PLUTO Gateway

User Manual

PROFIBUS GATE-P1/P2
DeviceNet GATE-D1/D2
CANopen GATE-C1/C2
Ethernet GATE-E1/E2

Revision history:

Version	Date	Change
1A	2006-04-20	First release
2A	2006-10-12	New functions on K-button. Update PROFIBUS (req/resp data, diagnostic data...). Update CANopen (EDS file, DIP-switch...). Update DeviceNet (EDS file, DIP-switch…).
2B	2007-04-19	Update information and update incorrect information.
3A	2007-12-10	Update description for Ethernet gateway GATE-E1. Update additional data for GATE-P1 and GATE-E1. Minor update in other part of the text.
4A	2008-06-16	Clarifications regarding Modbus TCP communication (GATE-E1). Update additional data to DeviceNet (GATE-D1).
4B	2008-08-07	Update numbering to two headings (missing), causing renumbering of all subsequent chapters. Updated references.
5A	2009-09-11	Update information about Profinet. Update information about additional data timeout. Update information about Gateway Node Number set by PLC.
6A	2010-11-22	Update CANopen with additional data/gateway node number. Uppdate with GATE-x2 version of the gateway.
7A	2011-05-19	Minor corrections in text. Updated tables for Standard blocks (For B42 AS-i).
8A	2011-05-30	Added recommendation about “Managed switch” for -E1/E2.
9A	2011-06-15	Added table for Global variables for B42 AS-i. Clarification regarding input “No” on User defined blocks. Minor corrections in text.
9B	2011-12-07	Edited picture page 60 (“Data – INT”)
9C	2011-12-19	Corrections page 41 (SW2 instead of SW1) and page 44 (112 instead of 113)
10A	2012-06-08	Updated with Pluto variables for B22, D20 and D45 New table for PROFIBUS diagnostic data
10B	2013-04-24	Minor corrections
10C	2014-03-06	Minor corrections
10D	2014-08-13	Minor corrections
10E	2014-08-22	Minor corrections + added Pluto O2

Reference:

No:	Text
1	Pluto Operating instructions, Hardware Pluto Programming manual
2	www.profibus.com Homepage for PROFIBUS and PROFINET.
3	www.odva.org Homepage for DeviceNet and EtherNet/IP (EIP).
4	www.can-cia.org Homepage for CANopen.
5	www.modbus.org Homepage for Modbus TCP.

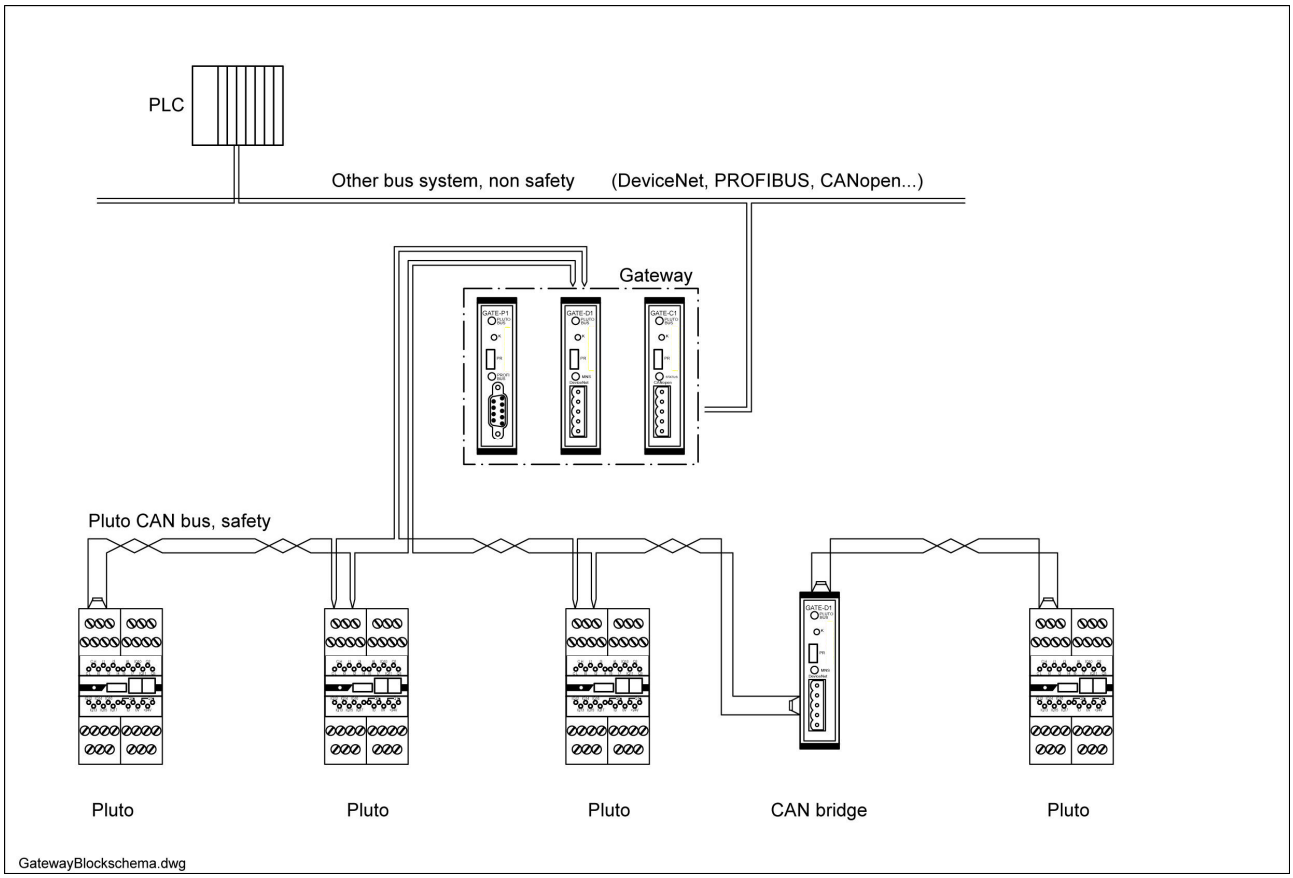
Table of contents:

1 General

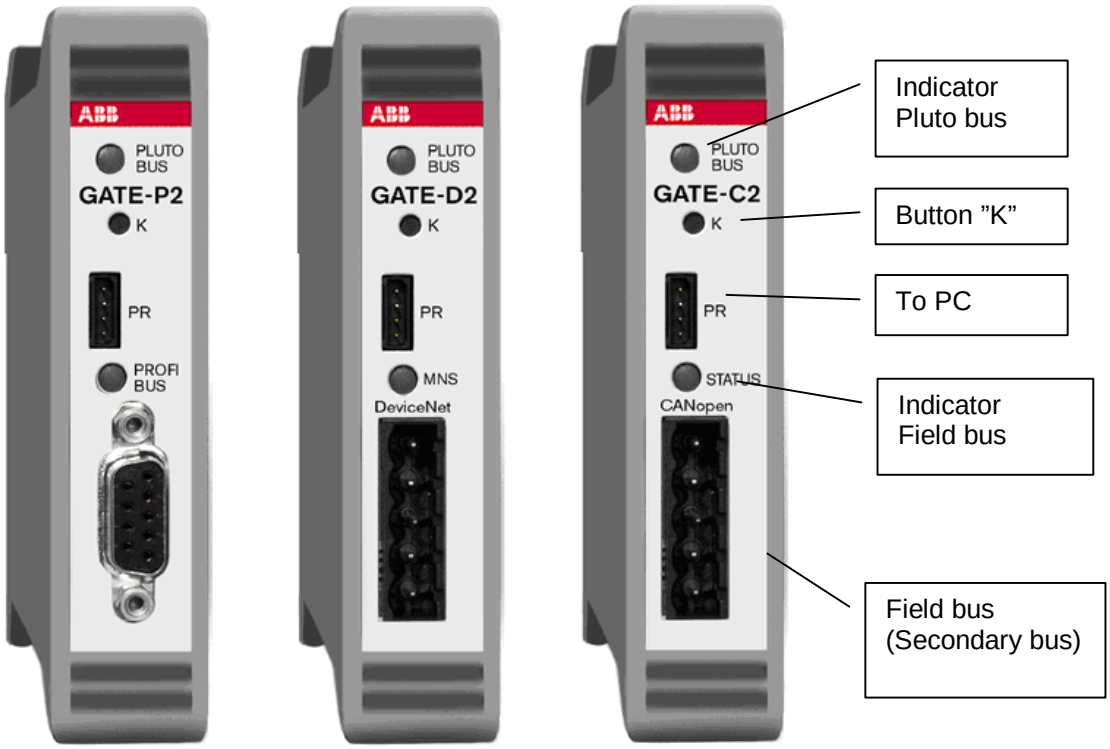
The gateways exist in two versions GATE-x1 and GATE-x2. The GATE-x2 will replace the GATE-x1 version. The GATE-x2 can be used to replace existing GATE-x1 in current installations.

The gateways are devices for transfer of data in both directions between the Pluto bus and other fieldbuses. They are made in these versions:
- GATE-P1/P2 for Profibus-DP.
- GATE-D1/D2 for DeviceNet.
- GATE-C1/C2 for CANopen.
- GATE-E1/E2 for Ethernet using Modbus TCP, EtherNet/IP (EIP) or PROFINET.

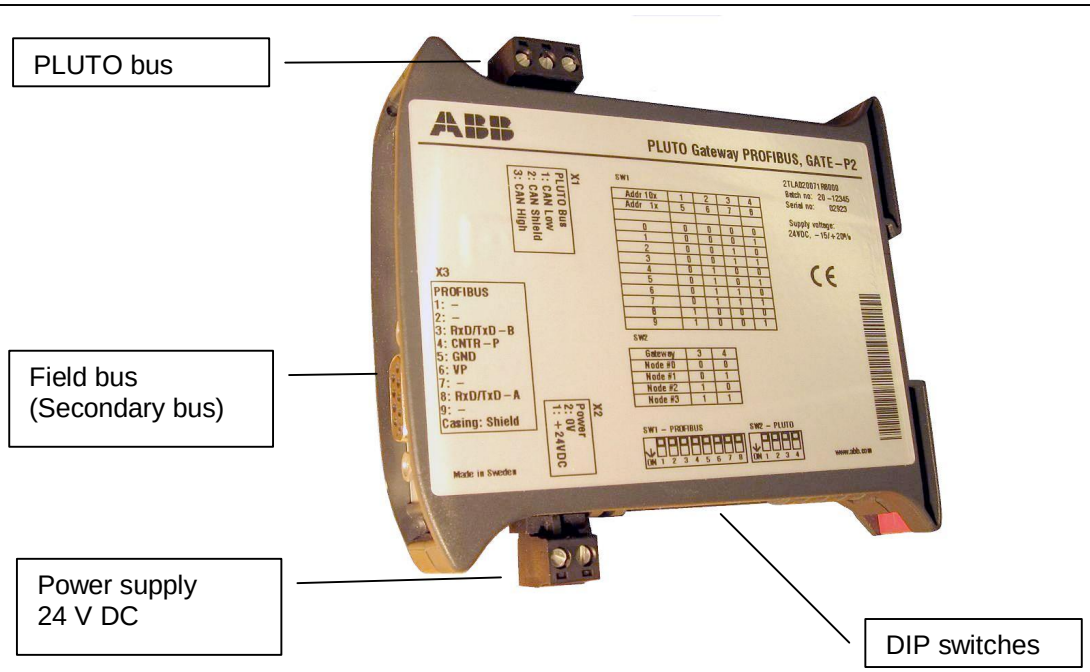
A second usage for GATE-D1/D2 and GATE-C1/C2 is as a CAN-bridge (repeater) which transfers CAN telegrams between two CAN-buses. This can be useful when long cables are needed. As CAN-bridge the usage is not limited to any special protocol such as the Pluto CAN bus system, but it can be used for most CAN bus systems.



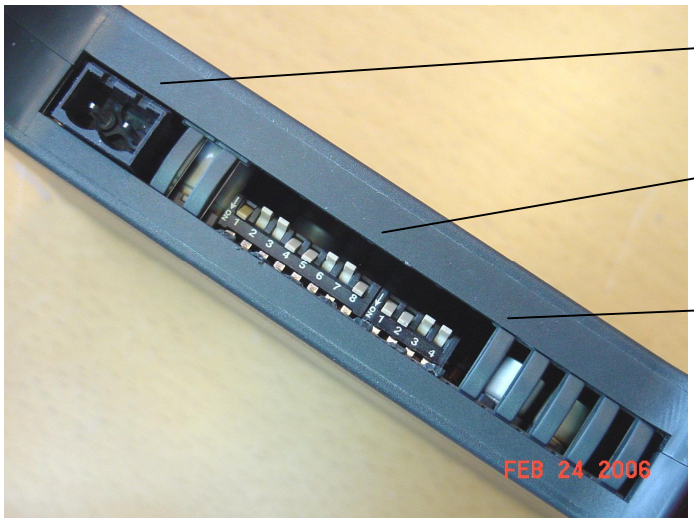
2 Hardware



Location of connectors, indicators, etc.



Location of connectors and DIP switches



Lower side of gateway, DIP switch location

2.1 Mounting

The gateway is mounted on a 35 mm DIN rail.

2.2 Power supply

The unit is powered with 24V DC. The connector is located on the lower side of the enclosure.

Terminal	Description
1	+24 V DC
2	0V

2.3 Galvanic insulation of buses

The CAN bus and PROFIBUS are galvanic isolated from each other and from the 24 VDC supply.

2.4 Bus cable screen

The bus connectors have terminals for connection of cable shield.

2.5 K-button

Via the K-button several functions can be started. If pressed during boot (power on) the gateway starts in monitor mode from where it is possible to load new operating system. It is also possible to select following functions by short[.] and long[-] press of the button (short press shall be less then 400 ms and long longer than 400 ms).

For example to send PLUTO reboot command, press the button short/long/long/short and the gateway will send the reboot command on the PLUTO bus.

Press	Function
.	Restart the PLUTO bus.
..	Restart the network bus (CANopen, DeviceNet or PROFIBUS).
.-.	Reset the gateway.
-…	Move the gateway into monitor mode.
.--.	Send PLUTO reboot command.

3 PLUTO bus

The Pluto bus is a CAN bus which means the connection shall follow the common rules for all CAN buses. For more information about PLUTO SAFETY PLC see **REF 1**.

3.1 Connection

The connector for the Pluto bus is located on the upper side of the enclosure (normal mounting). If the gateway is placed first or at the end of the bus a 120Ω end terminating resistor must be mounted.

PIN	Label	Description
1	CL	Pluto CAN-L
2	SE	Pluto CAN bus shield
3	CH	Pluto CAN-H

3.2 Baud rate detection, PLUTO bus

The gateway will auto detect the baud rate on the Pluto bus when there is traffic on the bus.

3.3 Indicator “PLUTO bus”

The indicator labeled Pluto bus indicates the status of the Pluto bus

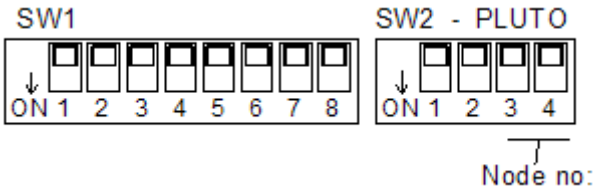
LED – Pluto bus		Description	Remark
Flashing GREEN/RED		Pluto bus baud rate search	When bus is not connected or no traffic on the bus.
GREEN short off flash		Pluto unit detected and baud rat is set In bridge function mode: Full operation	
Flashing GREEN 40 /60 (on/off)		Gateway in full operation Pluto bus is running and receiving SYNC/POLL/OUTPUT on the field bus. (Not for bridge function mode)	
Continuously RED		Fatal error detected	

3.4 Address on Pluto bus

3.4.1 Address set by DIP-switch

The gateway has an address switch for giving it an address on the Pluto bus, switch SW2. The address makes it possible to receive data to the Pluto bus from up to 16 different gateways.

Note: The address setting is important to differ between several gateways when “Data to Pluto” is used.



Address is set according to table below.

SW2:3	SW2:4	Function
0 (OFF)	0 (OFF)	Node Address 0
0 (OFF)	1 (ON)	Node Address 1
1 (ON)	0 (OFF)	Node Address 2
1 (ON)	1 (ON)	Node Address 3

3.4.2 Address set by PLC

The gateway address can also be set via parameter from PLC. By setting gateway address from PLC it's possible to address up to 16 gateways, compare with only 4 via DIP-switch.

The parameter setting shall be values according to table below. The default value is 0 which give that the address is read from DIP-switch. Note that if the DIP-switch is changed then the gateway will use address according to DIP-switch until overwritten by PLC.

Value	Function
0 (default)	Address read from DIP-switch
1	Node Address 0
2	Node Address 1
3	Node Address 2
4	Node Address 3
5	Node Address 4
6	Node Address 5
7	Node Address 6
8	Node Address 7
9	Node Address 8
10	Node Address 9
11	Node Address 10
12	Node Address 11
13	Node Address 12
14	Node Address 13
15	Node Address 14
16	Node Address 15

Note: To use Gateway Node Address higher than 7 the Pluto may need new OS version.

4 Data to/from Pluto

This chapter will describe the different type of data sent to/from Pluto via the gateway. It will also be the reference chapter regarding encoding of the data.

How much data, from how many and from which Pluto units is selected in different ways for each type of gateway (PROFIBUS, DeviceNet, CANopen and Ethernet).

4.1 Pluto Status

The size of this module is 4 bytes or 2 words. These data contain information about which Pluto units that are active on the Pluto bus. When a Pluto is active the corresponding bit is set to “1”.

The coding of the status data in byte is,

Byte	MSB							LSB
0	Pluto 7	Pluto 6	Pluto 5	Pluto 4	Pluto 3	Pluto 2	Pluto 1	Pluto 0
1	Pluto 15	Pluto 14	Pluto 13	Pluto 12	Pluto 11	Pluto 10	Pluto 9	Pluto 8
2	Pluto 23	Pluto 22	Pluto 21	Pluto 20	Pluto 19	Pluto 18	Pluto 17	Pluto 16
3	Pluto 31	Pluto 30	Pluto 29	Pluto 28	Pluto 27	Pluto 26	Pluto 25	Pluto 24

4.2 Global Data from Pluto

When selected, the global Pluto variables are always transferred. There are 32 global Pluto variables from each Pluto and they are always available on the Pluto bus, 1024 for a full net with 32 Pluto. All variables are bit variables.

The global Pluto variables are:

Ix.0 – Ix.17	Inputs (16)
Qx.0 – Qx.3	Safety outputs (4)
GMx.0 – GMx.11	Global memories (12)

(x = Pluto node no)

The size of this module is 4 bytes or 2 words. The coding of the data from a Pluto is according to the table below.

The coding of the Pluto variables for A20 and Double family in byte is,

Byte	MSB							LSB
0	Ix.7	Ix.6	Ix.5	Ix.4	Ix.3	Ix.2	Ix.1	Ix.0
1	Ix.17	Ix.16	Ix.15	Ix.14	Ix.13	Ix.12	Ix.11	Ix.10
2	GMx.3	GMx.2	GMx.1	GMx.0	Qx.3	Qx.2	Qx.1	Qx.0
3	GMx.11	GMx.10	GMx.9	GMx.8	GMx.7	GMx.6	GMx.5	GMx.4

x is Pluto node number.

The coding of the status variables for Pluto AS-i family in byte is,

Byte	MSB							LSB
0	ASIx.7	ASIx.6	ASIx.5	ASIx.4	ASIx.3	ASIx.2	ASIx.1	Ix.0
1	ASIx.15	ASIx.14	ASIx.13	ASIx.12	ASIx.11	ASIx.10	ASIx.9	ASIx.8
2	GMx.3	GMx.2	GMx.1	GMx.0	Qx.3	Qx.2	Qx.1	Qx.0
3	GMx.11	GMx.10	GMx.9	GMx.8	GMx.7	GMx.6	GMx.5	GMx.4

x is Pluto node number and ASIx.y is the safety node y.

The coding of the Pluto variables for Pluto B42 AS-i in byte is,

Byte	MSB							LSB
0	GMx.3	GMx.2	GMx.1	GMx.0	Ix.3	Ix.2	Ix.1	Ix.0
1	GMx.11	GMx.10	GMx.9	GMx.8	GMx.7	GMx.6	GMx.5	GMx.4
2	GMx.19	GMx.18	GMx.17	GMx.16	GMx.15	GMx.14	GMx.13	GMx.12
3	GMx.27	GMx.26	GMx.25	GMx.24	GMx.23	GMx.22	GMx.21	GMx.20

x is Pluto node number.

4.3 Additional Data from Pluto

Additional Data is currently possible to use with:

- PROFIBUS (**GATE-P1/P2**) with software version from 2.0 and GSD file revision 2.0.
- DeviceNet (**GATE-D1/D2**) with software version from 2.0 and updated EDS-file.
- CANopen (**GATE-C1/C2**) with software version from 2.0
- Ethernet (**GATE-E1/E2**) with software version from 1.2.

Every Pluto on the Pluto-bus can send out additional data blocks where each block has:

- The Pluto node number.
- An IO-type number (for user block a user identity number).
  - 0 (zero) data is not used.
  - 1-99 are user defined numbers used at the additional data blocks in PLC code.
  - ≥100 are standard additional data types (see tables below).
  - 111 are IO-type for Pluto global data (used in GATE-D1/D2 and GATE-C1/C2).
- 32 bit of data according to IO-type.

This additional data configuration is implemented in different way depending on gateway:

- PROFIBUS (**GATE-P1/P2**)

  For PROFIBUS the additional data will increase number of modules with 32 additional data areas, e.g. the gateway will be able to handle a selection of 32 Pluto global areas and 32 additional data areas. Note that **all can't** be used simultaneously because of to much data, if that amount of data is needed an extra gateway is needed. For each additional data module there is configuration data for Pluto node number and IO-type.

- DeviceNet (**GATE-D1/D2**) and CANopen (**GATE-C1/C2**)

  For DeviceNet and CANopen the total data up to PLC is fixed to 32 data areas. Each of these data areas can be allocated to additional data or Pluto global data, e.g. the additional data and Pluto global data share the same data areas. For each data area the gateway can be configured with data regarding Pluto node number and IO-type number. The IO-type number can be set to be Pluto global data, see 6.4.1.

  Configuration can be done via terminal connection to the gateway or via explicit/SDO message from the PLC system.

  Note: For DeviceNet when using additional data the expected node bit value shall be zero e.g. no node shall be enabled in this data.

- Ethernet (**GATE-E1/E2**).

  For Ethernet gateway both Pluto global data and additional data is available simultaneously in different memory locations for both Modbus TCP and EtherNet/IP. For configuration of the additional data the gateway can be configured via terminal connection or via messages from Ethernet PLC. For each additional data area there is configuration for Pluto node number and IO-type.

Note: It is possible to allocate several Additional Data Areas with the same IO type data from the same Pluto. In this case only the first allocated Additional Data Area will get the correct data from selected Pluto.

4.3.1 Terminal configuration, GATE-E1/E2

For Ethernet gateway the configuration can be done via terminal setting using the “addc, adds, add and bw” commands. Below shows an example of configuration of additional area 2 which shall retrieve data from Pluto 10 and the data shall be of IO type 103 which will be the “ToGateway_ASi_16_31_Safe” block se below. These settings are stored in internal EEPROM.

Note that in each Pluto there is needed to be “ToGateway_X” function block in the PLC code, in the example below the block “ToGateway_ASi_16_31_Safe” is needed in Pluto 10.

```
// Setup of Additional Data.
e_gw> adds
Additional Data Area [0] : 2
Data from Pluto [0] : 10
IO type :
-    0 = Not used
 - 1-99 = User block
 -  100 = Error Code
 -  101 = B46 I20-I47
 -  102 = ASi 16-31 Safe
 -  103 = ASi  1- 3 NonSafe In
 -  104 = ASi  4- 7 NonSafe In
 -  105 = ASi  8-11 NonSafe In
 -  106 = ASi 12-15 NonSafe In
 -  107 = ASi 16-19 NonSafe In
 -  108 = ASi 20-23 NonSafe In
 -  109 = ASi 24-27 NonSafe In
 -  110 = ASi 28-31 NonSafe In
Select IO type [0] : 102
EEPROM write [3].
Configuration of additional data 2 done.
e_gw>

// Check input of Additional Data area 2
e_gw> add02
ADD 02.02  32767
e_gw>

// Check current configuration.
// A * before ‘10’ indicates active receive of data.
e_gw> bw
...
----------------------------
PLC OUTPUT DATA :
  Enabled To PLUTO package 0-3 : - - - -, Timeout 0 ms
ADDITIONAL DATA CONFIGURATION :
 Area Pluto IO-type
  02   *10  ASIsafe
----------------------------
e_gw>

// Clear all setting of Additional Data.
e_gw> addc
Clear Additional Data setting [Yes/No] ?  YES
EEPROM write [2].
Done!
e_gw>
```

4.3.2 Terminal configuration, GATE-D1/D2 and GATE-C1/C2

For Ethernet gateway the configuration can be done via terminal setting using the “cs” and bw” commands when DIP-switch is set to PROG mode. For more information see 6.4.1.

```
// Setup of Additional Data.
dnet_gw> cs

Input Assembly Instance :
0 : Status Only  [100]
1 : Data   Only  [101]
2 : Status/Data  [102]
Select [1] : 1
Output Assembly Instance :
0 : No Data       [112]
1 : To Pluto Data [113]
Select [0] : 0
IO Configuration way :
0 : Expected Node Configuration [Only global data]
1 : Additional Data Configuration [Clear current configuration]
2 : Additional Data Configuration [Keep  current configuration]
Select [0]: 1
Area 00 data from PLUTO 00  24
Area 00 data IO type 000    111
Area 01 data from PLUTO 00  24
Area 01 data IO type 000    100
Area 02 data from PLUTO 00  5
Area 02 data IO type 000    111
Area 03 data from PLUTO 00  5
Area 03 data IO type 000    1
Area 04 data from PLUTO 00
...
Area 31 data from PLUTO 00
Area 31 data IO type 000
Enable To PLUTO package 0  [N] ?
Enable To PLUTO package 1  [N] ?
Enable To PLUTO package 2  [N] ?
Enable To PLUTO package 3  [N] ?
To PLUTO Timeout [0 ms] :
To PLUTO update time [100 ms] :
Save the new configuration [y/n]  YES

EEPROM write [28].

e_gw>

// Check current configuration.
// A * before ‘24’ and ‘05’ indicates active receive of data.
dnet_gw> bw
----------------------------
DeviceNet bus status.
----------------------------
Node number : 3 [0x3]
Bus speed   : 125 kbits
Bus power   : VALID
Bus status  : OFFLINE
----------------------------
Input  assembly 1 = PLUTO Data Only   [102]
 Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type
  00   *24  GLOBAL  |  01    24  ErrCode |  02   *05  GLOBAL  |  03    05  USER:01
Output assembly 1 = To PLUTO Data     [113]
 Enabled To PLUTO package 0-3 : - - - -, Timeout 0 ms, Update 100 ms.
----------------------------
dnet_gw>
```

4.3.3 Layout of additional data

All blocks which can be used in the Pluto PLC program for sending additional data are listed below.

> **Note:** For the user defined blocks each block in each Pluto must be allocated a unique number between 1 and 99 (on input “No”) to identify the data block. This number is then used to identify the block in the receiving field bus system.

The standard blocks have defined data.

4.3.3.1 User defined blocks

User defined “ToGateway_User_A” (ToGateway_UserNumber_x),

Byte	MSB							LSB
0	Reg_0.7	Reg_0.6	Reg_0.5	Reg_0.4	Reg_0.3	Reg_0.2	Reg_0.1	Reg_0.0
1	Reg_0.15	Reg_0.14	Reg_0.13	Reg_0.12	Reg_0.11	Reg_0.10	Reg_0.9	Reg_0.8
2	Reg_1.7	Reg_1.6	Reg_1.5	Reg_1.4	Reg_1.3	Reg_1.2	Reg_1.1	Reg_1.0
3	Reg_1.15	Reg_1.14	Reg_1.13	Reg_1.12	Reg_1.11	Reg_1.10	Reg_1.9	Reg_1.8

Unique user number (x) set in block.

User defined “ToGateway_User_B” (ToGateway_UserNumber_x),

Byte	MSB							LSB
0	Reg_0.7	Reg_0.6	Reg_0.5	Reg_0.4	Reg_0.3	Reg_0.2	Reg_0.1	Reg_0.0
1	Reg_0.15	Reg_0.14	Reg_0.13	Reg_0.12	Reg_0.11	Reg_0.10	Reg_0.9	Reg_0.8
2	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1	Bit_0
3	Pluto Error Code							

Unique user number (x) set in block.

User defined “ToGateway_User_C” (ToGateway_UserNumber_x),

Byte	MSB							LSB
0	Reg_0.7	Reg_0.6	Reg_0.5	Reg_0.4	Reg_0.3	Reg_0.2	Reg_0.1	Reg_0.0
1	Reg_0.15	Reg_0.14	Reg_0.13	Reg_0.12	Reg_0.11	Reg_0.10	Reg_0.9	Reg_0.8
2	Bit_7	Bit_6	Bit_5	Bit_4	Bit_3	Bit_2	Bit_1	Bit_0
3	Bit_15	Bit_14	Bit_13	Bit_12	Bit_11	Bit_10	Bit_9	Bit_8

Unique user number (x) set in block.

4.3.3.2 Standard blocks

Standard “ToGateway_ErrorCode” (IO-type number 100, 0x64),

Byte	MSB							LSB
0	-	-	-	-	-	-	-	-
1	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-
3	Pluto Error Code							

The ‘-‘ character indicate undefined value.

Standard “ToGateway_B46_I20_I47” (IO-type number 101, 0x65),

Byte	MSB							LSB
0	Ix.27	Ix.26	Ix.25	Ix.24	Ix.23	Ix.22	Ix.21	Ix.20
1	Ix.37	Ix.36	Ix.35	Ix.34	Ix.33	Ix.32	Ix.31	Ix.30
2	Ix.47	Ix.46	Ix.45	Ix.44	Ix.43	Ix.42	Ix.41	Ix.40
3	Pluto Error Code							

Standard “ToGateway_ASi_16_31_Safe” (IO-type number 102, 0x66),

Byte	MSB							LSB
0	Ix.13*	Ix.12*	Ix.11*	Ix.10*	Ix.3*	Ix.2*	Ix.1*	-
1	ASIx.23	ASIx.22	ASIx.21	ASIx.20	ASIx.19	ASIx.18	ASIx.17	ASIx.16
2	ASIx.31	ASIx.30	ASIx.29	ASIx.28	ASIx.27	ASIx.26	ASIx.25	ASIx.24
3	Pluto Error Code							

*For B42 AS-i: Undefined
ASIx.y is safety slave y from Pluto AS-i unit (x is Pluto node number).
The ‘-‘ character indicates undefined value.

Standard “ToGateway_ASi_1_3_NonSafe_In” (IO-type number 103, 0x67),

Byte	MSB							LSB
0	-	-	-	-	-	-	-	-
1	Ax.1B.4	Ax.1B.3	Ax.1B.2	Ax.1B.1	Ax.1.4	Ax.1.3	Ax.1.2	Ax.1.1
2	Ax.2B.4	Ax.2B.3	Ax.2B.2	Ax.2B.1	Ax.2.4	Ax.2.3	Ax.2.2	Ax.2.1
3	Ax.3B.4	Ax.3B.3	Ax.3B.2	Ax.3B.1	Ax.3.4	Ax.3.3	Ax.3.2	Ax.3.1

ASIx.<slave>.<bit> from Pluto x.
The ‘-‘ character indicate undefined value.

Standard “ToGateway_ASi_4_7_NonSafe_In” (IO-type number 104, 0x68),

Byte	MSB							LSB
0	Ax.4B.4	Ax.4B.3	Ax.4B.2	Ax.4B.1	Ax.4.4	Ax.4.3	Ax.4.2	Ax.4.1
1	Ax.5B.4	Ax.5B.3	Ax.5B.2	Ax.5B.1	Ax.5.4	Ax.5.3	Ax.5.2	Ax.5.1
2	Ax.6B.4	Ax.6B.3	Ax.6B.2	Ax.6B.1	Ax.6.4	Ax.6.3	Ax.6.2	Ax.6.1
3	Ax.7B.4	Ax.7B.3	Ax.7B.2	Ax.7B.1	Ax.7.4	Ax.7.3	Ax.7.2	Ax.7.1

ASIx.<slave>.<bit> from Pluto x.

Standard “ToGateway_ASi_8_11_NonSafe_In” (IO-type number 105, 0x69),

Byte	MSB							LSB
0	Ax.8B.4	Ax.8B.3	Ax.8B.2	Ax.8B.1	Ax.8.4	Ax.8.3	Ax.8.2	Ax.8.1
1	Ax.9B.4	Ax.9B.3	Ax.9B.2	Ax.9B.1	Ax.9.4	Ax.9.3	Ax.9.2	Ax.9.1
2	Ax.10B.4	Ax.10B.3	Ax.10B.2	Ax.10B.1	Ax.10.4	Ax.10.3	Ax.10.2	Ax.10.1
3	Ax.11B.4	Ax.11B.3	Ax.11B.2	Ax.11B.1	Ax.11.4	Ax.11.3	Ax.11.2	Ax.11.1

ASIx.<slave>.<bit> from Pluto x.

Standard “ToGateway_ASi_12_15_NonSafe_In” (IO-type number 106, 0x6A),

Byte	MSB							LSB
0	Ax.12B.4	Ax.12B.3	Ax.12B.2	Ax.12B.1	Ax.12.4	Ax.12.3	Ax.12.2	Ax.12.1
1	Ax.13B.4	Ax.13B.3	Ax.13B.2	Ax.13B.1	Ax.13.4	Ax.13.3	Ax.13.2	Ax.13.1
2	Ax.14B.4	Ax.14B.3	Ax.14B.2	Ax.14B.1	Ax.14.4	Ax.14.3	Ax.14.2	Ax.14.1
3	Ax.15B.4	Ax.15B.3	Ax.15B.2	Ax.15B.1	Ax.15.4	Ax.15.3	Ax.15.2	Ax.15.1

ASIx.<slave>.<bit> from Pluto x.

Standard “ToGateway_ASi_16_19_NonSafe_In” (IO-type number 107, 0x6B),

Byte	MSB							LSB
0	Ax.16B.4	Ax.16B.3	Ax.16B.2	Ax.16B.1	Ax.16.4	Ax.16.3	Ax.16.2	Ax.16.1
1	Ax.17B.4	Ax.17B.3	Ax.17B.2	Ax.17B.1	Ax.17.4	Ax.17.3	Ax.17.2	Ax.17.1
2	Ax.18B.4	Ax.18B.3	Ax.18B.2	Ax.18B.1	Ax.18.4	Ax.18.3	Ax.18.2	Ax.18.1
3	Ax.19B.4	Ax.19B.3	Ax.19B.2	Ax.19B.1	Ax.19.4	Ax.19.3	Ax.19.2	Ax.19.1

ASIx.<slave>.<bit> from Pluto x.

Standard “ToGateway_ASi_20_23_NonSafe_In” (IO-type number 108, 0x6C),

Byte	MSB							LSB
0	Ax.20B.4	Ax.20B.3	Ax.20B.2	Ax.20B.1	Ax.20.4	Ax.20.3	Ax.20.2	Ax.20.1
1	Ax.21B.4	Ax.21B.3	Ax.21B.2	Ax.21B.1	Ax.21.4	Ax.21.3	Ax.21.2	Ax.21.1
2	Ax.22B.4	Ax.22B.3	Ax.22B.2	Ax.22B.1	Ax.22.4	Ax.22.3	Ax.22.2	Ax.22.1
3	Ax.23B.4	Ax.23B.3	Ax.23B.2	Ax.23B.1	Ax.23.4	Ax.23.3	Ax.23.2	Ax.23.1

ASIx.<slave>.<bit> from Pluto x.

Standard “ToGateway_ASi_24_27_NonSafe_In” (IO-type number 109, 0x6D),

Byte	MSB							LSB
0	Ax.24B.4	Ax.24B.3	Ax.24B.2	Ax.24B.1	Ax.24.4	Ax.24.3	Ax.24.2	Ax.24.1
1	Ax.25B.4	Ax.25B.3	Ax.25B.2	Ax.25B.1	Ax.25.4	Ax.25.3	Ax.25.2	Ax.25.1
2	Ax.26B.4	Ax.26B.3	Ax.26B.2	Ax.26B.1	Ax.26.4	Ax.26.3	Ax.26.2	Ax.26.1
3	Ax.27B.4	Ax.27B.3	Ax.27B.2	Ax.27B.1	Ax.27.4	Ax.27.3	Ax.27.2	Ax.27.1

ASIx.<slave>.<bit> from Pluto x.

Standard “ToGateway_ASi_28_31_NonSafe_In” (IO-type number 110, 0x6E),

Byte	MSB							LSB
0	Ax.28B.4	Ax.28B.3	Ax.28B.2	Ax.28B.1	Ax.28.4	Ax.28.3	Ax.28.2	Ax.28.1
1	Ax.29B.4	Ax.29B.3	Ax.29B.2	Ax.29B.1	Ax.29.4	Ax.29.3	Ax.29.2	Ax.29.1
2	Ax.30B.4	Ax.30B.3	Ax.30B.2	Ax.30B.1	Ax.30.4	Ax.30.3	Ax.30.2	Ax.30.1
3	Ax.31B.4	Ax.31B.3	Ax.31B.2	Ax.31B.1	Ax.31.4	Ax.31.3	Ax.31.2	Ax.31.1

ASIx.<slave>.<bit> from Pluto x.

Standard “GLOBAL DATA” (IO-type number 111, 0x6F),

Byte	MSB							LSB
0	See 4.2							
1								
2								
3								

Used for DeviceNet (GATE_D1/D2) and CANopen (GATE-C1/C2) allocation of Pluto IO Data Area.

Standard “ToGateway_B42_ASi_I20_I47” (IO-type number 112, 0x70),

Byte	MSB							LSB
0	Ix.27	Ix.26	Ix.25	Ix.24	Ix.23	Ix.22	Ix.21	Ix.20
1	Ix.37	Ix.36	Ix.35	Ix.34	Ix.33	Ix.32	Ix.31	Ix.30
2	Ix.47	Ix.46	Ix.45	Ix.44	Ix.43	Ix.42	Ix.41	Ix.40
3	Pluto Error Code							

Standard “ToGateway_ASi_1_15_Safe” (IO-type number 113, 0x71),

Byte	MSB							LSB
0	Ix.17	Ix.16	Ix.15	Ix.14	Ix.13	Ix.12	Ix.11	Ix.10
1	ASIx.7	ASIx.6	ASIx.5	ASIx.4	ASIx.3	ASIx.2	ASIx.1	0
2	ASIx.15	ASIx.14	ASIx.13	ASIx.12	ASIx.11	ASIx.10	ASIx.9	ASIx.8
3	Pluto Error Code							

ASIx.y is safety slave y from Pluto AS-i unit (x is Pluto node number).
The ‘-‘ character indicates undefined value.

4.3.4 Programming in Pluto PLC

4.3.4.1 Function block library

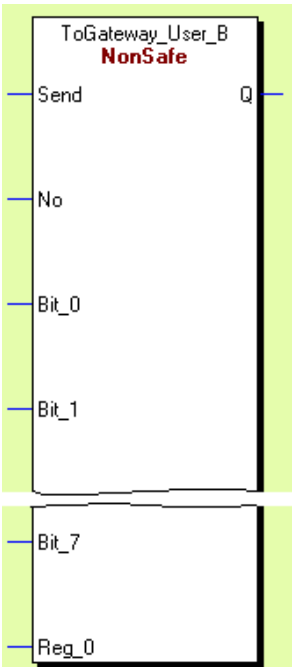
To use the function “Additional data from Pluto” the function block library “Ext01_1.fps” must be selected. The library contains all blocks listed above (4.3.3.1 and 4.3.3.2).

4.3.4.2 Use of the function blocks

As described before there are standard blocks and user defined blocks. The standard blocks have a fixed content as for example “ToGateway_B46_I20_I47” transmitting the local inputs and error code of a Pluto B46-6. The user defined blocks have inputs for bit variables (M, I, Q..) and registers which makes it possible for the user to compose his own telegram.

Example of a standard block. Transmission of AS-i slave inputs 16-31 and error code.

Each block generates a CAN telegram on the Pluto bus. In order to control and limit bus load and execution time all blocks have an input named “Send”. When the input conditions for “Send” are true (1) the block transmits a telegram. All blocks have also an output “Q” which is high (1) by transmission and can for example be used for inhibiting other blocks to transmit.



Example of a user defined block: Transmission of 8 bits and one register.
***Note:** Each block in each Pluto shall have its unique number in input “No”.*

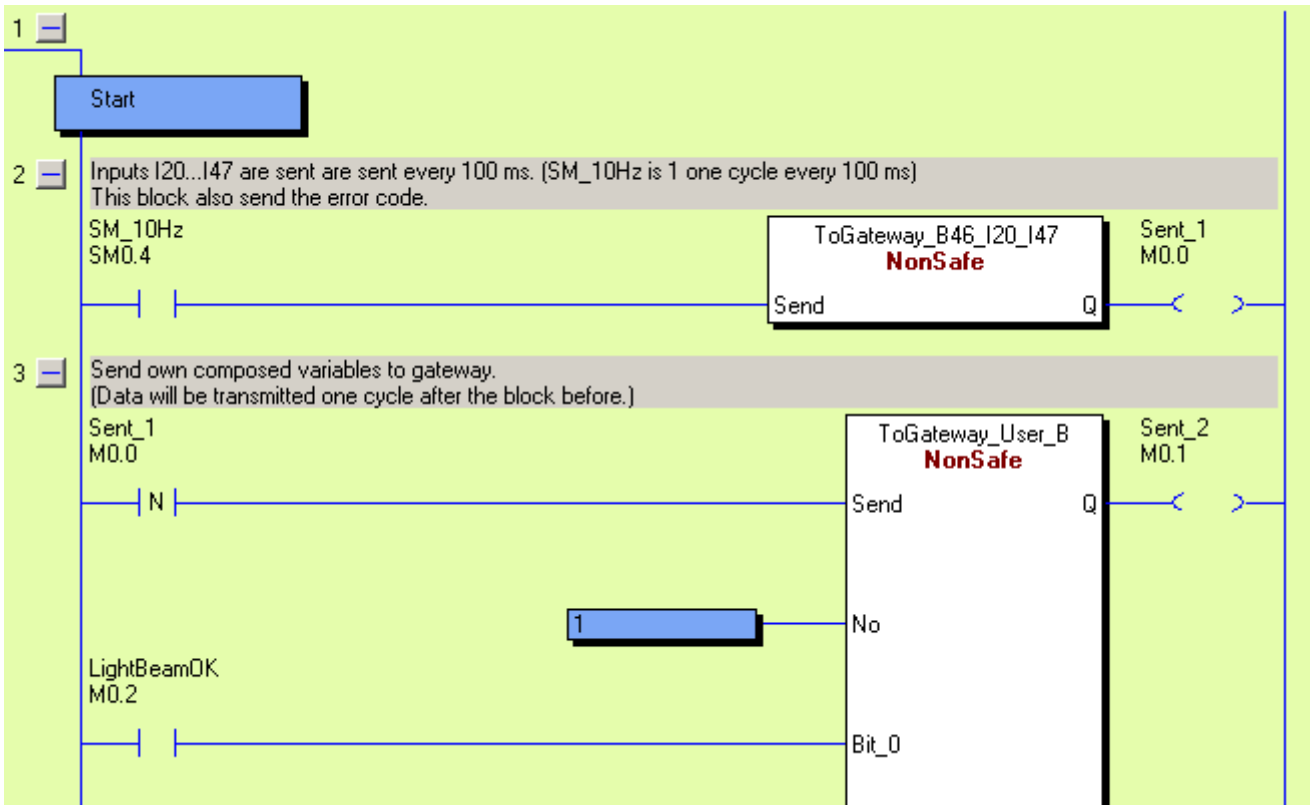
If “Send” is continuously activated a CAN message is transmitted every 10 ms which of course will give the best performance in reaction time. If there is need for limiting the transmission depends on how many Pluto units there are on the bus and how many of these blocks are used.

Note: Pluto can only send 4 telegrams every PLC cycle.

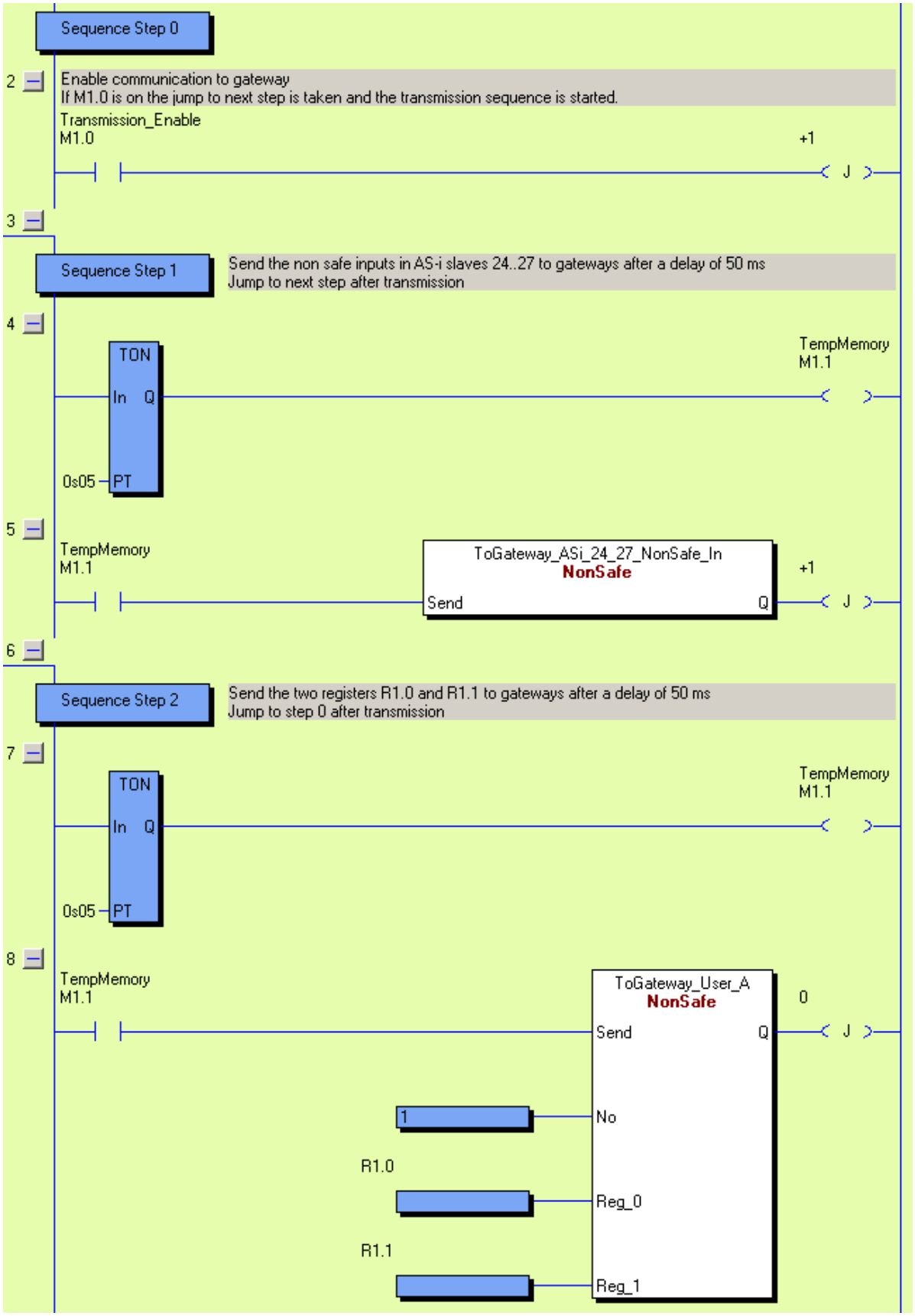
Note: The gateway has **300 ms** timeout on additional data. Therefore data from Pluto shall be sent with maximum 250 ms interval when if for example TON is used (see example below).

4.3.4.3 Example of usage in Pluto program

The following two examples show how transmission rate can be controlled in order to limit the CAN bus load and program execution time in Pluto.



Example 1: Transmission of local IO:s in a Pluto B46-6 user defined block.
The input "Send" in the first block is connected to the system memory for 10Hz to decrease the CAN bus load to 1 telegram / 100ms.
The second block will be transmitted one PLC cycle after the first because "Send" is connected to negative edge of Sent_1.



Example for transmission from a Pluto AS-i sequence. The transmission can be enabled by memory M0.0 in sequence 0 then a telegram will be transmitted every 50 ms.
This is a recommended method when a lot of blocks are used since it limits the CAN bus load and the Pluto do not need to execute the code in inactive sequence steps.

4.4 Data to Pluto

A gateway can totally transfer 64 bit variables and 8 registers from other field buses to the Pluto bus. The area “Data to Pluto” is divided into four packets each with16 bit variables and two registers and is organized as below table.

To Pluto Area Packet	Type	Data
0	Bit (16 bits)	Bit variables 0…15
	Register (16 bits)	Register 0
	Register (16 bits)	Register 1
1	Bit (16 bits)	Bit variables 0…15
	Register (16 bits)	Register 0
	Register (16 bits)	Register 1
2	Bit (16 bits)	Bit variables 0…15
	Register (16 bits)	Register 0
	Register (16 bits)	Register 1
3	Bit (16 bits)	Bit variables 0…15
	Register (16 bits)	Register 0
	Register (16 bits)	Register 1

4.4.1 Enable bit

A PLC system on the field bus can enable the usage of 0 to 4 of the packets for Data to Pluto, for example enable the gateway to transfer the data in packet 0 and 1 to the units on the Pluto bus. The gateway then transmits one packet in one CAN telegram.

4.4.2 Cyclic transmission time

The gateway will transmit each data package cyclically every 100 ms to the Pluto bus. For some gateways (see note below) this cycle time can be changed by the PLC system if needed. The time interval is 4 – 255 ms with a default value of 100 ms.

Note: Low cycle time will load the Pluto bus more.
Therefore this value shall not be set lower than needed and with consideration of the load of the Pluto bus.

4.4.3 Timeout time

A PLC system on the field bus can also set a timeout value time in the range of 0 – 60000 ms. The default value is **0** which is the same as **no timeout**. If the gateway does not receive data telegrams from the field bus within the timeout time the data will be cleared and the gateway will transmit “0”.

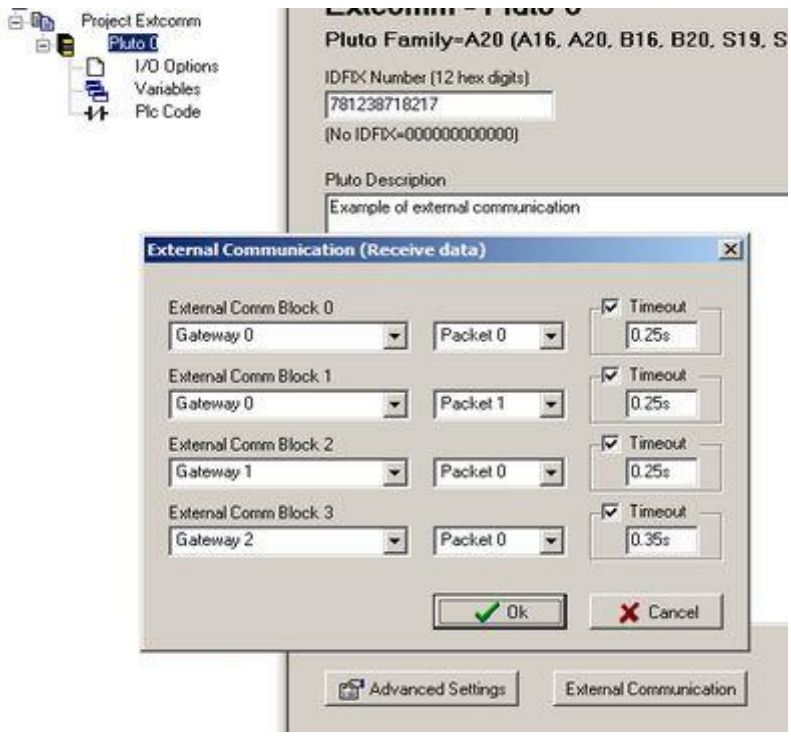
Note: For GATE-E1 there is a limitation of the timeout.
It shall be 0 or between 1000 ms and 60000 ms.

4.5 In PLUTO - Reception of external data from gateway.

A PLUTO has a corresponding data area for external communication divided in four data blocks which enables each PLUTO unit to receive four packets of data from different sources e.g. four different gateways. A data block in a PLUTO is programmed to receive data from a certain gateway address (0-3) and a certain packet number (0-3).

4.5.1 Set up in PLUTO for reception

For each PLUTO which shall receive data from a gateway, a setup must be made to decide from where the data comes. If the same gateway shall send to more than one block it must send in two different packets. (One packet is one CAN telegram).



Example of setup in Pluto for reception of external data from three different gateways.

Note: The timeout shall be greater than the gateway cycle time which has a default value of 100 ms, see chapter 4.4.2.

4.5.2 Addressing of external data in Pluto

In the PLUTO the variables are numbered as in following table.

Data block	Data in Pluto
External Comm Block 0	Data bit 0…15
	Reg 0
	Reg 1
External Comm Block 1	Data bit 16…31
	Reg 2
	Reg 3
External Comm Block 2	Data bit 32…47
	Reg 4
	Reg 5
External Comm Block 3	Data bit 48…63
	Reg 6
	Reg 7

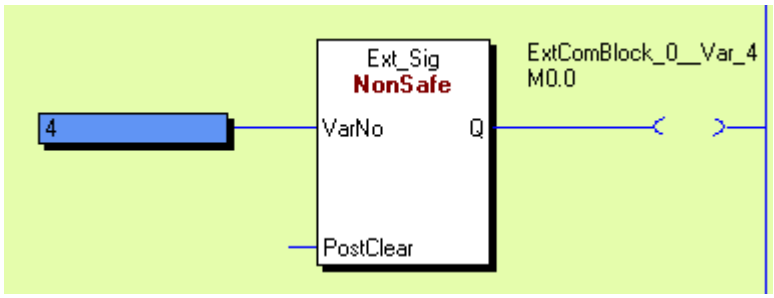
4.5.3 Connection of external variables in PLC code

When the setup in “External Communication” is made the data can be used in the PLC code. Then there are function blocks for linking the variables to the ordinary PLC variables M, Q, GM or R. The blocks are available in the library “Ext01.fps” which must be selected.

The function library Ext01.fps must be selected

4.5.3.1 Function block ”Ext_Sig”

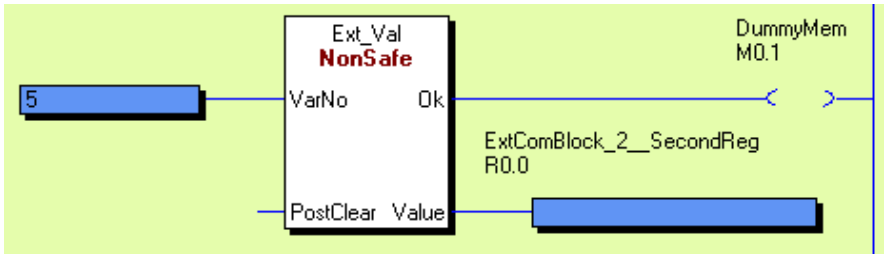
The function block Ext_Sig links the data bits to the PLC code.



Reception of Data bit 4 (located in Data block 0).

4.5.3.2 Function block ”Ext_Val”

The function block Ext_Val links the registers to the PLC code.

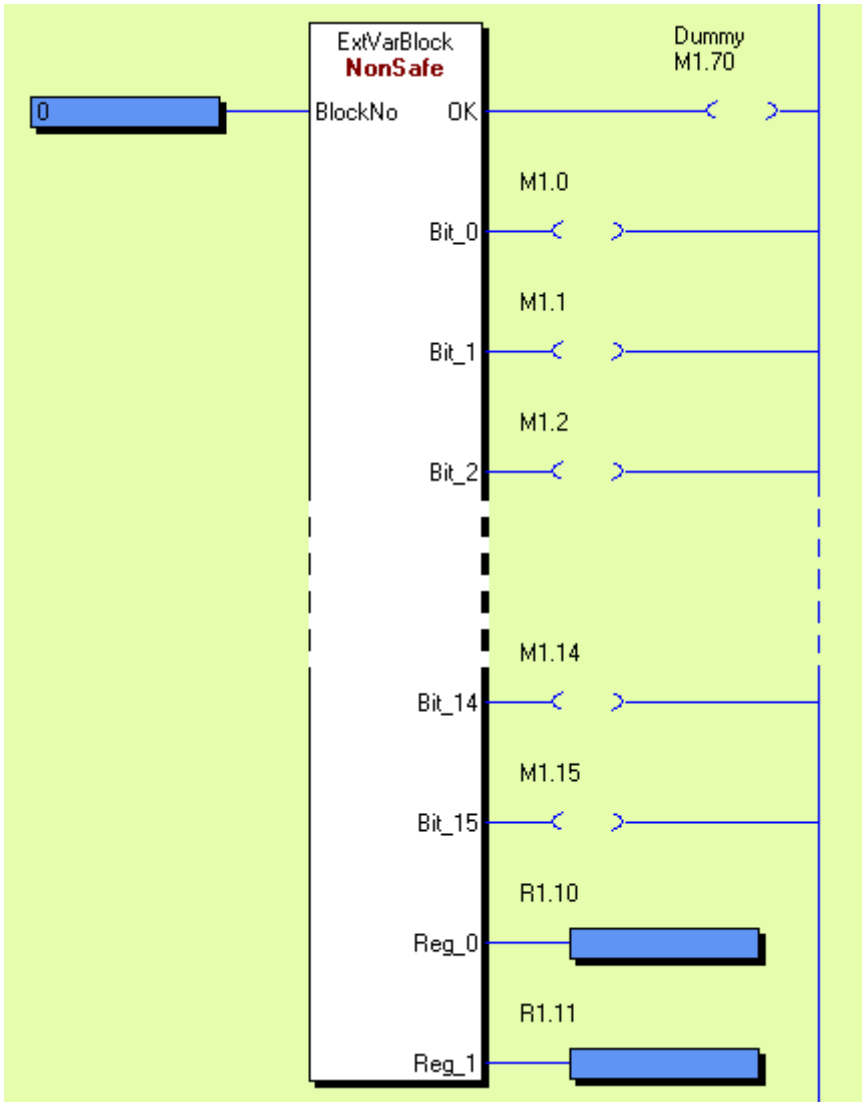


Reception of register 5 (located in Data block 2).

4.5.3.3 Function block ”ExtVarBlock”

The function block ExtVarBlock makes it possible to link all variables in one of the “External comm blocks” to the PLC code. The function block is very big but is easier to use since the only input parameter is the number of the “External comm. Block”.

By setting of BlockNo = 0: Bits 0…15 and Reg 0..1 are given.
By setting of BlockNo = 1: Bits 16…31 and Reg 2..3 are given.
By setting of BlockNo = 2: Bits 32…47 and Reg 4..5 are given.
By setting of BlockNo = 3: Bits 48…63 and Reg 6..7 are given.
(According to table 4.5.2)



Reception of all variables in “External Comm Block” 0,
(16 bits, 2 registers).

5 PROFIBUS

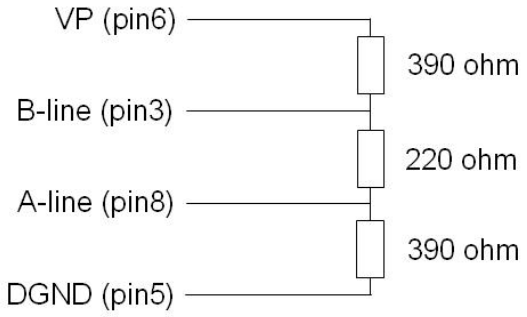
The PROFIBUS implementation in the gateway is as a DP slave using the DP-V0 protocol.
For more information about PROFIBUS see **REF 2**. The DPV0 protocol is fully compatible with the DPV1 and DPV2 protocol.

5.1 Connection

A standard PROFIBUS 9-pole D-sub connector is located at the front of the unit.

Pin	Signal	Description
1	Shield	Shield/functional ground
2	-	-
3	RxD/TxD-P	Receive/Transmit data – plus (B wire – red)
4	CNTR-P	Repeater control signal (direction control), RTS signal
5	DGND	Data ground (reference potential for VP)
6	VP	Supply voltage – plus (P5V)
7	-	-
8	RxD/TxD-N	Receive/Transmit data – minus (A wire – green)
9	-	-

The PROFIBUS cable must have a termination in **each end of the bus.** If not, reflections will cause errors and the communication stops. The termination is done by connecting the two data lines via resistors to the DC voltage in the connector, see the picture below. Normally use PROFIBUS contact with built in termination network (yellow case) in each end of the network and normal PROFIBUS connector (gray case) for all other nodes.



5.1.1 Baud rate

The PROFIBUS speed is auto detected. Supported speeds are:
9.6 kbit/s,
19.2 kbit/s
93.75 kbit/s
187.5 kbit/s
500 kbit/s
1.5 Mbit/s
3 Mbit/s
6 Mbit/s
12 Mbit/s

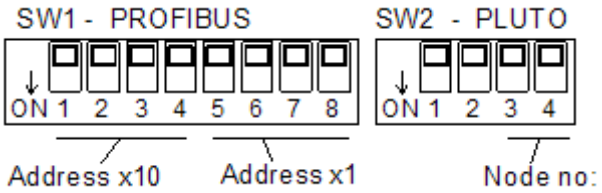
5.2 Indicator PROFIBUS

The PROFIBUS indicator is located just above the PROFIBUS connector.

LED		Description	Remark
Fast flashing red/green		Baud Search	Search traffic data and then set correct baud rate.
Fast flashing green		Waiting Parameter	Discovered working PROFIBUS and waiting for master parameter setting.
Slow flashing green		Waiting Configuration	Discovered working PROFIBUS and waiting correct configuration from master.
Fixed green		Data exchange state	Gateway up running.
Fixed red		Error detected	Bad address setting, see 5.3. Internal error.

5.3 Address switch

The PROFIBUS address is set by DIP-switches “SW1” in the range 00 – 99 with BCD code setting. The units are set on SW1:5-8 and the tens on SW1:1-4 according to table below. If any of the address switches is using the “not used” setting then the PROFIBUS LED will light steady red.



SW1

Address 10x	SW1:1	SW1:2	SW1:3	SW1:4
Address 1x	SW1:5	SW1:6	SW1:7	SW1:8
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
Not used	1	0	1	0
Not used	1	0	1	1
Not used	1	1	0	0
Not used	1	1	0	1
Not used	1	1	1	0
Not used	1	1	1	1

Example:
Address 25 = 0010 0101

5.4 GSD file

The GSD file will show the gateway as a unit where it is possible to add modules according to the needs. The following list of modules is selectable in the GSD file,

Module	Data	Direction	Chapter
PLUTO Status	Pluto Status Data	From Pluto	4.1
PLUTO address 00	Global variables for Pluto 0	From Pluto	4.2
PLUTO address 01	Global variables for Pluto 1	From Pluto	
PLUTO address 02	Global variables for Pluto 2	From Pluto	
PLUTO address 03	Global variables for Pluto 3	From Pluto	
PLUTO address 04	Global variables for Pluto 4	From Pluto	
PLUTO address 05	Global variables for Pluto 5	From Pluto	
PLUTO address 06	Global variables for Pluto 6	From Pluto	
PLUTO address 07	Global variables for Pluto 7	From Pluto	
PLUTO address 08	Global variables for Pluto 8	From Pluto	
PLUTO address 09	Global variables for Pluto 9	From Pluto	
PLUTO address 10	Global variables for Pluto 10	From Pluto	
PLUTO address 11	Global variables for Pluto 11	From Pluto	
PLUTO address 12	Global variables for Pluto 12	From Pluto	
PLUTO address 13	Global variables for Pluto 13	From Pluto	
PLUTO address 14	Global variables for Pluto 14	From Pluto	
PLUTO address 15	Global variables for Pluto 15	From Pluto	
PLUTO address 16	Global variables for Pluto 16	From Pluto	
PLUTO address 17	Global variables for Pluto 17	From Pluto	
PLUTO address 18	Global variables for Pluto 18	From Pluto	
PLUTO address 19	Global variables for Pluto 19	From Pluto	
PLUTO address 20	Global variables for Pluto 20	From Pluto	
PLUTO address 21	Global variables for Pluto 21	From Pluto	
PLUTO address 22	Global variables for Pluto 22	From Pluto	
PLUTO address 23	Global variables for Pluto 23	From Pluto	
PLUTO address 24	Global variables for Pluto 24	From Pluto	
PLUTO address 25	Global variables for Pluto 25	From Pluto	
PLUTO address 26	Global variables for Pluto 26	From Pluto	
PLUTO address 27	Global variables for Pluto 27	From Pluto	
PLUTO address 28	Global variables for Pluto 28	From Pluto	
PLUTO address 29	Global variables for Pluto 29	From Pluto	
PLUTO address 30	Global variables for Pluto 30	From Pluto	
PLUTO address 31	Global variables for Pluto 31	From Pluto	
Data to PLUTO Packet 0	Data to Pluto, packet 0	To Pluto	5.4.2
Data to PLUTO Packet 1	Data to Pluto, packet 1	To Pluto	
Data to PLUTO Packet 2	Data to Pluto, packet 2	To Pluto	
Data to PLUTO Packet 3	Data to Pluto, packet 3	To Pluto	
Req/Resp of local data	Retrieve Local Data	From Pluto	5.4.2
Additional Data 00	Additional data from Pluto	From Pluto	4.3
Additional Data 01	Additional data from Pluto	From Pluto	
Additional Data 02	Additional data from Pluto	From Pluto	
Additional Data 03	Additional data from Pluto	From Pluto	
Additional Data 04	Additional data from Pluto	From Pluto	
Additional Data 05	Additional data from Pluto	From Pluto	
Additional Data 06	Additional data from Pluto	From Pluto	
Additional Data 07	Additional data from Pluto	From Pluto	
Additional Data 08	Additional data from Pluto	From Pluto	
Additional Data 09	Additional data from Pluto	From Pluto	
Additional Data 10	Additional data from Pluto	From Pluto	
Additional Data 11	Additional data from Pluto	From Pluto	
Additional Data 12	Additional data from Pluto	From Pluto	
Additional Data 13	Additional data from Pluto	From Pluto	
Additional Data 14	Additional data from Pluto	From Pluto	
Additional Data 15	Additional data from Pluto	From Pluto	

Module	Data	Direction	Chapter
Additional Data 16	Additional data from Pluto	From Pluto	
Additional Data 17	Additional data from Pluto	From Pluto	
Additional Data 18	Additional data from Pluto	From Pluto	
Additional Data 19	Additional data from Pluto	From Pluto	
Additional Data 20	Additional data from Pluto	From Pluto	
Additional Data 21	Additional data from Pluto	From Pluto	
Additional Data 22	Additional data from Pluto	From Pluto	
Additional Data 23	Additional data from Pluto	From Pluto	
Additional Data 24	Additional data from Pluto	From Pluto	
Additional Data 25	Additional data from Pluto	From Pluto	
Additional Data 26	Additional data from Pluto	From Pluto	
Additional Data 27	Additional data from Pluto	From Pluto	
Additional Data 28	Additional data from Pluto	From Pluto	
Additional Data 29	Additional data from Pluto	From Pluto	
Additional Data 30	Additional data from Pluto	From Pluto	
Additional Data 31	Additional data from Pluto	From Pluto	

Additional Data is only valid in GSD file revision v2.00 and higher.
For more information about each module read the reference chapters.
Note that each module can only be added once in the PROFIBUS configuration.

5.4.1 Common configuration

Timeout
A gateway has a common parameter for timeout setting, see 4.4.

Cycle time
A gateway has a common parameter for cycle time setting, see 4.4.

Gateway Node Address
A gateway has a common parameter for gateway address setting, see 3.4.2.

5.4.2 Module – Data to PLUTO Packet

Each module of data to Pluto has following format, for information see 4.4:

Word	Register	Value type
0	Bit variables	16 bit
1	Register 0	16 bit
2	Register 1	16 bit

Enable bit
When a module for Data to Pluto is added, the usage of the module in the gateway can be enabled/disabled via module parameter setting (the usage is enabled as default).

5.4.3 Module – Req/Resp of local data

With the module “Req/Resp of local data” it is possible for the Profibus system to read local Pluto variables such as (M, SM, R, SR, ..) in the Pluto units. In contrary to the global Pluto variables these are not automatically transmitted to the Pluto bus, the gateway has to ask a Pluto to transmit a telegram with the requested data.
The module has therefore both output and input data, 2 words output and 3 words input data.

5.4.3.1 Output data

To retrieve local data from a Pluto unit output data is set as follows,

Word	Register	Value type
0	Pluto Unit Id	16 bit
1	Local Data Address	16 bit

Pluto Unit Id
Set the Pluto address from which to retrieve local data, a number between 0 and 31.
To retrieve local data from the gateway set the Pluto Unit Id to 0x00FF.

Also bit 15 is used to start the retrieving of local data, for more information see chapter about retrieve sequence.

Bit 15	Value type
1	Indicate that id/address is set for retrieve of local data. (clear it when received acknowledgement in input data)
0	Process data.

Local Data Address
Set the address of the local data you want to retrieve. When retrieving local data from a Pluto unit the 2 most significant bits will indicate the type of data to retrieve, se table below.

When retrieving local data from the gateway the response data will always be double word register value (uint32).

Bit 15	Bit 14	Value type	Return value
0	0	Global memory	0/1
0	1	Local memory	0/1
1	0	Local register	uint16
1	1	Local parameter	uint32

5.4.3.2 Input data

The response of the output data will be input data as follows,

Word	Register	Value type
0	Response Pluto Unit Id	16 bit
1 – 2	Local Data Value	32 bit

Response Pluto Unit Id
This will be set to the same value as the output data together with response error coding.

Bit 15	Bit 11	Bit 10	Bit 9	Bit 8	Value type
0	x	x	x	X	Waiting for output data to be set.
1	x	x	x	X	Acknowledgement of output setting.
0	0	0	0	1	**Wait**, retrieving data from unit.
0	0	0	1	0	**Bad parameter**, id/address error.
0	0	1	0	0	**Timeout**, no unit active/bad address.
0	1	0	0	0	**OK**, Local Data Value valid.

Local Data Value
All retrieved data will be formatted into a double word value in the input data. If only Boolean and word data is retrieved this data can be read according to table below as word value (uint16),

Value type	Use word	value range
Boolean value	2	0 or 1
Word register value (uint16)	2	full range

5.4.3.3 Retrieve sequence

To retrieve data the following sequence shall be used,

Step	To do	Remark
1	Set **Local Data Address**.	
2	Set **Pluto Unit Id**.	
3	Set Bit 15 of **Pluto Unit Id**.	
4	Wait for Bit 15 set in **Response Pluto Unit Id**.	
5	Clear Bit 15 of **Pluto Unit Id**.	
6	Read status bits 8 – 11 in **Response Pluto Unit Id**.	
7	If status Bit 11 is set in **Response Pluto Unit Id** read value in **Local Data Value**.	

5.4.3.4 Organization of PLUTO variables

Below tables describe how the Pluto variables are organized. If the Profibus system shall retrieve local variables in the Pluto system “Local data address” must be used which corresponds to Pluto variables according to the tables.

Pluto A20 family (A16, A20, B16, B20, B22, D20, S19, S20)		
Local Pluto Register	Local data address	
SR0..99	0..99	SR number
R0..R199	100..299	R number + 100
**R200…R347*	*300..447*	*R number + 100*
Local Pluto Bit Variable	Local data address	
Q10..Q17	0..7	Q number – 10
I20..I25 (Pluto B22 only)	8..13	I number - 12
SM0..SM199	16..215	SM number + 16
M0..M807	216..1023	M number + 216

**With Instruction set 3*

Pluto Double Family (B46, S46, D45)		
Local Pluto Register	Local data address	
SR0..99	0..99	SR number
R0..R199	100..299	R number + 100
**R200…R347*	*300..447*	*R number + 100*
Local Pluto Bit Variable	Local data address	
Q10..Q17	0..7	Q number – 10
Q20..Q27	8..15	Q number – 12
SM0..SM199	16..215	SM number + 16
M0..M775	216..991	M number + 216
I20..I27	992..999	I number + 972
I30..I37	1000..1007	I number + 970
I40..I47	1008..1015	I number + 968
Q4..Q5	1020..1021	Q number + 1016

**With Instruction set 3*

Pluto AS-i		
Local Pluto Register	Local data address	
SR0..99	0..99	SR number
R0..R199	100..299	R number + 100
**R200…R347*	*300..447*	*R number + 100*
Local Pluto Bit Variable	Local data address	
Q10..Q13	0..3	Q number – 10
I1..I3	9..11	I number + 8
I10..13	12..15	I number + 2
SM0..SM199	16..215	SM number + 16
M0..M295	216..511	M number + 216
ASI16..ASI31	512..527	
ASI1.1..ASI31B.4	528..775	
ASQ1.1..ASQ31B.4	776..1023	

**With Instruction set 3*

Pluto B42 AS-i		
Local Pluto Register	Local data address	
SR0..99	0..99	SR number
R0..R347	100..447	R number + 100
Local Pluto Bit Variable	Local data address	
Q10..Q17	0..7	Q number – 10
Q20..Q27	8..15	Q number – 12
SM0..SM199	16..215	SM number + 16
M0..M239	216..455	M number + 216
I10..I17	456..463	I number + 446
I20..I27	464..471	I number + 444
I30..I37	472..479	I number + 442
I40..I47	480..487	I number + 440
Q0..Q5	488..493	Q number + 484
ASI1..ASI31	497..527	
ASI1.1..ASI31B.4	528..775	
ASQ1.1..ASQ31B.4	776..1023	

Pluto O2		
Local Pluto Register	Local data address	
SR0..99	0..99	SR number
R0..R199	100..299	R number + 100
**R200…R347*	*300..447*	*R number + 100*
Local Pluto Bit Variable	Local data address	
Q10..Q11	0..1	Q number – 10
SM0..SM199	16..215	SM number + 16
M0..M807	216..1023	M number + 216

**With Instruction set 3*

5.4.3.5 Example in Structured Text

The following example written in structured text language shows how to get the error codes of all Pluto units on the bus. The error code for a Pluto is stored in the local system register SR11.

```
PROGRAM MAIN
VAR
	(*output data value*)
	outPlutoId AT %Q*: UINT;		(*output data, Pluto Unit Id		[word 0]*)
	outPlutoAddress AT %Q*: UINT;		(*output data, Local Data Address	[word 1]*)
	(*input data value*)
	inPlutoId AT %I*: UINT;			(*input data, Response Pluto Unit Id [word 0]*)
	inPlutoData_0 AT %I*: UINT;		(*input data, Local Data Value		[word 1]*)
	inPlutoData_1  AT %I*: UINT;		(*input data, Local Data Value		[word 2]*)
	(*state of the state machine*)
	State: UINT := 0;
	(*the requested PLUTO id number 0-31*)
	pluto: UINT := 0;
	(*counter for counting the number of different response message*)
	respBad: UDINT := 0;
	respTimeout: UDINT := 0;
	respOK: UDINT := 0;
	(*data storage for the respons value if a OK response*)
	respLastValue_0: UINT := 0;
	respLastValue_1: UINT := 0;
END_VAR
```

```
(*This state machine will retrive the Pluto error code stored in SR11 (address 11) from the PLUTO.*)
(*Note this program doesn't have any error handling which shall be added for production use.*)
CASE State OF
0:
	(*This is the start state of the state machine.*)
	(*This will set request data (pluto number, memory address and memory type.*)
	(*Start the retrieval by setting the bit 15.*)
	outPlutoId := pluto;			(*set PLUTO id number*)
	outPlutoAddress := 16#8000+11;		(*set value type and memory address*)
	outPlutoId := outPlutoId + 16#8000;	(*set the bit 15 of PLUTO data*)
	State := 1;				(*goto next state*)
1:
	(*This is next step where the program waits for an respone on bit 15, bit set*)
	IF inPlutoId >= 16#8000 THEN	(*wait to get bit 15 set in the PLUTO id input response*)
		outPlutoId := pluto;	(*clear the bit 15 of PLUTO data*)
		State := 2;		(*goto next state*)
	END_IF
2:
	(*This is next step where the program waits for a response on bit 15, bit cleared*)
	IF inPlutoId < 16#8000 THEN	(*wait to get bit 15 cleared in the PLUTO id input response*)
		State := 3;		(*goto next state*)
	END_IF
3:
	(*In this state the program will check the response bit 11, 10 and 9*)
	(*Note need to test highest value first and the lower and lower value*)
	IF inPlutoId >= 16#0800 THEN		(*check if response bit 11 is set => response OK*)
		respOK := respOK + 1;			(*count number of OK response message*)
		respLastValue_0 := inPlutoData_0;	(*get response value 0*)
		respLastValue_1 := inPlutoData_1;	(*get response value 1*)
		State := 0;				(*goto start state*)
	ELSIF inPlutoId >= 16#0400 THEN		(*check if response bit 10 is set => response timeout*)
		respTimeout := respTimeout +1;		(*count number of timeout response message*)
		State := 0;				(*goto start state*)
	ELSIF inPlutoId >= 16#0200 THEN		(*check if response bit 9 is set => response bad*)
		respBad := respBad + 1;			(*count number of bad response message*)
		State := 0;				(*goto start state*)
	END_IF
	(*Use this code if you want to loop more pluto units*)
	(*Note that response value 0/1 shall be stored in an array or similar if used in a system*)
	IF State = 0  THEN			(*move to next pluto if state set to 0*)
		pluto := pluto + 1;		(*next pluto*)
		IF pluto > 31 THEN		(*if over high limit of pluto*)
			pluto := 0;		(*set low pluto number*)
		END_IF
	END_IF
END_CASE
```

5.4.4 Verification of configuration

Via the serial port function (see chapter 10) there is a possibility to check the status of the gateway and also see which configuration the gateway has received from the PROFIBUS master. This information is printed when doing the “bw” command, see below.

```
pb_gw> bw
-----------------------------
PROFIBUS bus status.
-----------------------------
Node number : 21 [0x15]
Bus speed   : 1500 kbits
Bus status  : DATA EXCHANGE
-----------------------------
PLC INPUT DATA (P=PLUTO) :
 RESP P00 A00
PLC OUTPUT DATA (T=To PLUTO) :
 T0 REQ
 Enabled To PLUTO package 0-3 : 0 - - -, Timeout 0 ms, Update 100 ms.
PLC ADDITIONAL DATA :
 Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type
  00   *10  ASIsafe
-----------------------------
pb_gw>
```

The “PLC Additional Data” will show the configuration if used.

Status information

The text above says that the gateway has address 21 (decimal) 0x15 (hexadecimal) on the PROFIBUS and it has detected the bus speed to be 1.5 Mbits. The unit is in data exchange mode with the master.

Configuration information

This part for this example says that the PROFIBUS master will,

- Receive as input data from module RESP, which is the response data of local data request/response, and P00 which is global data from Pluto number 0.
- Transmit as output data to module T0, which is Data to Pluto package 0, and REQ which is the request data of local data request/response.

During configuration the PROFIBUS master has enabled the use of Data to Pluto package 0 and the timeout is set to 0 ms which means that the Data to Pluto timeout is disabled.

5.4.5 Diagnostic data

For status information and trouble shooting the gateway will give some information via the PROFIBUS diagnostic message. The unit diagnostic data is coded in the following way,

Byte number	Diagnostic function	Data type	Value
1	Station status 1	Bit	
2	Station status 2	Bit	
3	Station status 3	Bit	
4	Diagnostic master address	Bit	
5	PNO identification number	Word	
6			
7	Extended diagnostic header	Byte	10
8	Error flag	Bit	Not used
9	Gateway Pluto node number	Byte	0 – 15
10	Pluto bus speed	Word	0, 100, 125, 200, 250, 400, 800 or 1000
11			
12	Number of active Pluto units	Byte	0 – 32
13	Missing Pluto	Bit	0: Pluto 24 1: Pluto 25 … 7: Pluto 31
14		Bit	0: Pluto 16 1: Pluto 17 … 7: Pluto 23
15		Bit	0: Pluto 8 1: Pluto 9 … 7: Pluto 15
16		Bit	0: Pluto 0 1: Pluto 1 … 7: Pluto 7

6 DeviceNet

The DeviceNet implementation in the gateway is according to ODVA version 2.0.
For more information about DeviceNet see **REF 3**.

6.1 Connection

A standard DeviceNet screw terminal connector is located at the front.

Terminal	Signal	Description
1	V-	0V for the 24VDC.
2	CL	CAN low signal.
3	SE	CAN screen.
4	CH	CAN high signal.
5	V+	+24VDC.

Note that a 120 ohm end terminating resistor must be mounted between CL-CH if the gateway is located as the first or the last unit on the bus.

6.2 Indicator MNS

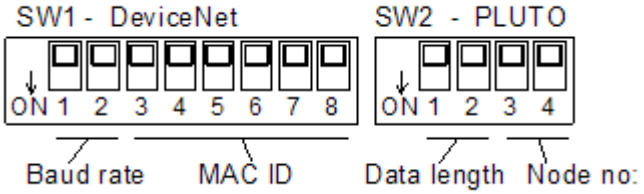
The DeviceNet indicator “MNS” is located just above the DeviceNet connector. The behavior is according to the DeviceNet specification for “Combined Module/Network Status led” (MNS).

LED		Description	Remark
Off		Not Powered Not On-line	- No gateway bus power. - Not done Dup_MAC_ID check yet.
Flashing green/red		Communication Fault.	
Fixed red		Critical Fault.	
Flashing red		Minor Fault, Connection Timeout and/or mismatching configuration.	
Flashing green		Device Operational and Online, Not Connected or Device Online and needs commissioning.	
Fixed green		Device Operational and Online, Connected	

6.3 DIP-switches

The following functions are set with the DIP-switches:

- Baud rate,
- MAC ID,
- Amount of transferred data from Pluto bus (in combination with EDS file, see below) and
- CAN bridge mode



6.3.1 Baud rate setting

Baud rate is set with switches 1 and 2 on switch block “SW1 – DeviceNet” according to following table.

SW1

1	2	Speed [kbits]	Remark
0	0	125	default value
0	1	250	
1	0	500	
1	1	PROG	The baud rate is configured in software via serial port (or via Pluto CAN bus) or via the DeviceNet network.

In program mode, PROG MODE the baud rate and MAC ID are set via the PC port with the “gs” command (available baud rates are: 125, 250, 500 kbit/s). In PROG MODE the baud rate and MAC ID can also be set via the DeviceNet network set command.

6.3.2 MAC ID

MAC ID is set with switches 3…8 on switch block “SW1 – DeviceNet” according to following table. This switch is NOT used when baud rate switch is set in PROG mode.

SW1

3	4	5	6	7	8	Address decimal	Address hexadecimal
0	0	0	0	0	0	0	0x00
0	0	0	0	0	1	1	0x01
0	0	0	0	1	0	2	0x02
0	0	0	0	1	1	3	0x03
0	0	0	1	0	0	4	0x04
0	0	0	1	0	1	5	0x05
0	0	0	1	1	0	6	0x06
0	0	0	1	1	1	7	0x07
0	0	1	0	0	0	8	0x08
0	0	1	0	0	1	9	0x09
0	0	1	0	1	0	10	0x0A
0	0	1	0	1	1	11	0x0B
0	0	1	1	0	0	12	0x0C
0	0	1	1	0	1	13	0x0D
0	0	1	1	1	0	14	0x0E
0	0	1	1	1	1	15	0x0F
0	1	0	0	0	0	16	0x10
0	1	0	0	0	1	17	0x11
0	1	0	0	1	0	18	0x12
0	1	0	0	1	1	19	0x13
0	1	0	1	0	0	20	0x14

6.3.3 PROG mode

If switch SW1 for baud rate setting is set to PROG it is possible to set baud rate and MAC ID via software, either via the PC port or from the DeviceNet master, see page 78. Settings are stored in an EEPROM memory and will be kept in the memory even after power off.

In PROG mode it also possible to via the PC port set input and output parameters and stores them into the EEPROM. In this way configuration can be made without any configuration setting from the DeviceNet master. The DeviceNet master can override these settings but the DeviceNet master settings are not stored into EEPROM.

> **Note:** If using the "PROG" mode then the switch shall always be in this position!

6.3.3.1 Available settings in PROG mode

Under the command "gs" to set the baud rate and MAC ID for the DeviceNet bus.

```
dnet_gw> gs

Gateway interface baudrate :
 1 :  125 kbits
 2 :  250 kbits
 3 :  500 kbits
Select [1] : 1
MACID [63] : 63
```

Under the command "cs" the input assembly can be set.
See 6.5.1, Input Data Assignment - Data from Pluto
Following example shows the dialog.

```
dnet_gw> cs

Input Assembly Instance :
1 : Status Only  [100]
2 : Data   Only  [101]
3 : Status/Data  [102]
Select [0] :
…
```

6.4 Configuration

With the release of the DeviceNet gateway software 2.x the gateway can be configured in two different ways depending on demands,

- Additional Data Configuration
  This configuration adds the feature to handle additional data from Pluto.
  Has the same possibility as the expected node configuration.

- Expected Node Configuration
  This is compliant with software version 1.x and can be called ***the old way of configuration***. This way of configuration is fully functional on gateway with software version 2.x.

6.4.1 Additional Data Configuration

With the release of the software version 2.x the additional data configuration was added.

This configuration has the possibility to retrieve Pluto global data from selected Pluto units and also add the feature to retrieve "Additional Data from Pluto" see 4.3.

For each Pluto IO Data Area the configuration needs to set both the Pluto number and the IO-type number.

The table below shows an example of this new configuration there start of allocation is at Pluto IO Data Area 0 with data from Pluto 24 with IO-type 111 (global data). In this configuration example a total of 4 Pluto IO Data Areas have been used. When Pluto IO Data Area is not wanted/used the IO-type shall be set to zero.

Pluto IO Area	0	1	2	3	4	…	25	26	27	28	29	30	31
Data	x	x	x	x	x	…	x	x	x	x	x	x	x
Pluto	24	24	5	5	0	…	0	0	0	0	0	0	0
IO-type	111	100	111	1	0	…	0	0	0	0	0	0	0

Example of new Pluto IO Data Area configuration (for IO-type see 4.3),

- Pluto IO Data Area 0 allocated to Pluto 24 with IO-type 111 (global data).
- Pluto IO Data Area 1 allocated to Pluto 24 with IO-type 100 (error code).
- Pluto IO Data Area 2 allocated to Pluto 5 with IO-type 111 (global data).
- Pluto IO Data Area 3 allocated to Pluto 5 with IO-type 1 (user data 1).
- Pluto IO Data Area 4 – 31 not allocated (e.g. IO-type set to zero).

The new configuration can only be configured from PLC or via terminal setting, not via DIP-switch.

The way to do a new configuration from PLC is the following (see example above),

- Write Class 100, Instance 0 and Attribute 30 “Pluto IO Data Area 0, Pluto” with 24.
- Write Class 100, Instance 0 and Attribute 31 “Pluto IO Data Area 0, IO-type” with 111.
- Write Class 100, Instance 0 and Attribute 32 “Pluto IO Data Area 0, Pluto” with 24.
- Write Class 100, Instance 0 and Attribute 33 “Pluto IO Data Area 0, IO-type” with 100.
- Write Class 100, Instance 0 and Attribute 34 “Pluto IO Data Area 0, Pluto” with 5.
- Write Class 100, Instance 0 and Attribute 35 “Pluto IO Data Area 0, IO-type” with 111.
- Write Class 100, Instance 0 and Attribute 36 “Pluto IO Data Area 0, Pluto” with 5.
- Write Class 100, Instance 0 and Attribute 37 “Pluto IO Data Area 0, IO-type” with 1.

Note: In additional data configuration never **set/user/read** the attribute expected node!

6.4.2 Expected Node Configuration

The expected node configuration is the default way of configuration in all software versions.

The configuration of data from Pluto was done by setting the expected node bit mask for those Pluto units which shall be included into the IO data to the PLC system. The data from each Pluto were only the Pluto global data.

6.5 EDS file and data length setting

There are several EDS files for the GATE-D1/D2.

File name	Function
GATE- D2	
ABB_GATE-D2_v3.eds	EDS version 3 for GATE-D2
GATE-D1	
JokabDeviceNet_GATE-D1_v3.eds	EDS version 3 with parameter Gateway Node Number else same as JokabDeviceNet_GATE-D1_v2.eds
JokabDeviceNet_GATE-D1_v2.eds	EDS version 2 with additional data else same as JokabDeviceNet_GATE-D1_v1.eds
JokabDeviceNet_GATE-D1_v1.eds	Full size input data block. status data 4 bytes + Pluto data 32x4 bytes = Total of 132 bytes.
JokabDeviceNet_GATE-D1_L02.eds	Input block for only 2 Pluto (8 bytes).
JokabDeviceNet_GATE-D1_L08.eds	Input block for only 8 Pluto (32 bytes).
JokabDeviceNet_GATE-D1_L16.eds	Input block for only 16 Pluto (64 bytes).
JokabDeviceNet_GATE-D1_L32.eds	Input block for only 32 Pluto (128 bytes).

For GATE-D2 use ABB_GATE-D2_vX.eds file.

The EDS file “JokabDeviceNet_GATE-D1_v2.eds” is a full version where the master can control how much data and from which Pluto unit data shall be transferred.
But since all DeviceNet masters do not support this function there are four files with different default data length. All these four limited versions have all the full functionality but with predefined size of the input block. These files shall be selected in combination with the setting of switch 1 and 2 of SW2. The combination shall be made according to following table.

Switch SW2		Pluto nodes	Data size	EDS file
1	2			
0	0	0 – 1	8 bytes	JokabDeviceNet_GATE-D1_L02.eds
0	1	0 – 7	32 bytes	JokabDeviceNet_GATE-D1_L08.eds
1	0	0 – 15	64 bytes	JokabDeviceNet_GATE-D1_L16.eds
1	1	0 – 31	128 bytes	JokabDeviceNet_GATE-D1_L32.eds

The normal EDS file “JokabDeviceNet_GATE-D1_v2.eds”have a predefined input block of the maximal possible input data size, status and all Pluto units enabled.

In all EDS files there are parameter settings, possible for a DeviceNet master to set by initialization of the gateway. Via these parameters it is possible to enable/disable Pluto units into the input data block with a resize of the total block. There are also parameters for enable/disable data to Pluto.

For full details about the EDS file read page 76.

6.5.1 Input Data Assignment - Data from Pluto

The input data from Pluto to the DeviceNet master can be formatted in 3 different ways by setting the parameter “Input Assembly Instance” in the gateway, the default is “Only Pluto Data”, see table below.

Which Pluto units that will be included into the input data can be set by the parameter “Expected Nodes Bitmap”. The default value is depending on the mode switch SW2 see 6.5. This setting will change the input assembly size and the current size can be read from the gateway via parameter read “Input Assembly Size”.

For more information see page 76.

The table below shows the organization of input data.

	Only status Data	Only Pluto Data	Status/Pluto Data
0	Status	Pluto 0	Status
4	-	Pluto 1	Pluto 0
8	-	Pluto ...	Pluto …
12	-	…	…

- If using “only status data” mode then input data size will be fixed to 4 bytes.
- If using “only Pluto data” or “status/Pluto data” the size will depend on the mode setting on switch SW2 or via software, either DeviceNet command “expected Pluto” or PC port.

6.5.1.1 Status data

See chapter 4.1.

6.5.1.2 Pluto global variables

See chapter 4.2.

6.5.2 Output Data Assignment - Data to Pluto

To enable data to Pluto the DeviceNet master needs to set some parameters in the gateway:
- Output Assembly Instance.
- Enable Data To Pluto
- Data to Pluto Timeout (if used, default set to 0 e.g. disabled).

For more information see page 76.

As described in 4.4 the gateway can transfer totally 64 Boolean variables and 8 registers to the Pluto bus. The data is divided in four areas.
DeviceNet is different from CANopen and PROFIBUS since it can not be set to write only some of the areas, if the parameter “Enable Data To Pluto” is set all four areas are written even if they are not used.

As described in 4.4 the data to Pluto is organized as follows:

Word	Area	Register	Value type
0	0	Bit Variables	16 bit
1		Register 0	16 bit
2		Register 1	16 bit
3	1	Bit Variables	16 bit
4		Register 0	16 bit
5		Register 1	16 bit
6	2	Bit Variables	16 bit
7		Register 0	16 bit
8		Register 1	16 bit
9	3	Bit Variables	16 bit
10		Register 0	16 bit
11		Register 1	16 bit

6.5.3 Local Data

It is also possible for the DeviceNet system to read local Pluto variables such as (M, SM, R, SR, …) in the connected Pluto. In contrary to the global Pluto variables these are not automatically transmitted to the Pluto bus so the gateway has to ask a Pluto to transmit a telegram with the requested data. This is done by using explicit messaging. For more information about the usage of this module read the page 76.

6.5.4 Verification of configuration

Via the serial port function (see chapter 10) there is a possibility to check the status of the gateway and also see which configuration the gateway has received from the DeviceNet master. This information is printed when doing the “bw” command, see below.

```
Expected Node Configuration

dnet_gw> bw
-----------------------------
DeviceNet bus status.
-----------------------------
Node number : 3 [0x3]
Bus speed   : 500 kbits
Bus power   : VALID
Bus status  : OFFLINE
-----------------------------
Input  assembly 1 = PLUTO Data Only   [102]
 Expected PLUTO 00-15 : 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15
 Expected PLUTO 16-31 : 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Output assembly 0 = No Data           [112]
 Enabled To PLUTO package 0-3 : 0 1 2 3, Timeout 1000 ms, Update 100 ms.
-----------------------------
dnet_gw>

Additional Data Configuration

dnet_gw> bw
-----------------------------
DeviceNet bus status.
-----------------------------
Node number : 3 [0x3]
Bus speed   : 500 kbits
Bus power   : VALID
Bus status  : OFFLINE
-----------------------------
Input  assembly 1 = PLUTO Data Only   [102]
 Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type
  00    00  USER:01 |  01    00  USER:02 |  02    00  USER:03 |  03    00  USER:04
  04    00  USER:05 |  05    00  USER:06 |  06    00  USER:07 |  07    00  USER:08
  08    00  USER:09 |  09    00  USER:10 |  10    00  USER:11 |  11    00  USER:12
  12    00  USER:13 |  13    00  USER:14 |  14    00  USER:15 |  15    00  USER:16
  16    00  USER:17 |  17    00  USER:18 |  18    00  USER:19 |  19    00  USER:20
  20    00  USER:21 |  21    00  USER:22 |  22    00  USER:23 |  23    00  USER:24
  24    00  USER:25 |  25    00  USER:26 |  26    00  USER:27 |  27    00  USER:28
  28    00  USER:29 |  29   *31  GLOBAL  |  30    00  ErrCode |  31   *00  GLOBAL
Output assembly 0 = No Data           [112]
 Enabled To PLUTO package 0-3 : - - - -, Timeout 0 ms, Update 100 ms.
-----------------------------
dnet_gw>
```

Status information

The text above says that the gateway has address 3 (decimal) 0x3 (hexadecimal) on the DeviceNet bus, bus speed is set to 500 kbits and the gateway detects valid bus power. The unit is in connection with the master.

Configuration information

This part of the example says that the DeviceNet master will,

- Receive “Pluto Data Only” as input data according to instant 102.
- Transmit “Data to Pluto” as output data according to instant 112.
- Set Expected Pluto stations only to expect data from Pluto number 0.
- During configuration enable the use of Data to Pluto package 0 and 2 and set the timeout to 0 ms which mean that the Data to Pluto timeout is disabled.

The text “BY PLC” indicates that the DeviceNet master (PLC) has updated these data.

7 CANopen

The CANopen gateway conforms to version 4.02 of CIA Draft Standard 301.
For more information about CANopen see **REF 4**.

7.1 Connection

A screw terminal connector is located at the front of the unit.

Terminal	Signal	Description
1	-	-
2	CL	CAN low signal.
3	SE	CAN screen.
4	CH	CAN high signal.
5	-	-

Note that a 120 ohm end terminating resistor must be mounted between CL-CH if the gateway is located as the first or the last unit on the bus.

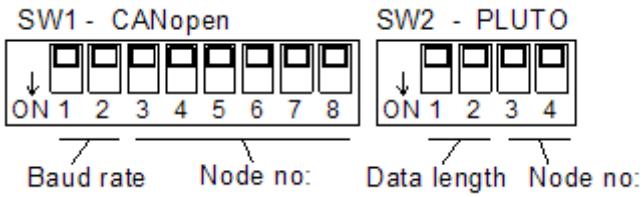
7.2 Indicator - STATUS

The CANopen indicator “STATUS” is located just above the CANopen connector. The behavior is according to the CANopen specification for “CANopen run LED”.

LED		Description	Remark
RED steady		Fatal error	
RED flashing		STOPPED	
GREEN flashing		PRE-OPERATION	
GREEN steady		OPERATION	

7.3 DIP-switch

The following functions are set with the DIP switch:
- Node number
- Baud rate
- Amount of transferred data from Pluto bus (with limited EDS file, see below).
- CAN bridge mode



7.3.1 Baud rate setting

Baud rate is set with switches 1 and 2 on switch block SW1 according to following table.

SW1

1	2	Speed [kbits]	Remark
0	0	125	default value
0	1	250	
1	0	500	
1	1	PROG	The baud rate and MAC ID are configured in software via serial port (or via Pluto CAN bus).

In program mode, PROG MODE the baud rate and MAC ID are set via the PC port with the “gs” command. Available baud rates are: 10, 20, 50, 100, 125, 250, 500, 800, 1000 kbits.

7.3.2 Node number

Node number is set with switches 3…8 on switch block SW1 according to following table. This switch is NOT used when baud rate switch is set in PROG mode.

SW1

3	4	5	6	7	8	Address decimal	Address hexadecimal
0	0	0	0	0	0	0	0x00
0	0	0	0	0	1	1	0x01
0	0	0	0	1	0	2	0x02
0	0	0	0	1	1	3	0x03
0	0	0	1	0	0	4	0x04
0	0	0	1	0	1	5	0x05
0	0	0	1	1	0	6	0x06
0	0	0	1	1	1	7	0x07
0	0	1	0	0	0	8	0x08
0	0	1	0	0	1	9	0x09
0	0	1	0	1	0	10	0x0A
0	0	1	0	1	1	11	0x0B
0	0	1	1	0	0	12	0x0C
0	0	1	1	0	1	13	0x0D
0	0	1	1	1	0	14	0x0E
0	0	1	1	1	1	15	0x0F
0	1	0	0	0	0	16	0x10
0	1	0	0	0	1	17	0x11
0	1	0	0	1	0	18	0x12
0	1	0	0	1	1	19	0x13
0	1	0	1	0	0	20	0x14

7.3.3 Amount of transferred data from Pluto

With switch SW2 (1…2) it is possible to set the number of Pluto units that data shall be transferred to the CANopen bus. The Pluto variables are packed into PDO:s, one PDO contains variables from two Pluto nodes. The switches are read during boot up of the gateway, so the setting can not be changed during operation.

The setting is made according to the table below.

Switch SW2		Data from Pluto nodes	No of PDO:s	Remark
1	2			
0	0	0 – 1	1	Data from the first 2 Pluto units
0	1	0 – 7	4	Data from the first 8 Pluto units
1	0	0 – 15	8	Data from the first 16 Pluto units
1	1	0 – 31	16	Data from the first 32 Pluto units

The CANopen master can however override the switch setting by using features in the CANopen EDS file.

7.3.4 PROG mode

If switch SW1 for baud rate setting is set to PROG it is possible to set baud rate and MAC ID via the PC port. Settings are stored in an EEPROM memory and will be kept in the memory also after power off.

In PROG mode it is also possible via the PC port to set input and output parameters and store them into EEPROM. These settings are very limited and shall only be used when needed. Also in this mode the configuration can be made from the CANopen master and these settings will override settings done via the PC port and they are not stored into EEPROM.

Note: If using the “PROG” mode then the switch shall always be in this position!

7.3.4.1 Available settings in PROG mode

Under the command “gs” Baud rate for the CANopen bus can be set.

```
co_gw> gs

Gateway interface baudrate :
 1 :   10 kbits
 2 :   20 kbits
 3 :   50 kbits
 4 :  100 kbits
 5 :  125 kbits
 6 :  250 kbits
 7 :  500 kbits
 8 :  800 kbits
 9 : 1000 kbits
Select  [5] :
MAC ID [63] :
```

Under the command “cs” four different settings can be made:

- Expected data from Pluto.
  From which Pluto nodes data shall be transferred to the CANopen bus.
- Enabling of data areas to Pluto.
- Time out for data to Pluto.

Following example shows the dialog when using only global data.

```
co_gw> cs

NOTE set TPDO parameters for ALL enabled TPDO:s
================================================
TPDO Transmission Type (0-255) [1] :
TPDO Inhibit Time [50] (ms) :
TPDO Event Time [30000] (ms) :
IO Configuration way :
0 : Expected Node Configuration [Only global data]
1 : Additional Data Configuration [Clear current configuration]
2 : Additional Data Configuration [Keep  current configuration]
Select [0]: 0
Expected data from PLUTO 00  [Y] ?  YES
Expected data from PLUTO 01  [Y] ?  YES
Expected data from PLUTO 02  [Y] ?  YES
Expected data from PLUTO 03  [Y] ?  NO
Expected data from PLUTO 04  [Y] ?  NO
...
Expected data from PLUTO 29  [Y] ?  NO
Expected data from PLUTO 30  [Y] ?  NO
Expected data from PLUTO 31  [Y] ?  NO
Enable To PLUTO package 0  [N] ?  YES
Enable To PLUTO package 1  [N] ?  YES
Enable To PLUTO package 2  [N] ?  NO
Enable To PLUTO package 3  [N] ?  NO
To PLUTO Timeout [0 ms] : 1000
To PLUTO update time [100 ms] :
Save the new configuration [y/n]  YES


co_gw> bw
-----------------------------
CANopen bus status.
-----------------------------
Node number : 3 [0x3]
Bus speed   : 125 kbits
Bus status  : PRE-OPERATIONAL
-----------------------------
Current setup done by EEPROM setting (PROG MODE).
Expected PLUTO 00-15 : 00 01 02 03 -- -- -- -- -- -- -- -- -- -- -- --
Expected PLUTO 16-31 : -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
 Enabled To PLUTO package 0-3 : 0 1 - -, Timeout 1000 ms, Update 100 ms.
-----------------------------
co_gw>
```

Following example shows the dialog when using additional data.

```
co_gw> cs

NOTE set TPDO parameters for ALL enabled TPDO:s
================================================
TPDO Transmission Type (0-255) [1] :
TPDO Inhibit Time [50] (ms) :
TPDO Event Time [30000] (ms) :
IO Configuration way :
0 : Expected Node Configuration [Only global data]
1 : Additional Data Configuration [Clear current configuration]
2 : Additional Data Configuration [Keep  current configuration]
Select [0]: 1
Area 00 data from PLUTO 00  0
Area 00 data IO type 000    111
Area 01 data from PLUTO 00  1
Area 01 data IO type 000    111
Area 02 data from PLUTO 00  0
Area 02 data IO type 000    100
Area 03 data from PLUTO 00  1
Area 03 data IO type 000    100
Area 04 data from PLUTO 00  0
Area 04 data IO type 000    1
Area 05 data from PLUTO 00  1
Area 05 data IO type 000    1
Area 06 data from PLUTO 00
Area 06 data IO type 000
Area 07 data from PLUTO 00
Area 07 data IO type 000
...
Area 30 data from PLUTO 00
Area 30 data IO type 000
Area 31 data from PLUTO 00
Area 31 data IO type 000
Enable To PLUTO package 0  [Y] ?  YES
Enable To PLUTO package 1  [Y] ?  YES
Enable To PLUTO package 2  [N] ?  NO
Enable To PLUTO package 3  [N] ?  NO
To PLUTO Timeout [1000 ms] : 500
To PLUTO update time [100 ms] : 50
Save the new configuration [y/n]  YES

co_gw> bw
-----------------------------
CANopen bus status.
-----------------------------
Node number : 3 [0x3]
Bus speed   : 125 kbits
Bus status  : PRE-OPERATIONAL
-----------------------------
Current setup done by EEPROM setting (PROG MODE).
 Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type
  00   *00  GLOBAL  |  01    01  GLOBAL  |  02   *00  ErrCode |  03    01  ErrCode
  04   *00  USER:01 |  05    01  USER:01 |
 Enabled To PLUTO package 0-3 : 0 1 - -, Timeout 500 ms, Update 48 ms.
-----------------------------
co_gw>
```

7.3.5 CAN bridge mode

See chapter 8 CAN bridge mode.

7.4 EDS file

In the EDS file there are parameters for initialization of the gateway by the CANopen master. Via these parameters it's possible to enable and disable Pluto PDO data in proportion to from which Pluto unit's data is needed. Note that these settings override the switch settings on SW2:(1,2) , described above in 7.3.3
There are also parameters to enable and disable data to Pluto.

For full details about the EDS file read the chapter Appendix B, CANopen EDS description on page 90.

7.4.1 Configuration TPDO

The gateway will send data to PLC (data from Pluto) using the configuration set on TPDO index 0x1800 to 0x180F (TPDO0 to TPDO16). Each TPDO hold data for two Pluto or additional data. For each TPDO there is following parameters;

Sub-index	Data	Sync operation	On change
0x01	COB-ID	Yes	Yes
0x02	Transmission Type	1 - 240	254/255
0x03	Inhibit Time (ms)	-	Yes
0x05	Event Timer (ms)	-	Yes

COB-ID: Clear bit 31 to enable the TPDO.
Transmission Type: Value 1 will give data on every SYNC command received by the gateway. With 2 the gateway will send data every second SYNC command and so on.
Inhibit Time (ms): Is used for on change data is specified the minimum time between data sent by the gateway e.g. faster data changes will be filtered.
Event Timer (ms): Is used for on change data is defined the maximum time between data if data is not changed.

Note: By default all TPDO is disabled in the EDS-file e.g. bit 31 is set in the COB-ID!

From OS version 2.0 there is a special index **0x2005** to easy enable several TPDO messages.

Sub-index	Data	Sync operation	On change
0x01	Transmission Type	1 - 240	254/255
0x02	Inhibit Time (ms)	-	Yes
0x03	Event Timer (ms)	-	Yes
0x04	Enable TPDO	Yes	Yes

Transmission type, Inhibit Time and Event Timer is the same as for the normal TPDO configuration and all these shall be written before writing data to Enable TPDO.

The “Enable TPDO” is a bit field where bit 0 is TPDO1, bit 1 is TPDO2 and so on up to bit 15 which is TPDO16. By writing this information to “Enable TPDO” those TPDO with bit set will be enabled with data according to sub-index 0x01, 0x02 and 0x03 information. The other TPDO which was cleared will be disabled (e.g. no data transfer).

Example	Write 0x2005:0x01	0xff	(on change operation)
	Write 0x2005:0x02	0x64	(Inhibit time of 100 ms)
	Write 0x2005:0x03	0x1388	(Event timer of 5000 ms)
	Write 0x2005:0x04	0x8002	(Enable TPDO1 and TPDO16)

7.4.2 Configuration Data to Pluto

The PLC on CANopen can send data to Pluto (see chapter 4.4) via RPDO but the gateway need to be configured to enable data to be sent on the Pluto bus. This can be done via index 0x2002.

Sub-index	Data	Information
0x01	Enable data to Pluto areas 0 – 3.	Bit field data 0x01 enable area 0 0x02 enable area 1 0x04 enable area 2 0x08 enable area 3
0x02	Data to Pluto timeout (ms)	Gateway timeout if no update of data from CANopen PLC. 0 = disabled. 1 – 60000 ms
0x03	Cycle update time (ms) **Only from OS 2.0**	Gateway update time on the Pluto bus. Lower value will load the Pluto bus (100 ms). 0 – 254 ms

7.4.3 Additional data

From CANopen OS 2.0 it is possible to handle additional data (see chapter 4.3) via the CANopen gateway. The configuration is done via the index 0x2011 – 0x2020 for TPDO1 until TDPO15. For each index the following data shall be written, note that each TPDO holds to additional data blocks.

Note: For proper operation of additional data all index and sub index for 0x2011 to 0x2020 shall be written!

Sub-index	Data
0x01	Additional data area x
0x02	Additional data area x+1

Data for each additional data areas are 16 bits value there the high byte hold Pluto node number with a value 0x00 – 0x1F (0 – 31) and low byte hold type of additional data (IO-type). To configure additional data area 0 with data from Pluto 10 and with global data the following data shall be written:

Write 0x2011:0x01 0x0A6F 0x0A is Pluto 10
0x6F is global data (decimal value 111)

For more information regarding the different IO-type numbers is chapter 4.3.3.

Index 0x2010 can be used to determine if additional data is configured for any of the TPDOs. This is a bit field where bit 0 is TPDO1, bit 1 is TPDO2 and so on up to bit 15 which is TPDO16. If bit is set the TPDO is used for additional data. By writing 0x0000 to this index all current additional data configuration will be cleared easy and quick.

Write 0x2010:0x01 0x0000 Will disable additional data configuration

7.4.4 Gateway Pluto node number

Each gateway have a node number read from DIP-switch (0 – 3). From CANopen OS 2.0 this node number can be set in range 0 – 15 and also by writing to index 0x2006, for more information see page 90.

Sub-index	Data
0x01	Pluto gateway node number.

7.4.5 Mapping the PDO's

The default mapping of TX/RX PDO for data from Pluto and data to Pluto is according to the table below. Following chapters describe the mapping of each type of PDO's.

11-bit Can Header (COB ID)	Message Description
0x000	NMT (Network Management)
0x080	SYNC
0x080 + Node ID	Emergency Message
0x100	Time Stamp
0x580 + Node ID	Transmit SDO
0x600 + Node ID	Receive SDO
0x700 + Node ID	NMT Error Control / Heartbeat
	Tx PDO
0x180 + Node ID	Tx PDO 1 – Pluto Inputs (Nodes 0-1)
0x280 + Node ID	Tx PDO 2 – Pluto Inputs (Nodes 2-3)
0x380 + Node ID	Tx PDO 3 – Pluto Inputs (Nodes 4-5)
0x480 + Node ID	Tx PDO 4 – Pluto Inputs (Nodes 6-7)
0x1A0 + Node ID	Tx PDO 5 – Pluto Inputs (Nodes 8-9)
0x2A0 + Node ID	Tx PDO 6 – Pluto Inputs (Nodes 10-11)
0x3A0 + Node ID	Tx PDO 7 – Pluto Inputs (Nodes 12-13)
0x4A0 + Node ID	Tx PDO 8 – Pluto Inputs (Nodes 14-15)
0x1C0 + Node ID	Tx PDO 9 – Pluto Inputs (Nodes 16-17)
0x2C0 + Node ID	Tx PDO 10 – Pluto Inputs (Nodes 18-19)
0x3C0 + Node ID	Tx PDO 11 – Pluto Inputs (Nodes 20-21)
0x4C0 + Node ID	Tx PDO 12 – Pluto Inputs (Nodes 22-23)
0x1E0 + Node ID	Tx PDO 13 – Pluto Inputs (Nodes 24-25)
0x2E0 + Node ID	Tx PDO 14 – Pluto Inputs (Nodes 26-27)
0x3E0 + Node ID	Tx PDO 15 – Pluto Inputs (Nodes 28-29)
0x4E0 + Node ID	Tx PDO 16 – Pluto Inputs (Nodes 30-31)
	Rx PDO
0x200 + Node ID	Rx PDO 1 – Network Output Area 0
0x300 + Node ID	Rx PDO 2 – Network Output Area 1
0x400 + Node ID	Rx PDO 3 – Network Output Area 2
0x500 + Node ID	Rx PDO 4 – Network Output Area 3

7.4.6 Input Data Assignment – Data to Pluto

A PDO contains data from two Pluto units (additional data areas). By default the gateway will enable PDO's according to mode switch SW2(1,2), see 7.3.3. The organization of the data from Pluto within a PDO is according to below table (with offset for Pluto expected position).

Byte	Pluto node no:	MSB							LSB
0	Even no. 0, 2, 4,…	Example Pluto 4							
1									
2									
3									
4	Odd no. 1, 3, 5, …	Pluto 4+1 = Pluto 5							
5									
6									
7									

For detailed description of data see chapter 4.2.

The enabling of PDO's for the wanted Pluto units can be done via CANopen SDO message.
For more information see Appendix B, CANopen EDS description on page 90.

7.4.7 Output Data Assignment – Data to Pluto

To enable data to Pluto the CANopen master needs to set some parameters in the gateway:
- Data to Pluto Setting, Enable Areas 0 – 3.
- Data to Pluto Setting, Data to Pluto Timeout (default 0 = disabled).
- Data to Pluto Setting, Cycle Update Time (default 100 ms).

For more information see Appendix B, CANopen EDS description on page 90.

As described in 4.4 the gateway can transfer totally 64 Boolean variables and 8 registers divided in four areas.
Each area is written by four separate PDO messages (different COB ID).

The format of each PDO is as following:

Byte	Register	Value type
0 – 1	Bit Variables	16 bit
2 – 3	Register 0	16 bit
4 – 5	Register 1	16 bit

For detailed description of the data see chapter 4.4.

7.4.8 Local Data

It is also possible for the CANopen system to read local Pluto variables such as (M, SM, R, SR, ..) in the connected Pluto units. In opposite to the global Pluto variables these are not automatically transmitted to the Pluto bus so the gateway has to request a Pluto to transmit a telegram with the data. This is done by using SDO messages, for more information about the usage of this module read the Appendix B, CANopen EDS description on page 90.

7.4.9 Gateway Node Number

Gateway node number can be set via SDO. Note that set value zero will read node number from DIP-switch. To set gateway node number to node number zero the set value shall be 1. For more information see object 0x2005 in Appendix B, CANopen EDS description on page 90.

7.4.10 Enable TPDO

Each TPDO can be easy be enabled/disabled via a single set instruction, see object 0x2005 in Appendix B, CANopen EDS description on page 90.

7.4.11 Configuration of additional data

Configuration of additional data can be done for each TPDO via object 0x2011 – 0x2020 in Appendix B, CANopen EDS description on page 90.

Note that when using additional data all configurations shall be done with additional data setting. The global data need also to be configured via the same commands. When using additional data it is a good practice to start allocating (using) additional data first in TPDO1. Then add more additional data in TPDO2 and so on up to end TPDO16.

Each TPDO handles two additional data areas and are configured using a single 16 bits value there the higher 8 bits is the Pluto number and the low 8 bits are the IO type.

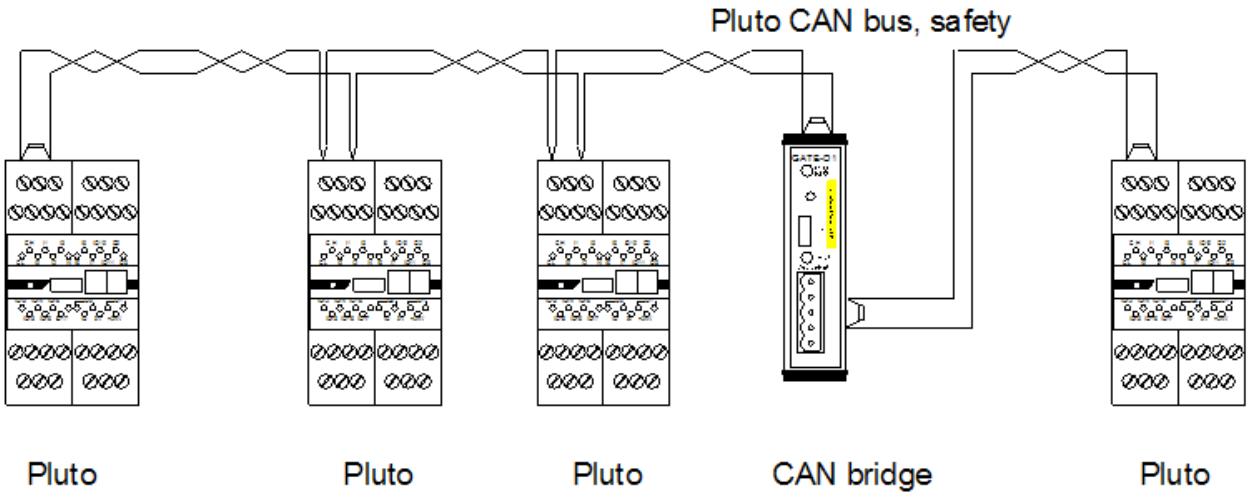
Example 0x026F

0x02 is decimal 2, which gives Pluto number 2.
0x6F is decimal 111 which is the IO type number for global data, see chapter 4.3.3.

After configuration the bw command will show the current configuration of the unit.

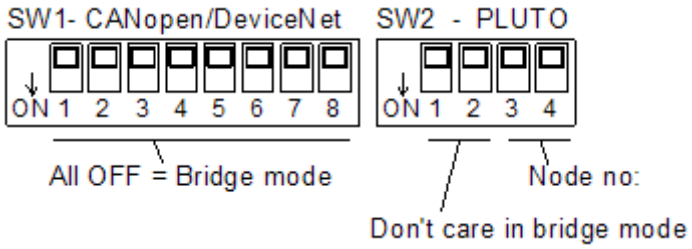
Via the object 0x2010 the additional data configuration can be cleared by writing non zero value. When read this object the number of additional areas can be read.

8 CAN bridge mode



The versions GATE-D1/D2 and GATE-C1/C2 have an operation mode where the normal DeviceNet or CANopen function is disabled and instead it works as a bridge between two CAN buses. It can then for example be used when the needed cable length is longer than what the baud rate permits. The communication through this Gateway bridge is safe. By placing a bridge in the middle of the bus it is possible to get twice as long cable.
There are also filter functions in bridge mode. The filter can be set so I/O information from some Pluto nodes will be blocked, which will reduce the traffic on a bus segment.

This mode is enabled if all switches on SW1 are set to 0/OFF.
NOTE: The unit must be powered OFF/ON before the setting takes effect.



8.1 Pluto filter

In bridge mode it is possible to set filter so the cyclic I/O messages from some units are not bridged. The function can be used when it is needed to bring down the bus load.
The filter is set via the PC port see chapter 10, Serial port Functions.

Type “fs” for entering the filter set. Then the user has to answer questions with yes or no [Y/N]. The following example shows the procedure where Pluto 1, 26...31 are blocked by the gateway. The first question is whether the filter functions shall be used at all, and the last question is to confirm the setting.

```
co_gw> fs
Enable bridge filter    [N] ?  YES
Enable bridge Pluto 00  [Y] ?
Enable bridge Pluto 01  [Y] ?  NO
Enable bridge Pluto 02  [Y] ?
.
.
Enable bridge Pluto 24  [Y] ?
Enable bridge Pluto 25  [Y] ?
Enable bridge Pluto 26  [Y] ?  NO
Enable bridge Pluto 27  [Y] ?  NO
Enable bridge Pluto 28  [Y] ?  NO
Enable bridge Pluto 29  [Y] ?  NO
Enable bridge Pluto 30  [Y] ?  NO
Enable bridge Pluto 31  [Y] ?  NO
Save new filter setting [Y/N] ?  YES
co_gw>
```

The filtering has effect on the cyclic I/O telegrams for the selected Pluto units. But it has also effect on telegrams from encoders which are filtered out as soon as “Enable bridge filter” is selected. Encoder telegrams have CAN-ID: 0x80, 0x281..290, 0x581..590, 0x601..610.
Other telegrams are passed through.

By typing “bs” for bus status, the following data is shown.
Pluto 0, 1 are connected to bus 1, Pluto 0 is bridged and Pluto 1 is not bridged (blocked).
Pluto 24…31 are connected to bus 2, 24 and 25 are bridged and 26..31 are not.
Pluto 2…23 are not active on the bus.

Bus 1 is connected to the “Pluto connector” and bus 2 to the “CANopen connector”, however the two sides has the same function in this mode.

```
co_gw> bs
**** BRIDGE MODE **** Bridge filter ENABLE.
PLUTO gateway node 0.
CAN bus 1 (PLUTO bus) speed is 400 kbits.
CAN bus 2 (CANopen) speed is 400 kbits.

PLUTO 00 : A20  1  BRIDGE      PLUTO 16 : -       BRIDGE
PLUTO 01 : A20  1              PLUTO 17 : -       BRIDGE
PLUTO 02 : -       BRIDGE      PLUTO 18 : -       BRIDGE
PLUTO 03 : -       BRIDGE      PLUTO 19 : -       BRIDGE
PLUTO 04 : -       BRIDGE      PLUTO 20 : -       BRIDGE
PLUTO 05 : -       BRIDGE      PLUTO 21 : -       BRIDGE
PLUTO 06 : -       BRIDGE      PLUTO 22 : -       BRIDGE
PLUTO 07 : -       BRIDGE      PLUTO 23 : -       BRIDGE
PLUTO 08 : -       BRIDGE      PLUTO 24 : A20  2  BRIDGE
PLUTO 09 : -       BRIDGE      PLUTO 25 : A20  2  BRIDGE
PLUTO 10 : -       BRIDGE      PLUTO 26 : B16  2
PLUTO 11 : -       BRIDGE      PLUTO 27 : B16  2
PLUTO 12 : -       BRIDGE      PLUTO 28 : B16  2
PLUTO 13 : -       BRIDGE      PLUTO 29 : B20  2
PLUTO 14 : -       BRIDGE      PLUTO 30 : B20  2
PLUTO 15 : -       BRIDGE      PLUTO 31 : B20  2
co_gw>
```

9 Ethernet gateway

The Ethernet gateway GATE-E1/E2 implements several Ethernet protocols. All protocols are running simultaneously on the gateway. Even if the servers are running simultaneously some of them shall not be used simultaneously. The table below shows which protocols can be used simultaneously.

Protocol	EtherNet/IP	PROFINET	Modbus TCP	Binary TCP	Web page	Terminal
EtherNet/IP						
PROFINET						
Modbus TCP						
Binary TCP						
Web page						
Terminal						

For more information see **REF 3**.
For more information see **REF 2**.
For more information see **REF 5**.
Binary TCP/IP server.
Web page for status and network configuration.
Terminal server for configuration, status and diagnostic use (similar to telnet).

Description of colors and text,

	Can be used simultaneously.
	Can't be used simultaneously.

Note: The recommendation is to **only use one** of the protocols.

Note: Use of the "Binary TCP" protocol shall be **avoided**.
It will maybe **not be supported** in future gateway products.

9.1 Connection

The Ethernet port is using a standard RJ45 connector.
Only screened cable shall be used (cat5e FTP).

Note: The preferred network connection of GATE-E1/2 is via a **"Managed switch"** to lower the network traffic on the gateway network port. For small networks this may not be necessary if the network load is low, but it is still the recommendation to use a "Managed switch".

9.2 DIP-switch

The following functions are set with the DIP switch:

- Gateway node number see chapter 3.4.
- Module and Network Status selection see chapter 9.5.1.

9.3 Ethernet Network setting

Ethernet network units need an IP-address, IP subnet mask and IP gateway address if used.
The table below shows the default settings for this information at delivery.

Ethernet Network Setting	Default setting
IP address	192.168.0.100
IP subnet mask	255.255.255.0
IP gateway address	0.0.0.0
PNIO Device Name	GATE-E2

9.3.1 Change IP-address

The IP-address is viewed and changed in three ways..

- The serial port (preferred).

  By connecting a cable to the serial port of the gateway, see 10.
  List all commands by the “h” command.
  See current setting by the “bw” command.
  Change address by the “ipaddr” command.
  Change PNIO Device Name by the “name” command (PROFINET).

- The terminal server connection.

  By a telnet client connect to current (default) IP-address and port 50100, see 9.4.5.
  List all commands by the “h<enter>” command.
  See current setting by the “bw<enter>” command.
  Change address by the “ipaddr<enter>” command.
  Change PNIO Device Name by the “name” command (PROFINET)

- The web server.

  By a web browser connect to the current (default) IP-address and standard port 80, see 9.4.1.
  View the IP-address and change then by a click on the edit text.

Note: For PROFINET the unit shall have a unique PNIO Device Name.
Both TCP/IP address and PNIO Device Name must be correct!

9.4 Protocol

This chapter will describe each protocol.

9.4.1 Web page

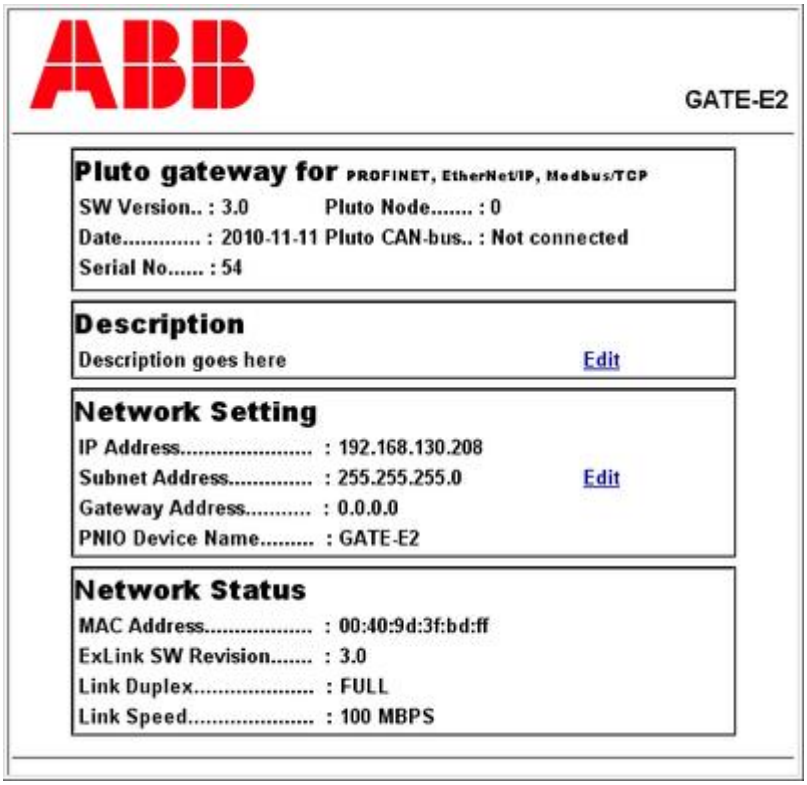
Via a web browser the following page can be read from the Ethernet gateway. On this webpage the gateway presents status information and a possibility to edit network settings etc.

- GATE-E1/E2 status
  Give status of the gateway software version and date.
  Gateway serial number.
  Pluto gateway node number and detected Pluto CAN bus speed.

- Description
  A user defined description text for identification (can be edited from the web page).

- Network setting
  Current IP address, subnet and network gateway settings (can be edited from the web page).

- Network status
  Gives gateway MAC address and Ethernet module software version.
  Current network link status regarding duplex and link speed.



9.4.2 Modbus TCP

For more information about Modbus TCP see **REF 5**.

Modbus TCP is based on version 1.0b, see **REF 5**.

The Modbus TCP protocol in the gateway has been implemented according to Modbus TCP description in **appendix D (page 127).**

Request interval for Modbus TCP shall be minimum 50 ms.

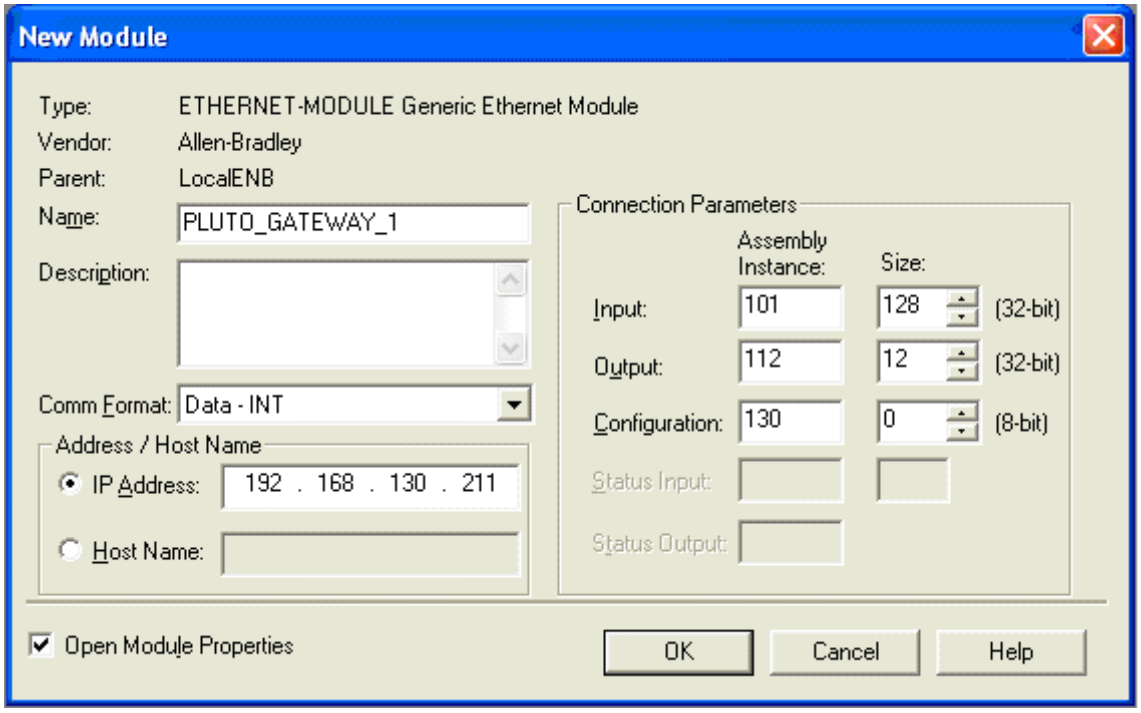
9.4.3 EtherNet/IP (EIP)

For more information about EtherNet/IP (EIP) see **REF 3**.

EtherNet/IP is based on ODVA “CIP” Edition 3.2 and “EtherNet/IP Adaption of CIP” Edition 1.3.

The EtherNet/IP protocol in the gateway has been implemented according to EtherNet/IP object description in **appendix C (page 99)**.

Example of configuration an Allen-Bradley system under I/O configuration and Ethernet add new module for communication of type Generic Ethernet Module,



Important settings are:

- Name of the Ethernet unit which will give names to the controller tags as,
  PLUTO_GATEWAY_1:C control data
  PLUTO_GATEWAY_1:I input data
  PLUTO_GATEWAY_1:O output data
- IP address of the gateway (see chapter 9.3).
- Communication data size format (Comm Format, preferred format is “**Data – INT**”).
- Input assembly instance number and size.
- Output assembly instance number and size.
- Configuration assembly instance number and size.
- Requested Packet Interval (RPI).
- Set configuration data.

Input assembly setting

If only input data is used the size can be any of the three showed in the table. If output data is used or will maybe be used in future the size of INT shall be used.

Input data	Instance number	Instance size		
		Data - SINT	**Data – INT**	Data – DINT
Status Only	100	4	2	1
Data Only	101	256	128	64
Status and Data	102	260	130	65

Data structure for each instance is like table below. For detailed information about each part see chapter 4.1 (status), 4.2 (Data from Pluto) and 4.3 (Additional Data from Pluto). The table below indicates on which byte/word the data is located in depending on data type and used assembly number (no mapping for DINT data have been shown in this table).

Data	Data - SINT (byte)			**Data - INT (word)**		
	100	101	102	100	101	102
Status	0 – 3	-	0 – 3	0 – 1	-	0 – 1
Data Pluto 0	-	0 – 3	4 – 7	-	0 – 1	2 – 3
Data Pluto 1	-	4 – 7	8 – 11	-	2 – 3	4 – 5
Data Pluto 2	-	8 – 11	12 – 15	-	4 – 5	6 – 7
Data Pluto 3	-	12 – 15	16 – 19	-	6 – 7	8 – 9
Data Pluto 4	-	16 – 19	20 – 23	-	8 – 9	10 – 11
Data Pluto 5	-	20 – 23	24 – 27	-	10 – 11	12 – 13
Data Pluto 6	-	24 – 27	28 – 31	-	12 – 13	14 – 15
Data Pluto 7	-	28 – 31	32 – 35	-	14 – 15	16 – 17
Data Pluto 8	-	32 – 35	36 – 39	-	16 – 17	18 – 19
Data Pluto 9	-	36 – 39	40 – 43	-	18 – 19	20 – 21
Data Pluto 10	-	40 – 43	44 – 47	-	20 – 21	22 – 23
Data Pluto 11	-	44 – 47	48 – 51	-	22 – 23	24 – 25
Data Pluto 12	-	48 – 51	52 – 55	-	24 – 25	26 – 27
Data Pluto 13	-	52 – 55	56 – 59	-	26 – 27	28 – 29
Data Pluto 14	-	56 – 59	60 – 63	-	28 – 29	30 – 31
Data Pluto 15	-	60 – 63	64 – 67	-	30 – 31	32 – 33
Data Pluto 16	-	64 – 67	68 – 71	-	32 – 33	34 – 35
Data Pluto 17	-	68 – 71	72 – 75	-	34 – 35	36 – 37
Data Pluto 18	-	72 – 75	76 – 79	-	36 – 37	38 – 39
Data Pluto 19	-	76 – 79	80 – 83	-	38 – 39	40 – 41
Data Pluto 20	-	80 – 83	84 – 87	-	40 – 41	42 – 43
Data Pluto 21	-	84 – 87	88 – 91	-	42 – 43	44 – 45
Data Pluto 22	-	88 – 91	92 – 95	-	44 – 45	46 – 47
Data Pluto 23	-	92 – 95	96 – 99	-	46 – 47	48 – 49
Data Pluto 24	-	96 – 99	100–103	-	48 – 49	50 – 51
Data Pluto 25	-	100–103	104–107	-	50 – 51	52 – 53
Data Pluto 26	-	104–107	108–111	-	52 – 53	54 – 55
Data Pluto 27	-	108–111	112–115	-	54 – 55	56 – 57
Data Pluto 28	-	112–115	116–119	-	56 – 57	58 – 59
Data Pluto 29	-	116–119	120–123	-	58 – 59	60 – 61
Data Pluto 30	-	120–123	124–127	-	60 – 61	62 – 63
Data Pluto 31	-	124–127	128–131	-	62 – 63	64 – 65
Additional Data 00	-	128–131	132–135	-	64 – 65	66 – 67
Additional Data 01	-	132–135	136–139	-	66 – 67	68 – 69
Additional Data 02	-	136–139	140–143	-	68 – 69	70 – 71
Additional Data 03	-	140–143	144–147	-	70 – 71	72 – 73

Data	Data - SINT (byte)			**Data - INT (word)**		
	100	101	102	100	101	102
Additional Data 04	-	144–147	148–151	-	72 – 73	74 – 75
Additional Data 05	-	148–151	152–155	-	74 – 75	76 – 77
Additional Data 06	-	152–155	156–159	-	76 – 77	78 – 79
Additional Data 07	-	156–159	160–163	-	78 – 79	80 – 81
Additional Data 08	-	160–163	164–167	-	80 – 81	82 – 83
Additional Data 09	-	164–167	168–171	-	82 – 83	84 – 85
Additional Data 10	-	168–171	172–175	-	84 – 85	86 – 86
Additional Data 11	-	172–175	176–179	-	86 – 86	88 – 89
Additional Data 12	-	176–179	180–183	-	88 – 89	90 – 91
Additional Data 13	-	180–183	184–187	-	90 – 91	92 – 93
Additional Data 14	-	184–187	188–191	-	92 – 93	94 – 95
Additional Data 15	-	188–191	192–195	-	94 – 95	96 – 97
Additional Data 16	-	192–195	196–199	-	96 – 97	98 – 99
Additional Data 17	-	196–199	200–203	-	98 – 99	100–101
Additional Data 18	-	200–203	204–207	-	100–101	102–103
Additional Data 19	-	204–207	208–211	-	102–103	104–105
Additional Data 20	-	208–211	212–215	-	104–105	106–107
Additional Data 21	-	212–215	216–219	-	106–107	108–109
Additional Data 22	-	216–219	220–223	-	108–109	110–111
Additional Data 23	-	220–223	224–227	-	110–111	112–113
Additional Data 24	-	224–227	228–231	-	112–113	114–115
Additional Data 25	-	228–231	232–235	-	114–115	116–117
Additional Data 26	-	232–235	236–239	-	116–117	118–119
Additional Data 27	-	236–239	240–243	-	118–119	120–121
Additional Data 28	-	240–243	244–247	-	120–121	122–123
Additional Data 29	-	244–247	248–251	-	122–123	124–125
Additional Data 30	-	248–251	252–255	-	124–125	126–127
Additional Data 31	-	252–255	256–259	-	126–127	128–129

Output assembly setting

It is recommended only to use INT data for output data because output data is 16-bits registers.
For description of “Data to Pluto” structure see chapter 4.4.

Input data	Instance number	Instance size		
		Data - SINT	**Data – INT**	Data – DINT
Data to Pluto (Output data)	112	-	12	-
Input only (No data to Pluto)	128	0	0	0

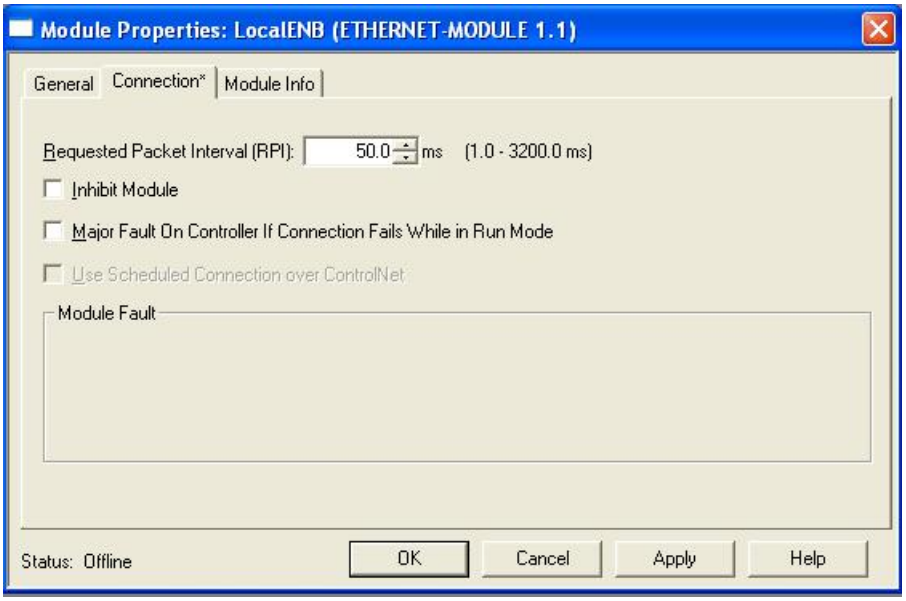
Configuration assembly setting

There is no configuration data so size is zero.

Input data	Instance number	Instance size		
		Data - SINT	**Data – INT**	Data – DINT
Configuration data	130	0	0	0

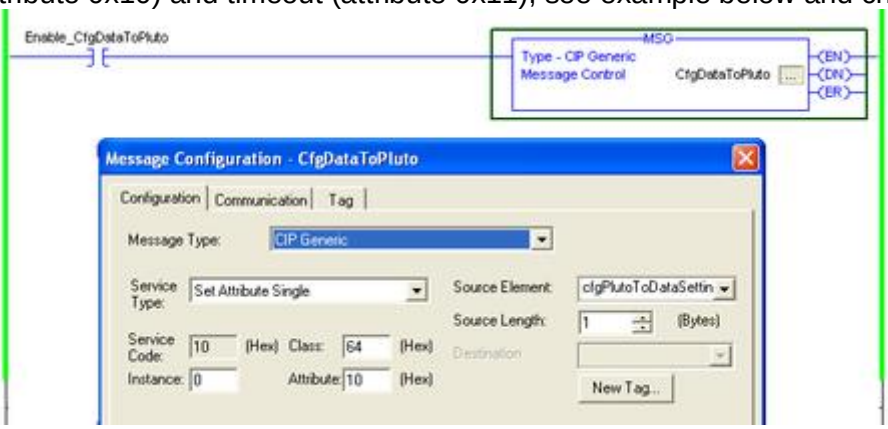
Requested Packet Interval (RPI)

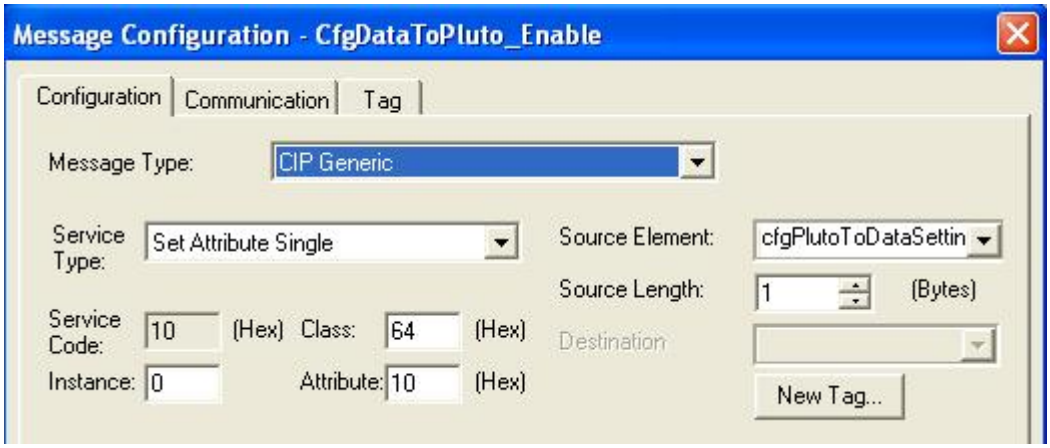
Under the connection tab the Requested Packet Interval (RPI) shall set to be **minimum 50 ms**.



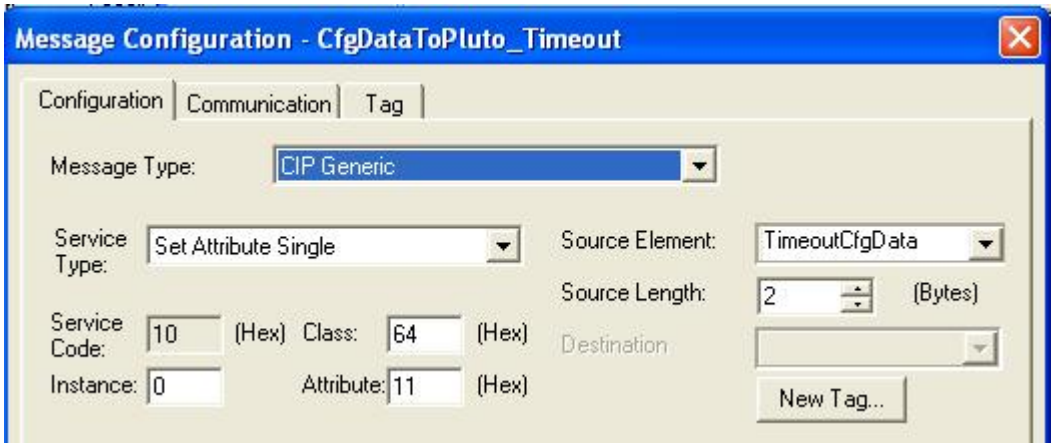
Set configuration data

After the PLC have got connection and/or done a reconnection to the gateway the PLC can/shall send configuration messages to the gateway if needed. It is possible to send configuration data by using message blocks. Configuration settings are related to “Data to Pluto” information to enable packet area (attribute 0x10) and timeout (attribute 0x11), see example below and chapter 4.4.





Example of setup message. Set enable bits by a write to attribute 0x10.



Example of setup message. Set timeout value by a write to attribute 0x11.

There is also configuration get/set for additional data via the same feature but using other attribute numbers in the message. For more information see appendix and also chapter 4.3.

9.4.4 PROFINET

For more information about PROFINET see **REF 2**.

To configure the PROFINET gateway the unit shall be updated with **both** correct TCP/IP address setting and the PNIO Device Name. These settings are then used in the PLC system running PROFINET to connect to the gateway.

The gateway do not support TCP/IP address setting from PNIO Device Name.

9.4.4.1 Configuration file

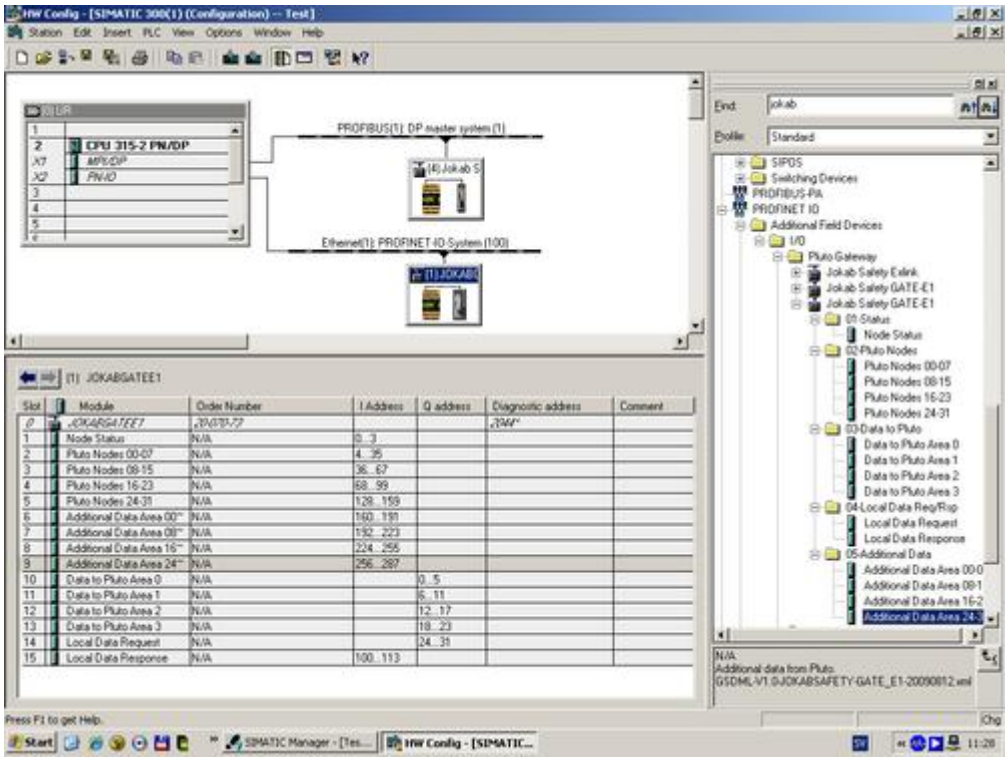
The configuration of the gateway is described in a GSDML-file (PROFINET GSD-file) which shall be loaded into the controlling PLC system. This configuration file will control how to use the gateway. It has the possibility to add modules depending on what's needed by the user. Following modules can be used (added to user configuration). For information about module and data see **appendix E (page 134)**.

Slot	Name
1	Node Status
2	Pluto Nodes 00 – 07
3	Pluto Nodes 08 – 15
4	Pluto Nodes 16 – 23
5	Pluto Nodes 24 – 31
6	Additional Data 00 – 07
7	Additional Data 08 – 15
8	Additional Data 16 – 23
9	Additional Data 24 – 31
10	Data to Pluto Area 0
11	Data to Pluto Area 1
12	Data to Pluto Area 2
13	Data to Pluto Area 3
14	Local Data Request
15	Local Data Response

Note that each module has a fixed slot location in the configuration.

9.4.4.2 Siemens configuration

After adding the GSDML file for PROFINET into the Siemens system the configuration is similar to PROFIBUS by using the hardware configuration tool. The picture below show two units on the PROFIBUS and two units on the PROFINET (lower two units).



During configuration it's important to check,

- Hardware configuration tool shall have correct IP-address and PNIO Device Name for each gateway added to the PROFINET system.

- Each gateway shall have correct IP-address and PNIO Device Name according to the setting hardware configuration settings. How to set IP-address and PNIO Device Name on the gateway see page 58.

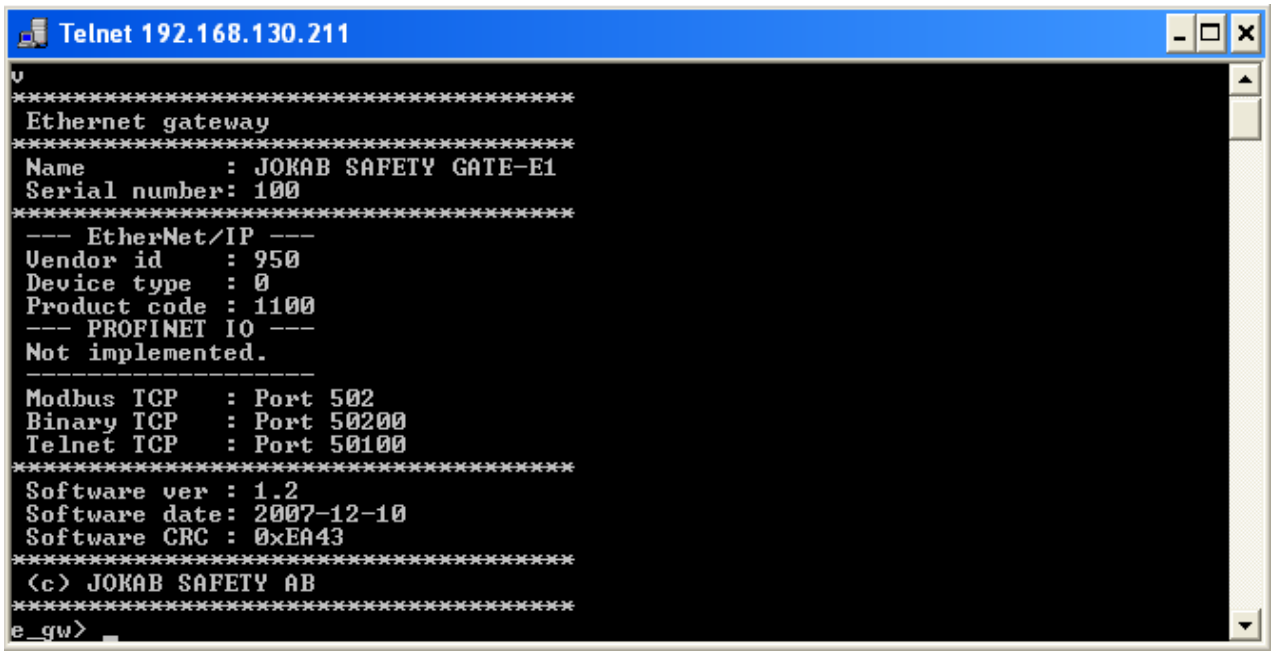
- In hardware configuration tool the IO cycle time shall be set to 64 ms or higher value.

- In hardware configuration add the needed modules, note that each module have fixed slot in the unit.

9.4.5 Terminal ASCII TCP server

The gateway has a terminal server similar to a telnet server. By using a telnet client and connecting to this server the client has a parallel connection to the hardware terminal port. The only difference is that all commands need to be exit with the CR (Enter) button.

The port number for this server is 50100.



9.4.6 Binary TCP server

Note: This protocol is unique and usage of it shall be avoided. It may not be supported in the future!

The binary TCP server is using a special binary protocol. The frame of the protocol will be described within this chapter. The data within the frame is the same as the Modbus TCP protocol description in annex D.

The port number for this server is 50200.

The data frame sent to and received from the binary TCP server is described in the table below,

Address	Data Name	Data Type
1	Slave address	UINT
2, 3, 4…	Data	UINT

Slave address is written as 0xFFxx where XX are the slave address in hexadecimal value.

Example below shows a client sending a local data request,

0xFF 0x02 0x00 0x03 0x00 0x00 0x00 0x01 0x00 0x11

Address	Data Name	Data
1	Slave address 0x02	0xFF02
2	Data flag (length)	0x0003
3	Pluto station id 0	0x0000
4	Data Type 1	0x0001
5	Address 17 (0x11)	0x0011

Only slave address 1, 2, 3 and 4 can be sent via the binary TCP server. Trying to send other slave address messages will result in the data being buffered and will throw off the receive buffer. See below paragraph for further information.

With the binary TCP server, data needs to be sent with the correct data length. If additional data is sent with the request, the extra data will be buffered and used the next time the server receives data. If invalid data has been sent, the user will not receive the correct response data or the command will not happen. The binary TCP server will need to be reset. To reset this state and clear the receive buffer, the client needs to disconnect and reconnect.

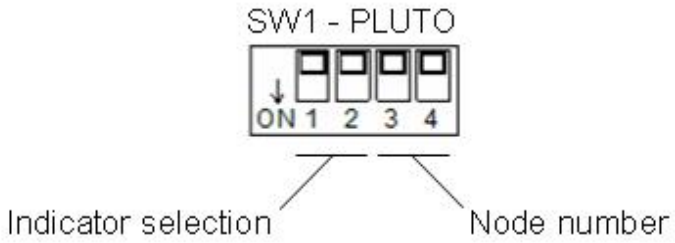
Slave address 33 will be sent by the gateway when Pluto status is changed. Slave address 34 and 35 will be sent if received correct and complete request messages.

9.5 Indicators

There are two status LED indicators for the network part on the GATE-E1.

9.5.1 Indicator selection

Via the DIP switch part 1 and 2 the user can select which protocol will be indicated on the two LED's.



SW1

1	2	Protocol	Remark
0	0	Modbus TCP	When operational is indicated on the LED the gateway has at least one Modbus TCP client connected.
0	1	EtherNet/IP	-
1	0	PROFINET	-
1	1	-	-

9.5.2 Module Status

The LED named “Mod Status” will indicate the module status of the gateway.

LED		Modbus TCP	EtherNet/IP	PROFINET
OFF		Unit off	Unit off	Unit off
GREEN flashing		Standby	Standby	-
GREEN steady		Operational	Operational	Operational
RED flashing		Minor fault	Minor fault	-
RED steady		Major fault	Major fault	-
GREEN/RED flashing		Start-up/Test	Start-up/Test	Start-up/Test

9.5.3 Network Status

The LED named “Net Status” will indicate the network status of the gateway Ethernet protocol.

LED		Modbus TCP	EtherNet/IP	PROFINET
OFF		Unit off	Unit off	Unit off
GREEN flashing		No connection	No connection	No connection
GREEN steady		Connected	Connected	Connected
RED flashing		-	Connection timeout	-
RED steady		-	Duplicate IP	-
GREEN/RED flashing		Start-up/Test	Start-up/Test	Start-up/Test

9.5.4 Module and Network Status

If both the Module and Network Status LED is **off** there is a major fault in the Ethernet module within the gateway.

9.6 Verification of configuration

Via the serial port function (see chapter 10) there is a possibility to check the status of the gateway and also see which configuration the gateway has received from the master. This information is printed when doing the “bw” command, see below.

```
e_gw> bw
-----------------------------
IP Address  : 192.168.130.212
Subnet Mask : 255.255.255.0
Gateway     : 0.0.0.0
MAC Address : 00-40-9D-2B-F6-6C
Link Speed  : 10 MBit[Half Duplex]
Software ver: 01.08 (see w command)
-----------------------------
Connection Status
LED MS/NS for : EtherNet/IP
EtherNet/IP   : OPERATIONAL, CONNECTED
PROFINET      : -, -
Modbus/TCP    : 0 users
ASCII server  : 0 users
Binary server : 0 users
-----------------------------
PLC OUTPUT DATA :
  Enabled To PLUTO package 0-3 : - - - -, Timeout 0 ms, Update 100 ms.
ADDITIONAL DATA CONFIGURATION :
 Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type | Area Pluto IO-type
  00    00  USER:01 |  01    00  USER:02 |  02    00  USER:03 |  03    00  USER:04
  04    00  ErrCode |  08    23  ASIsafe
-----------------------------
e_gw>
```

The first part views the IP address configuration and status information.

The second part views the selected LED status indication at “LED MS/NS for:” and the status information for each protocol server within the unit.

The last part views the current configuration from the master. The “PLC OUTPUT DATA” is related to “Data to Pluto”. Here is a parameter “Update 100 ms” which is a fixed time for update of data to Pluto system. The “ADDITIONAL DATA CONFIGURATION” part views the configuration of additional data from Pluto if used.

10 Serial port Functions

10.1 Connection

The gateway has a serial port for debugging and software updating. The connector is the same as for the Pluto units.
Also the communication parameters are the same so it is possible to use the terminal window function within Pluto Manager.

Communication parameters are:

Speed:	57.6 kbits
Bits:	8
Parity:	none
Handshake:	none

10.2 Serial port communication

Via the serial port it is possible to communicate with a PC and a terminal program as “Terminal window” in Pluto Manager or Hyper term in Windows. The commands are similar to the commands for Pluto by using “Terminal window”. With these commands it is possible to read the status of the variables of the connected Pluto units and some local information.

By boot of the gateway or by typing “v” the below information is shown (DeviceNet),

```
dnet_gw> v
**********************************
 DeviceNet gateway
**********************************
 Name         : GATE-D2
 Vendor id    : 950
 Device type  : 0
 Product code : 1000
 Serial number: 5009
**********************************
 Software ver : 3.0
 Software date: 2010-12-12
 Software CRC : 0x7ECA
**********************************
dnet_gw>
```

By typing “h” for help all available commands are listed (DeviceNet),

```
dnet_gw> h

gw  <a>     Read gateway SysRegister value
i   <p.a>   Read Input status
q   <p.a>   Read Output status
g   <p.a>   Read Globle mem status
m   <p.a>   Read Memory bit status
sm  <p.a>   Read SysMem bit status
r   <p.a>   Read Register value
sr  <p.a>   Read SysRegister value
s   <p.a>   Read Sequence step
t   <p.a>   Read Timer value
sp  <p.a>   Read SysParameter value
to  <a.r>   Read <To PLUTO> data <area.reg>
area <a>    Read Pluto IO Data Area <area>

<p.a> : [pluto[.address]]

boot   Reboot all PLUTO units
reset  Restart gateway unit
rp     Restart PLUTO bus
rw     Restart DeviceNet
bs     Bus status PLUTO bus
bw     Bus status gateway bus

gs     Gateway MACID/baudrate setup (PROG MODE)
cs     Configuration setup (PROG MODE)
fs     Filter setup (BRIDGE MODE)
def    Restore factory setting

time   Gateway run time [sec]
v      Gateway version
h      Help text
dnet_gw>
```

11 Technical data

11.1 GATE-P1/P2

Pluto bus	CAN (with galvanic insulation)
Pluto bus speed	100, 200, 250, 400, 500, 800 and 1000 kbit/s (automatic speed detection)

PROFIBUS	RS485 (with galvanic insulation)
PROFIBUS Speed	9.6, 19.2, 45.45, 93.75, 187.5, 500 kbits and 1.5, 3, 6, 12 Mbits (automatic baud rate detection)
PROFIBUS Protocol	DP Slave, DP-V0
PROFIBUS Address	DIP switch setting

Connectors	Upper side 3-pole terminal for Pluto bus (included) Front side standard 9-pole PROFIBUS connector Bottom side2-pole terminal for 24VDC (included)
Status indication	Pluto bus status via LED (Pluto Bus) PROFIBUS status indication

DC power	24 VDC, -15 % to +20 %
Power consumption at 24V	< 100 mA (recommended external fuse ≤ 6 A)

Enclosure	Width = 22.5 mm, height = 101 mm and depth = 119 mm
Mounting	35 mm DIN-rail

Ambient air temperature	-10°C to + 55ºC
Temperature, transportation and storage	-25°C to + 55ºC
Humidity	EN 60 204-1 50 % at 40ºC (ex 90 % at 20ºC)
Degree of protection	Enclosure IP 20 - IEC 60 529 Terminals IP 20 - IEC 60 529

11.2 GATE-D1/D2

Pluto bus	CAN (with galvanic insulation)
Pluto bus speed	100, 200, 250, 400, 500, 800 and 1000 kbit/s (automatic speed detection)

DeviceNet	CAN (with galvanic insulation)
DeviceNet Speed	125, 250 and 500 kbits (DIP switch setting)
DeviceNet Protocol	ODVA version 2.0
DeviceNet Address	DIP switch setting

Connectors	Upper side 3-pole terminal for Pluto bus (included) Front side with standard 5-pole DeviceNet connector (included) Bottom side2-pole terminal for 24VDC (included)
Status indication	Pluto bus status via LED (Pluto Bus) DeviceNet MNS status indication

DC power	24 VDC, -15 % to +20 %
Power consumption at 24V	< 100 mA (recommended external fuse ≤ 6 A)

Enclosure	Width = 22.5 mm, height = 101 mm and depth = 119 mm
Mounting	35 mm DIN-rail

Ambient air temperature	-10°C to + 55ºC
Temperature, transportation and storage	-25°C to + 55ºC
Humidity	EN 60 204-1 50 % at 40ºC (ex 90 % at 20ºC)
Degree of protection	Enclosure IP 20 - IEC 60 529 Terminals IP 20 - IEC 60 529

11.3 GATE-C1/C2

Pluto bus	CAN (with galvanic insulation)
Pluto bus speed	100, 200, 250, 400, 500, 800 and 1000 kbit/s (automatic speed detection)

CANopen	CAN (with galvanic insulation)
CANopen Speed	125, 250 and 500 kbits (DIP switch setting) 10, 20, 50, 100, 125, 250, 500, 800 and 1000 kbits (software setting)
CANopen Protocol	Version 4.02 of the CiA Draft Standard 301
CANopen Address	DIP switch setting (software setting)

Connectors	Upper side 3-pole terminal for Pluto bus (included) Front side with standard 5-pole CANopen connector (included) Bottom side2-pole terminal for 24VDC (included)
Status indication	Pluto bus status via LED (Pluto Bus) CANopen status indication

DC power	24 VDC, -15 % to +20 %
Power consumption at 24V	< 100 mA (recommended external fuse ≤ 6 A)

Enclosure	Width = 22.5 mm, height = 101 mm and depth = 119 mm
Mounting	35 mm DIN-rail

Ambient air temperature	-10°C to + 55ºC
Temperature, transportation and storage	-25°C to + 55ºC
Humidity	EN 60 204-1 50 % at 40ºC (ex 90 % at 20ºC)
Degree of protection	Enclosure IP 20 - IEC 60 529 Terminals IP 20 - IEC 60 529

11.4 GATE-E1/E2

Pluto bus	CAN (with galvanic insulation)
Pluto bus speed	100, 200, 250, 400, 500, 800 and 1000 kbit/s (automatic speed detection)

Ethernet	10/100 Mbit/s Half and full duplex
Ethernet protocol	Status from and to Pluto safety-PLC - EtherNet/IP (EIP) - PROFINET - Modbus TCP - Binary server (TCP/IP) **Note:** The recommendation is to only use one to the above protocol at a give time. Gateway status and IP address configuration - Webb server - Terminal server (TCP/IP)
EtherNet/IP	According to ODVA “CIP Edition 3.2” and “EtherNet/IP Adaptation of CIP Edition 1.3” with minimum RPI value of 50 ms.
PROFINET	According to PNIO with minimum poll time of 64 ms.
Modbus TCP	According to Modbus organisation version 1.0b (20 messages per second).
Binary server (TCP/IP)	Simple TCP/IP protocol for status from and to the Pluto system.
Web server	For some status information and IP address setting.
Terminal server (TCP/IP)	Simple server with same commands as via the serial port of the unit.
IP address	Static setting via web server or via serial port.
Gateway configuration	Via EtherNet/IP, PROFINET, Modbus TCP or the binary TCP/IP server.

Connectors	Upper side 3-pole terminal for Pluto bus (included) Front Ethernet connection via RJ-45 (screened cable cat5e FTP) Bottom side2-pole terminal for 24VDC (included)
Status indication	Pluto bus status via LED (Pluto Bus) Ethernet module status via LED (Mod Status) Ethernet network status via LED (Net Status)

DC power	24 VDC, -15 % to +20 %
Power consumption at 24V	< 150 mA (recommended external fuse ≤ 6 A)

Enclosure	Width = 35 mm, height = 101 mm and depth = 119 mm
Mounting	35 mm DIN-rail

Ambient air temperature	-10°C to + 55ºC
Temperature, transportation and storage	-25°C to + 55ºC
Humidity	EN 60 204-1 50 % at 40ºC (ex 90 % at 20ºC)
Degree of protection	Enclosure IP 20 - IEC 60 529 Terminals IP 20 - IEC 60 529

1 Appendix A, DeviceNet EDS description

This is a description of the different data types that are used in the documentation of the object model. These are standard definitions of the Open DeviceNet Vendor Association (ODVA). ODVA is an independent supplier organization that manages the DeviceNet specification and supports the worldwide growth of DeviceNet.

1.1 Definitions

The following table describes the used data types.

USINT	Unsigned Short Integer (8-bit)
UINT	Unsigned Integer (16-bit)
UDINT	Unsigned Double Integer (32-bit)
STRING	Character String (1 byte per character)
BYTE	Bit String (8-bits)
WORD	Bit String (16-bits)
DWORD	Bit String (32-bits)

1.2 Reference Documents

- ODVA Volume 1: CIP Common Specification, Edition 2.0 ©2004 ODVA
- ODVA Volume 3: DeviceNet Adaptation of CIP, Edition 1.0 ©2004 ODVA

1.3 Identity Object (01_{HEX} - 1 Instance)

Class Attributes (Instance 0)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Revision	UINT	1	Get

Instance Attributes (Instance 1)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Vendor Number	UINT	950_{DEC}	Get
2	Device Type	UINT	00_{HEX}	Get
3	Product Code Number	UINT	1000_{DEC}	Get
4	Product Major Revision Product Minor Revision	USINT USINT	2 3	Get
5	Status	WORD	See Below	Get
6	Serial Number	UDINT	Unique 32 Bit Value	Get
7	Product Name	String of USINT	Jokab Safety GATE-D1 or GATE-D2	Get

Status Word

Bit	Bit = 0	Bit = 1
0	Not Owned	Owned
1	Unused	Unused
2	No configuration since the last Out of Box reset.	The device has been configured since the last Out of Box reset.
3 – 15	Unused	Unused

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get_Attribute_Single
05_{HEX}	No	Yes	Reset

1.4 Message Router Object (02_{HEX} - 0 Instances)

No attributes are accessible over the network.

1.5 DeviceNet Object (03_{HEX} - 1 Instance)

Class Attributes (Instance 0)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Revision	UINT	2	Get

Instance Attributes (Instance 1)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Mac ID	USINT	63	Get / Set[1]
2	Baud Rate	USINT	0	Get / Set[2]
5	**Structure of:** Allocation Choice Byte Master's Mac ID	 BYTE USINT	 0xFF 0	 Get Get
6	MAC ID Switch Changed	BOOL	0	Get
7	Baud Rate Switch Changed	BOOL	0	Get
8	MAC ID Switch Value	USINT	63	Get
9	Baud Rate Switch Value	USINT	0	Get

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get_Attribute_Single
10_{HEX}	No	Yes	Set_Attribute_Single

[1] Settable when baud rate switch are set into **PROG MODE**, see page 35.
[2] Settable when baud rate switch are set into **PROG MODE**, see page 35.

1.6 Assembly Object (04_{HEX} – 5 Instances)

Class Attributes (Instance 0)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Revision	UINT	2	Get
2	Max Instance	UINT	113	Get

Input Instance Attributes (Instances 100 - 112)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
3	Input Data	USINT[4-132]	0	Get

Input Instance 100 – 4 Bytes (Node Status Only)

Bytes	Class, Instance, Attribute	Description
0 – 3	0x64, 0x00, 11	Node Status

Input Instance 101 – 128 Bytes (Node Data Only)

Old configuration gives “Combined 32 Bit Data – Node x”.
New configuration gives “Pluto IO 32 Bit Data – Area x”.

Bytes	Class, Instance, Attribute	Description
0 – 3	0x64, 0x01, 0x04	Combined 32 Bit Data – Node 0 Pluto IO 32 Bit Data – Area 0
4 – 7	0x64, 0x02, 0x04	Combined 32 Bit Data – Node 1 Pluto IO 32 Bit Data – Area 1
8 – 11	0x64, 0x03, 0x04	Combined 32 Bit Data – Node 2 Pluto IO 32 Bit Data – Area 2
12 – 15	0x64, 0x04, 0x04	Combined 32 Bit Data – Node 3 Pluto IO 32 Bit Data – Area 3
…		
112 – 115	0x64, 0x1D, 0x04	Combined 32 Bit Data – Node 28 Pluto IO 32 Bit Data – Area 028
116 – 119	0x64, 0x1E, 0x04	Combined 32 Bit Data – Node 29 Pluto IO 32 Bit Data – Area 29
120 – 123	0x64, 0x1F, 0x04	Combined 32 Bit Data – Node 30 Pluto IO 32 Bit Data – Area 30
124 – 127	0x64, 0x20, 0x04	Combined 32 Bit Data – Node 31 Pluto IO 32 Bit Data – Area 31

Input Instance 102 – 132 Bytes (Node Status and Data)

Old configuration gives “Combined 32 Bit Data – Node x”.
New configuration gives “Pluto IO 32 Bit Data – Area x”.

Bytes	Class, Instance, Attribute	Description
0 – 3	0x64, 0x00, 0x0B	Node Status
4 – 7	0x64, 0x01, 0x04	Combined 32 Bit Data – Node 0 Pluto IO 32 Bit Data – Area 0
8 – 11	0x64, 0x02, 0x04	Combined 32 Bit Data – Node 1 Pluto IO 32 Bit Data – Area 1
12 – 15	0x64, 0x03, 0x04	Combined 32 Bit Data – Node 2 Pluto IO 32 Bit Data – Area 2
16 – 19	0x64, 0x04, 0x04	Combined 32 Bit Data – Node 3 Pluto IO 32 Bit Data – Area 3
…		
116 – 119	0x64, 0x1D, 0x04	Combined 32 Bit Data – Node 28 Pluto IO 32 Bit Data – Area 028
120 – 123	0x64, 0x1E, 0x04	Combined 32 Bit Data – Node 29 Pluto IO 32 Bit Data – Area 29
124 – 127	0x64, 0x1F, 0x04	Combined 32 Bit Data – Node 30 Pluto IO 32 Bit Data – Area 30
128 – 132	0x64, 0x20, 0x04	Combined 32 Bit Data – Node 31 Pluto IO 32 Bit Data – Area 31

Output Instance Attributes (Instances 112 - 113)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
3	Output Data	USINT[0-24]	0	Get

Output Instance 112 – 0 Bytes (No Data)

Bytes	Class, Instance, Attribute	Description
N/A	N/A	No Data

Output Instance 113 – 24 Bytes (Data to Pluto)

Bytes	Class, Instance, Attribute	Description
0 – 5	0x64, 0x00, 20	Data to Pluto area 0
6 – 11	0x64, 0x00, 21	Data to Pluto area 1
12 – 17	0x64, 0x00, 22	Data to Pluto area 2
18 – 23	0x64, 0x00, 23	Data to Pluto area 3

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get_Attribute_Single
10_{HEX}	No	Yes	Set_Attribute_Single

1.7 Connection Object (05_{HEX} – 3 - 8 Instances)

Class Attributes (Instance 0)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Revision	UINT	1	Get

Instance Attributes (Instances 1-2) Explicit, Polled I/O

Attribute ID	Name	DeviceNet Data Type	Data Value		Access Rule
			Instance 1*	Instance 2**	
1	State	USINT	0 = NonExistent 3 = Established 5 = Deferred Delete	0 = NonExistent 1 = Configuring 3 = Established 4 = Timed Out	Get
2	Instance Type	USINT	0	1	Get
3	Transport Trigger	USINT	83_{HEX}	82_{HEX}	Get
4	Produced Connection ID	UINT	$10xxxxxx011_{BIN}$ xxxxxx = Node Address	$01111xxxxxx_{BIN}$ xxxxxx = Node Address	Get
5	Consumed Connection ID	UINT	$10xxxxxx100_{BIN}$ xxxxxx = Node Address	$10xxxxxx100_{BIN}$ xxxxxx = Node Address	Get
6	Initial Comm. Character	USINT	21_{HEX}	01_{HEX}	Get
7	Produced Connection Size	UINT	VARIES	4	Get
8	Consumed Connection Size	UINT	VARIES	4	Get
9	Expected Packet Rate	UINT	2500 msec	0	Get / Set
12	Watchdog Timeout Action	USINT	4 = Deferred Delete	0 = Timeout	Get / Set
13	Produced Connection Path Length	UINT	0	6	Get
14	Produced Connection Path	USINT Array	NULL	20h 04h 24h 64h 30h 03h	Get
15	Consumed Connection Path Length	UINT	0	6	Get
16	Consumed Connection Path	USINT Array	NULL	20h 04h 24h 70h 30h 03h	Get

*Instance 1 is an Explicit Message Connection.
**Instance 2 is a Polled I/O Message Connection.

Instance Attributes (Instance 4) Change of State/Cyclic Acknowledged

Attribute ID	Name	DeviceNet Data Type	Data Value		Access Rule
			Change of State	Cyclic	
1	State	USINT	0 = NonExistent 1 = Configuring 3 = Established 4 = Timed Out	0 = NonExistent 1 = Configuring 3 = Established 4 = Timed Out	Get
2	Instance Type	USINT	1	1	Get
3	Transport Trigger	USINT	12_{HEX}	02_{HEX}	Get
4	Produced Connection ID	UINT	$01101xxxxxx_{BIN}$ xxxxxx = Node Address	$01101xxxxxx_{BIN}$ xxxxxx = Node Address	Get
5	Consumed Connection ID	UINT	$10xxxxxx010_{BIN}$ xxxxxx = Node Address	$10xxxxxx010_{BIN}$ xxxxxx = Node Address	Get
6	Initial Comm. Character	USINT	01_{HEX}	01_{HEX}	Get
7	Produced Connection Size	UINT	4	4	Get
8	Consumed Connection Size	UINT	0	0	Get
9	Expected Packet Rate	UINT	0	0	Get / Set
12	Watchdog Timeout Action	USINT	0 = Timeout	0 = Timeout	Get / Set
13	Produced Connection Path Length	UINT	6	6	Get
14	Produced Connection Path	USINT Array	20h 04h 24h 64h 30h 03h	20h 04h 24h 64h 30h 03h	Get
15	Consumed Connection Path Length	UINT	4	4	Get
16	Consumed Connection Path	USINT Array	20h 2Bh 24h 01h	20h 2Bh 24h 01h	Get

Instance Attributes (Instance 4) Change of State/Cyclic Unacknowledged

Attribute ID	Name	DeviceNet Data Type	Data Value		Access Rule
			Change of State	Cyclic	
1	State	USINT	0 = NonExistent 1 = Configuring 3 = Established 4 = Timed Out	0 = NonExistent 1 = Configuring 3 = Established 4 = Timed Out	Get
2	Instance Type	USINT	1	1	Get
3	Transport Trigger	USINT	12_{HEX}	02_{HEX}	Get
4	Produced Connection ID	UINT	$01101xxxxxx_{BIN}$ xxxxxx = Node Address	$01101xxxxxx_{BIN}$ xxxxxx = Node Address	Get
5	Consumed Connection ID	UINT	$FFFF_{HEX}$	$FFFF_{HEX}$	Get
6	Initial Comm. Character	USINT	$0F_{HEX}$	$0F_{HEX}$	Get
7	Produced Connection Size	UINT	4	4	Get
8	Consumed Connection Size	UINT	0	0	Get
9	Expected Packet Rate	UINT	0	0	Get / Set
12	Watchdog Timeout Action	USINT	0 = Timeout	0 = Timeout	Get / Set
13	Produced Connection Path Length	UINT	0	0	Get
14	Produced Connection Path	USINT Array	NULL	NULL	Get
15	Consumed Connection Path Length	UINT	0	0	Get
16	Consumed Connection Path	USINT Array	NULL	NULL	Get

UCMM Instances (Instance ID's 10-255, Max 5 at a time – if supported)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
			Instance 1*	
1	State	USINT	0 = NonExistent 3 = Established 5 = Deferred Delete	Get
2	Instance Type	USINT	0	Get
3	Transport Trigger	USINT	83_{HEX}	Get
4	Produced Connection ID	UINT	Varies	Get
5	Consumed Connection ID	UINT	Varies	Get
6	Initial Comm. Character	USINT	Varies	Get
7	Produced Connection Size	UINT	VARIES	Get
8	Consumed Connection Size	UINT	VARIES	Get
9	Expected Packet Rate	UINT	2500 msec	Get / Set
12	Watchdog Timeout Action	USINT	4 = Deferred Delete	Get / Set
13	Produced Connection Path Length	UINT	0	Get
14	Produced Connection Path	USINT Array	NULL	Get
15	Consumed Connection Path Length	UINT	0	Get
16	Consumed Connection Path	USINT Array	NULL	Get

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get_Attribute_Single
10_{HEX}	No	Yes	Set_Attribute_Single

1.8 Acknowledge Handler Object ($2B_{HEX}$ - 1 Instance)

Class Attributes (Instance 0)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Revision	UINT	1	Get

Instance Attributes (Instance 1)

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Acknowledge Timer	UINT	16	Get/Set
2	Retry Limit	USINT	1	Get/Set
3	COS Producing Connection Instance	UINT	4	Get

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get Attribute Single
10_{HEX}	No	Yes	Set Attribute Single

1.9 Application Object (64_{HEX} - 32 Instances)

Class Attributes (Instance 0)

"Expected Node Configuration", only Pluto global data from selected Pluto units.
- Set expected node bitmap according to wanted Pluto units in IO data.

"Additional Data Configuration", gives a flexible IO area allocation with the possibility to get additional data from Pluto units.
- Allocate each wanted Pluto IO Data Area with Pluto number and IO-type.
  Preferred is **first** write Pluto number and **second** IO-type for **each** used Pluto IO Data Area.
  When using "Additional Data" the PLC **shall never write data** the "Expected Nodes Bitmap" parameter.

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Revision	UINT	1	Get
10	Expected Nodes Bitmap	DWORD	0	Get/Set
11	Node Status Bitmap	DWORD	0	Get
12	Data To Pluto 1	UINT[3]	0,0,0	Get/Set
13	Data To Pluto 2	UINT[3]	0,0,0	Get/Set
14	Data To Pluto 3	UINT[3]	0,0,0	Get/Set
15	Data To Pluto 4	UINT[3]	0,0,0	Get/Set
16	Enable Data to Pluto (0 = Disabled; 1 = Enabled) Bit 0 – Data To Pluto 1 Bit 1 – Data To Pluto 2 Bit 2 – Data To Pluto 3 Bit 3 – Data To Pluto 4	BYTE	0	Get/Set
17	Data To Pluto Timeout (ms)	UINT16	0	Get/Set
18	Data To Pluto Cycle Time (ms)	BYTE	100	Get/Set
19	Gateway node address (0-16) 0 = DIP-switch setting 1 = Node address 0 2 = Node address 1 … 16 = Node address 15	BYTE	0	Get/Set
20	Input Assembly Instance 0 = Assembly Instance 100 1 = Assembly Instance 101 2 = Assembly Instance 102 3-255 = INVALID	USINT	0	Get/Set
21	Output Assembly Instance 0 = Assembly Instance 112 1 = Assembly Instance 113 2-255 = INVALID	USINT	0	Get/Set
22	Input Assembly Size	INT	4	Get
23	Output Assembly Size	INT	0	Get

Pluto IO Data Area Allocation (new configuration), for description of IO-type value see chapter 4.3.

Attribute ID	Name	Data Type	Data Value	Access Rule
30	Pluto IO Data Area 00, Node (0-31)	BYTE	0	Get/Set
31	Pluto IO Data Area 00, IO-type	BYTE	0	Get/Set
32	Pluto IO Data Area 01, Node (0-31)	BYTE	0	Get/Set
33	Pluto IO Data Area 01, IO-type	BYTE	0	Get/Set
34	Pluto IO Data Area 02, Node (0-31)	BYTE	0	Get/Set
35	Pluto IO Data Area 02, IO-type	BYTE	0	Get/Set
36	Pluto IO Data Area 03, Node (0-31)	BYTE	0	Get/Set
37	Pluto IO Data Area 03, IO-type	BYTE	0	Get/Set
38	Pluto IO Data Area 04, Node (0-31)	BYTE	0	Get/Set
39	Pluto IO Data Area 04, IO-type	BYTE	0	Get/Set
40	Pluto IO Data Area 05, Node (0-31)	BYTE	0	Get/Set
41	Pluto IO Data Area 05, IO-type	BYTE	0	Get/Set
42	Pluto IO Data Area 06, Node (0-31)	BYTE	0	Get/Set
43	Pluto IO Data Area 06, IO-type	BYTE	0	Get/Set
44	Pluto IO Data Area 07, Node (0-31)	BYTE	0	Get/Set
45	Pluto IO Data Area 07, IO-type	BYTE	0	Get/Set
46	Pluto IO Data Area 08, Node (0-31)	BYTE	0	Get/Set
47	Pluto IO Data Area 08, IO-type	BYTE	0	Get/Set
48	Pluto IO Data Area 09, Node (0-31)	BYTE	0	Get/Set
49	Pluto IO Data Area 09, IO-type	BYTE	0	Get/Set
50	Pluto IO Data Area 10, Node (0-31)	BYTE	0	Get/Set
51	Pluto IO Data Area 10, IO-type	BYTE	0	Get/Set
52	Pluto IO Data Area 11, Node (0-31)	BYTE	0	Get/Set
53	Pluto IO Data Area 11, IO-type	BYTE	0	Get/Set
54	Pluto IO Data Area 12, Node (0-31)	BYTE	0	Get/Set

55	Pluto IO Data Area 12, IO-type	BYTE	0	Get/Set
56	Pluto IO Data Area 13, Node (0-31)	BYTE	0	Get/Set
57	Pluto IO Data Area 13, IO-type	BYTE	0	Get/Set
58	Pluto IO Data Area 14, Node (0-31)	BYTE	0	Get/Set
59	Pluto IO Data Area 14, IO-type	BYTE	0	Get/Set
60	Pluto IO Data Area 15, Node (0-31)	BYTE	0	Get/Set
61	Pluto IO Data Area 15, IO-type	BYTE	0	Get/Set
62	Pluto IO Data Area 16, Node (0-31)	BYTE	0	Get/Set
63	Pluto IO Data Area 16, IO-type	BYTE	0	Get/Set
64	Pluto IO Data Area 17, Node (0-31)	BYTE	0	Get/Set
65	Pluto IO Data Area 17, IO-type	BYTE	0	Get/Set
66	Pluto IO Data Area 18, Node (0-31)	BYTE	0	Get/Set
67	Pluto IO Data Area 18, IO-type	BYTE	0	Get/Set
68	Pluto IO Data Area 19, Node (0-31)	BYTE	0	Get/Set
69	Pluto IO Data Area 19, IO-type	BYTE	0	Get/Set
70	Pluto IO Data Area 20, Node (0-31)	BYTE	0	Get/Set
71	Pluto IO Data Area 20, IO-type	BYTE	0	Get/Set
72	Pluto IO Data Area 21, Node (0-31)	BYTE	0	Get/Set
73	Pluto IO Data Area 21, IO-type	BYTE	0	Get/Set
74	Pluto IO Data Area 22, Node (0-31)	BYTE	0	Get/Set
75	Pluto IO Data Area 22, IO-type	BYTE	0	Get/Set
76	Pluto IO Data Area 23, Node (0-31)	BYTE	0	Get/Set
77	Pluto IO Data Area 23, IO-type	BYTE	0	Get/Set
78	Pluto IO Data Area 24, Node (0-31)	BYTE	0	Get/Set
79	Pluto IO Data Area 24, IO-type	BYTE	0	Get/Set
80	Pluto IO Data Area 25, Node (0-31)	BYTE	0	Get/Set
81	Pluto IO Data Area 25, IO-type	BYTE	0	Get/Set
82	Pluto IO Data Area 26, Node (0-31)	BYTE	0	Get/Set
83	Pluto IO Data Area 26, IO-type	BYTE	0	Get/Set
84	Pluto IO Data Area 27, Node (0-31)	BYTE	0	Get/Set
85	Pluto IO Data Area 27, IO-type	BYTE	0	Get/Set
86	Pluto IO Data Area 28, Node (0-31)	BYTE	0	Get/Set
87	Pluto IO Data Area 28, IO-type	BYTE	0	Get/Set
88	Pluto IO Data Area 29, Node (0-31)	BYTE	0	Get/Set
89	Pluto IO Data Area 29, IO-type	BYTE	0	Get/Set
90	Pluto IO Data Area 30, IO-type	BYTE	0	Get/Set
91	Pluto IO Data Area 30, Node (0-31)	BYTE	0	Get/Set
92	Pluto IO Data Area 31, IO-type	BYTE	0	Get/Set
93	Pluto IO Data Area 31, Node (0-31)	BYTE	0	Get/Set

Instance Attributes (Instances 1-32)

Explicit read of the Pluto node global data values (instance equal Pluto node number + 1).

Attribute ID	Name	DeviceNet Data Type	Data Value	Access Rule
1	Input Bits	WORD	0	Get
2	Output Bits	BYTE	0	Get
3	Global Bits	WORD	0	Get
4	Combined 32 Bits	DWORD	0	Get

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get Attribute Single
10_{HEX}	Yes	No	Set Attribute Single
32_{HEX}	No	Yes	Read Local Pluto Data
33_{HEX}	No	Yes	Read Local Gateway Data
34_{HEX}	No	Yes	Serial Pass Through

Read Local Pluto Data (0x32)

Instance value 1 – 32 is equal to Pluto address 0 – 31.

Request Service Code Data

Bytes	Description
0 – 1	Address value

Local data from Pluto can be of 3 different types. The local address data shall be coded with type information in bits 14 and 15 according to the table below.

Bit 15	Bit 14	Data type	Address (range)/value
0	**0**	**Global memory (0/1)**	**(0 – 31)**
0	**1**	**Local memory (0/1)**	**(0 – 1024) | 0x4000**
1	**0**	**Local register (uint16)**	**(0 – 300) | 0x8000**
1	**1**	**Local parameter (uint32)**	**(0 – 999) | 0xC000**

Response Service Code Data

The respond value is always converted to UINT32 value even if the requested data is retrieving Boolean or UINT16 value.

Bytes	Description
0 – 3	UINT32 Data Value

Read Local Gateway Data (0x33)

Instance value is currently not used.

Request Service Code Data

Bytes	Description
0 – 1	Local Address

Response Service Code Data

Bytes	Description
0 – 3	UINT32 Data Value

Serial Pass Through (0x34)

Request Service Code Data

Bytes	Description
0 – 5	Anything

Response Service Code Data

Bytes	Description
0 – 5	Anything

2 Appendix B, CANopen EDS description

2.1 Object Dictionary

Index	Name	Sub Index	Description	Data Type	Data Value	Access Rule
0x1000	Device Type	0x00	N/A	UINT32	0	Get
0x1001	Error Register	0x00	N/A	UINT8	0	Get
0x1018	Identity Object	0x00	Number of sub-index entries	UINT8	4	Get
		0x01	Vendor ID	UINT32	0x000001B0	Get
		0x02	Product Code	UINT32	1000	Get
		0x03	Revision Number	UINT32	1	Get
		0x04	Serial Number	UINT32	0xnnnnnnnn	Get
0x1002	MFR Status Register	0x00	32-bitmap of Pluto Nodes Online	UINT32	0x00000000	Get
0x1017	Producer Heartbeat Time	0x00	Producer Heartbeat Time [ms]	UINT16	0	Get/Set
0x1400	RPDO Comm Param 1	0x00	Number of sub-index entries	UINT8	2	Get
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x200	Get / Set[3]
		0x02	Transmission Type	UINT8	255	Get / Set
0x1401	RPDO Comm Param 2	0x00	Number of sub-index entries	UINT8	2	Get
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x300	Get / Set[3]
		0x02	Transmission Type	UINT8	255	Get / Set
0x1402	RPDO Comm Param 3	0x00	Number of sub-index entries	UINT8	2	Get
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x400	Get / Set[3]
		0x02	Transmission Type	UINT8	255	Get / Set
0x1403	RPDO Comm Param 4	0x00	Number of sub-index entries	UINT8	2	Get
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x500	Get / Set[3]
		0x02	Transmission Type	UINT8	255	Get / Set
0x1600	RPDO 1 Mapping	0x00	Number of used map entries	UINT8	3	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x62000110	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x62000210	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x62000310	Get
0x1601	RPDO 2 Mapping	0x00	Number of used map entries	UINT8	3	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x62010110	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x62010210	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x62010310	Get
0x1602	RPDO 3 Mapping	0x00	Number of used map entries	UINT8	3	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x62020110	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x62020210	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x62020310	Get
0x1603	RPDO 4 Mapping	0x00	Number of used map entries	UINT8	3	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x62030110	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x62030210	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x62030310	Get
0x1800	TPDO Comm Param 1	0x00	Number of sub-index entries	UINT8	5	Get
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x180	Get / Set[4]
		0x02	Transmission Type	UINT8	255	Get / Set
		0x03	Inhibit Time [ms]	UINT16	50	Get / Set
		0x04	Reserved	UINT8	0	Get
		0x05	Event Timer [ms]	UINT16	30000	Get / Set
0x1801	TPDO Comm Param 2	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x280	Get / Set
		0x02-0x05	See TPDO Comm Param 1			
0x1802	TPDO Comm Param 3	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x380	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			

[3] Bit 31 is settable (0 = enable, 1 = disable RPDO)
[4] Bit 31 is settable (0 = enable, 1 = disable TPDO)

Index	Name	Sub Index	Description	Data Type	Data Value	Access Rule
0x1803	TPDO Comm Param 4	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x480	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x1804	TPDO Comm Param 5	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x1A0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x1805	TPDO Comm Param 6	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x2A0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x1806	TPDO Comm Param 7	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x3A0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x1807	TPDO Comm Param 8	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x4A0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x1808	TPDO Comm Param 9	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x1C0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x1809	TPDO Comm Param 10	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x2C0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x180A	TPDO Comm Param 11	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x3C0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x180B	TPDO Comm Param 12	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x4C0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x180C	TPDO Comm Param 13	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x1E0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x180D	TPDO Comm Param 14	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x2E0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x180E	TPDO Comm Param 15	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x3E0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x180F	TPDO Comm Param 16	0x00	See TPDO Comm Param 1			
		0x01	COB-ID Used by PDO	UINT32	Node ID + 0x4E0	Get / Set[4]
		0x02-0x05	See TPDO Comm Param 1			
0x1A00	TPDO 1 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60000108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60000208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60000308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60000408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60010108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60010208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60010308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60010408	Get
0x1A01	TPDO 2 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60020108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60020208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60020308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60020408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60030108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60030208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60030308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60030408	Get

Index	Name	Sub Index	Description	Data Type	Data Value	Access Rule
0x1A02	TPDO 3 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60040108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60040208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60040308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60040408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60050108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60050208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60050308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60050408	Get
0x1A03	TPDO 4 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60060108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60060208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60060308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60060408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60070108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60070208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60070308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60070408	Get
0x1A04	TPDO 5 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60080108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60080208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60080308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60080408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60090108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60090208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60090308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60090408	Get
0x1A05	TPDO 6 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x600A0108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x600A0208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x600A0308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x600A0408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x600B0108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x600B0208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x600B0308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x600B0408	Get
0x1A06	TPDO 7 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x600C0108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x600C0208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x600C0308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x600C0408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x600D0108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x600D0208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x600D0308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x600D0408	Get
0x1A07	TPDO 8 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x600E0108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x600E0208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x600E0308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x600E0408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x600F0108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x600F0208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x600F0308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x600F0408	Get
0x1A08	TPDO 9 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60100108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60100208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60100308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60100408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60110108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60110208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60110308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60110408	Get
0x1A09	TPDO 10 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60120108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60120208	Get

Index	Name	Sub Index	Description	Data Type	Data Value	Access Rule
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60120308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60120408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60130108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60130208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60130308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60130408	Get
0x1A0A	TPDO 11 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60140108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60140208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60140308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60140408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60150108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60150208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60150308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60150408	Get
0x1A0B	TPDO 12 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60160108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60160208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60160308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60160408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60170108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60170208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60170308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60170408	Get
0x1A0C	TPDO 13 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x60180108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x60180208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x60180308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x60180408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x60190108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x60190208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x60190308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x60190408	Get
0x1A0D	TPDO 14 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x601A0108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x601A0208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x601A0308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x601A0408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x601B0108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x601B0208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x601B0308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x601B0408	Get
0x1A0E	TPDO 15 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x601C0108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x601C0208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x601C0308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x601C0408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x601D0108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x601D0208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x601D0308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x601D0408	Get
0x1A0F	TPDO 16 Mapping	0x00	Number of used map entries	UINT8	8	Get
		0x01	Map Entry 1 (Index, Subindex, # bits)	UINT32	0x601E0108	Get
		0x02	Map Entry 2 (Index, Subindex, # bits)	UINT32	0x601E0208	Get
		0x03	Map Entry 3 (Index, Subindex, # bits)	UINT32	0x601E0308	Get
		0x04	Map Entry 4 (Index, Subindex, # bits)	UINT32	0x601E0408	Get
		0x05	Map Entry 5 (Index, Subindex, # bits)	UINT32	0x601F0108	Get
		0x06	Map Entry 6 (Index, Subindex, # bits)	UINT32	0x601F0208	Get
		0x07	Map Entry 7 (Index, Subindex, # bits)	UINT32	0x601F0308	Get
		0x08	Map Entry 8 (Index, Subindex, # bits)	UINT32	0x601F0408	Get
0x2000	Pass Through Request	0x00	Number of sub-index entries	UINT8	7	Get
		0x01	Pass Through PLUTO ID (0-31)	UINT8	0	Get / Set
		0x02	Pass Through Request Data [0]	UINT8	0	Get / Set
		0x03	Pass Through Request Data [1]	UINT8	0	Get / Set
		0x04	Pass Through Request Data [2]	UINT8	0	Get / Set
		0x05	Pass Through Request Data [3]	UINT8	0	Get / Set

Index	Name	Sub Index	Description	Data Type	Data Value	Access Rule
		0x06	Pass Through Request Data [4]	UINT8	0	Get / Set
		0x07	Pass Through Request Data [5]	UINT8	0	Get / Set
0x2001	Pass Through Response	0x00	Number of sub-index entries **Note** Sub items are updated with new values if any of the ”Pass Through Request” sub items have been set	UINT8	7	Get
		0x01	Error Code 0x01 : Wait for response. 0x02 : OK, Data Value valid. 0x04 : Timeout. 0x08 : Bad Parameter.	UINT8	0	Get
		0x02	Pass Through Response Data [0]	UINT8	0	Get
		0x03	Pass Through Response Data [1]	UINT8	0	Get
		0x04	Pass Through Response Data [2]	UINT8	0	Get
		0x05	Pass Through Response Data [3]	UINT8	0	Get
		0x06	Pass Through Response Data [4]	UINT8	0	Get
		0x07	Pass Through Response Data [5]	UINT8	0	Get
0x2002	Data To Pluto Settings	0x00	Number of sub-index entries	UINT8	3	Get
		0x01	Enable Data To Pluto Areas 0-3	UINT8	0	Get / Set
		0x02	Data To Pluto Timeout (ms)	UINT16	0	Get / Set
		0x03	Cycle Update Time (ms)	UINT8	100	Get / Set
0x2003	Read Local Data Request	0x00	Number of sub-index entries	UINT8	2	Get
		0x01	PLUTO ID (0-31), Gateway 0x00FF).	UINT8	0	Get / Set
		0x02	Address Global memory data 0-31. Local memory data (0-1024)|0x4000 Local register data (0-300)|0x8000 Local parameter data (0-999)|0xC000	UINT16	0	Get / Set
0x2004	Read Local Data Response	0x00	Number of sub-index entries **Note** Sub items are updated with new values if any of the ”Read Local Data Request” sub items have been set.	UINT8	2	Get
		0x01	Error Code 0x01 : Wait for response. 0x02 : OK, Data Value valid. 0x04 : Timeout. 0x08 : Bad Parameter.	UINT8	0	Get
		0x02	Data Value	UINT32	0	Get
0x2005	TPDO configuration	0x00	Number of sub-index entries	UINT8	4	Get
		0x01	Transmission Type	UINT8	255	Get / Set
		0x02	Inhibit Time [ms]	UINT16	50	Get / Set
		0x03	Event Timer [ms]	UINT16	30000	Get / Set
		0x04	Enable TPDO. Easy command to read/set if TPDO is enabled, see “TPDO Comm Param x” and value “COB-ID Used by PDO”. When enable TPDO configuration the above value are used to enable each TPDO. Bit 0: Enable TPDO1 Bit 2: Enable TPDO2 Bit 3: Enable TPDO3 Bit 4: Enable TPDO4 Bit 5: Enable TPDO5 Bit 6: Enable TPDO6 Bit 7: Enable TPDO7 Bit 8: Enable TPDO8 Bit 9: Enable TPDO9 Bit 10: Enable TPDO10 Bit 11: Enable TPDO11 Bit 12: Enable TPDO12 Bit 13: Enable TPDO13	UINT16	0	Get / Set

Index	Name	Sub Index	Description	Data Type	Data Value	Access Rule
			Bit 14: Enable TPDO14 Bit 15: Enable TPDO15			
0x2006	Gateway configuration	0x00	Number of sub-index entries	UINT8	1	Get
		0x01	Pluto gateway node number. 0: node number read from DIP-switch. 1: PLC set node number 0. 2: PLC set node number 1. 3: PLC set node number 2. 4: PLC set node number 3. 5: PLC set node number 4. 6: PLC set node number 5. 7: PLC set node number 6. 8: PLC set node number 7. 9: PLC set node number 8. 10: PLC set node number 9. 11: PLC set node number 10. 12: PLC set node number 11. 13: PLC set node number 12. 14: PLC set node number 13. 15: PLC set node number 14. 16: PLC set node number 15.	UNIT8	0	Get / Set
0x2010	Additional data configuration	0x00	Number of sub-index entries	UINT8	1	Get
		0x01	Get: report number of configured TPDO's for additional data. If zero additional data not configured. Set: if value not zero the current additional data configuration will be cleared.	UINT8	0	Get / Set
0x2011	Additional data TPDO1	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 0. High byte: Pluto number 0 – 31. Low byte: IO type 0 – 255. Example 0x016F is 01 => Pluto 1 6F => 111 decimal => global data	UINT16	0	Get / Set
		0X02	Additional data configuration area 1.	UINT16	0	Get / Set
0x2012	Additional data TPDO2	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 2.	UINT16	0	Get / Set
		0X02	Additional data configuration area 3.	UINT16	0	Get / Set
0x2013	Additional data TPDO3	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 4.	UINT16	0	Get / Set
		0X02	Additional data configuration area 5.	UINT16	0	Get / Set
0x2014	Additional data TPDO4	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 6.	UINT16	0	Get / Set
		0X02	Additional data configuration area 7.	UINT16	0	Get / Set
0x2015	Additional data TPDO5	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 8.	UINT16	0	Get / Set
		0X02	Additional data configuration area 9.	UINT16	0	Get / Set
0x2016	Additional data TPDO6	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 10.	UINT16	0	Get / Set
		0X02	Additional data configuration area 11.	UINT16	0	Get / Set
0x2017	Additional data TPDO7	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 12.	UINT16	0	Get / Set
		0X02	Additional data configuration area 13.	UINT16	0	Get / Set
0x2018	Additional data TPDO8	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 14.	UINT16	0	Get / Set
		0X02	Additional data configuration area 15.	UINT16	0	Get / Set

Index	Name	Sub Index	Description	Data Type	Data Value	Access Rule
0x2019	Additional data TPDO9	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 16.	UINT16	0	Get / Set
		0X02	Additional data configuration area 17.	UINT16	0	Get / Set
0x201A	Additional data TPDO10	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 18.	UINT16	0	Get / Set
		0X02	Additional data configuration area 19.	UINT16	0	Get / Set
0x201B	Additional data TPDO11	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 20.	UINT16	0	Get / Set
		0X02	Additional data configuration area 21.	UINT16	0	Get / Set
0x201C	Additional data TPDO12	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 22.	UINT16	0	Get / Set
		0X02	Additional data configuration area 23.	UINT16	0	Get / Set
0x201D	Additional data TPDO13	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 24.	UINT16	0	Get / Set
		0X02	Additional data configuration area 25.	UINT16	0	Get / Set
0x201E	Additional data TPDO14	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 26.	UINT16	0	Get / Set
		0X02	Additional data configuration area 27.	UINT16	0	Get / Set
0x201F	Additional data TPDO15	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 28.	UINT16	0	Get / Set
		0X02	Additional data configuration area 29.	UINT16	0	Get / Set
0x2020	Additional data TPDO16	0x00	Number of sub-index entries	UINT8	2	Get
		0X01	Additional data configuration area 30.	UINT16	0	Get / Set
		0X02	Additional data configuration area 31.	UINT16	0	Get / Set
0x6000	Pluto Node 0 / Area 0	0x00	Number of Inputs	UINT8	4	Get
	Depending on configuration if using additional data setup.	0x01	Pluto Inputs 0–7 / Additional 0–7	UINT8	0	Get
		0x02	Pluto Inputs 8–15 / Additional 8–15	UINT8	0	Get
		0x03	Pluto Inputs 16–23 / Additional 16–23	UINT8	0	Get
		0x04	Pluto Inputs 24–31 / Additional 24–31	UINT8	0	Get
0x6001	Pluto Node 1 / Area 1	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6002	Pluto Node 2 / Area 2	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6003	Pluto Node 3 / Area 3	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6004	Pluto Node 4 / Area 4	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6005	Pluto Node 5 / Area 5	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6006	Pluto Node 6 / Area 6	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6007	Pluto Node 7 / Area 7	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6008	Pluto Node 8 / Area 8	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6009	Pluto Node 9 / Area 9	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x600A	Pluto Node 10 / Area 10	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x600B	Pluto Node 11 / Area 11	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x600C	Pluto Node 12 / Area 12	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x600D	Pluto Node 13 / Area 13	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x600E	Pluto Node 14 / Area 14	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x600F	Pluto Node 15 / Area 15	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6010	Pluto Node 16 / Area 16	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6011	Pluto Node 17 / Area 17	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6012	Pluto Node 18 / Area 18	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6013	Pluto Node 19 / Area 19	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6014	Pluto Node 20 / Area 20	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6015	Pluto Node 21 / Area 21	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6016	Pluto Node 22 / Area 22	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6017	Pluto Node 23 / Area 23	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6018	Pluto Node 24 / Area 24	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x6019	Pluto Node 25 / Area 25	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x601A	Pluto Node 26 / Area 26	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x601B	Pluto Node 27 / Area 27	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x601C	Pluto Node 28 / Area 28	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x601D	Pluto Node 29 / Area 29	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x601E	Pluto Node 30 / Area 30	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0x601F	Pluto Node 31 / Area 31	0x00 – 0x04	Same as Pluto Node 0 / Area 0			
0X6200	Data To Pluto Area 0	0x00	Number of Outputs	UINT8	3	Get
		0x01	Data To Pluto bit 0 – 16	UINT16	0	Set
		0x02	Data To Pluto reg 0	UINT16	0	Set

Index	Name	Sub Index	Description	Data Type	Data Value	Access Rule
		0x03	Data To Pluto reg 1	UINT16	0	Set
0X6201	Data To Pluto Area 1	0x00	Number of Outputs	UINT8	3	Get
		0x01	Data To Pluto bit 0 – 16	UINT16	0	Set
		0x02	Data To Pluto reg 0	UINT16	0	Set
		0x03	Data To Pluto reg 1	UINT16	0	Set
0X6202	Data To Pluto Area 2	0x00	Number of Outputs	UINT8	3	Get
		0x01	Data To Pluto bit 0 – 16	UINT16	0	Set
		0x02	Data To Pluto reg 0	UINT16	0	Set
		0x03	Data To Pluto reg 1	UINT16	0	Set
0X6203	Data To Pluto Area 3	0x00	Number of Outputs	UINT8	3	Get
		0x01	Data To Pluto bit 0 – 16	UINT16	0	Set
		0x02	Data To Pluto reg 0	UINT16	0	Set
		0x03	Data To Pluto reg 1	UINT16	0	Set

2.2 CAN ID's

11-bit Can Header (COB ID)	Message Description
0x000	NMT (Network Management)
0x080	SYNC
0x080 + Node ID	Emergency Message
0x100	Time Stamp
0x580 + Node ID	Transmit SDO
0x600 + Node ID	Receive SDO
0x700 + Node ID	NMT Error Control / Heartbeat
	Tx PDO
0x180 + Node ID	Tx PDO 1 – Pluto Inputs (Nodes 0-1)
0x280 + Node ID	Tx PDO 2 – Pluto Inputs (Nodes 2-3)
0x380 + Node ID	Tx PDO 3 – Pluto Inputs (Nodes 4-5)
0x480 + Node ID	Tx PDO 4 – Pluto Inputs (Nodes 6-7)
0x1A0 + Node ID	Tx PDO 5 – Pluto Inputs (Nodes 8-9)
0x2A0 + Node ID	Tx PDO 6 – Pluto Inputs (Nodes 10-11)
0x3A0 + Node ID	Tx PDO 7 – Pluto Inputs (Nodes 12-13)
0x4A0 + Node ID	Tx PDO 8 – Pluto Inputs (Nodes 14-15)
0x1C0 + Node ID	Tx PDO 9 – Pluto Inputs (Nodes 16-17)
0x2C0 + Node ID	Tx PDO 10 – Pluto Inputs (Nodes 18-19)
0x3C0 + Node ID	Tx PDO 11 – Pluto Inputs (Nodes 20-21)
0x4C0 + Node ID	Tx PDO 12 – Pluto Inputs (Nodes 22-23)
0x1E0 + Node ID	Tx PDO 13 – Pluto Inputs (Nodes 24-25)
0x2E0 + Node ID	Tx PDO 14 – Pluto Inputs (Nodes 26-27)
0x3E0 + Node ID	Tx PDO 15 – Pluto Inputs (Nodes 28-29)
0x4E0 + Node ID	Tx PDO 16 – Pluto Inputs (Nodes 30-31)
	Rx PDO
0x200 + Node ID	Rx PDO 1 – Data To Pluto Area 0
0x300 + Node ID	Rx PDO 2 – Data To Pluto Area 1
0x400 + Node ID	Rx PDO 3 – Data To Pluto Area 2
0x500 + Node ID	Rx PDO 4 – Data To Pluto Area 3

3 Appendix C. Object description EtherNet/IP

This is a description of the different data types that are used in the documentation of the object model. These are standard definitions of the Open DeviceNet Vendor Association (ODVA).

3.1 Definitions

The following table has a description of all of the data types used.

USINT	Unsigned Short Integer (8-bit)
UINT	Unsigned Integer (16-bit)
UDINT	Unsigned Double Integer (32-bit)
STRING	Character String (1 byte per character)
BYTE	Bit String (8-bits)
WORD	Bit String (16-bits)
DWORD	Bit String (32-bits)

3.2 Identity Object (01_{HEX} - 1 Instance)

Class Attributes

Attribute ID	Name	Data Type	Data Value	Access Rule
1	Revision	UINT	1	Get

Instance Attributes

Attribute ID	Name	Data Type	Data Value	Access Rule
1	Vendor Number	UINT	950	Get
2	Device Type	UINT	0	Get
3	Product Code Number	UINT	1100	Get
4	Product Major Revision Product Minor Revision	USINT USINT	2 11	Get
5	Status Word (see below for definition)	WORD	See Below	Get
6	Product Serial Number	UDINT	Unique 32 Bit Value	Get
7	Product Name	String of USINT	JOKAB SAFETY GATE-E1 or GATE-E2	Get

Status Word

Bit	Bit = 0	Bit = 1
0	Not Owned	Owned
1	Unused	Unused
2	No configuration since the last Out of Box reset.	The device has been configured since the last Out of Box reset.
3 – 15	Unused	Unused

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get_Attribute_Single
05_{HEX}	No	Yes	Reset

3.3 Message Router Object (02_{HEX})

This object has no supported attributes.

3.4 Assembly Object (04_{HEX} – 5 Instances)

Class Attributes (Instance 0)

Attribute ID	Name	Data Type	Data Value	Access Rule
1	Revision	UINT	2	Get
2	Max Instance	UINT	113	Get

Input Instance Attributes (Instance 100 - 102)

Attribute ID	Name	Data Type	Default Data Value	Access Rule
3	Input Data	USINT [4-132]	0	Get

Input Instance 100 – 4 Bytes (Node Status Only)

For more information about data structure see chapter 6.5.1.

Bytes	Class, Instance, Attribute	Description
0 – 3	0x64, 0x00, 0B	Node Status

Input Instance 101 – 256 Bytes (Node Data Only)

For more information about data structure see chapter 6.5.1.

Bytes	Class, Instance, Attribute	Description
0 – 3	0x64, 0x01, 0x04	Combined 32 Bit Data – Node 0
4 – 7	0x64, 0x02, 0x04	Combined 32 Bit Data – Node 1
…		
120 – 123	0x64, 0x1F, 0x04	Combined 32 Bit Data – Node 30
124 – 127	0x64, 0x20, 0x04	Combined 32 Bit Data – Node 31
128 – 131	0x64, 0x01, 0x0A	Additional Data 00
132 – 135	0x64, 0x02, 0x0A	Additional Data 01
…		
248 – 251	0x64, 0x1F, 0x0A	Additional Data 30
252 – 255	0x64, 0x20, 0x0A	Additional Data 31

Input Instance 102 – 260 Bytes (Node Status and Data)

For more information about data structure see chapter 6.5.1.

Bytes	Class, Instance, Attribute	Description
0 – 3	0x64, 0x00, 0x0B	Node Status
4 – 7	0x64, 0x01, 0x04	Combined 32 Bit Data – Node 0
8 – 11	0x64, 0x02, 0x04	Combined 32 Bit Data – Node 1
…		
124 – 127	0x64, 0x1F, 0x04	Combined 32 Bit Data – Node 30
128 – 131	0x64, 0x20, 0x04	Combined 32 Bit Data – Node 31
132 – 135	0x64, 0x01, 0x0A	Additional Data 00
136 – 139	0x64, 0x02, 0x0A	Additional Data 01
…		
252 – 255	0x64, 0x1F, 0x0A	Additional Data 30
256 – 259	0x64, 0x20, 0x0A	Additional Data 31

Output Instance Attributes (Instance 112)

Attribute ID	Name	Data Type	Default Data Value	Access Rule
3	Output Data	USINT [0-24]	0	Get

Output Instance 112 – 24 Bytes (Data to Pluto)

For more information about data structure see chapter 6.5.2.

Bytes	Class, Instance, Attribute	Description
0 – 5	0x64, 0x00, 0x0C	Data to Pluto area 0
6 – 11	0x64, 0x00, 0x0D	Data to Pluto area 1
12 – 17	0x64, 0x00, 0x0E	Data to Pluto area 2
18 – 23	0x64, 0x00, 0x0F	Data to Pluto area 3

Output Instance 128 (Heartbeat Instance – Input Only)

This instance allows client to monitor input data without providing output data.

Output Instance 129 (Heartbeat Instance – Listen Only)

This instance allows client to monitor input data without providing output data. To utilize this connection type, an owning connection must exist from a second client and the configuration of the connection must match exactly.

Output Instance 130 (Configuration Instance)

This instance allows client to download necessary configuration information to the gateway when the I/O connection is opened. The configuration instance supports 0 – 400 bytes of data. If no configuration data is needed this instance may be omitted.

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get_Attribute_Single
10_{HEX}	No	Yes	Set_Attribute_Single

3.5 Connection Manager Object (06_{HEX})

This object has no attributes.

3.6 TCP Object ($F5_{HEX}$ - 1 Instance)

Class Attributes

Attribute ID	Name	Data Type	Data Value	Access Rule
1	Revision	UINT	1	Get

Instance Attributes

Attribute ID	Name	Data Type	Default Data Value	Access Rule
1	Status[5]	DWORD	1	Get
2	Configuration Capability[6]	DWORD	0	Get
3	Configuration Control[7]	DWORD	0	Get
4	Physical Link Object[8] **Structure of:** Path Size Path	 UINT Array Of WORD	 2 0x20F6 0x2401	Get
5	Interface Configuration[9] **Structure of:** IP Address Network Mask Gateway Address Name Server Name Server 2 Domain Name Size Domain Name	 UDINT UDINT UDINT UDINT UDINT UINT STRING	 0 0 0 0 0 0 0	Get
6	Host Name[10] **Structure of:** Host Name Size Host Name	 UINT STRING	 0 0	Get

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get_Attribute_Single
10_{HEX}	No	Yes	Set_Attribute_Single
01_{HEX}	No	Yes	Get_Attribute_All

[5] See section 5-3.2.2.1 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.
[6] See section 5-3.2.2.2 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.
[7] See section 5-3.2.2.3 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.
[8] See section 5-3.2.2.4 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.
[9] See section 5-3.2.2.5 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.
[10] See section 5-3.2.2.6 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.

3.7 Ethernet Link Object ($F6_{HEX}$ - 1 Instance)

Class Attributes

Attribute ID	Name	Data Type	Data Value	Access Rule
1	Revision	UINT	1	Get

Instance Attributes

Attribute ID	Name	Data Type	Default Data Value	Access Rule
1	Interface Speed[11]	UDINT	100	Get
2	Interface Flags[12]	DWORD	3	Get
3	Physical Address[13]	USINT Array[6]	0	Get

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get_Attribute_Single
01_{HEX}	No	Yes	Get_Attribute_All

[11] See section 5-4.2.2.1 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.
[12] See section 5-4.2.2.2 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.
[13] See section 5-4.2.2.3 of “Volume 2: EtherNet/IP Adaptation of CIP” from ODVA for more details on this attribute.

3.8 Application Object (64_{HEX} - 32 Instances)

Class Attributes (Instance 0)

For more information about “Data to Pluto” structure see chapter 4.4.

Attribute ID	Name	Data Type	Default Data Value	Access Rule
1	Revision	UINT	1	Get
10	Expected Nodes Bitmap **Not used!**	DWORD	0	Get/Set
11	Node Status Bitmap	DWORD	0	Get
12	Data to Pluto 1	UINT[3]	0,0,0	Get/Set
13	Data to Pluto 2	UINT[3]	0,0,0	Get/Set
14	Data to Pluto 3	UINT[3]	0,0,0	Get/Set
15	Data to Pluto 4	UINT[3]	0,0,0	Get/Set
16	Enable Data to Pluto (0 = Disabled; 1 = Enabled) Bit 0 – Data To Pluto 1 Bit 1 – Data To Pluto 2 Bit 2 – Data To Pluto 3 Bit 3 – Data To Pluto 4	BYTE	0	Get/Set
17	Data to Pluto Timeout (ms) 0 = timeout disabled Valid value ≥ 1000 ms.	UINT16	0	Get/Set
18	Data to Pluto Update Time (ms). Value modulus of 4 e.g. 0, 4, 8, 16… 252.	UINT8	100	Get/Set
19	Gateway node address (0-16) 0 = DIP-switch setting 1 = Node address 0 2 = Node address 1 … 16 = Node address 15	UINT8	0	Get/Set

Additional data configuration see chapter 4.3.

Attribute ID	Name	Data Type	Default Data Value	Access Rule
32	Additional Data 00, Node (0-31)	BYTE	0	Get/Set
33	Additional Data 00, IO-type	BYTE	0	Get/Set
34	Additional Data 01, Node (0-31)	BYTE	0	Get/Set
35	Additional Data 01, IO-type	BYTE	0	Get/Set
36	Additional Data 02, Node (0-31)	BYTE	0	Get/Set
37	Additional Data 02, IO-type	BYTE	0	Get/Set
38	Additional Data 03, Node (0-31)	BYTE	0	Get/Set
39	Additional Data 03, IO-type	BYTE	0	Get/Set
40	Additional Data 04, Node (0-31)	BYTE	0	Get/Set
41	Additional Data 04, IO-type	BYTE	0	Get/Set
42	Additional Data 05, Node (0-31)	BYTE	0	Get/Set
43	Additional Data 05, IO-type	BYTE	0	Get/Set
44	Additional Data 06, Node (0-31)	BYTE	0	Get/Set
45	Additional Data 06, IO-type	BYTE	0	Get/Set
46	Additional Data 07, Node (0-31)	BYTE	0	Get/Set
47	Additional Data 07, IO-type	BYTE	0	Get/Set
48	Additional Data 08, Node (0-31)	BYTE	0	Get/Set

49	Additional Data 08, IO-type	BYTE	0	Get/Set
50	Additional Data 09, Node (0-31)	BYTE	0	Get/Set
51	Additional Data 09, IO-type	BYTE	0	Get/Set
52	Additional Data 10, Node (0-31)	BYTE	0	Get/Set
53	Additional Data 10, IO-type	BYTE	0	Get/Set
54	Additional Data 11, Node (0-31)	BYTE	0	Get/Set
55	Additional Data 11, IO-type	BYTE	0	Get/Set
56	Additional Data 12, Node (0-31)	BYTE	0	Get/Set
57	Additional Data 12, IO-type	BYTE	0	Get/Set
58	Additional Data 13, Node (0-31)	BYTE	0	Get/Set
59	Additional Data 13, IO-type	BYTE	0	Get/Set
60	Additional Data 14, Node (0-31)	BYTE	0	Get/Set
61	Additional Data 14, IO-type	BYTE	0	Get/Set
62	Additional Data 15, Node (0-31)	BYTE	0	Get/Set
63	Additional Data 15, IO-type	BYTE	0	Get/Set
64	Additional Data 16, Node (0-31)	BYTE	0	Get/Set
65	Additional Data 16, IO-type	BYTE	0	Get/Set
66	Additional Data 17, Node (0-31)	BYTE	0	Get/Set
67	Additional Data 17, IO-type	BYTE	0	Get/Set
68	Additional Data 18, Node (0-31)	BYTE	0	Get/Set
69	Additional Data 18, IO-type	BYTE	0	Get/Set
70	Additional Data 19, Node (0-31)	BYTE	0	Get/Set
71	Additional Data 19, IO-type	BYTE	0	Get/Set
72	Additional Data 20, Node (0-31)	BYTE	0	Get/Set
73	Additional Data 20, IO-type	BYTE	0	Get/Set
74	Additional Data 21, Node (0-31)	BYTE	0	Get/Set
75	Additional Data 21, IO-type	BYTE	0	Get/Set
76	Additional Data 22, Node (0-31)	BYTE	0	Get/Set
77	Additional Data 22, IO-type	BYTE	0	Get/Set
78	Additional Data 23, Node (0-31)	BYTE	0	Get/Set
79	Additional Data 23, IO-type	BYTE	0	Get/Set
80	Additional Data 24, Node (0-31)	BYTE	0	Get/Set
81	Additional Data 24, IO-type	BYTE	0	Get/Set
82	Additional Data 25, Node (0-31)	BYTE	0	Get/Set
83	Additional Data 25, IO-type	BYTE	0	Get/Set
84	Additional Data 26, Node (0-31)	BYTE	0	Get/Set
85	Additional Data 26, IO-type	BYTE	0	Get/Set
86	Additional Data 27, Node (0-31)	BYTE	0	Get/Set
87	Additional Data 27, IO-type	BYTE	0	Get/Set
88	Additional Data 28, Node (0-31)	BYTE	0	Get/Set
89	Additional Data 28, IO-type	BYTE	0	Get/Set
90	Additional Data 29, Node (0-31)	BYTE	0	Get/Set
91	Additional Data 29, IO-type	BYTE	0	Get/Set
92	Additional Data 30, Node (0-31)	BYTE	0	Get/Set
93	Additional Data 30, IO-type	BYTE	0	Get/Set
94	Additional Data 31, Node (0-31)	BYTE	0	Get/Set
95	Additional Data 31, IO-type	BYTE	0	Get/Set

Instance Attributes (Instances 1-32)

Instance value 1-32 is equal to Pluto station address 0-31.

Attribute ID	Name	Data Type	Default Data Value	Access Rule
1	Input Bits	WORD	0	Get
2	Output Bits	BYTE	0	Get
3	Global Bits	WORD	0	Get
4	Combined 32 Bits	DWORD	0	Get
10	Additional Data 32 Bits	DWORD	0	Get

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$0E_{HEX}$	Yes	Yes	Get Attribute Single
10_{HEX}	Yes	No	Set Attribute Single
32_{HEX}	No	Yes	Read Local Pluto Data
33_{HEX}	No	Yes	Read Local Gateway Data
34_{HEX}	No	Yes	Serial Pass Through

3.8.1 Service Code 0x32

This service code will read local data from the selected Pluto unit.
Instance value 1-32 is equal to Pluto station address 0-31.

Request Service Code Data

Bytes	Description
0 – 1	UINT16, Address value

For more information regarding Pluto address range see chapter 5.4.3.4. Local data from Pluto can be of 3 different types. The local address data shall be coded with type information in bits 14 and 15 of the address value according to table below.

Bit 15	Bit 14	Data Type	Address (range)/value
0	0	Global memory (0/1)	(0 – 31)
0	1	Local memory (0/1)	(0 – 1024) | 0x4000
1	0	Local Register (uint16)	(0 – 300) | 0x8000
1	1	Local Parameter (uint32)	(0 – 999) | 0xC000

Response Service Code Data

The respond value is always a UINT32 value even if the requested data is retrieving Boolean or UINT16 value. These values are converted into UINT32 value.

Bytes	Description
0 – 3	UINT32, Response value

3.8.2 Service Code 0x33

This service code will read local within the gateway (“gw”) registers.

Request Service Code Data

Bytes	Description
0 – 1	UINT16, Address value

Response Service Code Data

The response value is always a UINT32 value.

Bytes	Description
0 – 3	UINT32, Response value

3.8.3 Service Code 0x34

Serial Pass Through is currently **not** implemented.

3.9 PCCC Object (67_{HEX} - 1 Instance)

Class Attributes

No class attributes.

Instance Attributes

No instance attributes.

Common Services

Service Code	Implemented for		Service Name
	Class Level	Instance Level	
$4B_{HEX}$	No	Yes	Execute PCCC Request

Execute PCCC Request (Service Code $4B_{HEX}$)

Allen-Bradley (AB) /Rockwell Automation (RA) devices use the “Execute PCCC Request” service code to communicate with their legacy products like the PLC5E and SLC 5/05. This product emulates a PLC5E, thus enabling communication to legacy AB/RA devices.

Communications via the PCCC Object are connectionless in nature and don’t allow the outputs to leave the safe state. If the Legacy PLC is the only EtherNet/IP Client, a user-defined mechanism must be established for transition out of the safe state.

PCCC Mapping (Read Only Parameters)

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data From Pluto	33	133	2	Node Status	UDINT
Data From Pluto	33	133	4	PLUTO 00	UDINT
Data From Pluto	33	133	6	PLUTO 01	UDINT
Data From Pluto	33	133	8	PLUTO 02	UDINT
Data From Pluto	33	133	10	PLUTO 03	UDINT
Data From Pluto	33	133	12	PLUTO 04	UDINT
Data From Pluto	33	133	14	PLUTO 05	UDINT
Data From Pluto	33	133	16	PLUTO 06	UDINT
Data From Pluto	33	133	18	PLUTO 07	UDINT
Data From Pluto	33	133	20	PLUTO 08	UDINT
Data From Pluto	33	133	22	PLUTO 09	UDINT
Data From Pluto	33	133	24	PLUTO 10	UDINT
Data From Pluto	33	133	26	PLUTO 11	UDINT
Data From Pluto	33	133	28	PLUTO 12	UDINT
Data From Pluto	33	133	30	PLUTO 13	UDINT
Data From Pluto	33	133	32	PLUTO 14	UDINT
Data From Pluto	33	133	34	PLUTO 15	UDINT
Data From Pluto	33	133	36	PLUTO 16	UDINT
Data From Pluto	33	133	38	PLUTO 17	UDINT
Data From Pluto	33	133	40	PLUTO 18	UDINT
Data From Pluto	33	133	42	PLUTO 19	UDINT
Data From Pluto	33	133	44	PLUTO 20	UDINT
Data From Pluto	33	133	46	PLUTO 21	UDINT
Data From Pluto	33	133	48	PLUTO 22	UDINT
Data From Pluto	33	133	50	PLUTO 23	UDINT

Data From Pluto	33	133	52	PLUTO 24	UDINT
Data From Pluto	33	133	54	PLUTO 25	UDINT
Data From Pluto	33	133	56	PLUTO 26	UDINT
Data From Pluto	33	133	58	PLUTO 27	UDINT
Data From Pluto	33	133	60	PLUTO 28	UDINT
Data From Pluto	33	133	62	PLUTO 29	UDINT
Data From Pluto	33	133	64	PLUTO 30	UDINT
Data From Pluto	33	133	66	PLUTO 31	UDINT
Data From Pluto	33	133	68	Additional 00	UDINT
Data From Pluto	33	133	70	Additional 01	UDINT
Data From Pluto	33	133	72	Additional 02	UDINT
Data From Pluto	33	133	74	Additional 03	UDINT
Data From Pluto	33	133	76	Additional 04	UDINT
Data From Pluto	33	133	78	Additional 05	UDINT
Data From Pluto	33	133	80	Additional 06	UDINT
Data From Pluto	33	133	82	Additional 07	UDINT
Data From Pluto	33	133	84	Additional 08	UDINT
Data From Pluto	33	133	86	Additional 09	UDINT
Data From Pluto	33	133	88	Additional 10	UDINT
Data From Pluto	33	133	90	Additional 11	UDINT
Data From Pluto	33	133	92	Additional 12	UDINT
Data From Pluto	33	133	94	Additional 13	UDINT
Data From Pluto	33	133	96	Additional 14	UDINT
Data From Pluto	33	133	98	Additional 15	UDINT
Data From Pluto	33	133	100	Additional 16	UDINT
Data From Pluto	33	133	102	Additional 17	UDINT
Data From Pluto	33	133	104	Additional 18	UDINT
Data From Pluto	33	133	106	Additional 19	UDINT
Data From Pluto	33	133	108	Additional 20	UDINT
Data From Pluto	33	133	110	Additional 21	UDINT
Data From Pluto	33	133	112	Additional 22	UDINT
Data From Pluto	33	133	114	Additional 23	UDINT
Data From Pluto	33	133	116	Additional 24	UDINT
Data From Pluto	33	133	118	Additional 25	UDINT
Data From Pluto	33	133	120	Additional 26	UDINT
Data From Pluto	33	133	122	Additional 27	UDINT
Data From Pluto	33	133	124	Additional 28	UDINT
Data From Pluto	33	133	126	Additional 29	UDINT
Data From Pluto	33	133	128	Additional 30	UDINT
Data From Pluto	33	133	130	Additional 31	UDINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Local Data Response	34	134	2	PLUTO node	UINT
Local Data Response	34	134	3	Data Type	UINT
Local Data Response	34	134	4	Address	UINT
Local Data Response	34	134	5	Error Code	UINT
Local Data Response	34	134	6	Data MSW	UINT
Local Data Response	34	134	7	Data LSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Serial Pass Through Response	35	135	2	PLUTO node	UINT
Serial Pass Through Response	35	135	3	Error Code	UINT
Serial Pass Through Response	35	135	4	Data	UINT
Serial Pass Through Response	35	135	5	Data	UINT
Serial Pass Through Response	35	135	6	Data	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Gateway Configuration	36	136	2	Valid value	UINT
Gateway Configuration	36	136	3	Enable Data To PLUTO	UINT
Gateway Configuration	36	136	4	Valid value	UINT
Gateway Configuration	36	136	5	Data To PLUTO Timeout	UINT
Gateway Configuration	36	136	6	Valid value	UINT
Gateway Configuration	36	136	7	Expected Nodes Bitmap	UDINT (MSW)
Gateway Configuration	36	136	8	Expected Nodes Bitmap	UDINT (LSW)
Gateway Configuration	36	136	9	Valid value	UINT
Gateway Configuration	36	136	10	Additional Data 00	UINT
Gateway Configuration	36	136	11	Additional Data 01	UINT

Gateway Configuration	36	136	12	Additional Data 02	UINT
Gateway Configuration	36	136	13	Additional Data 03	UINT
Gateway Configuration	36	136	14	Additional Data 04	UINT
Gateway Configuration	36	136	15	Additional Data 05	UINT
Gateway Configuration	36	136	16	Additional Data 06	UINT
Gateway Configuration	36	136	17	Additional Data 07	UINT
Gateway Configuration	36	136	18	Additional Data 08	UINT
Gateway Configuration	36	136	19	Additional Data 09	UINT
Gateway Configuration	36	136	20	Additional Data 10	UINT
Gateway Configuration	36	136	21	Additional Data 11	UINT
Gateway Configuration	36	136	22	Additional Data 12	UINT
Gateway Configuration	36	136	23	Additional Data 13	UINT
Gateway Configuration	36	136	24	Additional Data 14	UINT
Gateway Configuration	36	136	25	Additional Data 15	UINT
Gateway Configuration	36	136	26	Additional Data 16	UINT
Gateway Configuration	36	136	27	Additional Data 17	UINT
Gateway Configuration	36	136	28	Additional Data 18	UINT
Gateway Configuration	36	136	29	Additional Data 19	UINT
Gateway Configuration	36	136	30	Additional Data 20	UINT
Gateway Configuration	36	136	31	Additional Data 21	UINT
Gateway Configuration	36	136	32	Additional Data 22	UINT
Gateway Configuration	36	136	33	Additional Data 23	UINT
Gateway Configuration	36	136	34	Additional Data 24	UINT
Gateway Configuration	36	136	35	Additional Data 25	UINT
Gateway Configuration	36	136	36	Additional Data 26	UINT

Gateway Configuration	36	136	37	Additional Data 27	UINT
Gateway Configuration	36	136	38	Additional Data 28	UINT
Gateway Configuration	36	136	39	Additional Data 29	UINT
Gateway Configuration	36	136	40	Additional Data 30	UINT
Gateway Configuration	36	136	41	Additional Data 31	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Status information (host CPU to ExLink)	62	162	2	Host rev	UINT
Status information (host CPU to ExLink)	62	162	3	year	UINT
Status information (host CPU to ExLink)	62	162	4	month	UINT
Status information (host CPU to ExLink)	62	162	5	day	UINT
Status information (host CPU to ExLink)	62	162	6	serial no	UDINT (MSW)
Status information (host CPU to ExLink)	62	162	7	serial no	UDINT (LSW)
Status information (host CPU to ExLink)	62	162	8	PlutoNode	UINT
Status information (host CPU to ExLink)	62	162	9	PlutoBus	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Raw TCP Server out	64	164	2	data	UINT
Raw TCP Server out	64	164	3	data	UINT
Raw TCP Server out	64	164	…	free	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Configuration	65	165	ALL ADDRESSES		

PCCC Mapping (Read/Write Parameters)

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data To Pluto	1	101	1	Length	UINT
Data To Pluto	1	101	2	Area Info	UINT
Data To Pluto	1	101	3	Area 0, Bits	UINT
Data To Pluto	1	101	4	Area 0/Reg 0	UINT
Data To Pluto	1	101	5	Area 0/Reg 1	UINT
Data To Pluto	1	101	6	Area 1, Bits	UINT
Data To Pluto	1	101	7	Area 1/Reg 0	UINT
Data To Pluto	1	101	8	Area 1/Reg 1	UINT
Data To Pluto	1	101	9	Area 2, Bits	UINT
Data To Pluto	1	101	10	Area 2/Reg 0	UINT
Data To Pluto	1	101	11	Area 2/Reg 1	UINT
Data To Pluto	1	101	12	Area 3, Bits	UINT
Data To Pluto	1	101	13	Area 3/Reg 0	UINT
Data To Pluto	1	101	14	Area 3/Reg 1	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Local Data Request	2	102	1	Handshake	UINT
Local Data Request	2	102	2	PLUTO node	UINT
Local Data Request	2	102	3	Data Type	UINT
Local Data Request	2	102	4	Address	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Serial Pass Through Request	3	103	1	Handshake	UINT
Serial Pass Through Request	3	103	2	PLUTO node	UINT
Serial Pass Through Request	3	103	3	Data	UINT
Serial Pass Through Request	3	103	4	Data	UINT
Serial Pass Through Request	3	103	5	Data	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Gateway Configuration	4	104	1	Length	UINT
Gateway Configuration	4	104	2	Enable Data To PLUTO	UINT
Gateway Configuration	4	104	3	Data To PLUTO Timeout	UINT
Gateway Configuration	4	104	4	Expected Nodes Bitmap	UDINT (MSW)
Gateway Configuration	4	104	5	Expected Nodes Bitmap	UDINT (LSW)
Gateway Configuration	4	104	6	Additional Data 00	UINT

Gateway Configuration	4	104	7	Additional Data 01	UINT
Gateway Configuration	4	104	8	Additional Data 02	UINT
Gateway Configuration	4	104	9	Additional Data 03	UINT
Gateway Configuration	4	104	10	Additional Data 04	UINT
Gateway Configuration	4	104	11	Additional Data 05	UINT
Gateway Configuration	4	104	12	Additional Data 06	UINT
Gateway Configuration	4	104	13	Additional Data 07	UINT
Gateway Configuration	4	104	14	Additional Data 08	UINT
Gateway Configuration	4	104	15	Additional Data 09	UINT
Gateway Configuration	4	104	16	Additional Data 10	UINT
Gateway Configuration	4	104	17	Additional Data 11	UINT
Gateway Configuration	4	104	18	Additional Data 12	UINT
Gateway Configuration	4	104	19	Additional Data 13	UINT
Gateway Configuration	4	104	20	Additional Data 14	UINT
Gateway Configuration	4	104	21	Additional Data 15	UINT
Gateway Configuration	4	104	22	Additional Data 16	UINT
Gateway Configuration	4	104	23	Additional Data 17	UINT
Gateway Configuration	4	104	24	Additional Data 18	UINT
Gateway Configuration	4	104	25	Additional Data 19	UINT
Gateway Configuration	4	104	26	Additional Data 20	UINT
Gateway Configuration	4	104	27	Additional Data 21	UINT
Gateway Configuration	4	104	28	Additional Data 22	UINT
Gateway Configuration	4	104	29	Additional Data 23	UINT
Gateway Configuration	4	104	30	Additional Data 24	UINT
Gateway Configuration	4	104	31	Additional Data 25	UINT
Gateway Configuration	4	104	32	Additional Data 26	UINT
Gateway Configuration	4	104	33	Additional Data 27	UINT
Gateway Configuration	4	104	34	Additional Data 28	UINT

Gateway Configuration	4	104	35	Additional Data 29	UINT
Gateway Configuration	4	104	36	Additional Data 30	UINT
Gateway Configuration	4	104	37	Additional Data 31	UINT
Gateway Configuration	4	104	38	Data To Pluto Cycle Time	UINT
Gateway Configuration	4	104	39	Enabel Pluto Status (only PROFINET)	UINT
Gateway Configuration	4	104	40	Enabel Local Data Req/Resp (only PROFINET)	UNIT
Gateway Configuration	4	104	41	Enable Serial Pass Through Req/Resp (only PROFINET)	UINT
Gateway Configuration	4	104	42	Gateway Node Address	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Status information (ExLink to host CPU)	30	130	1	Length	UINT
Status information (ExLink to host CPU)	30	130	2	Modbus/TCP	UINT
Status information (ExLink to host CPU)	30	130	3	EtherNet/IP	UINT
Status information (ExLink to host CPU)	30	130	4	PROFINET	UINT
Status information (ExLink to host CPU)	30	130	5	TCP ASCII	UINT
Status information (ExLink to host CPU)	30	130	6	TCP Binary	UINT
Status information (ExLink to host CPU)	30	130	7	LED Start	UINT
Status information (ExLink to host CPU)	30	130	8	profinetHz	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
TCP ASCII Server in	32	132	1	Length	UINT [>0]
TCP ASCII Server in	32	132	2	data	UINT
TCP ASCII Server in	32	132	3	data	UINT
TCP ASCII Server in	32	132	…	free	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data From Pluto	33	133	1	Length	UINT
Data From Pluto	33	133	2	Node Status MSW	UINT
Data From Pluto	33	133	3	Node Status LSW	UINT
Data From Pluto	33	133	4	Pluto 00 MSW	UINT
Data From Pluto	33	133	5	Pluto 00 LSW	UINT
Data From Pluto	33	133	6	Pluto 01 MSW	UINT
Data From Pluto	33	133	7	Pluto 01 LSW	UINT
Data From Pluto	33	133	8	Pluto 02 MSW	UINT
Data From Pluto	33	133	9	Pluto 02 LSW	UINT
Data From Pluto	33	133	10	Pluto 03 MSW	UINT
Data From Pluto	33	133	11	Pluto 03 LSW	UINT
Data From Pluto	33	133	12	Pluto 04 MSW	UINT
Data From Pluto	33	133	13	Pluto 04 LSW	UINT
Data From Pluto	33	133	14	Pluto 05 MSW	UINT
Data From Pluto	33	133	15	Pluto 05 LSW	UINT
Data From Pluto	33	133	16	Pluto 06 MSW	UINT
Data From Pluto	33	133	17	Pluto 06 LSW	UINT
Data From Pluto	33	133	18	Pluto 07 MSW	UINT
Data From Pluto	33	133	19	Pluto 07 LSW	UINT
Data From Pluto	33	133	20	Pluto 08 MSW	UINT
Data From Pluto	33	133	21	Pluto 08 LSW	UINT
Data From Pluto	33	133	22	Pluto 09 MSW	UINT
Data From Pluto	33	133	23	Pluto 09 LSW	UINT
Data From Pluto	33	133	24	Pluto 10 MSW	UINT
Data From Pluto	33	133	25	Pluto 10 LSW	UINT
Data From Pluto	33	133	26	Pluto 11 MSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data From Pluto	33	133	27	Pluto 11 LSW	UINT
Data From Pluto	33	133	28	Pluto 12 MSW	UINT
Data From Pluto	33	133	29	Pluto 12 LSW	UINT
Data From Pluto	33	133	30	Pluto 13 MSW	UINT
Data From Pluto	33	133	31	Pluto 13 LSW	UINT
Data From Pluto	33	133	32	Pluto 14 MSW	UINT
Data From Pluto	33	133	33	Pluto 14 LSW	UINT
Data From Pluto	33	133	34	Pluto 15 MSW	UINT
Data From Pluto	33	133	35	Pluto 15 LSW	UINT
Data From Pluto	33	133	36	Pluto 16 MSW	UINT
Data From Pluto	33	133	37	Pluto 16 LSW	UINT
Data From Pluto	33	133	38	Pluto 17 MSW	UINT
Data From Pluto	33	133	39	Pluto 17 LSW	UINT
Data From Pluto	33	133	30	Pluto 18 MSW	UINT
Data From Pluto	33	133	41	Pluto 18 LSW	UINT
Data From Pluto	33	133	42	Pluto 19 MSW	UINT
Data From Pluto	33	133	43	Pluto 19 LSW	UINT
Data From Pluto	33	133	44	Pluto 20 MSW	UINT
Data From Pluto	33	133	45	Pluto 20 LSW	UINT
Data From Pluto	33	133	46	Pluto 21 MSW	UINT
Data From Pluto	33	133	47	Pluto 21 LSW	UINT
Data From Pluto	33	133	48	Pluto 22 MSW	UINT
Data From Pluto	33	133	49	Pluto 22 LSW	UINT
Data From Pluto	33	133	40	Pluto 23 MSW	UINT
Data From Pluto	33	133	51	Pluto 23 LSW	UINT
Data From Pluto	33	133	52	Pluto 24 MSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data From Pluto	33	133	53	Pluto 24 LSW	UINT
Data From Pluto	33	133	54	Pluto 25 MSW	UINT
Data From Pluto	33	133	55	Pluto 25 LSW	UINT
Data From Pluto	33	133	56	Pluto 26 MSW	UINT
Data From Pluto	33	133	57	Pluto 26 LSW	UINT
Data From Pluto	33	133	58	Pluto 27 MSW	UINT
Data From Pluto	33	133	59	Pluto 27 LSW	UINT
Data From Pluto	33	133	50	Pluto 28 MSW	UINT
Data From Pluto	33	133	61	Pluto 28 LSW	UINT
Data From Pluto	33	133	62	Pluto 29 MSW	UINT
Data From Pluto	33	133	63	Pluto 29 LSW	UINT
Data From Pluto	33	133	64	Pluto 30 MSW	UINT
Data From Pluto	33	133	65	Pluto 30 LSW	UINT
Data From Pluto	33	133	66	Pluto 31 MSW	UINT
Data From Pluto	33	133	67	Pluto 31 LSW	UINT
Data From Pluto	33	133	68	Additional 00 MSW	UINT
Data From Pluto	33	133	69	Additional 00 LSW	UINT
Data From Pluto	33	133	60	Additional 01 MSW	UINT
Data From Pluto	33	133	71	Additional 01 LSW	UINT
Data From Pluto	33	133	72	Additional 02 MSW	UINT
Data From Pluto	33	133	73	Additional 02 LSW	UINT
Data From Pluto	33	133	74	Additional 03 MSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data From Pluto	33	133	75	Additional 03 LSW	UINT
Data From Pluto	33	133	76	Additional 04 MSW	UINT
Data From Pluto	33	133	77	Additional 04 LSW	UINT
Data From Pluto	33	133	78	Additional 05 MSW	UINT
Data From Pluto	33	133	79	Additional 05 LSW	UINT
Data From Pluto	33	133	80	Additional 06 MSW	UINT
Data From Pluto	33	133	81	Additional 06 LSW	UINT
Data From Pluto	33	133	82	Additional 07 MSW	UINT
Data From Pluto	33	133	83	Additional 07 LSW	UINT
Data From Pluto	33	133	84	Additional 08 MSW	UINT
Data From Pluto	33	133	85	Additional 08 LSW	UINT
Data From Pluto	33	133	86	Additional 09 MSW	UINT
Data From Pluto	33	133	87	Additional 09 LSW	UINT
Data From Pluto	33	133	88	Additional 10 MSW	UINT
Data From Pluto	33	133	89	Additional 10 LSW	UINT
Data From Pluto	33	133	90	Additional 11 MSW	UINT
Data From Pluto	33	133	91	Additional 11 LSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data From Pluto	33	133	92	Additional 12 MSW	UINT
Data From Pluto	33	133	93	Additional 12 LSW	UINT
Data From Pluto	33	133	94	Additional 13 MSW	UINT
Data From Pluto	33	133	95	Additional 13 LSW	UINT
Data From Pluto	33	133	96	Additional 14 MSW	UINT
Data From Pluto	33	133	97	Additional 14 LSW	UINT
Data From Pluto	33	133	98	Additional 15 MSW	UINT
Data From Pluto	33	133	99	Additional 15 LSW	UINT
Data From Pluto	33	133	100	Additional 16 MSW	UINT
Data From Pluto	33	133	101	Additional 16 LSW	UINT
Data From Pluto	33	133	102	Additional 17 MSW	UINT
Data From Pluto	33	133	103	Additional 17 LSW	UINT
Data From Pluto	33	133	104	Additional 18 MSW	UINT
Data From Pluto	33	133	105	Additional 18 LSW	UINT
Data From Pluto	33	133	106	Additional 19 MSW	UINT
Data From Pluto	33	133	107	Additional 19 LSW	UINT
Data From Pluto	33	133	108	Additional 20 MSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data From Pluto	33	133	109	Additional 20 LSW	UINT
Data From Pluto	33	133	110	Additional 21 MSW	UINT
Data From Pluto	33	133	111	Additional 21 LSW	UINT
Data From Pluto	33	133	112	Additional 22 MSW	UINT
Data From Pluto	33	133	113	Additional 22 LSW	UINT
Data From Pluto	33	133	114	Additional 23 MSW	UINT
Data From Pluto	33	133	115	Additional 23 LSW	UINT
Data From Pluto	33	133	116	Additional 24 MSW	UINT
Data From Pluto	33	133	117	Additional 24 LSW	UINT
Data From Pluto	33	133	118	Additional 25 MSW	UINT
Data From Pluto	33	133	119	Additional 25 LSW	UINT
Data From Pluto	33	133	120	Additional 26 MSW	UINT
Data From Pluto	33	133	121	Additional 26 LSW	UINT
Data From Pluto	33	133	122	Additional 27 MSW	UINT
Data From Pluto	33	133	123	Additional 27 LSW	UINT
Data From Pluto	33	133	124	Additional 28 MSW	UINT
Data From Pluto	33	133	125	Additional 28 LSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Data From Pluto	33	133	126	Additional 29 MSW	UINT
Data From Pluto	33	133	127	Additional 29 LSW	UINT
Data From Pluto	33	133	128	Additional 30 MSW	UINT
Data From Pluto	33	133	129	Additional 30 LSW	UINT
Data From Pluto	33	133	130	Additional 31 MSW	UINT
Data From Pluto	33	133	131	Additional 31 LSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Local Data Response	34	134	1	Length	UINT
Local Data Response	34	134	2	Pluto Node	UINT
Local Data Response	34	134	3	Data Type	UINT
Local Data Response	34	134	4	Address	UINT
Local Data Response	34	134	5	Error Code	UINT
Local Data Response	34	134	6	Data MSW	UINT
Local Data Response	34	134	7	Data LSW	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Serial Pass Through Response	35	135	1	Length	UINT
Serial Pass Through Response	35	135	2	Pluto Node	UINT
Serial Pass Through Response	35	135	3	Error Code	UINT
Serial Pass Through Response	35	135	4	Data	UINT
Serial Pass Through Response	35	135	5	Data	UINT
Serial Pass Through Response	35	135	6	Data	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Gateway Configuration	36	136	1	Length	UINT
Gateway Configuration	36	136	2	Valide value	UINT
Gateway Configuration	36	136	3	Enable Data To Pluto	UINT
Gateway Configuration	36	136	4	Valid value	UINT
Gateway Configuration	36	136	5	Data To Pluto Timeout	UINT
Gateway Configuration	36	136	6	Valid value	UINT
Gateway Configuration	36	136	7	Expected Nodes Bitmap	UDINT (MSW)
Gateway Configuration	36	136	8	Expected Nodes Bitmap	UDINT (LSW)
Gateway Configuration	36	136	9	Valid value	UINT

Gateway Configuration	36	136	10	Additional Data 00	UINT
Gateway Configuration	36	136	11	Additional Data 01	UINT
Gateway Configuration	36	136	12	Additional Data 02	UINT
Gateway Configuration	36	136	13	Additional Data 03	UINT
Gateway Configuration	36	136	14	Additional Data 04	UINT
Gateway Configuration	36	136	15	Additional Data 05	UINT
Gateway Configuration	36	136	16	Additional Data 06	UINT
Gateway Configuration	36	136	17	Additional Data 07	UINT
Gateway Configuration	36	136	18	Additional Data 08	UINT
Gateway Configuration	36	136	19	Additional Data 09	UINT
Gateway Configuration	36	136	20	Additional Data 10	UINT
Gateway Configuration	36	136	21	Additional Data 11	UINT
Gateway Configuration	36	136	22	Additional Data 12	UINT
Gateway Configuration	36	136	23	Additional Data 13	UINT
Gateway Configuration	36	136	24	Additional Data 14	UINT
Gateway Configuration	36	136	25	Additional Data 15	UINT
Gateway Configuration	36	136	26	Additional Data 16	UINT
Gateway Configuration	36	136	27	Additional Data 17	UINT
Gateway Configuration	36	136	28	Additional Data 18	UINT
Gateway Configuration	36	136	29	Additional Data 19	UINT
Gateway Configuration	36	136	30	Additional Data 20	UINT
Gateway Configuration	36	136	31	Additional Data 21	UINT
Gateway Configuration	36	136	32	Additional Data 22	UINT
Gateway Configuration	36	136	33	Additional Data 23	UINT
Gateway Configuration	36	136	34	Additional Data 24	UINT
Gateway Configuration	36	136	35	Additional Data 25	UINT
Gateway Configuration	36	136	36	Additional Data 26	UINT
Gateway Configuration	36	136	37	Additional Data 27	UINT
Gateway Configuration	36	136	38	Additional Data 28	UINT
Gateway Configuration	36	136	39	Additional Data 29	UINT
Gateway Configuration	36	136	40	Additional Data 30	UINT
Gateway Configuration	36	136	41	Additional Data 31	UINT
Gateway Configuration	36	136	42	Valid value	UINT
Gateway Configuration	36	136	43	Data To Pluto Cycletime	UINT
Gateway Configuration	36	136	44	Valid value	UINT
Gateway Configuration	36	136	45	Enable Pluto Status	UINT
Gateway Configuration	36	136	46	Valid value	UINT
Gateway Configuration	36	136	47	Enable Local data req/resp	UINT
Gateway Configuration	36	136	48	Valid value	UINT
Gateway Configuration	36	136	49	Enable Serial pass req/resp	UINT
Gateway Configuration	36	136	50	Gateway Node Address	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Status information (host CPU to ExLink)	62	162	1	Length	UINT
Status information (host CPU to ExLink)	62	162	2	Host rev	UINT
Status information (host CPU to ExLink)	62	162	3	Year	UINT
Status information (host CPU to ExLink)	62	162	4	Month	UINT
Status information (host CPU to ExLink)	62	162	5	Day	UINT
Status information (host CPU to ExLink)	62	162	6	Serial no	UDINT (MSW)
Status information (host CPU to ExLink)	62	162	7	Serial no	UDINT (LSW)
Status information (host CPU to ExLink)	62	162	8	Pluto Node	UINT
Status information (host CPU to ExLink)	62	162	9	Pluto Bus	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Status information (host CPU to ExLink)	62	162	10	Free	UINT
Status information (host CPU to ExLink)	62	162	11	Free	UINT
Status information (host CPU to ExLink)	62	162	12	Free	UINT
Status information (host CPU to ExLink)	62	162	13	Free	UINT
Status information (host CPU to ExLink)	62	162	14	Free	UINT
Status information (host CPU to ExLink)	62	162	15	Free	UINT
Status information (host CPU to ExLink)	62	162	16	Free	UINT

Modbus Slave Name	Modbus Slave	PCCC Data Table Address	Modbus and PCCC Address	Data Name	Data Type
Raw TCP Server out	64	164	1	Length	UINT [>0]
Raw TCP Server out	64	164	2	Data	UINT
Raw TCP Server out	64	164	3	Data	UINT
Raw TCP Server out	64	164	…	free	UINT

4 Appendix D, Modbus TCP Information

The Modbus TCP server is running on the standard port number 502.
The server will respond on the following "**slave address**" or "**Unit Identifier number**".

Slave address or Unit Identifier	Data	Access Rule
1 (0x01)	Data to Pluto	read/write
2 (0x02)	Local Data Request	read/write
3 (0x03)	Serial Pass through Request	read/write
4 (0x04)	Gateway Configuration	write
33 (0x21)	Data from Pluto	read
34 (0x22)	Local Data Response	read/write
35 (0x23)	Serial Pass through Response	read/write

The access rules are,

Access Rule	Modbus TCP function
read	03 (0x03) Read Holding Register
write	16 (0x10) Preset Multiple Registers

4.1 Data from Pluto

Modbus TCP slave address 33, (0x21), for read node status and combined data from each Pluto.

Note: Only 100 words can be read in one request!

If more data is needed divide them in two or more request with selected start/end address in the request. For example a request with start 1 and end 66 will give node status and Pluto global data. A request with start 67 and end 130 will give additional data.

Address	Data Name	Data Type	Addr + 0	Addr + 1
1	Node Status	UDINT	MSW	LSW
3	PLUTO 00	UDINT	MSW	LSW
5	PLUTO 01	UDINT	MSW	LSW
7	PLUTO 02	UDINT	MSW	LSW
9	PLUTO 03	UDINT	MSW	LSW
11	PLUTO 04	UDINT	MSW	LSW
13	PLUTO 05	UDINT	MSW	LSW
15	PLUTO 06	UDINT	MSW	LSW
17	PLUTO 07	UDINT	MSW	LSW
19	PLUTO 08	UDINT	MSW	LSW
21	PLUTO 09	UDINT	MSW	LSW
23	PLUTO 10	UDINT	MSW	LSW
25	PLUTO 11	UDINT	MSW	LSW
27	PLUTO 12	UDINT	MSW	LSW
29	PLUTO 13	UDINT	MSW	LSW
31	PLUTO 14	UDINT	MSW	LSW
33	PLUTO 15	UDINT	MSW	LSW
35	PLUTO 16	UDINT	MSW	LSW
37	PLUTO 17	UDINT	MSW	LSW
39	PLUTO 18	UDINT	MSW	LSW
41	PLUTO 19	UDINT	MSW	LSW
43	PLUTO 20	UDINT	MSW	LSW
45	PLUTO 21	UDINT	MSW	LSW

47	PLUTO 22	UDINT	MSW	LSW
49	PLUTO 23	UDINT	MSW	LSW
51	PLUTO 24	UDINT	MSW	LSW
53	PLUTO 25	UDINT	MSW	LSW
55	PLUTO 26	UDINT	MSW	LSW
57	PLUTO 27	UDINT	MSW	LSW
59	PLUTO 28	UDINT	MSW	LSW
61	PLUTO 29	UDINT	MSW	LSW
63	PLUTO 30	UDINT	MSW	LSW
65	PLUTO 31	UDINT	MSW	LSW
67	Additional Data 00	UDINT	MSW	LSW
69	Additional Data 01	UDINT	MSW	LSW
71	Additional Data 02	UDINT	MSW	LSW
73	Additional Data 03	UDINT	MSW	LSW
75	Additional Data 04	UDINT	MSW	LSW
77	Additional Data 05	UDINT	MSW	LSW
79	Additional Data 06	UDINT	MSW	LSW
81	Additional Data 07	UDINT	MSW	LSW
83	Additional Data 08	UDINT	MSW	LSW
85	Additional Data 09	UDINT	MSW	LSW
87	Additional Data 10	UDINT	MSW	LSW
89	Additional Data 11	UDINT	MSW	LSW
91	Additional Data 12	UDINT	MSW	LSW
93	Additional Data 13	UDINT	MSW	LSW
95	Additional Data 14	UDINT	MSW	LSW
97	Additional Data 15	UDINT	MSW	LSW
99	Additional Data 16	UDINT	MSW	LSW
101	Additional Data 17	UDINT	MSW	LSW
103	Additional Data 18	UDINT	MSW	LSW
105	Additional Data 19	UDINT	MSW	LSW
107	Additional Data 20	UDINT	MSW	LSW
109	Additional Data 21	UDINT	MSW	LSW
111	Additional Data 22	UDINT	MSW	LSW
113	Additional Data 23	UDINT	MSW	LSW
115	Additional Data 24	UDINT	MSW	LSW
117	Additional Data 25	UDINT	MSW	LSW
119	Additional Data 26	UDINT	MSW	LSW
121	Additional Data 27	UDINT	MSW	LSW
123	Additional Data 28	UDINT	MSW	LSW
125	Additional Data 29	UDINT	MSW	LSW
127	Additional Data 30	UDINT	MSW	LSW
129	Additional Data 31	UDINT	MSW	LSW

Data in UDINT word as follows and detailed description in chapter 4,

UDINT																															
MSW								**LSW**																							
Byte 4								Byte 3								Byte 1								Byte 0							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

4.2 Data to Pluto

Modbus TCP slave address 1, (0x01), for read/write data to Pluto system.

Note: This data is common for all connected clients. E.g. valid data to Pluto will be the data written by the last client writing data to this slave address.

Address	Data Name	Data Type
0	Length [13, 0x000D]	UINT
1	Area valid bit information (0=Invalid, 1=Valid) - bit 0, valid data for area 0 - bit 1, valid data for area 1 - bit 2, valid data for area 2 - bit 3, valid data for area 3	UINT
2	Area 0, Bits	UINT
3	Area 0, Register 0	UINT
4	Area 0, Register 1	UINT
5	Area 1, Bits	UINT
6	Area 1, Register 0	UINT
7	Area 1, Register 1	UINT
8	Area 2, Bits	UINT
9	Area 2, Register 0	UINT
10	Area 2, Register 1	UINT
11	Area 3, Bits	UINT
12	Area 3, Register 0	UINT
13	Area 3, Register 1	UINT

4.3 Gateway Configuration

Modbus TCP slave address 4, (0x04), to write new configuration to the gateway.

Note: This data is common for all connected clients. E.g. valid configuration will be the data written by the last client writing data to this slave address.

Note: For additional data Pluto number and IO-type shall be set to zero if data area is not used.

Address	Data Name	Data Type	Addr + 0	Addr + 1
0	Length [36, 0x0024]	UNIT		
1	Enable Data to Pluto (0 = Disabled; 1 = Enabled) - bit 0 – Data To Pluto 1 - bit 1 – Data To Pluto 2 - bit 2 – Data To Pluto 3 - bit 3 – Data To Pluto 4	UINT	-	-
2	Data to Pluto Timeout (ms)	UINT	-	-
3	Expected Nodes Bitmap	UDINT	MSW	LSW
			High byte	**Low byte**
5	Additional Data Area 0	UINT	Pluto no.	IO-type
6	Additional Data Area 1	UINT	Pluto no.	IO-type
7	Additional Data Area 2	UINT	Pluto no.	IO-type
8	Additional Data Area 3	UINT	Pluto no.	IO-type
9	Additional Data Area 4	UINT	Pluto no.	IO-type
10	Additional Data Area 5	UINT	Pluto no.	IO-type
11	Additional Data Area 6	UINT	Pluto no.	IO-type
12	Additional Data Area 7	UINT	Pluto no.	IO-type
13	Additional Data Area 8	UINT	Pluto no.	IO-type
14	Additional Data Area 9	UINT	Pluto no.	IO-type
15	Additional Data Area 10	UINT	Pluto no.	IO-type
16	Additional Data Area 11	UINT	Pluto no.	IO-type
17	Additional Data Area 12	UINT	Pluto no.	IO-type
18	Additional Data Area 13	UINT	Pluto no.	IO-type
19	Additional Data Area 14	UINT	Pluto no.	IO-type
20	Additional Data Area 15	UINT	Pluto no.	IO-type
21	Additional Data Area 16	UINT	Pluto no.	IO-type
22	Additional Data Area 17	UINT	Pluto no.	IO-type
23	Additional Data Area 18	UINT	Pluto no.	IO-type
24	Additional Data Area 19	UINT	Pluto no.	IO-type
25	Additional Data Area 20	UINT	Pluto no.	IO-type
26	Additional Data Area 21	UINT	Pluto no.	IO-type
27	Additional Data Area 22	UINT	Pluto no.	IO-type
28	Additional Data Area 23	UINT	Pluto no.	IO-type
29	Additional Data Area 24	UINT	Pluto no.	IO-type
30	Additional Data Area 25	UINT	Pluto no.	IO-type
31	Additional Data Area 26	UINT	Pluto no.	IO-type
32	Additional Data Area 27	UINT	Pluto no.	IO-type
33	Additional Data Area 28	UINT	Pluto no.	IO-type
34	Additional Data Area 29	UINT	Pluto no.	IO-type
35	Additional Data Area 30	UINT	Pluto no.	IO-type
36	Additional Data Area 31	UINT	Pluto no.	IO-type
37	Data to Pluto Cycle time (ms)	UINT		
38	(PROFINET setting)	UINT		
39	(PROFINET setting)	UINT		
40	(PROFINET setting)	UINT		
41	Gateway Node Address (0-16)	UINT		

4.4 Local Data Request/Response

Note: Only one connected client can make local data request/response at a time. If more clients want to do local data request/response these clients need to share this resource between them.

Local Data Request

Modbus TCP slave address 2, (0x02), to read/write new request for local data.
For more information regarding Pluto address range see chapter 5.4.3.4.

Address	Data Name	Data Type
0	Flag 0 = request read by gateway. 3 = request set at write!	UINT
1	Request Pluto station id 0-31 Request Gateway 255 (0xFF)	UINT
2	Data Type 0 = global data 1 = Local memory 2 = Local register 3 = Local Parameter	UINT
3	Address	UINT

Local Data Response

Modbus TCP slave address 34, (0x22), to read response of written request.

Address	Data Name	Data Type	Addr + 0	Addr + 1
0	Flag 6 = New data valid. 0 = set at write!	UINT	-	-
1	Pluto station id [0-31]	UINT	-	-
2	Data Type 0 = global data 1 = Local memory 2 = Local register 3 = Local Parameter	UINT	-	-
3	Requested address	UINT	-	-
4	Error Code 0x0001 = Response OK 0x0002 = Request timeout 0x0004 = Request bad data 0x0008 = Request unknown	UINT	-	-
5	Response data	UDINT	MSW	LSW

Sequence of use

The following sequence of commands shall be used when retrieveing local data,

- Read slave address 2 and at least the first word.
  Check that this flag is zero.

- Write the request to slave address 2.
  The flag shall be set to 3 in the request data.

- Read slave address 34.
  If flag data set to 6 then new data is valid. Check rest of data especially the error code. If all data correct then use the response data.

- Write clear flag to slave address 34.
  By writing zero value to first word the flag information is cleared.

4.5 Serial Pass through Request/Response

Note: This function is **not** implemented.

Serial Pass through Request

Modbus TCP slave address 3, (0x03), to read/write new request of serial pass through data.

Address	Data Name	Data Type
0	Length information [2-4]	UINT
1	Pluto station id [0-31]	UINT
2	Data	UINT
3	Data	UINT
4	Data	UINT

Serial Pass through Response

Modbus TCP slave address 35, (0x23), to read response of written request.

Address	Data Name	Data Type
0	Length information [3-6]	UINT
1	Pluto station id [0-31]	UINT
2	Error Code 0x0001 = Response OK 0x0002 = Request timeout 0x0004 = Request bad data 0x0008 = Request unknown	UINT
3	Data	UINT
4	Data	UINT
5	Data	UINT

Data format

Data string “123456” will in both request and response be sent in following format,

Data Position	Data
1	0x3132
2	0x3334
3	0x3536

If a shorter string shall be sent the not used positions will be padded with zero. Example data string “123” will in both request and response be sent in following format,

Data Position	Data
1	0x3132
2	0x3300
3	0x0000

The length information shall also be set according to number of valid words in the message.

5 Appendix E, PROFINET Information

Pluto Gateway	
Vendor	Jokab Safety AB (GATE-E1) ABB AB (GATE-E2)
Vendor ID	0x0184
Product family	Pluto Gateway
Device ID	0x03E8
Details	Pluto Gateway PROFINET

5.1 Device Access Points

Module: Jokab Safety GATE-E1 or GATE-E2	
Name	Jokab Safety GATE-E1 or GATE-E2
Module Identity Number	0x00000100
Details	Pluto Gateway PROFINET
Order Number	20-070-73
Software Version	1.0
Hardware Version	1.0
Maximal Input Length	1440 Bytes
Maximal Output Length	1440 Bytes
Useable Slots	0..15
Minimal Device Interval	8 ms
Based on	RTA ConnectMe
DNS Compliant Name	JOKABGATEE1 or GATE-E2
Fixed in Slots	0

Gateway Data to Pluto timeout (Index: 1 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Timeout [ms]	Unsigned16	0	0	-	0	0..60000

Pluto Data to Pluto cycle time (Index: 2 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Cycle time [ms]	Unsigned8	0	0	-	100	0..250

Gateway Node Address (Index: 42 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Gateway Node Address	Unsigned8	0	0	-	0	0..16

Useable Modules

Name	Information's	Useable Slots	Fixed in Slots
Node Status	Show which Pluto units are active on Pluto bus.	1..1	
Pluto Nodes 00-07	Global variables from Pluto 0-7.	2..2	
Pluto Nodes 08-15	Global variables from Pluto 8-15.	3..3	
Pluto Nodes 16-23	Global variables from Pluto 16-23.	4..4	
Pluto Nodes 24-31	Global variables from Pluto 24-31.	5..5	

Additional Data Area 00-07	Additional data from Pluto.	6..6
Additional Data Area 08-15	Additional data from Pluto.	7..7
Additional Data Area 16-23	Additional data from Pluto.	8..8
Additional Data Area 24-31	Additional data from Pluto.	9..9
Data to Pluto Area 0	Data to Pluto.	10..10
Data to Pluto Area 1	Data to Pluto.	11..11
Data to Pluto Area 2	Data to Pluto.	12..12
Data to Pluto Area 3	Data to Pluto.	13..13
Local Data Request	Request to Pluto for variable data.	14..14
Local Data Response	Response from Pluto for variable data.	15..15

5.2 Modules

Module: Node Status

Name	Node Status
Module Identity Number	0x00000101
Details	Show which Pluto units are active on Pluto bus.
Order Number	N/A
Category	01-Status
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Node Status	Unsigned32	Yes	

Status (Index: 3 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Defaultvalue	Value Range
Module usage	Bit	0	0	-	Enable	0..1

Module: Pluto Nodes 00-07

Name	Pluto Nodes 00-07
Module Identity Number	0x00000201
Details	Global variabels from Pluto 0-7.
Order Number	N/A
Category	02-Pluto Nodes
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Node 00 Data	Unsigned32	Yes	
Node 01 Data	Unsigned32	Yes	
Node 02 Data	Unsigned32	Yes	
Node 03 Data	Unsigned32	Yes	
Node 04 Data	Unsigned32	Yes	

Node 05 Data	Unsigned32	Yes
Node 06 Data	Unsigned32	Yes
Node 07 Data	Unsigned32	Yes

Pluto Nodes 0-7 (Index: 4 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Defaultvalue	Value Range
Module usage	Bit	0	0	-	Enable	0..1

Module: Pluto Nodes 08-15

Name	Pluto Nodes 08-15
Module Identity Number	0x00000202
Details	Global variabels from Pluto 8-15.
Order Number	N/A
Category	02-Pluto Nodes
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Node 08 Data	Unsigned32	Yes	
Node 09 Data	Unsigned32	Yes	
Node 10 Data	Unsigned32	Yes	
Node 11 Data	Unsigned32	Yes	
Node 12 Data	Unsigned32	Yes	
Node 13 Data	Unsigned32	Yes	
Node 14 Data	Unsigned32	Yes	
Node 15 Data	Unsigned32	Yes	

Pluto Nodes 8-15 (Index: 5 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	0	-	Enable	0..1

Module Pluto Nodes 16-23

Name	Pluto Nodes 16-23
Module Identity Number	0x00000203
Details	Global variabels from Pluto 16-23.
Order Number	N/A
Category	02-Pluto Nodes
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Node 16 Data	Unsigned32	Yes	
Node 17 Data	Unsigned32	Yes	
Node 18 Data	Unsigned32	Yes	
Node 19 Data	Unsigned32	Yes	

Node 20 Data	Unsigned32	Yes
Node 21 Data	Unsigned32	Yes
Node 22 Data	Unsigned32	Yes
Node 23 Data	Unsigned32	Yes

Pluto Nodes 16-23 (Index: 6 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	0	-	Enable	0..1

Module Pluto Nodes 24-31

Name	Pluto Nodes 24-31
Module Identity Number	0x00000204
Details	Global variabels from Pluto 24-31.
Order Number	N/A
Category	02-Pluto Nodes
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Node 24 Data	Unsigned32	Yes	
Node 25 Data	Unsigned32	Yes	
Node 26 Data	Unsigned32	Yes	
Node 27 Data	Unsigned32	Yes	
Node 28 Data	Unsigned32	Yes	
Node 29 Data	Unsigned32	Yes	
Node 30 Data	Unsigned32	Yes	
Node 31 Data	Unsigned32	Yes	

Pluto Nodes 24-31 (Index: 7 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	0	-	Enable	0..1

Module: Additional Data Area 00-07

Name	Additional Data Area 00-07
Module Identity Number	0x00000301
Details	Additional data from Pluto.
Order Number	N/A
Category	05-Additional Data
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Additional Data 00	Unsigned32	Yes	
Additional Data 01	Unsigned32	Yes	
Additional Data 02	Unsigned32	Yes	

Additional Data 03	Unsigned32	Yes
Additional Data 04	Unsigned32	Yes
Additional Data 05	Unsigned32	Yes
Additional Data 06	Unsigned32	Yes
Additional Data 07	Unsigned32	Yes

Additional Data 00 (Index: 8 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 01 (Index: 9 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 02 (Index: 10 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 03 (Index: 11 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 04 (Index: 12 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 05 (Index: 13 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 06 (Index: 14 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 07 (Index: 15 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Module: Additional Data Area 08-15

Name	Additional Data Area 08-15
Module Identity Number	0x00000302
Details	Additional data from Pluto.
Order Number	N/A
Category	05-Additional Data
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data			
Name	**Data Type**	**Display as Bits**	**Length [Bytes]**
Additional Data 08	Unsigned32	Yes	
Additional Data 09	Unsigned32	Yes	
Additional Data 10	Unsigned32	Yes	
Additional Data 11	Unsigned32	Yes	
Additional Data 12	Unsigned32	Yes	
Additional Data 13	Unsigned32	Yes	
Additional Data 14	Unsigned32	Yes	
Additional Data 15	Unsigned32	Yes	

Additional Data 08 (Index: 16 -- Length: 2 Byte)						
Name of Parameter	**Data Type**	**Byte Offset**	**Bit Offset**	**Bit Length**	**Default value**	**Value Range**
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 09 (Index: 17 -- Length: 2 Byte)						
Name of Parameter	**Data Type**	**Byte Offset**	**Bit Offset**	**Bit Length**	**Default value**	**Value Range**
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 10 (Index: 18 -- Length: 2 Byte)						
Name of Parameter	**Data Type**	**Byte Offset**	**Bit Offset**	**Bit Length**	**Default value**	**Value Range**
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 11 (Index: 19 -- Length: 2 Byte)						
Name of Parameter	**Data Type**	**Byte Offset**	**Bit Offset**	**Bit Length**	**Default value**	**Value Range**
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 12 (Index: 20 -- Length: 2 Byte)						
Name of Parameter	**Data Type**	**Byte Offset**	**Bit Offset**	**Bit Length**	**Default value**	**Value Range**
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 13 (Index: 21 -- Length: 2 Byte)						
Name of Parameter	**Data Type**	**Byte Offset**	**Bit Offset**	**Bit Length**	**Default value**	**Value Range**
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 14 (Index: 22 -- Length: 2 Byte)						
Name of Parameter	**Data Type**	**Byte Offset**	**Bit Offset**	**Bit Length**	**Default value**	**Value Range**
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 15 (Index: 23 -- Length: 2 Byte)						
Name of Parameter	**Data Type**	**Byte Offset**	**Bit Offset**	**Bit Length**	**Default value**	**Value Range**
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Module: Additional Data Area 16-23	
Name	Additional Data Area 16-23
Module Identity Number	0x00000303

Details	Additional data from Pluto.
Order Number	N/A
Category	05-Additional Data
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Additional Data 16	Unsigned32	Yes	
Additional Data 17	Unsigned32	Yes	
Additional Data 18	Unsigned32	Yes	
Additional Data 19	Unsigned32	Yes	
Additional Data 20	Unsigned32	Yes	
Additional Data 21	Unsigned32	Yes	
Additional Data 22	Unsigned32	Yes	
Additional Data 23	Unsigned32	Yes	

Additional Data 16 (Index: 24 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 17 (Index: 25 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 18 (Index: 26 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 19 (Index: 27 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 20 (Index: 28 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 21 (Index: 29 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 22 (Index: 30 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 23 (Index: 31 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range

From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Module: Additional Data Area 24-31

Name	Additional Data Area 24-31
Module Identity Number	0x00000304
Details	Additional data from Pluto.
Order Number	N/A
Category	05-Additional Data
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Additional Data 24	Unsigned32	Yes	
Additional Data 25	Unsigned32	Yes	
Additional Data 26	Unsigned32	Yes	
Additional Data 27	Unsigned32	Yes	
Additional Data 28	Unsigned32	Yes	
Additional Data 29	Unsigned32	Yes	
Additional Data 30	Unsigned32	Yes	
Additional Data 31	Unsigned32	Yes	

Additional Data 24 (Index: 32 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 25 (Index: 33 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 26 (Index: 34 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 27 (Index: 35 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 28 (Index: 36 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 29 (Index: 37 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 30 (Index: 38 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Additional Data 31 (Index: 39 -- Length: 2 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
From Pluto Node	Unsigned8	0	0	-	Pluto 00	0..31
IO type	Unsigned8	1	0	-	UNUSED	0..110

Module: Data to Pluto Area 0

Name	Data to Pluto Area 0
Module Identity Number	0x00000401
Details	Data to Pluto.
Order Number	N/A
Category	03-Data to Pluto
Software Version	1.0
Hardware Version	1.0

Cyclic Output Data

Name	Data type	Display as Bits	Length [Bytes]
Area 0 Bits	Unsigned16	Yes	
Area 0 Register 0	Unsigned16	Yes	
Area 0 Register 1	Unsigned16	Yes	

Enable Area 0 (Index: 40 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	0	-	Enable	0..1

Module: Data to Pluto Area 1

Name	Data to Pluto Area 1
Module Identity Number	0x00000402
Details	Data to Pluto.
Order Number	N/A
Category	03-Data to Pluto
Software Version	1.0
Hardware Version	1.0

Cyclic Output Data

Name	Data Type	Display as Bits	Length [Bytes]
Area 1 Bits	Unsigned16	Yes	
Area 1 Register 0	Unsigned16	Yes	
Area 1 Register 1	Unsigned16	Yes	

Enable Area 1 (Index: 40 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	1	-	Enable	0..1

Module: Data to Pluto Area 2

Name	Data to Pluto Area 2
Module Identity Number	0x00000403
Details	Data to Pluto.

Order Number	N/A
Category	03-Data to Pluto
Software Version	1.0
Hardware Version	1.0

Cyclic Output Data

Name	Data Type	Display as Bits	Length [Bytes]
Area 2 Bits	Unsigned16	Yes	
Area 2 Register 0	Unsigned16	Yes	
Area 2 Register 1	Unsigned16	Yes	

Enable Area 2 (Index: 40 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	2	-	Enable	0..1

Module: Data to Pluto Area 3

Name	Data to Pluto Area 3
Module Identity Number	0x00000404
Details	Data to Pluto.
Order Number	N/A
Category	03-Data to Pluto
Software Version	1.0
Hardware Version	1.0

Cyclic Output Data

Name	Data Type	Display as Bits	Length [Bytes]
Area 3 Bits	Unsigned16	Yes	
Area 3 Register 0	Unsigned16	Yes	
Area 3 Register 1	Unsigned16	Yes	

Enable Area 3 (Index: 40 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	3	-	Enable	0..1

Module: Local Data Request

Name	Local Data Request
Module Identity Number	0x00000501
Details	Request to Pluto for variable data.
Order Number	N/A
Category	04-Local Data Req/Rsp
Software Version	1.0
Hardware Version	1.0

Cyclic Output Data

Name	Data Type	Display as Bits	Length [Bytes]
Sequence Number	Unsigned16	Yes	
Pluto Node	Unsigned16	Yes	
Data Type	Unsigned16	Yes	
Address	Unsigned16	Yes	

Local Data Request Enable (Index: 41 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	0	-	Enable	0..1

Module: Local Data Response	
Name	Local Data Response
Module Identity Number	0x00000502
Details	Response from Pluto for variable data.
Order Number	N/A
Category	04-Local Data Req/Rsp
Software Version	1.0
Hardware Version	1.0

Cyclic Input Data

Name	Data Type	Display as Bits	Length [Bytes]
Sequence Number	Unsigned16	Yes	
Pluto Node	Unsigned16	Yes	
Data Type	Unsigned16	Yes	
Address	Unsigned16	Yes	
Error Code	Unsigned16	Yes	
Data MSW	Unsigned16	Yes	
Data LSW	Unsigned16	Yes	

Local Data Response Enable (Index: 41 -- Length: 1 Byte)

Name of Parameter	Data Type	Byte Offset	Bit Offset	Bit Length	Default value	Value Range
Module usage	Bit	0	1	-	Enable	0..1

5.3 Parameter of Modules

Parameter: Module use

Value	Content
0	Disable
1	Enable

Parameter: Gateway Node Address

Value	Content
0	DIP-Switch Setting
1	Node Address 0
2	Node Address 1
3	Node Address 2
4	Node Address 3
5	Node Address 4
6	Node Address 5
7	Node Address 6
8	Node Address 7
9	Node Address 8
10	Node Address 9
11	Node Address 10
12	Node Address 11

13	Node Address 12
14	Node Address 13
15	Node Address 14
16	Node Address 15

Parameter: Gateway Node Address

Value	Content
0	DIP-Switch Setting
1	Node Address 0
2	Node Address 1
3	Node Address 2
4	Node Address 3
5	Node Address 4
6	Node Address 5
7	Node Address 6
8	Node Address 7
9	Node Address 8
10	Node Address 9
11	Node Address 10
12	Node Address 11
13	Node Address 12
14	Node Address 13
15	Node Address 14
16	Node Address 15

Parameter: From Pluto Node

Value	Content
0	Pluto 00
1	Pluto 01
2	Pluto 02
3	Pluto 03
4	Pluto 04
5	Pluto 05
6	Pluto 06
7	Pluto 07
8	Pluto 08
9	Pluto 09
10	Pluto 10
11	Pluto 11
12	Pluto 12
13	Pluto 13
14	Pluto 14
15	Pluto 15
16	Pluto 16
17	Pluto 17
18	Pluto 18

19	Pluto 19
20	Pluto 20
21	Pluto 21
22	Pluto 22
23	Pluto 23
24	Pluto 24
25	Pluto 25
26	Pluto 26
27	Pluto 27
28	Pluto 28
29	Pluto 29
30	Pluto 30
31	Pluto 31

Parameter: IO Type

Value	Content
0	UNUSED
1	ToGateway_UserNumber_1
2	ToGateway_UserNumber_2
3	ToGateway_UserNumber_3
4	ToGateway_UserNumber_4
5	ToGateway_UserNumber_5
6	ToGateway_UserNumber_6
7	ToGateway_UserNumber_7
8	ToGateway_UserNumber_8
9	ToGateway_UserNumber_9
10	ToGateway_UserNumber_10
11	ToGateway_UserNumber_11
12	ToGateway_UserNumber_12
13	ToGateway_UserNumber_13
14	ToGateway_UserNumber_14
15	ToGateway_UserNumber_15
16	ToGateway_UserNumber_16
17	ToGateway_UserNumber_17
18	ToGateway_UserNumber_18
19	ToGateway_UserNumber_19
20	ToGateway_UserNumber_20
21	ToGateway_UserNumber_21
22	ToGateway_UserNumber_22
23	ToGateway_UserNumber_23
24	ToGateway_UserNumber_24
25	ToGateway_UserNumber_25
26	ToGateway_UserNumber_26
27	ToGateway_UserNumber_27
28	ToGateway_UserNumber_28
29	ToGateway_UserNumber_29

30	ToGateway_UserNumber_30
31	ToGateway_UserNumber_31
32	ToGateway_UserNumber_32
33	ToGateway_UserNumber_33
34	ToGateway_UserNumber_34
35	ToGateway_UserNumber_35
36	ToGateway_UserNumber_36
37	ToGateway_UserNumber_37
38	ToGateway_UserNumber_38
39	ToGateway_UserNumber_39
40	ToGateway_UserNumber_40
41	ToGateway_UserNumber_41
42	ToGateway_UserNumber_42
43	ToGateway_UserNumber_43
44	ToGateway_UserNumber_44
45	ToGateway_UserNumber_45
46	ToGateway_UserNumber_46
47	ToGateway_UserNumber_47
48	ToGateway_UserNumber_48
49	ToGateway_UserNumber_49
50	ToGateway_UserNumber_50
51	ToGateway_UserNumber_51
52	ToGateway_UserNumber_52
53	ToGateway_UserNumber_53
54	ToGateway_UserNumber_54
55	ToGateway_UserNumber_55
56	ToGateway_UserNumber_56
57	ToGateway_UserNumber_57
58	ToGateway_UserNumber_58
59	ToGateway_UserNumber_59
60	ToGateway_UserNumber_60
61	ToGateway_UserNumber_61
62	ToGateway_UserNumber_62
63	ToGateway_UserNumber_63
64	ToGateway_UserNumber_64
65	ToGateway_UserNumber_65
66	ToGateway_UserNumber_66
67	ToGateway_UserNumber_67
68	ToGateway_UserNumber_68
69	ToGateway_UserNumber_69
70	ToGateway_UserNumber_70
71	ToGateway_UserNumber_71
72	ToGateway_UserNumber_72
73	ToGateway_UserNumber_73
74	ToGateway_UserNumber_74
75	ToGateway_UserNumber_75

76	ToGateway_UserNumber_76
77	ToGateway_UserNumber_77
78	ToGateway_UserNumber_78
79	ToGateway_UserNumber_79
80	ToGateway_UserNumber_80
81	ToGateway_UserNumber_81
82	ToGateway_UserNumber_82
83	ToGateway_UserNumber_83
84	ToGateway_UserNumber_84
85	ToGateway_UserNumber_85
86	ToGateway_UserNumber_86
87	ToGateway_UserNumber_87
88	ToGateway_UserNumber_88
89	ToGateway_UserNumber_89
90	ToGateway_UserNumber_90
91	ToGateway_UserNumber_91
92	ToGateway_UserNumber_92
93	ToGateway_UserNumber_93
94	ToGateway_UserNumber_94
95	ToGateway_UserNumber_95
96	ToGateway_UserNumber_96
97	ToGateway_UserNumber_97
98	ToGateway_UserNumber_98
99	ToGateway_UserNumber_99
100	ToGateway_ErrorCode
101	ToGateway_B46_I20_I47
102	ToGateway_ASi_16_31_Safe
103	ToGateway_ASi_1_3_NonSafe_In
104	ToGateway_ASi_4_7_NonSafe_In
105	ToGateway_ASi_8_11_NonSafe_In
106	ToGateway_ASi_12_15_NonSafe_In
107	ToGateway_ASi_16_19_NonSafe_In
108	ToGateway_ASi_20_23_NonSafe_In
109	ToGateway_ASi_24_27_NonSafe_In
110	ToGateway_ASi_28_31_NonSafe_In

Note: This page shows the content of a GSD file transformed into HTML format. In the case of disparity between this and the XML view, the content of the XML file takes precedence.