

PENKO Engineering B.V.

Your Partner for Fully Engineered Factory Solutions



Protocol description:
PENKO Modbus



PENKO

an ETC Company

PENKO Modbus protocol

Table of Contents

Introduction.....	5
1 Basics	6
1.1 Devices	6
1.2 Data types.....	6
1.3 Address codes.....	6
2 Addresses	7
2.1 Inputs, outputs and markers	7
2.2 Indicators and registers	7
2.3 Weigher control.....	8
2.4 Weigher status	10
3 Indicator values	12
4 Examples.....	14
4.1 Read inputs/outputs.....	14
4.2 Read/write markers.....	16
4.3 Read indicator values	18
4.4 Read/write register values	20
4.5 Weigher control.....	22
4.6 Weigher status	24
4.7 Tag list.....	26
5 Advanced features.....	27
5.1 Function codes	29
5.2 Error codes	30
5.3 Calibration functions	32
5.3.1 CAL_ZERO	32
5.3.2 CAL_SPAN	32
5.3.3 CAL_MV	33
5.3.4 CAL_DEADLOAD.....	34
5.3.5 CAL_INSERT	34
5.3.6 CAL_POINT.....	35
5.3.7 CAL_DELETE.....	35

PENKO Modbus protocol

5.3.8	CAL_GEOGRAPHIC_ORIGIN_SET	36
5.3.9	CAL_GEOGRAPHIC_ORIGIN_GET.....	36
5.3.10	CAL_GEOGRAPHIC_LOCAL_SET	37
5.3.11	CAL_GEOGRAPHIC_LOCAL_GET	37
5.4	Indicator functions	38
5.4.1	IND_MAXLOAD_SET	38
5.4.2	IND_MAXLOAD_GET.....	38
5.5	PDI functions	39
5.5.1	PDI_PATH_SET.....	39
5.5.2	PDI_PROPERTY_SET.....	41
5.5.3	PDI_PROPERTY_GET	41
5.6	Printer functions.....	43
5.6.1	PRINT	43
5.6.2	PRINT_SUBTOTAL	44
5.6.3	PRINT_TOTAL.....	45
5.6.4	PRINT_DAYTOTAL.....	46
5.6.5	PRINT_BATCHTOTAL.....	47
5.6.6	PRINT_LAYOUT	48
5.6.7	PRINT_ALIBI.....	49
5.6.8	PRINT_ALIBIMEMORY.....	50
5.6.9	PRINT_EVENTMEMORY	51
5.7	Total functions.....	52
5.7.1	TOTAL_TOTALIZE	52
5.7.2	TOTAL_SUBTOTAL	53
5.7.3	TOTAL_TOTAL.....	54
5.7.4	TOTAL_DAYTOTAL	55
5.7.5	TOTAL_BATCHTOTAL.....	56
5.8	Controller functions.....	57
5.8.1	RFN_PROCESS_RECIPES_GET	57
5.8.2	RFN_PROCESS_RECIPES_SET.....	57
5.8.3	RFN_PROCESS_CONFIG_GET.....	58

PENKO Modbus protocol

5.8.4 RFN_PROCESS_CONFIG_SET 58

5.8.5 RFN_PROCESS_DATA..... 59



PENKO Modbus protocol

Introduction

The PENKO Modbus protocol supports the following basic features:

- Read weigher data and control weighers
- Read inputs and outputs
- Read/write markers
- Read/write registers and levels

Advanced features are available by using the register command mode, described in chapter 5.

TCP/IP: Modbus TCP/IP is supported. Use port 502 for a TCP/IP sessions.

Serial: Modbus RTU and ASCII are supported.

Only one connection at the time can be active. Modbus TCP/IP is only active when both RS232 and RS422 are **not** set to Modbus.



PENKO Modbus protocol

1 Basics

The PENKO devices that support the protocol and the used data types and address codes.

1.1 Devices

The following PENKO devices support the Modbus protocol:

Device Series	Modbus support
SGM7xx SGM8xx	Yes, starting at version v1.3.3.9.0.7
1020 hardware version 1*	No Modbus support
1020 hardware version 2*	Yes, starting at version v1.3.6.9.0.1
Flex 2100	Yes, starting at version v1.8.4.9.0.1
Flex	Yes, starting at version v1.8.4.9.0.1
Flex Multichannel	Yes, starting at version v1.8.4.9.0.1

* See Main Menu - Info - Hardware Version

Flex multichannel supports Weigher 1, 2, 3 and 4. All other instruments only support Weigher 1.

1.2 Data types

Data type	Description
Float	Double Word floating point IEEE 754, 4 bytes
Long	Double Word signed, 4 bytes
Word	Unsigned Word, 2 bytes

1.3 Address codes

The following address codes are supported:

Modbus name	Address code	Usage
Read coils	0x	<i>Read internal markers</i>
Read discrete inputs	1x	<i>Read physical inputs</i> <i>Read physical outputs</i> <i>Weigher status</i>
Read input register	3x	<i>Read indicators as Float</i> <i>Read indicators as Long</i> <i>Read extended registers</i>
Write single coil	0x	<i>Write internal markers</i> <i>Weigher control</i>
Write single register	4x	<i>Write extended register</i>
Write multiple coils	0x	<i>Write internal markers</i>
Write multiple registers	4x	<i>Write extended registers</i>

PENKO Modbus protocol

2 Addresses

The following addresses are used:

2.1 Inputs, outputs and markers

Inputs/Outputs/Markers	Data type	Address code	Address	Combined
Inputs 1 - 200	Bit	1x	1 - 200	100001 to 100200
Outputs 1 - 200	Bit	1x	201 - 400	100201 to 100400
Markers 1 - 600	Bit	0x	401 - 1000	000401 to 001000

Note: The outputs in PENKO devices cannot be written through Modbus.

2.2 Indicators and registers

Indicators/Registers	Data type	Address code	Address	Combined
Indicator 1 - 50*	Float	3x	1 - 99	300001 to 300099
Indicator 1 - 50*	Long	3x	101 - 199	300101 to 300199
Ext Registers 1 - 900**	Long**	3x	1001 - 2799	301001 to 302799
		4x		401001 to 402799

Note: Indicators and extended registers are 32 bit numbers. Modbus registers are 16 bit, so each indicator or extended register is accessed through 2 consecutive registers. Even memory locations are the highest 16 bits, odd memory locations are the lowest 16 bits. The Modbus default is first word low. Therefore OPC servers offer settings for word swapping.

* The 1020 and SGM7xx/8xx series have 19 indicator values (1020 FMD = 13 values). The Flex series have configurable indicators. Consult the Flex manual on how to set the indicators.

** The Flex series have 900 ext. registers and support the use of floating point registers starting at a configurable position. The SGM7xx/8xx and 1020 have 150 ext. registers and support floating point registers above 100.

PENKO Modbus protocol

2.3 Weigher control

Weigher 1 control	Data type	Address code	Address	Combined
Zero reset*	Bit	0x	1001	001001
Zero set*	Bit	0x	1002	001002
Tare reset*	Bit	0x	1003	001003
Tare set*	Bit	0x	1004	001004
Toggle tare*	Bit	0x	1005	001005
Activate preset tare*	Bit	0x	1006	001006
Reserved	Bit	0x	1007	001007
Reserved	Bit	0x	1008	001008

Weigher 2 control	Data type	Address code	Address	Combined
Zero reset*	Bit	0x	1009	001009
Zero set*	Bit	0x	1010	001010
Tare reset*	Bit	0x	1011	001011
Tare set*	Bit	0x	1012	001012
Toggle tare*	Bit	0x	1013	001013
Activate preset tare*	Bit	0x	1014	001014
Reserved	Bit	0x	1015	001015
Reserved	Bit	0x	1016	001016

Weigher 3 control	Data type	Address code	Address	Combined
Zero reset*	Bit	0x	1017	001017
Zero set*	Bit	0x	1018	001018
Tare reset*	Bit	0x	1019	001019
Tare set*	Bit	0x	1020	001020
Toggle tare*	Bit	0x	1021	001021
Activate preset tare*	Bit	0x	1022	001022
Reserved	Bit	0x	1023	001023
Reserved	Bit	0x	1024	001024

Weigher 4 control	Data type	Address code	Address	Combined
Zero reset*	Bit	0x	1025	001025
Zero set*	Bit	0x	1026	001026
Tare reset*	Bit	0x	1027	001027
Tare set*	Bit	0x	1028	001028
Toggle tare*	Bit	0x	1029	001029
Activate preset tare*	Bit	0x	1030	001030
Reserved	Bit	0x	1031	001031
Reserved	Bit	0x	1032	001032

* Action on rising edge of bit.

PENKO Modbus protocol

The 1020 FMD has different control bits:

1020 FMD control	Data type	Address code	Address	Combined
Zero reset*	Bit	0x	1001	001001
Zero set*	Bit	0x	1002	001002
Peak reset*	Bit	0x	1003	001003
Valley reset*	Bit	0x	1004	001004
Hold reset*	Bit	0x	1005	001005
T.I.R. reset*	Bit	0x	1006	001006
Reserved	Bit	0x	1007	001007
Reserved	Bit	0x	1008	001008

* Action on rising edge of bit.

PENKO Modbus protocol

2.4 Weigher status

Weigher 1,2,3,4 status	Data type	Address code	Weigher	Address	Combined
Hardware overload / underload detected on loadcell	Bit	1x	1	1089	101089
			2	1105	101105
			3	1121	101121
			4	1137	101137
Overload detected on loadcell	Bit	1x	1	1090	101090
			2	1106	101106
			3	1122	101122
			4	1138	101138
Weigher signal is stable	Bit	1x	1	1091	101091
			2	1107	101107
			3	1123	101123
			4	1139	101139
Weigher signal is in stable range	Bit	1x	1	1092	101092
			2	1108	101108
			3	1124	101124
			4	1140	101140
Weigher zero is corrected	Bit	1x	1	1093	101093
			2	1109	101109
			3	1125	101125
			4	1141	101141
Weigher in center of zero range	Bit	1x	1	1094	101094
			2	1110	101110
			3	1126	101126
			4	1142	101142
Weigher is in zero range, zero is possible	Bit	1x	1	1095	101095
			2	1111	101111
			3	1127	101127
			4	1143	101143
Weigher signal is in zero tracking range, zero tracking is possible	Bit	1x	1	1096	101096
			2	1112	101112
			3	1128	101128
			4	1144	101144
Weigher tare is active	Bit	1x	1	1097	101097
			2	1113	101113
			3	1129	101129
			4	1145	101145
Weigher preset tare is active	Bit	1x	1	1098	101098
			2	1114	101114
			3	1130	101130
			4	1146	101146
Used by internal process handling	Bit	1x	1	1099	101099
			2	1115	101115
			3	1131	101131
			4	1147	101147

PENKO Modbus protocol

Weigher 1,2,3,4 status	Data type	Address code	Weigher	Address	Combined
Calibration is bad, invalid, not available	Bit	1x	1	1100	101100
			2	1116	101116
			3	1132	101132
			4	1148	101148
Calibration is enabled, used by internal process handling	Bit	1x	1	1101	101101
			2	1117	101117
			3	1133	101133
			4	1149	101149
If set weigher runs in industrial mode, if reset weigher runs certified operation mode	Bit	1x	1	1102	101102
			2	1118	101118
			3	1134	101134
			4	1150	101150
Weigher system in blocking, warming up or scale is not level	Bit	1x	1	1103	101103
			2	1119	101119
			3	1135	101135
			4	1151	101151
Register command mode active	Bit	1x	1	1104	101104
			2	1120	101120
			3	1136	101136
			4	1152	101152

The 1020 FMD has different status bits:

1020 FMD status	Data type	Address code	Address	Combined
Hardware overload / underload detected on loadcell	Bit	1x	1089	101089
Overload detected on loadcell	Bit	1x	1090	101090
Force signal is stable	Bit	1x	1091	101091
Force signal is in stable range	Bit	1x	1092	101092
Force zero is corrected	Bit	1x	1093	101093
Used by internal process handling	Bit	1x	1099	101099
Calibration is bad, invalid, not available	Bit	1x	1100	101100
Calibration is enabled, used by internal process handling	Bit	1x	1101	101101
Indicator runs in industrial mode	Bit	1x	1102	101102
Invalid force	Bit	1x	1103	101103
Register command mode active	Bit	1x	1104	101104

PENKO Modbus protocol

3 Indicator values

The indicator values are available as float and long values. To gain the long values, add an offset of 100 to the address of the float value.

Note: This list is applicable for the 1020 and SGM7xx/8xx series. The Flex series has configurable indicators. Consult the Flex manual on how to set the indicators.

Indicator	Name	Description	Address code	Address	Combined Float	Long
1	WEIGHT	multi range net weigher value	3x	1	300001	300101
2	FAST GROSS	unfiltered gross weigher value	3x	3	300003	300103
3	FAST NET	unfiltered net weigher value	3x	5	300005	300105
4	DISPLAY GROSS	filtered gross weigher value	3x	7	300007	300107
5	DISPLAY NET	filtered net weigher value	3x	9	300009	300109
6	TARE	tare value	3x	11	300011	300111
7	PEAK	highest reached weigher value	3x	13	300013	300113
8	VALLEY	lowest reached weigher value	3x	15	300015	300115
9	HOLD*	stored weigher value	3x	17	300017	300117
10	WEIGHTx10	multi range net weigher value shown with extra decimal	3x	19	300019	300119
11	FAST GROSSx10	unfiltered gross weigher value shown with extra decimal	3x	21	300021	300121
12	FAST NETx10	unfiltered net weigher value shown with extra decimal	3x	23	300023	300123
13	DISPLAY GROSSx10	filtered gross weigher value shown with extra decimal	3x	25	300025	300125
14	DISPLAY NETx10	filtered net weigher value shown with extra decimal	3x	27	300027	300127
15	TAREx10	tare value shown with extra decimal	3x	29	300029	300129
16	PEAKx10	highest reached weigher value shown with extra decimal	3x	31	300031	300131
17	VALLEYx10	lowest reached weigher value shown with extra decimal	3x	33	300033	300133
18	HOLDx10*	stored weigher value shown with extra decimal	3x	35	300035	300135
19	SIGNAL	direct mV value from the load cell(s)	3x	37	300037	300137

*Hold and Holdx10 are added in the 1020 firmware version V1.5.1.9.0.6 and above. And in the SGM7xx/SGM8xx series firmware version V1.6.1.9.0.3 and above.

PENKO Modbus protocol

The 1020 FMD has different indicator values:

Indicator	Name	Description	Address code	Address	Combined Float	Long
1	FAST TRACKING	unfiltered tracking value	3x	1	300001	300101
2	TRACKING	filtered tracking value	3x	3	300003	300103
3	PEAK	highest reached weigher value	3x	5	300005	300105
4	VALLEY	lowest reached weigher value	3x	7	300007	300107
5	HOLD	hold value - stored with zero button in hold mode, or with hold input	3x	9	300009	300109
6	T.I.R.	total Indicated Readout value - the difference between the peak hold and valley hold value	3x	11	300011	300111
7	FAST TRACKING x10	fast tracking shown with extra decimal	3x	13	300013	300113
8	TRACKING x10	tracking shown with extra decimal	3x	15	300015	300115
9	PEAKx10	peak shown with extra decimal	3x	17	300017	300117
10	VALLEYx10	valley shown with extra decimal	3x	19	300019	300119
11	HOLDx10	hold shown with extra decimal	3x	21	300021	300121
12	T.I.R.x10	T.I.R. shown with extra decimal	3x	23	300023	300123
13	SAMPLE	direct ADC value from the load cell(s)	3x	25	300025	300125

PENKO Modbus protocol

4 Examples

The following examples show the use of the Modbus commands using *OPC KEPServerEX* and *Weintek EasyBuilder Pro*.

4.1 Read inputs/outputs

Read the status of input 1:

KepServer

The used format is 1xxxxx

Input # 1 = address 100001

Input # 2 = address 100002 etc.

Address: 100001

Data type: Boolean

The screenshot shows the configuration window for a Modbus input in KepServer. It is divided into two main sections: 'Identification' and 'Data properties'.
In the 'Identification' section, there are three fields: 'Name' with the value 'Input 1', 'Address' with the value '100001', and 'Description' with the value 'Read input 1'. To the right of these fields are several icons for help, save, and delete.
In the 'Data properties' section, there are three settings: 'Data type' is set to 'Boolean', 'Client access' is set to 'Read Only', and 'Scan rate' is set to '100 milliseconds'.

EasyBuilder

The used address code is 1x

Input # 1 = address 1

Input # 2 = address 2 etc.

The screenshot shows the configuration window for a Modbus input in EasyBuilder. It is titled 'Read address'.
There are two main settings: 'PLC name' is set to 'MODBUS TCP/IP' and 'Address' is set to '1x'. A 'Setting...' button is located to the right of the PLC name dropdown.
Below these settings is a checkbox labeled 'Invert signal', which is currently unchecked.

PENKO Modbus protocol

Read the status of output 1:

KepServer

The used format is 1xxxxx

Output # 1 = address 100201

Output # 2 = address 100202 etc.

Address: 100201

Data type: Boolean

The screenshot shows the 'Identification' section with the following fields: Name: Output 1, Address: 100201, and Description: Read output 1. The 'Data properties' section includes: Data type: Boolean, Client access: Read Only, and Scan rate: 100 milliseconds. There are navigation buttons on the right side of the window.

EasyBuilder

The used address code is 1x

Output # 1 = address 201

Output # 2 = address 202 etc.

The screenshot shows the 'Read address' section with the following fields: PLC name: MODBUS TCP/IP, Address: 1x, and a value field containing 201. There is an 'Invert signal' checkbox and a 'Setting...' button.

PENKO Modbus protocol

4.2 Read/write markers

Read/write marker 1:

KepServer

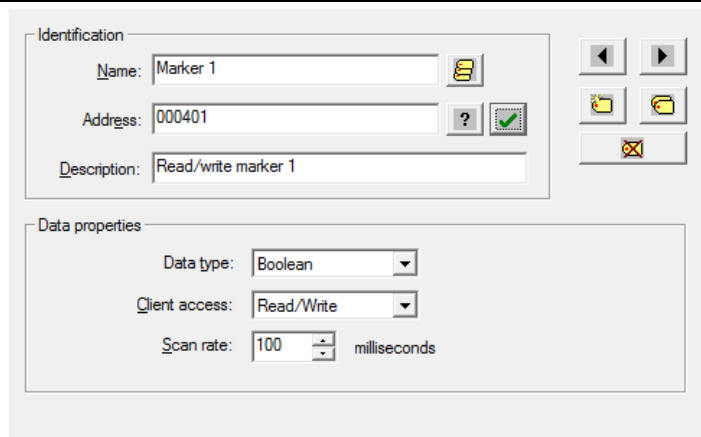
The used format is 0xxxxx

Marker # 1 = address 000401

Marker # 2 = address 000402 etc.

Address: 000401

Data type: Boolean



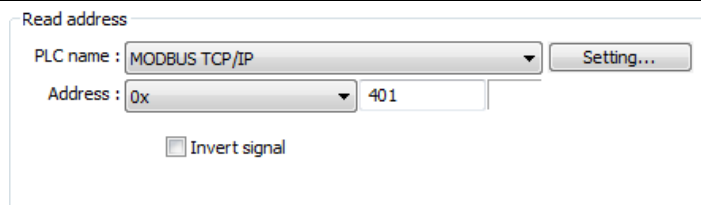
The screenshot shows the 'Identification' and 'Data properties' sections of the KepServer configuration window. In the 'Identification' section, the 'Name' is 'Marker 1', the 'Address' is '000401', and the 'Description' is 'Read/write marker 1'. In the 'Data properties' section, the 'Data type' is set to 'Boolean', the 'Client access' is 'Read/Write', and the 'Scan rate' is '100 milliseconds'.

EasyBuilder

The used address code is 0x

Marker # 1 = address 401

Marker # 2 = address 402 etc.



The screenshot shows the 'Read address' configuration window in EasyBuilder. The 'PLC name' is set to 'MODBUS TCP/IP'. The 'Address' is set to '0x' and '401'. There is an 'Invert signal' checkbox which is currently unchecked.

PENKO Modbus protocol

Read/write marker 40:

KepServer

The used format is 0xxxxx

Marker # 40 = address 000440

Marker # 41 = address 000441 etc.

Address: 000440

Data type: Boolean

The screenshot shows the configuration window for a marker in KepServer. It is divided into two main sections: 'Identification' and 'Data properties'.
In the 'Identification' section, there are three input fields: 'Name' with the value 'Marker 40', 'Address' with the value '000440', and 'Description' with the value 'Read/write marker 40'. To the right of these fields are several icons for file operations and a refresh button.
In the 'Data properties' section, there are three settings: 'Data type' is set to 'Boolean', 'Client access' is set to 'Read/Write', and 'Scan rate' is set to '100 milliseconds'.

EasyBuilder

The used address code is 0x

Marker # 40 = address 440

Marker # 41 = address 441 etc.

The screenshot shows the configuration window for a marker in EasyBuilder. It is titled 'Read address'.
There are two main settings: 'PLC name' is set to 'MODBUS TCP/IP' and 'Address' is set to '0x 440'. A 'Setting...' button is located to the right of the PLC name dropdown.
Below these settings is a checkbox labeled 'Invert signal' which is currently unchecked.

PENKO Modbus protocol

4.3 Read indicator values

Read indicator 1 as Float:

KepServer

The used format is 3xxxxx

Indicator # 1 = address 300001

Indicator # 2 = address 300003 etc.¹

Address: 300001

Data type: Float

The screenshot shows the configuration window for a Modbus indicator in KepServer. It is divided into two main sections: 'Identification' and 'Data properties'.
In the 'Identification' section, there are three input fields: 'Name' with the value 'Indicator 1 (Float)', 'Address' with the value '300001', and 'Description' with the value 'Read indicator 1 as Float'. To the right of these fields are several control buttons, including navigation arrows, a help icon, a refresh icon, and a close icon.
In the 'Data properties' section, there are three settings: 'Data type' is set to 'Float', 'Client access' is set to 'Read Only', and 'Scan rate' is set to '100 milliseconds'.

EasyBuilder

The used address code is 3x

Indicator # 1 = address 1

Indicator # 2 = address 3 etc.

The screenshot shows the configuration window for a Modbus indicator in EasyBuilder. It is divided into two main sections: 'Read address' and 'Display'.
In the 'Read address' section, there are two input fields: 'PLC name' with the value 'MODBUS TCP/IP' and 'Address' with the value '3x_Double'. To the right of the 'PLC name' field is a 'Setting...' button. The 'Address' field is followed by a small input box containing the value '1'.
In the 'Display' section, there are two settings: 'Data format' is set to '32-bit Float' and there is a checked checkbox labeled 'Mask'.

¹ PENKO indicators and registers are 32-bit values where Modbus register are 16-bit values and must therefore be accessed as a signed 32-bit data type. Also see chapter 2.2

PENKO Modbus protocol

Read indicators 2 as Long:

KepServer

The used format is 3xxxxx

Indicator # 1 = address 300101

Indicator # 2 = address 300103 etc.²

Address: 300103

Data type: Long (signed 32bit)

The screenshot shows the configuration window for a Modbus indicator in KepServer. It is divided into two main sections: 'Identification' and 'Data properties'.
In the 'Identification' section, the 'Name' field is set to 'Indicator 2 (Long)', the 'Address' is '300103', and the 'Description' is 'Read indicator 2 as Long'.
In the 'Data properties' section, the 'Data type' is set to 'Long', 'Client access' is 'Read Only', and the 'Scan rate' is '100 milliseconds'.
Navigation buttons (back, forward, search, help, refresh) are visible on the right side of the window.

EasyBuilder

The used address code is 3x

Indicator # 1 = address 101

Indicator # 2 = address 103

The screenshot shows the configuration window for a Modbus read address in EasyBuilder. The 'Read address' section includes a 'PLC name' dropdown set to 'MODBUS TCP/IP' and a 'Setting...' button. The 'Address' field is set to '3x_Double' and '103'.

The screenshot shows the 'Display' section of the configuration window. The 'Data format' dropdown is set to '32-bit Signed', and the 'Mask' checkbox is checked.

² PENKO indicators and registers are 32-bit values where Modbus register are 16-bit values and must therefore be accessed as a signed 32-bit data type. Also see chapter 2.2

PENKO Modbus protocol

4.4 Read/write register values

Read extended register 1:

KepServer

The used format is 3xxxxx

Register # 1 = address 301001

Register # 2 = address 301003 etc. ³

Address: 301001

Data type: Long (signed 32bit)

The screenshot shows the 'Identification' and 'Data properties' sections of the KepServer configuration. In the 'Identification' section, the 'Name' is 'Ext Register 1 read', the 'Address' is '301001', and the 'Description' is 'Read extended register 1'. In the 'Data properties' section, the 'Data type' is set to 'Long', 'Client access' is 'Read Only', and the 'Scan rate' is '100 milliseconds'.

EasyBuilder

The used address code is 3x

Register # 1 = address 1001

Register # 2 = address 1003

The screenshot shows the 'Read address' and 'Display' sections of the EasyBuilder configuration. In the 'Read address' section, the 'PLC name' is 'MODBUS TCP/IP' and the 'Address' is '3x_Double' with a value of '1001'. In the 'Display' section, the 'Data format' is '32-bit Signed' and the 'Mask' checkbox is checked.

³ PENKO indicators and registers are 32-bit values where Modbus register are 16-bit values and must therefore be accessed as a signed 32-bit data type. Also see chapter 2.2.

PENKO Modbus protocol

Write extended register 2:

KepServer

The used format is 4xxxxx

Register # 1 = address 401001

Register # 2 = address 401003 etc.⁴

Address: 401003

Data type: Long (signed 32bit)

The screenshot shows the 'KepServer' configuration window. It has two main sections: 'Identification' and 'Data properties'. In the 'Identification' section, the 'Name' field contains 'Ext Register 2 write', the 'Address' field contains '401003', and the 'Description' field contains 'Write extended register 2'. In the 'Data properties' section, the 'Data type' is set to 'Long', 'Client access' is set to 'Read/Write', and the 'Scan rate' is set to '100 milliseconds'. There are navigation buttons on the right side of the window.

EasyBuilder

The used address code is 4x

Register # 1 = address 1001

Register # 2 = address 1003

The screenshot shows the 'EasyBuilder' configuration window. It has a 'Read address' section with a 'PLC name' dropdown set to 'MODBUS TCP/IP' and a 'Setting...' button. Below it, the 'Address' field is set to '4x_Double' and '1003'. There is also a 'Display' section with a 'Data format' dropdown set to '32-bit Signed' and a 'Mask' checkbox.

⁴ PENKO indicators and registers are 32-bit values where Modbus register are 16-bit values and must therefore be accessed as a signed 32-bit data type. Also see chapter 2.2

PENKO Modbus protocol

4.5 Weigher control

Set zero:

KepServer

The used format is 0xxxxx

Reset zero = address 001001

Set zero = address 001002

See page 4

The screenshot shows the configuration window for a device named 'Set zero' at address 001002. The 'Identification' section includes fields for Name, Address, and Description. The 'Data properties' section shows the data type is Boolean, client access is Read/Write, and the scan rate is 100 milliseconds. Navigation buttons are visible on the right side.

Address: 001002

Data type: Boolean

EasyBuilder

The used address code is 0x

The screenshot shows the configuration window for a device named 'MODBUS TCP/IP' at address 0x1002. The 'Write address' section includes fields for PLC name and Address. A 'Setting...' button is also present.

Reset zero = address 1001

Set zero = address 1002

PENKO Modbus protocol

Set tare:

KepServer

The used format is 0xxxxx

Reset tare = address 001003

Set tare = address 001004

See page 4

The screenshot shows the configuration window for a device in KepServer. It is divided into two main sections: 'Identification' and 'Data properties'.
In the 'Identification' section, there are three fields: 'Name' with the value 'Set tare', 'Address' with the value '001004', and 'Description' with the value 'Weigher control set tare'. To the right of these fields are several icons: a left arrow, a right arrow, a yellow folder icon, a yellow document icon, and a red 'X' icon.
In the 'Data properties' section, there are three settings: 'Data type' is set to 'Boolean', 'Client access' is set to 'Read/Write', and 'Scan rate' is set to '100' milliseconds.

Address: 001004

Data type: Boolean

EasyBuilder

The used address code is 0x

The screenshot shows the 'Write address' configuration window in EasyBuilder. It contains two rows of settings. The first row has a 'PLC name' dropdown menu set to 'MODBUS TCP/IP' and a 'Setting...' button. The second row has an 'Address' dropdown menu set to '0x' and a text input field containing the value '1004'.

Reset tare = address 1003

Set tare = address 1004

PENKO Modbus protocol

4.6 Weigher status

Check if weigher is stable:

KepServer

The used format is 1xxxxx

Stable = address 101091

See page 4

Address: 101091

Data type: Boolean

The screenshot shows the configuration window for a Modbus point in KepServer. It is divided into two main sections: 'Identification' and 'Data properties'.
In the 'Identification' section, there are three input fields: 'Name' with the value 'Weigher Status Stable', 'Address' with the value '101091', and 'Description' with the value 'Check if weigher is stable'. To the right of these fields are several icons for file operations and help.
In the 'Data properties' section, there are three settings: 'Data type' is set to 'Boolean', 'Client access' is set to 'Read Only', and 'Scan rate' is set to '100 milliseconds'.

EasyBuilder

The used address code is 1x

Stable = address 1091

The screenshot shows the configuration window for a Modbus point in EasyBuilder. It is titled 'Read address'.
There are three main settings: 'PLC name' is set to 'MODBUS TCP/IP', 'Address' is set to '1x' with a sub-field containing '1091', and there is an unchecked checkbox for 'Invert signal'. A 'Setting...' button is located to the right of the PLC name dropdown.

PENKO Modbus protocol

Check if tare is active:

KepServer

The used format is 1xxxxx

Tare active = address 101097

See page 4

Address: 101097

Data type: Boolean

The screenshot shows the 'Identification' and 'Data properties' sections of the KepServer configuration window. In the 'Identification' section, the 'Name' is 'Weigher Status Tare', the 'Address' is '101097', and the 'Description' is 'Check if tare is active'. In the 'Data properties' section, the 'Data type' is set to 'Boolean', 'Client access' is 'Read Only', and the 'Scan rate' is '100 milliseconds'. There are navigation buttons on the right side of the window.

EasyBuilder

The used address code is 1x


























Tare active = address 1097

The screenshot shows the 'Read address' configuration window in EasyBuilder. The 'PLC name' is set to 'MODBUS TCP/IP'. The 'Address' is set to '1x' with a value of '1097' next to it. There is an 'Invert signal' checkbox which is currently unchecked. A 'Setting...' button is located to the right of the PLC name dropdown.

PENKO Modbus protocol

4.7 Tag list

An example of a tag list in the KEPServerEX OPC Server.

Tag Name	Address	Data Type	Scan Rate	Scaling
 Marker 401	000401	Boolean	100	None
 Marker 408	000408	Boolean	100	None
 Reset zero	001001	Boolean	100	None
 Set zero	001002	Boolean	100	None
 Reset tare	001003	Boolean	100	None
 Set tare	001004	Boolean	100	None
 Input 1	100001	Boolean	100	None
 Input 2	100002	Boolean	100	None
 Input 3	100003	Boolean	100	None
 Output 1	100201	Boolean	100	None
 Output 2	100202	Boolean	100	None
 Output 3	100203	Boolean	100	None
 Output 4	100204	Boolean	100	None
 Weigher Status Stable	101091	Boolean	100	None
 Weigher Status Tare	101097	Boolean	100	None
 Indicator 1 weight	300001	Float	100	None
 Indicator 2 gross	300003	Float	100	None
 Indicator 3 net	300005	Float	100	None
 Indicator 1 weight (long)	300101	Long	100	None
 Indicator 2 gross (long)	300103	Long	100	None
 Indicator 3 net (long)	300105	Long	100	None
 Ext Register 1 read	301001	Long	100	None
 Ext Register 2 read	301003	Long	100	None
 Ext Register 1 write	401001	Long	100	None
 Ext Register 2 write	401003	Long	100	None

PENKO Modbus protocol

5 Advanced features

By using register commands, all device parameters can be read and/or written.

The following parameters are used:

Parameter	Data type	Address code	Address	Combined
Input - parameter 1	Double Word	4x	1149	401149
Input - parameter 2	Double Word	4x	1151	401151
Input - parameter 3	Double Word	4x	1153	401153
Input - parameter 4	Double Word	4x	1155	401155
Output - result 1	Double Word	3x	1141	301141
Output - result 2	Double Word	3x	1143	301143
Output - result 3	Double Word	3x	1145	301145
Output - result 4	Double Word	3x	1147	301147
Enable register command mode	Bit	0x	1007	001007
Status register command mode	Bit	1x	1104	101104

Input parameter 1:

Low word → Function code
High word → 0

Output result 1:

Low word → Function code
High word → Error code

The other parameter and result registers depend on the used function.

Activating:











The register command mode is activated by setting bit 1007. On the rising edge of this bit extended registers 71 -78 are cleared. Bit 1104 indicates that the register command mode is active.

Usage:

First write parameter 2, 3 and 4. Write parameter 1 as last because this parameter triggers the action.

PENKO Modbus protocol

Example tags in KepServer:

Tag Name	Address	Data Type	Scan Rate	Scaling
 Enable Reg Func	001007	Boolean	100	None
 Reg Func Active	101104	Boolean	100	None
 Ext 71 read	301141	Long	100	None
 Ext 72 read	301143	Long	100	None
 Ext 73 read	301145	Long	100	None
 Ext 74 read	301147	Long	100	None
 Ext 75 write	401149	Long	100	None
 Ext 76 write	401151	Long	100	None
 Ext 77 write	401153	Long	100	None
 Ext 78 write	401155	Long	100	None

PENKO Modbus protocol

5.1 Function codes

The following function codes are present:

Name	Code	Description
NOP	0	No Operation
CAL_ZERO	1	Calibrate zero by weight
CAL_SPAN	2	Calibrate span by weight
CAL_MV	3	Calibrate in mV/V
CAL_DEADLOAD	4	Calibrate dead load by measuring weight
CAL_INSERT	5	Calibrate multipoint insert by measuring weight
CAL_POINT	6	Calibrate multipoint read point at parameter index
CAL_DELETE	7	Calibrate multipoint delete point at parameter index
CAL_GEOGRAPHIC_ORIGIN_SET	8	Calibrate set geographic origin calibration
CAL_GEOGRAPHIC_ORIGIN_GET	9	Calibrate get geographic origin calibration
CAL_GEOGRAPHIC_LOCAL_SET	10	Calibrate set geographic local calibration
CAL_GEOGRAPHIC_LOCAL_GET	11	Calibrate get geographic local calibration
IND_MAXLOAD_SET	101	Indicator set maximum load
IND_MAXLOAD_GET	102	Indicator get maximum load
PDI_PATH_SET	201	PDI path set
PDI_PROPERTY_SET	202	PDI property set
PDI_PROPERTY_GET	203	PDI property get
PRINT	301	Print ticket or line to printer
PRINT_SUBTOTAL	302	Subtotals to printer not supported by SGM series
PRINT_TOTAL	303	Totals to printer
PRINT_DAYTOTAL	304	Day totals to printer not supported by SGM series
PRINT_BATCHTOTAL	305	Batch totals to printer not supported by SGM series
PRINT_LAYOUT	306	Custom total layout to printer not supported by SGM series and 1020
PRINT_ALIBI	307	Print to Alibi memory not supported by SGM700 series
PRINT_ALIBIMEMORY	308	print full alibi memory to printer not supported by SGM700 series
PRINT_EVENTMEMORY	309	print full event memory to printer not supported by SGM700 series
TOTAL_TOTALIZE	401	Totalize actual stable weight
TOTAL_SUBTOTAL	402	Read or reset actual subtotal
TOTAL_TOTAL	403	Read or reset actual totals
TOTAL_DAYTOTAL	404	Read or reset actual day totals
TOTAL_BATCHTOTAL	405	Read or reset actual batch totals

PENKO Modbus protocol

RFN_PROCESS_RECIPES_GET	501	Read MFL/CHK/BLT recipe
RFN_PROCESS_RECIPES_SET	502	Write MFL/CHK/BLT recipe
RFN_PROCESS_CONFIG_GET	601	Read MFL/CHK/BLT configuration
RFN_PROCESS_CONFIG_SET	602	Write MFL/CHK/BLT configuration
RFN_PROCESS_DATA	701	Read MFL/CHK/BLT process data

5.2 Error codes

The following error codes are present:

Name	Code	Description
SUCCESS	0	Function successful
WRN_WARNING	1000	System warnings:
WRN_TIMEOUT	1001	Generic time-out warning
WRN_TOLOW	1002	Generic parameter to low warning
WRN_TOHIGH	1003	Generic parameter to high warning
WRN_ZERO	1004	Generic parameter/result is zero warning
WRN_NOTZERO	1005	Generic parameter/result is not zero warning
WRN_POSITIVE	1006	Generic parameter is positive warning
WRN_NEGATIVE	1007	Generic parameter is negative warning
WRN_FULL	1008	Generic something is full warning
WRN_EMPTY	1009	Generic something is empty warning
WRN_NOTFOUND	1010	Generic search not found warning
WER_WARNING	1100	Weigher warnings:
WER_NO_TARE	1101	Zero tare level, tare rst
ERR_ERROR	2000	System errors:
ERR_PARAMETER_INCORRECT	2001	Generic parameter error
ERR_TIMEOUT	2002	Generic time-out error
ERR_TOLOW	2003	Generic parameter to low error
ERR_TOHIGH	2004	Generic parameter to high error
ERR_ZERO	2005	Generic parameter/result is zero error
ERR_NOTZERO	2006	Generic parameter/result is not zero error
ERR_POSITIVE	2007	Generic parameter is positive error
ERR_NEGATIVE	2008	Generic parameter is negative error
ERR_FULL	2009	Generic something is full error
ERR_EMPTY	2010	Generic something is empty error
ERR_NOTFOUND	2011	Generic search not found error
ERR_FILE_NOT_FOUND	2012	Generic file not found error
WER_ERROR	2100	Weigher errors:
WER_NOT_STABLE	2101	Weigher not stable



PENKO Modbus protocol

WER_ABOVE_MAXLOAD	2102	Parameter above max load
WER_BELOW_ZERO	2103	Parameter below zero
WER_NOT_IN_ZERO_RANGE	2104	Not in zero range
WER_ARITHMIC_OVERFLOW	2105	Arrhythmic overflow occurred
WER_ADC_OVERFLOW	2106	A/D reads all 1's
WER_ADC_UNDERFLOW	2107	A/D reads all 0's
WER_GAIN_NEGATIVE	2108	Gain ref. < zero ref.
WER_GAIN_OVERFLOW	2109	Gain limit
WER_SAVE	2110	Save errors:
WER_SAVE_FLASH_EXHAUSTED	2111	Flash ROM exhausted
WER_SAVE_CREATE_HEADER	2112	Error on header creation
WER_SAVE_DATA_WRITE	2113	Error on data write
WER_SAVE_HEADER_VALIDATE	2114	Header validation failed
WER_SAVE_DEACTIVATE	2115	Deactivate old data fail
WER_LOAD	2116	Load errors
WER_LOAD_NOT_FOUND	2117	Item not found in store
WER_LOAD_DATA_ERROR	2118	Error in stored data
WER_BAD_CALIBRATION	2119	No calibration available
WER_NOT_ENABLED	2120	Action not enabled
WER_MCAL_NOT_FOUND	2121	Multi-point not found
WER_MCAL_OVERFLOW	2122	Calibration table full
WER_TARE_ACTIVE	2123	Not allowed, tare active
WER_NOT_ALLOWED	2124	Action is not allowed
WER_ADC_NOPOWER	2125	ADC has no power
ERR_DOSER	2200	Doser errors
ERR_POSITION	2300	Position errors
ERR_SPCAPP	2400	SPC-application errors
ERR_SCOPE	2500	Scope errors
ERR_INTERPRETER	2600	Interpreter errors
ERR_USB	3000	USB errors - use USB routines for returning error texts
ERR_FLASH	3100	FLASH file system errors

PENKO Modbus protocol

5.3 Calibration functions

This chapter describes the calibration function codes.

5.3.1 CAL_ZERO

Calibrate zero by weight. Function code = 1.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
1	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
1	Not used	Not used	Not used

5.3.2 CAL_SPAN

Calibrate span by weight. Function code = 2.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
2	Span weight	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
2	Not used	Not used	Not used

Example - calibrate span on 1.200kg:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
2	1200	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
2	Not used	Not used	Not used

PENKO Modbus protocol

Example - calibrate span without loading scale - will result in **error**:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
2	1200	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
138215426	Not used	Not used	Not used

The result is **138215426**

- The function code (low word) is **2**
- The error code (high word) is **2109 - WER_GAIN_OVERFLOW, Gain limit**

5.3.3 CAL_MV

Theoretic calibration by millivolts. Function code = 3.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
3	Fixed point mV/V value	Maximum weight at mV/V	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
3	Not used	Not used	Not used

Example - theoretic calibration of load cell 200kg @ 2.0012mV/V:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
3	20012	200	Not used

PENKO Modbus protocol

5.3.4 CAL_DEADLOAD

Calibrate dead load by measuring weight. Function code = 4.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
4	Actual weight on scale	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
4	Not used	Not used	Not used

Example - calibration of dead load with 12kg on the scale:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
4	12	Not used	Not used

5.3.5 CAL_INSERT

Multipoint calibration up to 10 points. Insert or replace a calibration point. Function code = 5.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
5	Actual weight on scale	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
5	Not used	Not used	Not used

Example - add calibration point of 10.000kg - if the point already exists, its ADC value is replaced:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
5	10000	Not used	Not used

PENKO Modbus protocol

5.3.6 CAL_POINT

Multipoint calibration up to 10 points. Read the calibration point at index (1...10). Function code = 6.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
6	Index (1...10)	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
6	Index	Calibration reference weight	Calibration in mV

Example - read calibration point 1 (10.000kg @ 9.9975mV/V):

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
6	1	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
6	1	10000	9.9975

5.3.7 CAL_DELETE

Multipoint calibration up to 10 points. Delete the calibration point at index (1...10). Function code = 7.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
7	Index (1...10)	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
7	Index	Not used	Not used

Example - delete calibration point 1:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
7	1	Not used	Not used

PENKO Modbus protocol

5.3.8 CAL_GEOGRAPHIC_ORIGIN_SET

Geographic correction. Set the origin calibration location. Function code = 8.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
8	Fixed point latitude degrees	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
8	Not used	Not used	Not used

Example - set origin latitude to 50.00 degrees:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
8	5000	Not used	Not used

5.3.9 CAL_GEOGRAPHIC_ORIGIN_GET

Geographic correction. Get the origin calibration location. Function code = 9.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
9	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
9	Fixed point latitude degrees	Not used	Not used

Example - get origin latitude (50.00 degrees):

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
9	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
9	5000	Not used	Not used

PENKO Modbus protocol

5.3.10 CAL_GEOGRAPHIC_LOCAL_SET

Geographic correction. Set the actual scale location. Function code = 10.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
10	Fixed point latitude degrees	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
10	Not used	Not used	Not used

Example - set actual latitude to 50.00 degrees:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
10	5000	Not used	Not used

5.3.11 CAL_GEOGRAPHIC_LOCAL_GET

Geographic correction. Get the actual scale location. Function code = 11.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
11	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
11	Fixed point latitude degrees	Not used	Not used

Example - get location latitude (50.00 degrees):

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
11	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
11	5000	Not used	Not used

PENKO Modbus protocol

5.4 Indicator functions

This chapter describes the indicator function codes.

5.4.1 IND_MAXLOAD_SET

Set the indicator maximum load. Function code = 101.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
101	Max load	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
101	Not used	Not used	Not used

Example - set the maximum load to 10.020kg:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
101	10020	Not used	Not used

5.4.2 IND_MAXLOAD_GET

Get the indicator maximum load. Function code = 102.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
102	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
102	Max load	Not used	Not used

Example - get the maximum load (10.020kg):

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
102	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
102	10020	Not used	Not used

PENKO Modbus protocol

5.5 PDI functions

This chapter describes the PDI function codes.

5.5.1 PDI_PATH_SET

Set the PDI path to perform the action on. Function code = 201.

Request:

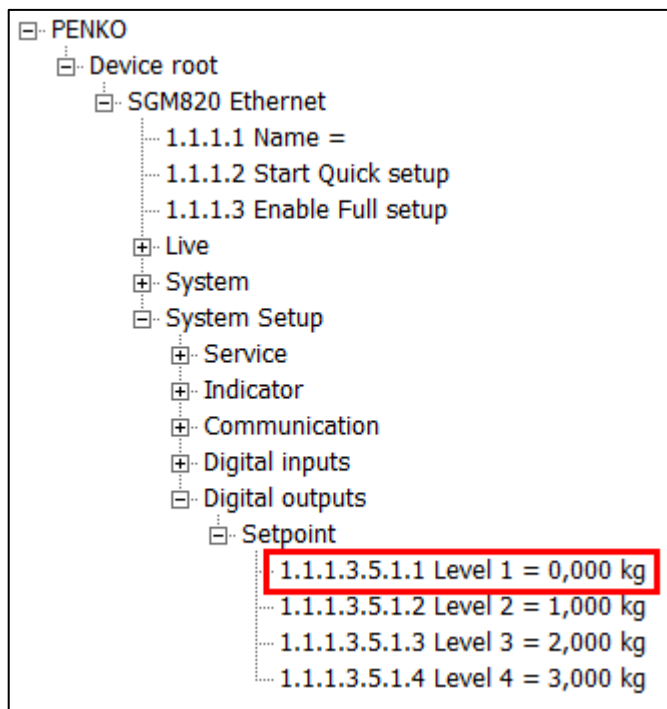
Parameter 1	Parameter 2	Parameter 3	Parameter 4
201	Path no. 1,2,3,4	Path no. 5,6,7,8	Path no. 9,10,11,12

Reply:

Result 1	Result 2	Result 3	Result 4
201	Path no. 1,2,3,4	Path no. 5,6,7,8	Path no. 9,10,11,12

PDI (PENKO Device Interface) represents the device configuration in a tree structure. Every property has its own unique path number. The tree is used in the PENKO configuration tools Pi Mach II and PDI Client, both available at www.penko.com/software

For example, a part of the PENKO SGM820 looks like this:



PENKO Modbus protocol

Setpoint 1 has path number 1.1.1.3.5.1.1. This results in the following input parameters:

Fill the path with zeros to make 12 numbers: 1.1.1.3.5.1.1.0.0.0.0.0

Parameter 1	PDI_PATH_SET	201
Parameter 2	0x01 0x01 0x01 0x03	16843011
Parameter 3	0x05 0x01 0x01 0x00	83951872
Parameter 4	0x00 0x00 0x00 0x00	0

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
201	16843011	83951872	0

Reply:

Result 1	Result 2	Result 3	Result 4
201	16843011	83951872	0

The PDI path is now set.

If the path is not found, all zeros are returned.

PENKO Modbus protocol

5.5.2 PDI_PROPERTY_SET

Set a PDI property for the selected PDI path. Function code = 202.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
202	Property value	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
202	Not used	Not used	Not used

Example - set setpoint 1 to 0.500kg (path must be selected with PDI_PATH_SET):

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
202	500	Not used	Not used

5.5.3 PDI_PROPERTY_GET

Get a PDI property from the selected PDI path. Function code = 203.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
203	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
203	Property value integer/string	Property value string optional	Property value string optional

Example - get setpoint 1 (0.500kg) (path must be selected with PDI_PATH_SET):

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
203	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
203	500	Not used	Not used

PENKO Modbus protocol

Example - get the software version number (1.4.3.9.0.1) (path must be selected with PDI_PATH_SET):

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
203	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
203	825111598	858667310	808333568

The result is a string containing the version number:

Result 2	825111598	0x31 0x2E 0x34 0x2E	1.4.
Result 3	858667310	0x33 0x2E 0x39 0x2E	3.9.
Result 4	808333568	0x30 0x2E 0x31 0x00	0.1

1.4.3.9.0.1

PENKO Modbus protocol

5.6 Printer functions

This chapter describes the printer function codes.

5.6.1 PRINT

Print ticket or line layout, depending on set layout in device. Function code = 301.

For the SGM series printing is only available at the SGM720/820 and SGM750/850.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
301	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
301	Gross weight	Net weight	Tare weight

Example - print:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
301	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
301	699	620	79

Device printer layout setting: **ticket**

```
-----  
DATE                03-09-14  
TIME                11:02.51  
TICKET NUMBER:      42  
  
NET                 0,620 kg  
Tare                0,079 kg  
                   ----- +  
GROSS               0,699 kg  
-----
```

Device printer layout setting: **line**

```
-----  
NR    (PRESET)TARE kg    NET kg  
75    0,079             0,620  
76    0,079             0,620  
77    0,079             0,620  
-----
```

PENKO Modbus protocol

5.6.2 PRINT_SUBTOTAL

Print subtotal to printer. Function code = 302.

Not applicable for the SGM7xx/8xx series.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
302	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
302	Subtotal gross weight	Subtotal net weight	Subtotal tare weight

Example - print subtotal:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
302	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
302	3078	2742	336

Printed ticket:

DATE	03-09-14
TIME	13:53.25
TICKET NUMBER:	3
SUBTOTAL NET	2,742 kg
SUBTOTAL TARE	0,336 kg
	----- +
SUBTOTAL GROSS	3,078 kg

PENKO Modbus protocol

5.6.3 PRINT_TOTAL

Print total to printer. Function code = 303.

For the SGM series printing is only available at the SGM720/820 (Ethernet) and SGM750/850 (serial).

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
303	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
303	Total gross weight	Total net weight	Total tare weight

Example - print total:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
303	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
303	7182	6398	784

Printed ticket:

DATE	03-09-14
TIME	14:02.04
TICKET NUMBER:	7
TOTAL NET	6,398 kg
TOTAL TARE	0,784 kg
	----- +
TOTAL GROSS	7,182 kg

PENKO Modbus protocol

5.6.4 PRINT_DAYTOTAL

Print day total to printer. Function code = 304.

Not applicable for the SGM7xx/8xx series.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
304	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
304	Day total gross weight	Day total net weight	Day total tare weight

Example - print day total:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
304	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
304	3454	3118	336

Printed ticket:

DATE	03-09-14
TIME	14:09.36
TICKET NUMBER:	3
DAY TOTAL NET	3,118 kg
DAY TOTAL TARE	0,336 kg
	----- +
DAY TOTAL GROSS	3,454 kg

PENKO Modbus protocol

5.6.5 PRINT_BATCHTOTAL

Print batch total to printer. Function code = 305.

Not applicable for the SGM7xx/8xx series.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
305	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
305	Batch total gross weight	Batch total net weight	Batch total tare weight

Example - print batch total:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
305	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
305	10636	9516	1120

Printed ticket:

DATE	03-09-14
TIME	14:12.08
TICKET NUMBER:	10
BATCH TOTAL NET	9,516 kg
BATCH TOTAL TARE	1,120 kg
	----- +
BATCH TOTAL GROSS	10,636 kg

PENKO Modbus protocol

5.6.6 PRINT_LAYOUT

Print to Printer function custom layout 1-n. Function code = 306.

Not supported by 1020 and SGM series.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
306	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
306	Layout number 1-n	Not used	Not used

Example - print with custom layout:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
306	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
306	1	Not used	Not used

PENKO Modbus protocol

5.6.7 PRINT_ALIBI

Print to Alibi memory function. Store the actual stable weight in Alibi memory. Function code = 307.

For the SGM series Alibi memory is only available at the SGM8xx series.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
307	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
307	UID*	Gross/Net	Preset (Tare)

* UID can exceed the maximum positive value of the LONG data type (signed). Use the DWORD data type (unsigned).

Example - write to Alibi memory, with active tare:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
307	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
307	1944985600	1315	112

Corresponding Alibi records:

Entry	Code	Date/Value	Time/Unit	UID
00001/00004	Alibi 001	03-09-14	15:00:46	1944985600
00002/00004	Net	1.315	kg	3803586561
00003/00004	Tare	0.112	kg	1269178371
00004/00004	Gross	1.427	kg	0718544901

Example - write to Alibi memory, without active tare:

Reply:

Result 1	Result 2	Result 3	Result 4
307	1660428288	1711	0

Corresponding Alibi records:

Entry	Code	Date/Value	Time/Unit	UID
00001/00002	Alibi 001	03-09-14	15:02:34	1660428288
00002/00002	Gross	1.711	kg	1133518849

PENKO Modbus protocol

5.6.8 PRINT_ALIBIMEMORY

Print the complete Alibi memory to a printer. Function code = 308.

For the SGM series Alibi/printing is only available at the SGM820 and SGM850.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
308	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
308	Not used	Not used	Not used

Example - print the complete Alibi memory:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
308	Not used	Not used	Not used

Printed ticket:

```
Device: 1020
Serial Number FFFFFFFF
Date : 03-09-14   Time : 15:19:19
Alibi Memory
Number UID      Code      Date/Value      Time/Unit
  1 1660428288 Alibi 001      03-09-14      15:11:28
  2 1133518849 Gross      1.711         kg
  3 1941708803 Alibi 001      03-09-14      15:19:08
  4 3786547204 Net        1.162         kg
  5 3158056966 Tare       0.350         kg
  6 0510926856 Gross      1.512         kg
  7 1941708810 Alibi 001      03-09-14      15:19:08
  8 3786547211 Net        1.162         kg
  9 3158056973 Tare       0.350         kg
 10 0510926863 Gross      1.512         kg
```

PENKO Modbus protocol

5.6.9 PRINT_EVENTMEMORY

Print the complete Event log to a printer. Function code = 309.

For the SGM series Event/printing is only available at the SGM820 and SGM850.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
309	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
309	Not used	Not used	Not used

Example - print the complete Event log:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
309	Not used	Not used	Not used

Printer ticket:

```
Device: 1020
Serial Number FFFFFFFF
Date : 03-09-14   Time : 15:33:59
Event Log
Number UID      Code          Date/Value    Time/Unit
  1 0841613312 TAC Changed   30-06-14     11:43:48
  2 1371668481 Events Cleared 30-06-14     11:43:48
  3 1251344386 System Default 30-06-14     12:08:50
  4 1182662659 SoftwareUpdate 30-06-14     12:10:10
  5 4005953540 Alibi Cleared  30-06-14     13:50:20
  6 4005953541 Alibi Cleared  30-06-14     13:50:20
  7 2012479494 CAL Changed   30-06-14     13:58:34
  8 1503395847 CAL Changed   30-06-14     13:58:42
  9 4230086664 Alibi Cleared  30-06-14     14:29:58
 10 3949592585 SoftwareUpdate 31-07-14     09:11:28
```

PENKO Modbus protocol

5.7 Total functions

This chapter describes the total function codes.

5.7.1 TOTAL_TOTALIZE

Totalize actual stable weight. Function code = 401.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
401	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
401	Added gross weight	Added net weight	Added tare weight

Example - add actual stable weight to total:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
401	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
401	1512	1162	350

PENKO Modbus protocol

5.7.2 TOTAL_SUBTOTAL

Get the subtotal weights. Function code = 402.

Not applicable for the SGM7xx/8xx series.

Leave parameter 2 empty to read the subtotal weights.

Set parameter 2 to **0x55AA55AA** to reset the subtotal weights.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
402	Optional	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
402	Subtotal gross weight	Subtotal net weight	Subtotal tare weight

Example - read the subtotal weights:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
402	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
402	12096	9296	2800

Example - reset the subtotal weights:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
402	1437226410	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
402	12096	9296	2800

When reading again, all subtotal weights are 0.

PENKO Modbus protocol

5.7.3 TOTAL_TOTAL

Get the total weights. Function code = 403.

Leave parameter 2 empty to read the total weights.

Set parameter 2 to **0x55AA55AA** to reset the total weights.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
403	Optional	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
403	Total gross weight	Total net weight	Total tare weight

Example - read the total weights:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
403	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
403	12096	9296	2800

Example - reset the total weights:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
403	1437226410	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
403	12096	9296	2800

When reading again, all total weights are 0.

PENKO Modbus protocol

5.7.4 TOTAL_DAYTOTAL

Get the day total weights. Function code = 404.

Not applicable for the SGM7xx/8xx series.

Leave parameter 2 empty to read the day total weights.

Set parameter 2 to **0x55AA55AA** to reset the day total weights.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
404	Optional	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
404	Day total gross weight	Day total net weight	Day total tare weight

Example - read the day total weights:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
404	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
404	12096	9296	2800

Example - reset the day total weights:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
404	1437226410	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
404	12096	9296	2800

When reading again, all day total weights are 0.

PENKO Modbus protocol

5.7.5 TOTAL_BATCHTOTAL

Get the batch total weights. Function code = 405.

Not applicable for the SGM7xx/8xx series.

Leave parameter 2 empty to read the batch total weights.

Set parameter 2 to **0x55AA55AA** to reset the batch total weights.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
405	Optional	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
405	Batch total gross weight	Batch total net weight	Batch total tare weight

Example - read the batch total weights:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
405	Not used	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
405	12096	9296	2800

Example - reset the batch total weights:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
405	1437226410	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
405	12096	9296	2800

When reading again, all batch total weights are 0.

PENKO Modbus protocol

5.8 Controller functions

This chapter describes the controller functions for the belt weigher, check weigher and mono filler.

5.8.1 RFN_PROCESS_RECIPE_GET

Get the value of the selected recipe parameter. Function code = 501.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
501	Recipe param	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
501	Recipe param	Value	Not used

Example - get the value of recipe parameter 1:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
501	1	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
501	1	2000	Not used

5.8.2 RFN_PROCESS_RECIPE_SET

Set the value of the selected recipe parameter. Function code = 502.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
502	Recipe param	Value	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
502	Recipe param	Not used	Not used

Example - set the value of recipe parameter 2 to 500:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
502	2	500	Not used

PENKO Modbus protocol

5.8.3 RFN_PROCESS_CONFIG_GET

Get the value of the selected configuration parameter. Function code = 601.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
601	Config param	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
601	Config param	Value	Not used

Example - get the value of configuration parameter 1:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
601	1	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
601	1	2000	Not used

5.8.4 RFN_PROCESS_CONFIG_SET

Set the value of the selected configuration parameter. Function code = 602.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
602	Config param	Value	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
602	Config param	Not used	Not used

Example - set the value of configuration parameter 2 to 500:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
602	2	500	Not used

PENKO Modbus protocol

5.8.5 RFN_PROCESS_DATA

Get the value of the selected process data parameter. Function code = 701.

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
701	Process param	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
701	Process param	Value	Not used

Example - get the value of process parameter 1:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
701	1	Not used	Not used

Reply:

Result 1	Result 2	Result 3	Result 4
701	1	2000	Not used



About PENKO

At PENKO Engineering we specialize in weighing. Weighing is inherently chemically correct, independent of consistency, type or temperature of the raw material. This means that weighing any kind of material guarantees consistency and thus, it is essential to sustainable revenue generation in any industry. As a well-established and proven solution provider, we strive for the ultimate satisfaction of custom design and/or standard applications, increasing your efficiencies and saving you time, saving you money.

Whether we are weighing raw materials, components in batching, ingredients for mixing or dosing processes, - or weighing of static containers and silos, or - in-motion weighing of railway wagons or trucks, by whatever means required during a process, we are essentially forming vital linkages between processes and businesses, anywhere at any time. We design, develop and manufacture state of the art technologically advanced systems in accordance with your strategy and vision. From the initial design brief, we take a fresh approach and a holistic view of every project, managing, supporting and/or implementing your system every step of the way. Curious to know how we do it? www.penko.com

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. Training sessions on request: www.penko.com/training



PENKO Alliances

PENKO's worldwide network: Australia, Brazil, China, Denmark, Germany, Egypt, Finland, France, India, Italy, Netherlands, Norway, Poland, Portugal, Slovakia, Spain, Syria, Turkey, United Kingdom, South Africa, Slovakia Sweden and Switzerland, Singapore.

A complete overview you will find on: www.penko.com/dealers

