

PENKO Engineering B.V.

Your Partner for Fully Engineered Factory Solutions



Protocol description:
PENKO FLEX EtherNet/IP



an ETC Company

PENKO EtherNet/IP protocol

Revision

PENKO EtherNet/IP protocol

Table of Contents

Introduction.....	6
1 Basics	7
1.1 EDS.....	7
1.2 Devices	7
1.3 Classes	7
2 Class 0x01 (1) - Identity	8
2.1 Class Attributes.....	8
2.2 Class Services.....	8
2.3 Instance Attributes	9
2.4 Instance Services	9
3 Class 0x02 (2) - Message router	10
3.1 Class Attributes.....	10
3.2 Class Services.....	10
4 Class 0x04 (4) - Assembly	11
4.1 Class Attributes.....	11
4.2 Class Services.....	11
4.3 Instance Attributes	11
4.4 Indicator Data Type Definition	21
4.5 Weigher-Status word	21
4.6 Weigher-Control word	22
4.7 Weigher-Format word.....	22
4.8 Instance Services	23
4.9 Exclusive Owner Connections	23
4.10 Input Only Connections.....	25
5 Class 0x06 (6) - Connection manager	26
5.1 Class Attributes.....	26
5.2 Instance Attributes	26
5.3 Instance Services	26
5.4 Exclusive Owner Connections	26
5.5 Input Only Connections.....	26
5.6 Output Only Connections.....	26
5.7 Exclusive Owner Connections	26
5.8 Input Only Connections.....	26
5.9 Output Only Connections.....	26
5.10 Exclusive Owner Connections	26
5.11 Input Only Connections.....	26
5.12 Output Only Connections.....	26
5.13 Exclusive Owner Connections	26
5.14 Input Only Connections.....	26
5.15 Output Only Connections.....	26
5.16 Exclusive Owner Connections	26
5.17 Input Only Connections.....	26
5.18 Output Only Connections.....	26
5.19 Exclusive Owner Connections	26
5.20 Input Only Connections.....	26
5.21 Output Only Connections.....	26
5.22 Exclusive Owner Connections	26
5.23 Input Only Connections.....	26
5.24 Output Only Connections.....	26
5.25 Exclusive Owner Connections	26
5.26 Input Only Connections.....	26
5.27 Output Only Connections.....	26
5.28 Exclusive Owner Connections	26
5.29 Input Only Connections.....	26
5.30 Output Only Connections.....	26
5.31 Exclusive Owner Connections	26
5.32 Input Only Connections.....	26
5.33 Output Only Connections.....	26
5.34 Exclusive Owner Connections	26
5.35 Input Only Connections.....	26
5.36 Output Only Connections.....	26
5.37 Exclusive Owner Connections	26
5.38 Input Only Connections.....	26
5.39 Output Only Connections.....	26
5.40 Exclusive Owner Connections	26
5.41 Input Only Connections.....	26
5.42 Output Only Connections.....	26
5.43 Exclusive Owner Connections	26
5.44 Input Only Connections.....	26
5.45 Output Only Connections.....	26
5.46 Exclusive Owner Connections	26
5.47 Input Only Connections.....	26
5.48 Output Only Connections.....	26
5.49 Exclusive Owner Connections	26
5.50 Input Only Connections.....	26
5.51 Output Only Connections.....	26
5.52 Exclusive Owner Connections	26
5.53 Input Only Connections.....	26
5.54 Output Only Connections.....	26
5.55 Exclusive Owner Connections	26
5.56 Input Only Connections.....	26
5.57 Output Only Connections.....	26
5.58 Exclusive Owner Connections	26
5.59 Input Only Connections.....	26
5.60 Output Only Connections.....	26
5.61 Exclusive Owner Connections	26
5.62 Input Only Connections.....	26
5.63 Output Only Connections.....	26
5.64 Exclusive Owner Connections	26
5.65 Input Only Connections.....	26
5.66 Output Only Connections.....	26
5.67 Exclusive Owner Connections	26
5.68 Input Only Connections.....	26
5.69 Output Only Connections.....	26
5.70 Exclusive Owner Connections	26
5.71 Input Only Connections.....	26
5.72 Output Only Connections.....	26
5.73 Exclusive Owner Connections	26
5.74 Input Only Connections.....	26
5.75 Output Only Connections.....	26
5.76 Exclusive Owner Connections	26
5.77 Input Only Connections.....	26
5.78 Output Only Connections.....	26
5.79 Exclusive Owner Connections	26
5.80 Input Only Connections.....	26
5.81 Output Only Connections.....	26
5.82 Exclusive Owner Connections	26
5.83 Input Only Connections.....	26
5.84 Output Only Connections.....	26
5.85 Exclusive Owner Connections	26
5.86 Input Only Connections.....	26
5.87 Output Only Connections.....	26
5.88 Exclusive Owner Connections	26
5.89 Input Only Connections.....	26
5.90 Output Only Connections.....	26
5.91 Exclusive Owner Connections	26
5.92 Input Only Connections.....	26
5.93 Output Only Connections.....	26
5.94 Exclusive Owner Connections	26
5.95 Input Only Connections.....	26
5.96 Output Only Connections.....	26
5.97 Exclusive Owner Connections	26
5.98 Input Only Connections.....	26
5.99 Output Only Connections.....	26
5.100 Exclusive Owner Connections	26

PENKO EtherNet/IP protocol

6.1	Class Attributes.....	27
6.2	Class Services.....	27
6.3	Instance Attributes.....	28
6.4	Instance Services	28
7	Class 0x300 (768) - Weigher	29
7.1	Class Attributes.....	29
7.2	Class Services.....	29
7.3	Instance Attributes.....	30
7.4	Instance Attributes Status	31
7.5	Instance Services	32
8	Register functions.....	33
8.1	Function codes	34
8.2	Error codes	35
8.3	Calibration functions	37
8.3.1	CAL_ZERO	37
8.3.2	CAL_SPAN	37
8.3.3	CAL_MV	38
8.3.4	CAL_DEADLOAD.....	39
8.3.5	CAL_INSERT	39
8.3.6	CAL_POINT.....	40
8.3.7	CAL_DELETE	40
8.3.8	CAL_GEOGRAPHIC_ORIGIN_SET	41
8.3.9	CAL_GEOGRAPHIC_ORIGIN_GET.....	41
8.3.10	CAL_GEOGRAPHIC_LOCAL_SET	42
8.3.11	CAL_GEOGRAPHIC_LOCAL_GET	42
8.4	Indicator functions	43
8.4.1	IND_MAXLOAD_SET	43
8.4.2	IND_MAXLOAD_GET.....	43
8.5	PDI functions	44
8.5.1	PDI_PATH_SET	44
8.5.2	PDI_PROPERTY_SET.....	46

PENKO EtherNet/IP protocol

8.5.3	PDI_PROPERTY_GET	46
8.6	Printer functions.....	48
8.6.1	PRINT	48
8.6.2	PRINT_SUBTOTAL	49
8.6.3	PRINT_TOTAL.....	50
8.6.4	PRINT_DAYTOTAL	51
8.6.5	PRINT_BATCHTOTAL.....	52
8.6.6	PRINT_LAYOUT	53
8.6.7	PRINT_ALIBI	54
8.6.8	PRINT_ALIBIMEMORY.....	55
8.6.9	PRINT_EVENTMEMORY	56
8.7	Total functions.....	57
8.7.1	TOTAL_TOTALIZE	57
8.7.2	TOTAL_SUBTOTAL	58
8.7.3	TOTAL_TOTAL.....	59
8.7.4	TOTAL_DAYTOTAL	60
8.7.5	TOTAL_BATCHTOTAL.....	61
8.8	Controller functions.....	62
8.8.1	RFN_PROCESS_RECIPE_GET	62
8.8.2	RFN_PROCESS_RECIPE_SET	62
8.8.3	RFN_PROCESS_CONFIG_GET	63
8.8.4	RFN_PROCESS_CONFIG_SET	63
8.8.5	RFN_PROCESS_DATA.....	64
9	Examples.....	65
9.1	Class 0x01 execute PDI	65
9.2	Class 0x04 read/write weigher data.....	69
9.3	Class 0x300 read/write weigher data.....	70

PENKO EtherNet/IP protocol

Introduction

EtherNet/IP(TM) is an industrial Ethernet network that combines standard Ethernet technologies with the media-independent Common Industrial Protocol or "CIP."

This document describes the PENKO EtherNet/IP implementation for the PENKO FLEX device range.

PENKO EtherNet/IP protocol

1 Basics

The EDS information, PENKO devices that support the protocol and the available classes.

1.1 EDS

This document describes the following Electronic Data Sheet revisions:

EDS filename FLEX 2100:	Flex 2100 V1.5.0.EDS
EDS filename FLEX 2:	Flex V1.6.0.EDS
EDS filename FLEX Multichannel:	Flex Multichannel V1.6.0.EDS

1.2 Devices

The following PENKO devices support this EDS version:

Device	EIP
Flex 2	Yes, starting at version v1.9.1.9.0.1
Flex Multichannel	Yes, starting at version v1.9.1.9.0.1
Flex 2100	Yes, starting at version v1.9.1.9.0.1

1.3 Classes

The following classes are supported:

Classes
Class 0x01 - Identity
Class 0x02 - Message router
Class 0x04 - Assembly
Class 0x06 - Connection manager
Class 0xF5 - TCP/IP
Class 0x300 - Weigher

For every class the following items are described:

- Class attributes
- Class services/actions for these attributes
- Instance attributes
- Instance services/actions for these attributes

PENKO EtherNet/IP protocol

2 Class 0x01 (1) - Identity

According to the EIP standard, every PENKO device has an Identity object. The devices support one instance of the Identity object.

2.1 Class Attributes

Number	Access	Name	Data type	Value	Description
1	Get	Revision	UINT	1	Revision of this object.
2	Get	Max instance	UINT	1	Maximum instance number of an object currently created in this class level of the device.
3	Get	Number of instance	UINT	1	Number of object instances currently created at this class level of the device.
6	Get	Max ID Number Class Attributes	UINT	7	The attribute ID number of the last class attribute of the class definition implemented in the device.
7	Get	Maximum ID Number Instance Attributes	UINT	7	The attribute ID number of the last instance attribute of the class definition implemented in the device.

2.2 Class Services

Service code	Name	Parameters	Description
1	Get Attribute all	-	Read class attribute 1,2,6,7
5	Reset	UINT 0 1 2	Emulate as closely as possible cycling power on the item the <i>Identity Object</i> represents. This value is the default if this parameter is omitted. Return to the factory default configuration, then emulate cycling power. Return out-of-box configuration with the exception of communication link parameters and emulate cycling power.
14	Get Attribute single	-	Read selected class attribute

PENKO EtherNet/IP protocol

2.3 Instance Attributes

Number	Access	Name	Data type	Value	Description
1	Get	Vendor ID	UINT	1240	Identification of each vendor by number
2	Get	Device Type	UINT	12	Indication of general type of product
3	Get	Product Code	UINT	Id*	Identification of a particular product of an individual vendor
4	Get	Revision Major Revision Minor Revision	STRUCT OF UINT UINT	Rev*	Revision of the item the Identity Object represents
5	Get	Status	WORD		
6	Get	Serial Number	UDINT		Production generated serial number
7	Get	Product Name	SHORT STRING	Name*	Human readable identification

*	Id	Rev	Name
FLEX 2	200	01 04	FLEX
FLEX 2100	201	01 04	FLEX 2100
FLEX MC	202	01 04	FLEX MULTICHANNEL

2.4 Instance Services

Service code	Name	Parameters	Description
1	Get Attribute all	-	Read instance attribute 1-7
5	Reset	UINT 0 1	Emulate as closely as possible cycling power on the item the <i>Identity Object</i> represents. This value is the default if this parameter is omitted. Return to the factory default configuration, and then emulate cycling power.
14	Get Attribute single	-	Read selected instance attribute
125	Execute PDI	Request BYTE ARRAY[] path Reply BYTE ARRAY[] path + BYTE ARRAY[] result	Executes the PDI interface, see examples chapter .

PENKO EtherNet/IP protocol

3 Class 0x02 (2) - Message router

The object within a node that distributes explicit message requests to the appropriate application objects.

3.1 Class Attributes

Number	Access	Name	Data type	Value	Description
1	Get	Revision	UINT	1	Revision of this object.
2	Get	Max instance	UINT	1	Maximum instance number of an object currently created in this class level of the device.
3	Get	Number of instance	UINT	1	Number of object instances currently created at this class level of the device.
6	Get	Max ID Number Class Attributes	UINT	7	The attribute ID number of the last class attribute of the class definition implemented in the device.
7	Get	Maximum ID Number Instance Attributes	UINT	0	The attribute ID number of the last instance attribute of the class definition implemented in the device.

3.2 Class Services

Service code	Name	Parameters	Description
1	Get Attribute all	-	Read class attribute 1,2,3,6,7
14	Get Attribute single	-	Read selected class attribute



Instance attributes and services are not applicable for this class

PENKO EtherNet/IP protocol

4 Class 0x04 (4) - Assembly

Assembly objects provide the option of mapping data from attributes of different instances of various classes into one single attribute, an Assembly Object. This mapping is generally used for I/O messages to maximize the efficiency of the control data exchange on the network.

4.1 Class Attributes

Number	Access	Name	Data type	Value	Description
1	Get	Revision	UINT	2	Revision of this object.
2	Get	Max instance	UINT	49 / 52*	Maximum instance number of an object currently created in this class level of the device.
3	Get	Number of instance	UINT	49 / 52*	Number of object instances currently created at this class level of the device.

* FLEX Multichannel has 52 instances

4.2 Class Services

Service code	Name	Parameters	Description
14	Get Attribute single	-	Read selected class attribute
16	Set Attribute single	Data	Write selected class attribute

4.3 Instance Attributes

Number	Access	Name	Data type	Value	Description
3	Get	Byte array[]	BYTE ARRAY[]		Instance depended data, see instance # See examples chapter

Available instances
Instance 0x0300 (768) Markers group 1 Configuration
Instance 0x0301 (769) Markers group 2 Configuration
Instance 0x0302 (770) Markers group 3 Configuration
Instance 0x0303 (771) Markers group 4 Configuration
Instance 0x0304 (772) Markers input
Instance 0x0305 (773) Markers input
Instance 0x0306 (774) Markers input
Instance 0x0307 (775) Markers input

PENKO EtherNet/IP protocol

Instance 0x0308 (776) Markers output
Instance 0x0309 (777) Markers output
Instance 0x030A (778) Markers output
Instance 0x030B (779) Markers output
Instance 0x0310 (784) Weigher configuration
Instance 0x0311 (785) ... 0x314 (788) Weigher
Instance 0x0321 (801) ... 0x0328 (808) Heartbeat
Instance 0x0330 (816) Indicator group 1 configuration
Instance 0x0331 (817) Indicator group 2 configuration
Instance 0x0332 (818) Indicator group 3 configuration
Instance 0x0333 (819) Indicator group 4 configuration
Instance 0x0334 (820) Indicator group 1
Instance 0x0335 (821) Indicator group 2
Instance 0x0336 (822) Indicator group 3
Instance 0x0337 (823) Indicator group 4
Instance 0x0340 (832) Register group 1 configuration
Instance 0x0341 (833) Register group 2 configuration
Instance 0x0342 (834) Register group 3 configuration
Instance 0x0343 (835) Register group 4 configuration
Instance 0x0344 (836) Register group 1 read
Instance 0x0345 (837) Register group 2 read
Instance 0x0346 (838) Register group 3 read
Instance 0x0347 (839) Register group 4 read
Instance 0x0348 (840) Register group 1 write
Instance 0x0349 (841) Register group 2 write
Instance 0x034A (842) Register group 3 write
Instance 0x034B (843) Register group 4 write
Instance 0x0360 (864) Device configuration
Instance 0x0364 (868) Device in
Instance 0x0368 (872) Device out
Instance 0x0370 (880) Control configuration
Instance 0x0374 (884) Control in
Instance 0x0378 (888) Control out

Instance 0x0300 (768) Markers group 1 Configuration

Access	Name	Data type	Description
Get/Set	Markers group 1 Configuration	STRUCT OF WORD offset input WORD offset output	Markers configuration offset Default read start at 401 (multiple of 8) Default write start at 433 (multiple of 8)

PENKO EtherNet/IP protocol

Instance 0x0301 (769) Markers group 2 Configuration

Access	Name	Data type	Description
Get/Set	Markers group 2 Configuration	STRUCT OF WORD offset input WORD offset output	Markers configuration offset Default read start at 465 (multiple of 8) Default write start at 497 (multiple of 8)

Instance 0x0302 (770) Markers group 3 Configuration

Access	Name	Data type	Description
Get/Set	Markers group 3 Configuration	STRUCT OF WORD offset input WORD offset output	Markers configuration offset Default read start at 529 (multiple of 8) Default write start at 561 (multiple of 8)

Instance 0x0303 (771) Markers group 4 Configuration

Access	Name	Data type	Description
Get/Set	Markers group 4 Configuration	STRUCT OF WORD offset input WORD offset output	Markers configuration offset Default read start at 593 (multiple of 8) Default write start at 625 (multiple of 8)

PENKO EtherNet/IP protocol

Instance 0x0304 (772) Markers input

Access	Name	Data type	Description
Get	Markers input	BYTE ARRAY[4]	Markers 4x8=32 default read at 401-432

Instance 0x0305 (773) Markers input

Access	Name	Data type	Description
Get	Markers input	BYTE ARRAY[4]	Markers 4x8=32 default read at 465-496

Instance 0x0306 (774) Markers input

Access	Name	Data type	Description
Get	Markers input	BYTE ARRAY[4]	Markers 4x8=32 default read at 529-624

Instance 0x0307 (775) Markers input

Access	Name	Data type	Description
Get	Markers input	BYTE ARRAY[4]	Markers 4x8=32 default read at 593-624

Instance 0x0308 (776) Markers output

Access	Name	Data type	Description
Set	Markers output	BYTE ARRAY[4]	Markers 4x8=32 default write at 433-464

Instance 0x0309 (777) Markers output

Access	Name	Data type	Description
Set	Markers output	BYTE ARRAY[4]	Markers 4x8=32 default write at 497-528

Instance 0x030A (778) Markers output

Access	Name	Data type	Description
Set	Markers output	BYTE ARRAY[4]	Markers 4x8=32 default write at 561-592

Instance 0x030B (779) Markers output

Access	Name	Data type	Description
Set	Markers output	BYTE ARRAY[4]	Markers 4x8=32 default write at 625-656



PENKO EtherNet/IP protocol

Instance 0x0310 (784) Weigher configuration

Access	Name	Data type	Description
	Weigher Configuration	STRUCT OF BYTE[]	Reserved for future operation

Instance 0x0311 (785) Weigher

Access	Name	Data type	Description
Get	Weigher*	STRUCT OF DINT WEIGHER DINT GROSS DINT NET DINT TARE DINT WEIGHERx10 DINT GROSSx10 DINT NETx10 DINT TAREx10 WORD FORMAT WORD STATUS	Display rate weigher data Fast Gross weight Fast Net weight Active Tare weight Display rate weigher data x10 Fast Gross weight x10 Fast Net weight x10 Active Tare weight x10 Format bits, see Weigher-Format word Status bits, see Weigher-Status word

* FLEX Multichannel uses 4 weighers:

- 0x0311 (785) weigher 1
- 0x0312 (786) weigher 2
- 0x0313 (787) weigher 3
- 0x0314 (788) weigher 4

Instance 0x0321 (801) ... 0x0328 (808) Heartbeat

Access	Name	Data type	Description
	Heartbeat		Heartbeat

Instance 0x0330 (816) Indicator group 1 configuration

Access	Name	Data type	Description
Get/Set	Indicator group 1 configuration	WORD offset read	Indicator configuration read offset Default read at 1-10

Instance 0x0331 (817) Indicator group 2 configuration

Access	Name	Data type	Description
Get/Set	Indicator group 2 configuration	WORD offset read	Indicator configuration read offset Default read at 11-20

PENKO EtherNet/IP protocol

Instance 0x0332 (818) Indicator group 3 configuration

Access	Name	Data type	Description
Get/Set	Indicator group 3 configuration	WORD offset read	Indicator configuration read offset Default read at 21-30

Instance 0x0333 (819) Indicator group 4 configuration

Access	Name	Data type	Description
Get/Set	Indicator group 4 configuration	WORD offset read	Indicator configuration read offset Default read at 31-40

Instance 0x0334 (820) Indicator group 1

Access	Name	Data type	Description
Get	Indicator group 1	ARRAY[10] OF STRUCT OF INDICATOR	Read indicators group 1, default start read at 1

Instance 0x0335 (821) Indicator group 2

Access	Name	Data type	Description
Get	Indicator group 2	ARRAY[10] OF STRUCT OF INDICATOR	Read indicators group 1, default start read at 11

Instance 0x0336 (822) Indicator group 3

Access	Name	Data type	Description
Get	Indicator group 3	ARRAY[10] OF STRUCT OF INDICATOR	Read indicators group 1, default start read at 21

Instance 0x0337 (823) Indicator group 4

Access	Name	Data type	Description
Get	Indicator group 4	ARRAY[10] OF STRUCT OF INDICATOR	Read indicators group 1, default start read at 31

PENKO EtherNet/IP protocol

Instance 0x0340 (832) Register group 1 configuration

Access	Name	Data type	Description
Get/Set	Register group 1 configuration	STRUCT OF WORD offset read WORD offset write	Registers configuration offset 4 bytes Default read at 101-110 Default write at 111-120

Instance 0x0341 (833) Register group 2 configuration

Access	Name	Data type	Description
Get/Set	Register group 2 configuration	STRUCT OF WORD offset read WORD offset write	Registers configuration offset 4 bytes Default read at 121-130 Default write at 131-140

Instance 0x0342 (834) Register group 3 configuration

Access	Name	Data type	Description
Get/Set	Register group 3 configuration	STRUCT OF WORD offset read WORD offset write	Registers configuration offset 4 bytes Default read at 141-150 Default write at 151-160

Instance 0x0343 (835) Register group 4 configuration

Access	Name	Data type	Description
Get/Set	Register group 4 configuration	STRUCT OF WORD offset read WORD offset write	Registers configuration offset 4 bytes Default read at 161-170 Default write at 171-180

Instance 0x0344 (836) Register group 1 read

Access	Name	Data type	Description
Get	Register group 1 read	ARRAY OF DINT[10]	External Registers [10], default start read at 101

Instance 0x0345 (837) Register group 2 read

Access	Name	Data type	Description
Get	Register group 2 read	ARRAY OF DINT[10]	External Registers [10] , default start read at 121

PENKO EtherNet/IP protocol

Instance 0x0346 (838) Register group 3 read

Access	Name	Data type	Description
Get	Register group 3 read	ARRAY OF DINT[10]	External Registers [10] , default start read at 141

Instance 0x0347 (839) Register group 4 read

Access	Name	Data type	Description
Get	Register group 4 read	ARRAY OF DINT[10]	External Registers [10] , default start read at 161

Instance 0x0348 (840) Register group 1 write

Access	Name	Data type	Description
Set	Register group 1 write	ARRAY OF DINT[10]	External Registers [10] , default start write at 111

Instance 0x0349 (841) Register group 2 write

Access	Name	Data type	Description
Set	Register group 2 write	ARRAY OF DINT[10]	External Registers [10] , default start write at 131

Instance 0x034A (842) Register group 3 write

Access	Name	Data type	Description
Set	Register group 3 write	ARRAY OF DINT[10]	External Registers [10] , default start write at 151

Instance 0x034B (843) Register group 4 write

Access	Name	Data type	Description
Set	Register group 4 write	ARRAY OF DINT[10]	External Registers [10] , default start write at 171

PENKO EtherNet/IP protocol

Instance 0x0360 (864) Device configuration

Access	Name	Data type	Description
Get/Set	Device Configuration		
	Indicator configuration	STRUCT OF WORD offset read	Indicator configuration read offset Default read 1-10
	Register configuration	STRUCT OF WORD offset read WORD offset write	Registers configuration offset 4 bytes Default read at 101-110 Default write at 111-120
	Markers	STRUCT OF WORD offset input WORD offset output	Markers configuration offset Default read start at 401 (multiple of 8) Default write start at 433 (multiple of 8)

Instance 0x0364 (868) Device in

Access	Name	Data type	Description
Get	Device In	STRUCT OF	
	Weigher 1	DINT WEIGHER DINT GROSS DINT NET DINT TARE DINT WEIGHERx10 DINT GROSSx10 DINT NETx10 DINT TAREx10 WORD FORMAT WORD STATUS	Display rate weigher data Fast Gross weight Fast Net weight Active Tare weight Display rate weigher data x10 Fast Gross weight x10 Fast Net weight x10 Active Tare weight x10 Format bits, see Weigher-Format word Status bits, see Weigher-Status word
	Weigher 2*	Same as weigher 1	
	Weigher 3*	Same as weigher 1	
	Weigher 4*	Same as weigher 1	
	Indicator	ARRAY[10] OF INDICATOR	Read indicators, default start read at 1
	Register read	ARRAY OF DINT[10]	External Registers [10], default start read at 101
Controller software	¹ Markers Input	BYTE ARRAY[4]	Markers 4x8=32 default read at 401-432
Indicator software	² Inputs	BYTE ARRAY[2]	Inputs 1-16
	² Outputs	BYTE ARRAY[2]	Outputs 201-216

¹ In case of controller software the instance numbers are used as markers input

² In case of indicator software the instance numbers are used as inputs and outputs

* FLEX Multichannel only

PENKO EtherNet/IP protocol

Instance 0x0368 (872) Device out

Access	Name	Data type	Description
Set	Device Out	STRUCT OF	
	Register write	ARRAY OF DINT[10]	External Registers [10] , default start write at 111
	Markers Output	BYTE ARRAY[4]	Markers 4x8=32 default write at 497-528

Instance 0x0370 (880) Control configuration

Access	Name	Data type	Description
Get/Set	Control Configuration	STRUCT OF	
	Indicator configuration	WORD offset read	Indicator configuration read offset, Default read at 1-10
	Register configuration	STRUCT OF WORD offset read WORD offset write	Registers configuration offset 4 bytes Default read at 101-110 Default write at 111-120
	Markers	STRUCT OF WORD offset input WORD offset output	Markers configuration offset Default read start at 401 (multiple of 8) Default write start at 433 (multiple of 8)

Instance 0x0374 (884) Control in

Access	Name	Data type	Description
Get	Control In	STRUCT OF	
	Weigher	DINT WEAHER DINT GROSS DINT NET DINT TARE DINT WEAHERx10 DINT GROSSx10 DINT NETx10 DINT TAREx10 WORD FORMAT WORD STATUS	Display rate weigher data Fast Gross weight Fast Net weight Active Tare weight Display rate weigher data x10 Fast Gross weight x10 Fast Net weight x10 Active Tare weight x10 Format bits, see Weigher-Format word Status bits, see Weigher-Status word
	Weigher 2*	Same as weigher 1	
	Weigher 3*	Same as weigher 1	
	Weigher 4*	Same as weigher 1	
	Indicator	ARRAY[10] OF INDICATOR	Read indicators, default start read at 1

PENKO EtherNet/IP protocol

Register read	ARRAY OF DINT[10]	External Registers [10], default start read at 101
Markers Input	BYTE ARRAY[4]	Markers 4x8=32 default read at 401-432

* FLEX Multichannel only

Instance 0x0378 (888) Control out

Access	Name	Data type	Description
Set	Control Out	STRUCT OF	
	Weigher 1 Control	ARRAY OF BYTE[2]	Weigher control word, see also Weigher-Control word
	Reserved Control	ARRAY OF BYTE[2]	Set to 0x0000
	Weigher 2 Control*	ARRAY OF BYTE[2]	Weigher control word, see also Weigher-Control word
	Reserved Control*	ARRAY OF BYTE[2]	Set to 0x0000
	Weigher 3 Control*	ARRAY OF BYTE[2]	Weigher control word, see also Weigher-Control word
	Reserved Control*	ARRAY OF BYTE[2]	Set to 0x0000
	Weigher 4 Control*	ARRAY OF BYTE[2]	Weigher control word, see also Weigher-Control word
	Reserved Control*	ARRAY OF BYTE[2]	Set to 0x0000
	Register write	ARRAY OF DINT[10]	External Registers [10] , default start write at 111
	Markers Output	BYTE ARRAY[4]	Markers 4x8=32 default write at 433-464

* FLEX Multichannel only

4.4 Indicator Data Type Definition

TYPE OF INDICATOR:

STRUCT OF

BYTE FMTSTAT
BYTE WEIGHT [3];

4.5 Weigher-Status word

Bit #	Called	Definition
0	OVERLOAD	Hardware overload/underload detected on loadcell
1	MAXLOAD	Overload detected on loadcell
2	STABLE	Weigher signal is stable
3	STABLE RANGE	Weigher signal is in stable range

PENKO EtherNet/IP protocol

4	ZERO SET	Weigher zero is corrected
5	ZERO CENTER	Weigher in center of zero range
6	ZERO RANGE	Weigher is in zero range, zero is possible
7	ZERO TRACK	Weigher signal is in zero tracking range, zero tracking is possible
8	TARE	Weigher tare is active
9	PTARE	Weigher preset tare is active
10	SAMPLE	Used by internal process handling
11	BAD CAL	Calibration is bad, invalid, not available
12	CAL ENABLED	Calibration is enabled, used by internal process handling
13	INDUSTRIAL	If set weigher runs in industrial mode, if reset weigher runs certified operation mode
14	NOT LEVEL	Weigher system in blocking, warming up or scale is not level
15	RESERVED	Reserved mode always 0

4.6 Weigher-Control word

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

* Remark: action on rising edge of bit

4.7 Weigher-Format word

Bit number	Description
#15	Signed/unsigned
	0 = Unsigned
	1 = Signed
#14	Zero suppressing
	0 = Nonzero suppressing
	1 = Zero suppressing
#11 - #8	Display step size
	0000 = Step 1
	0001 = Step 2
	0010 = Step 5
	0011 = Step 10
	0100 = Step 20
	0101 = Step 50
	0110 = Step 100
	0111 = Step 200

PENKO EtherNet/IP protocol

	1000 = Step 500
	1001 = Step 1000
	1010 = Step 2000
	1011 = Step 5000
#2 - #0	Decimal point position
	000 = 000000
	001 = 00000.0
	010 = 0000.00
	011 = 000.000
	100 = 00.0000
	101 = 0.00000

4.8 Instance Services

Service code	Name	Parameters	Description
14	Get Attribute single	-	Read selected instance attribute
16	Set Attribute single	Data	Write selected instance attribute

4.9 Exclusive Owner Connections

FLEX 2 and 2100:

Connection	Name	Assembly O->T	Assembly T->O	Assembly Configuration	Description
1	Marker Group 1	776	772	768	Read/Write 32 markers
2	Marker Group 2	777	773	769	Read/Write 32 markers
3	Marker Group 3	778	774	770	Read/Write 32 markers
4	Marker Group 4	779	775	771	Read/Write 32 markers
10	Register Group 1	840	836	832	Read/Write 10 Extended Registers
11	Register Group 2	841	837	833	Read/Write 10 Extended Registers
12	Register Group 3	842	838	834	Read/Write 10 Extended Registers
13	Register Group 4	843	839	835	Read/Write 10 Extended Registers

PENKO EtherNet/IP protocol

14	FLEX	872	868	864	Read Weigher, Read Indicators, Read/Write Registers, Read/Write Markers
15	FLEX CONTROL	888	884	880	Read Weigher, Read Indicators, Read/Write Registers, Read/Write Markers, Weigher Control

FLEX Multichannel:

Connection	Name	Assembly O->T	Assembly T->O	Assembly Configuration	Description
1	Marker Group 1	776	772	768	Read/Write 32 markers
2	Marker Group 2	777	773	769	Read/Write 32 markers
3	Marker Group 3	778	774	770	Read/Write 32 markers
4	Marker Group 4	779	775	771	Read/Write 32 markers
13	Register Group 1	840	836	832	Read/Write 10 Extended Registers
14	Register Group 2	841	837	833	Read/Write 10 Extended Registers
15	Register Group 3	842	838	834	Read/Write 10 Extended Registers
16	Register Group 4	843	839	835	Read/Write 10 Extended Registers
17	FLEX	872	868	864	Read Weigher, Read Indicators, Read/Write Registers, Read/Write Markers
18	FLEX CONTROL	888	884	880	Read Weigher, Read Indicators, Read/Write Registers, Read/Write Markers, Weigher Control

PENKO EtherNet/IP protocol

4.10 Input Only Connections

FLEX 2 and 2100:

Connection	Name	Assembly O->T	Assembly T->O	Assembly Configuration	Description
5	Weigher 1	801	785	768	Weigher 1 live data
6	Indicator Group 1	805	820	816	Read 10 Indicator Registers
7	Indicator Group 2	806	821	817	Read 10 Indicator Registers
8	Indicator Group 3	807	822	818	Read 10 Indicator Registers
9	Indicator Group 4	808	823	819	Read 10 Indicator Registers

FLEX Multichannel:

Connection	Name	Assembly O->T	Assembly T->O	Assembly Configuration	Description
5	Weigher 1	801	785	768	Weigher 1 live data
6	Weigher 2	802	786	769	Weigher 2 live data
7	Weigher 3	803	787	770	Weigher 3 live data
8	Weigher 4	804	788	771	Weigher 4 live data
9	Indicator Group 1	805	820	816	Read 10 Indicator Registers
10	Indicator Group 2	806	821	817	Read 10 Indicator Registers
11	Indicator Group 3	807	822	818	Read 10 Indicator Registers
12	Indicator Group 4	808	823	819	Read 10 Indicator Registers

PENKO EtherNet/IP protocol

5 Class 0x06 (6) - Connection manager

The Connection manager describes connections supported by the PENKO device.

5.1 Class Attributes

Number	Access	Name	Data type	Value	Description
1	Get	Revision	UINT	1	Revision of this object.
2	Get	Max instance	UINT	1	Maximum instance number of an object currently created in this class level of the device.
3	Get	Number of instance	UINT	1	Number of object instances currently created at this class level of the device.
6	Get	Max ID Number Class Attributes	UINT	7	The attribute ID number of the last class attribute of the class definition implemented in the device.
7	Get	Maximum ID Number Instance Attributes	UINT	0	The attribute ID number of the last instance attribute of the class definition implemented in the device.

5.2 Class Services

Service code	Name	Parameters	Description
1	Get Attribute all	-	Read class attribute 1,2,3,6,7
14	Get Attribute single	-	Read selected class attribute



Instance attributes and services are not applicable for this class

PENKO EtherNet/IP protocol

6 Class 0xF5 (245) - TCP/IP

The TCP/IP Interface object provides a mechanism for configuring a device's TCP/IP network interface. Examples of configurable items include the device's IP address, network mask and gateway address.

6.1 Class Attributes

Number	Access	Name	Data type	Value	Description
1	Get	Revision	UINT	1	Revision of this object.
2	Get	Max instance	UINT	1	Maximum instance number of an object currently created in this class level of the device.
3	Get	Number of instance	UINT	1	Number of object instances currently created at this class level of the device.
6	Get	Max ID Number Class Attributes	UINT	7	The attribute ID number of the last class attribute of the class definition implemented in the device.
7	Get	Maximum ID Number Instance Attributes	UINT	6	The attribute ID number of the last instance attribute of the class definition implemented in the device.

6.2 Class Services

Service code	Name	Parameters	Description
1	Get Attribute all	-	Read class attribute 1,2,3,6,7
14	Get Attribute single	-	Read selected class attribute

PENKO EtherNet/IP protocol

6.3 Instance Attributes

Number	Access	Name	Data type	Value	Description
1	Get	Status	DWORD		Interface status, see Vol21.11 5-3.3.2.1
2	Get	Configuration Capability	DWORD		Interface capability flags, , see Vol21.11 5-3.3.2.2
3	Get	Configuration Control	DWORD		Interface control flags, , see Vol21.11 5-3.3.2.3.1
4	Get	Physical Link Object	STRUCT of UINT EPATH		Physical link object, see also Vol21.11 Path Size Path
5	Get	Interface configuration IP Address Network Gateway Address Name Server Name Server2 Domain Name	STRUCT OF UDINT UDINT UDINT UDINT UDINT STRING		
6	Get	Host Name	STRING		

6.4 Instance Services

Service code	Name	Parameters	Description
1	Get Attribute all	-	Read instance attribute 1 - 6
14	Get Attribute single	-	Read selected instance attribute

PENKO EtherNet/IP protocol

7 Class 0x300 (768) - Weigher

The weigher class is a custom PENKO class used to read/write weigher data.

7.1 Class Attributes

Number	Access	Name	Data type	Value	Description
1	Get	Revision	UINT	2	Revision of this object.
2	Get	Max instance	UINT	1 / 4*	Maximum instance number of an object currently created in this class level of the device.
3	Get	Number of instance	UINT	1 / 4*	Number of object instances currently created at this class level of the device.
6	Get	Max ID Number Class Attributes	UINT	7	The attribute ID number of the last class attribute of the class definition implemented in the device.
7	Get	Maximum ID Number Instance Attributes	UINT	18	The attribute ID number of the last instance attribute of the class definition implemented in the device.

* FLEX Multichannel has 4 instances - weigher 1, 2, 3, 4

7.2 Class Services

Service code	Name	Parameters	Description
1	Get Attribute all	-	Read class attribute 1,2,6,7
14	Get Attribute single	-	Read selected class attribute

PENKO EtherNet/IP protocol

7.3 Instance Attributes

Number	Access	Name	Data type	Value	Description
1	Get	WEIGHER	DINT		Display rate weigher data including multi range/interval step size
2	Get	FAST GROSS	DINT		Unfiltered weigher data gross
3	Get	FAST NET	DINT		Unfiltered weigher data net
4	Get	GROSS	DINT		Display rate weigher data gross
5	Get	NET	DINT		Display rate weigher data net
6	Get	TARE	DINT		Active tare weight
7	Get	PEAK	DINT		Measured peak weight since last reset
8	Get	VALLEY	DINT		Measured valley weight since last reset
9	Get	WEIGHER x10	DINT		Display rate weigher data x10
10	Get	FAST GROSS x10	DINT		Unfiltered weigher data gross x10
11	Get	FAST NET x10	DINT		Unfiltered weigher data net x10
12	Get	GROSS x10	DINT		Display rate weigher data gross x10
13	Get	NET x10	DINT		Display rate weigher data net x10
14	Get	TARE x10	DINT		Active tare weight x10
15	Get	PEAK x10	DINT		Measured peak weight since last reset x10
16	Get	VALLEY x10	DINT		Measured valley weight since last reset x10
17	Get	SAMPLE	DINT		Internal resolution
18	Get	STATUS	WORD		Weigher status bits Instance Attributes Status

PENKO EtherNet/IP protocol

7.4 Instance Attributes Status

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

PENKO EtherNet/IP protocol

7.5 Instance Services

Service code	Name	Parameters	Description
1	Get Attribute all	-	Read instance attributes 1-18
14	Get Attribute single	-	Read selected instance attribute
50	Zero Set	-	Weigher zero function
51	Zero Reset	-	Weigher zero reset function
52	Tare On	-	Weigher tare on function
53	Tare Off	-	Weigher tare off function
54	Tare Toggle	-	Weigher tare toggle function
55	Preset Tare	Tare weight DINT	Weigher preset tare function
56	Hold Set	-	Weigher hold set function, Not available in revision 1.1
57	Peak Reset	-	Weigher peak reset function Not available in revision 1.1
58	Valley Reset	-	Weigher valley reset function Not available in revision 1.1
64	Calibrate Zero	Security code= 0x0055AAFF	Actual weight is 0
65	Calibrate Span	Security code= 0x0055AAFF Weight DINT	Calibrate Span Actual weight on scale
66	Calibrate mV	Security code= 0xFFAA5500 mV DINT Max.loadcell	Removed the deadload, the span is unattended fixed point 0.00000mV Maximum weigh performance loadcell
67	Calibrate Deadload	Security code= 0xFFAA5500 Weight DINT	Measure the deadload and correct, the span is unattended. Correction weight if scale is not empty
80	Register Functions	DINT in[4] DINT out[4]	See Register Functions In parameters Output parameters

[See examples chapter](#)

PENKO EtherNet/IP protocol

8 Register functions

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

In other PENKO protocols these functions use the device registers to send and receive data. With EtherNet/IP, complete blocks of data can be sent, making the use of the device registers unnecessary.

Register functions are found in the Weigher class, class 0x0300 (768), instance service 80.

The following parameters are used:

Parameter	Data type
Input - parameters	DINT [4]
Output - result	DINT [4]

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 EtherNet/IP protocol

8.1 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
PRINT_TOTAL	303	Totals to printer
PRINT_DAYTOTAL	304	Day totals to printer
PRINT_BATCHTOTAL	305	Batch totals to printer
PRINT_LAYOUT	306	Custom total layout to printer
PRINT_ALIBI	307	Print to Alibi memory
PRINT_ALIBIMEMORY	308	print full alibi memory to printer
PRINT_EVENTMEMORY	309	print full event memory to printer
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
RFN_PROCESS_RECIPE_GET	501	Read MFL/CHK/BLT recipe
RFN_PROCESS_RECIPE_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

PENKO EtherNet/IP protocol

RFN_PROCESS_DATA	701	Read MFL/CHK/BLT process data
------------------	-----	-------------------------------

8.2 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
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

PENKO EtherNet/IP protocol

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 EtherNet/IP protocol

8.3 Calibration functions

This chapter describes the calibration function codes.

8.3.1 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

8.3.2 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 EtherNet/IP 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**

8.3.3 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 EtherNet/IP protocol

8.3.4 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

8.3.5 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 EtherNet/IP protocol

8.3.6 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

8.3.7 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 EtherNet/IP protocol

8.3.8 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

8.3.9 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 EtherNet/IP protocol

8.3.10 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

8.3.11 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 EtherNet/IP protocol

8.4 Indicator functions

This chapter describes the indicator function codes.

8.4.1 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

8.4.2 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 EtherNet/IP protocol

8.5 PDI functions

This chapter describes the PDI function codes.

8.5.1 PDI_PATH_SET

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

Request:

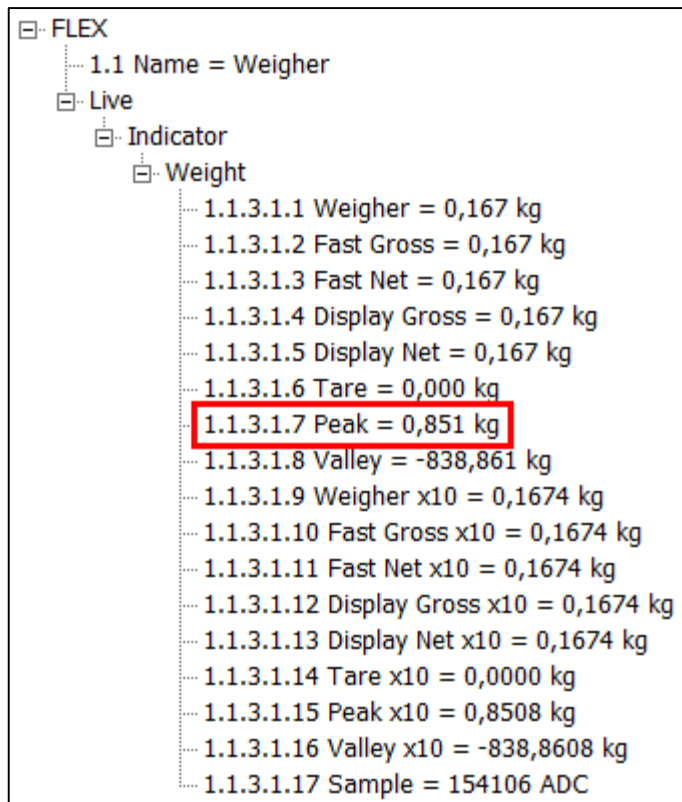
Parameter 1	Parameter 2	Parameter 3	Parameter 4
201	Path no. 1,2,3,4	Path no. 5,6,7,8	Path no. 9,10,11,12

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 FLEX 2 looks like this:



PENKO EtherNet/IP protocol

Peak has path number 1.1.3.1.7. This results in the following input parameters:

Fill the path with zeros to make 12 numbers: 1.1.3.1.7.0.0.0.0.0.0

Parameter 1	PDI_PATH_SET	201
Parameter 2	0x01 0x01 0x03 0x01	16843521
Parameter 3	0x07 0x00 0x00 0x00	117440512
Parameter 4	0x00 0x00 0x00 0x00	0

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
201	16843521	117440512	0

Reply:

Result 1	Result 2	Result 3	Result 4
201	16843521	117440512	0

The PDI path is now set.

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

PENKO EtherNet/IP protocol

8.5.2 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 max load to 8000 (path must be selected with PDI_PATH_SET):

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
202	8000	Not used	Not used

8.5.3 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 peak value (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	851	Not used	Not used

PENKO EtherNet/IP protocol

Example - get the software version number (1.9.1.9.0.1) (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	825112878	825112878	808333568

The result is a string containing the version number:

Result 2	825112878	0x31 0x2E 0x39 0x2E	1.9.
Result 3	825112878	0x31 0x2E 0x39 0x2E	1.9.
Result 4	808333568	0x30 0x2E 0x31 0x00	0.1

1.9.1.9.0.1

PENKO EtherNet/IP protocol

8.6 Printer functions

This chapter describes the printer function codes.

8.6.1 PRINT

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

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 EtherNet/IP protocol

8.6.2 PRINT_SUBTOTAL

Print subtotal to printer. Function code = 302.

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 EtherNet/IP protocol

8.6.3 PRINT_TOTAL

Print total to printer. Function code = 303.

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 EtherNet/IP protocol

8.6.4 PRINT_DAYTOTAL

Print day total to printer. Function code = 304.

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 EtherNet/IP protocol

8.6.5 PRINT_BATCHTOTAL

Print batch total to printer. Function code = 305.

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 EtherNet/IP protocol

8.6.6 PRINT_LAYOUT

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

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 EtherNet/IP protocol

8.6.7 PRINT_ALIBI

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

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 EtherNet/IP protocol

8.6.8 PRINT_ALIBIMEMORY

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

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 EtherNet/IP protocol

8.6.9 PRINT_EVENTMEMORY

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

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 EtherNet/IP protocol

8.7 Total functions

This chapter describes the total function codes.

8.7.1 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 EtherNet/IP protocol

8.7.2 TOTAL_SUBTOTAL

Get the subtotal weights. Function code = 402.

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 EtherNet/IP protocol

8.7.3 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 EtherNet/IP protocol

8.7.4 TOTAL_DAYTOTAL

Get the day total weights. Function code = 404.

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 EtherNet/IP protocol

8.7.5 TOTAL_BATCHTOTAL

Get the batch total weights. Function code = 405.

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 EtherNet/IP protocol

8.8 Controller functions

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

8.8.1 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

8.8.2 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 EtherNet/IP protocol

8.8.3 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

8.8.4 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 EtherNet/IP protocol

8.8.5 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

PENKO EtherNet/IP protocol

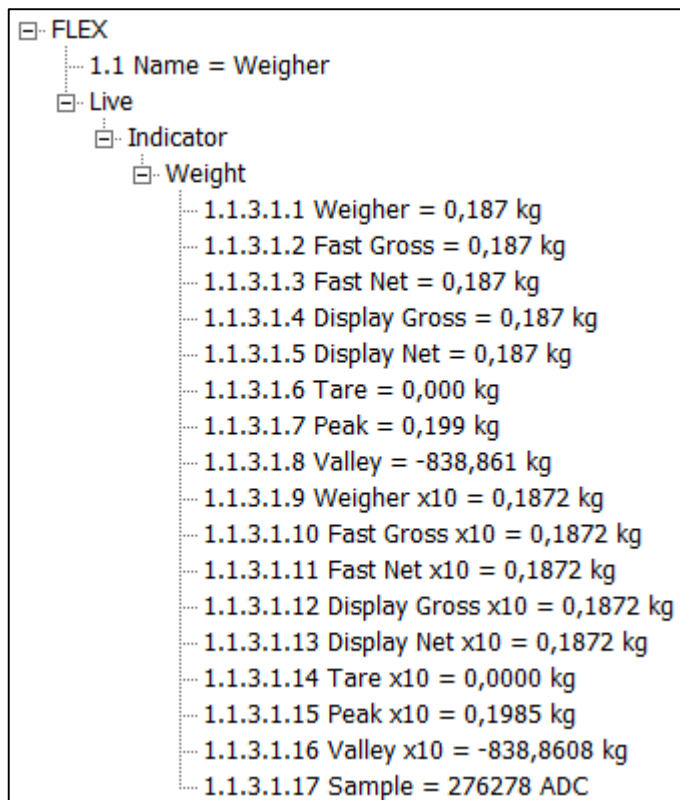
9 Examples

These examples are made with the ODVA EtherNet/IP™ Protocol Conformance Test.

9.1 Class 0x01 execute PDI

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 PDI tree looks like this:



Consult the PENKO PDI documentation for more information, available at www.penko.com

The PDI request exists out of the PDI command, a PDI function and a path. The command is a fixed value; 0xB4. The PDI functions are shown in the following table. The path indicates the parameter.

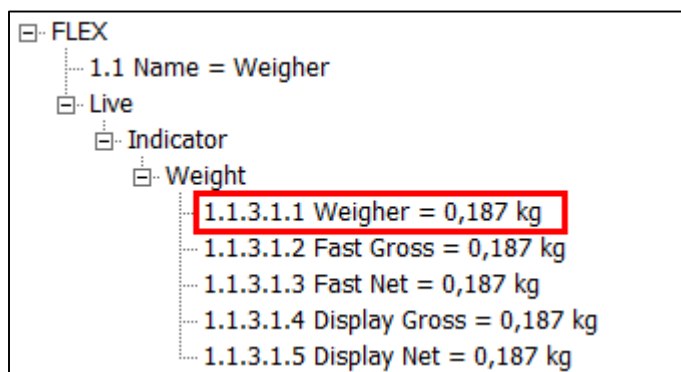
The PDI reply exists out of the full request and the result.

PENKO EtherNet/IP protocol

PDI function	Description
0x00 PDI_AVAIL	Function available
0x01 PDI_ENUMERATE	Enumerate structure
0x02 PDI_PROPERTY	Property structure
0x03 PDI_GET	Value read
0x04 PDI_SET	Value write
0x05 PDI_SETANDREPLY	Value write and get reply code

Read weigher value

The weigher value is found at PDI path 1.1.3.1.1



The used PDI function is PDI_GET

Request:

Command	Function	Path
0xB4	0x03	0x01 01 03 01 01

Reply:

Command	Function	Path	Status	Property value
0xB4	0x03	0x01 01 03 01 01	0x01	0x00 00 00 BB

- Status 0x01 is OK
- The weight is 0x00 00 00 BB → 187

With test tool:

PENKO EtherNet/IP protocol

Conform Tools

Send CIP Message

IP Address: 10 . 1 . 2 . 55

Service: 125 Class: 1 Instance: 1

Attribute:

Service data (hex e.g. 0a): B4030101030101

Send Msg

Response (1):

Reply service: fd

General status: 00 Additional status:

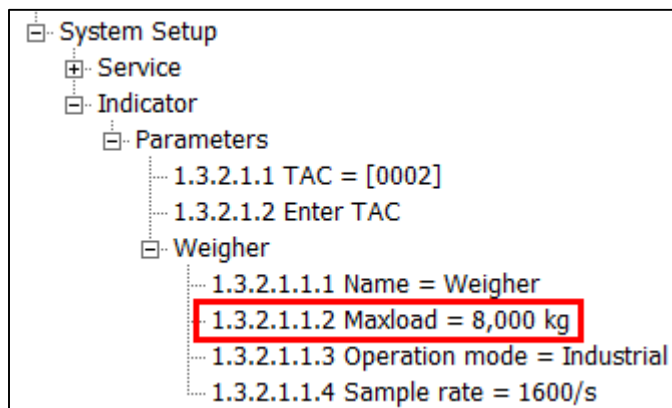
Data (12 bytes):

B4030101030101000000BB

Close

Write max load

Max load is found at PDI path 1.3.2.1.1.2



The used PDI function is PDI_SET

A null terminator is added to separate the path and the value.

To set this setpoint to 10.000, the following request is sent:

Command	Function	Path	Terminator	Property value
0xB4	0x04	0x01 03 02 01 01 02	0x00	0x00 00 27 10

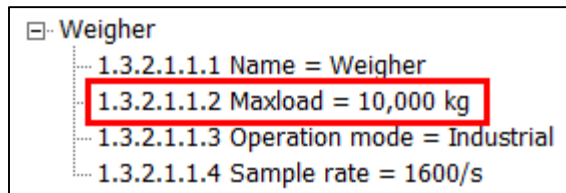
PENKO EtherNet/IP protocol

The reply will be as follows:

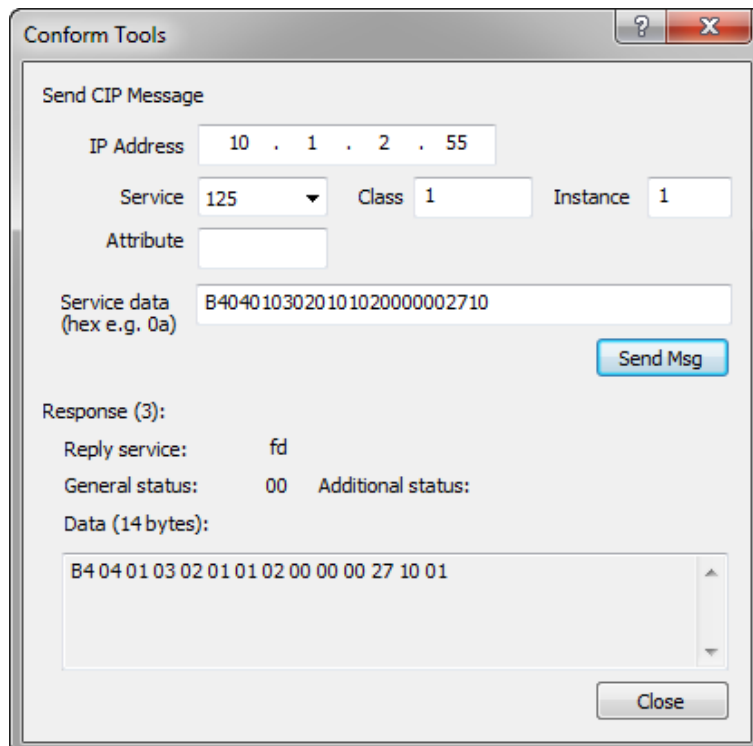
Command	Function	Path	Terminator	Property value	Save OK
0xB4	0x04	0x01 03 02 01 01 02	0x00	0x00 00 27 10	0x01

- The set value is 0x00 00 27 10 → decimal 10.000
- Save OK 0x01 is succeeded

Changed value:



With test tool:



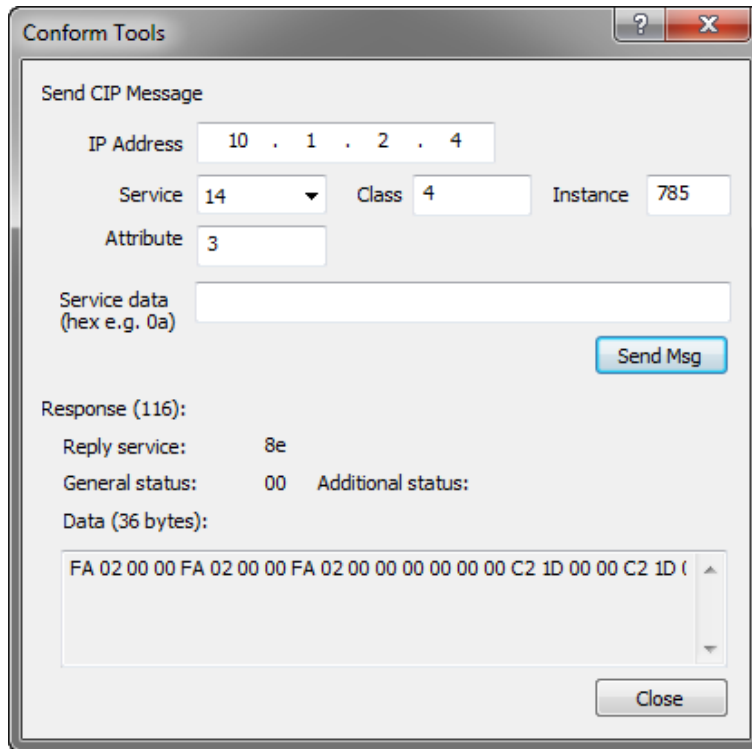
PENKO EtherNet/IP protocol

9.2 Class 0x04 read/write weigher data

The assembly class contains several instances to read or write weigher data. The used attribute number for all instances is 3.

Get weigher data

Instance 785 holds a list of weigher data.



The following data is returned. Mind the byte order within every DINT and WORD.

Data	Data type	Value HEX	Value
WEIGHER	DINT	00 00 00 BB	187
GROSS	DINT	00 00 00 BB	187
NET	DINT	00 00 00 BB	187
TARE	DINT	00 00 00 00	0
WEIGHERx10	DINT	00 00 07 50	1872
GROSSx10	DINT	00 00 07 50	1872
NETx10	DINT	00 00 07 50	1872
TAREx10	DINT	00 00 00 00	0
FORMAT	WORD	C0 03	Bit# 0, 1, 14 and 15*
STATUS	WORD	24 CC	Bit# 2, 3, 6, 7, 10 and 13**

* Decimal point position = 3, display step-size = 1, zero-suppressing = true, signed = true

** Stable = true, stable range = true, zero range = true, zero track = true, sample = true, industrial = true

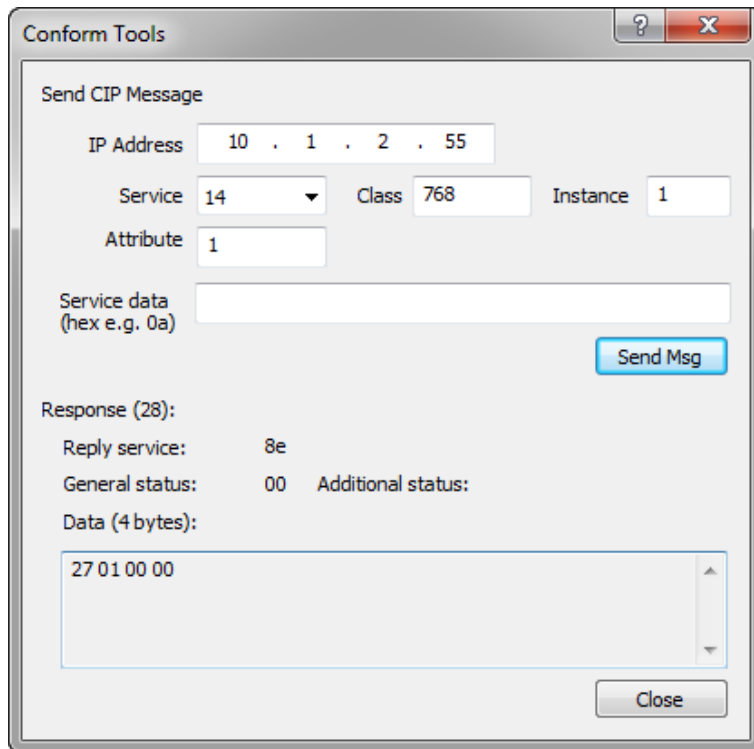
PENKO EtherNet/IP protocol

9.3 Class 0x300 read/write weigher data

The weigher class is a custom PENKO class used to read/write weigher data.

Get weigher value

Instance attribute 1 holds the weigher value. Instance service 14 reads a single instance attribute.



The screenshot shows a 'Conform Tools' dialog box with the following fields and values:

- Send CIP Message**
 - IP Address: 10 . 1 . 2 . 55
 - Service: 14 (dropdown)
 - Class: 768
 - Instance: 1
 - Attribute: 1
 - Service data (hex e.g. 0a): (empty text box)
 - Send Msg** button
- Response (28):**
 - Reply service: 8e
 - General status: 00 Additional status:
 - Data (4 bytes): 27 01 00 00 (text box with scrollbar)
 - Close** button

The reply 0x00 00 01 27 → 295

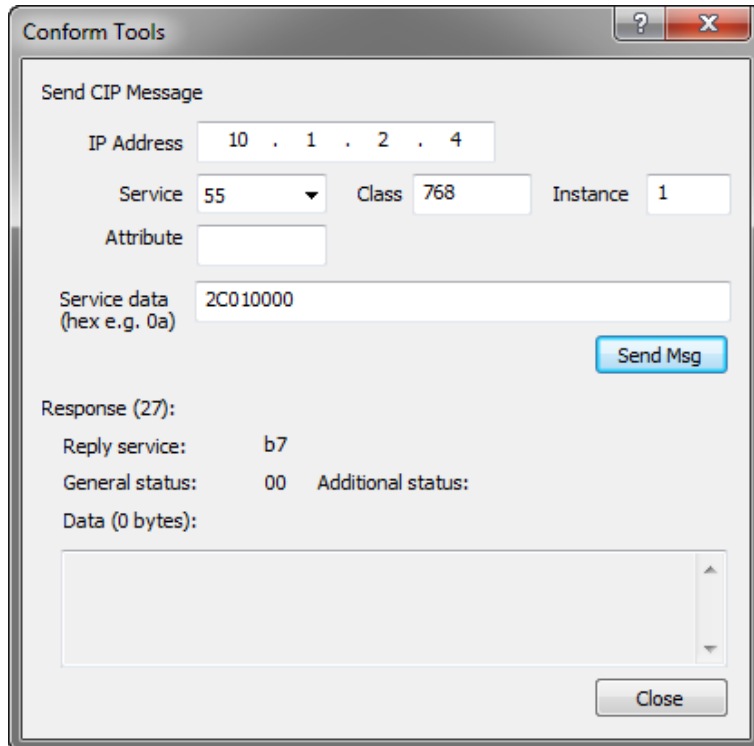
PENKO EtherNet/IP protocol

Set the preset tare

Instance service 55 sets the preset tare value.

Enter the value as parameter. Data type is DINT.

Set the preset tare to 300 → 00 00 01 2C

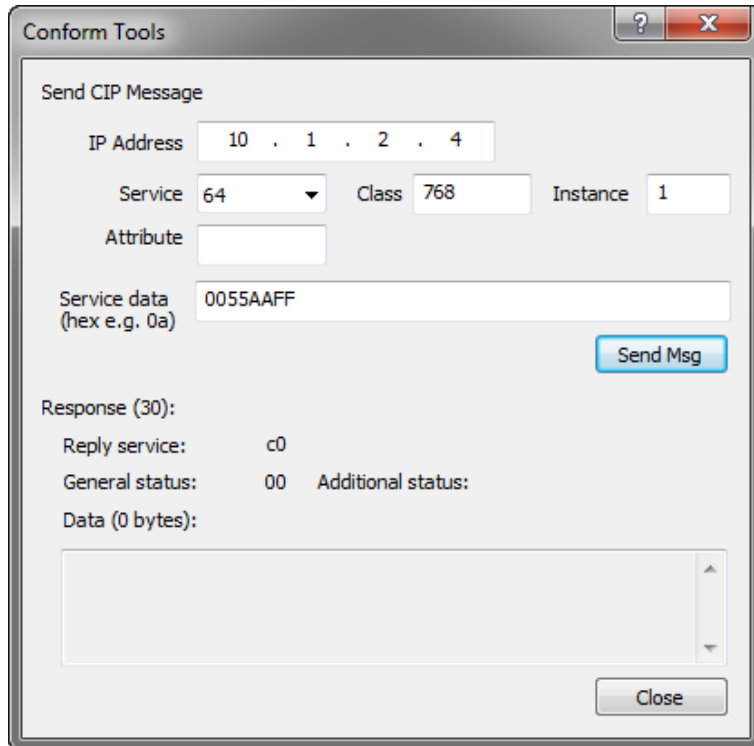


The screenshot shows a Windows-style dialog box titled "Conform Tools". Inside, there is a section labeled "Send CIP Message". The "IP Address" field is set to "10 . 1 . 2 . 4". The "Service" dropdown is set to "55", "Class" is "768", and "Instance" is "1". The "Attribute" field is empty. The "Service data (hex e.g. 0a)" field contains "2C010000". A blue "Send Msg" button is to the right of this field. Below the "Send CIP Message" section, there is a "Response (27):" section. It shows "Reply service:" as "b7", "General status:" as "00", and "Additional status:" as an empty field. Below this, it says "Data (0 bytes):" followed by an empty text area. A "Close" button is at the bottom right of the dialog.

PENKO EtherNet/IP protocol

Calibrate zero

Instance service 64 calibrates the zero point. For calibration actions a security code is needed. This is FFAA5500.



The screenshot shows a software window titled "Conform Tools" with a standard Windows-style title bar (minimize, maximize, close buttons). Inside the window, there is a section titled "Send CIP Message". Below this title, there are several input fields: "IP Address" with the value "10 . 1 . 2 . 4", "Service" with a dropdown menu showing "64", "Class" with the value "768", and "Instance" with the value "1". There is also an empty "Attribute" field. Below these, there is a "Service data (hex e.g. 0a)" field containing the value "0055AAFF". To the right of this field is a blue button labeled "Send Msg". Below the "Send CIP Message" section, there is a section titled "Response (30):". This section contains three lines of text: "Reply service: c0", "General status: 00 Additional status:", and "Data (0 bytes):". Below this text is a large, empty rectangular area, likely a text box or a placeholder for a response. At the bottom right of the window is a grey button labeled "Close".

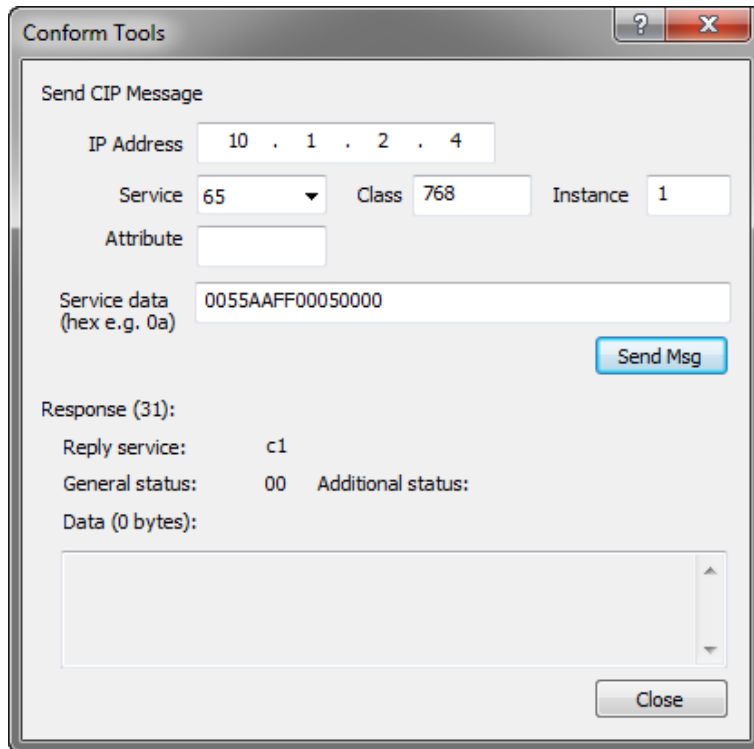
PENKO EtherNet/IP protocol

Calibrate span

Instance service 65 calibrates the span point. For calibration actions a security code is needed. This is FFAA5500.

Place the span weight after the security code.

Calibrate span at 1280 → 00 00 05 00



The screenshot shows the 'Conform Tools' application window. Inside, there is a 'Send CIP Message' section with the following fields: IP Address (10 . 1 . 2 . 4), Service (65), Class (768), Instance (1), and Attribute (empty). Below these is a 'Service data (hex e.g. 0a)' field containing '0055AAFF00050000'. A 'Send Msg' button is to the right of this field. Below the 'Send CIP Message' section is a 'Response (31):' section with fields for 'Reply service:' (c1), 'General status:' (00), and 'Additional status:'. Below these is a 'Data (0 bytes):' section with an empty text area. A 'Close' button is at the bottom right of the window.

PENKO EtherNet/IP protocol

Use register function

Instance service 80 executes the register functions. Enter the input values as parameter.

Each input value has to be entered as DINT[4]. Not used parameter has to be entered as 0.

Calibrating the zero point is function code 1. The input values are:

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
0x00 00 00 01	0x00 00 00 00	0x00 00 00 00	0x00 00 00 00

Conform Tools

Send CIP Message

IP Address: 10 . 1 . 2 . 4

Service: 80 Class: 768 Instance: 1

Attribute:

Service data (hex e.g. 0a): 01000000000000000000000000000000

Send Msg

Response (33):

Reply service: d0

General status: 00 Additional status:

Data (16 bytes):

01000000000000000000000000000000

Close

The reply shows the function code.

Reply:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
0x00 00 00 01	0x00 00 00 00	0x00 00 00 00	0x00 00 00 00

PENKO EtherNet/IP protocol

In case an error occurs, the error code is passed in the reply.

The screenshot shows a 'Conform Tools' window with the following fields and values:

- Send CIP Message**
 - IP Address: 10 . 1 . 2 . 4
 - Service: 80 (dropdown)
 - Class: 768
 - Instance: 1
 - Attribute: (empty)
 - Service data (hex e.g. 0a): 01000000000000000000000000000000
 - Send Msg button
- Response (37):**
 - Reply service: d0
 - General status: 00 Additional status:
 - Data (16 bytes): 01 00 3A 08 00 00 00 00 00 00 00 00 00 00 00 00
 - Close button

Reply:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
0x08 3A 00 01	0x00 00 00 00	0x00 00 00 00	0x00 00 00 00

Error code = 0x083A = 2106 = A/D reads all 1's, Hardware Overload on loadcell

PENKO EtherNet/IP protocol

Calibrating the gain point is function code 2. Parameter 2 is used to enter the weight.

Calibrate span at 1280 → 00 00 05 00

Request:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
0x00 00 00 02	0x00 00 05 00	0x00 00 00 00	0x00 00 00 00

The screenshot shows a window titled "Conform Tools" with a "Send CIP Message" section. The IP Address is set to 10.1.2.4, Service to 80, Class to 768, and Instance to 1. The Service data (hex e.g. 0a) is set to 02000000000050000000000000000000. A "Send Msg" button is visible. Below, the "Response (34):" section shows "Reply service: d0", "General status: 00", and "Additional status:". The "Data (16 bytes):" section displays the hex value 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00. A "Close" button is at the bottom right.

The reply shows the function code.

Reply:

Parameter 1	Parameter 2	Parameter 3	Parameter 4
0x00 00 00 02	0x00 00 00 00	0x00 00 00 00	0x00 00 00 00



About PENKO

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

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

Certifications

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

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



PENKO Professional Services

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

PENKO Alliances

PENKO's worldwide network: Australia, Belgium, 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

