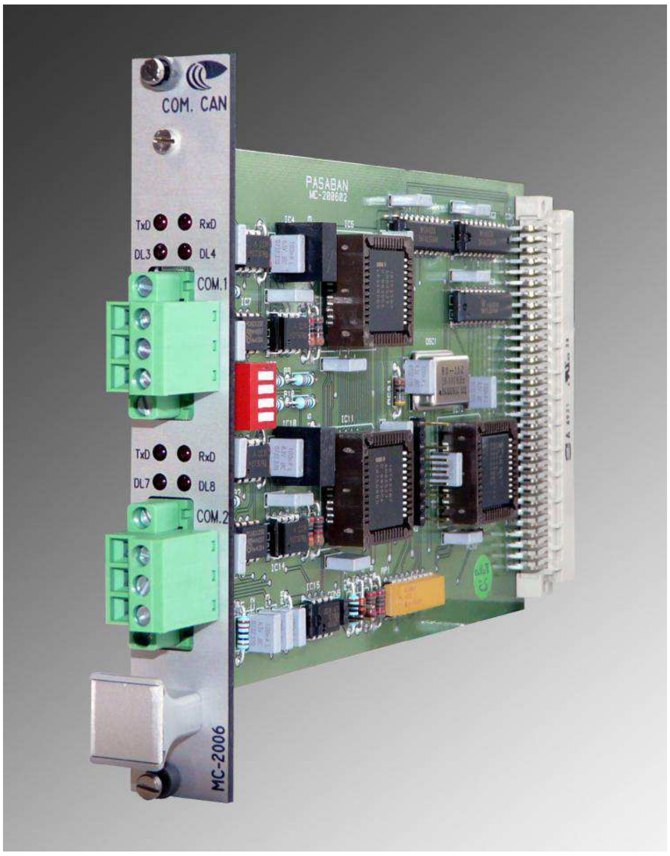
CAN COMMUNICATION BOARD (MC-2006)

Description

Board for connexion of several peripheal that require CAN and OPENCAN communication protocol. Has 2 independent gates “with intern protocol adaptable to users needs”. Its frontal part is made up of:

- 2 gates for 2 CAN lines (max. 128 elements per line)
- 4 LED diodes per gate to inform about the communication state:
  TXD (transmission) RXD (reception)
  DL3 (communication error) DL4 (without use)



Application

Devices that require CAN or OPENCAN protocol, for example:

- Remote I/O boards (opencan)
- CPUs intercommunication.
- Magnetostrictive displacement sensors.
- Micromaster motors (opencan)
- Remote equipments: CLM (Motorized longitudinal cut)

Additional data

- ✓ Europe board format.
- ✓ 2 CAN channels per board (82527 CAN controller).
- ✓ Opto-isolated before TxD and RxD signals
- ✓ DC/DC (5V/5V) stage to isolate 0V from the bus regarding to those of each channel.
- ✓ Information about each gate communication state through LED diodes.
  - TXD: usually (FLASHING)
  - RXD: usually (FLASHING)
  - DL3/DL8: usually (FLASHING)
  - DL4/DL8: usually (OFF)

Common data

- ✓ Starting line resistor with RS-485 protocol configurable through switch (∞_{Ω}, 60_{Ω}, 120_{Ω}). The two first switch positions correspond to gate 1 and the following two to gate 2.
- ✓ Internconnexion between PLCs.
- ✓ Device base address according to board position in the RACK.

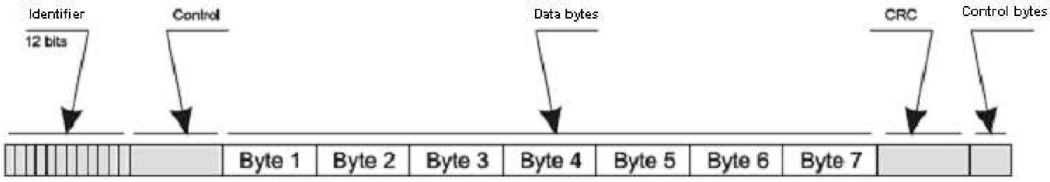
CANOPEN

It is an standardized protocol that defines the use of identifiers and the data bytes using way of the frames and speeds. There are rules for remote board use; frequency converters, PLCs, encoders, etc.

CAN

The CAN communication is based on transmissions of "CAN frames".(*)

The frame starts with a number of bits named **IDENTIFIER** that defines the message and the priority of this frame, then, there is a bits of **CONTROL** serie, the **DATA** bytes between 0 and 8 maximum, **CRC** bits (Cyclical Redundancy Check) and the last **CONTROL** bits of end of frame.



The integrates used for CAN communication generate all the control bits and CRC. Just the identifier number, the data (bytes) that want to be send and the execution order have to be indicated. If the CAN controllers try to send two different frames at the same time, the frame with lower identifier has priority and will leave first from bus. All the priorities management is done by the integrates.

The integrates can be also programmed to capture the data of a frame with a particular identifier. Integrates can also be configurated so they can answer with a frame when they receive a what is called "remote frame"

82527 CONTROLLER *(Intel®)*

It's the controller used in the CPU to communicate with CANbus. The MC-2006 card has 2 CAN communication channels, each of them controlled by a 82527 integrated.

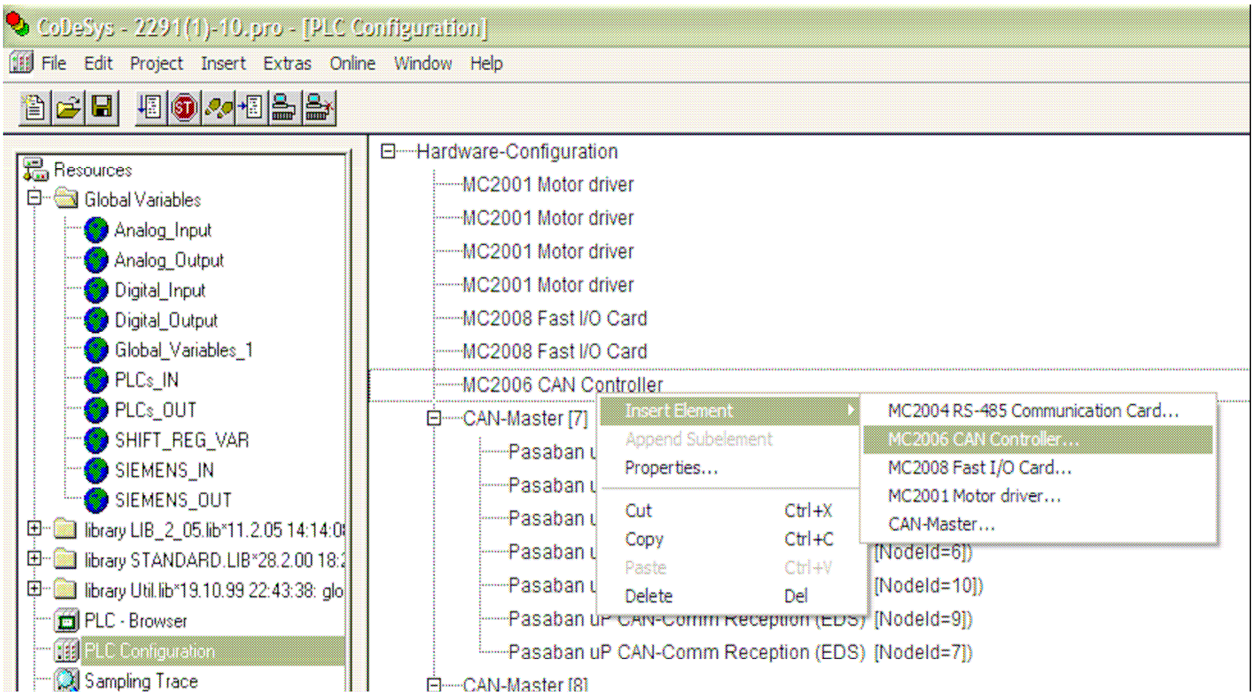
In this integrate 15 different frames can be programmed with its identifier and its data, 14 of them can be used to send or receive CAN frames, and one just to receive.

- If two CPUs want to send information at the same time there are not short-circuit problems. 0 prevails over 1.
- Frames with lower identifier will go out from the bus before, although it is send or although there is another frame.

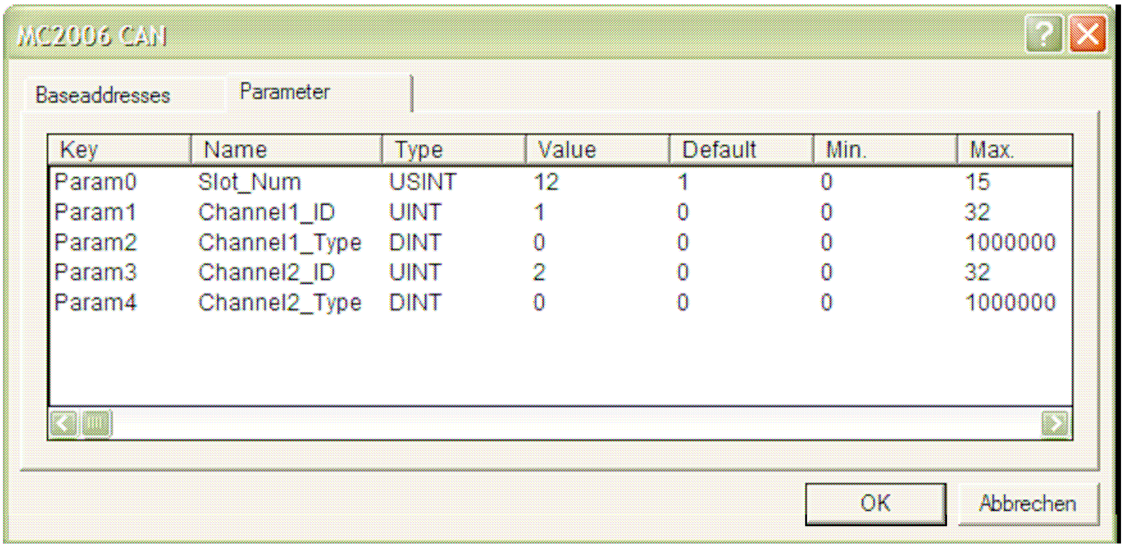
(*) *A CAN frame consist of a bits serie where the maximum number of information bytes is 8.*
(**) *A remote frame is a frame without data, for example, if we have a remote inputs board, we can prepare the CAN controller of this card with an X identifier and continuously go writing on the integrate the inputs state. When the integrate receives a remote frame whose identifier is X, automatically, the integrate sends another frame with the same X identifier and the data with the inputs state.*

CAN board configuration *(Codesys)*

When we have opened the Codesys program, placing our mouse on the resources file/PLC configuration, with a right click on the mouse…



…a new menu is opened…

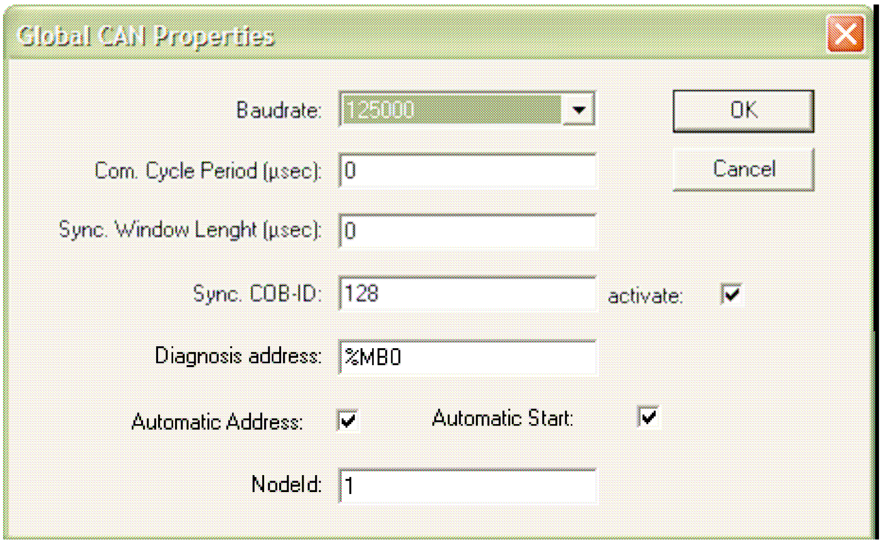


Slot_Num
Slot number where the card is located in the Rack (Observe "Value").

Channel1_ID and Channel2_ID
They define the CAN channel number inside the RACK.
The default values, minimum and maximum are not configurable and **Channel1_Type** and **Channel2_Type** are not used as parameters. Neither are used the **Baseaddresses** options of the configuration menu of the MC2006 element.

CAN element insertion

Once the CAN controller parameters are configurated is proceed to insert a CAN master element (channel configuration of the card that is going to be used). The insertion way of this element is quite similar to the MC2006. This element defines the communication speed of every physic gate of the MC2006.



Baudrate
Defines the channel communication speed

Sync. COB-ID
Indicates what channel number wants to be configurated with the established Baudrate. The value introduced in this field has to correspond with the identifier given to any of the MC2006 card channels that are defined as Channel1_ID and Channel2_ID. If the field does not correspond with any of the identifiers of the MC2006 card channel, CoDeSys will inform through the message "Runtime error #101 (There is not any target assigned to CANMASTER)".

NodeId
It's a required field and a high numeration is assigned to it (128 the first, 127 the second, etc.) not to interfere in the assignment of the communication channels Channel1_ID and Channel2_ID of the MC2006 board.

Not commented parameters must not be altered.

STARTING LINE/ENDING LINE RESISTANCE

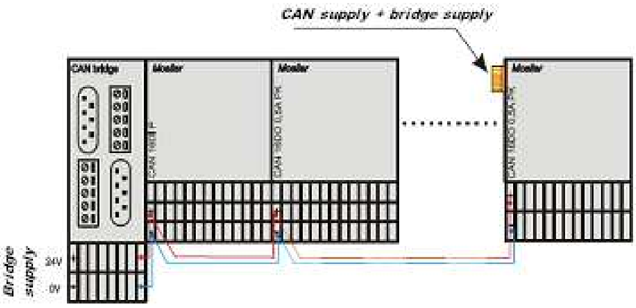
Through the intern switch in the MC-2006 the starting line resistor value is indicated. The resistor in the other end will probably be incorporated in the last device of the line, probably through a unique switch.

	Switch				Resistor
	1	2	3	4	(Ω)
Gate 1	0	0	X	X	∞
	0	1			120
	1	0			120
	1	1			60
Gate 2	X	X	0	0	∞
			0	1	120
			1	0	120
			1	1	60

MOELLER REMOTE MODULES *(codesys)*

The moeller modules are connected over the bases called "bridge", To theses bases the communication and supply cables are connected (the one of the module (24_{DC}) as well as the ones of the cards).
It's individually communicated with each of the modules and not with the bridges.

- The bridge is a CAN amplifier to which several I/Os modules can be connected and supply's the modules communication part.
- For the actuation of every module inputs and outputs, every module has to be supplied.
- Every module has two <u>led diodes</u>, **RUN** and **BF.**



The first time that the modules are supplied through the bridge, both leds remain turn on. RUN led means correct supply (green) and BF the connexion state (red: disconnected, orange: connexion error, green: connected).

- Physically the starting and ending line have to be marked with a 120Ω resistor. Each MC-2006 channel (starling line) has a switch by which two possible resistance value can be obtained (60Ω: 00 and 11, 120Ω: 01 and 10).
- The PLC has to initiate the remotes and put them in connected mode. (RUN: green, BF: green), being able to activate the outputs and read the inputs.
- If the PLC does not send the frame "Guarding Life" during 1.5 sec., the module returns to disconnection state and the BF led gets red (all outputs set to zero).

- On the back side of the module there are two circular switches, through them a number can be assigned to mange the module to communicate with MC-2006. This number can be configurated between 01 and 7F. 00 can not be used because this number is used for communication speed configuration.

COMMUNICATION SPEED MOELLER MODULE CONFIGURATION

To select the transmission speed (from 10kbit/s to 1Mbit/s)of a I/O WINbloc module (125 kBit/s by default) the following process has to be realized:

- Move with a small flat screwdriver the two selectors labelled as H and L of the superior side at 0 without tension.
- Apply tension to the I/O WINbloc module. The transmission ratio by default is stored in EEPROM serie. This process is indicated for 5 seconds of flickering of the red LED "BF" (flickering frequency 2 Hz)
- During 5 seconds the LED will flick, during that time selector H must change to F and selector L set to 0.
- As soon as red LED "BF" stops flickering, there are 10 seconds to fix a value (through selector L) the speed transmission (between 0 and 8 according to following table). Therefore, the module must have a value of F0h to F8h at the end of this 10 seconds.

Index	0	1	2	3	4	5	6	7	8
Bit rate (kbit/s)	1000	800	500	250	125	100	50	20	10

- The confirmation that the transmission index has been selected is indicated by the LED “BF” flickering (flickering frequency 1 Hz). This LED will flicker from 0 to 8 times, depending on the selected value.
- The confirmation that the programming process has finished is indicated by the flickering of the “BF” red LED during 5 seconds (flickering frequency 5 Hz).

The communication is done through CAN protocol, therefore, it is necessary to use some physical gate of the MC-2006. Placing over CAN-master previously introduced, through “insert” option or with the right bottom of the mouse, choosing the option “Appen Subelement”, the card model is chosen.

The unique configurable field is Node ID (decimal value). This one has to correspond with the physical direction of the card given by the "switches" (hexadecimal value). For agreement Node ID corresponds with the input or analogous output.

INPUTS

They will be assigned to node identifiers between **1 and 63.**

DIGITAL

To access to the inputs configurated at bit level, it will have to be done as %QX64.0 for lower weight bit and %QX64.F for the one of highest weight.
If the access to realize is at word level a %IW1 will allow knowing the state of all the bits corresponding to word input 1, that for agreement corresponds to X01.

ANALOG

The PLC's real directions are determined by the following way:

Address = 128 Node ID x 4 being { 128 -> origin area memory for analog inputs.
Node ID -> nodo identifier (remote card switches)

OUTPUTS

They will be assigned to node identifiers between **64 and 127.**

DIGITAL

To access to the outputs configurated at bit level, it will have to be done as %QX64.0 for lower weight bit and %QX64.F for the one of highest weight.
If the access to realize is at word level a %QW64 will allow knowing the state of all the bits corresponding to word input 1, that for agreement corresponds to Y00.

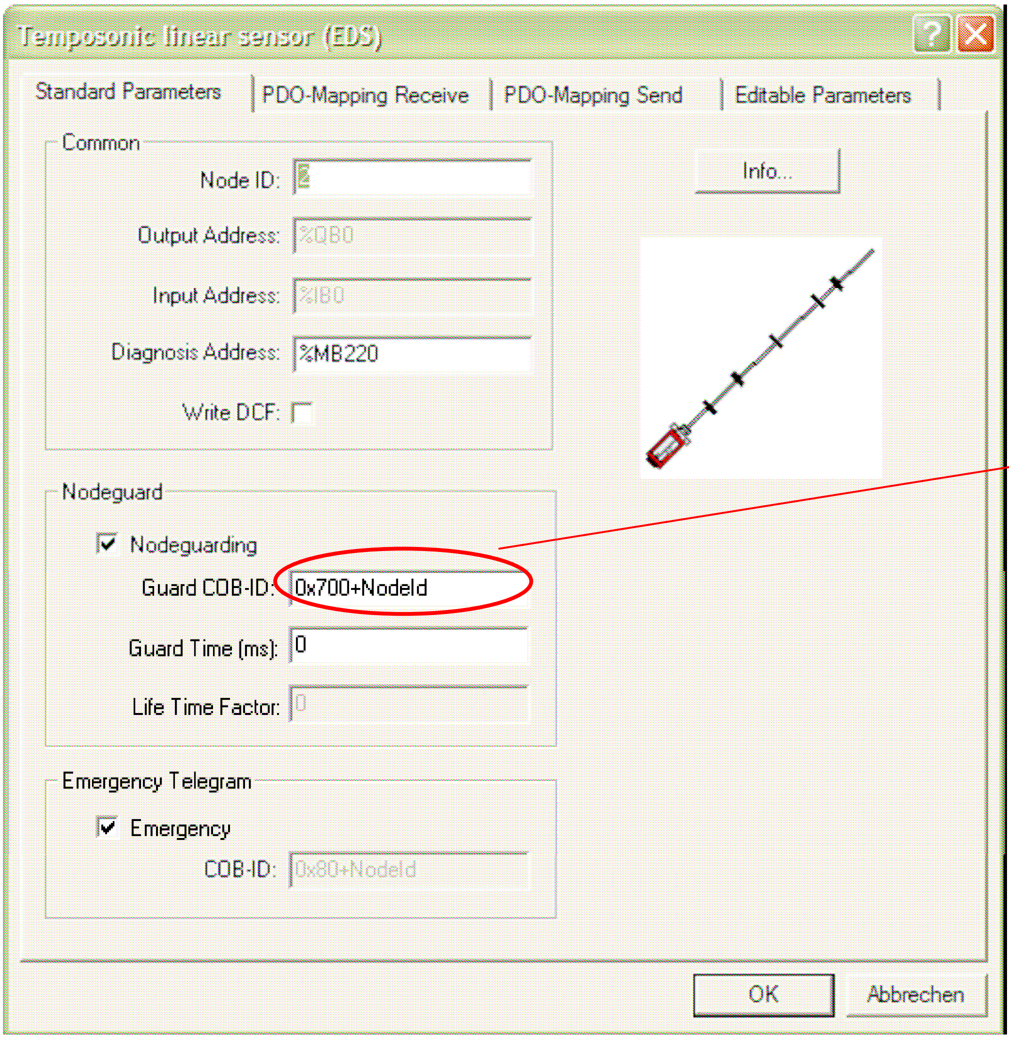
ANALOG

The PLC's real directions are equally determined:

Address = 128 Node ID x 4 being { 128 -> origin area memory for analog outputs.
Node ID -> nodo identifier (remote card switches)

MAGNETOSTRICTIVE DISPLACEMENT SENSOR *(codesys)*

Thanks to the software Codesys, the incorporation of a magnetostrictive bar under a CAN gate is easy. Having the Codesys programme opened, placing the mouse on the resources file\PLC configuration, selecting the CAN element, with the right bottom of the mouse chose the magnetostrictive bar. After, next window will be opened, in which only it will be necessary to introduce the code of the bar in the corresponding field.

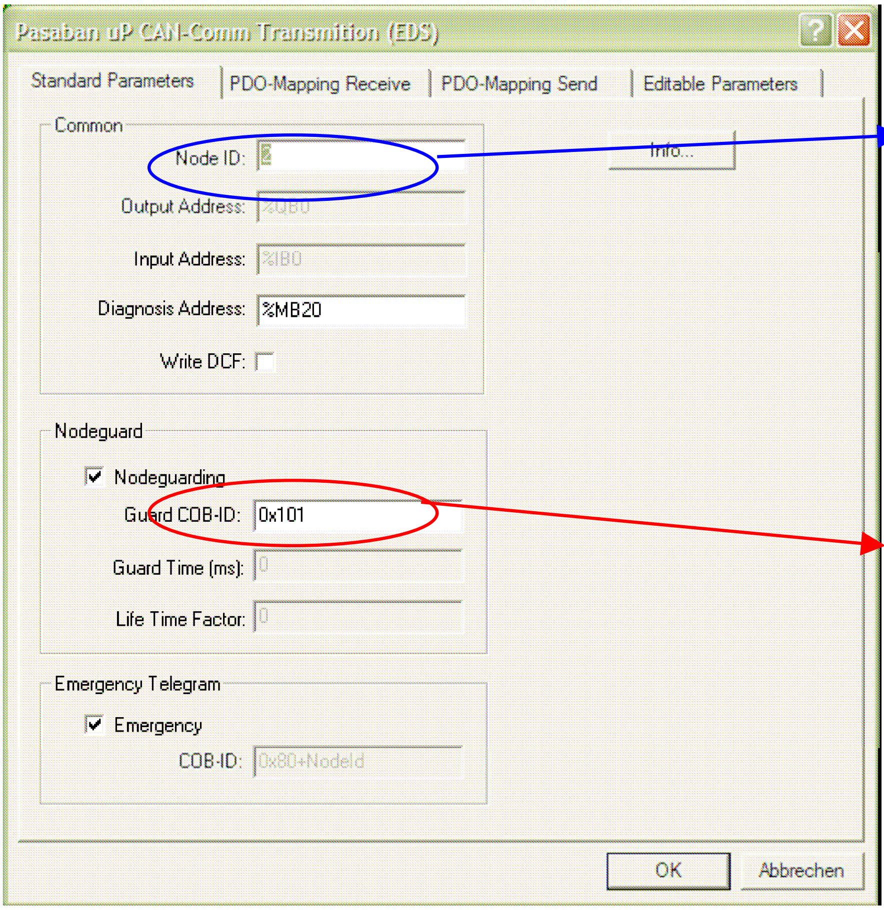


Magnetostrictive bar code

Sometimes, instead of introducing the mentioned code through the PLC configuration, it's carried out introducing it directly through the constants area of the principal PC of the machine, as, for the clients maintenance department is more comfortable not to open the program whenever they need to replace this element. Even so, despite introducing the bar code from the PC it is necessary to have indicated in the PLC configuration that this element exists.

COMMUNICATION BETWEEN PLCs *(codesys)*

Selecting the CAN-Master element on which the communication between PLCs is found, are included the reception and transmission frames that are necessaries of analogous form to the rest of elements on the PLC configuration
Whenever a new frame is incorporated, the system assigns a number and the user indicates an identification name. In these names has to be indicated in which PLCs are relationated through this frame.



Incremental value that system automatically assigns whenever a new frame is introduced.

Identification name set by user.

Identification names are assigned according to following theory:

- Output frame (transsmision) 0x origin PLC Word number
  Example: 0x101 => Origin PLC: 10
  Word: 0

- Input frame (reception) 0x destination PLC Word number
  Example: 0x110 => destination PLC: 11
  Word : 0

NOTE: Very important to bear in mind that the number of output frames in a PLC has to coincide with the number of inputs frames in the other PLC to which is intertwine. In the opposite case, the led diodes will inform about an anomalous situation in the CAN line. The same thing has to be considered for input frames.