used	Reject hold value	See list below
10	Code	10	9	Not used	Code value	See list below

Write recipe

Recipe		Cyclic output data			Cyclic input data	
Nr	Description	Command	Parameter	Exchange	Result data	Command result code
1	Low level	11	0	Low level value	Low level value	See list below
2	High level	11	1	High level value	High level value	See list below
3	Preset tare	11	2	Preset tare value	Preset tare value	See list below
4	Sample time	11	3	Sample time value	Sample time value	See list below
5	Correction	11	4	Correction value	Correction value	See list below
6	Check delay	11	5	Check delay value	Check delay value	See list below
7	Belt speed	11	6	Belt speed value	Belt speed value	See list below
8	Rejector delay	11	7	Rejector delay value	Rejector delay value	See list below
9	Reject hold	11	8	Reject hold value	Reject hold value	See list below
10	Code	11	9	Code value	Code value	See list below

Register read

The register values can be read using the Cyclic output data parameters.

Cyclic output data	
DWord	Command
DWord	Parameter
DInt	Exchange

The result data can be read using the Cyclic input data.

Cyclic input data	
DInt	Result data
Byte	Command Result Code

Read registers

Recipe		Cyclic output data			Cyclic input data	
Nr	Description	Command	Parameter	Exchange	Result data	Command result code
1	Net checked value	5	0	Not used	Net checked value	See list below
2	Net value	5	1	Not used	Net value	See list below
3	Internal checked value *10	5	2	Not used	Internal checked value *10	See list below
4	DAC value	5	3	Not used	DAC value	See list below
5	Registration	5	4	Not used	Registration	See list below
6	Alibi number	5	5	Not used	Alibi number	See list below
7	Custom code	5	6	Not used	Custom code	See list below

Command result codes

When you try to read or write a recipe value, you will receive a Command result

ID	Code	Description
0	RPC_SUCCES	Command executed success
1	RPC_EXECUTING	Command is executing
2	RPC_UNKNOWN_COMMAND	Unknown Penko Profinet command
3	RPC_UNKNOWN_FUNCTION	Unknown function
4	RPC_NOTIDLE	Busy executing a command
5	RPC_FAILED	Command executing failed
6	RPC_ERROR	Command error
7	RPC_NOT_ALLOWED	Command executing not allowed
8-127	RESERVED	Reserved error codes
128	RPC_PARAMETER_ERROR	Invalid parameter set
129	RPC_NOTSTABLE	Weight not stable
130	RPC_NEGATIVE	Weight negative
131	RPC_NO_TARE	Tare not set
132	RPC_OUTOFRANGE	Weight out of range
134	RPC_NOT_STABLE	Weigher not stable
135	RPC_ABOVE_MAXLOAD	Weight is above maxload
136	RPC_BELOW_ZERO	Weigher below zero
137	RPC_NOT_IN_ZERO_RANGE	Weigher not in zero range
138	RPC_ARITMIC_OVERFLOW	Aritmic overflow
139	RPC_ADC_OVERFLOW	Overload by ADC conversion
140	RPC_ADC_UNDERFLOW	Underload by ADC conversion
141	RPC_GAIN_NEGATIVE	Weight should increase and not decrease
142	RPC_GAIN_OVERFLOW	Weight to low, value between zero and end weight required
143	RPC_ACCESSDENIED	Command executing denied first enter TAC or CAL code

About PENKO

Our design expertise include systems for manufacturing plants, bulk weighing, check weighing, force measuring and process control. For over 35 years, PENKO Engineering B.V. has been at the forefront of development and production of high-accuracy, high-speed weighing systems and our solutions continue to help cut costs, increase ROI and drive profits for some of the largest global brands, such as Cargill, Sara Lee, Heinz, Kraft Foods and Unilever to name but a few.

Whether you are looking for a simple stand-alone weighing system or a high-speed weighing and dosing controller for a complex automated production line, PENKO has a comprehensive range of standard solutions you can rely on.

Certifications

PENKO sets high standards for its products and product performance which are tested, certified and approved by independent expert and government organizations to ensure they meet – and even – exceed metrology industry guidelines. A library of testing certificates is available for reference on:

http://penko.com/nl/publications_certificates.html

PENKO Professional Services

PENKO is committed to ensuring every system is installed, tested, programmed, commissioned and operational to client specifications. Our engineers, at our weighing center in Ede, Netherlands, as well as our distributors around the world, strive to solve most weighing-system issues within the same day. On a monthly basis PENKO offers free training classes to anyone interested in exploring modern, high-speed weighing instruments and solutions. A schedule of training sessions is found on: **www.penko.com/training**

PENKO Distributor

A complete overview you will find on: www.penko.com/Find-A-Dealer

PENKO Engineering B.V. ▪ Schutterweg 35, NL 6718XC Ede ▪ Tel +31 (0) 318525630 ▪ info@penko.com
Web ▪ www.penko.com ▪