

PENKO Engineering B.V.

Your Partner for Fully Engineered Factory Solutions

```
PPPPPPPPPPPPPPPPPPPP EEEEEEEEEEEEEEEEEENNNNNNNN NNNNNNNKKKKKKKK KKKKKK 00000000
P:::::P E:::::EN:::::N N:::::NK:::::K K:::::K OO:::::OO
P:::::PPPPPP:::::P E:::::EN:::::N N:::::NK:::::K K:::::K OO:::::OO
PP:::::P P:::::PEE:::::EEEEEEEE:::::EN:::::N N:::::NK:::::K K:::::KO:::::OOO:::::O
P:::::P P:::::P E:::::E EEEEEEN:::::N N:::::NKK:::::K K:::::KKKO:::::O O:::::O
P:::::P P:::::P E:::::E N:::::N N:::::N K:::::K K:::::K O:::::O O:::::O
P:::::PPPPPP:::::P E:::::EEEEEEEE N:::::N N:::::N K:::::K K:::::K O:::::O O:::::O
P:::::PP E:::::E N:::::N N:::::N N:::::N K:::::K O:::::O O:::::O
P:::::PPPPPPPP E:::::E N:::::N N:::::N N:::::N K:::::K O:::::O O:::::O
P:::::P E:::::EEEEEEEE N:::::N N:::::N K:::::K K:::::K O:::::O O:::::O
P:::::P E:::::E N:::::N N:::::N K:::::K K:::::K O:::::O O:::::O
P:::::P E:::::E EEEEEEN:::::N N:::::NKK:::::K K:::::KKKO:::::O O:::::O
PP:::::PP EE:::::EEEEEEEE:::::EN:::::N N:::::NK:::::K K:::::KO:::::OOO:::::O
P:::::P E:::::EN:::::N N:::::NK:::::K K:::::K OO:::::OO
P:::::P E:::::EN:::::N N:::::NK:::::K K:::::K OO:::::OO
PPPPPPPP EEEEEEEEEEEEEEEEEENNNNNNNN NNNNNNNKKKKKKKK KKKKKK 00000000
```

Protocol description:
PENKO ASCII



PENKO

an ETC Company

PENKO ASCII protocol

Table of Contents

Introduction.....	5
1 Basics	6
1.1 Devices	6
1.2 Interface	6
2 Serial parameters	7
3 Request/Reply format	8
4 Indicators.....	9
5 Supported commands	10
5.1 Open/Close connection.....	10
5.2 PDI commands.....	11
5.3 Interpreter commands	12
5.4 Calibration commands	13
5.5 Configuration commands	14
5.6 Weighing commands.....	15
5.7 Status and checksum.....	17
6 Setup example.....	18
7 Advanced features.....	19
7.1 Function codes	21
7.2 Error codes	22
7.3 Calibration functions	24
7.3.1 CAL_ZERO	24
7.3.2 CAL_SPAN	24
7.3.3 CAL_MV	25
7.3.4 CAL_DEADLOAD.....	26
7.3.5 CAL_INSERT	26
7.3.6 CAL_POINT.....	27
7.3.7 CAL_DELETE.....	27
7.3.8 CAL_GEOGRAPHIC_ORIGIN_SET	28
7.3.9 CAL_GEOGRAPHIC_ORIGIN_GET.....	28
7.3.10 CAL_GEOGRAPHIC_LOCAL_SET.....	29

PENKO ASCII protocol

7.3.11	CAL_GEOGRAPHIC_LOCAL_GET	29
7.4	Indicator functions	30
7.4.1	IND_MAXLOAD_SET	30
7.4.2	IND_MAXLOAD_GET.....	30
7.5	PDI functions	31
7.5.1	PDI_PATH_SET.....	31
7.5.2	PDI_PROPERTY_SET.....	33
7.5.3	PDI_PROPERTY_GET	33
7.6	Printer functions.....	35
7.6.1	PRINT	35
7.6.2	PRINT_SUBTOTAL	36
7.6.3	PRINT_TOTAL.....	37
7.6.4	PRINT_DAYTOTAL.....	38
7.6.5	PRINT_BATCHTOTAL.....	39
7.6.6	PRINT_LAYOUT	40
7.6.7	PRINT_ALIBI	41
7.6.8	PRINT_ALIBIMEMORY.....	42
7.6.9	PRINT_EVENTMEMORY	43
7.7	Total functions.....	44
7.7.1	TOTAL_TOTALIZE	44
7.7.2	TOTAL_SUBTOTAL	45
7.7.3	TOTAL_TOTAL.....	46
7.7.4	TOTAL_DAYTOTAL	47
7.7.5	TOTAL_BATCHTOTAL.....	48
7.8	Controller functions.....	49
7.8.1	RFN_PROCESS_RECIPES_GET	49
7.8.2	RFN_PROCESS_RECIPES_SET.....	49
7.8.3	RFN_PROCESS_CONFIG_GET.....	50
7.8.4	RFN_PROCESS_CONFIG_SET	50
7.8.5	RFN_PROCESS_DATA.....	51



PENKO ASCII protocol



PENKO ASCII protocol

Introduction

ASCII, or PC protocol, is an ASCII based protocol available over Ethernet and Serial connections. It's not binary data, but an asynchronous half duplex protocol.

RS232/422:

This protocol supports an address field which is used to select devices on a bus. Address 255 is used as an auto transmit mode in which the instrument continues sends the selected indicator value with a fixed time interval. Address 0 disables the address mode.

Ethernet:

Introduced later is the ASCII protocol over Ethernet/TCP. One TCP connection can be setup at port 23. ASCII over TCP is automatically enabled when both RS232 and RS422 are not set to ASCII.

Register commands:

Introduced together with TCP support are the Register command functions. A number of predefined functions, controlled by using the device registers.

PENKO ASCII protocol

1 Basics

The PENKO devices that support the protocol and the used interface.

1.1 Devices

The following PENKO devices support the ASCII protocol:

Device Series	ASCII support	TCP and Register command support
SGM7xx SGM8xx series	Yes	Yes, starting at version v1.5.0.9.0.1
1020 series	Yes	Yes, starting at version v1.4.0.9.0.5
Flex 2100	Yes	No, in progress
Flex	Yes	No, in progress
Flex Multichannel	Yes	No, in progress

1.2 Interface

The PENKO ASCII protocol supports RS232 and RS422 multi drop. RS485 2 wire is not supported.

When both RS232 and RS422 are not set to ASCII, the TCP connection is automatically enabled. One TCP connection can be made at port 23.

PENKO ASCII protocol

2 Serial parameters

The following parameters are used within the ASCII protocol over RS232 and RS422.

Baud rate/Auto transmit interval

Baud rate	Auto transmit interval (minimum time in ms.)
1200	40
2400	40
4800	20
9600	10
19k2	5
38k4	3
57k6	2
115k2	1

Parity

Parity
None(default)
Odd
Even
Mark
Space

Stop bits

Stop bits
1 stop bit(default)
2 stop bit

Data bits

Data bits
8 data bits fixed

PENKO ASCII protocol

3 Request/Reply format

ASCII characters are always in UPPERCASE.

Request frame

ASCII... ASCII<CR>

Reply frame

ASCII... ASCII<CR>

Reply successful

OK<CR>

Reply error

ERR<CR>

PENKO ASCII protocol

4 Indicators

When address is set to 255 the selected indicator will be sent out continuously. The following indicators are available:

Indicator no.	Name	Description
0, 1	WEIGHER	Multi range net weigher value
2	FAST GROSS	Unfiltered gross weigher value
3	FAST NET	Unfiltered net weigher value
4	DISPLAY GROSS	Filtered gross weigher value
5	DISPLAY NET	Filtered net weigher value
6	TARE	Tare value
7	PEAK	Highest reached weigher value
8	VALLEY	Lowest reached weigher value
9	HOLD*	Stored weigher value
10	WEIGHER x10	Multi range net weigher value shown with extra decimal
11	FAST GROSS x10	Unfiltered gross weigher value shown with extra decimal
12	FAST NET x10	Unfiltered net weigher value shown with extra decimal
13	DISPLAY GROSS x10	Filtered gross weigher value shown with extra decimal
14	DISPLAY NET x10	Filtered net weigher value shown with extra decimal
15	TARE x10	Tare value shown with extra decimal
16	PEAK x10	Highest reached weigher value shown with extra decimal
17	VALLEY x10	Lowest reached weigher value shown with extra decimal
18	HOLD x10*	Stored weigher value shown with extra decimal
19	SIGNAL	Direct mV value from the load cell(s)

*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.

The 1020 FMD has different indicator values:

Indicator no.	Name	Description
0, 1	FAST TRACKING	Unfiltered tracking value
2	TRACKING	Filtered tracking value
3	PEAK	Highest reached weigher value
4	VALLEY	Lowest reached weigher value
5	HOLD	Hold value - stored with zero button in hold mode, or with hold input
6	T.I.R.	Total Indicated Readout value - the difference between the peak hold and valley hold value
7	FAST TRACKING x10	Fast tracking shown with extra decimal
8	TRACKING x10	Tracking shown with extra decimal
9	PEAK x10	Peak shown with extra decimal
10	VALLEY x10	Valley shown with extra decimal
11	HOLD x10	Hold shown with extra decimal
12	T.I.R x10	T.I.R. shown with extra decimal

PENKO ASCII protocol

13

SAMPLE

Direct ADC value from the load cell(s)

5 Supported commands

The following commands are available in the ASCII protocol.

5.1 Open/Close connection

Use these commands to open or close a connection. These commands do not apply for a TCP connection.

Device address > 0 and < 255

Request	Reply	Description
OP _[space] 1<CR>	OK<CR>	Open connection to device with address = 1
OP<CR>	O:001<CR>	Check open connection
CL<CR>		Close connection, no reply
OP<CR>		Check open connection, no reply so no connection

Device address 0

Always open mode

Open or close connection is not possible. The connection is always open.

Request	Reply	Description
OP<CR>	O:000<CR>	Check open connection, connection is always open if device address is 0
CL<CR>		Close connection not possible, is always open, no reply

Device address 255

Auto transmit mode

Open or close connection is not possible. The connection is always open.

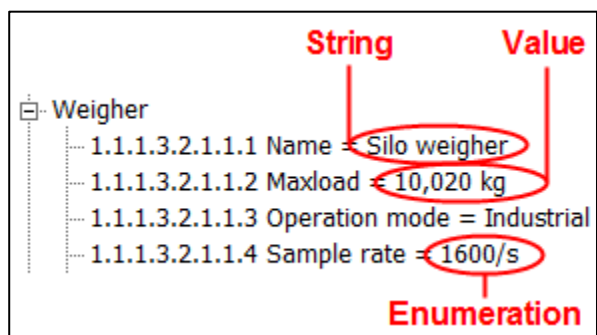
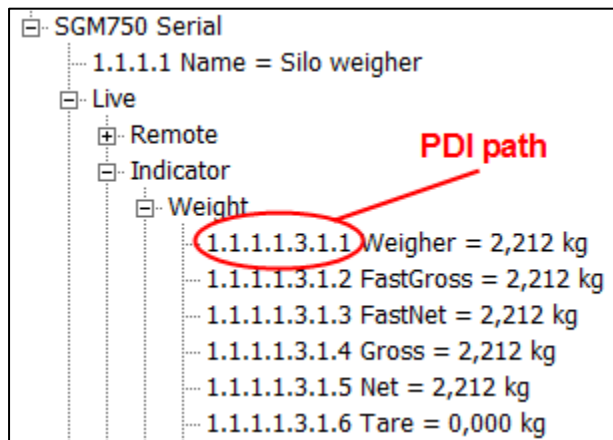
The instrument will send the selected indicator value continuously.

Request	Reply	Description
OP<CR>		Open connection not possible, connection is always open if device address is 255
CL<CR>		Close connection not possible, is always open, auto reply

PENKO ASCII protocol

5.2 PDI commands

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



Request	Reply	Description
PDI available?		
GM<CR>	OK<CR>/ERR<CR>	Check if PDI interface is available
Values		
GM1.1.1.1.3.1.1<CR>	M1.1.1.1.3.1.1: 2.212kg<CR>	Get PDI property Weigher
GM1.1.1.3.2.1.1.2<CR>	M1.1.1.3.2.1.1.2: 10.020kg	Get PDI property Maxload
GM1.1.1.3.2.1.1.2=10050<CR>	OK<CR>/ERR<CR>	Set PDI property Maxload
Strings		
GM1.1.1.3.2.1.1.1<CR>	M1.1.1.3.2.1.1.1:Silo weigher<CR>	Get PDI property Name
GM1.1.1.3.2.1.1.1=Silo 2<CR>	OK<CR>/ERR<CR>	Set PDI property Name
Enumerations		
GM1.1.1.3.2.1.1.4<CR>	M1.1.1.3.2.1.1.4:8<CR>	Get PDI property Sample rate
GM1.1.1.3.2.1.1.4=6<CR>	OK<CR>/ERR<CR>	Set PDI property Sample rate
<i>(Enumerations always start at 0. Sample rate for example has 9 options. The options correspond with values 0 - 8)</i>		
Auto repeat		
SM1.1.1.1.3.1.1<CR>	M1.1.1.1.3.1.1: 2.212kg<CR>	Set auto repeat read PDI property

PENKO ASCII protocol

5.3 Interpreter commands

The interpreter commands are available on the FLEX series, 1020 series and SGM700 series.

Request	Reply	Description
Interpreter Extended Register		
IX<CR>	X000900<CR>	Get number of registers
IX[space]5:[space]1234<CR>	OK<CR>/ERR<CR>	Set register 5
IX[space]5<CR>	X001234<CR>	Get register 5*
Recipe parameters		
IR<CR>	R000014<CR>	Get number of recipes
IR[space]3:[space/sign]1234<CR>	OK<CR>/ ERR<CR>	Set recipe value 3 [**1]
IR[space]3<CR>	R001234<CR>	Get recipe value 3 [**1]
Indicators		
II<CR>	I000030<CR>	Get number of indicators
II[space]18:[space/sign]1234<CR>	OK<CR>/ ERR<CR>	Set indicator 18
II[space]18<CR>	I+01234<CR>	Get indicator 18
II[space]19<CR>	I-- --<CR>	Get indicator 19 (=not available)
Inputs, Outputs and Marker		
IM<CR>	IM001000<CR>	Get number of I/O/M
IM[space]401:[space]1<CR>	OK<CR>/ ERR<CR>	Set marker 401
IM[space]401<CR>	IM 00100000<CR>	Get markers 401..408
IM[space]409:[space]0<CR>	OK<CR>/ ERR<CR>	Reset marker 409
Query and Mark result		
I=<CR>	=+00029+001000121< CR>	Query result value <i>Result index: +00029</i> <i>Result value: +00100</i> <i>Result tag: 01 (hex format)</i> <i>Checksum: 21(hex format)</i>
I=[space]29<CR>	OK<CR>/ ERR<CR>	Mark result value (=delete)

* If you read out a register with a value above 99999 the return value will always be 99999. On 1020 and SGM series the registers above 100 are floating registers and will always return 99999.

****1 - Not supported on 1020 series and SGM700 series**

PENKO ASCII protocol

5.4 Calibration commands

The calibration commands are available on the 1020 series and SGM700 series.

Request	Reply	Description
Calibration support		
CE<CR>	E000002<CR>	Get CAL code
CE _[space] 02<CR>	OK<CR>/ERR<CR>	Enter CAL code
CZ<CR>	OK<CR>/ERR<CR>	Set zero value
CG _[space] 0500<CR> (value)	OK<CR>/ERR<CR>	Set gain value
CS<CR>	OK<CR>/ERR<CR>	Save calibration
CM<CR>	M+10009<CR>	Get max load
CM _[space] 10009<CR> (value)	OK<CR>/ERR<CR>	Set max load

When using the ASCII protocol for calibration, follow these steps.

- Empty the weigher
- Enter CE<CR> to get the CAL code // Reply for example E000003
- Enter CE 3<CR>
- Enter CZ<CR> to set the zero point
- Load the weigher // For example 500kg
- Enter CE<CR> to get the CAL code // Reply for example E000004
- Enter CE 4<CR>
- Enter CG 500<CR> to set the gain point
- Enter CS<CR> to save the calibration

PENKO ASCII protocol

5.5 Configuration commands

The configuration commands are available on the 1020 series and SGM700 series.

Request	Reply	Description
s/w damping factor		
FL<CR>	F000005<CR>	Get s/w damping factor (overall filter)
FL[space]5<CR>	OK<CR>/ERR<CR>	Set s/w damping factor (overall filter)
Virtual display support		
DR<CR>	R000004<CR>	Get display refresh rate
DR[space]4<CR>	OK<CR>/ERR<CR>	Set display refresh rate
DS<CR>	S000006<CR>	Get weigher format step size
DS[space]6<CR>	OK<CR>/ERR<CR>	Set weigher format step size
DP<CR>	D000003<CR>	Get weigher format decimal point position
DP[space]3<CR>	OK<CR>/ERR<CR>	Set weigher format decimal point position
Display filter support		
DD<CR>	D000006<CR>	Get display filter damping
DD[space]6<CR>	OK<CR>/ERR<CR>	Set display filter damping
DZ<CR>	Z+00.050<CR>	Get display zero suppress
DZ[space]00050<CR>	OK<CR>/ERR<CR>	Set display zero suppress
DA<CR>	A+00.060<CR>	Get display filter range
DA[space]00060<CR>	OK<CR>/ERR<CR>	Set display filter range
Zero-tracking support		
TR<CR>	R+00.020<CR>	Get zero tracking range
TR[space]00020<CR>	OK<CR>/ERR<CR>	Set zero tracking range
TS<CR>	S+00.020<CR>	Get zero tracking step
TS[space]00020<CR>	OK<CR>/ERR<CR>	Set zero tracking step
TT<CR>	T000020<CR>	Get zero tracking time
TT[space]000020<CR>	OK<CR>/ERR<CR>	Set zero tracking time
No-motion/stable support		
NR<CR>	R+00.002<CR>	Get stable range
NR[space]00002<CR>	OK<CR>/ERR<CR>	Set stable range
NT<CR>	T000100<CR>	Get stable time
NT[space]000100<CR>	OK<CR>/ERR<CR>	Set stable time

PENKO ASCII protocol

5.6 Weighing commands

The weighing commands are available on the FLEX series, 1020 series and SGM700 series.

Request	Reply	Description
Current display value		
SD<CR>	+02.212<CR>/ERR<CR>	Set auto-transmit Display value
GD<CR>	+02.212<CR>/ERR<CR>	Get Display value
Set/reset system zero		
SZ<CR>	OK<CR>/ERR<CR>	Set Zero
RZ<CR>	OK<CR>/ERR<CR>	Reset Zero
Set/reset tare		
ST<CR>	OK<CR>/ERR<CR>	Set Tare
RT<CR>	OK<CR>/ERR<CR>	Reset Tare
Reset peak/valley		
RP<CR>	OK<CR>/ERR<CR>	Reset Peak
RV<CR>	OK<CR>/ERR<CR>	Reset Valley
Get various channels		
GN<CR>	N+00.456<CR>	Get net
GG<CR>	G+00.694<CR>	Get gross
GT<CR>	T+00.238<CR>	Get tare
GP<CR>	P+03.074<CR>	Get peak
GV<CR>	V+00.082<CR>	Get valley
GF<CR>	F+00.456<CR>	Get fast net (no display damping)
GS<CR>	S000.985<CR>	Get A/D sample
GX<CR>	X+0.0456<CR>	Get extended net (net x 10)
Long string commands		
GW<CR>	W+00456+006944CD9<CR>	Get fast net+gross, status & checksum
LW<CR>	W+00456+006944CD9<CR>	Get long net+gross, status & checksum
LN<CR>	N+00456+004564CE6<CR>	Get net+fast net , status & checksum
LF<CR>	F+00456+006944CEA<CR>	Get long fast net+gross, status & checksum
LX<CR>	X+04556+069364CCE<CR>	Get long extended net (net x 10) + extended gross (gross x 10), status & checksum
Auto-transmit mode support		
SN<CR>	N+00.456<CR>	Set auto-transmit net
SG<CR>	G+00.694<CR>	Set auto-transmit gross
SW<CR>	W+00456+006944CD9<CR>	Set auto-transmit long weight
SP<CR>	P+03.074<CR>	Set auto-transmit peak
SV<CR>	V-00.082<CR>	Set auto-transmit valley
SF<CR>	F+00.456<CR>	Set auto-transmit fast net
SX<CR>	X+0.0456<CR>	Set auto-transmit extended net (net x 10)
Weigher preset tare support		
PT<CR>	P+00.231<CR>	Get preset tare value
PT _[space] 00231<CR>	OK<CR>/ERR<CR>	Set preset tare value
PS<CR>	OK<CR>/ERR<CR>	Set preset tare on
Stub function to fool SAIASSI		
AG<CR>	OK<CR>/ERR<CR>	Stub function to fool SAIASSI
Info		
IV<CR>	V:0101<CR>	Get version number
IS<CR>*	S:001000<CR>	Get system status

PENKO ASCII protocol

ID<CR>	D:0624<CR>	Get device ID code
--------	------------	--------------------

* IS (System status) definition:

Bit #	Called	Definition
0	Stable weight	Weigher signal is stable
1	Zero set	Weigher zero is corrected
2	Tare active	Tare or preset tare is active
7	Register commands	Register command mode is enabled

The 1020 FMD has a few exceptions:

Request	Reply	Description
NOT AVAILABLE		
GG<CR>	G+00.694<CR>	Get gross
SG<CR>	G+00.694<CR>	Set auto-transmit gross
T.I.R. support		
GT<CR>	T+14.238<CR>	Get T.I.R.
ST<CR>	T+14.238<CR>	Set auto-transmit T.I.R.
RT<CR>	OK<CR>/ERR<CR>	Reset T.I.R.

PENKO ASCII protocol

5.7 Status and checksum

The long string commands return 2 weigher values, status and checksum. The status and checksum are represented in a hexadecimal format.

For example, the command `GW<CR>` returns `W+00324+003244CE9<CR>` where “4C” represents the status bits and “E9” represents the checksum. Status bits:

Bit #	Called	Definition
0	Hardware overload	Hardware overload/underload detected on load cell
1	Maximum load	Overload detected on load cell
2	Stable weight	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 range	Weigher signal is in zero tracking range, zero tracking is possible

“4C” from the example is binary 0100 1100, so stable weight, stable range and zero range are active.

The checksum is the inverted sum of all ASCII characters. The return value `W+00324+003244C` from the example:

W + + + 0 + 0 + 3 + 2 + 4 + + + 0 + 0 + 3 + 2 + 4 + 4 + C =

57+2B+30+30+33+32+34+2B+30+30+33+32+34+34+43 = 316 (hex)

Remove the most significant digit = 16 (hex)

Invert the hex value = E9

Convert the hex value E9 to ASCII characters ‘E’ and ‘9’

PENKO ASCII protocol

6 Setup example

PDI setup RS232 address 0, indicator 1

```
[-] RS232
  ... 1.1.3.3.4.1 Protocol = ASCII
  ... 1.1.3.3.4.2 Address = 0
  ... 1.1.3.3.4.3 Stopbits = 1
  ... 1.1.3.3.4.4 Parity = None
  ... 1.1.3.3.4.5 Baudrate = 57600
  ... 1.1.3.3.4.6 Indicator = 1
```

PDI setup RS422 address 2, indicator 1

```
[-] RS422
  ... 1.1.3.3.5.1 Protocol = ASCII
  ... 1.1.3.3.5.2 Address = 2
  ... 1.1.3.3.5.3 Stopbits = 1
  ... 1.1.3.3.5.4 Parity = None
  ... 1.1.3.3.5.5 Baudrate = 9600
  ... 1.1.3.3.5.6 Indicator = 1
```

PDI setup RS232 address 255(=auto transmit), indicator 8

```
[-] RS232
  ... 1.1.3.3.4.1 Protocol = ASCII
  ... 1.1.3.3.4.2 Address = 255
  ... 1.1.3.3.4.3 Stopbits = 1
  ... 1.1.3.3.4.4 Parity = None
  ... 1.1.3.3.4.5 Baudrate = 57600
  ... 1.1.3.3.4.6 Indicator = 8
```

Ethernet

Both RS232 and RS422 must not be set to ASCII. Connect to the IP address of device on port number 23.

```
[-] Ethernet
  ... 1.1.3.3.2.1 MAC = 1E:30:6C:A2:45:5E
  ... 1.1.3.3.2.2 Name = Penko
  ... 1.1.3.3.2.3 Address = 10.1.2.66
  ... 1.1.3.3.2.4 Mask = 255.255.255.0
  ... 1.1.3.3.2.5 Gateway = 0.0.0.0
  ... 1.1.3.3.2.6 DHCP = DISABLE
```

For this setting, connect to 10.1.2.66:23



PENKO ASCII protocol

7 Advanced features

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

The following parameters are used:

Request	Reply	Description
Input registers (request parameters)		
IX[space]75:[space]000000<CR>	OK<CR>/ERR<CR>	Write register 75 (write function register 1)
IX[space]76:[space]000000<CR>	OK<CR>/ERR<CR>	Write register 76 (write function register 2)
IX[space]77:[space]000000<CR>	OK<CR>/ERR<CR>	Write register 77 (write function register 3)
IX[space]78:[space]000000<CR>	OK<CR>/ERR<CR>	Write register 78 (write function register 4)
Output registers (reply results)		
IX[space]71<CR>	X000000<CR>	Read register 71 (read function register 1)
IX[space]72<CR>	X000000<CR>	Read register 72 (read function register 2)
IX[space]73<CR>	X000000<CR>	Read register 73 (read function register 3)
IX[space]74<CR>	X000000<CR>	Read register 74 (read function register 4)
Controls		
RE<CR>	OK<CR>/ ERR<CR>	Enable register command mode
RD<CR>	OK<CR>/ ERR<CR>	Disable register command mode
RX<CR>	OK<CR>/ ERR<CR>	Execute the action
IS<CR>	S:128000<CR>	Status, bit 7 (0x80) indicates mode is active

Input:

Parameter	Data type	Description
1	DINT	Low 2 bytes = function code High 2 bytes = 0
2	DINT	Input parameter, depending on function code
3	DINT	Input parameter, depending on function code
4	DINT	Input parameter, depending on function code <i>For FLEX Multichannel, enter the weigher number (1,2,3 or 4) to perform the action on*</i>

* Not applicable for PDI functions (function codes 2xx)

Output:

Result	Data type	Description
1	DINT	Low 2 bytes = function code High 2 bytes = error code
2	DINT	Result, depending on function code
3	DINT	Result, depending on function code
4	DINT	Result, depending on function code

PENKO ASCII protocol

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 sending command RE. Extended registers 71 -78 are cleared. Bit 7 (0x80) of the IS command indicates that the register command mode is active.

Deactivate the register command mode by sending command RD.

Usage:

Write register 75, and depending on the function code also write register 76, 77 and 78.

Execute the function by sending command RX.

Read register 71, 72, 73 and 74 for the result of the function.

PENKO ASCII protocol

7.1 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 ASCII 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

7.2 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 ASCII 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 ASCII protocol

7.3 Calibration functions

This chapter describes the calibration function codes.

7.3.1 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

7.3.2 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 ASCII 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**

7.3.3 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 ASCII protocol

7.3.4 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

7.3.5 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 ASCII protocol

7.3.6 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

7.3.7 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 ASCII protocol

7.3.8 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

7.3.9 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 ASCII protocol

7.3.10 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

7.3.11 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 ASCII protocol

7.4 Indicator functions

This chapter describes the indicator function codes.

7.4.1 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

7.4.2 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 ASCII protocol

7.5 PDI functions

This chapter describes the PDO function codes.

7.5.1 PDI_PATH_SET

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

Request:

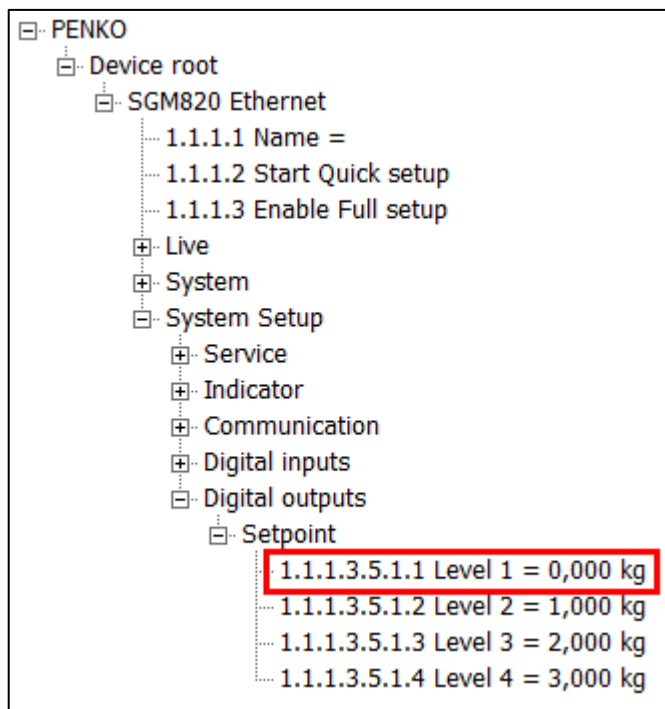
Parameter 1	Parameter 2	Parameter 3	Parameter 4
201	Path no. 1,2,3,4	Path no. 5,6,7,8	Path no. 9,10,11,12

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 ASCII 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 ASCII protocol

7.5.2 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

7.5.3 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 ASCII 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 ASCII protocol

7.6 Printer functions

This chapter describes the printer function codes.

7.6.1 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 ASCII protocol

7.6.2 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 ASCII protocol

7.6.3 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 ASCII protocol

7.6.4 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 ASCII protocol

7.6.5 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 ASCII protocol

7.6.6 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 ASCII protocol

7.6.7 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 ASCII protocol

7.6.8 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 ASCII protocol

7.6.9 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 ASCII protocol

7.7 Total functions

This chapter describes the total function codes.

7.7.1 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 ASCII protocol

7.7.2 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 ASCII protocol

7.7.3 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 ASCII protocol

7.7.4 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 ASCII protocol

7.7.5 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 ASCII protocol

7.8 Controller functions

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

7.8.1 RFN_PROCESS_RECIPES_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

7.8.2 RFN_PROCESS_RECIPES_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 ASCII protocol

7.8.3 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

7.8.4 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 ASCII protocol

7.8.5 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

