Eilersen Electric A/S
Kokkedal Industripark 4
DK-2980 Kokkedal
DENMARK
Tel: +45 49 18 01 00
Fax: +45 49 18 02 00

MCE9637 DeviceNet Module

MCE9637 for transmitting status and weight of digital loadcells

Applies for:

PIC no.:	9637C_DN.CONCTR_4.000224.0
Document no.:	224MU0a.DOC
Date:	2006-06-01
Rev.:	0a

1) CONTENTS

2) INTRODUCTION

2.1 Introduction

This document describes the use of the Eilersen Electric MCE9637 DeviceNet communication module, when equipped with program listed on the front page. With this program the MCE9637 communication module is capable of transmitting weight and status of up to 4 loadcells in a single telegram. Each loadcell is connected to the communicationsmodule through a MCE9610 loadcell interfacemodule.

It is possible to connect the MCE9637 communication module to DeviceNet, where it will act as a slave. It will then be possible from the DeviceNet master to read status and weight for each of the connected loadcells. Functions as zeroing, calibration and calculation of system weight **must** be implemented on the DeviceNet master.

Exchange of data between DeviceNet master and slave is performed as described in the following.

3) DATA EXCHANGE

3.1 DeviceNet communication using PPO

DeviceNet communication with the MCE9637 communication module uses a so called 'parameter-process data object' (PPO) consisting of 26 bytes. This telegram (object) is only used when transferring data from the slave to the master, since no data are transmitted from the master to the slave. The structure for this telegram is as follows:

Lc Register		**Lc Status(0)**		**Lc Weight(0)**					**Lc Status(3)**		**Lc Weight(3)**			
0	1	2	3	4	5	6	7		20	21	22	23	24	25

The byte order (MSB/LSB first?) for the individual parts of the telegram is determined by jumper JU3. Normally this jumper is set from the factory so that LSB comes first. In the following bit 0 will represent the least significant bit in a register.

LcRegister is two bytes that constitute a bit register for indication of connected loadcells detected during power on. Hence bit 0-3 will be ON, if the corresponding loadcell (address) was detected during power on.

LcStatus(X) is two bytes that constitute a register containing the actual status for loadcell **X**. The individual bits in the status register have the following function:

BIT-NO	FUNCTION
0	**Invalid/missing ’sample’ ID** Bad connection between communicationsmodule and loadcell module. Not all telegrams from the communicationsmodule are received in the LC module.
1	**Timeout – Loadcell not connected ?** Check that the loadcell is connected to the loadcell module.
2	**Loadcell not synchronised** Bad connection between loadcell and loadcell module or very heavy over- or underload.
3	**Hardware synchronisation error** Loadcell samplings are not synchronised.
4	**Power failure** The supply voltage is to low.
5	**Overflow in weight calculation** Internal error in loadcell module.
6	**Invalid/missing ’latch’ ID** Bad connection between communicationsmodule and loadcell module. Not all telegrams from the communicationsmodule are received in the loadcell module.
7	**No answer from loadcellmodule** No data are received from this loadcell module. This can be caused by the removal of the loadcell module, no power to the module or that the connection between loadcell module and communicationsmodule is broken.

LcWeight(X) is four bytes constituting a register containing the actual weight signal from loadcell **X** in gram. Depending on JU4 the weight will be in the following format:

JU4=OFF: in ”32 bit signed integer” format.
JU4=ON: in ”IEEE754” format.

Since only status and weight for the loadcells are transmitted in the telegram functions such as status handling, calculation of system weight, zeroing and calibration **must** be implemented on the DeviceNet master.

Calculation of system weight is done by addition of the weight registers for the connected loadcells. Note that the result is only valid if all status registers for the connected loadcells indicate **no** errors. It should also be noted that it is up to the master to ensure the usage of consistent loadcell data when calculating the system weight (the used data should come from the same telegram).

3.2 IEEE754 format

If by usage of JU4 representation of data on IEEE754 format is selected this is done as follows:

Byte1 (MSB)		**Byte2**		**Byte3**	**Byte4 (LSB)**
bit7	bit6 bit0	bit7	bit6 bit0	bit7 bit0	bit7 bit0
S	2^{7} ……… 2^{1}	2^{0}	2^{-1} …….. 2^{-7}	2^{-8} ………….... 2^{-15}	2^{-16} ……….…. 2^{-23}
Sign	Exponent		Mantissa	Mantissa	Mantissa

Formula:

$$\text{Value} = (-1)^{S} * 2^{(\text{exponent}-127)} * (I+\text{Mantissa})$$

Example:

Byte1	Byte2	Byte3	Byte4
0100 0000	1111 0000	0000 0000	0000 0000

$$\text{Value} = (-1)^{0} * 2^{(129-127)} * (1 + 2^{-1} + 2^{-2} + 2^{-3}) = 7.5$$

Please note that if MSB first has been selected using JU3 the byte with the “sign” will come first in the weight indications, and if LSB first has been selected the byte with the “sign” will come last in the weight indications.

4) ZEROING AND CALIBRATION

4.1 Zeroing procedure

Zeroing of the system (all loadcells) should be performed as follows:

1) The weighing arrangement should be empty and clean.

2) The DeviceNet master reads and stores the actual weight values for the connected loadcells in corresponding zeroing registers.

3) After this the actual weight for loadcell **X** can be calculated as:

   LcGross(**X**) = LcWeight(**X**) – LcZero(**X**)

   and the system weight (uncalibrated) for the connected loadcells is calculated as:

   SystemWeight = LcGross(0) + LcGross(1) + ….

4.2 Calibration procedure

Fine calibration of the system should be performed as follows:

1) Check that the weighing arrangement is empty, and that the gross weight is zero. Zero if necessary.

2) Place a known load (calibrationweight) on the weighing arrangement.

3) Calculate the calibrationfactor that should be multiplied on the system weight in order to achieve correct showing as:

   Calibrationfactor = (Calibrationweight)/(Actual showing)

   After this the determined calibration factor is used to calculate the calibrated weight as follows:

   Sys.Weight(Calibrated) = Calibrationfactor * Sys.Weight(Uncalibrated)

   If the determined calibrationfactor falls outside the interval 0.9 to 1.1 it is very likely that there is something wrong with the mechanical part of the system.

5) INSTALATION OF SYSTEM

5.1 Checklist during installation

During installation of the system the following should be checked:

1) If necessary the DeviceNet master should be configured to communicate with the DeviceNet module (MCE9637) using the supplied EDS file.

2) The loadcells are mounted mechanically and connected to the DeviceNet module (MCE9637) using their corresponding loadcell interface module (MCE9610). The loadcell addresses are set using the DIP-switches (Sw1.5-Sw1.8) on the MCE9610 modules, so that they forth running from address 0 (0-3).

3) The DeviceNet module (MCE9637) is connected to the DeviceNet, and possibly a termination is made at this DeviceNet slave.

4) The baudrate of the DeviceNet module (MCE9637) is set using Sw2.1-Sw2.2 and its address is set using Sw2.3- Sw2.8. Power is applied and the DeviceNet communication is started.

5) Verify that the yellow LED (D1) on the DeviceNet module (MCE9637) is lit, and that the red LED (D2) is not lit. Verify that the TXBB LED on the DeviceNet module is lit and that the TXBB LED's on the loadcell modules (MCE9610) are also lit (can flash slightly). Verify that both the MS and NS LED's end up being lit green constantly.

6) Verify that the DeviceNet module (MCE9637) has found the correct loadcells (**LcRegister**), and that no loadcell errors are indicated (**LcStatus(x)**).

7) Verify that every loadcell gives a signal (**LcWeight(x)**) by placing a load directly above each loadcell one after the other (possibly with a known load).

The system is now installed and a a possible zero and fine calibration is made as described earlier. Finally verify that the weighing system(s) returns a value corresponding to a known actual load.

Note that in the above checklist no consideration has been made on which functions are implemented on the DeviceNet master.

6) MCE9637 HARDWARE DESCRIPTION

6.1 DIP-switch settings

The MCE9637 module is equipped with two DIP-switch blocks. DIP-switch block 1 has the following function:

SWITCH	FUNCTION
Sw1.1-Sw1.4	*Reserved for future use*

DIP-switch block 2 has the following function:

SWITCH	FUNCTION
Sw2.1-Sw2.2	**Setting of DeviceNet DataRate (DR)** The desired baudrate is set according to the table shown below. Note that these switches are only read during power-on.
Sw2.3-Sw2.8	**Setting of DeviceNet Node Address (NA)** The address (0-63) is set as the DIP-switches are binary coded, so that Sw2.8 is LSB and Sw2.3 is MSB. Note that these switches are only read during power-on.

The baudrate of the MCE9637 module is set according to this table:

Sw2.2	Sw2.1	Baudrate
OFF	OFF	125 kbps
ON	OFF	250 kbps
OFF	ON	500 kbps
ON	ON	Not allowed

6.2 Jumpers

The MCE9637 module is equipped with 5 internal jumpers. These jumpers have the following function:

JUMPER	FUNCTION
JU2	**Test mode** JU2 OFF: Normal mode (Default at delivery. Should'nt be changed) JU2 ON: Test mode The jumper <u>must</u> be OFF during normal operation.
JU3	**Selection of LSB/MSB data format** The jumper determines the byte order in which data are transmitted/received. JU3 OFF: LSB first (normal setting from factory) JU3 ON: MSB first
JU4	**Selection of (32 Bit Signed Integer) / (IEEE754) data format** The jumper determines if the weight indications in the telegram are in 32 bit signed integer or in IEEE754 floating point format. JU4 OFF: 32 bit signed integer format (normal setting from factory) JU4 ON: IEEE754 floating point format
JU5	*Reserved for future use*
JU6	**Test mode** JU6 OFF: Normal mode (Default at delivery. Should'nt be changed) JU6 ON: Test mode The jumper <u>must</u> be OFF during normal operation.

6.3 Light emitting diodes

The MCE9637 module is equipped with 6 light emitting diodes (LED's). These LED's have the following function:

LED	FUNCTION
TxBB (Green LED)	**Communication with loadcells** The MCE9637 communicates with the loadcells
D1 (Yellow LED)	**DeviceNet Voltage Detected** The MCE9637 module has detected DeviceNet voltage on the DeviceNet connector.
D2 *(Red LED)*	*Reserved for future use*
TxCAN (Green LED)	**CAN bus TxD (Transmit Data)** The MCE9637 module transmits data across the CAN bus.
MS (Green/Red LED)	**Module Status LED** The MCE9637 Module Status LED, that can be lit/flashing in different colours depending on the status of the module. The function of the MS LED is given in the table below.
NS (Green/Red LED)	**Network Status LED** The MCE9637 Network Status LED, that can be lit/flashing in different colours depending on the status of the network. The function of the NS LED is given in the table below.

Please note that the LED's will flash shortly during power-up during the selftest of the module. The MS and NS LED's will shortly flash Green/Red. The MS and NS LED's can in conjunction with the table below be used for error finding.

Light emitting diode	Colour	Status	Description
MS	Green	ON	**Normal Operation.** Communication performed normally.
		Flashing	**Standby State.** The unit needs supervision.
	Red	ON	**Unrecoverable fault.** A timer error, memory error or other system error. The unit may need replacing.
		Flashing	**Recoverable fault.** Configuration error, DIP-switch not set correct or similar error. Correct error and restart unit.
	---	OFF	**No power.** The power is disconnected or the unit is being restarted.
NS	Green	ON	**On-Line, Connection OK.** The unit is On-Line and a connection with the master has been established.
		Flashing	**On-Line, No Connection.** The unit is On-Line but no connection to the master has been established.
	Red	ON	**Critical Communication Error.** The unit has detected an error that makes it impossible to communicate on the net-work (duplicate MAC Id or Bus-Off error).
		Flashing	**Communication Time-Out.** One or more I/O connections are in the Time-Out state.
	---	OFF	**No power/Off-line.** The device may not be powered.

6.4 EE-bus connector

The MCE9637 module is equipped with a 10 pole connector for connection to the Eilersen Electric EE-bus. Hereby connection to the individual MCE9610 loadcell modules as well as to the power supply for the MCE9637 module is achieved. The connection is made using a flat cable with mounted connectors for the individual modules. The 10 pole connector has the following connections:

J4 Connector	Function
J1.1-J1.2	RS485-B (**negative** line)
J1.3-J1.4	RS485-A (**positive** line)
J1.5-J1.6	0VDC (Gnd1)
J1.7-J1.8	+24VDC (Vin1)
J1.9-J1.10	I/O line

6.5 DeviceNet connector

The MCE9637 module is equipped with a 5 pole connector for connection to DeviceNet. The connection is according to the DeviceNet specification and is made as follows:

J2 Connector	Function	Colour
J2.1	V-	(Black)(0VDC input)
J2.2	CAN_L	(Blue)
J2.3	SHIELD	(Grey)
J2.4	CAN_H	(White)
J2.5	V+	(Red)(24VDC input)