µCAN.1.ti-IP65

Manual for the Temperature Acquisition Module
Version 1.10

Order No.: 12.10.901

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

Inhaltsverzeichnis

µCAN.1.ti-IP65

Remarks on CE-conformance of µCAN-modules

µCAN-modules which have CE-conformance label, have passed test specifications of EU-criteria 89/336/EWG "Electromagnetic Emission and Immunitiy" and standardized European norms (EN).
Papers of declaration for EU-conformance, according to Art.10 of EN, are available at:

MicroControl GmbH & Co. KG
Lindlaustr. 2 c
D-53842 Troisdorf
Germany

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.

1. Safety Regulations

Symbol Explanation

Attention !

This symbol marks a paragraph which explains possible danger. This danger might cause a damage to the system / plant or damage to personnel.

Note

This symbol marks a paragraph which contains useful information for the work with the device or which gives just a hint.

1.1 General Safety Regulations

Attention !

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

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.

Terms of use

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

Safety Regulations

General Safety Regulations

Attention !	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.
Hints for Installation	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.
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.
Attention !	By means of hardware and software safety precautions have to be taken in order to avoid undefined operation of a 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.1.ti-IP65

2.1 Overview

µCAN.1.ti - Temperature Acquisition

The µCAN.1.ti is the right module for temperature acquisition with sensors of type Pt100 and thermo couples J/K. The linearized temperatures are transmitted via CAN bus

.

Pt100 sensors can be connected in 2-wire, 3-wire and 4-wire technology.

The modules are designed for applications where temperatures are measured over a long distance. This reduces the costs for long and expensive thermo couple wire.

Operation of µCAN.1.ti-IP65

Overview

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.1.ti module. The µCAN.1.ti runs on the standard fieldbus CAN.

Typical applications for the µCAN.1.ti are industrial automation, transportation, food industry and environmental technology.

The µCAN.1.ti runs with the protocol

according to the device profile DSP-404. Other protocol stacks are available on request.

compatible

By means of modern chip technology the protocol CANopen was implemented. This gives you the possibility for flexible use of the µCAN.1.ti module.

space saving and compact

The µCAN.1.ti is designed for heavy duty applications. The aluminium cast ensures protection class IP65. The compact, space saving case gives the freedom to mount the module in many places.

inexpensive and service friendly

The quick and easy integration of the µCAN.1. ti in your application reduces the development effort. Costs for material and personnel are reduced. The easy installation makes maintenance and replacement quite simple.

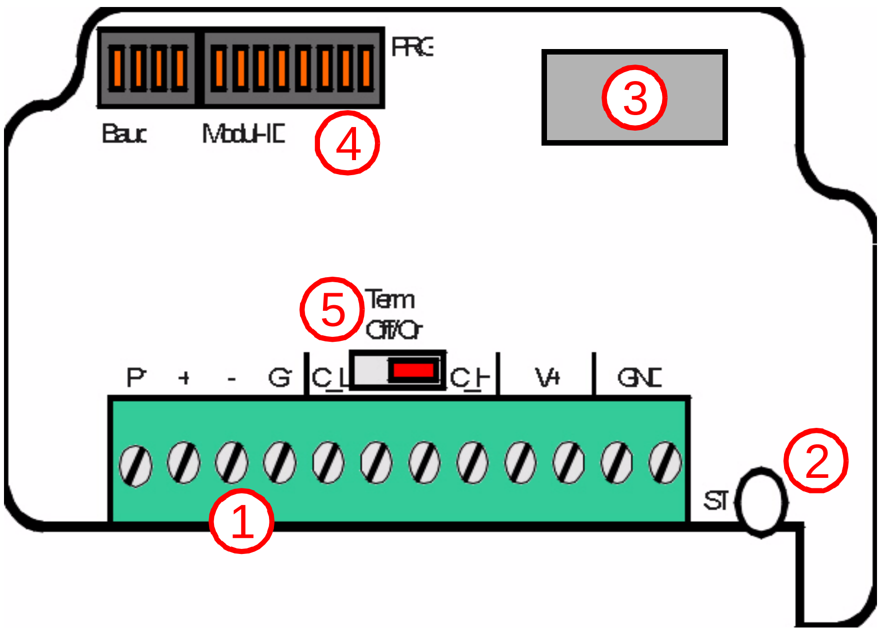
3. Project Planning

3.1 Introduction

The chapter Project Planning contains information which are important for the system engineer. These information include case dimensions and conditions of use.

3.2 Module Layout

The following figure shows the top view of the µCAN.1.ti PCB. Use the figure to identify the terminal blocks, LED's and DIP-switches.



1 - Terminal block for sensor inputs / power supply / CAN
2 - Status LED
3 - DC/DC converter
4 - Baudrate Setup / Module ID
5 - Termination switch for CAN bus

Fig. 1: Top view of the µCAN.1.xx-IP65 PCB

3.3 Operation Area

The µCAN.1.ti is a robust field module for acquisition and linearization of temperatures in industrial applications. Temperatures can be acquired by different kinds of sensors. The following sensors can be connected to the µCAN.1.ti:

- Thermo couple type J,
- Thermo couple type K,
- Pt100 resistor

The module gathers the analogue signals of temperature sensors and performs a linearization. The temperature is transmitted in degree Celsius via CAN bus. Fraction of sensor (thermo couple / Pt100) and short circuit of sensor (Pt100) are detected.
The PCB is incorporated in a robust case of protection class IP65. The µCAN.1.ti is suited for mounting outside the switch cabinet. The idea behind that concept is to acquire the signals direct at the test point. Long wires for the sensors are not longer necessary. Influence of EMI is reduced.

The µCAN.1.ti needs four wires for connection to the power supply and CAN bus. Special CAN bus cables can be ordered from MicroControl (refer to Ordering Information).

3.4 Maximum System Layout

For an operational system at least one network manager (or supervisor system) must be connected to the bus. This network manager might be a PLC or PC equipped with a CAN card. Every µCAN.1.ti module is an active node.

A CAN network may have one network manager and up to 127 network slaves (refer to Fig. 2, “Maximum System Layout”). Every module gets a unique address, which is set up via a DIP switch. The CAN 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 29).

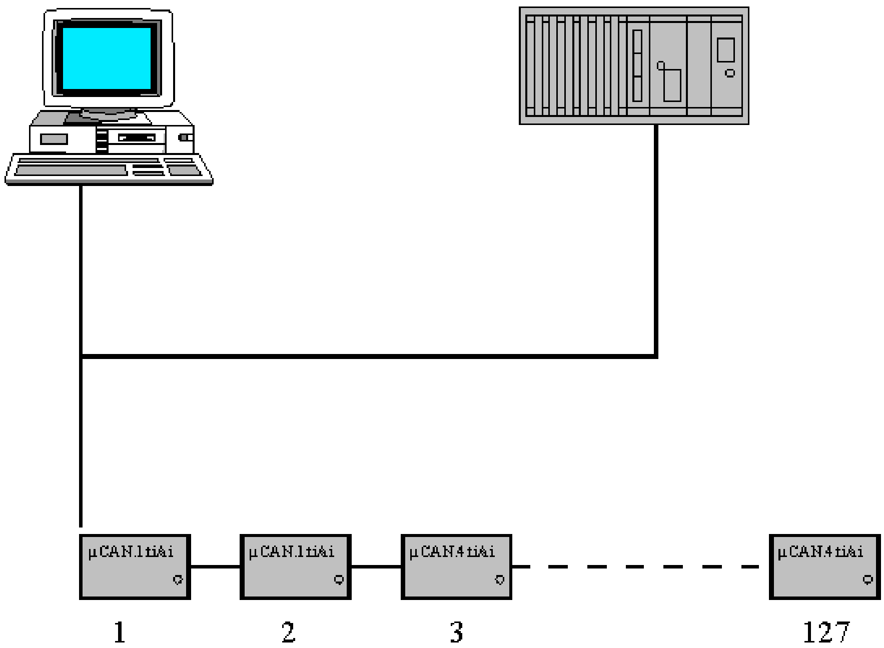


Fig. 2: Maximum System Layout

Maximum System Layout

The maximum cable length depends on the selected baudrate. The following table shows the maximum cable length recommended by CiA*. These distances can be realized with the µCAN.1.ti .

Baudrate in kBit/s	Cable Length in m
1000	25
800	50
500	100
250	250
125	500
100	650
50	1000
20	2500
10	5000

Table 1: Dependence of baudrate from cable length

*CAN in Automation Internationial Users and Manufacturers Group e.V. MicroControl is a member of CiA and joins the working groups for development of new protocols and standards.

3.5 Case Dimensions

The case dimensions of the module are given in the drawing below. The high protection class IP65 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.

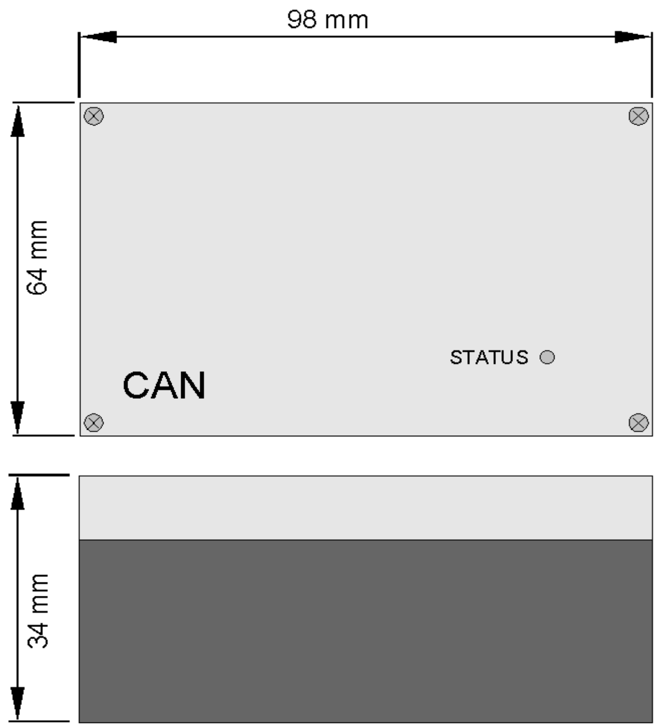


Fig. 3: Case Dimensions

4. Assembly and Disassembly

4.1 Safety Regulations

Attention !

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.

Attention !

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 modules 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

You need a cable with two conductors 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 cable with four conductors in order to run the CAN bus over the same cable.

Earthed Conductor

The non-fused earthed conductor is connected at the terminal outside the case (refer to Fig. 4, "Connection of earthed conductor"). The non-fused earthed conductor may not lead inside the case because of EMI.

Attention !

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

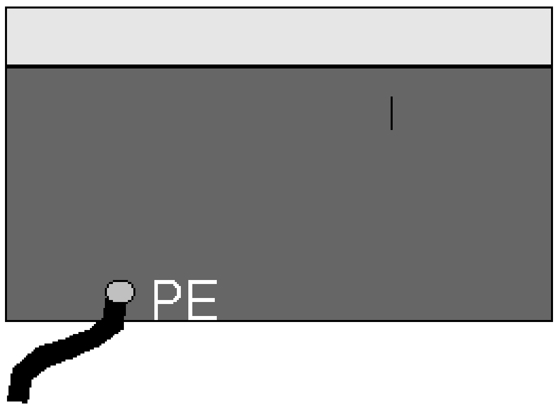


Fig. 4: Connection of earthed conductor

Attention !

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.

Note

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

For a quick identification of the modules during operation you may use paper sticker. Please write down the ID that is set for the module.

Note

Please make sure that the last node that is installed to the CAN bus is terminated with a resistor (refer to “Termination” on page 29).

4.4 Disassembly

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

Open the cover from the module and remove the temperature sensors first. Now you can remove the cables for CAN bus and power supply from the terminals.

Unlock the fixing screws and remove the module. 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 µCAN modules is characterized by following features:

- The CAN bus potential is isolated from the power supply.
- The electronic of the µCAN modules is isolated from the power supply.
- All µCAN modules have a separate power supply.
- 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

General

All inactive metal plates must be grounded with low impedance. By this step 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.

Grounding of µCAN-Modules

The µCAN modules are grounded by non-fused earthed conductor at the terminal outside the case (“Connection of earthed conductor” on page 16). The ground potential may not be connected to a terminal inside the case.

Grounding of other modules

If µCAN modules are shipped in a plastic case they have to be grounded with a metal tape.

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 Types

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 Layout

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 (temperature sensors).

Attention !

The shield of the CAN bus cable may never lead inside the µCAN case. Never connect the shield to one of the terminals inside the case.

For a fixed operation the shield of the CAN bus cable should be connected to safety earth.
The CAN cable must meet the requirements of ISO11898. The

cable must have the following features:

- Impedance, measured between two signal lines, in the range of 108-132 Ohm (nom. 120 Ohm)
- Specific Resistance of 70 mOhm/m
- Specific Signal Delay of 5 ns/m

The CAN bus cable is connected to the µCAN module via screw-type terminals inside the case. For the pinning of the terminal refer to “Connection of the CAN Bus” on page 24 of this manual.

Attention !

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

5.3 Power Supply

The modules of the µCAN family are designed for industrial applications. The nominal supply voltage is 24 V DC. By means of a DC/DC converter the electronic 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. For further information please refer to the technical data of the module.

The input is protected against confusing the poles.

Terminals for Power Supply

Please make sure not to confuse the poles when connecting to the power supply. The positive supply is connected to the terminal V+. The two V+ terminals are internally linked to feed the supply through the module. The negative supply is connected to the terminal GND. The two GND terminals are also internally linked.

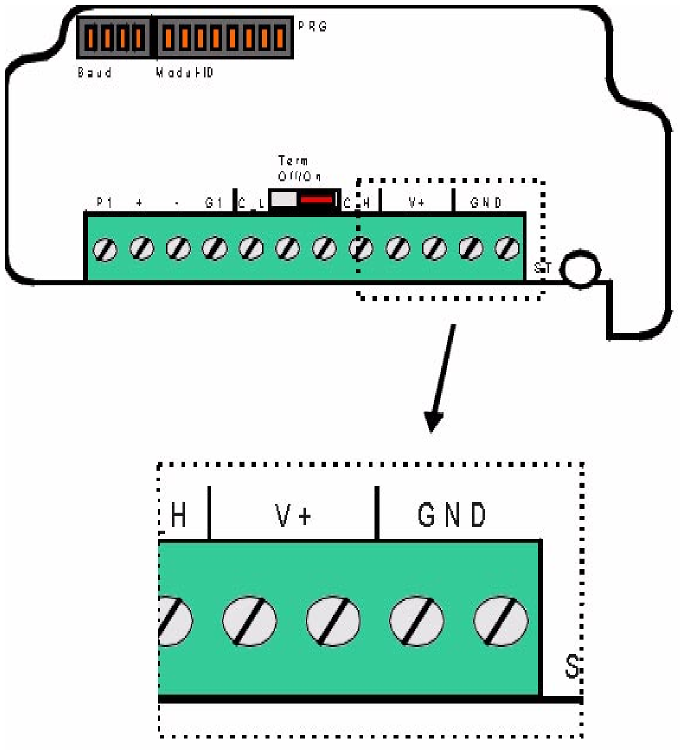


Fig. 5: Connection of power supply

Attention !

A cable shield may not lead into the housing or may not be connected to a terminal inside the housing. Cable shields have to be connected to the terminals outside the housing (“Connection of earthed conductor” on page 16).

5.4 CAN-Bus

The two wires of the CAN bus are connected to the appropriate terminals. Please make sure that the CAN bus is fed from the right side into the module and keep the wires as short as possible. The terminals for CAN-H respective CAN-L are internally linked. By this the CAN bus can be connected through the module.

Note

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

Terminals for CAN

The CAN bus conductor with positive potential must be connected to the terminal C-H (CAN High). The CAN bus conductor with negative potential must be connected to the terminal C-L (CAN Low).

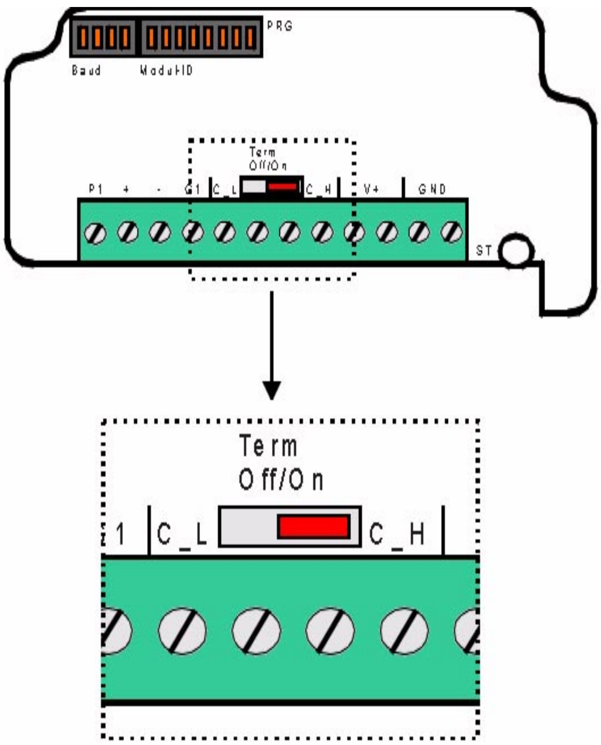


Fig. 6: Connection of the CAN Bus

Attention !

If you confuse the poles the communication on the bus will not be possible. The shield of the CAN bus 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 (refer to Fig. 4 on page 16).

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

Adress selection and baudrate setting is done with two DIP-switches which you can see in the following two figures.

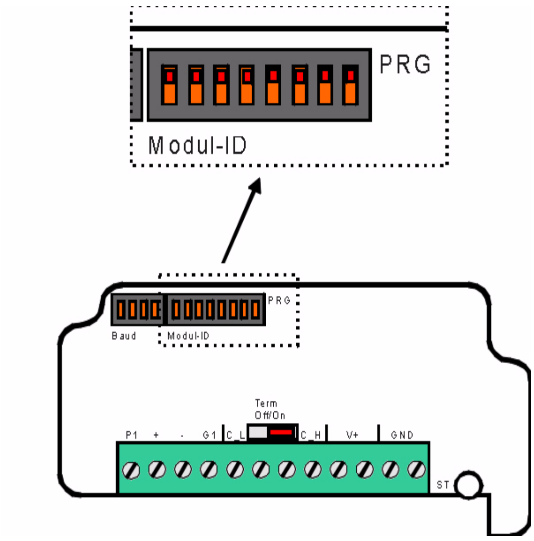


Fig. 7: Adress setting

Address selection of the µCAN modules is done with the DIP switche, marked "Modul-ID" which are located at the top of the printed board. Selection of the address may be done with a small screw driver.

The address of the module (so called Node ID) is set by one byte. 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 CAN) or the power supply is switched off.

Address Selection

Address range

As mentioned before, the Node ID is coded with one byte. In CANopen networks the highest address is limited to 127, which means the last switch (label 8) has no function.

Also, a Node ID of 0 is not allowed. If a Node ID of 0 is selected, the µCAN module will stay in the initialization routine. The module will only run through the initialization process if the address is within the range from 1 to 127.

The decimal address range from 1 to 127 corresponds to the hexadecimal notation 01h to 7Fh.
Every node within a CANopen network must have a unique ID. Two nodes with the same ID are not allowed.

You find a table with calculation for switch positions for Node IDs from 1 to 31 in the appendix of this manual.

5.6 Baudrate Selection

Baudrate selection of the µCAN modules is done with a single HEX-rotation coder, which is located at the lower site of the printed circuit board. Selection of the baudrate may be done with a small screw driver.

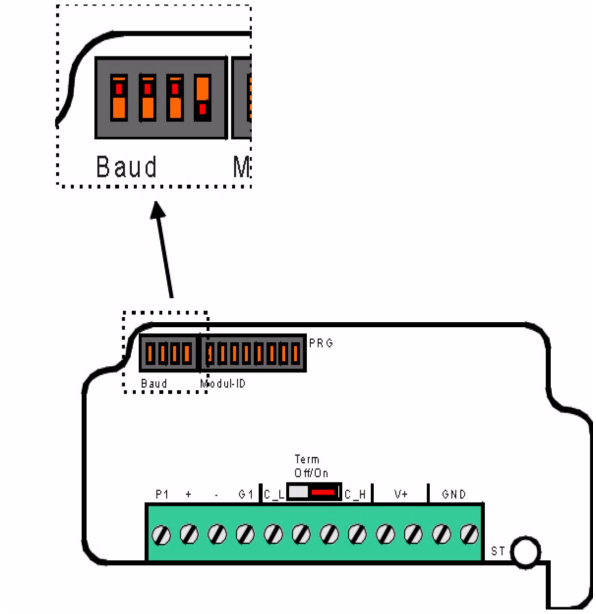


Fig. 8: Setup of baudrate

Baudrate Selection

The supported baudrates of the µCAN module are given in the following table. The values are recommended by the CiA. The table also shows the position of the DIP switch.

Baudrate (kBit/s)	Value	DIP-setting
1000	9	1 0 0 1
800	8	0 0 0 1
500	7	1 1 1 0
250	6	0 1 1 0
125	5	1 0 1 0
100	4	0 0 1 0
50	3	1 1 0 0
20	2	0 1 0 0
10	1	1 0 0 0

Table 2: Baudrate Settings

5

5.7 Termination

The last module in the network has to be terminated with a resistor of 120 ohms. That means the last module is not reflecting back power and the communication can not be disturbed.

Termination of the µCAN modules is easy to perform. Simply set the switch shown in the figure to position "ON". Then there will be internally a resistor of 120Ohm connected between the terminal block C_L and C_H.

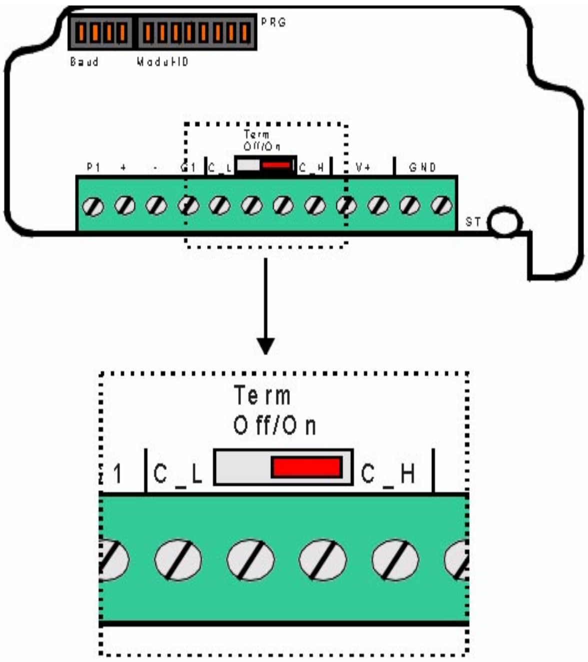


Fig. 9: Termination of CAN Bus

Attention !

Please make sure that only the devices at both ends of a CAN bus are terminated.

6. Signal Inputs

This chapter of the manual will show you how different kinds of temperature sensors and analogue standard signals are connected to the µCAN modules. 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.

Marking of Channels

The µCAN.1.ti has one input. The terminal is marked with P1, +, - and G1. The terminal with + and - belongs to the differential signal input, the terminals with the marking P1 and G1 are for additional sensor (power) supply ie. Pt100 or strain gauge.

Attention !

All sensor types or analogue signals may only be connected in power off state in order to prevent a damage of the electronic.

6.1 Connection of Pt100

As mentioned before, the µCAN.1.ti works with Pt100 sensors as well as with thermo couples. Sensors of type Pt100 can be connected in three different ways.

2-wire

Connection between the Pt100 resistor and the electronic is done with 2 wires (refer to fig. “Possible wiring for Pt-100” on page 32). As every conductor these wires have an resistance, which is switched in series to the Pt100 resistor. As a result the Pt100 resistor and the resistance of the wires are added. That means a higher temperature than the really present temperature is measured. To reduce this effect, the resistance of the wires must be compensated manually.

3-wire

In most cases the Pt100 resistor is used in a 3-wire version. For this type of sensor an additional wire is connected to the Pt100 resistor. By this a second measuring circuit is present. The second measuring circuit is used as reference. For a 3-wire Pt100 sensor the offset by the conductor resistance and the influence of the ambient temperature can be compensated. However the ambient temperature must influence all three wires.

4-wire

For a 4-wire Pt100 the current is fed into the resistance via two additional conductors. The voltage drop over the resistor is measured with the parallel conductors. A compensation is not necessary. For a high-impedance input the resistance of the conductor material can be neglected. The voltage drop over the Pt100 resistor is independent from the conductor resistance.

The three possible kinds of connection are shown in Figure 10. As an example the connection to channel 1 is displayed.

Connection of Pt100

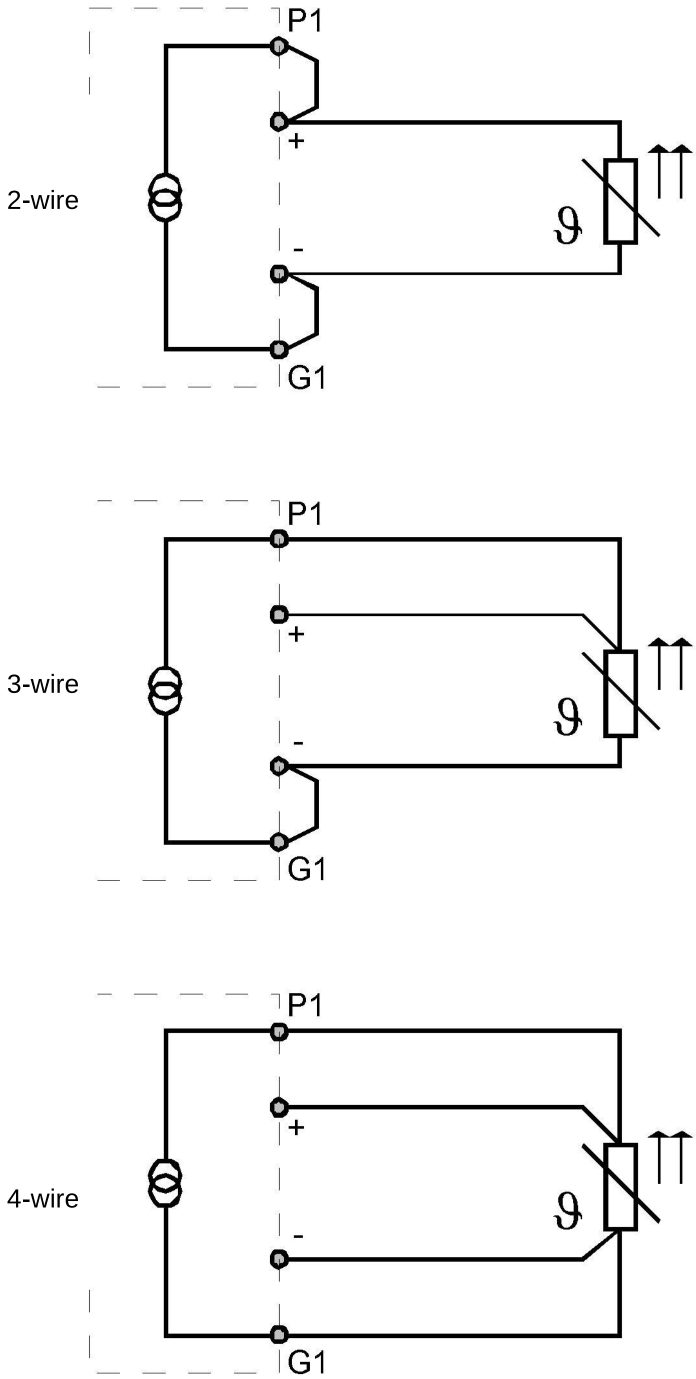


Fig. 10: Possible wiring for Pt-100

Cross Bars

If 2-wire or 3-wire Pt100 sensors are used, some cross bars have to be installed (refer to Figure 10).

For a 2-wire Pt100 a cross bar has to be installed between terminal P1 and the corresponding terminal with label „+“. Also a cross bar has to be installed between terminal G1 and the corresponding terminal with label „-“.

For a 3-wire Pt100 only one cross bar for each channel has to be installed. This cross bar must be between terminal G1 and the corresponding terminal with label „-“.

For a 4-wire Pt100 no cross bars have to be installed.

The following figure shows input channel 1 with all four terminals and the terminal labels.

6

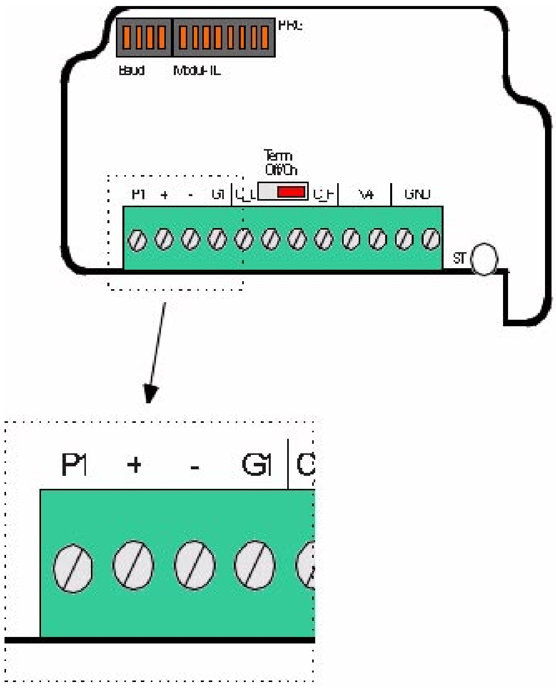


Fig. 11: Outline of Measurement Input

Connection of Pt100

Limitation of 3-wire Pt100

The µCAN.1.ti does not have a compensation for 3-wire Pt100 resistors, so the resistance of the conductor and temperature of the conductor will have an influence on the measured value. However this influence is half as bad as you would use 2-wire Pt100 resistors. In practical applications a cable length of up to 5 meters (3-wire Pt100; 0,2 mm^2 copper) can be used which has an impact on the measured temperature of less than 0,5K (absolute).

6.2 Connection of thermo couple

Temperature sensors of type thermo couple can be connected very easy. No additional links have to be installed. The sensor is connected to the terminals „+“ and „-“ of the desired measurement channel.

Note

Please take care not to confuse the poles when connecting the thermo couple.

The following figure shows the connection of a thermo couple to measurement input 1.

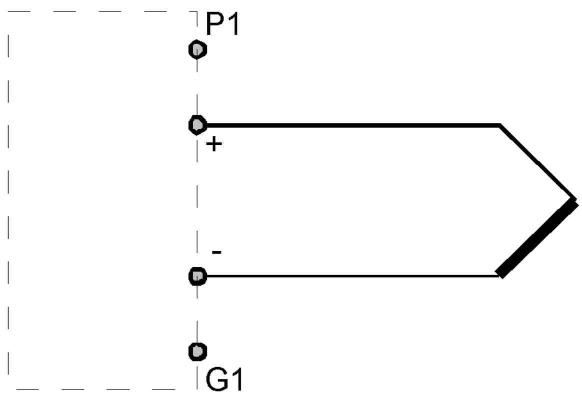


Fig. 12: Connection of Thermo Couple

Attention !

The shield of the temperature sensor may not lead inside the case in order to avoid EMI. The shield has to be connected outside the case to the appropriate terminal.

7. Diagnosis

7.1 Position of LEDs

All modules of the µCAN family have LEDs to display the operating state and to signalize an error state. The light of the LEDs can be seen through beam waveguides on top of the housing (refer to “Case Dimensions” on page 14).

The modules incorporate a bi-color LED named with „Status“. The following figure shows the position of the LEDs.

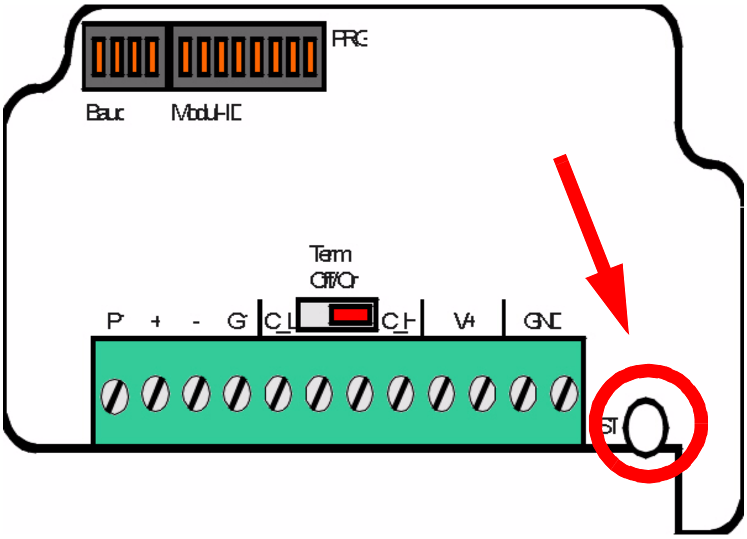


Fig. 13: Position of LEDs on the module

7.2 Diagnosis in Pre-Operational State

7.2.1 Diagnosis in Pre-Operational State

After Power-on the LED will be turned on and off for a short moment and turned on again.

Status of LED	Meaning
quick flashing (2 times per second)	Device is in Pre-Operational state
off	Supply voltage not OK, Hardware not OK
flashing red and green	Supply voltage OK, but no CAN bus connection (the module is waiting for a ACK-pulse from a nother CAN node) maybe the wrong baudrate is selected or any other physical error on the bus
flashing red	Inputs are left open or wrong sensor type

Note

The default setting for sensor type is thermo couple type J (refer to “Sensor-Type” on page 53).

7.2.2 Diagnosis in Operational State

With help of the LED the communication on the CAN bus as well as the state of the sensor inputs can be supervised.

Status of LED	Meaning
quick flashing (2 times per second) green	Device is in Pre-Operational state
slow blinking (1Hz) green	Device is in Operational State and there is communication with the device via CAN bus
on, green	Device is in Operational State and there is no communication
flashing red	Inputs are left open or wrong sensor type

8. CANopen Protocol

8.1 Introduction

This chapter provides the user information on how to connect the modules of the µCAN-series to a CANopen-Master (CANopen-Manager). This CANopen-Master can be a PLC, a PC with CAN-Card or any other CAN-Device with master functionality.

This documentation provides the actual implemented functions and services of the µCAN-devices.

For further information on the protocol you can also contact us via e-mail: info@MicroControl.net

CANopen related www-sites:

http://www.can-cia.de (CAN in Automation Organization)

http://www.MicroControl.net

Literature:

CiA Draft Standart 404 (CANopen)
CAN in Automation (CiA)
International Headquarters
Am Weichselgarten 26
D-91058 Erlangen

CiA Magazine
CANopen -
Lean Networking for Industrial Automation
Published April 1998

CAN - Controller Area Network
Wolfhard Lawrenz (Editor)
Hüthig, 1994
ISBN 3-7785-2263-7

8.2 Network Management

After power-up the µCAN-module transmits a "Boot-Up Message". This message does not contain any information except the "Heart Beat"-ID which is by default 700h (1793d) + module-address.

Predefined Connection Set
COB-ID = Function Code (4 Bit) + Module-ID (7 Bit)

SDO-requests by the master have the message identifier 1536d (600h) + module-address. The slave answers with a message identifier 1408d (580h) + module-address.

Object	COB-ID (decimal)
Network Management	0
SYNC	128
EMERGENCY	129 - 255
PDO 1 (tx)	385 - 511
PDO 2 (tx)	641 - 767
SDO (rx)	1409 - 1535
SDO (tx)	1537 - 1663
Heart Beat Protocol	1793 - 1919

Table 3: Pre-defined connection set

Network Management

Start Node

Start Node

ID	DLC	Byte 1	Byte 2
0	2	01h	Node

Node = module address, 0 = all modules

By transmitting the "Start Node" command the CAN-node will be set in Operational mode. This means that the node can handle PDO-communication (e.g. sending all process values in one message).

Stop Node

Stop Node

ID	DLC	Byte 1	Byte 2
0	2	02h	Node

Node = module address, 0 = all modules

The command "Stop Node" sets the CAN-node into Pre-Operational-mode.

Reset Node

Reset Node

ID	DLC	Byte 1	Byte 2
0	2	81h	Node

Node = module address, 0 = all modules

On receiving the "Reset Node" command the CAN-node will do an internally software-triggered hardware reset. This will force the node to initialize the complete hardware and also will generate the sending of the "Boot-Up Message" (see before).

8.3 SDO-Commands

The µCAN.1.ti-IP65 supports the below listed SDO-Indices:.

Index	Name	Page
1000h	Device Profile	45
1001h	Error Register	46
1005h	COB-ID SYNC-Message	72
1008h	Manufact. Device Name	47
1009h	Manufact. Hardware Version	48
100Ah	Manufact. Software Version	48
1010h	Store Parameters	49
1011h	Restore Default Parameters	50
1014h	Emergency ID	75
1017h	Producer Time / Heart Beat	64
1018h	Identity Object	51
1800h	1st Transmit PDO Parameter	67
1801h	2nd Transmit PDO Parameter	69
1A00h	1st Transmit PDO Mapping	71
1A01h	2nd Transmit PDO Mapping	71
6110h	Sensor Type	53
6111h	Autocalibration	54
6112h	Operating Mode	55
6131h	Physical Unit Process Value	56
6132h	Decimal Digits Process Value	57
6150h	Analog Input Status	58
7100h	Analog Input Field Value	58
7120h	Input Scaling 1 Field Value	59
7121h	Input Scaling 1 Process Value	60
7122h	Input Scaling 2 Field Value	61
7123h	Input Scaling 2 Process Value	61

SDO-Commands

Index	Name	Page
7124h	Input Offset	62
7130h	Analog Input Process Value	62

Structure of SDO-Commands

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
	8	CMD	Index		Sub-Indx	Data			

The calculation of the ID for a SDO message is shown under “Network Management” on page 40.

The Command Byte (**CMD**) has the following meaning:

Master wants to read from Slave	40h
Slave answers on the read-request	42h
Master wants to write to Slave	22h
Slave answers on the write-request	60h

Note

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

Attention !

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

8.3.1 SDO-Error Messages

The access to an unsupported object (index) leads to an SDO-Error Message. This SDO-Error Message has the following format:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
	8	80h	Index		Sub-Indx	Additional Code		Error Code	Error Class

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

The error messages may have the following contents:

Error Class	Error Code	Additional Code	Meaning
05h	04h	00h 01h	Client/Server command specifier not valid or unknown
06h	01h	00h 00h	Attempt to read a write only object
06h	01h	00h 01h	Attempt to write a read only object
06h	09h	00h 11h	Sub-index does not exist
06h	02h	00h 00h	Object does not exist

8.4 Communication Profile Objects

The index range from 1000h to 1FFFh describes the Communication-Profile for CANopen devices. These indices provide all parameters which concern the CANopen-network. This area is common for all CANopen devices.

Device Profile

By a read-access on Index 1000h the Device Profile can be polled.

Example: Read-Access, Module-Adress = 2, Index 1000h, Sub-Index = not supported within this Index

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	40h	00	10h	00	00	00	00	00

As answer you will receive from the µCAN.1.ti is:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42	00	01h	00	94h	01h	02h	00

Byte 5 + Byte 6 = 0194h = 404d (Device Profile Number)
Byte 7 + Byte 8 = 0002h = 2 (Additional Information) - Analog Input.

Index 1000h is Read-Only, no Sub-Indices are supported. By writing on this Index (or reading a Sub-Index unequal to "0") you will receive a SDO-Communication Error (see “SDO-Error Messages” on page 44).

8.4.1 Error-Register

Error Register

By a read-access on Index 1001h the state of the error register can be polled.

Example: Read-Access, Module-Adress = 2, Index 1001h, Sub-Index = not supported within this Index

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	40h	01h	10h	00	00	00	00	00

As answer you will receive the Error status of the device. The following errors are supported:

Generic Error

Error 1: Bit 0 in Byte 5 is set.
The Generic Error is generated by the signal input of the µCAN.1.ti. This error means: Break or Short-Circuit of the connected sensor, positive or negative overload on one or more channels.

Communication Error

Error 2: Bit 4 in Byte 5 is set.
The Communication Error will be generated by (physical) errors on the CAN-network. A complete list of supported Communication Errors can be seen under “Emergency Message” on page 73.

Index 1001h is Read-Only, no Sub-Indices are supported. By writing to this Index (or reading a Sub-Index unequal to "0") you will receive a SDO-Communication Error (see “SDO-Error Messages” on page 44).

8.4.2 Device Name

Manufacturer Device Name

By a read-access on Index 1008h the Manufacturer Device Name can be polled.

Example: Read-Access, Module-Adress = 2, Index 1008h, Sub-Index = not supported within this Index

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	40h	08h	10h	00	00	00	00	00

As answer you will receive from the µCAN.1.ti:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42	00	01h	00	31h	2Eh	74h	69h

Byte 5 = 31h represents ASCII =1
Byte 6 = 2Eh represents ASCII =.
Byte 7 = 74h represents ASCII =t
Byte 7 = 69h represents ASCII =i

Index 1008h is Read-Only, no Sub-Indices are supported. By writing on this Index (or reading a Sub-Index unequal to "0") you will receive a SDO-Communication Error (see “SDO-Error Messages” on page 44).

8.4.3 Hardware Version

Manufacturer Hardware Version

By a read-access on Index 1009h the Manufacturer Hardware Version can be polled.

Example: Read-Access, Module-Adress = 2, Index 1009h, Sub-Index = not supported within this Index

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	40h	09h	10h	00	00	00	00	00

As answer you will receive from the µCAN.1.ti a message with ASCII-coded Hardware version (see “Device Name” on page 47).

Index 1009h is Read-Only, no Sub-Indices are supported. By writing on this Index (or reading a Sub-Index unequal to "0") you will receive a SDO-Communication Error (see “SDO-Error Messages” on page 44).

8.4.4 Software Version

Manufacturer Software Version

By a read-access on Index 100Ah the Manufacturer Device Name can be polled.

Example: Read-Access, Module-Adress = 2, Index 100Ah, Sub-Index = not supported within this Index

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	40h	0Ah	10h	00	00	00	00	00

As answer you will receive from the µCAN.1.ti a message with ASCII-coded Hardware version (see “Device Name” on page 47).

Index 100Ah is Read-Only, no Sub-Indices are supported. By writing on this Index (or reading a Sub-Index unequal to "0") you will receive a SDO-Communication Error (see “SDO-Error Messages” on page 44).

8.4.5 Store All Parameters

Store Parameter

By writing on Index 1010h all parameters in the following table a stored on the module. Parameters are stored on an EEPROM. The default parameter values are shown in the list below.

Index	Name	Default
6110h	Sensor Type	Thermo J
6112h	Operating Mode	Channel On
7120h	Input Scaling 1 Field Value	0
7121h	Input Scaling 1 Process Value	0
7122h	Input Scaling 2 Field Value	0
7123h	Input Scaling 2 Process Value	0
7124h	Input Offset	0
1014h	Emergency ID	80h + ID
1017h	Producer Time / Heart Beat	0 ms
1005h	SYNC-ID	80 h
1800h	PDO 1 - Parameter	PDO active, ID 180h + Modul-Address, Trans-missiontype 1
1801h	PDO 2 - Parameter	PDO not active, ID 280h + Modul-Address, Transmissi-ontype 1

Table 4: Stored Parameters

The storing will be generated by transmitting the code "save" as ASCII in the data area of Index 1010h. The message has the following format:

Example: Write-Access, Module-Adress = 2, Index 1010h, Sub-Index = 01h

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	22h	10h	10h	01h	73h	61h	76h	65h

The answer (after succesful storing) you will receive is:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	60h	10h	10h	01h	00	00	00	00

8.4.6 Load Default Parameters

Restore Default Parameter

By writing on Index 1011h the default parameters can be loaded. The values of the default parameters can be found under “Store All Parameters” on page 49.

The loading will be generated by transmitting the code "load" as ASCII in the data-area of Index 1011h. The message has to look like the following example.

Example: Write-Access, Module-Adress = 2, Index 1011h, Sub-Index = 01h

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	22h	11h	10h	01h	6Ch	6Fh	61h	64h

The answer (after successful loading) you will receive is:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	60h	11h	10h	01h	00	00	00	00

The loading process will overwrite the values of the EEPROM with the default parameters. So the user doesn´t has to store these "new" parameters on EEPROM by using Index 1010h.

8.4.7 Identity Object

Identity Object

Index 1018h contains the so called "Identity Object" which includes several parameters.

The Index has the following structure:

Index	*Sub-Index*	*Parameter*	*Access*
1018h	0	Number of Subindices	ro
	1	Vendor ID	ro
	2	Product Code	ro
	3	Revision Number	ro
	4	Serial Number	ro

ro - Read Only, r / w - Read / Write

Vendor ID

The vendor ID is an unique number which can be used to identify the manufacturer of the module. The numbers are managed by the CiA (http://www.can-cia.de) worldwide. **MicroControl has the ID = 0E h.**

Product Code

The product code is a mnufacturer specific code, which in our case represents the ordering number of the MicroControl catalogue.

Revision Number

The revision number consits of two 16bit values. The higher 16bit represents the revision of the CANopen parts of the software and the lower 16 bits represents the general firmware release.

Serial Number

The serial number is also manufacturer specific and represents the coded date, on which the module was checked and calibratded. So on request of this Index as answer you will get:

Byte1: Day (Format DD) (hex)
Byte2: Month (Format MM) (hex)
Byte3: Year (Format YY) (hex)
Byte4: counter number (hex)

8.5 Device Profile Objects

In this section you will find all device profile specific indices for the µCAN-modules. These indices are implemented according to the DS-404 device profile.

Index	Name	Page
6110h	Sensor Type Selected sensor input	53
6111h	Autocalibration Start of calibration	54
6112h	Operating Mode Switching ON/OFF of input	55
6131h	Physical Unit Process Value	56
6132h	Decimal Digits Process Value	57
6150h	Analog Input Status	58
7100h	Analog Input Field Value Direct ADC value	58
7120h	Input Scaling 1 Field Value	59
7121h	Input Scaling 1 Process Value	60
7122h	Input Scaling 2 Field Value	61
7123h	Input Scaling 2 Process Value	61
7124h	Input Offset	62
7130h	Analog Input Process Value	62

Table 5: Device specific indices (DS-404)

8.5.1 Sensor-Type

Index 6110h

Writing (Selecting a different/new Sensor) and reading (Monitoring of the selected Sensor) of sensor inputs is done with index 6110h. The object has got the following structure.

Index	Sub-Index	Parameter	Access
6110h	0	Number of supported channels	ro
	1	Parameter channel 1	r / w

ro - Read Only, r / w - Read / Write

A write access to index 6110h does always change the parameters on four channels at one time. The user can not use different sensor types with one module.

The possible sensor types which are supported are shown in the following list:

Parameter	Sensor Type
01 h	Thermocouple Type J (Fe-CuNi)
02 h	Thermocouple Type K (NiCr-NiAl)
1E h	Pt 100

Default parameter on first boot-up is Thermocouple Type J.

After first power-up of the device the user can directly take values with Thermocouple Type J. No further definitions or parameters have to be written.

Example:

Instead of Thermo J the Sensors should be Pt100 on all of the four channels. The message has to look like this:

Write Access, module address = 2, Index 6110h, Sub-Index = 1

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	22h	10h	61h	01h	1Eh	00	00	00

The answer (after successful selection of the new sensor) you will receive is:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	60h	10h	61h	01h	00	00	00	00

Now the module works with Pt100 on all of the four input-channels. By reading from Index 6110h, Sub-Index 1 the user can verify the selected Sensor-Type.
In case of an access-error (e.g. trying to write on a read-only object) the module will send an SDO-Error Message (see “SDO-Error Messages” on page 44).

Attention !

The changed parameter of the input signal / Sensor Type will not be stored on EEPROM automatically. This has to be done by the user (see “Store All Parameters” on page 49).

8.5.2 Auto-Calibration

Index 6111h

A write operation to index 6111h will start the calibration routine on the input.

Attention !

The module µCAN.1.ti is calibrated in factory. The end-user may not calibrate the device.

8.5.3 Operating Mode

Index 6112h

Writing and Reading of Operating Mode from the different channels is done with Index 6110h. The object has the following structure.

Index	Sub-Index	Parameter	Access
6112h	0	Number of supported channels	ro
	1	Parameter channel 1	r / w

ro - Read Only, r / w - Read / Write

The different Operating Modes which are supported are shown in the following list
:

Parameter	Status
00 h	Channel Off
01 h	Normal Operation

By default the channel is turned on. That means the temperatures on the input channel can be measured just after the initial power-up.

Example:

You do not want to use the input for your application. To turn the input off, you have to send the following message:
Write access, Module-ID = 2, Index = 6112h

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	22h	12h	61h	01h	00	00	00	00

As answer you will receive the folloing message from the module µCAN.1.ti:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	60h	12h	61h	01h	00	00	00	00

The input is now turned off. If this input has caused an error, this error will be reset now (emergency message with error reset code). When requesting data from that input you will always receive the value 0.

Attention !

A change of this parameter will not be stored inside the EEPROM automatically. This has to be done manually by the user (see “Store All Parameters” on page 49).

8.5.4 Physical Unit

Index 6131h

By a read-access on Index 6131h the Physical Unit of the Process Value can be requested. This object is read-only and has the following structure:

Index	Sub-Index	Parameter	Access
6131h	0	Number of Inputs	ro
	1	Physical Unit for Input 1	ro

ro - Read Only, r / w - Read / Write

Example:

Read the Physical Unit for input 1.
Read Access, Module-ID = 2, Index = 6131h

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	40h	31h	61h	01h	00	00	00	00

The answer to this request on input 1 is:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	31h	61h	01h	10h	30h	00	00

The data bytes 5 + 6 hold the value according to DS-404 with the physical unit. Here in the example you see 3010h - Degree Celsius.

8.5.5 Number of Decimal Digits

Index 6132h

By a read-access on Index 6132h the Number of Decimal Digits of the Process Value can be requested. This object is read-only and has the following structure:

Index	Sub-Index	Parameter	Access
6132h	0	Number of Inputs	ro
	1	Number Decimal Digits for Input 1	ro

ro - Read Only, r / w - Read / Write

A read-access to this index (refer to “Physical Unit” on page 56) will cause the following answer:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	32h	61h	01h	01h	00	00	00

Data byte 5 holds the value 01h, i.e. the process values are re-presented with one decimal digit.

8.5.6 Input State

Index 6150h

By a read-access on Index 6150h the Input State of the device can be requested. This object is read-only and has the following structure:

Index	Sub-Index	Parameter	Access
6150h	0	Number of Inputs	ro
	1	State for Input 1	ro

ro - Read Only, r / w - Read / Write

A read-access to this index (refer to "Physical Unit" on page 56) will cause the following answer:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	50h	61h	01h	02h	00	00	00

Data byte 5 holds the value 02h, i.e. there is a positive overload on channel 1.

The data byte 5 of the CAN message represents the input state in the following manner:

Parameter	Status
00 h	Measuring Value valid
01 h	Measuring Value not valid
02 h	Positive Overload
04 h	Negative Overload

If the measured value is within the input range, the request will return the value 00h, i.e. process value is valid.

8.5.7 Field Value (A/D-Converter)

Index 7100h

By a read-access on Index 7100h the value of the A/D-converter (field value) can be requested. This is a raw value, without any linearization / computation. This object is read-only and has the following structure:

Index	Sub-Index	Parameter	Access
7100h	0	Number of Inputs	ro
	1	A/D-value for Input 1	ro

ro - Read Only, r / w - Read / Write

A read-access to this index (refer to “Physical Unit” on page 56) will cause the following answer:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	00h	71h	01h	LBh	HBh	00	00

Byte 4 holds the number of the input channel, i.e. 1 in this example. In byte 4 and 5 the value of the A/D-converter is given in Intel-Format (Low-Byte first).

8.5.8 Scaling first field value

Change of scaling

By changing the values of objects 7120h to 7124h the Process values for the device can be scaled. The scaling will then be calculated by the following routine:
(Shortcuts: Process Value = PV, Field Value = FV)

$$\text{PV (7130h)} = \text{Offset (7124h)} + \frac{\text{2nd PV (7123h)} - \text{1st PV (7121h)}}{\text{2nd FV (7122h)} - \text{1st FV (7120h)}}$$

On selecting parameters from objects 7120h to 7124h to 0 there will be no scaling to the process value. It is possible to only select an offset to the process value by changing object 7124h.

Attention !

A change of these parameters will not be stored inside the EEPROM automatically. This has to be done manually by the user (see “Store All Parameters” on page 49).

Index 7120h

Index 7120h has read and write access. By selecting this index the first field value can be changed. The index has the following structure:.

Index	Sub-Index	Parameter	Access
7120h	0	Number of channels	ro
	1	Field value (16 bit)	r / w

ro - Read Only, r / w - Read / Write

On a read acces to this object, the module will answer with the following message:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	20h	71h	01h	LBh	HBh	00	00

Byte 5+6 contains the field value data as 16bit.

8.5.9 Scaling first process value

Index 7121h

Index 7121h has read and write access. By selecting this index the first process value can be changed. The process value always has the actual physical unit number of decimal digits. The index has the following structure:

Index	Sub-Index	Parameter	Access
7121h	0	Number of channels	ro
	1	Process value (16 bit)	r / w

ro - Read Only, r / w - Read / Write

On a read acces to this object, the module will answer with the following message:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	21h	71h	01h	LBh	HBh	00	00

Byte 5+6 contains the process value data as 16bit.

8.5.10 Scaling second field value

Index 7122h

Index 7122h has read and write access. By selecting this index the second field value can be changed. The index has the following structure:.

Index	Sub-Index	Parameter	Access
7122h	0	Number of channels	ro
	1	Field value (16 bit)	r / w

ro - Read Only, r / w - Read / Write

On a read acces to this object, the module will answer with the following message:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	22h	71h	01h	LBh	HBh	00	00

Byte 5+6 contains the field value data as 16bit.

8.5.11 Scaling second process value

Index 7123h

Index 7123h has read and write access. By selecting this index the second process value can be changed. The process value always has the actual physical unit and number of decimal digits. The index has the following structure:

Index	Sub-Index	Parameter	Access
7123h	0	Number of channels	ro
	1	Process value (16 bit)	r / w

ro - Read Only, r / w - Read / Write

On a read acces to this object, the module will answer with the following message:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	23h	71h	01h	LBh	HBh	00	00

Byte 5+6 contains the process value data as 16bit.

8.5.12 Offset

Index 7124h

Index 7124h has read and write access. By selecting this index the offset to the process value can be changed. The offset always has the actual physical unit and number of decimal digits. The index has the following structure:

Index	Sub-Index	Parameter	Access
7123h	0	Number of channels	ro
	1	Offset (16 bit)	r / w

ro - Read Only, r / w - Read / Write

On a read acces to this object, the module will answer with the following message:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	23h	71h	01h	LBh	HBh	00	00

Byte 5+6 contains the offset data as 16bit.

8.5.13 Process Value

Index 7130h

By a read-access on Index 7130h the process value can be requested. This is a linearized value (degree Celsius). This object is read-only and has the following structure:

Index	Sub-Index	Parameter	Access
7130h	0	Number of Inputs	ro
	1	Process Value for Input 1	ro

ro - Read Only, r / w - Read / Write

A read-access to this index (refer to “Physical Unit” on page 56) will cause the following answer:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	42h	00h	71h	01h	LBh	HBh	00	00

Byte 4 holds the number of the input channel, i.e. 1 in this example. In byte 4 and 5 the process value given in Intel-Format (Low-Byte first, 16-Bit Signed Integer).

Attention !

In case of an input signal error the value will be $0xEEEE_{hex}$. At the same time the objects 1001h (“Error-Register” on page 46) and 6150h (“Input State” on page 58) will hold an appropriate error code. Also an emergency message will be sent by the device in that case (refer to “Emergency Message” on page 73).

8.6 Heart Beat Protocol

On switching on the Heart Beat protocol any other node on the CAN-bus can survey the sending node (Heart Beat producer). This technique is used to monitor the active nodes on the bus for safety reasons. For example nodes which send data autonomiosly only every 2 minutes, it can be better for the master to survey the node in the meantime.

Heart Beat ID

The Identifier for the Heart Beat producer is set to 700h + module address. The ID can not be changed.

The delay for sending the Haert Beat message is selected under Index 1014h.

8.6.1 Producer Time

Index 1014h

Within this Index the timer for the transmission of Heart Beat messages can be changed. The Index has the following structure:

Index	***Parameter***	***Access***
1014h	Timer in ms (16 bit)	r / w

ro - Read Only, r / w - Read / Write

By default the timer is set to 0. When changing the timer to a value greater 5ms the module will begin to send Heart Beat messages autonomiosly.

Note

The timer is a ms-timer with multiples of 5ms. Any value unequal to a multiple of 5 written to the module will be rounded. The value is a 16-bit value. If for example written to the module a value of 112(ms), this value will be changed to 110(ms).

Example:

The Heart Beat timer has to be changed to 1s. The ID for configuration has to look like this.

Write parameter, Module-ID = 2, Index 1014h

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1538	8	22h	14h	10h	E8h	03h	00	00	00

Data bytes 4+5 contain the value 03E8h which represents a producer time of 1000ms.

The answer you will receive from the module is:

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
1410	8	60h	0Ch	10h	00	00	00	00	00

Attention !

A change of these parameters will not be stored inside the EE-PROM automatically. This has to be done manually by the user (see “Store All Parameters” on page 49).

8.7 PDO-Communication

The PDO (Process Data Objects) communication service is a method to receive all temperature values from the node in one CAN message. The µCAN.1.ti uses a fixed PDO-Mapping, i.e. the contents of the PDO cannot be modified. The PDO is requested via a SNYC-message.

Note

PDO communication with the CAN-module is only possible in Operational Mode.

8.7.1 Transmit PDO 1

Index 1800h

The object 1800h defines communication parameters for the first transmit PDO. The PDO can only be used in Operational Mode.

The µCAN.1.ti uses a fixed PDO-Mapping. With this first PDO the temperature values (degree Celsius) can be requested.

Attention !

The PDO may not be requested faster than 20ms by the SYNC-service. It is recommended to request the PDO not faster than 40ms, because new temperature values for the input is available every 40ms.

The object has the following structure:

Index	Sub-Index	Parameter	Access
1800h	0	Largest Sub-Index supported	ro
	1	Identifier of PDO 1 (COB-ID)	r / w
	2	Transmission Type	r / w
	3	reserved	
	4	reserved	
	5	Timer (16 bit)	r/w

ro - Read Only, r / w - Read / Write

The value on Sub-Index 1 defines the identifier that is used for PDO 1. The 32-bit value has the following structure:

Bit 31	Bit 30 - 11	Bit 10 - 0
0 / 1	0	ID 11 bit

The default identifier is 180h + module-address.

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.

In the default setting the PDO is active (Bit 31 = 0).

8

PDO-Communication

Transmission Type

The transmission type (Sub-Index 2) defines the transmission charactar of the PDO. The µCAN.1.ti three different types of transmission.

By writing to Sub-Index 02 the transmission type can be selected. The following types are supported.

Transmission Type	***Description***
0	azyclic synchron The module sends on every SYNC-Message
1 - 240 d	zyclic synchron The module sends on every n-th SYNC-Message (with n = 1 .. 240)
254 d	Manufacturer specific The module sends autonomiosly every x ms. The value for x can be selected in Sub-Index 05.

If selected Transmission Type 254d (manufacturer specific) and the value for transmission delay (Sub-Index 05) is greater 5ms, the module will begin to transmitt PDOs after receicing a "Start Node"-command from the master.

Timer

As mentioned before, the transmission timer for PDOs can be selected under Sub-Index 05. The timer is a ms-timer with multiples of 5ms. Any value unequal to a multiple of 5 written to the module will be rounded. The value is a 16-bit value. If for example written to the module a value of 112(ms), this value will be changed to 110(ms).

Only in Operational Mode the module will send PDOs with timer delay.

Attention !

A change of these parameters will not be stored inside the EE-PROM automatically. This has to be done manually by the user (see “Store All Parameters” on page 49).

8.7.2 Transmit PDO 2

Index 1801h

The object 1801h defines communication parameters for the second transmit PDO. The PDO can only be used in Operational Mode.

The µCAN.1.ti uses a fixed PDO-Mapping. With this second PDO the A/D-converter values (raw values) can be requested.

Attention !

The PDO may not be requested faster than 20ms by the SYNC-service. It is recommended to request the PDO not faster than 40ms, because new temperature values for the input is available every 40ms.

The object has the following structure:

Index	Sub-Index	Parameter	Access
1801h	0	Largest Sub-Index supported	ro
	1	Identifier of PDO 2 (COB-ID)	r / w
	2	Transmission Type	r / w
	3	reserved	
	4	reserved	
	5	Timer (16 bit)	r/w

ro - Read Only, r / w - Read / Write

The value on Sub-Index 1 defines the identifier that is used for PDO 1. The 32-bit value has the following structure:

Bit 31	Bit 30 - 11	Bit 10 - 0
0 / 1	0	ID 11 bit

The default identifier is 280h + module-address.

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.

In the default setting of the PDO is **not active** (Bit 31 = 1).

PDO-Communication

Transmission Type

The transmission type (Sub-Index 2) defines the transmission charactar of the PDO. The µCAN.1.ti three different types of transmission.

By writing to Sub-Index 02 the transmission type can be selected. The following types are supported.

Transmission Type	***Description***
0	azyclic synchron The module sends on every SYNC-Message
1 - 240 d	zyclic synchron The module sends on every n-th SYNC-Message (with n = 1 .. 240)
254 d	Manufacturer specific The module sends autonomiosly every x ms. The value for x can be selected in Sub-Index 05.

If selected Transmission Type 254d (manufacturer specific) and the value for transmission delay (Sub-Index 05) is greater 5ms, the module will begin to transmitt PDOs after receicing a "Start Node"-command from the master.

Timer

As mentioned before, the transmission timer for PDOs can be selected under Sub-Index 05. The timer is a ms-timer with multiples of 5ms. Any value unequal to a multiple of 5 written to the module will be rounded. The value is a 16-bit value. If for example written to the module a value of 112(ms), this value will be changed to 110(ms).

Only in Operational Mode the module will send PDOs with timer delay.

Attention !

A change of these parameters will not be stored inside the EEPROM automatically. This has to be done manually by the user (see “Store All Parameters” on page 49).

8.7.3 Mapping Parameter

Index 1A00h

The object 1A00h defines the PDO mapping for the first PDO. The µCAN.1.ti uses a fixed PDO-Mapping. The first PDO holds the temperature values (degree Celsius).

The CAN message for the first PDO has the following structure:

Transmit-PDO 1: **Process Values**

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
PDO1 ID	8	HB Input 1	LB Input 1						

Index 1A01h

The object 1A01h defines the PDO mapping for the second PDO. The µCAN.1.ti uses a fixed PDO-Mapping. The second PDO holds the data from the A/D-converter (raw data).

The CAN message for the first PDO has the following structure:

Transmit-PDO 2: **Field Values**

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
PDO2 ID	8	HB Input 1	LB Input 1						

8.7.4 Synchronisation Message

Index 1005h

The object 1005h defines the identifier for the SYNC-message. On reception of a message with this identifier the transmission of PDOs is initiated (refer to “PDO-Communication” on page 66).

The object has the following structure:

Index	Parameter	Access
1005h	ID (32 bit)	r / w

ro - Read Only, r / w - Read / Write

The 32-bit value has the following structure:

Bit 31	Bit 30 - 11	Bit 10 - 0
1	0	ID 11 bit

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

The most significant bit (Bit 31) is always set to 1 (device consumes SYNC-message, refer to CANopen DS-301).

Attention !

A change of this parameter will not be stored inside the EEPROM automatically. This has to be done manually by the user (see “Store All Parameters” on page 49).

8.8 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.

Note

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

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

ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
	8	Error Code		00h	Manufacturer Specific Error Field				

Supported Error Codes:

Error Code	Meaning
00 00h	No Error
FF 00h	Device Specific Error
81 00h	Communication Error

Device Specific Error

Pt100: fraction of sensor and short-circuit of sensor
Thermo couple: short-circuit of sensor

Byte 4 (first byte of the Manufacturer Specific Error Field) holds the information about the specific error type.

b7	b6	b5	b4	b3	b2	b1	b0
			Input 1				Input 1
Short-Circuit of Sensor				Fraction of Sensor			

Communication Error

The µCAN.1.ti supports the generation of communication errors. For this error type the internal state of the CAN controller (Infineon C505) is monitored.

Emergency Message

Byte 4 (first byte of the Manufacturer Specific Error Field) holds the value $0xBE_{hex}$ in case of a CAN-bus error. The value of byte 5 defines the error type.

Description of byte 5:

b7	b6	b5	b4	b3	b2	b1	b0
Status				Error Code			

Status:

Bus Off Status

b7 - Bit is set
The CAN controller is in bus-off state because the error counters reached the maximum value.

Warning Status

b6 - Bit is set
The CAN controller is in warning status.

Error Code:

0h - **No error**

1h - **Stuff Error**
More than 5 consecutive bits with the same logical value.

2h - **Form Error**
Wrong format of the CAN message.

3h - **Acknowledgement Error**
No dominant bit inside the acknowledgement slot (no further node on the bus).

4h - **Bit 1 Error**
Node tried to send a 1, but a 0 was received.

5h - **Bit 0 Error**
Node tried to send a 0, but a 1 was received.

6h - **CRC Error**
The CRC-code of the received message was wrong.

7h -**not used**

8

8.8.1 Changing of Emergency ID

Index 1014h

By accessing this object the ID of the Emergency message can be changed. The object has the following structure:

Index	***Parameter***	***Access***
1014h	ID (32 bit)	r / w

ro - Read Only, r / w - Read / Write

The 32-bit value contains the following settings:

Bit 31/30	***Bit 29***	***Bit 28 - 11***	***Bit 10 - 0***
res.	0	0	ID 11 bit

By default the ID is set to 80h + module address. This ID reserves high priority for the emergency messages.

Attention !

A change of this parameter will not be stored inside the EEPROM automatically. This has to be done manually by the user (see “Store All Parameters” on page 49).

9. Appendix

9.1 Technical Data µCAN.1.ti-IP65

Technical Data

Power Supply

Supply Voltage	7 V DC - 60 V DC, reverse current protected
Power Consumption	typ. 350mW, max. 600mW (can be reduced by software to 125mW)
Optical isolation	Isolation between supply voltage and module electronic 500 VDC, Isolation between analogue input and supply voltage 500 VDC
Physical Interface	Terminal Block (2,5 mm^2)

CAN-Interface

Baudrates	10 kBit/s to 1 MBit/s (according to CiA recommendation)
Maximum number of modules on the bus	127
Status on the bus	active node
Protocol	CANopen, DSP-404, other protocols on request
Specification	2.0A and 2.0B
Physical Interface	Terminal Block (2,5 mm^2)

Environment

Operating Temperature	-40°C ... +85°C (-40°C ... +110°C on request)
Storage Temperature	-50°C ... +110°C
Humidity	15-95%

Diagnosis / bi-color LED

Operation / CAN-Communication	LED on (green), LED blinking (green)

9

Technical Data µCAN.1.ti-IP65

Error	LED blinking (red) 1 Hz
General	
Dimensions	64 x 98 x 34 mm
Weigth	ca. 280g
Sensor-Input	
Sensor Types	Thermo couple type J,K,L (other types on request) Pt100, 4-wires, fully compensated Strain gauge, 350 Ohm, full bridge Pressure transducer
Sensor supply	5V DC, 10V DC (Option) 0,8mA
Standard Signal Types	0-10 V 0-20mA, 4-20mA
Physical Interface	Terminal Block (2,5 mm^2)
Resolution	16 bit for all types of signals, conversion times shorter than 5ms (200Hz) are converted with 12 bit
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,

9.2 Calculation table dezimal in hexadezimal

dez	hex	dez	hex	dez	hex	dez	hex
0	00	16	10	32	20	48	30
1	01	17	11	33	21	49	31
2	02	18	12	34	22	50	32
3	03	19	13	35	23	51	33
4	04	20	14	36	24	52	34
5	05	21	15	37	25	53	35
6	06	22	16	38	26	54	36
7	07	23	17	39	27	55	37
8	08	24	18	40	28	56	38
9	09	25	19	41	29	57	39
10	0A	26	1A	42	2A	58	3A
11	0B	27	1B	43	2B	59	3B
12	0C	28	1C	44	2C	60	3C
13	0D	29	1D	45	2D	61	3D
14	0E	30	1E	46	2E	62	3E
15	0F	31	1F	47	2F	63	3F

Appendix

Calculation table dezimal in hexadezimal

dez	*hex*	*dez*	*hex*	*dez*	*hex*	*dez*	*hex*
64	00	80	50	96	60	112	70
65	41	81	51	97	61	113	71
66	42	82	52	98	62	114	72
67	43	83	53	99	63	115	73
68	44	84	54	100	64	116	74
69	45	85	55	101	65	117	75
70	46	86	56	102	66	118	76
71	47	87	57	103	67	119	77
72	48	88	58	104	68	120	78
73	49	89	59	105	69	121	79
74	4A	90	5A	106	6A	122	7A
75	4B	91	5B	107	6B	123	7B
76	4C	92	5C	108	6C	124	7C
77	4D	93	5D	109	6D	125	7D
78	4E	94	5E	110	6E	126	7E
79	4F	95	5F	111	6F	127	7F

9.3 EMI Certificate

ELEKLUFT GmbH
EMV-Zentrum
Justus-von-Liebig-Straße 18
D-53121 Bonn
Tel: int 49 (0)228 / 6681 - 558
Fax: int 49 (0)228 / 6681 - 792

Prüfberichtnummer/Report No.: 0050/99
Seite/Page: 1/23

Prüfbericht über die Störaussendung und -beeinflussung elektronischer Geräte
Report on the Electromagnetic Emission and Immunity of electronic equipment

Prüfvorschriften: Test Specifications:	EN 50081-2:1993 Teile/Parts: EN 55011 EN 50082-2:1995 Teile/Parts: EN 61000-4-2, EN 61000-4-3, ENV 50204, EN 61000-4-4, EN 61000-4-6
Auftraggeber: Customer:	MicroControl GmbH & Co. KG Josef-Kitz-Str. 9 53840 Troisdorf
Prüfgegenstand: Equipment tested:	µCAN.1.ti/ai S/N: 070699023
Eingangsdatum: Incoming Date:	26.07.1999
Prüfende Abteilung: Testing Department:	S/E EMV-Zentrum S/E EMC-Centre
Prüfer: Test Engineer:	Reß Gierlach
Prüfort: Test Location:	Bonn
Prüfdatum: Date of Test:	26.07.1999
Bemerkungen: Remarks:	keine none
Prüfergebnis: Test Result:	Bestanden Approved

EMV-Zentrum
EMC-Centre

Qualitätssicherung
Quality Assurance

Das Prüfergebnis bezieht sich ausschließlich auf den oben genannten Prüfgegenstand und nicht auf das Verhalten der Serie. Vervielfältigungen, auch auszugsweise, nur mit schriftlicher Genehmigung des Prüflabors. Test results relate only to the item tested and described hereafter. This report shall not be reproduced completely or in parts without the prior written approval of the test laboratory.

9