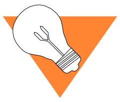
MicroControl
Systemhaus für Automatisierung

µCAN.4.ci-BOX

Manual counter module
Version 2.02

Document conventions

For better handling of this manual the following icons and headlines are used:



This symbol marks a paragraph containing useful information about the device operation or giving hints on configuration.

This symbol marks a paragraph which explains possible danger. This danger might cause a damage to the system or damage to personnel. Read these sections carefully!

Keywords

Important keywords appear in the border column to help the reader when browsing through this document.

MicroControl GmbH & Co. KG
Lindlaustraße 2c
D-53842 Troisdorf
Fon: +49 / 2241 / 25 65 9 - 0
Fax: +49 / 2241 / 25 65 9 - 11
http://www.microcontrol.net

1. Safety Regulations 1
1.1 General Safety Regulations. 1
1.2 Safety Notice 2

2. Operation of µCAN.4.ci-BOX 3
2.1 Overview 3

3. Project Planning 5
3.1 Module Layout 5
3.2 Operation Area 6
3.3 Maximum System Configuration 7
3.4 Case Dimensions. 9

4. Assembly and Disassembly 11
4.1 Safety Regulations. 11
4.2 General Information 12
4.3 Assembly 13
4.4 Disassembly 14

5. Installation 15
5.1 Potential Basics 15
5.2 EMC Considerations 16
5.2.1 Grounding 17
5.2.2 Shielding of cables 17
5.2.3 CAN Cable 19
5.3 Power Supply. 20
5.4 CAN Bus 22
5.5 Address Selection 23
5.6 Baudrate 24
5.7 Termination 25

6. Digital Signals 27
6.1 Counter mode 28
6.2 Frequency mode 29
6.3 Function principle. 30
6.4 Pin Assignment 31

7. Diagnosis 33
7.1 Network Status 34
7.1.1 Representation of NMT state machine 34
7.1.2 Representation of CAN controller state 35
7.1.3 Combined representation 35
7.2 Module Status 36
7.3 Signal Status 37

8. CANopen Protocol 39
8.1 Introduction 40
8.2 Network Management 41
8.3 SDO Communication 43
8.3.1 SDO Abort Protocol 44
8.4 Object Dictionary 45
8.4.1 Communication Profile 46
8.4.2 Manufacturer specific objects 55
8.4.3 Device Profile 60
8.5 Device monitoring 66
8.5.1 Heartbeat protocol 66
8.5.2 Node guarding 69
8.6 PDO Communication 70
8.6.1 Transmission Modes. 70
8.6.2 Transmit PDO 1 71
8.6.3 Transmit PDO 2 72
8.6.4 Transmit PDO 3 73
8.6.5 Transmit PDO 4 74
8.6.6 Transmit PDO example 75
8.6.7 Synchronisation Message. 76
8.7 Emergency Message 77

9. Technical Data 79

Index 81

1. Safety Regulations

Please read the following chapter in any case, because it contains important information about the secure handling of electrical devices.

1.1 General Safety Regulations

This paragraph gives important information about the conditions of use. It was written for personnel which is qualified and trained on electrical devices.

Qualified and trained personnel are persons who fulfil at least one of the following conditions:

- You know the safety regulations for automated machines and you are familiar with the machine.
- You are the operator for the machine and you have been trained on operation modes. You are familiar with the operation of devices described in this manual.
- You are responsible for setting into operation or service and you are trained on repairing automated machines. In addition you are trained in setting electrical devices into operation, to connect the earthing conductor and to label these devices.

The devices described in this manual may only be used for the mentioned applications. Other devices used in conjunction have to meet the safety regulations and EMI requirements.

To ensure a trouble free and safe operation of the device please take care of proper transport, appropriate storage, proper assembly as well as careful operation and maintenance.

Please take care to observe the actual local safety regulations.

If devices are used in a fixed machine without a mains switch for all phases or fuses, this equipment has to be installed. The fixed machine must be connected to safety earth.

If devices are supplied by mains please take care that the selected input voltage fits to the local mains.

1.2 Safety Notice

If devices are supplied by 24V DC, this voltage has to be isolated from other voltages.

The cables for power supply, signal lines and sensor lines must be installed in a way that the device function is not influenced by EMI.

Devices or machines for industrial automation must be constructed in a manner that an unintentional operation is impossible.

By means of hardware and software safety precautions have to be taken in order to avoid undefined operation of an automated machine in case of a cable fraction.

If automated machines can cause damage of material or personnel in case of a malfunction the system designer has to take care for safety precautions. Possible safety precautions might be a limit switch or locking.

2. Operation of µCAN.4.ci-BOX

2.1 Overview

2

The µCAN.4.ci-BOX is the right solution for counting digital pulses and for frequency measurement via CAN.

Fig. 1: Four channel counter module µCAN.4.ci-BOX

Use of a fieldbus for signal acquisition has the advantage of reduced costs because expensive I/O cards for a PLC or PC can be omitted. In addition, the design of an application is more flexible and modifications are more easily to achieve.

2

The development in automation towards decentralized „intelligent“ systems makes the communication between these components quite important.

Modern automated systems require the possibility to integrate components from different manufacturers. The solution for this problem is a common bus system.

All these requirements are fulfilled by the µCAN.4.ci-BOX module. The µCAN.4.ci-BOX runs on the standard fieldbus CAN.

Typical applications for the µCAN.4.ci-BOX are industrial automation, transportation, food industry and environmental technology.

The µCAN.4.ci-BOX operates with the CAN protocol

CANopen

according to DS-301 (version 4.02). Other protocol stacks are available on request.

space saving and compact

The µCAN.4.ci-BOX is designed for heavy duty applications. The aluminium cast ensures protection class IP66. The compact, space saving case gives the freedom to mount the module in many places.

cost-effective and service friendly

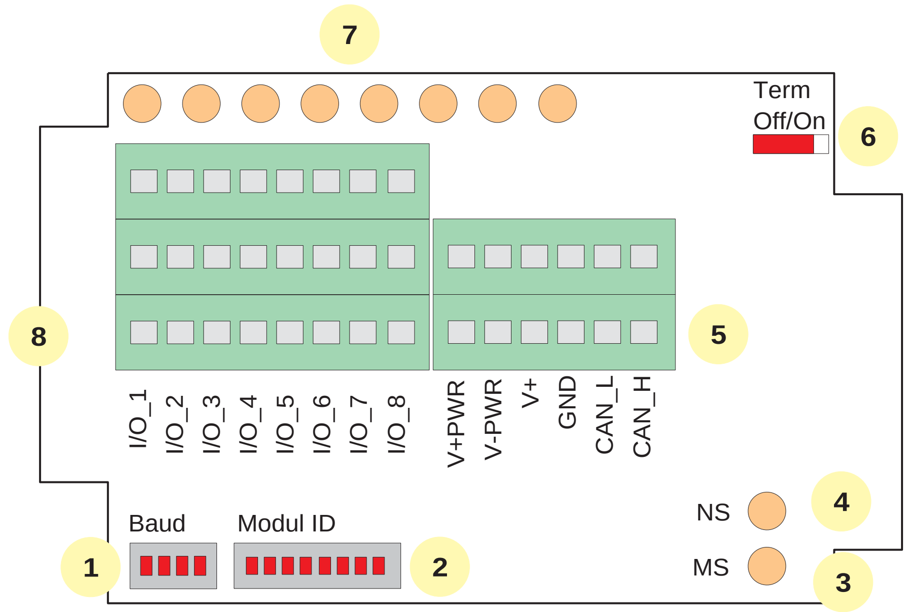
The quick and easy integration of the µCAN.4.ci-BOX in your application reduces the development effort. Costs for material and personnel are reduced. The easy installation makes maintenance and replacement quite simple.

3. Project Planning

The chapter Project Planning contains information which are important for the system engineer when using the µCAN.4.ci-BOX. These information include case dimensions and conditions of use.

3.1 Module Layout

The following figure shows the top view of the µCAN.4.ci-BOX PCB. Use the figure to identify the terminal blocks, LED's and DIP-switches.



1: Baudrate switch
2: Module address / ID
3: Bi-color LED for module status
4: Bi-color LED for network status
5: Terminal block for Power / CAN
6: Switch for CANbus termination
7: Bi-color LED for signal status
8: Terminal block for digital signals

Fig. 2: Top view of the µCAN.4.ci-BOX PCB

3.2 Operation Area

The µCAN.4.ci-BOX is a robust field module for counting digital signals and for frequency measurement via the CANbus. Every module can handle up to 4 input signals. Input configuration (counter/frequency measurement) is done via software. It is not required to setup DIP-switches or jumpers for a configuration change. The module has a power supply range of 8V - 60V DC.

The µCAN.4.ci-BOX needs a four core cable for connection of power supply and CAN bus, in order to reduce the amount of cabling. Special CAN bus cables are available as accessories.

3.3 Maximum System Configuration

For an operational system at least one network manager must be connected to the bus. This network manager might be a PLC or PC equipped with a CAN card. Every µCAN.4.ci-BOX module is an active node.

A CANopen network manager can access **logically** up to 127 CANopen slaves (refer to Fig. 3, "Maximum system configuration"). Every module gets a unique address, which is set up via a DIP switch. The CANbus bus is connected through the µCAN modules. The last module in the network must be terminated by a termination switch (refer to "Termination" on page 25).

3

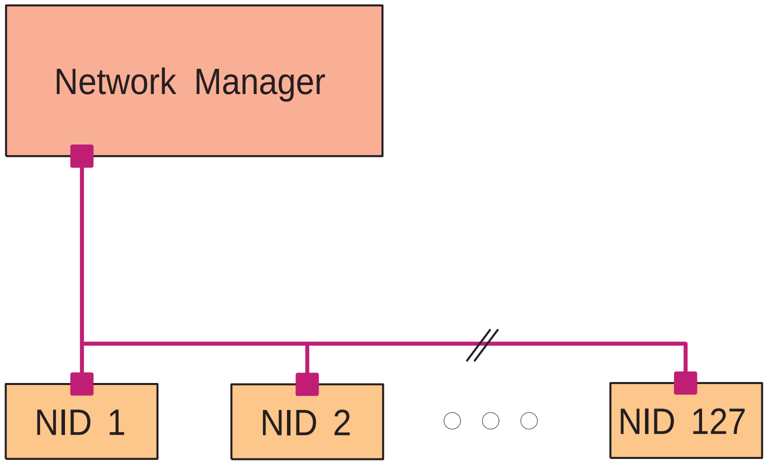


Fig. 3: Maximum system configuration

The maximum cable length depends on the selected baudrate. The following table shows the maximum cable length recommended by the CAN in Automation (http://www.can-cia.org). These distances can be realized with the µCAN.4.ci-BOX.

Baudrate	Cable length
1000 kBit/s	25 m
800 kBit/s	50 m
500 kBit/s	100 m
250 kBit/s	250 m
125 kBit/s	500 m
100 kBit/s	650 m
50 kBit/s	1000 m
20 kBit/s	2500 m

Table 1: Dependence of baudrate from cable length

It is recommended by the CAN in Automation **not to use** the baudrate 100 kBit/s in new CANopen systems.

3.4 Case Dimensions

The case dimensions of the module are given in the following drawing. The high protection class IP66 of the module allows an assembly at places with a harsh environment. It is possible to mount the module inside a switching cabinet as well as direct on a machine. Please check the technical data section for detailled information about maximum environment conditions.

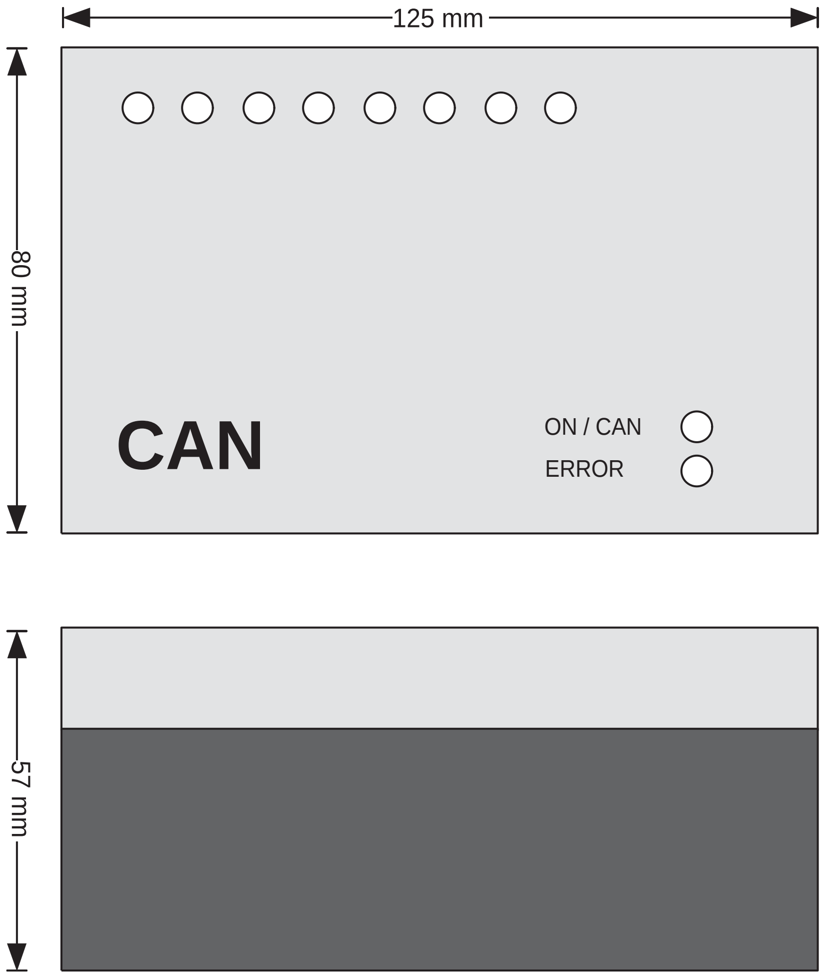


Fig. 4: Case dimensions

4. Assembly and Disassembly

4.1 Safety Regulations

This paragraph gives important information about the conditions of use. It was written for personnel which is qualified and trained on electrical devices.

Qualified and trained personnel are persons who fulfill at least one of the following conditions:

- You know the safety regulations for automated machines and you are familiar with the machine.
- You are the operator for the machine and you have been trained on operation modes. You are familiar with the operation of devices described in this manual.
- You are responsible for setting into operation or service and you are trained on repairing automated machines. In addition you are trained in setting electrcal devices into operation, to connect the earthing conductor and to label these devices.

Terms of Use

The devices described in this manual can only be used for the mentioned applications. Other devices used in conjuction have to meet the safety regulations and EMI requirements.

To ensure a trouble free and safe operation of the device please take care of proper transport, appropriate storage, proper assembly as well as careful operation and maintenance.

4.2 General Information

Assembly

The µCAN module should be assembled on an at least 2 mm thick mounting plate or direct in the plant. The module is fixed with 2 screws of type M4, which are plugged into the bottom part of the case. You find an assembly template in the appendix of this manual.

Power Supply

The µCAN module requires a two core cable for power supply. The cable is inserted from the right side into the case, where the terminals for power supply are located. However it makes sense to use a four core cable in order to run the CAN bus over the same cable.

The non-fused earthed conductor is connected at the terminal outside the case (refer to Fig. 5, “Connection of earthed conductor”). The non-fused earthed conductor may not lead inside the case because of EMI.

The non-fused earthed conductor may not lead inside the µCAN case and may not be connected to a terminal inside the case.

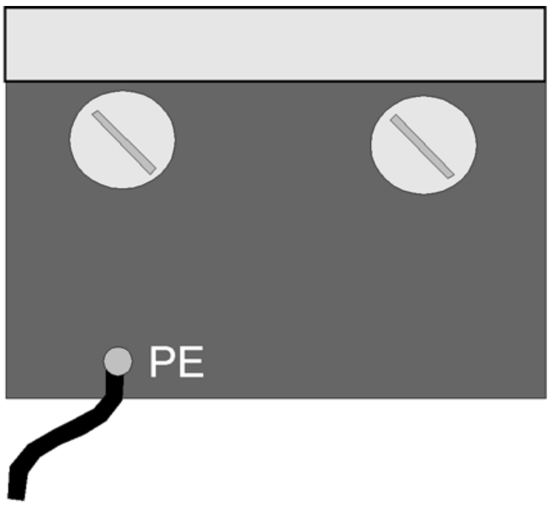


Fig. 5: Connection of earthed conductor

Operation of the µCAN module is only permitted with closed case.

4.3 Assembly

Assembly is performed with help of the template attached to this manual. With the template all necessary bore-holes for screws of type M4 can easily be drilled. If the module is directly fixed to the machine make sure to take the proper drill size for tapping.

When assembling several modules at the same location please make sure to leave some area for the PG screws.

For a quick identification of the modules during operation you may use an adhesive label / tag on top of the module. Please write down the node ID that is set for the module.

Please make sure that the first node and the last node in the CAN network are terminated with a resistor (refer to “Termination” on page 25).

4.4 Disassembly

Please make sure to disconnect the power supply from the device first!

Open the cover of the module and remove all signal lines first. Next remove the cables for CAN bus and power supply from the terminals.

For a safe transport remove the PG screws and close the cover again.

5. Installation

5.1 Potential Basics

The potential environment of a system that is realized with a µCAN.4.ci-BOX module is characterized by following features:

- The CAN bus potential is isolated from the power supply.
- The electronic of the µCAN.4.ci-BOX module is not isolated from the power supply.
- All /O signal lines are not isolated among each other.
- All I/O signals are optically isolated from the CAN bus potential.

5.2 EMC Considerations

EMC (Electromagnetic Compatibility) is the ability of a device to work in a given electromagnetic environment without influencing this environment in a not admissible way.

All µCAN modules fit these requirements and are tested for electromagnetic compatibility in a EMC laboratory. However a EMC plan should be done for the system in order to exclude potential noise sources.

Noise signals can couple in different ways. Depending on that way (guided wave propagation or non-guided wave propagation) and the distance to the noise source the kinds of coupling are differentiated.

DC Coupling

If two electronic circuits use the same conductor we speak of a DC coupling. Noise sources are in that case: starting motors, frequency converters (switching devices in general) and different potentials of cases or of the common power supply.

Inductance Coupling

An inductance coupling is given between two current-carrying conductors. The current in a conductor will cause a magnetic field which induces a voltage in the second conductor (transformer principle). Typical noise sources are transformer, power cables and RF signal cables.

Capacitive Coupling

A capacitive coupling is given between two conductors which have a different potential (principle of a capacitor). Noise sources are in that case: parallel running conductors, static discharge and contactors.

RF Coupling

A RF coupling is given when electromagnetic fields hit a conductor. This conductor works like an antenna for the electromagnetic field and couples the noise into the system. Typical noise sources are spark plugs and electric motors. Also a radio set might be a noise source.

To reduce the impact of noise sources please take care to follow the basic EMC rules.

5.2.1 Grounding

All inactive metal plates must be grounded with low impedance. This method ensures that all elements of the system will have the same potential.

Please take care that the ground potential never carries a dangerous voltage. The grounding must be connected to the safety earth.

The µCAN modules are grounded by the contact which is located under one of the PG screws (see fig. 5, “Connection of earthed conductor”). Additional contacts can be mounted under the PG screws for shielding purposes on demand. The ground potential may not be connected to a terminal inside the case.

5.2.2 Shielding of cables

If noise is coupled to a cable shield it is grounded to safety earth via the metal cover. The cable shields have to be connected to the safety earth with low impedance.

Cable type

For installation of the µCAN module you should only use cable with a shield that covers at least 80% of the core. Do not use cable with a shield made from metallized foil because it can be damaged very easy and has not a good shielding.

Cable connection

In general the cable shield should be grounded on both ends. The cable shield should only be grounded on one end if an attenuation is necessary in the low frequency range. The cable shield can not be grounded on both ends for temperature sensors. The grounding on one end of the cable is necessary if

- there is no contact to the safety earth possible,
- analogue signals with only a few mV or mA are transmitted (e.g. temperature sensors).

The shield of the CAN bus cable may not lead inside the housing of the µCAN.4.ci-BOX. Never connect the shield to the terminals inside the device.

5.2.3 CAN Cable

The CAN cable must meet the requirements of ISO11898. The cable must meet the following specifications:

Parameter	Value
Impedance	108 - 132 Ohm (nom. 120 Ohm)
Specific Resistance	70 mOhm/Meter
Specific Signal Delay	5 ns/Meter

Table 2: Specifications of CAN bus cable

The CAN bus cable is connected to the µCAN.4.ci-BOX module via terminals inside the case. For the pinning of the terminals refer to “CAN Bus” on page 22 of this manual.

Do not confuse the signal lines of the CAN bus, otherwise communication between the modules is impossible.

5.3 Power Supply

The µCAN.4.ci-BOX module is designed for industrial applications. By means of a DC/DC converter the CAN bus of the module is isolated from the supply voltage. The supply voltage must be within the range from 8 V DC to 60 V DC. The input is protected against confusing the poles.

Please make sure not to confuse the poles when connecting the power supply. The positive supply is connected to the terminal **V+**. The positive supply for the output stage is connected to the terminal **V+PWR**.

The negative supply is connected to the terminal **GND**. The negative supply for the output stage is connected to the terminal **V-PWR**.

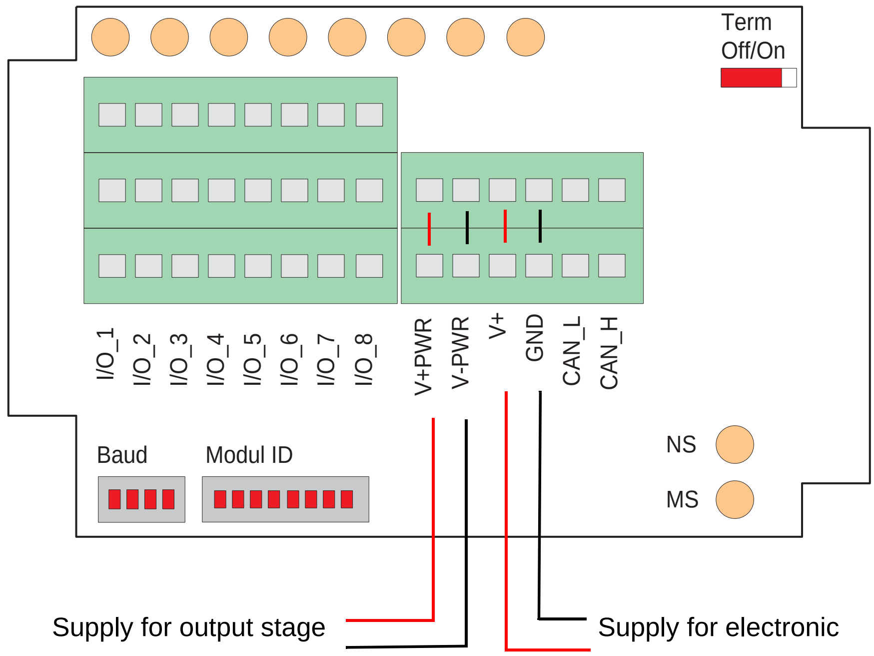


Fig. 6: Connection of power supply

The output stage can be supplied via a separate power source or links have to be made between **V+** and **V+PWR** as well as **GND** and **V-PWR**.

The maximum supply voltage for the **output drivers** is **50V** DC. The maximum supply voltage for the electronic is **60V** DC. Higher voltages will destroy the electronic.

The terminals **GND** and **V-PWR** are not linked internally. The maximum potential difference between these terminals may not exceed 50mV.

Always connect the power supply for the output stage, even if the four digital outputs of the module are not used.

5.4 CAN Bus

The two wires of the CAN bus are connected to the corresponding terminals.

To reduce the influence of EMI please take care that the CAN bus cable does not cross the wires of the signal lines.

The CAN bus line with positive potential must be connected to the terminal **CAN_H**. The CAN bus line with negative potential must be connected to the terminal **CAN_L**.

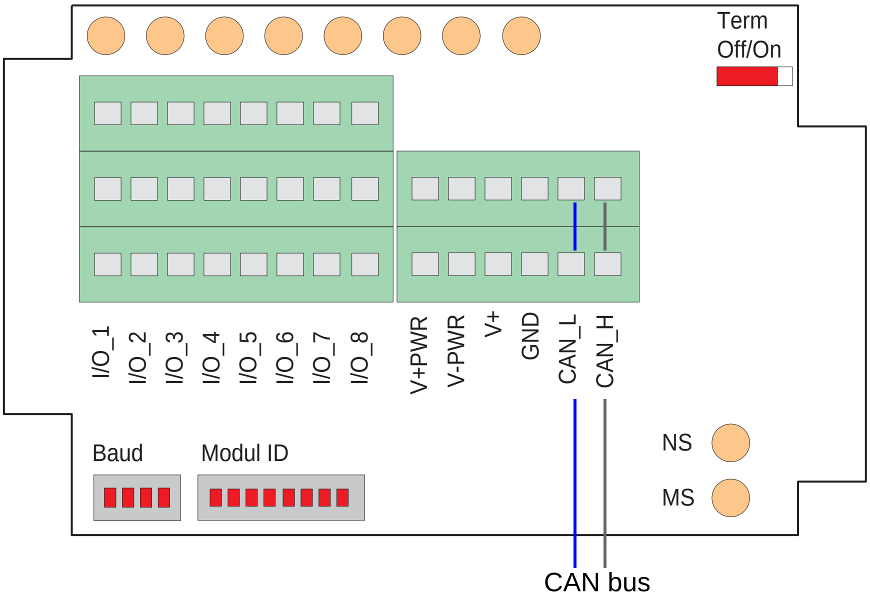
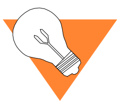


Fig. 7: Connection of CAN bus

Confusing the poles of the CAN bus lines will lead to a communication error on the complete network. The shield of the CAN bus cable may not lead into the housing and may not be connected to a terminal inside the housing. Cable shields have to be connected to the terminals outside the housing.



If you use a Sub-D connector with 9 pins (according to CiA standard), the conductor **CAN_H** is connected to pin 7 and the conductor **CAN_L** is connected to pin 2.

5.5 Address Selection

Address selection of the µCAN.4.ci-BOX module is done via an 8-pin DIP-switch, marked "Modul-ID" which is located at the lower left corner of the PCB. Selection of the address may be done with a small screw driver.

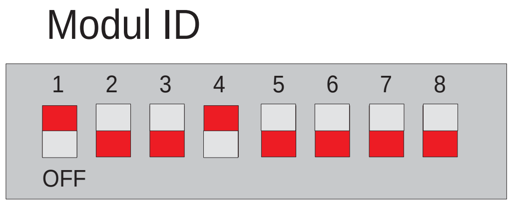


Fig. 8: Setup of module address (here address 9 is shown)

The 8-pin DIP-switch sets the binary code for the module address. The first pin of the switch (marked with ’1’) represents bit 0 of a byte. The last pin of the switch (marked with ’8’) represents bit 7 of a byte.

5

Valid module addresses are within the range from 1..127, resp. 01h..7Fh. Each node within a CANopen network must have a unique module address (node ID). Two nodes with the same node ID are not allowed.

The selected address is read during initialization of the module, after Power-on or Reset. The module runs with the selected node ID until a new node ID is selected and a Reset is performed (via the CAN bus) or the power supply is switched off

Switch 8 must always be in OFF position. Do not put all switches in the OFF position. In these configurations the module will not start to communicate on the bus.

5.6 Baudrate

Baudrate selection of the µCAN.4.ci-BOX module is done via a 4-pin DIP-switch, marked "Baud" which is located at the lower left corner of the PCB. Selection of the baudrate may be done with a small screw driver.

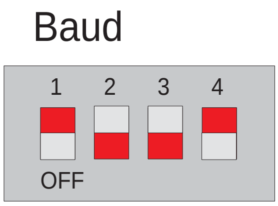


Fig. 9: Setup of baudrate (drawing shows 1 MBit/s)

The 4-pin DIP-switch sets the binary code for the module baudrate. The first pin of the switch (marked with ’1’) represents bit 0 of a byte. The last pin of the switch (marked with ’4’) represents bit 3 of a byte.

The supported baudrates of the µCAN.4.ci-BOX module are given in the following table. The values are recommended by the CiA.

Baudrate	**DIP-switch position**			
	1	**2**	**3**	**4**
Autobaud	0	0	0	0
Autobaud	1	0	0	0
20 kBit/s	0	1	0	0
50 kBit/s	1	1	0	0
100 kBit/s	0	0	1	0
125 kBit/s	1	0	1	0
250 Kbit/s	0	1	1	0
500 kBit/s	1	1	1	0
800 kBit/s	0	0	0	1
1 MBit/s	1	0	0	1

Table 3: Einstellung der Baudrate

The baudrate 10 kBit/s is not supported with the µCAN.4.ci-BOX module. In the position **Autobaud** an automatic detection of the baudrate on the CAN bus is started.

5.7 Termination

The modules at both ends in the CAN network have to be terminated with a resistor of 120 ohms. That means the modules at the end of the bus line are not reflecting back power and the communication can not be disturbed.

For termination of the µCAN.4.ci-BOX the "**Term**" switch must be turned from position "Term Off“ to position "Term On“.

Please make sure that only the devices at both ends of a CAN bus are terminated. In un-powered condition the correct termination value is 60 Ohm between the lines CAN-H and CAN-L.

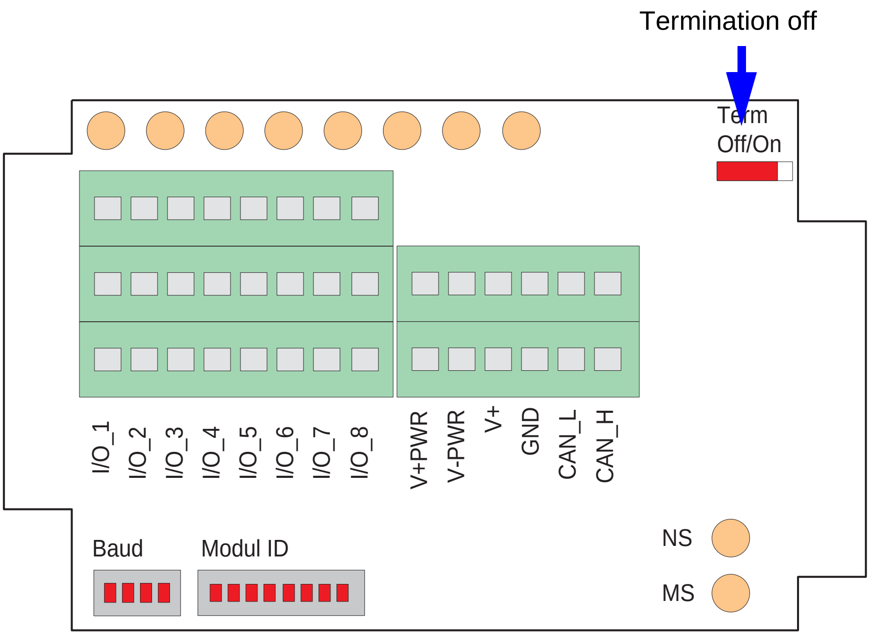


Fig. 10: Termination of CAN bus

Figure 10 shows the termination of the module in "off" position.

5

6. Digital Signals

The µCAN.4.ci-BOX has eight digital I/O terminals. The terminals are labeled with "**I/O_1**" to "**I/O_8**" from left to right on the PCB.

Please keep the basics of EMI rules in mind when planning the wiring. Only proper wiring and EMI precautions make sure that the module runs without trouble.

The four independent input channels are assigned to the terminals "**I/O_1**" to "**I/O_4**".

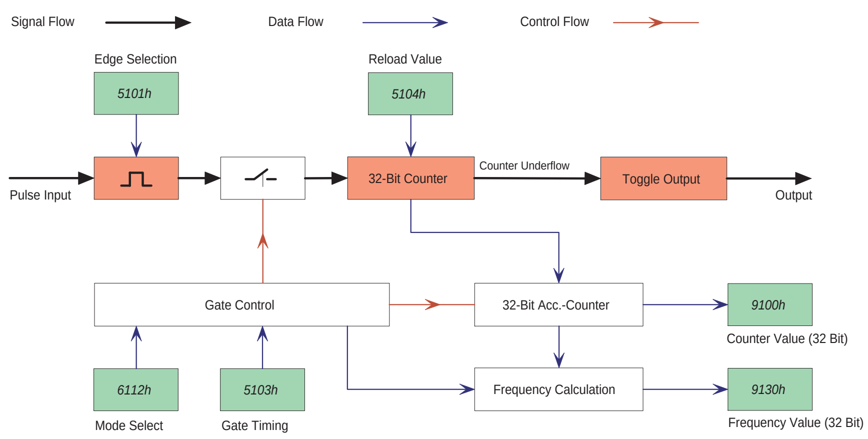


Fig. 11: Signal flow of the counter input

The input signal is feeded via the "Gate Control" unit to a 32-bit counter. The counter overflow value is responsible for changing the digital output assigned to the counter channel. The overflow value as well as the actual counter value can be accessed via the CANopen interface.

The µCAN.4.ci-BOX module can be used in two modes:

- **Counter Mode**: Up-/Down-counter
- **Frequency Mode**: Frequency measurement

Mode selection is carried out via the CANopen object "Operation mode" (refer to “Operation mode” on page 62)

6.1 Counter mode

In counter mode digital input pulses are added to (or substracted from) an accumulator. Upon reaching an overflow value the counter value is reset and the state of the digital output assigned to the counter changes.

Up/Down-Counter

Mode selection is carried out via the CANopen object "Operation mode" (refer to “Operation mode” on page 62)

In counter mode the frequency of the input signal is calculated via the number of pulses received during a gate time. Setup of this parameter is done via the CANopen object "Gate Time". The accuracy of the frequency value depends on the selected gate time and is calculated by the following formular:

$$\Delta f = \frac{1}{T_{Gate}}$$

A small gate time leads to a high update rate of the measured frequency, but also increases the measurement error. Hence using a bigger gate time reduces the measurement error, but also reduces the update rate.

Example: Gate time = 100ms

$$\Delta f = \frac{1}{100ms} = 10Hz$$

Using a gate time of 100ms leads to a total measurement error of ±10Hz.

In counter mode the calculated fequency value is transmitted in fixed-comma representation (32-bit) with one digit after the comma. The digit after the comma is always set to 0, independet from the selected gate time.

6.2 Frequency mode

In frequency mode the value is calculated via a multiple periodic time measurement. The maximum measurement period is determined via the CANopen object "Gate Time" (refer to “Gate Timing” on page 59). The maximum measurement period also determines the lower cut-off-frequency:

$$f_{cutoff} = \frac{1}{T_{Gate}}$$

Small values for the maximum measurement period increase the lower cut-off-frequency. A frequency value of 0 Hz (on signal present) can only be shown after the maximum measurement period elapses.

Example: Maximum measurement period = 100ms

$$\Delta f = \frac{1}{100ms} = 10Hz$$

Using a maximum measurement period of 100 ms limits the lower cut-off frequency to 10 Hz.

In frequency mode the calculated fequency value is transmitted in fixed-comma representation (32-bit) with one digit after the comma. The used measuring method allows a fast and accurate determination of the unkown input signal.

In frequency mode the incoming digital pulses are not counted. The value of the CANopen object 9100h (field value) is undefined.

6.3 Function principle

Configuration of each terminal (input or output) is done via firmware. The Power-MOSFET is always turned off when the terminal is used as input (pulse counter). The input voltage at the terminal is compared with a reference voltage, which can be configured via software (CANopen parameters 5FF0h to 5FF2h). Hence the trigger level of the input signal can be adjusted.

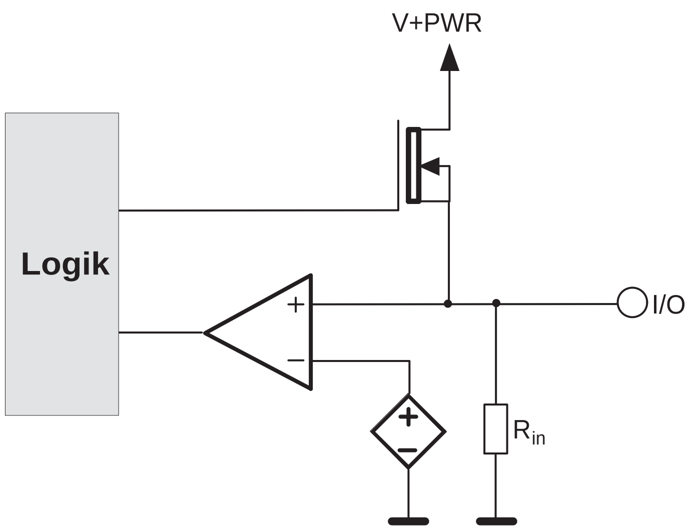


Fig. 12: Schematic digital I/O terminal

If the terminal is used as digital output, the Power-MOSFET is driven via the control logic. The logic block detects over current, short current and thermal overload.

Parameter	Value
V+PWR	10 .. 50 V
Impedance R_{in}	24,2 kOhm
I_{out}	1,4 A maximum
Trigger Level	adjustable via software

Table 4: Electrical Parameters

6.4 Pin Assignment

The terminal block of the µCAN.4.ci-BOX is designed to connect digital sensors with 3 wires. The sensor gets the positive supply voltage (**V+PWR)** from terminal row B. The ground potential is located in terminal row C (**V-PWR**).

The binary control lines are connected to row A. The state of each line is displayed by means of bi-color LEDs.

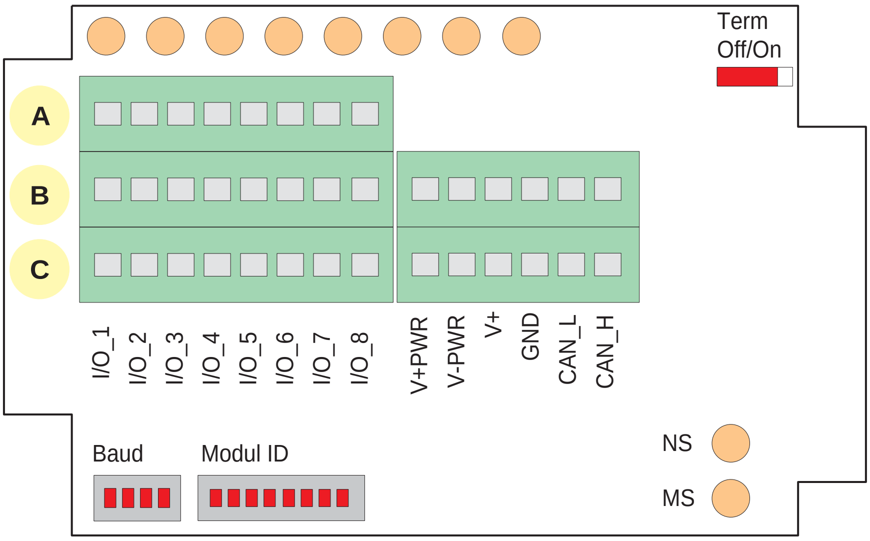


Fig. 13: Terminals for signal lines

Pin	Function
I/O_1 / A	Input 1: counter / frequency measurement
I/O_2 / A	Input 2: counter / frequency measurement
I/O_3 / A	Input 3: counter / frequency measurement
I/O_4 / A	Input 4: counter / frequency measurement
I/O_5 / A	Output 1: assigned to counter 1
I/O_6 / A	Output 2: assigned to counter 2
I/O_7 / A	Output 3: assigned to counter 3
I/O_8 / A	Output 4: assigned to counter 4

Table 5: Pin assignment

All signal lines may only be connected in power off state in order to prevent a damage of the electronic.

7. Diagnosis

All modules of the µCAN family have LEDs to display the operating state and to signalize an error state.

The µCAN.4.ci-BOX has two bi-color LEDs (green/red) labeled with "NS" (Network Status) and "MS" (Module Status) on the PCB.

On the case cover the LEDs are marked as **ON/CAN** for the network status and **ERROR** for the module status.

The state of the digital I/O terminals is displayed by eight bi-color LEDs (position 7 in the figure below).

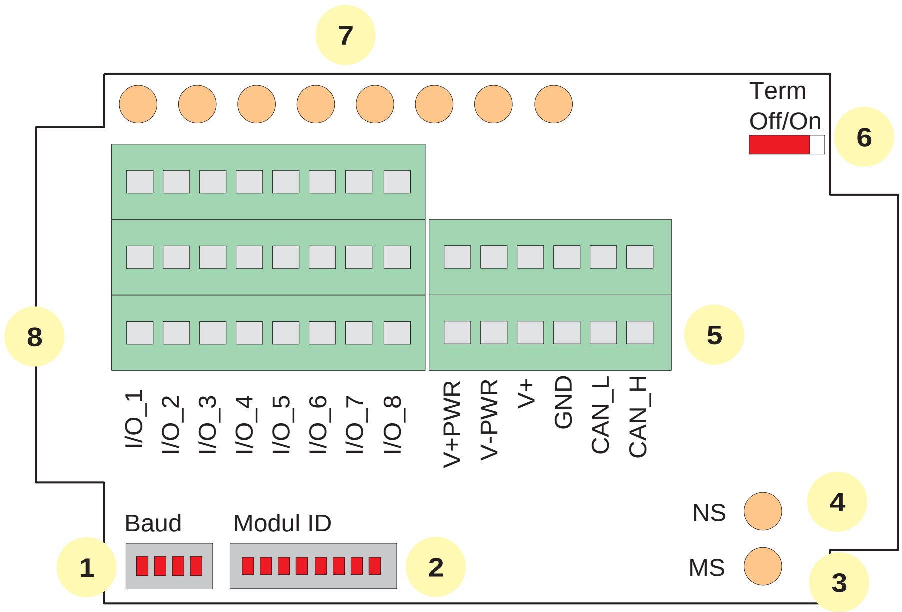


Fig. 14: Position of LEDs on the module

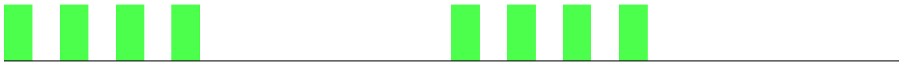
In normal operation all LEDs should have a green or orange color. A red steady light or a red blinking of a LED indicates an error condition.

7.1 Network Status

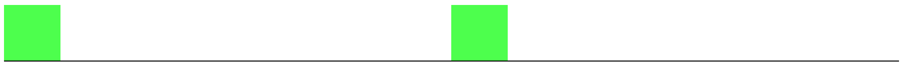
The bi-color LED labeled with "NS" (on the case cover marked as ON/CAN) shows the status of the CANopen state machine as well as the error state of the CAN controller.

7.1.1 Representation of NMT state machine

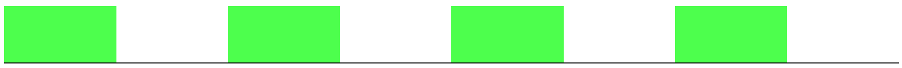
The green light of the NS-LED represents the status of the CAN-open network management state machine.



Initialisation (Autobaud detection)



NMT state: device is "stopped"

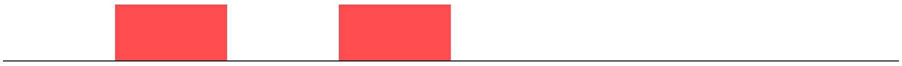


NMT state: device is "pre-operational"

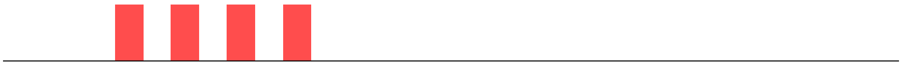
NMT statue: device is "operational"

7.1.2 Representation of CAN controller state

The red light of the NS-LED represents the error state of the CAN controller. The red light is off during error-free condition.



CAN status: controller in "warning" state

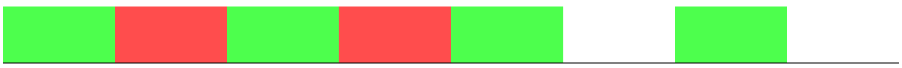


CAN status: controller in "error passive" state

CAN status: controller in "bus-off" state

7.1.3 Combined representation

The combination of the green light and the red light of the NS-LED allows the representation of both - the NMT status and the CAN controller status. The following pictures give an example of combined representation.



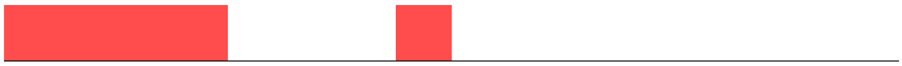
Device in "pre-operational" state, CAN controller in "warning" state

Device in "operational" state, controller in "error passive" state

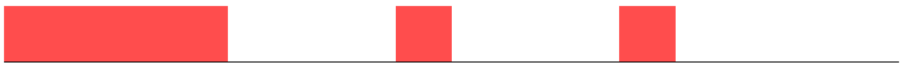
7.2 Module Status

The bi-color LED labeled with "MS" (on the case cover marked as ERROR) shows the status of the device hardware.

Module status: function/power OK



Module status: wrong baudrate setup



Module status: wrong address setup

7.3 Signal Status

The eight LEDs above the terminal block indicate the state of the digital signal present on each input / output.

Signal LED is	To indicate
Green	Digital high-signal at terminal
Orange	Terminal configured as output, output has been switched on
Red	Terminal configured as output, short-circuit on output driver

Table 6: LED for signal status

7

8. CANopen Protocol

This chapter provides detailed information on how to connect the modules of the µCAN-series to a CANopen manager. A CAN-open manager might be a PLC, a PC with a CAN interface or any other CAN device with NMT functionality.

For more information about CANopen manager please refer to the supplied manuals of your CANopen master device.

This documentation provides the actual implemented functions and services of the µCAN.4.ci-BOX.

8.1 Introduction

The identifiers of the µCAN.4.ci-BOX are set up according to the **Pre-defined Connection Set**, which is described in detail in the CANopen communication profile DS-301. The following table gives an overview of the supported services.

Object	COB-ID (dec.)	COB-ID (hex)
Network Management	0	0x000
SYNC	128	0x080
EMERGENCY	129 - 255	0x081 - 0x0FF
PDO 1 (Transmit)	385 - 511	0x181 - 0x1FF
PDO 2 (Transmit)	641 - 767	0x281 - 0x2FF
PDO 3 (Transmit)	897 - 1023	0x381 - 0x3FF
PDO 4 (Transmit)	1153 - 1279	0x481 - 0x4FF
SDO (Transmit)	1409 - 1535	0x581 - 0x5FF
SDO (Receive)	1537 - 1663	0x601 - 0x67F
Heartbeat / Boot-up	1793 - 1919	0x701 - 0x77F

Table 7: Identifier values according to the Pre-defined Connection Set

The direction (Transmit / Receive) has to be seen from the devices point of view.

8.2 Network Management

By means of the Network Management (**NMT**) messages the state of a CANopen node can be changed (Stopped / Pre-Operational / Operational).

Start Node

Start Node

ID	DLC	B0	B1
0	2	01h	Node

Node = module address, 0 = all modules

By transmitting the "Start Node" command the CAN-node will be set into Operational mode. This means that the node can handle PDO-communication.

Stop Node

Stop Node

ID	DLC	B0	B1
0	2	02h	Node

Node = module address, 0 = all modules

By transmitting the "Stop Node" command the CAN-node will be set into Stopped mode. This means that the node can not handle any services except NMT commands.

Pre-Operational

Enter Pre-Operational

ID	DLC	B0	B1
0	2	80h	Node

Node = module address, 0 = all modules

By transmitting the „Enter Pre-Operational“ command the CAN-node will be set into Pre-Operational mode. In this state the node can not handle PDO messages.

Reset Node

Reset Node

ID	DLC	B0	B1
0	2	81h	Node

Node = module address, 0 = all modules

By transmitting the „Reset Node“ command the CAN-node will issue a reset operation. After reset the node will send a "Boot-up message" (refer to “Heartbeat protocol” on page 66) and enter the Pre-operational state automatically.

8

8.3 SDO Communication

All parameters of the devices (organized in an object dictionary) are accessed via the SDO service (Service Data Object). A SDO message has the following contents:

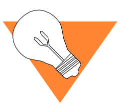
ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
	8	CMD	Index		Sub-Index	Data			

For calculation of the SDO message identifier please refer to “Introduction” on page 40.

The "Command Byte" (**CMD**) is defined according to the following table.

SDO client (CANopen master)	SDO server (CANopen slave)	Function
22_h	60_h	write, size not specified
23_h	60_h	write, size = 4 bytes
27_h	60_h	write, size = 3 bytes
$2B_h$	60_h	write, size = 2 bytes
$2F_h$	60_h	write, size = 1 byte
40_h	42_h	read, size not specified
40_h	43_h	read, size = 4 bytes
40_h	47_h	read, size = 3 bytes
40_h	$4B_h$	read, size = 2 bytes
40_h	$4F_h$	read, size = 1 byte

Table 8: Command byte for für SDO Expedited Botschaft



The byte order for the fields "**Index**" and "**Data**" is least significant byte first (Intel format).

The minimum time delay between two succeeding SDO messages must be greater than 20ms. Faster communication might lead to an unpredictible device status.

8.3.1 SDO Abort Protocol

The SDO abort protocol is used to signalize a fault when accessing an object. This SDO abort protocol has the following format:

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
	8	80h	Index		Sub-Index	Abort Code			

The identifier as well as the index and sub-index correspond to the SDO request.

The abort code may have the following values:

Abort code	Description
0504 0001h	Client / Server command specifier not valid / unknown
0601 0000h	Unsupported access to an object
0601 0001h	Attempt to read a "write-only" object
0601 0002h	Attempt to write a "read-only" object
0602 0000h	Object does not exist in the object dictionary
0609 0011h	Sub-index does not exist

Table 9: SDO abort codes

8

8.4 Object Dictionary

This chapter describes the implemented objects for the module µCAN.4.ci-BOX. For additional information please refer to the CANopen communication profile DS-301 and the device profile DS-404.

EDS

The implemented objects of the module µCAN.4.ci-BOX are listed in an "Electronic Data Sheet" (EDS). The EDS file can be downloaded from the MicroControl homepage.

8.4.1 Communication Profile

The module µCAN.4.ci-BOX supports the following objects from the communication profile DS-301:

Index	Name
1000h	Device Profile
1001h	Error Register
1002h	Manufacturer Status
1003h	Predefined Error-Register
1005h	COB-ID SYNC-Message
1008h	Manufacturer Device Name
1009h	Manufacturer Hardware Version
100Ah	Manufacturer Software Version
100Ch	Guard Time
100Dh	Life Time Factor
1010h	Store Parameters
1011h	Restore Default Parameters
1014h	COB-ID Emergency-Message
1016h	Heartbeat Consumer Time
1017h	Heartbeat Producer Time
1018h	Identity Object
1029h	Error Behaviour
1800h	1st Transmit PDO Parameters
1801h	2nd Transmit PDO Parameters
1802h	3rd Transmit PDO Parameters
1803h	4th Transmit PDO Parameters
1A00h	1st Transmit PDO Mapping
1A01h	2nd Transmit PDO Mapping
1A02h	3rd Transmit PDO Mapping

Table 10: Supported objects of the CANopen communication profile

Index	Name
1A03h	4th Transmit PDO Mapping
1F80h	NMT Startup

Table 10: Supported objects of the CANopen communication profile

Device Profile

Index 1000h

The object at index 1000h describes the type of device and its functionality.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned32	ro	Device Profile	0002 0194h

The object is read-only. Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Example: read parameter, module ID = 2, index = 1000h

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
602h	8	40h	00h	10h	00h	00h	00h	00h	00h

As response the µCAN.4.ci-BOX will send:

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
582h	8	42	00	01h	00	91h	01h	03h	00

Byte 5 + Byte 6 = 0194h = 404_d (Device Profile Number)
Byte 7 + Byte 8 = 0002h = 2 (Additional Information)

Error Register

Index 1001h

The object at index 1001h is an error register for the device.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Error Register	00h

The object is read-only. Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Example: read parameter, module ID = 2, Index = 1001h

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
602h	8	40h	01h	10h	00	00	00	00	00

As response the module will return its error register value. The following error types are supported:

Generic Error

Bit 0 is set to '1'. The generic error is set due to hardware faults.

Communication Error

Bit 4 is set to '1'. The communication error is set due to faults on the CAN bus.

The object is read-only. Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Pre-defined Error Field

Index 1003

The object at index 1003h holds the errors that have occured on the device. The object stores a maximum of 10 error conditions.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	rw	Number of errors	00h
1 .. 10	Unsigned32	ro	Standard error field	0000 0000h

The object supports the sub-indices 0 to 10. An access to other sub-indices will lead to an error message. Writing to sub-index 0 will clear the error history.

Example: read parameter, module ID = 2, Index = 1003h

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
602h	8	40h	03h	10h	05h	00h	00h	00h	00h

As response the module will return the error value at position 5 in the history.

Manufacturer Device Name

Index 1008

The object at index 1008h contains the manufacturer device name.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Visible String	ro	Device name	mCAN.4.ci-BOX

The object is read-only. Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Manufacturer Hardware Version

Index 1009h

The object at index 1009h contains the manufacturer hardware version.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Visible String	ro	Hardware version	-

The object is read-only. Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Manufaturer Software Version

Index 100Ah

The object at index 100Ah contains the manufacturer software version.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Visible String	ro	Software version	-

The object is read-only. Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Store Parameters

Index 1010h

The object at index 1010h supports the saving of parameters in a non volatile memory.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Number of objects	04h
1	Unsigned32	rw	Save all parameters	0000 0001h
2	Unsigned32	rw	Save communication	0000 0001h
3	Unsigned32	rw	Save application	0000 0001h
4	Unsigned32	rw	Save manufacturer	0000 0001h

In order to avoid storage of parameters by mistake, storage is only executed when a specific signature is written to the appropriate sub-index. The signature is "save".

Example: save all parameters, module ID = 2, index = 1010h

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
602h	8	23h	10h	10h	01h	73h	61h	76h	65h

As response the µCAN.4.ci-BOX will send:

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
582h	8	60h	10h	10h	01h	00h	00h	00h	00h

Parameters are stored in a non-volatile memory after reception of the store request message.

Restore Default Parameters

Index 1011h

The object at index 1011h supports the restore operation of default parameters.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Number of objects	04h
1	Unsigned32	rw	Restore all param.	0000 0001h
2	Unsigned32	rw	Restore commun.	0000 0001h
3	Unsigned32	rw	Restore application	0000 0001h
4	Unsigned32	rw	Restore manufacturer	0000 0001h

In order to avoid the restoring of default parameters by mistake, restoring is only executed when a specific signature is written to the appropriate sub-index. The signature is "load".

Example: restore all parameters, module ID = 2, Index = 1011h

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
602h	8	23h	11h	10h	01h	6Ch	6Fh	61h	64h

As response the µCAN.4.ci-BOX will send:

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
582h	8	60h	11h	10h	01h	00h	00h	00h	00h

COB-ID for emergency message

Index 1014h

The object at index 1014h defines the identifier value for the emergency message.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned32	rw	COB-ID EMCY	80h + Node-ID

Only sub-index 0 is supported. An access to other sub-indices will lead to an error message. The default value for the identifier is 80h + selected node ID.

Identity Object

Index 1018h

The object at index 1018h holds the identity object (LSS address) of device.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	4
1	Unsigned32	ro	Vendor ID	0000 000Eh
2	Unsigned32	ro	Product Code	0013 A745h
3	Unsigned32	ro	Revision Number	0298 00xxh
4	Unsigned32	ro	Serial Number	-

The object is read-only. Only sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message.

Vendor ID

The "Vendor ID" contains a unique value allocated to each manufacturer. The numbers are managed by the CAN in Automation (http://www.can-cia.org/canopen/vendor-id/list/index.html).

Product Code

The "Product Code" identifies a specific product, i.e. it is unique to the order code of devices from MicroControl.

Revision Number

The "Revision Number" consists of a major revision number (upper word) and a minor revision number (lower word). The major revision number identifies a specific CANopen behaviour. The minor revision number identifies different versions with the same CANopen behaviour.

Serial Number

The "Serial Number" contains the serial number of a device.

Error behaviour

Index 1029h

If a serious CANopen device failure is detected in NMT state Operational, the CANopen device will enter by default autonomously the NMT state Pre-operational. The object 1029h allows the device to enter alternatively the NMT state Stopped or remain in the current NMT state.

Sub-Index	Data Type	Acc.	Name	Defaul Value
0	Unsigned8	ro	number of entries	01h
1	Unsigned8	rw	Communication error	00h

The following codes are possible:

Value	Description
00h	Change to NMT state Pre-operational
01h	No change of the NMT state
02h	Change to NMT state Stopped

Table 11: Codes for error behaviour setup

The device detects the following communication errors:

- Bus-off conditions of the CAN interface
- Life guarding event with the state "occurred" and the reason "time out"
- Heartbeat event with state "occurred" and the reason "time out"

8

8.4.2 Manufacturer specific objects

The module µCAN.4.ci-BOX supports the following objects from the manufaturer profile area:

Index	Name
5020h	Device supply voltage
5101h	Edge Selection (in Counter Mode only)
5103h	Gate Timing
5104h	Reload Value (in Counter Mode only)
5FF0h	Input Level, absolut
5FF1h	Input Level, relative
5FF2h	Input Level Selection

Table 12: Manufacturer specific objects

The device profile DS-404 does not support the functionality of a digital counter / frequency measuring device in a sufficient way. That causes the necessity to place several objects in the manufacturer specific area.

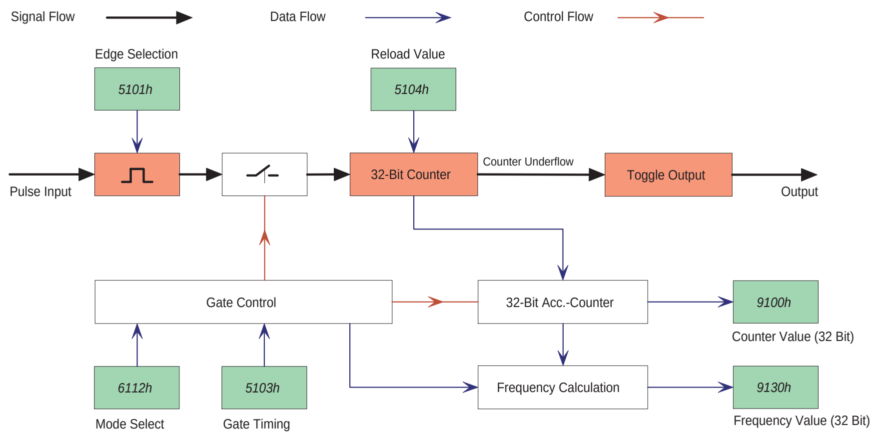


Table 13: Objects für counter / frequency measurement

Device supply voltage

Index 5020h

The supply voltage of the device can be gathered via index 5020h. The supply voltage is shown in [V] with one digit after the comma.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned16	ro	Device Supply Volt.	-

The object is read-only. Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Edge Selection

Index 5101h

The object at index 5101h defines which signal edge causes the counter to increase or decrease. The counter direction (up/down) is defined via the objekt 6112h.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned8	rw	Edge Sel. Counter 1	00h
2	Unsigned8	rw	Edge Sel. Counter 2	00h
3	Unsigned8	rw	Edge Sel. Counter 3	00h
4	Unsigned8	rw	Edge Sel. Counter 4	00h

The object allows read-write access. The sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message.

Possible values for signal edge selection are:

Value	Function
0	Counter is changed on positive edge (default)
1	Counter is changed on negative edge
2	Counter is changed on positive **and** negative edge

Table 14:

Gate Timing

Index 5103h

The object at index 5103h defines the gate time value. The time is defined in milli-seconds.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned16	rw	Gate Timing 1	1000
2	Unsigned16	rw	Gate Timing 2	1000
3	Unsigned16	rw	Gate Timing 3	1000
4	Unsigned16	rw	Gate Timing 4	1000

The object allows read-write access. The sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message.

This parameter is always used for frequency calculation, independent from the operation mode (object 6112h). The frequency value is also calculated when the device is in counter mode.

Reload Value

Index 5104h

The object at index 5104h defines the reload value of the counter. If the counter values reaches the reload value, the digital output assigned to the counter will toggle and the counter value is reset.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned32	rw	Reload Value 1	FFFF FFFFh
2	Unsigned32	rw	Reload Value 2	FFFF FFFFh
3	Unsigned32	rw	Reload Value 3	FFFF FFFFh
4	Unsigned32	rw	Reload Value 4	FFFF FFFFh

The object allows read-write access. The sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message.

Example: Set reload value for counter 2 to 1000

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
601h	8	23h	04h	51h	02h	E8h	03h	00h	00h

As response the µCAN.4.ci-BOX will send:

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
581h	8	60h	04h	51h	02h	00h	00h	00h	00h

The maximum switching frequency on the digital output may not exceed 1000 Hz. Thus, the minimum reload value must be adjusted to the maximum input frequency. The minimum reload value can be calculated as:

Reload $Value_{min}$ = Input $frequency_{max}$ / 1000

Input Level, absolute

Index 5FF0h

The object at index 5FF0h defines the absolute value for the input signal trigger level..

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned16	rw	Input Level, absolute	25

The voltage is set in multiples of 100 mV.

Example: Set trigger level to 4,5V

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
601h	8	2Bh	F0h	5Fh	00h	2Dh	00h	00h	00h

As response the µCAN.4.ci-BOX will send:

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
581h	8	60h	F0h	5Fh	00h	00h	00h	00h	00h

The object allows read-write access. Only sub-indices 0 is supported. An access to other sub-indices will lead to an error message.

Input Level, relative

Index 5FF1h

The object at index 5FF1h defines the relative value for the input signal trigger level..

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	rw	Input Level, relative	50

The relative value can be set in the range from 0% to 80%. It is referenced to the module supply voltage.

The object allows read-write access. Only sub-indices 0 is supported. An access to other sub-indices will lead to an error message.

Input Level Selection

Index 5FF2h

The object at index 5FF2h selects the type of the input signal trigger level (absoulte or relative).

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	rw	Input Level Selection	0

The object allows read-write access. Only sub-indices 0 is supported. An access to other sub-indices will lead to an error message.

Possible values for input level selection are:

Value	Configuration
0	absolute input level (object 5FF0h)
1	relative input level (object 5FF1h)

Table 15: Configuration options for input level

8.4.3 Device Profile

This section describes all device profile specific objects (DS-404) of the µCAN.4.ci-BOX:

Index	Name
6110h	Sensor type
6112h	Operating Mode
6131h	Physcial Unit
6132h	Decimal Degits
9100h	Field Value (Counter)
9130h	Process Value (Frequency)

Table 16: Supported objects of the device profile DS-404

Sensor Type

Index 6110h

The object at index 6110h defines the sensor type. The value of 3Ch (60d) is assigned to frequency measurement devices.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned16	ro	Sensor Type Input 1	003Ch
2	Unsigned16	ro	Sensor Type Input 2	003Ch
3	Unsigned16	ro	Sensor Type Input 3	003Ch
4	Unsigned16	ro	Sensor Type Input 4	003Ch

The object allows read-only access. Only sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message.
.

Example: Read sensor type, module addresse 1

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
601h	8	40h	10h	61h	01h	00h	00h	00h	00h

As response the µCAN.4.ci-BOX will send:

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
581h	8	4Bh	10h	61h	01h	3Ch	00h	00h	00h

Operation mode

Index 6112h

The object at index 6112h defines the operating mode for each of the four input channels.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned8	rw	Operating Mode Inp. 1	0Ah
2	Unsigned8	rw	Operating Mode Inp. 2	0Ah
3	Unsigned8	rw	Operating Mode Inp. 3	0Ah
4	Unsigned8	rw	Operating Mode Inp. 4	0Ah

The object allows read-write access. Only sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message.

Value	Function
0 (00h)	Input is turned off
10 (0Ah)	Counter mode, counter is incremented (count up)
11 (0Bh)	Counter mode, counter is decremented (count down)
20 (14h)	Frequency measurement

Table 17: Operation mode setup

Example: Select frequency measurement on input 1,

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
601h	8	2Fh	12h	61h	01h	14h	00h	00h	00h

As response the µCAN.4.ci-BOX will send:

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
581h	8	60h	12h	61h	01h	00h	00h	00h	00h

Physical Unit

Index 6131h

The object at index 6131h defines the physical unit of the process value.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned32	ro	Physical Unit Input 1	0020 0000h
2	Unsigned32	ro	Physical Unit Input 2	0020 0000h
3	Unsigned32	ro	Physical Unit Input 3	0020 0000h
4	Unsigned32	ro	Physical Unit Input 4	0020 0000h

The object allows read-only access. Only sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message. The physical unit Hertz [Hz] is represented by the code 0x00200000.

Decimal Digits

Index 6132h

The object at index 6132h defines the number of digits after the comma (fixed-point representation).

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned8	ro	Decimal Digits Input 1	01h
2	Unsigned8	ro	Decimal Digits Input 2	01h
3	Unsigned8	ro	Decimal Digits Input 3	01h
4	Unsigned8	ro	Decimal Digits Input 4	01h

8

The object allows read-only access. Only sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message. The number of decimal digits after the comma is always 1, i.e. the process value has a resolution of 0.1 Hz.

Field Value

Index 9100h

The object at index 9100h holds the 32-bit counter values (field values) of each input.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned32	ro	Counter Input 1	-
2	Unsigned32	ro	Counter Input 2	-
3	Unsigned32	ro	Counter Input 3	-
4	Unsigned32	ro	Counter Input 4	-

If a channel is configured in frequency mode (refer to “Operation mode” on page 62), the counter can not be used. It will show undefined values.

The object allows read-only access. Only sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message.

Example: Read counter 2 value, module address 3

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
603h	8	40h	00h	91h	02h	00h	00h	00h	00h

As response the µCAN.4.ci-BOX will send:

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
583h	8	43h	00h	91h	02h	E8h	03h	00h	00h

Counter 2 holds the value 0000 03E8h (1000dez).

Process Value

Index 9130h

The object at index 9130h holds the frequency values (process value) for each input.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	04h
1	Unsigned32	ro	Frequency Input 1	-
2	Unsigned32	ro	Frequency Input 2	-
3	Unsigned32	ro	Frequency Input 3	-
4	Unsigned32	ro	Frequency Input 4	-

The object allows read-only access. Only sub-indices 0 to 4 are supported. An access to other sub-indices will lead to an error message. The frequecy value is given in fixed-comma representation with a resolution of 0.1 Hz.

If a channel is configured in counter mode (refer to “Operation mode” on page 62), the last digit will always be zero.

Example: Read frequency value from input 2, module addresse 3

ID	***DLC***	***B0***	***B1***	***B2***	***B3***	***B4***	***B5***	***B6***	***B7***
603h	8	40h	30h	91h	02h	00h	00h	00h	00h

As response the µCAN.4.ci-BOX will send:

ID	***DLC***	***B0***	***B1***	***B2***	***B3***	***B4***	***B5***	***B6***	***B7***
583h	8	43h	30h	91h	02h	4Ch	2Ch	00h	00h

The frequency value at input 2 is 1,134.0 Hz (one digit after comma, 0000 2C4Ch = 11.340).

8.5 Device monitoring

For device monitoring CANopen provides two mechanisms (protocols):

- heartbeat
- node guarding

It is recommended by the CAN in Automation **not to use** node guarding for device monitoring (CiA AN802 V1.0: CANopen statement on the use of RTR messages).

8.5.1 Heartbeat protocol

The heartbeat protocol is used in order to survey other CANopen nodes in the network and retrieve their network state.

heartbeat ID

The identifier for the heartbeat protocol is set to 700h + module address. The identifier can not be changed. The message repetition time (called "heartbeat producer time") is configured with object 1017h.

The heartbeat protocol transmits one byte of data, which represents the network state.

Network State	Code (dec.)	Code (hex)
Bootup	0	00h
Stopped	4	04h
Operational	5	05h
Pre-Operational	127	7Fh

Table 18: Status Information for Heartbeat

After Power-on / Reset the module will send the "Boot-up message" to signal that it finished the initialization sequence.

Example: Power-on of module with address 2

ID	***DLC***	***B0***
702h	1	00h

Consumer heartbeat time

Index 1016h

The object at index 1016h defines the consumer heartbeat time.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Number of objects	2
1	Unsigned32	rw	Heartbeat Cons. 1	0000 0000h
2	Unsigned32	rw	Heartbeat Cons. 2	0000 0000h

The µCAN.4.ci-BOX can monitor the presence of two other devices (heartbeat producer) in the network. If a heartbeat producer message is not received within an adjustable period, an emergency message with value 8130h (life guard error or heartbeat error) is transmitted. The 32-bit value of the object defines heartbeat time and the producers node address.

Bit 31 ... 24	Bit 23 ... 16	Bit 15 ... 0
reserved (00h)	producer node address	heartbeat producer time

If the heartbeat time is 0 or the node-ID is 0 or greater than 127 the corresponding object entry is not used. The heartbeat time is given in multiples of 1 millisecond. Monitoring starts after reception of the first heartbeat.

Producer heartbeat time

Index 1017h

The object at index 1017h defines the cycle time of the heartbeat. The producer heartbeat time is 0 if it is not used. The time is a multiple of 1ms.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned16	rw	Producer Time	0000h

The object allows read-write access. Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Example: Producer time 1000 ms, module address 1

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
601h	8	22h	17h	10h	E8h	03h	00h	00h	00h

The answer you will receive from the module is:

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
581h	8	60h	17h	10h	00h	00h	00h	00h	00h

The heartbeat producer time is not saved inside the non-volatile memory autonomously. It is necessary to store this parameter via object 1010h (refer to “Store Parameters” on page 51).

8.5.2 Node guarding

The NMT master polls each NMT slave at regular time intervals. This time-interval is called the guard time. The response of the NMT slave contains the NMT state of that NMT slave. The node lifetime is given by the guard time multiplied by the lifetime factor. If the NMT slave has not been polled during its lifetime, a remote node error is indicated through the NMT service life guarding event.

Upon life guard error the µCAN.4.ci-BOX will transmit an emergency message with emergency code 8130h.

Guard time

Index 100Ch

The object at index 100Ch defines the guard time. The life time factor multiplied with the guard time gives the life time for the life guarding protocol.

Sub-Index	Data Type	Acc	Name	Default Value
0	Unsigned16	rw	Guard time	0000h

The value is given in multiple of 1 millisecond. The value of 0000h disables the life guarding.

Life time factor

Index 100Dh

The object at index 100Dh defines the life time factor. The life time factor multiplied with the guard time gives the life time for the life guarding protocol.

Sub-Index	Datentyp	Zugriff	Bedeutung	Defaultwert
0	Unsigned8	rw	Life time factor	00h

The value 00h disables the life guarding.

8.6 PDO Communication

The real-time data transfer is performed by means of "Process Data Objects" (PDO). The transfer of PDOs is performed with no protocol overhead.

PDO communication is only possible when the device is in the network state "Operational".

8.6.1 Transmission Modes

Event Driven

Message transmission is triggered by the occurrence of an object specific event. For synchronous PDOs this is the expiration of the specified transmission period, synchronised by the reception of the SYNC object. For acyclically transmitted synchronous PDOs and asynchronous PDOs the triggering of a message transmission is a device-specific event specified in the device profile.

Timer Driven

Message transmission is either triggered by the occurrence of a device-specific event or if a specified time has elapsed without occurrence of an event.

The µCAN.4.ci-BOX does not support the RTR-based PDO transmission types FCh (252_d) and FDh (253_d). Bit 30 inside the COB-ID field of the PDO communication parameter record is always set to 1.

8.6.2 Transmit PDO 1

Index 1800h

The object at index 1800h defines communication parameters for the Transmit-PDO 1.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	5
1	Unsigned32	rw	COB-ID for PDO	180h + Node
2	Unsigned8	rw	Transmission Type	01h
5	Unsigned16	rw	Event Timer	0000h

The object allows read-write access. Only sub-indices 0 to 2 and 5 are supported. An access to other sub-indices will lead to an error message.

COB-ID for PDO

Sub-Index 1 defined the identifier for the Transmit-PDO. The 32-bit value has the following structure.

Bit 31	Bit 30	Bit 29	Bit 28 - 0
PDO valid, 0 = valid 1 = not valid	RTR allowed, 0 = yes 1 = no RTR	Frame type, 0 = 11 Bit 1 = 29 Bit	Identifier,

Table 19: Definition of COB-ID for PDO

In order to enable the PDO the most significant bit (Bit 31) must be set to 0. In order to disable the PDO the most significant bit must be set to 1. The PDO is activated by default (Bit 31 = 0).

8

Transmission Type

The transmission type defines the transmission character of the PDO.

Transmission Type	Description
00h	acyclic synchronous,
01h - F0h (1 - 240 dez)	cyclic synchronous,
FFh (255 dez)	event driven, PDO is sent when Event Timer elapses

Table 20: Setup of transmission type

Transmit PDO 1 sends a message with 8 bytes of data. The data is taken from object 9130h, sub-index 1 and 2 (frequency value of channel 1 and 2).

8.6.3 Transmit PDO 2

Index 1801h

The object at index 1801h defines communication parameters for the Transmit-PDO 2.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	5
1	Unsigned32	rw	COB-ID for PDO	280h + Node
2	Unsigned8	rw	Transmission Type	01h
5	Unsigned16	rw	Event Timer	0000h

The object allows read-write access. Only sub-indices 0 to 2 and 5 are supported. An access to other sub-indices will lead to an error message.

COB-ID for PDO

Sub-Index 1 defined the identifier for the Transmit-PDO. The 32-bit value has the following structure.

Bit 31	Bit 30	Bit 29	Bit 28 - 0
PDO valid, 0 = valid 1 = not valid	RTR allowed, 0 = yes 1 = no RTR	Frame type, 0 = 11 Bit 1 = 29 Bit	Identifier,

Table 21: Definition of COB-ID for PDO

In order to enable the PDO the most significant bit (Bit 31) must be set to 0. In order to disable the PDO the most significant bit must be set to 1. The PDO is activated by default (Bit 31 = 0).

8

Transmission Type

The transmission type defines the transmission character of the PDO.

Transmission Type	Description
00h	acyclic synchronous,
01h - F0h (1 - 240 dez)	cyclic synchronous,
FFh (255 dez)	event driven, PDO is sent when Event Timer elapses

Table 22: Setup of transmission type

Transmit PDO 2 sends a message with 8 bytes of data. The data is taken from object 9130h, sub-index 3 and 4 (frequency value of channel 3 and 4).

8.6.4 Transmit PDO 3

Index 1802h

The object at index 1802h defines communication parameters for the Transmit-PDO 1.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	5
1	Unsigned32	rw	COB-ID for PDO	380h + Node
2	Unsigned8	rw	Transmission Type	01h
5	Unsigned16	rw	Event Timer	0000h

The object allows read-write access. Only sub-indices 0 to 2 and 5 are supported. An access to other sub-indices will lead to an error message.

COB-ID for PDO

Sub-Index 1 defined the identifier for the Transmit-PDO. The 32-bit value has the following structure.

Bit 31	Bit 30	Bit 29	Bit 28 - 0
PDO valid, 0 = valid 1 = not valid	RTR allowed, 0 = yes 1 = no RTR	Frame type, 0 = 11 Bit 1 = 29 Bit	Identifier,

Table 23: Definition of COB-ID for PDO

In order to enable the PDO the most significant bit (Bit 31) must be set to 0. In order to disable the PDO the most significant bit must be set to 1. The PDO is activated by default (Bit 31 = 0).

8

Transmission Type

The transmission type defines the transmission character of the PDO.

Transmission Type	Description
00h	acyclic synchronous,
01h - F0h (1 - 240 dez)	cyclic synchronous,
FFh (255 dez)	event driven, PDO is sent when Event Timer elapses

Table 24: Setup of transmission type

Transmit PDO 3 sends a message with 8 bytes of data. The data is taken from object 9100h, sub-index 1 and 2 (counter value of channel 1 and 2).

8.6.5 Transmit PDO 4

Index 1803h

The object at index 1803h defines communication parameters for the Transmit-PDO 4.

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned8	ro	Largest Sub-Index	5
1	Unsigned32	rw	COB-ID for PDO	480h + Node
2	Unsigned8	rw	Transmission Type	01h
5	Unsigned16	rw	Event Timer	0000h

The object allows read-write access. Only sub-indices 0 to 2 and 5 are supported. An access to other sub-indices will lead to an error message.

COB-ID for PDO

Sub-Index 1 defined the identifier for the Transmit-PDO. The 32-bit value has the following structure.

Bit 31	Bit 30	Bit 29	Bit 28 - 0
PDO valid, 0 = valid 1 = not valid	RTR allowed, 0 = yes 1 = no RTR	Frame type, 0 = 11 Bit 1 = 29 Bit	Identifier,

Table 25: Definition of COB-ID for PDO

In order to enable the PDO the most significant bit (Bit 31) must be set to 0. In order to disable the PDO the most significant bit must be set to 1. The PDO is activated by default (Bit 31 = 0).

8

Transmission Type

The transmission type defines the transmission character of the PDO.

Transmission Type	Description
00h	acyclic synchronous,
01h - F0h (1 - 240 dez)	cyclic synchronous,
FFh (255 dez)	event driven, PDO is sent when Event Timer elapses

Table 26: Setup of transmission type

Transmit PDO 4 sends a message with 8 bytes of data. The data is taken from object 9100h, sub-index 3 and 4 (counter value of channel 3 and 4).

8.6.6 Transmit PDO example

All transmit PDOs are set to the transmission type 01h by default (cyclic synchronous, cyclic every sync). PDO transmission is triggered by reception of a SYNC message (object 1005h).

Example: module address 1, SYNC transmission

ID	DLC
80h	0

As reponse the µCAN.4.ci-BOX will send the following messages:

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
181h	8	Index 9130, Sub 01h				Index 9130, Sub 02h			

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
281h	8	Index 9130, Sub 03h				Index 9130, Sub 04h			

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
381h	8	Index 9100h, Sub 01h				Index 9100h, Sub 02h			

ID	DLC	B0	B1	B2	B3	B4	B5	B6	B7
481h	8	Index 9100h, Sub 03h				Index 9100h, Sub 04h			

PDO communication is only possible when the device is in the network state "Operational".

8.6.7 Synchronisation Message

Index 1005h

The object at index 1005h defines the identifier for the SYNC-message. On reception of a message with this identifier the transmission of PDOs is triggered (refer to “Transmit PDO 1” on page 71)..

Sub-Index	Data Type	Acc.	Name	Default Value
0	Unsigned32	rw	COB-ID SYNC	80h

Only sub-index 0 is supported. An access to other sub-indices will lead to an error message.

Example: Set SYNC-ID to 10, module address 1

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
601h	8	22h	05h	10h	0Ah	00h	00h	00h	00h

As answer you will get the following message:

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
581h	8	60h	05h	10h	00h	00h	00h	00h	00h

The default identifier is 80h in order to ensure a high priority of the SYNC-message.

The SYNC-identifier is not saved inside the non-volatile memory autonomously. It is necessary to store this parameter via object 1010h (refer to “Store Parameters” on page 51)

8.7 Emergency Message

Emergency objects are triggered by the occurrence of a device internal error situation and are transmitted from an emergency producer on the device.

An emergency is different from a SDO error message. The last one only holds the access error to the object dictionary, whereas an emergency indicates a severe hardware/software failure.

The emergency identifier has the default value 128_d + module-address. The emergency message has the following structure:

ID	*DLC*	*B0*	*B1*	*B2*	*B3*	*B4*	*B5*	*B6*	*B7*
	8	Error Code		ER	Manufacturer Specific Error Field				

The following emergency error codes are supported:

Error Code	Description
0000h	Error reset or no error
1000h	generic error
5000h	module hardware
6000h	module software
8100h	CAN controller entered "warning" state
8110h	CAN controller overrun
8120h	CAN controller entered "error passive" state
8130h	heartbeat event / node guarding event
8140h	device recovered bus-off
8150h	identifier collision (Tx-ID reception)

Table 27: Emergency error codes

8

9. Technical Data

Power Supply	
Supply Voltage, U_{PWR}	8 .. 60 V DC, reverse current protected
Power Consumption	1,5 W (60 mA @ 24 V DC) without load
Isolation	Fieldbus/Supply: 500 Veff
Physical Interface	Terminal Block (2,5 mm^2)

CAN-Bus	
Baudrates	20 kBit/s .. 1 MBit/s
Status on the bus	active node
Protocol	CANopen, DS-404
Physical Interface	Terminal Block (2,5 mm^2)

EMC	
Electromagnetic immunity	according to EN 50082-2
Electrostatic discharge	8 kV air discharge, 4 kV contact discharge, according to EN 61000-4-2
Electromagnetic fields	10 V/m, according to ENV 50204
Burst	5 kHz, 2 kV according to EN 6100-4-4
Conducted RF-Disturbance	10 V, according to EN 61000-4-6
Electromagnetic emission	according to EN 50081-2

Mechanic	
Case	Aluminium
Dimensions	125 * 80 * 57 mm (L * B * H)
Weight	540 g
Protection class	IP66

Digital Inputs / Counter	
Impedance	24,2 kOhm
Trigger Level	variable, adjusted via software
Frequency range	0 Hz .. 1 MHz

Digital Outputs	
Type	Highside Power-MOSFET
Maximum voltage	50 V
Maximum current	1,4 A
Short circuit detection	5 A, each output short circuit protected
Module maximum current	6 A

Index

A

Address
  range **23**
  selection **23**
Autobaud **24**

B

Baudrate
  automatic detection **24**
  bus length **8**
  setup **24**
Bootup message **66**

C

Cable length **8**
CAN
  cable **19**
  connection **22**
CANopen
  DS-301 **40**
  DS-404 **60**
Case dimensions **9**
Communication Profile **45**
Counter mode **28**
Cut-off-frequency **29**

D

Device Profile **48**

E

EMC **16**
EMCY
  see Emergency message
Emergency message **77**

F

Frequency mode **29**

G

Gate timing
  description **28**
  object **57**

H

Heartbeat Protocol **66**
  Consumer **67**
  Producer **68**

I

Identity object **53**

M

Manufacturer Device Name **50**
Module Status
  LED **33**

N

Network Management **41**
  Enter Pre-Operational **41**
  Reset Node **42**
  Start Node **41**
  Stop Node **41**
Network manager **7**
Network Status
  LED **33**
NMT
  see Network Management
Node Guarding **69**
Non-fused earthed conductor **12**

O

Object
  1000h **48**
  1001h **48**
  1003h **49**
  1005h **76**
  1008h **50**
  1009h **50**
  100Ah **50**
  1010h **51**
  1011h **52**
  1014h **52**
  1017h **68**
  1018h **53**
  1029h **54**
  1800h **71**
  1801h **72**
  1802h **73**
  1803h **74**
  5020h **56**
  5101h **56**

P

PDO
  see Process Data Object
Power supply **20**
Pre-defined Connection Set **40**
Pre-defined Error Field **49**
Process Data Object **70**
  Transmission modes **70**

S

SYNC
see Synchronisation Message
Synchronisation Message **76**

T

Terminal
binary control lines **31**
CAN bus **22**
GND **20**
power supply **20**
V+PWR **20**
Termination **25**
Trigger level **30**
absolute **58**
relative **59**

MicroControl reserves the right to modify this manual and/or product described herein without further notice. Nothing in this manual, nor in any of the data sheets and other supporting documentation, shall be interpreted as conveying an express or implied warranty, representation, or guarantee regarding the suitability of the products for any particular purpose. MicroControl does not assume any liability or obligation for damages, actual or otherwise of any kind arising out of the application, use of the products or manuals.

The products described in this manual are not designed, intended, or authorized for use as components in systems intended to support or sustain life, or any other application in which failure of the product could create a situation where personal injury or death may occur.

MicroControl
Systemhaus für Automatisierung

MicroControl GmbH & Co. KG
Lindlaustraße 2c
D-53842 Troisdorf
Fon: +49 / 2241 / 25 65 9 - 0
Fax: +49 / 2241 / 25 65 9 - 11
http://www.microcontrol.net