



# DeviceNet

Manual / Users Guide

Part No 643197-01, Doc. No 643197 Ver. 00

LEINE  LINDE

## Content

<b>LIST OF TABLES</b> .....	<b>3</b>
<b>LIST OF FIGURES</b> .....	<b>3</b>
<b>1 GENERAL INFORMATION</b> .....	<b>4</b>
<b>1 GENERAL INFORMATION</b> .....	<b>4</b>
1.1 ABSOLUTE ENCODERS .....	4
1.2 DEVICE <sup>NET</sup> TECHNOLOGY .....	4
1.3 ABOUT LEINE & LINDE AB .....	5
1.3.1 <i>Technical and commercial support</i> .....	5
1.3.2 <i>Certification of fieldbus products</i> .....	5
1.4 REFERENCES .....	5
1.5 ABBREVIATIONS .....	5
<b>2 ENCODER INSTALLATION</b> .....	<b>6</b>
2.1 SETTINGS INSIDE THE ENCODER .....	6
2.2 NODE ADDRESS/MAC ID .....	6
2.3 BUS TERMINATION .....	6
2.4 BAUDRATE SETTING .....	6
2.5 CONNECTING THE ENCODER .....	7
2.5.1 <i>Power supply</i> .....	7
2.5.2 <i>BUS lines</i> .....	7
2.5.3 <i>Shielding philosophy</i> .....	7
2.6 EDS FILE .....	8
2.7 LED INDICATION .....	8
2.7.1 <i>Module LED</i> .....	8
2.8.2 <i>Network LED</i> .....	8
<b>3 PROFILE OVERVIEW</b> .....	<b>9</b>
<b>4 ENCODER FUNCTIONALITY</b> .....	<b>10</b>
4.1 BASIC ENCODER FUNCTIONALITY .....	10
4.2 DEVICE <sup>NET</sup> FUNCTIONALITY SUPPORTED .....	10
4.3 DEVICE <sup>NET</sup> ADOPTION OF CAN IDENTIFIERS .....	11
4.4 ASSEMBLY INSTANCES .....	11
4.5 POSITION SENSOR OBJECT (0x23 <sub>HEX</sub> ) .....	12
4.5.1 <i>Position value</i> .....	12
4.5.2 <i>Position sensor type</i> .....	12
4.5.3 <i>Code sequence</i> .....	12
4.5.4 <i>Commissioning diagnostic control</i> .....	13
4.5.5 <i>Scaling function</i> .....	13
4.5.6 <i>Position format</i> .....	14
4.5.7 <i>Preset, zero-set and offset value</i> .....	14
4.5.8 <i>COS/delta</i> .....	15
4.5.9 <i>Position limits</i> .....	15
4.5.10 <i>Velocity</i> .....	15
4.5.11 <i>Operating status</i> .....	15
4.5.12 <i>Measuring range</i> .....	15
4.5.13 <i>Alarms</i> .....	16
4.5.14 <i>Warnings</i> .....	16
4.5.15 <i>Operating time</i> .....	16
<b>5 ENCODER COMMISSIONING EXAMPLE</b> .....	<b>17</b>
<b>APPENDIX A, HISTORY</b> .....	<b>19</b>

## List of tables

<i>Table 1 Pinning bus in/out- lines M12 version</i> .....	7
<i>Table 2 Module LED indication</i> .....	8
<i>Table 3 Status LED indication</i> .....	8
<i>Table 4 CAN identifier</i> .....	11
<i>Table 5 Assembly instances</i> .....	11
<i>Table 6 Position sensor object</i> .....	12
<i>Table 7 Singleturn scaling attribute format</i> .....	13
<i>Table 8 Total measuring range attribute format</i> .....	13
<i>Table 9 Position state register</i> .....	15
<i>Table 10 Operating status</i> .....	15
<i>Table 11 Alarms</i> .....	16
<i>Table 12 Supported alarms</i> .....	16
<i>Table 13 Warnings</i> .....	16
<i>Table 14 Supported warnings</i> .....	16

## List of figures

Figure 1 PCB-view of a cable gland DeviceNet encoder.....	6
Figure 2 Orientation of M12 bus connectors .....	7
Figure 3 Cable assembling principal.....	7
Figure 4 Basic encoder functionality .....	10
Figure 5 Cyclic scaling .....	14
Figure 6 Non-cyclic scaling .....	14

Leine & Linde AB claims copyright on this documentation. It is not allowed to modify, extend or to hand over to a third party and/or copy this documentation without written approval from Leine & Linde AB.

Specifications and content in this document are subject to change without prior notice due to our continuous strives to improve functionality and performance of our products.

## 1 General information

### 1.1 Absolute Encoders

With an absolute encoder each angular position is assigned a coded position value generated by a code disc equipped with several parallel fine graduations tracks which are scanned individually. On singleturn encoders, i.e. an encoder producing absolute positions within one revolution, the absolute position information repeats itself with every revolution. A multiturn encoder can also distinguish between revolutions. The number of unique revolutions is determined by the resolution of the multiturn scanning and repeats itself after the total resolution is reached.

### 1.2 DeviceNet technology

DeviceNet is a low-level network that provides connections between simple industrial devices (sensors, actuators) and higher-level devices (controllers). DeviceNet provides Master/Slave and Peer-to-Peer capabilities over the CAN bus.

**DeviceNet has two primary purposes:**

- Transport of control-oriented information associated with low-level devices
- Transport of other information, which is indirectly related to the system being controlled, such as configuration parameters.

**DeviceNet makes use of abstract object modelling to describe:**

- The suite of communication services available
- The externally visible behaviour of a DeviceNet node
- A common means by which information within DeviceNet products is accessed and exchanged

A DeviceNet node is modelled as a collection of objects. An object provides an abstract representation of a particular component within a product. The realization of this abstract object model within a product is implementation dependent. In other words, a product internally maps this object model in a fashion specific to its implementation.

Based on the encoder profile, see chapter 3, the following objects are implemented in the Leine & Linde integrated encoder:

- **Identity Object.** Provides identification and general information about the device. In addition the Identity object includes the reset function.
- **Message Router.** The Message Router Object provides a messaging connection point through which a client may address a service to any object class or instance residing in the physical device.
- **Connection Class Object.** Provides the configuration and status of a poll connection port.
- **DeviceNet Object.** Provides the configuration and status of a DeviceNet port.
- **Assembly Object.** Binds attributes of multiple objects, which allows data to or from each object to be sent or received over a single connection. By addressing different instances of the Assembly object, different collections of data are returned in each poll cycle.
- **Position Sensor Object.** Provides the configuration and status of the Encoder
- **Acknowledge Object.** Used to manage the reception of message acknowledgments. This object communicates with a message producing Application Object within a device. The Acknowledge Handler Object notifies the producing application of acknowledge reception, acknowledge timeouts, and production retry limit.

### 1.3 About Leine & Linde AB

For almost 40 years the Swedish based company Leine & Linde has concentrated on one thing – development and manufacturing of advanced encoders that meet the most rigorous demands. That is why a wide selection of incremental and absolute encoders with obvious concentration on robust products and quality down to the last detail can be offered. Leine & Linde encoders provide the utmost in reliability year after year, in working conditions where vibration, dirt, cold and other harsh environments are common.

Leine & Linde can meet very specific individual customer demands. The encoders are easily adapted, due to a modular design, to the customer's exact need with respect to resolution, electrical connections and interfaces, casings, etc. That is due to the fact that tomorrow's technology already is used today in Leine & Linde's product lines. Leine & Linde concentrate on advanced development of intelligent encoders with integrated ASICs, new special features and with adaptations to different fieldbus systems such as DeviceNet. This enables us to meet the need for increasingly effective and dependable machines and automation to an even higher degree.

#### 1.3.1 Technical and commercial support

Leine & Linde are represented by subsidiaries in many countries around the world. In addition, there are many services agencies and distributors located worldwide ready to reply to commercial enquires or technical support. For more contact information, please visit our web site or contact Leine & Linde in Strängnäs, Sweden.

#### **Leine & Linde AB**

Olivehällsvägen 8  
SE-645 21 Strängnäs  
Sweden

E-mail: [info@leinelinde.se](mailto:info@leinelinde.se)  
web: [www.leinelinde.com](http://www.leinelinde.com)

Tel: +46 152 265 00  
Fax: +46 152 265 05

#### 1.3.2 Certification of fieldbus products

In order to achieve interoperability between vendors and appropriate device functionality Leine & Linde AB continuously strives to have all fieldbus products certified by external bodies. For questions regarding certification or conformance tests of specific products please visit our web site or contact Leine & Linde in Strängnäs, Sweden.

### 1.4 References

<a href="http://www.odva.org">http://www.odva.org</a>	ODVA
ODVA DeviceNet Specification 2.0	ODVA
Statement of Conformance	Leine & Linde AB
Encoder Profile Revision 2	ODVA/Leine & Linde AB
Encoder EDS-file	Leine & Linde AB

### 1.5 Abbreviations

CAN	Controller Area Network
CiA	CAN in Automation
CAL	CAN Application Layer
EDS	Electronic Data Sheet
DCF	Device Configuration File
MAC	Media Access Control

## 2 Encoder Installation

### 2.1 Settings inside the encoder

The encoder node address and bus termination must be configured during commissioning of the device. This is done by removing the back cover and screw off the three screws at the rear of the encoder.

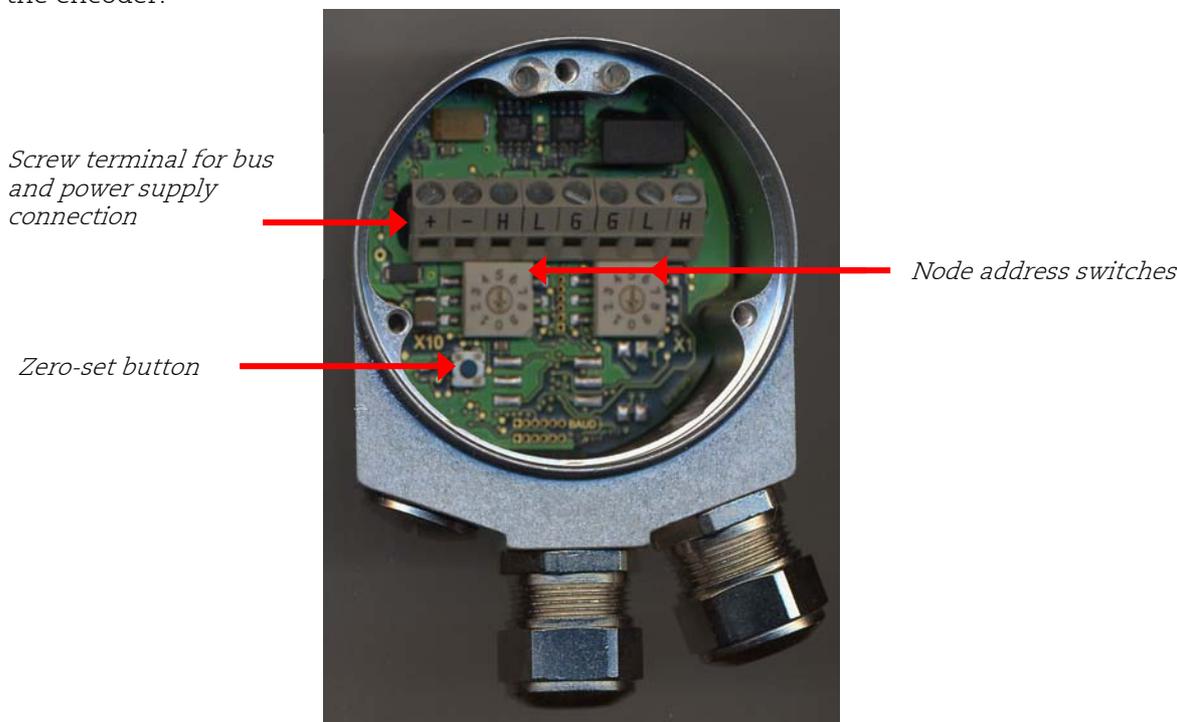


Figure 1 PCB-view of a cable gland DeviceNet encoder

### 2.2 Node address/MAC ID

The node address or MAC ID of the device can be set using two decimal rotary switches located inside the back cover. The weighting, x10 and x1 are specified beside the switches. Permissible address range is between 1 and 63 but the lower addresses 0 to 2 are usually used by the master and not recommended to be used by the device. The device address is read and adopted when the encoder power supply is switched on, i.e. a power off and On is required in order to adopt changes done to the address settings. The node address or MAC ID is adopted and used to generate the CAN message identifier, see chapter 4.3

### 2.3 Bus termination

In a DeviceNet network, all devices are connected in a bus structure. Up to 32 devices (master and/or slaves) can be connected in one segment. When more devices are needed repeaters should be used to amplify the signals between segments. Generally a DeviceNet network is of trunk-line characteristics where drop-lines are used for connecting devices on the bus. As a consequence termination is normally done in each end of the trunk-line. Based on this bus topology the Leine & Linde encoder doesn't enable the possibility to set active termination.

**Note:** If, for some reason, termination is required the encoder variants with M12 connectors can make use of the standard M12 termination plugs normally supplied for DeviceNet networks.

### 2.4 Baudrate setting

The Leine & Linde integrated DeviceNet encoder makes use of auto baud rate algorithm to set the baud rate. The baud rate is detected at power on. To be able to detect the baud rate the encoder must be commissioned on a network having at least one other node communicating

with defined baudrate, i.e. before power on the encoder make sure that there is network traffic.

## 2.5 Connecting the encoder

### 2.5.1 Power supply

The power supply connection of M12 equipped encoders is included in the BUS In/Out connection.

The cable gland encoders should always, in case of separate power supply cable, be equipped with a shielded power supply cable with conductor area between 0,34mm<sup>2</sup> to 1.5mm<sup>2</sup>. Permissible outer cable diameter is ø8mm to ø10mm. Located inside the back cover are two screw terminals containing the required power supply terminals marked (+) and (-).

The (+) terminal shall be used to connect the +EV-line (9-36Vdc).

The (-) terminal shall be used to connect the 0V-line.

### 2.5.2 BUS lines

The DeviceNet bus line connections of the M12 equipped encoder are constituted by a male A-coded 5 pin M12 connector (bus in), and a female A-coded 5 pin M12 connector (bus out).

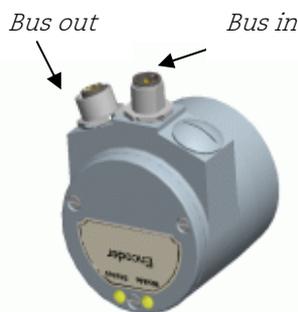


Figure 2 Orientation of M12 bus connectors

Bus in/out- lines M12 version	
Function	Pin
CAN shield	1
+E V	2
0 V	3
CAN_H	4
CAN_L	5

Table 1 Pinning bus in/out- lines M12 version

The cable gland encoders shall be equipped with twisted pair shielded cable in accordance with DeviceNet. The guidelines recommend a conductor area higher than 0,34mm<sup>2</sup>. Permissible outer cable diameter is ø8mm to ø10mm. Located inside the back cover are six screw terminals containing the required bus line terminals marked H, L and G.

The (H) terminal shall be connected to CAN\_H line.

The (L) terminal shall be connected to CAN\_L line.

The (G) terminal shall not be connected.

**Note:** The two H and L-terminals are internally connected to each other, i.e. it does not matter to which pair the bus lines are connected to.

### 2.5.3 Shielding philosophy

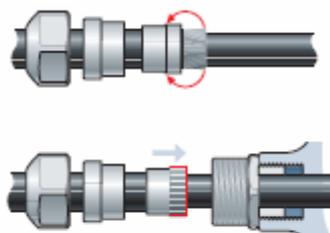


Figure 3 Cable assembling principal

To achieve the highest possible noise immunity and resistance against other EMI related disturbances the bus and power supply cables shall always be shielded. The screen should be connected to ground on both ends of the cable. In certain cases compensation current might flow over the screen. Therefore a potential compensation wire is recommended.

## 2.6 EDS file

An EDS-file is available on floppy disk (see datasheet for part number) or downloadable on our homepage, [www.leinelinde.se](http://www.leinelinde.se). Due to Leine & Linde's constant drive to support our customer with the latest updates of encoder functionality it is recommended to consult Leine & Linde representative for the latest releases. The EDS file describes:

- The communication functionality and objects as defined in the DeviceNet communication profile.
- The device specific objects as defined in the Encoder Profile.
- Manufacturer specific objects.

The EDS file serves as a template for different configurations of one device type. A DCF-file is generated from the EDS-file describing a specific configuration of the device including object values and module-Id.

**Note:** The EDS Installation procedure depends on your configuration tool; please consult your tool supplier if problems occur.

## 2.7 LED indication

In order to determine the status of the encoder two LED's are visible from the rear end of the encoder. The module LED indicates status of the encoder itself. The network status LED shows the encoder status on the network. The LED's can be **constantly on**, **off** and **flashing**. Flashing means **on** for 200 mS, **off** for 200 mS, **on** for 200 mS continuing.

### 2.7.1 Module LED

The module status LED is a bicolor LED with functionality as below.

Module LED	Indication
Off	No power.
Green	Device operational
Red	Unrecoverable fault
Flashing Green	Device in standby, commissioning incomplete
Flashing Red	Minor fault, the encoder indicates alarm

*Table 2 Module LED indication*

### 2.8.2 Network LED

The network status LED is a bicolor LED with functionality as below.

Network LED	Indication
Off	No power, Device not online
Green	Device On-line on the network
Red	Communication failure
Flashing Green	Device is On-line but has no connections in the established state
Flashing Red	One or more I/O connections are in timed-out state

*Table 3 Status LED indication*

When the encoder is on error free communication in on-line state both the module and the status LED should shown green.

### 3 Profile overview

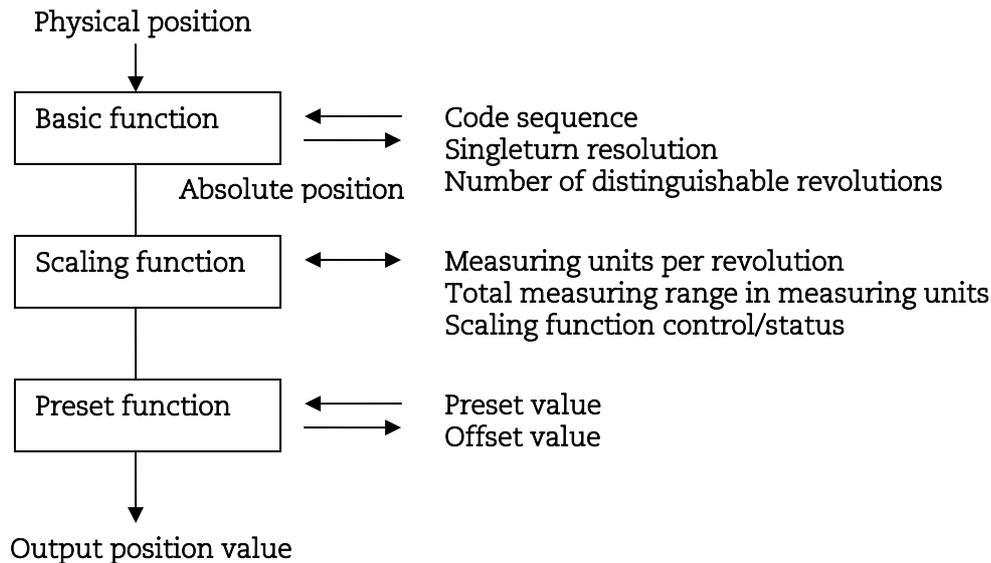
The Encoder Profile defines the functionality of encoders connected to the DeviceNet network. In the encoder profile all Objects used from DeviceNet Object library is defined and described. The profile describes the services available for reading position, scaling of position value and other useful info. For further information please see the Encoder Profile specification.

## 4 Encoder functionality

The functionality of the encoder can be divided into basic encoder functionality, describing how the physical position is transformed to a DeviceNet position value and DeviceNet functionality relating to functions, objects and attributes.

### 4.1 Basic encoder functionality

The figure below gives an overview of the basic encoder functions and how the functionality is conducted within the encoder.



*Figure 4 Basic encoder functionality*

### 4.2 DeviceNet functionality supported

The Leine & Linde DeviceNet encoder supports the predefined Master/Slave connection set (group 2 only server). For retrieving position and/or velocity value basically two principals exists, either use the Position Sensor object or to allocate a poll channel. A poll channel returns the active assembly object instance, see chapter 4.4. The advantage to use a poll channel instead of the position sensor object is that different poll allocation modes are possible, i.e. also more data then position/velocity can be retrieved on a single request. For reading position/velocity data, the following allocation modes are supported:

- **Poll I/O Connection.** When the master sends a poll request, the encoder immediately returns a poll response holding the latest calculated position value (returns the active assembly instance).
- **Bit-Strobe I/O Connection.** When sent by a master, all units affected (multi cast) respond with a bitstrobe response (returns the active assembly instance). This can be used to get a synchronised reading. Remember that the CAN-bus always priorities the node with the lowest address to send data before a node with higher address.
- **Change of State.** Returns a change of state response (returns the active assembly instance), when a change of data in the assembly is detected or a max timeout have occurred. Use the COS/delta attribute in the Position Sensor object to specify the number of position steps that must occur before a Change of State response is triggered. If e.g. the value 100 is used, the position value must be 100 steps larger or smaller compared to the position value last reported to the master before a new Change of State is triggered. Change of state can not be combined with Cyclic I/O Connection.

- **Cyclic I/O Connection.** Returns a Cyclic I/O response (returns the active assembly instance), when the specified periodic time has occurred. Cyclic I/O connection can not be combined with Change of State.

To specify which assembly instance that should be active when a poll response is returned, use the 'Set produced connection path' attribute of the Connection Class object.

#### 4.3 DeviceNet adoption of CAN identifiers

The DeviceNet communication has adopted the 11 identifier bits. They are subdivided into four separate message groups; Group 1, Group 2, Group 3 and Group4.

Identifier Bits											Hex Range	Identity Usage
10	9	8	7	6	5	4	3	2	1	0		
0	Group 1 Message ID				Source MAC ID						0x000 – 0x3FF	Message Group 1
1	0	MAC ID					Group 2 Message ID				0x400 – 0x5FF	Message Group 2
1	1	Group 3 Message ID			Source MAC ID						0x600 – 0x7BF	Message Group 3
1	1	1	1	1	Group 4 Message ID						0x7C0 – 0x7EF	Message Group 4
1	1	1	1	1	1	1	X	X	X	X	0x7F0 – 0x7FF	Invalid CAN ID's

*Table 4 CAN identifier*

- Message ID - Identifies a message within a Message Group inside a particular end-point.
- Source MAC ID - The MAC ID assigned to the transmitting node.
- Destination MAC ID - The MAC ID assigned to the receiving device. Message Group 2 allows the specification of either Source or Destination within the MAC ID portion of the CAN Identifier Field

The MAC ID uses 6 bits, which makes it possible to have 0-63 node addresses. Address 0 is not allowed. The Leine & Linde encoder uses Message Group 2.

#### 4.4 Assembly instances

The Leine & Linde encoder supports three assembly instances.

Byte	Instance 1	Instance 2	Instance 3
0	Position Data (Low byte)	Position Data (Low byte)	Position Data (Low byte)
1	Position Data	Position Data	Position Data
2	Position Data	Position Data	Position Data
3	Position Data (High byte)	Position Data (High byte)	Position Data (High byte)
4		Bit 7-2    Bit 1    Bit 0 Unused    Warning    Alarm Flag	Velocity value (Low byte)
5			Velocity value
6			Velocity value
7			Velocity value (High byte)

*Table 5 Assembly instances*

#### 4.5 Position Sensor object (0x23<sub>hex</sub>)

The position sensor object supports the following attributes.

Attribute	Attribute name	Data type
3	Position Value (Un-scaled)	UDINT
10	Position Value (Scaled)	DINT
11	Position sensor type	UINT
12	Code sequence	BOOL
13	Commissioning diagnostic control	BOOL
14	Scaling function control	BOOL
15	Position format	ENGUNITS
16	Measuring units per revolution	UDINT
17	Total measuring range in measuring units	UDINT
18	Position measuring step	UDINT
19	Preset value	DINT
20	COS/delta	UDINT
21	Work area state register	BYTE
22	Work area low limit	DINT
23	Work area high limit	DINT
24	Velocity value	DINT
25	Velocity format	ENGUNITS
26	Velocity Resolution	UDINT
41	Operating status	BYTE
42	Physical resolution	UDINT
43	Number of distinguishable revolutions	UINT
44	Alarms	WORD
45	Supported alarms	WORD
46	Alarm flag	BOOL
47	Warnings	WORD
48	Supported warnings	WORD
49	Warning flag	BOOL
50	Operating time	UDINT
51	Offset value	DINT
BOOL and BYTE = 1 byte, WORD,UINT and ENGUNITS = 2 byte UDINT and DINT = 4 byte		

*Table 6* Position sensor object

**Note:** The 'Data Type' Engineering Units (ENGUNITS) might not be common knowledge. For a description of how it works, please see Appendix K-2.30 in the DeviceNet Specification.

The remaining part of this chapter describes the majority of the attributes in the position sensor object. If more extensive explanation is required please advise the DeviceNet specification.

##### 4.5.1 Position value

Attribute 3 corresponds to the position data, un-scaled, i.e. the raw position data of the encoder. Attribute 10, position data scaled, is conditioned by the scaling attributes and the preset control.

##### 4.5.2 Position sensor type

The position sensor type attribute can either have value 01, singleturn encoder, or 02, multiturn encoder depending on encoder model commissioned.

##### 4.5.3 Code sequence

The code sequence defines whether increasing or decreasing position values are outputted when the encoder shaft rotates clockwise (CW) or counterclockwise (CCW) as seen from shaft end. If attribute 12 is defined as "0", clockwise (CW) the position value will increase when the shaft is turned clockwise seen from shaft end. In contrary "1", counter clockwise (CCW) will give increasing values when turning the shaft counter clockwise.

#### 4.5.4 Commissioning diagnostic control

The commissioning diagnostic control attribute enables internal encoder diagnostics function. If this attribute is set to "1" the diagnostics is enabled and the actual status is presented in the alarm attribute 44.

#### 4.5.5 Scaling function

With the scaling function the encoder internal numerical value is converted in software to change the physical resolution of the encoder. The scaling function control attribute 14 enables the scaling attributes to affect the position value. If this attribute is set to "1" (ON) the scaling attributes (16 and 17) affects the outputted position value. Attribute 16 defines the singleturn resolution, i.e. the number of measuring units per revolution. Attribute 17 defines the total measuring range in measuring units, i.e. singleturn time's multiturn resolution.

**Note:** When scaling a multiturn encoder the parameter "Measuring units per revolution" must be sent before the parameter "Total measuring range in measuring units".

The data type for both scaling attributes is unsigned 32 with a value range from 1 to  $2^{32}$  limited by the encoder resolution. For a 25 bit encoder with a singleturn resolution of 13 bits the permissible value for the "Measuring units per revolution" is between 1 and  $2^{13}$  (8192) and for the "Total measuring range in measuring units" the permissible value is between 1 and  $2^{25}$  (33 554 432). The scaling attributes are securely stored in case of voltage breakdown and reloaded at each start-up.

Byte	3	2	1	0
Bit	31 - 24	23 - 16	15 - 8	7 - 0
Data	$2^{31} - 2^{24}$	$2^{23} - 2^{16}$	$2^{15} - 2^8$	$2^7 - 2^0$
Attribute 16 - Measuring units per revolution				

*Table 7 Singleturn scaling attribute format*

Byte	3	2	1	0
Bit	31 - 24	23 - 16	15 - 8	7 - 0
Data	$2^{31} - 2^{24}$	$2^{23} - 2^{16}$	$2^{15} - 2^8$	$2^7 - 2^0$
Attribute 17 - Total measuring range in measuring units				

*Table 8 Total measuring range attribute format*

The encoder has two different operating modes depending on the specified measuring range. If the scaling is binary the encoder enter operation mode A, cyclic operation, and if the scaling value is non-binary the encoder enters operation mode B, non cyclic operation.

#### A. Cyclic operation (Binary scaling)

Used when operating with  $2^x$  number of turns (2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048 and 4096 number of turns). If the desired measuring range is equal to the specified singleturn resolution \*  $2^x$  (where  $x \leq 12$ ) the encoder operates in endless cyclic operation (0 - max - 0 - max ...). For example: If the position value increases above the maximum value (measuring range-1) by rotating the encoder beyond the maximum value the encoder continues from 0.

Example of a cyclic scaling:

Measuring units per revolution = 1000

Measuring range = 32000 ( $2^5 = 32$  number of turns)

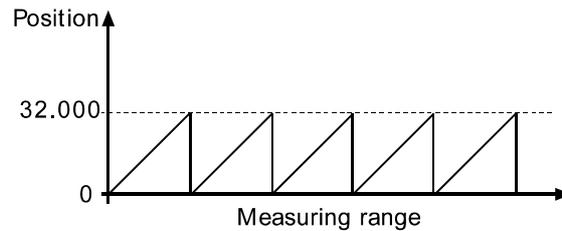


Figure 5 Cyclic scaling

### B. Non cyclic operation

If the measuring range is used to limit the encoder value range to a value not equal to the specified singleturn resolution \* 2<sup>x</sup> the output position value is limited within the operating range. If the position value increases or decreases outside the measuring range by rotating the encoder beyond the maximum value (measuring range-1) or below 0 the encoder outputs the total measuring range value.

Example of a non-cyclic scaling:

Measuring units per revolution = 100

Measuring range = 5000 (50 number of turns)

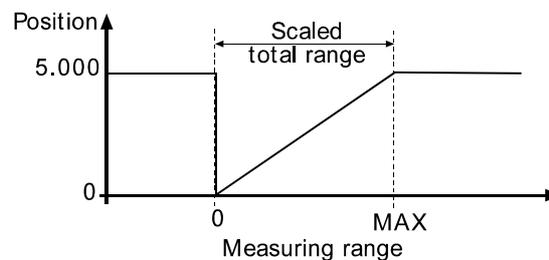


Figure 6 Non-cyclic scaling

The scaling function used in the DeviceNet encoder is limited to a singleturn resolution within one step. After downloading new scaling attributes the preset function should be used to set the encoder starting point.

**Note:** Changing the scaling function attributes should only be used at encoder standstill.

In the following formula a 25 bit multiturn encoder with a singleturn resolution of 13 bits is used as an example. Formula for the multiturn scaling function:

$$A = \frac{(\text{singleturn\_position} \times \text{measuring\_units\_per\_revolution})}{8192}$$

$$\text{output\_position} = (\text{revolution\_number} \times \text{measuring\_units\_per\_revolution}) + A$$

Where: *singleturn\_position* = the Absolute singleturn position value  
*revolution\_number* = the Absolute multiturn number

### 4.5.6 Position format

For the position format attribute only 1001<sub>b</sub>, i.e. counts, is valid as the device is a rotary encoder.

### 4.5.7 Preset, zero-set and offset value

The preset value, attribute 19, supports adapting of a desired position value to an actual position value. When a preset value is added to attribute 19 a preset to the position value is performed and an offset value is calculated.

$$\text{Setup calculation:} \quad \text{offset\_value} = \text{preset\_value} - \text{current\_value}$$

The offset value is continuously added to the internal encoder position value resulting in a position value continuously affected by the preset value.

$$\text{Runtime calculation:} \quad \text{output\_position} = \text{current\_position} + \text{offset\_value}$$

The calculated offset value will remain constant (until the preset value is changed) and can be accessed in attribute 51 and is securely stored in case of voltage breakdown and reloaded at each start-up.

**Note:** The preset function should only be used at encoder standstill. Depending on encoder type the number of possible preset cycles is limited; please consult Leine & Linde for more information.

Zero setting of the encoder can be done two folded. Using the preset attribute and set the preset value to zero makes a zero-set of the encoder. Also, if the zero set button, inside the back cover of the encoder, is pushed for at least 1 second the position value will be set to zero.

#### 4.5.8 COS/delta

The COS/delta attribute defines the number of position values that must be passed before a change of state response is generated. If this value is set to 0 a COS response will be generated at every change in position value.

#### 4.5.9 Position limits

Using the attributes 22-23 a work area can be defined, i.e. high and low limit value. In attribute 21 the status of the work area is presented. Whenever the actual position value is out of range bit 0 of the state register is set. Bit 1 and 2 indicates weather an under or over flow is detected.

Bit 7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Reserved	Reserved	Reserved	Reserved	Reserved	Range underflow	Range overflow	Out of range

*Table 9 Position state register*

#### 4.5.10 Velocity

Attribute 24 presents the velocity value of the encoder shaft. The resolution and unit of the represented velocity value is stated by attribute 25 and 26. Attribute 25, velocity format, is a read/write attribute defining the unit of the outputted velocity value. The attribute value 1F04<sub>hex</sub>, count/sec, is default but also DF0F<sub>hex</sub>, RPM, and DF0E<sub>hex</sub>, RPS, is accepted.

#### 4.5.11 Operating status

This attribute basically contains status of the code sequence and scaling functionality. No vendor specific status bits are implemented.

Bit	Description	FLASE (0)	TRUE (1)
0	Code sequence	Increasing	Decreasing
1	Scaling	Off	On
2..4	Reserved by DeviceNet		
5..7	Vendor specific		

*Table 10 Operating status*

#### 4.5.12 Measuring range

Attribute 42, singleturn resolution, and attribute 43, multiturn resolution, defines the physical resolution of the encoder, i.e. the basic resolution of the encoder when scaling function is disabled.

### 4.5.13 Alarms

The encoder supports position error alarm according to the definition below.

Bit	Description	FLASE (0)	TRUE (1)
0	Position error	NO	YES
1	Diagnostic error	NO	YES
2..11	Reserved by DeviceNet		
12..15	Vendor specific		

*Table 11 Alarms*

Bit	Description	FLASE (0)	TRUE (1)
0	Position error	Not supported	Supported
1	Diagnostic error	Not supported	Supported
2..11	Reserved by DeviceNet		
12..15	Vendor specific		

*Table 12 Supported alarms*

### 4.5.14 Warnings

The encoder supports warnings according to the definition below.

Bit	Description	FLASE (0)	TRUE (1)
0	Frequency exceeded	NO	YES
1	Light control reserve	Not reached	Error
2	CPU watchdog	OK	Reset generated
3..10	For specific definition, see encoder profile		
11..12	Reserved by CIP	Always 0	
13..15	Vendor specific		

*Table 13 Warnings*

Bit	Description	FLASE (0)	TRUE (1)
0	Frequency exceeded	Not supported	Supported
1	Light control reserve	Not supported	Supported
2	CPU watchdog	Not supported	Supported
3..10	For specific definition, see encoder profile	Not supported	Supported
11..12	Reserved by CIP	Always 0	
13..15	Vendor specific		

*Table 14 Supported warnings*

**Note:** Bit 1, light control reserve, is supported depending on encoder model; please consult Leine & Linde for more information.

### 4.5.15 Operating time

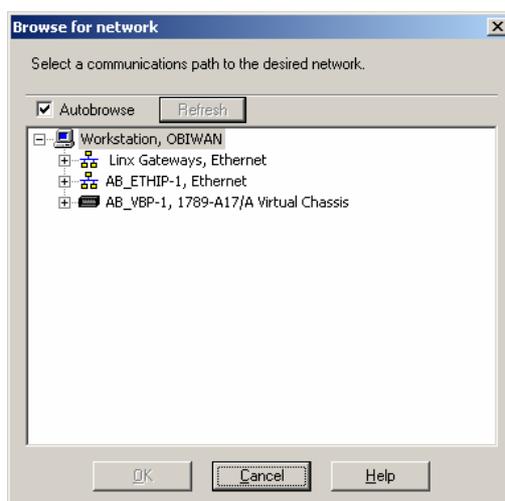
The operating time attribute is read only and is incremented as long as the encoder is powered on. The operating time is present as tenths (0.1) of an hour, i.e. attribute value 10 is equal to 1 hour operating time.

## 5 Encoder commissioning example

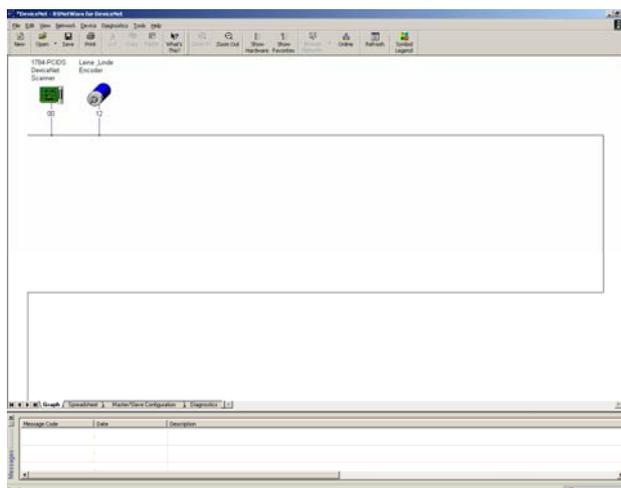
This example shows a simple setup of the encoder for cyclic transmission of the position value.

**Note:** Since commissioning is software tool dependent, the steps below must be adapted to the specific tool in use. Also the example only shows how the scanner part is made.

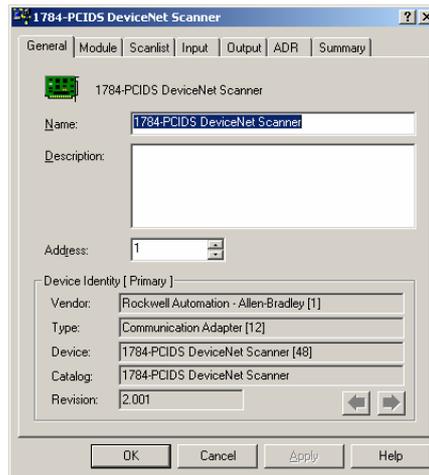
- 1 Set the physical address (Node Number) of the encoder using the address switches.
- 2 Power up the encoder.
- 3 In RSNetworx, click on the “go on-line” button to connect to the DeviceNet network.



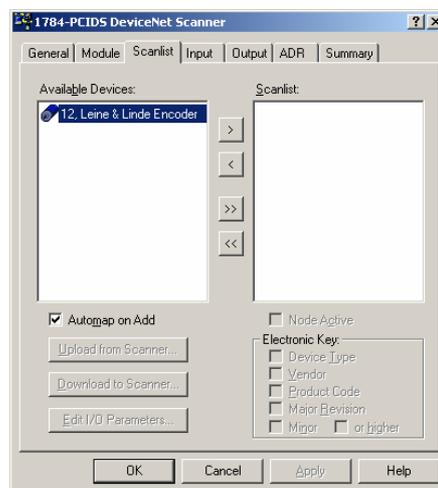
- 4 The encoder will send a Boot-up message (Duplicate MAC message) when the baudrate is detected. If no other node responds on the MAC request the encoder is online and waiting for a master to connect. In case another node is responding the encoder enters offline and indicates accordingly on the LED's.
- 5 Scan the network. When all expected units on the network have appeared, double-click the scanner icon to bring up its property view.



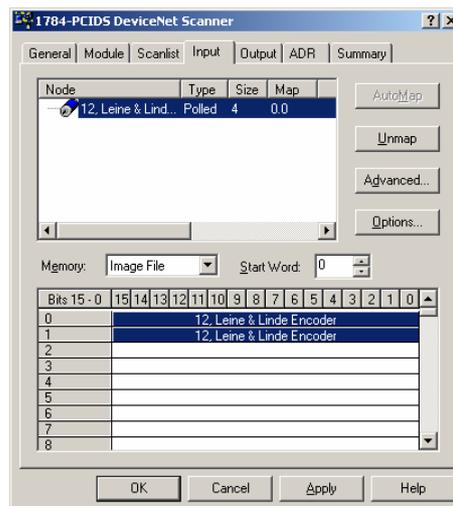
- 6 Select the Scanlist tab. All units added to the Scanlist will be polled by the scanner/PLC.



7 Click the “>” button to add the encoder to the Scanlist.



8 Select the Input tab.



9 Confirm the mapping of the encoder value in the PLC image file. Click OK to close the property page and download the changes to the scanner device.

10 Now the scanner will gather all position data from all units in its scanlist and put in the image file.

## Appendix A, History

Revision	Date	Changes
Rev. 1.0	07-07-01	First release
Rev. 1.1	07-10-30	Extended information regarding scaling parameters Excluded information regarding LLS service