# CO4031A-FL

## Single Chip CANopen Controller for Keyboard + LED + LCD-Display

## General Description

The CO4031 is a low cost, high performance Single Chip solution for CANopen Keyboards, LED-Panels and LCD-Display modules. It is especially designed for automotive and industrial applications. The device offers the complex implementation of the CANopen standards DS301 and DS401 in a single chip. The Matrix-Keyboard is processed as digital input lines. Additionally a special object describing a key status is implemented.

The LEDs are set from a digital output object, they are accessed as LED-Array. For interfacing an alphanumeric LCD module special data objects are implemented.

The Chip is suitable for simple CANopen HMI interfaces and operator panels with matrix keyboards up to 64 keys, up to 32 status LEDs arranged in a LED matrix and an alphanumeric LCD module. LCD size can be up to 4 lines with 20 characters per line. The chip can address nearly all modules based on the Hitachi HD44780 or compatible LCD controllers.

The CO4031 Chip requires only few external components, just like a crystal, a CAN transceiver and capacitors.

CO4031 is a software solution to run on Fujitsu MB90F497 micro controller. It is offered either as software runtime licence or as ready programmed chip (runtime licence included).

## Features

- **Single Chip CANopen Controller**
- **According to CiA Draft Standards DS301 Version 4.0 and DS401 Version 2.0**
- **Additional Objects for Matrix Keyboard, LEDs and LCD module**
- **Debounce logic for Keyboard**
- **Temperature ranges up to –40 to 105 °C**
- **Package QFP64**

## Applications

- **CANopen Operator panels**
- **Information terminals**
- **Industrial HMI for automation control**
- **CANopen Interface for keyboards and control panels**


## CANopen Features

- **2 Transmit- and 2 Receive PDOs**
- **Dynamic PDO mapping**
- **Variable PDO identifier**
- **All CANopen specific PDO transmission types supported: synchronous, asynchronous, event driven, cyclic, acyclic and remote frame dependent.**
- **Event timer and inhibit timer features for all transmit PDOs.**
- **Storing and restoring of object dictionary to non-volatile memory**
- **Node guarding, life guarding, heartbeat**
- **Baud rate up to 1MBit**
- **Emergency messages**
- **Minimum boot up**

## Ordering Information

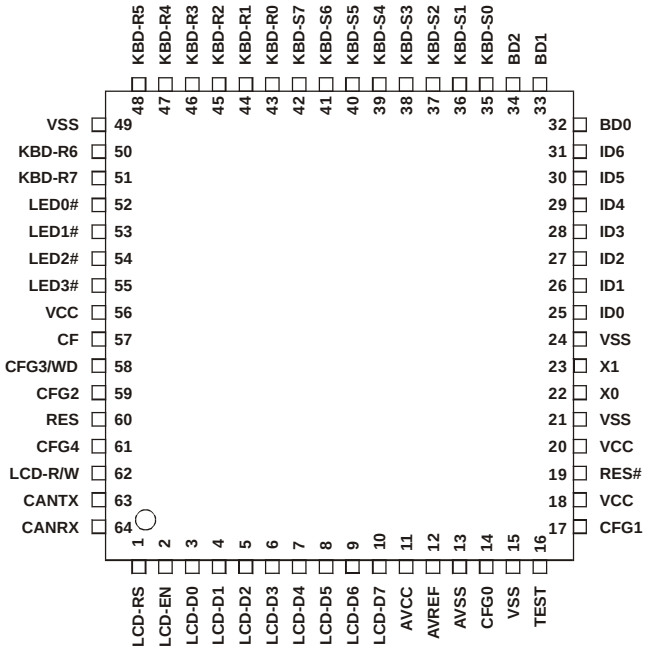
**CO4031 Chip (programmed, licence included)**

| Part | Temp. Range | Package |
|---|---|---|
| CO4031A-FL | -40 °C to 85 °C | QFP64 |
| CO4031AE-FL | -40 °C to 105 °C | QFP64 |

**Software licence**

| Part | Description |
|---|---|
| CO4031SRL-F | Software runtime licence for Fujitsu MB90F497 controller |

## Pin Assignment




## Pin Listing

| *Pin No.* | *Pin Name* | *Funktion* |
|---|---|---|
| 1 | LCD-RS | LCD-Interface.<br>Register Select Line |
| 2 | LCD-EN | LCD-Interface<br>Display Enable |
| 3 .. 10 | LCD-D0 ..<br>LCD-D7 | LCD-Interface<br>8 Bit Data Bus to LCD module |
| 11 | AVCC | Supply for A/D-Converter |
| 12 | AVREF | Reference Voltage for A/D-Converter |
| 13 | AVSS | Ground for A/D-Converter |
| 14, 17, 59, 58, 61 | CFG0 to CFG4 | Configuration bits 0 to 4 |
| 16 | TEST | Do not connect this pin |
| 19 | RES# | Reset input active low |
| 22 | X0 | Crystal oscillator input |
| 23 | X1 | Crystal oscillator output |
| 25*, 26*, 27 to 31 | ID0*, ID1*<br>ID2 to ID6 | Identifier selection input |
| 32 to 34 | BD0 to BD2 | Baud rate selection input |

## Pin Listing continued

| *Pin No.* | *Pin Name* | *Funktion* |
|---|---|---|
| 35 .. 42 | KBD-S0 ..<br>KBD-S7 | LED-/Keyboard interface<br>Scan output lines for selecting matrix keyboard line<br>Column for LED matrix |
| 43 .. 48, 50, 51 | KBD-R0 ..<br>KBD-R7 | Keyboard interface<br>Return input lines for reading keyboard column |
| 52 .. 55 | LED0# ..<br>LED3# | LED line output |
| 57 | CF | Filter capacitor |
| 60* | BUZ# | Buzzer output (active low) |
| 62* | LCD-RW | LCD-Interface<br>Read write control output line |
| 63 | CANTX | Transmitter output of CAN module |
| 64 | CANRX | Receiver input of CAN module |
| 15, 21*, 24, 49 | VSS | Ground |
| 18*, 20, 56 | VCC | Power supply |

*** Pins are used for in circuit programming of Fujitsu MB90F497. See "in circuit programming manual" for further information**.

## Pin Description

All input / output pins are high impedance during reset. The CO4031 does not support internal pull up/down resistors. Use external components where needed. All input pins have Schmitt trigger characteristics. See chapter “Typical Applications” for additional information.

## LCD-Interface Pins

The LCD interface allows direct drive of a alpha-numeric LCD module based on HD44780 or compatible LCD controllers. Nearly all modules are based on this standard.

**LCD-D0 to LCD-D7: Data bus of LCD module**

Bi-directional data bus to the LCD module. Connect this port directly to the LCD modules data port.

**LCD-RS: LCD Register Select Output**

The Register Select output line selects whether the actual data bus cycle addresses either the display RAM or the command register of the LCD module. LCD-RS at LOW level addresses the command register, while LCD-RS at HIGH level addresses the display data RAM. Connect this port directly to pin RS of the LCD module.

**LCD-R/W: LCD Read / Write Control Output**

The Read / Write control output line selects whether the actual data bus cycle is a read or a write cycle. LCD-R/W at LOW level indicates a write cycle while LCD-R/W at HIGH level indicates a read cycle. Connect this port directly to pin R/W of the LCD module.

**LCD-EN: LCD Enable Control Output**

The LCD Enable control output line controls a data bus cycle. During access to the LCD module, the CO4031 forces this pin to HIGH level. Connect this port directly to pin EN of the LCD module.

## LED-/Keyboard-Interface Pins

The LED-/Keyboard interface allows direct connection of a 8x8 matrix keyboard.
A maximum of 32 LEDs can be driven from the CO4031s LED controller. The LEDs must be arranged in an array of 4 lines (with common anode) and 8 columns. The LED lines will be multiplexed within the keyboard scan cycle. For interfacing a LED array of max 4x8 LEDs additional driver circuits must be used.
The LED-/Keyboard-Interface uses two different cycles to scan the keyboard and set the LEDs. This two cycles are multiplexed from the CO4031. They use the common lines KBD-S0 to KBD-S7
See also typical Application for further information.

**KBD-S0 to KBD-S7 : Scan Lines**

The keyboard scan lines output the Keyboard line address during Keyboard Scan Cycle. A low level is driven to the output driver to select a keyboard line. All other output lines stay at high impedance during this time. So the Keyboard-Scan-Cycle scans one keyboard line after the other by pulling one output line to low level and then reading the columns of the keyboard using the keyboard return lines KBD-R0 to KBD-R7.
During LED cycle the KBD-S0 to KBD-S7 output drivers output the LED information for the LED column selected by the LED line outputs LED0# to LED3#. For switching a LED on, the appropriated scan line is forced to low level. The output drivers for LEDs that must be switched off are held in tristated mode, so it is strongly recommended, to use pull up resistors of approx 4.7 kOhms at this port.
See also typical Application for further information.

**KBD-R0 to KBD-R7 : Keyboard Return Lines**

The keyboard return input lines are used to read the key column values for the line selected with KBD-Sx. A pressed key must force this line to LOW level. Use pull up resistors of approx 4.7kOhms at this port to meat this requirements.

**LED0# to LED3# : LED Line Select**

This LED line select output lines select the LED line of the LED array, that will be controlled by the actual cycle. The output drivers are set to LOW level to select a line, so PFET transistors may be used for this purposes.

## Configuration / Supply Pins

**CF: Filter Capacitor input**

For correct operation of the CO4031 a ceramic capacitor of 100 nF or 220 nF must be connected between Pin 57 (CF) and VSS. Place this component as close as possible to the CO4031.

**X0, X1: Crystal oscillator input**

Connect a crystal of 4 MHz between X0 and X1. Use additional ceramic capacitors of 22 pF between X0 and X1 to VSS.

**TEST: Reserved pin**

Leave pin TEST unconnected.

**VCC, AVCC, VSS, AVSS, AVREF: Supply Pins**

Make sure that all ground and power supply pins are connected to the same potential. Do not leave any ground or power supply pins open. Connect decoupling capacitors as close as possible to the device.
Because of the A/D converter is not used, connect the pins as follows: AVCC = VCC, AVSS = AVREF = VSS.
See chapter “Recommended Operation Conditions” for details

**RES#: Reset input pin**

For a correct device reset, provide an active low reset signal according to recommended operation conditions to input RES#.

**CFG0 to CFG4: Configuration input pins**

Set device configuration to preferred operation mode using CFG0 to CFG4. Leaving any configuration input unconnected may cause malfunction of the device. It is strongly recommended forcing all configuration input bits to either high or low level by using external pull up/down resistors. Do not use direct connection to VCC or VSS.
The CFG pins have additional features:

CFG0: Standard indicator LED output.
Always blinking:
- 10 % Duty cycle indicates no error
- 50 % Duty cycle indicates uncritical error or warning
- 90 % Duty cycle indicates critical error

CFG1: CANopen Run-LED Draft Standard according to DRP303-3

CFG2: CANopen Error-LED Draft Standard according to DRP303-3

CFG3: Watchdog Output

***Note:*** CFGx pins are input pins during start up and output pins in normal operation mode. In order to prevent short circuit overload a series resistor between CFGx and configuration device (example DIP switch) should be used. See also "typical application" for details.

**CANRX, CANTX: CAN interface pins**

The CAN interface pins may be used for direct connection to CAN transceivers like the 80C251. For longer bus length or noisy or disturbed environments it is strongly recommended to use galvanic isolation with opto isolators between bus interface and CANopen application, to improve system reliability.

**BUZ#: Buzzer output**

Buzzer output. Digital output line to switch a buzzer on. (active low)

## Handling the Device

**Preventing latch up**

The CO4031 is a CMOS device and may suffer latch up under the following conditions:

1) A voltage higher than VCC or lower than VSS is applied to any pin.
2) Absolute maximum ratings are exceeded
3) AVCC power is provided before VCC supply

**Handling unused Pins**

Do not leave unused input pins open. This might cause malfunction of the device.

**Power Supply Pins**

Make sure that all ground and power supply pins are connected to the same potential. Do not leave any ground or power pins open. Connect at least two ceramic capacitors of 100 nF and a tantalum capacitor of 1 uF between VCC and VSS as close as possible to the device.

**Power Supply for A/D converter**

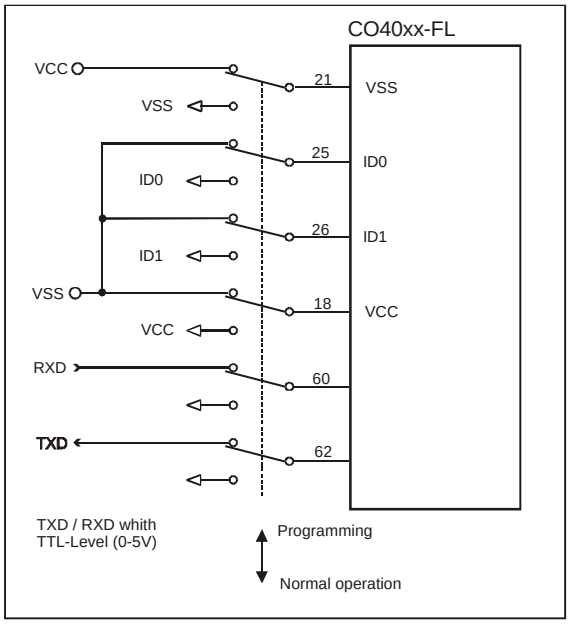
The power supply for the A/D converter must not be turned on before the power supply VCC.
If the A/D converter is not used, connect the pins as follows: AVCC = VCC, AVSS = AVREF = VSS.

**Pull up/down resistors**

The CO4031 does not support internal pull up/down resistors. Use external components where needed.

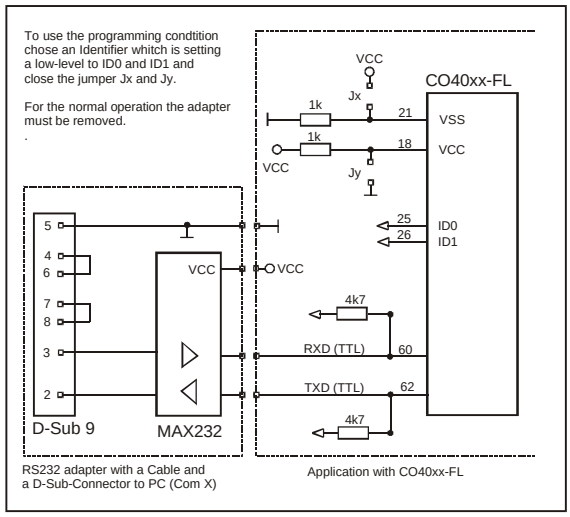
## *In Circuit programming of MB90F497*

If you want to enable in circuit programming of the Fujitsu MB90F497 micro controller for CO4031 software download or update, the following conditions must be met.




| Pin No. | Pin Name | Pin Setting for in circuit programming |
|---|---|---|
| 21 | VSS | Level = High |
| 18 | VCC | Level = Low |
| 25 | ID0 | Level = Low |
| 26 | ID1 | Level = Low |
| 60 | Res | RXD (TTL – logic level) |
| 62 | ILCD-R/W | TXD (TTL – logic level) |

Design example for programming adaptor.




To enable the programming condition chose an identifier that forces the Pins ID0 and ID1 to low level. For example identifiers ID= 3, 7, 0x0B, 0x0F … might be used. Close the jumpers Jx and Jy.

For optimised programming adapter design, the programming adapter should close the jumpers Jx and Jy directly by plugging the adaptor into the application board.

For normal operation mode the programming adapter must be removed.

## Device Configuration

The following sections describe the device configuration with meaning:
1: ViH logic high level
0: ViL logic low level

### CAN Identifier

The CAN Identifier will be set with Pins ID0 to ID6. This configuration pins use internal inverter. The ID is set as follows:

| ID6 | ID5 | ID4 | ID3 | ID2 | ID1 | ID0 | CAN-Identifier |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | Programmable ID |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 = 0x01 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 2 = 0x02 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 3 = 0x03 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 | 4 = 0x04 |
| | | | …. | | | | …. |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 63 = 0x3F |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 64 = 0x40 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 | 65 = 0x41 |
| | | | …. | | | | …. |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 125 = 0x7C |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 126 = 0x7E |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 127 = 0x7F |

All Identifiers from 1 to 127 are valid settings. Identifier 0 is used to load the ID from object 2100.

### Baud rate

The baud rate configuration will be done with configuration inputs BD0 to BD2

| BD2 | BD1 | BD0 | CAN-Baud Rate / Bus length | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | Mbit/sec | 25 m *1) |
| 1 | 1 | 0 | 800 | kbit/sec | 50 m *1) |
| 1 | 0 | 1 | 500 | kbit/sec | 100 m *2) |
| 1 | 0 | 0 | 250 | kbit/sec | 250 m *2) |
| 0 | 1 | 1 | 125 | kbit/sec | 500 m *3) |
| 0 | 1 | 0 | 50 | kbit/sec | 1000 m *3) |
| 0 | 0 | 1 | 20 | kbit/sec | 2500 m *3) |
| 0 | 0 | 0 | 10 | kbit/sec | 5000 m *3) |

*1) Calculation without optocouplers.
For optocouplers bus length is reduced for about 4m per 10 nsec propagation delay of employed optocoupler type
*2) Calculation with 40 nsec optocoupler propagation delay
*3) Calculation with 100 nsec optocoupler propagation delay
The calculation of the bus length is based on a line propagation delay of 5 nsec/m.

### LCD-Configuration

With configuration bits CFG0 and CFG1 the number of LCD lines that are supported from by the LCD module is set.

| CFG 1 | CFG 0 | Nr of LCD-lines |
|---|---|---|
| 1 | 1 | No LCD module present |
| 1 | 0 | 1 |
| 0 | 1 | 2 |
| 0 | 0 | 4 |

If there is no LCD interface present, the CO4031 controller disables the corresponding Objects and disables the PDO by setting the PDO-ID to 8000 0000h.

With configuration bits CFG2 and CFG3 the number of LCD characters per line that are supported from by the LCD module are set.

| CFG 3 | CFG 2 | Characters per line |
|---|---|---|
| 1 | 1 | 8 |
| 1 | 0 | 16 |
| 0 | 1 | 20 |
| 0 | 0 | 40 |

The CO4031 uses exact setting of this bits in order to enable correct calculation of the display RAM addresses.

CFG4 is reserved for future use and should be held at high level to keep compatibility to future versions.

***Note:*** CFGx pins are input pins during start up and output pins in normal operation mode. In order to prevent short circuit overload a series resistor between CFGx and configuration device (example DIP switch) should be used. See also "typical application" for details.

## Object Dictionary

The CO4031 Single Chip CANopen Controller implements a complex object dictionary for CANopen I/O devices.

*For detailed information about CANopen objects see additional brochure "Introduction to CANopen"*

For the Object tables all values are shown in hexa-decimal way.
For access type the following settings are valid

ro read only
wo write only
rw read and write access enabled

**DS301: global Objects**

| Index | Sub-Index | Name | Acc. |
|---|---|---|---|
| 0005 | - | Dummy 8 | wo |
| 0006 | - | Dummy 16 | wo |
| 0007 | - | Dummy 32 | wo |
| 1000 | - | Device Type | ro |
| 1001 | - | Error Register | ro |
| 1002 | - | Manufacturer Status Register | ro |
| 1005 | - | COB-ID Sync<br>Identifier Sync Object | rw |
| 1008 | - | Device Name | ro |
| 1009 | - | Hardware Version | ro |
| 100A | - | Software Version | |
| 100B | - | Node Id *4) | - |
| 100C | - | Guard Time | rw |
| 100D | - | Life Time Factor | rw |
| 100E | - | COB-ID Guard *4) | - |
| 1010 | - | Store Parameters *1) | wo |
| 1011 | - | Reload Default Parameter *1) | wo |
| 1014 | - | COB ID Emergency | rw |
| 1015 | - | Inhibit Time Emergency | rw |
| 1017 | - | Producer Heartbeat Time | rw |
| 1018 | 0 | Identity Object | ro |
| | 1 | Vendor ID | ro |
| | 2 | Product Code | ro |
| | 3 | Revision Number | ro |
| | 4 | Serial Number | ro |
| 1029 | 0 | Error Behavior | ro |
| | 1 | In case of bus errors | rw |
| | 2 | In case of LCD errors | rw |
| 2000 | - | Device Manufacturer *2) | ro |
| 2100 | - | New Node Id *3) | rw |
| 2101 | - | System Configuration | ro |
| 2103 | - | Enable Guarding Warning | rw |
| 2110 | - | Enable Boot Up Message | rw |
| 2180 | - | CAN Restart Time | rw |

Notes:

*1) This object cannot be written to in operational device state.
Only use this command in preoperational device state, otherwise the CO4031 will answer requests with SDO abort telegrams.

*3) The New Node Id object gives the possibility to set a Node-Nr independent from the Node-Id Input Bits. With this feature the device may be configured by CAN bus line.

*4) This objects are implemented but not accessible from CAN bus line. Description is added in order to give an easier understanding of the CO4031s functionality.

The data type entries Index 0005 to 0007 are implemented for compatibility reasons. They may be mapped to PDOs in order to define the appropriate space in the PDO.

For the read only objects following data is set:

| Index | Sub. | Name | Value in Hex. |
|---|---|---|---|
| 1000 | | Device Type | 0003 0191 h |
| 1018 | 0 | Identity Object | 04h |
| | 1 | Vendor ID | 0000 0058 h |
| | 2 | Product Code | 0140 3101h |
| | 3 | Revision Number | 0 .. 0xFFFFFFFF |
| | 4 | Serial Number | 0 |
| 2101 | | System Configuration | Set according to the setting of the configuration input bits. |
| | | | |

**DS301: PDO Parameter Objects**

Description of PDO Parameter objects:
These Objects enable dynamic PDO mapping, variable identifier distribution for PDOs and setting of the transmission mode, inhibit and event times.
For the CO4031 setting of all parameters may be done in the device state "operational" as well as in "preoperational" state.

| Index | Sub-Index | Name | Acc. |
|---|---|---|---|
| 1400 | 0 | Receive PDO1 Communication Parameter | ro |
| | 1 | COB-ID | rw |
| | 2 | Transmission Type | rw |
| 1404 | | Receive PDO5 Communication Parameter same as 1400.00 .. 1400.02 | rw |
| 1600 | 0 | Receive PDO1: Parameter mapping | rw |
| | 1 to n | Mapped Object (max. 8 objects mappable) | rw |
| 1604 | | Receive PDO5 Parameter mapping Same as 1600 | rw |
| 1800 | 0 | Transmit PDO1 Communication Parameter | ro |
| | 1 | COB-ID | rw |
| | 2 | Transmission Type | rw |
| | 3 | Inhibit Time | rw |
| | 4 | Reserved | rw |
| | 5 | Event Timer | rw |
| 1804 | | Transmit PDO5 Communication Parameter same as 1800.00 .. 1800.05 | rw |
| 1A00 | 0 | Transmit PDO1 Parameter mapping | rw |
| | 1 to n | Mapped Object (max. 8 objects mappable) | rw |
| 1A04 | | Transmit PDO5 Parameter mapping Same as 1A00 | rw |

**DS401: Digital Input Objects**

| Index | Sub-Index | Name | Acc. |
|---|---|---|---|
| 4000 | | Key-Code | ro |
| 4001 | | Key Roll Over Time | rw |
| 4002 | | Key Roll Over Delay | rw |
| 4003 | | Key Roll Over Release Delay | rw |
| 4004 | | Keyboard debounce cycles | rw |
| 4005 | | Good Read Beep Time | rw |
| 4006 | | Key Roll Over Signal Bit | rw |
| 6000 | 0 to 8 | Read digital input 8 bit Pressed keys are interpreted as digital input lines | Ro |
| 6002 | 0 to 8 | Polarity input 8-bit | rw |
| 6005 | | Global interrupt enable | rw |
| 6006 | 0 to 8 | Interrupt mask: any change | rw |
| 6007 | 0 to 8 | Interrupt mask rising edge | rw |
| 6008 | 0 to 8 | Interrupt mask falling edge | rw |

**DS401: Digital Output Objects**

| Index | Sub-Index | Name | Acc. |
|---|---|---|---|
| 5200 | - | Reset Output on Error Option | rw |
| 6200 | 0 to 4 | LED Output 8 Bit | rw |
| 6202 | 0 to 4 | Change Polarity Output 8 bit | rw |
| 6206 | 0 to 4 | Error Mode Output | rw |
| 6207 | 0 to 4 | Error State Output | rw |

**DS401: LCD-Objects**

| Index | Sub-Index | Name | Acc. |
|---|---|---|---|
| 4200 | | LCD-Access | wo |
| 4201 | | Character 7 Beep Time | rw |

## Description of Object Dictionary

The following list gives da short description of all dictionary entries.

### Index 0005

This object is implemented to enable reservation of data space in PDOs by mapping dummy entries.

| | |
|---|---|
| Index | 0005 |
| Name | Dummy 8 |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | WO |
| PDO Mapping | Yes |
| Value Range | - |
| Default Value | 0 |

### Index 0006

This object is implemented to enable reservation of data space in PDOs by mapping dummy entries.

| | |
|---|---|
| Index | 0006 |
| Name | Dummy 16 |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | WO |
| PDO Mapping | Yes |
| Value Range | - |
| Default Value | 0 |

### Index 0007

This object is implemented to enable reservation of data space in PDOs by mapping dummy entries.

| | |
|---|---|
| Index | 0007 |
| Name | Dummy 32 |
| Description | - |
| Data Type | Unsigned 32 |
| Access modes | WO |
| PDO Mapping | Yes |
| Value Range | - |
| Default Value | 0 |

## DS301: Global Objects

### Index 1000 : Device Type

Description of the device type. The Object gives the CiA device profile number and additionally the functionality of the device.

| | |
|---|---|
| Index | 1000h |
| Name | Device Type |
| Description | - |
| Data Type | Unsigned 32 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | |

### Index 1001 : Error Register

This object holds an error of the device.

| | |
|---|---|
| Index | 1001h |
| Name | Error Register |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | Yes |
| Value Range | - |
| Default Value | - |

The error register has the following structure

| Bit | Meaning |
|---|---|
| 0 | Generic error. This bit is set, if any error is active |
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | CAN bus or communication error |
| 5 | 0 |
| 6 | 0 |
| 7 | Device Error |

### *Index 1002 : Status Register*

This object gives additional information for the device

| | |
|---|---|
| Index | 1002h |
| Name | Status Register |
| Description | - |
| Data Type | Unsigned 32 |
| Access modes | RO |
| PDO Mapping | Yes |
| Value Range | - |
| Default Value | - |

The status register bits have the following meaning

| Bit | Meaning |
|---|---|
| 31 | The device is in operational state |
| 30 | Node guarding error |
| | |
| | |
| | |

### *Index 1005 : COB-ID Sync*

Identifier of Can Object for the Synchronisation message. The CO4031 may only operate in Sync consumer mode. Generating of Sync messages is not possible.

| | |
|---|---|
| Index | 1005h |
| Name | COB-ID Sync |
| Description | - |
| Data Type | Unsigned 32 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | 1 .. 7FFh |
| Default Value | 80h |

### *Index 1008 : Device Name*

This object shows the name of the device as visible string.

| | |
|---|---|
| Index | 1008h |
| Name | Device Name |
| Description | - |
| Data Type | Visible String |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | The maximum string length is 20 characters |
| Default Value | CO4031A |

### *Index 1009 : Hardware Version*

This object shows the name of the device as visible string.

| | |
|---|---|
| Index | 1009h |
| Name | Hardware Version |
| Description | - |
| Data Type | Visible String |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | The maximum string length is 20 characters |
| Default Value | - |

### *Index 100A : Software Version*

This object shows the name of the device as visible string.

| | |
|---|---|
| Index | 100Ah |
| Name | Software Version |
| Description | - |
| Data Type | Visible String |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | The maximum string length is 20 characters |
| Default Value | - |

## Index 100B : Node-ID

This object keeps the actual node Id. The Object is not represented in the object dictionary because of standard conforming reasons.

| | |
|---|---|
| Index | 100Bh |
| Name | Node ID |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | Not accessible |
| PDO Mapping | No |
| Value Range | 1 to 127 |
| Default Value | See below |

There are several modes to select a valid node ID.

**Hardware-ID > 0**
Setting the configuration input bits ID0 to ID6 to any combination other than zero will take the configuration input setting of bits ID0 to ID6 as valid Node-ID. The IDs for Emergency, Guarding and SDO transfer will be set to their default values according to draft standard.
**Behaviour at next start up if:**
Node-ID is changed
If the Node-ID is changed against the saved value (stored in Object 100B) The PDO IDs and the PDO-Mapping will be set to default values. This mode can be used to save only DS401 related objects
Node-ID is unchanged
If the Node-ID is still the same as saved in object 100B the PDO IDs keep the values they had during execution of the command “save dictionary”

**Hardware-ID = 0**
If setting the configuration input bits ID0 to ID6 to the combination zero (all input bits at high level) will activate the Node-ID object 2100h. With this object a Node-ID may be set independent from the configuration input bits. If there is no saved dictionary available, the node ID is set to 126.
In this case the IDs for Emergency, Guarding and SDO transfer will be set to their default values according to the node ID saved to object 2100. The PDO IDs keep the values they had during execution of the command “save dictionary”
If Object 2100h has value 0 (default factory setting if not written to by your application or CANopen configuration tool) the Node ID will be forced to 126 (7Eh).
Note! If using Hardware-ID = 0, the PDO-IDs must be set to their deired values before saving the dictionary. Otherwise the PDO-Ids will refere to node ID 126, which is the default ID when setting ID-Input Pins to Node-ID = 0.

**Applications without configuration switches**
For applications without any configuration switches the following configuration method must be used:
- All configuration input bits ID0 to ID6 must be forced to HIGH level. This will select Node-ID 0 (zero).
- When starting the device for the first time the Node ID will be set to 126 (7Eh).
- Now the Object 2100h may be set to the desired Node ID by using a CANopen configuration tool.
- Then the PDO Ids must be set to the Identifiers they must have after next restart. (If they are not changed, they will be set according to Node-ID 126)
- The last step is to save this setting by using object 1010h.

If the device will be restarted the Node ID will be taken from object 2100h and will be copied to object 100Bh. Additionally the COB-Ids for the Sync-, Emergency-, SDO- messages will be modified according to the new Node ID. The PDO IDs keep the values they had during execution of the command “save dictionary”.

### *Index 100C : Guard Time*

The objects at index 100Ch (Guard Time in milli-seconds) and 100Dh (Life Time Factor) are used to implement the life guarding protocol. The Guard Time multiplied with the Life Time Factor gives the Life Time in milliseconds.
It is 0 (zero) if not used.

| | |
|---|---|
| Index | 100Ch |
| Name | Guard Time |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | |
| Default Value | 0 |

### *Index 100D : Life Time Factor*

The objects at index 100Ch (Guard Time in milli-seconds) and 100Dh (Life Time Factor) are used to implement the life guarding protocol. The Guard Time multiplied with the Life Time Factor gives the Life Time in milliseconds.
It is 0 (zero) if not used.

| | |
|---|---|
| Index | 100Dh |
| Name | Life Time Factor |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | |
| Default Value | 0 |

### *Index 100E : COB-ID Guard*

Identifier of Can Object for the Node Guarding protocol. The Object is not represented in the object dictionary because of standard conforming reasons.

| | |
|---|---|
| Index | 100Eh |
| Name | COB-ID Guard |
| Description | - |
| Data Type | Unsigned 32 |
| Access modes | Not accessible |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 700h + Node-ID |

### *Index 1010 : Store Parameters*

This object supports the saving of parameters in non-volatile memory. By read access the device provides information about its saving capabilities. Several parameter groups are distinguished:
Sub-Index 0 contains the largest Sub-Index that is supported.
Writing to Sub-Index 1 means that all parameters shall be stored on the device.
In order to avoid storage of parameters by mistake, storage is only executed when a specific signature is written to the appropriate Sub-Index. (This means that the signature is transferred as data bytes 4 to 7 in the CAN message of the corresponding SDO protocol.

The signature is "save".

| MSB | | | LSB |
|---|---|---|---|
| e | v | a | s |
| 65h | 76h | 61h | 73h |

On reception of the correct signature on sub-index 1 the device stores the parameters and then confirms the SDO transmission (initiate download response). If the storing failed, the device responds with an Abort SDO Transfer (abort code: 0606 0000 h).
If a wrong signature is written, the device refuses to store and responds with Abort SDO Transfer (Abort code: 0800 002x h).

| | |
|---|---|
| Index | 1010h |
| Name | Store Parameters |
| Description | - |
| Data Type | Structure |

| | |
|---|---|
| Index | 1010h Subindex 0 |
| Name | Largest SubIndex supported |
| Description | - |
| Data Type | Unsigned char |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | 1 |
| Default Value | 1 |

| | |
|---|---|
| Index | 1010h Subindex 1 |
| Name | Store All Parameters |
| Description | Writing to this object stores the dictionary to non-volatile memory. |
| Data Type | Unsigned 32 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | Signature "save" |
| Default Value | - |

By read access to object 1010h Subindex 1, the device responds with SDO data 00000001h indicating that storing of data is only done on command.

***Note:***
***Writing to this objected is only allowed in pre-operational mode.***
***If the command "save" is directly executed after a chip bootup that restores the default configuration (using object 1011), the CO4031 clears the previously saved dictionary from non volatile memory.***
***If the Configuration Setting is changed against the setting according to the object dictionary saved to the non volatile memory, the chip will clear the saved dictionary and will resume to the default configuration.***

The command "save" stores the complete object dictionary to the non volatile memory.
Only the objects that represent the I/O, LCD and the keyboard state of the CO4031 will not be saved. (6000, 6200, .. )

The saving of the PDO-IDs depends on the mode setting the node ID. (See object 100B for further details)

***See also object 100B for setting a valid Node-ID using the "save dictionary" command***

## *Index 1011 : Restore Default Parameters*

With this object the default values of parameters according to the communication or device profile are restored. By read access the device provides information about its capabilities to restore these values.
Several parameter groups are distinguished:
Sub-Index 0 contains the largest Sub-Index that is supported.
Writing to Sub-Index 1 restores all parameters that can be restored.
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. (This means that the signature is transferred as data bytes 4 to 7 in the CAN message of the corresponding SDO protocol.

The signature is "load".

| MSB | | | LSB |
|---|---|---|---|
| d | a | o | l |
| 64h | 61h | 6Fh | 6Ch |

On reception of the correct signature on sub-index 1 the device restores the parameters and then confirms the SDO transmission (initiate download response). If the storing failed, the device responds with an abort SDO transfer (abort code: 0606 0000h).
If a wrong signature is written, the device refuses to store and responds with Abort SDO Transfer (Abort code: 0800 002x h).
The default values are set valid after the device is reset (reset node) or power cycled. If the device requires storing on command (see Object 1010h), the appropriate command has to be executed after the reset if the default parameters are to be stored permanently.

| | |
|---|---|
| Index | 1011h |
| Name | Restore Default Parameters |
| Description | - |
| Data Type | Structure |

| | |
|---|---|
| Index | 1011h Subindex 0 |
| Name | Largest SubIndex supported |
| Description | - |
| Data Type | Unsigned char |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | 1 |
| Default Value | 1 |

| | |
|---|---|
| Index | 1011h Subindex 1 |
| Name | Restore All Parameters |
| Description | Writing to this object restores the default values to the dictionary. |
| Data Type | Unsigned 32 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | Signature "load" |
| Default Value | - |

By read access to object 1011h Subindex 1, the device responds with SDO data 00000001h indicating that restoring of default data is available.

***Note:***
***Writing to this objected is only allowed in pre-operational mode.***

### Index 1014 : COB-ID Emergency

Identifier of Can Object for the emergency messages.

| | |
|---|---|
| Index | 1014h |
| Name | COB-ID Emergency |
| Description | - |
| Data Type | Unsigned 32 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 80h + Node-ID |

### Index 1015 : Inhibit Time Emergency

Inhibit Time for emergency messages. If the Inhibit Time is set to 0, inhibit delay is disabled. The Inhibit Time is a multiple of 100usec, but the CO4031A offers a maximum resolution of 1 millisecond.

| | |
|---|---|
| Index | 1015h |
| Name | Inhibit Time Emergency |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

### Index 1017 : Producer Heartbeat Time

The producer heartbeat time defines the cycle time of the heartbeat. The producer heartbeat time is 0 if it not used. The time has to be a multiple of 1ms.

| | |
|---|---|
| Index | 1017h |
| Name | Producer Heartbeat Time |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

Note:
Either Heartbeat or node guarding may be allowed at the same time. Do not use both protocols at the same time.
See additional brochure for further information about heartbeat protocol.

### Index 1018 : Identity Object

The object at index 1018h contains general information about the device and the manufacturer frenzel + berg elektronik. It cannot be modified.

| | |
|---|---|
| Index | 1018h |
| Name | Identity Object |
| Description | - |
| Data Type | Structure |

| | |
|---|---|
| Index | 1018h Subindex 0 |
| Name | Largest SubIndex supported |
| Description | - |
| Data Type | Unsigned char |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 4 |

| | |
|---|---|
| Index | 1018h Subindex 1 |
| Name | Vendor ID |
| Description | Registration Code of frenzel + berg elektronik at the CiA |
| Data Type | Unsigned 32 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 58h |

| Index | 1018h Subindex 2 |
|---|---|
| Name | Product Code |
| Description | Internal Product Code for CO4031A at frenzel + berg elektronik |
| Data Type | Unsigned 32 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0140 3101h |

| Index | 1018h Subindex 3 |
|---|---|
| Name | Revision Code |
| Description | |
| Data Type | Unsigned 32 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | Revision of the device |

| Index | 1018h Subindex 4 |
|---|---|
| Name | Serial Number |
| Description | |
| Data Type | Unsigned 32 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

### *Index 1029 : Error Behaviour*

With object 1029 the CO4031 can be configured to enter alternatively the preoperational or the stopped state or remain in the current state in case of a device failure. Device failures shall include the following communication errors:

Bus-off conditions of the CAN interface, Life guarding error, Serious device errors also can be caused by device internal failures.

The value of the Error Classes is as follows:
0 = pre-operational
(only if current state is operational)
1 = no state change
2 = stopped
3 .. 127 = reserved

| Index | 1029h |
|---|---|
| Name | Error Behaviour Object |
| Description | - |
| Data Type | Structure |

| Index | 1029h Subindex 0 |
|---|---|
| Name | Largest SubIndex supported |
| Description | - |
| Data Type | Unsigned char |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 2 |

| Index | 1029h Subindex 1 |
|---|---|
| Name | Communication Error |
| Description | NMT state change in case of communication error |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 00h |

| Index | 1029h Subindex 2 |
|---|---|
| Name | LCD Error |
| Description | Error Class |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 00h |

### *Index 2000 : Device Manufacturer*

This Object shows "Frenzel + Berg" as visible string. If OEMs do not want to give access to this entry, it may be removed from the EDS (electronic data sheet).

| | |
|---|---|
| Index | 2000h |
| Name | Device Manufacturer |
| Description | - |
| Data Type | Visible String |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | "Frenzel + Berg" |

### *Index 2100 : New Node ID*

This Object is implemented to enable applications without any switches for setting the node ID. This object enables setting of the node ID by using SDO transfer.

| | |
|---|---|
| Index | 2100h |
| Name | New Node ID |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

If no node ID is selected with configuration input bits ID0 to ID6. The New Node ID object is enabled.
See Index 100B for details in setting the node ID.

### *Index 2101 : System Configuration*

This Object returns the operation mode of the CO4031A. It represents the inverted Setting of the configuration input bits CFG0 to CFG4.

| | |
|---|---|
| Index | 2101h |
| Name | System Configuration |
| Description | - |
| Data Type | Unsigned 32 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | Depends on CFG0 to CFG4 |

### *Index 2103 : Enabled Guarding Warning*

This Object enables/disables transmission of emergency messages in case of a node guarding warning.
The condition of a guarding warning is met, if the time between two node guarding frames increases the guarding time given in object 100C independent of the setting of the life time (object 100D). The node guarding warning does not cause any NMT state change or switching the output pins to the error state. It is implemented to give the CANopen master an early information that the guarding interval has already exceeded the predefined value.
0 : Guarding Warning is disabled
1 : Guarding Warning is enabled

| | |
|---|---|
| Index | 2103h |
| Name | Enable Guarding Warning |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

### *Index 2110 : Enable Boot Up Message*

This Object enables or disables sending of the boot up message. The boot up message is CAN object with the identifier of the node guarding object and only one data byte with value 0 (zero).
If object 2110h is set to 0 the boot up message is disabled otherwise it is enabled.

| | |
|---|---|
| Index | 2110h |
| Name | Enable Boot Up Message |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 1 (enabled) |

### Index 2180 : CAN Restart Time

This Object gives the restart time out for the CAN communication layer in case of bus off errors in milliseconds.
If the restart time is set to 0 automatic restart of the device in case of bus off is prohibited.

| | |
|---|---|
| Index | 2180h |
| Name | CAN Restart Time |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 1000 (restart after 1000 ms) |

## DS301: PDO Parameter Objects

### Communication Parameter Objects

The following table shows the communication parameter objects for Index 140x (Receive PDOs) and Index 180x (Transmit PDOs). The tables show Index 1400 as an example for all PDOs

The transmission type (sub-index 2) defines the mode for transmission / reception of the PDO. See table for detailed description of this entry.

Description of transmission type:

| Type | PDO transmission | | | | |
|---|---|---|---|---|---|
| | cyclic | acyclic | Sync related | Async. | Only on remote |
| 0 | | X | X | | |
| 1-240 | X | | X | | |
| 241-251 | Reserved | | | | |
| 252 | | | X | | X |
| 253 | | | | X | X |
| 254 | | | | X | |
| 255 | | | | X | |

Synchronous (transmission types 0-240 and 252) means that the transmission of the PDO shall be related to the SYNC object. Asynchronous means that the transmission of the PDO is not related to the SYNC object.
A transmission type of zero means that the message shall be transmitted synchronously with the SYNC object but not periodically but only in case of data change.
A value between 1 and 240 means that the PDO is transferred synchronously and cyclically, the transmission type indicating the number of SYNC signals, which are necessary to trigger PDO transmissions or receptions.

The transmission types 252 and 253 mean that the PDO is only transmitted on reception of a remote frame. At transmission type 252, the data is updated (but not sent) immediately after reception of the SYNC object. At transmission type 253 the data is updated at the reception of the remote frame. These values are only possible for transmit PDOs.
Transmission type 255 means, the application event is defined in the device profile. For receive PDOs the reception of a PDO will update the mapped data (normally the analog or digital outputs).

Sub-index 3h contains the inhibit time. This time is a minimum interval for PDO transmission. The value is defined as multiple of 100us.
Sub-index 4h is reserved.
In mode 254/255 additionally an event time can be used for TPDO. If an event timer exists for a TPDO (value not equal to 0) the elapsed timer is considered to be an event. The event time is a multiple of 1 ms. This event will cause the transmission of this TPDO in addition to otherwise defined events.

| Index | 1400/1800h |
|---|---|
| Name | Receive/Transmit PDO1 Communication Parameters |
| Description | - |
| Data Type | Structure |

| Index | 1400h/1800h Subindex 0 |
|---|---|
| Name | Largest SubIndex supported |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 5 (1400h)  2 (1800h) |

| Index | 1400h/1800h Subindex 1 |
|---|---|
| Name | COB-ID |
| Description | Identifier for CAN-Object for PDO1 |
| Data Type | Unsigned 32 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 200h + Node-Id |

| Index | 1400h/1800h Subindex 2 |
|---|---|
| Name | Transmission Type |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0FFh |

| Index | 1800h Subindex 3 |
|---|---|
| Name | Inhibit Time |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

| Index | 1800h Subindex 4 |
|---|---|
| Name | Reserved |
| Description | - |
| Data Type | - |
| Access modes | - |
| PDO Mapping | No |
| Value Range | - |
| Default Value | - |

| Index | 1800h Subindex 5 |
|---|---|
| Name | Event Time |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

The PDO communication parameter objects have the same structure for all PDOs. The following Objects are used.

| Index | PDO |
|---|---|
| 1400h | Receive PDO1<br>digital output: LEDs |
| 1404h | Receive PDO5<br>LCD interface |
| 1800h | Transmit PDO1<br>digital input: Keyboard key matrix bits |
| 1804h | Transmit PDO5<br>digital input: Keyboard key code |

For the Receive PDOs, the CO4031 uses the following default PDO settings:

| Index | Entry | Explanation |
|---|---|---|
| Receive-PDO1 | | |
| 1400.00 | 2 | RPDO1: |
| 1400.01 | 200h + NodeId | CAN-Identifier for PDO |
| 1400.02 | 0FFh | Transmission mode |
| Receive-PDO5 | | |
| 1404.00 | 2 | RPDO1: |
| 1404.01 | 80000000h | CAN-Identifier for PDO<br>PDO is disabled by default |
| 1404.02 | 0FFh | Transmission mode |

For the Transmit PDOs, the CO4031 uses the following default PDO settings:

| Index | Entry | Explanation |
|---|---|---|
| Transmit -PDO1 | | |
| 1800.00 | 5 | TPDO1: |
| 1800.01 | 180h + NodeId | CAN-Identifier for PDO |
| 1800.02 | 0FFh | Transmission mode |
| 1800.03 | 0 | Inhibit Time |
| 1800.04 | - | - |
| 1800.05 | 0 | Event Time |
| Transmit -PDO5 | | |
| 1804.00 | 5 | TPDO1: |
| 1804.01 | 80000000h | CAN-Identifier for PDO<br>PDO is disabled by default |
| 1804.02 | 0FFh | Transmission mode |
| 1804.03 | 0 | Inhibit Time |
| 1804.04 | - | - |
| 1804.05 | 0 | Event Time |

## *PDO Mapping Objects*

The following table shows the PDO Mapping Objects. The principle of PDO mapping is the same for all PDOs. The PDO Mapping table is the cross reference between the Object dictionary entries (for example the data of an digital output byte) and the data field inside an PDO data field (position in the data field of a CAN message for PDO transfer).

Subindex 0 determines the valid number of objects that have been mapped. The CO4031 allows a maximum of 8 mapped objects for each PDO. For changing the PDO mapping first subindex 0 must be set to 0 (mapping is deactivated). Then the objects can be remapped. When a new object is mapped by writing a subindex between 1 and 8, the device may check whether the object specified by index /subindex exists. If the object does not exist or the object cannot be mapped, the SDO transfer will be aborted.

Subindexes 1 to 8 keep the pointers of the mapped objects as unsigned 32 values. The value is 0 if there is no mapped object. The structure for these pointers is as follows.

MSB LSB

| Byte3 | Byte2 | Byte1 | Byte0 |
|---|---|---|---|
| Mapped index | | Subindex | Length |

Mapped Index and Subindex together are the Pointer to the Object dictionary data to be mapped at this location.
Length gives the length of the mapped object in bits.

The following mapping object uses index 1600 as an example for all mapping objects.

| Index | 1600h |
|---|---|
| Name | Receive PDO1<br>Mapping Parameters |
| Description | - |
| Data Type | Array |

| Index | 1600h Subindex 0 |
|---|---|
| Name | Largest SubIndex supported |
| Description | Number of mapped objects |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | According to operation mode |

| Index | 1600h Subindex 1 to 8 |
|---|---|
| Name | Mapped object |
| Description | |
| Data Type | Unsigned 32 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | According to operation mode |

The PDO mapping objects have the same structure for all PDOs. The following Objects are used.

| Index | PDO |
|---|---|
| 1600h | Receive PDO1<br>digital output: LEDs |
| 1604h | Receive PDO5<br>LCD interface |
| 1A00h | Transmit PDO1<br>digital input: Keyboard key matrix bits |
| 1A04h | Transmit PDO5<br>digital input: Keyboard key code |

All objects with PDO mapping capabilities may be mapped to one of the PDOs.

For the Receive PDOs, the CO4031 uses the following default mapping entries:

| Index | Entry | Explanation |
|---|---|---|
| 1600.00 | 4 | RPDO1: 4 mapped objects |
| 1600.01 | 62000108h | First mapped object:<br>LED line 1 |
| 1600.02 | 62000208h | 2. mapped object:<br>LED line 2 |
| 1600.03 | 62000308h | 3. mapped object:<br>LED line 3 |
| 1600.04 | 62000408h | 4. mapped object:<br>LED line 4 |
| 1604.00 | 1 | RPDO5: 1 mapped object |
| 1604.01 | 42000010h | First mapped object:<br>LCD Command<br>(only if LCD Display present) |

For the Transmit PDOs, the CO4031 uses the following default mapping entries:

| Index | Entry | Explanation |
|---|---|---|
| 1A00.00 | 8 | TPDO1: 8 mapped objects |
| 1A00.01 | 60000108h | First mapped object:<br>Keyboard line 1 |
| 1A00.02 | 60000208h | 2. mapped object:<br>Keyboard line 2 |
| 1A00.03 | 60000308h | 3. mapped object:<br>Keyboard line 3 |
| 1A00.04 | 60000408h | Keyboard line 4 |
| 1A00.05 | 60000508h | Keyboard line 5 |
| 1A00.06 | 60000608h | Keyboard line 6 |
| 1A00.07 | 60000708h | Keyboard line 7 |
| 1A00.08 | 60000808h | Keyboard line 8 |
| 1A04.00 | 1 | TPDO5: 1 mapped object |
| 1A04.01 | 40000010h | First mapped object:<br>Keyboard Key-Code |

## *DS401: Keyboard (Digital Input) Objects*

The following objects are describing the functionality of the Keyboard features (digital input lines) of the CO4031A. The keyboard is represented in the digital input array bitmap (object 6000). This array represents each key in one bit. So this object gives exact information which key is pressed.

For flexible keyboard control an additional object representing a key code is added. This object represents the number of a pressed key. It enables a key rollover function as additional feature. This means, that a pressed key will result in many periodically generated keystrokes. This feature may improve handling of scroll buttons, because the user can scroll down menus by keeping the scroll button pressed.
For the key rollover function several timing control objects are implemented:

| | |
|---|---|
| 4000 | Key-Code |
| 4001 | Key Rollover Time |
| 4002 | Key Rollover Delay |
| 4003 | Key Rollover Release Delay |

The key rollover function works as follows:
Key is pressed --->
The “Key-Code” is set to the valid code to indicate a pressed key --->
After the “Key Rollover Release Delay” the “Key-Code” is reset to 0, to indicate that there is no pressed key, even if the key is still pressed. --->
After the “Key Rollover Delay” the “Key-Code” is set to the valid code again. --->
After the “Key Rollover Release Delay” the “Key-Code” is reset to 0, to indicate that there is no pressed key, even if the key is still pressed. --->
After the “Key Rollover Time” the “Key-Code” is set to the valid code again. --->
Now the key stroke is simulated in a frequency according to the “Key Rollover Time”.

The behaviour of the key rollover feature gives the same keyboard feeling as a standard PC keyboard.

### *Index 4000 : Key-Code*

The object 4000 gives a key code for the pressed key. The key code is given as 16 Bit unsigned integer value.

Bitpos

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ok | rb | - | - | - | - | - | - | Key - Code | | | | | | | |

Ok — Bit is set if there is a valid key pressed and reset if there is no valid key pressed or more than one key is pressed at the same time.

Rb — Roll Over Bit. If the Roll Over Bit is enabled with object 4006 this bit indicates whether the keyboard is in roll over mode.

Key-Code — Number of pressed key

The Key-Code is calculated only, if there is exactly one key pressed. The Key-Code is calculated as:
KeyCode = (KeyLine * 8) + KeyColumn + 1
The Ok bit is set on the rising edge of the key stroke.

| Index | 4000h |
|---|---|
| Name | Key-Code |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RO |
| PDO Mapping | YES |
| Value Range | - |
| Default Value | - |

### *Index 4001 : Key-Rollover Time*

The Key Rollover Time gives the period for simulating single keystrokes for a continuously pressed key.
The time is given in milli seconds.

| Index | 4001h |
|---|---|
| Name | Key Rollover Time |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | NO |
| Value Range | 0 .. 2000 |
| Default Value | 0 (rollover disabled) |

Note:
It is not recommended to use value range 1 .. 199
A Value less than the “Key Rollover Release Delay” of Object 4003 disables the Rollover function.
A recommended value range for a good keyboard feeling is 200 .. 250.

### *Index 4002 : Key-Rollover Delay*

The Key Rollover Delay gives the time delay for starting the key rollover feature for the first time.
The time is given in milli seconds.

| Index | 4002h |
|---|---|
| Name | Key Rollover Delay |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | NO |
| Value Range | 500 .. 5000 |
| Default Value | 750 |

### *Index 4003 : Key-Rollover Release Delay*

The Key Rollover Release Delay gives the time that a continuously pressed key sets the “Key-Code” to a valid code before releasing of the key is simulated. So it gives the time that the keystroke is set valid.
The time is given in milli seconds. It is recommended to set this value to the half of object 4001 (if rollover function is wanted)

| Index | 4003h |
|---|---|
| Name | Key Rollover Release Delay |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | NO |
| Value Range | 50 .. 200 |
| Default Value | 100 |

### *Index 4004 : Key-Debounce Cycles*

Sets the number of times a key must be scanned with the same value, before it is accepted as valid. With this object, the internal debounce logic is set.

| Index | 4004h |
|---|---|
| Name | Key Debounce Cycles |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | NO |
| Value Range | 0 .. 7 |
| Default Value | 4 |

### Index 4005 : Good Key Beep Time

If a new key is scanned, the buzzer is switched on for this time. The time is given in milli seconds. If key roll over is selected a good key beep is generated for each key read.

| | |
|---|---|
| Index | 4005h |
| Name | Good Key Beep Time |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | NO |
| Value Range | |
| Default Value | 0 (no beep) |

### Index 4006 : Key Roll Over Signal Bit

This object enables key roll over indication within the key code object 4000.
Any other value than 0 will enable key roll over indication.

| | |
|---|---|
| Index | 4006h |
| Name | Key Roll Over Signal Bit |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | NO |
| Value Range | |
| Default Value | 0 (key roll over is not indicated in key code object) |

### Index 6000 : Read Keyboard Matrix

This object represents the keyboard matrix as array of digital input lines. Each key is represented by a single bit indicating thje current key state.

The number of digital input bytes is fixed to 8 bytes, even if less keyboard lines are connected.
The mapping of the I/O lines to object 6000 is explained in chapter "Mapping I/O to Object Dictionary"

| | |
|---|---|
| Index | 6000h |
| Name | Digital Input 8 Bit |
| Description | - |
| Data Type | Array |

| | |
|---|---|
| Index | Subindex 0 |
| Name | Nr of Subobjects |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | Number of digital input bytes |

| | |
|---|---|
| Index | Subindex 1 to Nr of input bytes |
| Name | Digital Input 8 Bit Byte n |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | Yes |
| Value Range | - |
| Default Value | - |

### Index 6002 : Polarity Input 8 Bit

With this object, the digital inputs may be inverted. See also Index 6000 for additional information.
The number of digital input bytes depends on the selected operation mode.

| | |
|---|---|
| Index | 6002h |
| Name | Polarity Input 8 Bit |
| Description | - |
| Data Type | Array |

| | |
|---|---|
| Index | Subindex 0 |
| Name | Nr of Subobjects |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | Number of digital input bytes |

| | |
|---|---|
| Index | Subindex 1 to Nr of input bytes |
| Name | Polarity Input 8 Bit Byte n |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

### Index 6005 : Global Interrupt Enable

This object enables or disables globally the interrupt behaviour without changing the interrupt masks.
In event-driven mode the device transmits the input values depending on the interrupt masks in objects 6006h, 6007h, and 6008h and the PDO transmission type.
TRUE (1)= global interrupt enabled
FALSE (0)= global interrupt disabled

| | |
|---|---|
| Index | 6005h |
| Name | Global Interrupt Enable |
| Description | - |
| Data Type | Boolean |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | TRUE |

### Index 6006 : Interrupt Mask Any Change

This object determines, which input lines shall activate an interrupt by any change of the input line. Both negative and positive edge will cause an interrupt, if enabled.
An interrupt will cause a PDO transmission in case of event driven transmission mode.
1 = interrupt enabled
0 = interrupt disabled

| | |
|---|---|
| Index | 6006h |
| Name | Interrupt Mask any change |
| Description | - |
| Data Type | Array |

| | |
|---|---|
| Index | Subindex 0 |
| Name | Nr of Subobjects |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | Number of digital input bytes |

| | |
|---|---|
| Index | Subindex 1 to Nr of input bytes |
| Name | Interrupt Mask any change |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0FFh (interrupt enabled) |

### Index 6007 : Interrupt Mask Low to High

This object has the same structure and behaviour as object 6006h but will cause interrupts only on rising edge of object index 6000. Note that input lines are active low, so rising edge of input data (object 6000) means falling edge of input port line.
Default value is 0.

### Index 6008 : Interrupt Mask High to Low

This object has the same structure and behaviour as object 6006h but will cause interrupts only on falling edge of object index 6000. Note that input lines are active low, so falling edge of input data (object 6000) means rising edge of input port line.
Default value is 0.

## DS401: LED (Digital Output) Objects

The following objects are describing the functionality of the LED controlling digital output lines of the CO4031A.
With objects 6206 and 6207 there can be output a specific LED setting in order to indicate bus errors even if the bus is not able to access the CO4031.

### Index 5200 : Output Reset on Error Option

This Object selects the function for error handling if an error is detected.
If the value is 0, the Outputs were set in their inactive state as long as the error is active. If error ends, the outputs will return to values of the output object 6200h.
If the value is 1, the Outputs error handling depends on the objects 6206h and 6207h and the output object 6200 will be overwritten with the error output state according to objects 6206 and 6207.

| Index | 5200h |
|---|---|
| Name | Output Reset on Error Option |
| Description | - |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

### Index 6200 : Write to Digital Output

The digital output object represents the setting of the LEDs in the LED output matrix.
With object 6200, the digital outputs of the CO4031A can be written to. Before writing to the output ports, the value of the bitmap of object 6200 is processed with object 6202 (Change polarity output 8 bit) and the inverted. (Output port bits of the CO4031A device are active low)
If no additional output inversion is set in object 6202, the setting of a bit in object 6200 causes the related LED to be switched on.

The mapping of the I/O lines to object 6200 is explained in chapter "Mapping I/O to Object Dictionary"

| Index | 6200h |
|---|---|
| Name | Write to digital output |
| Description | - |
| Data Type | Array |

| Index | Subindex 0 |
|---|---|
| Name | |
| Description | Number of mapped objects |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | NO |
| Value Range | - |
| Default Value | Number of digital output bytes |

| Index | Subindex 1 to Nr of output bytes |
|---|---|
| Name | Write to digital output |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | WO |
| PDO Mapping | YES |
| Value Range | - |
| Default Value | 0 |

### Index 6202 : Polarity Output 8 Bit

With this object, the digital outputs may be inverted. See also Index 6200 for additional information.

| Index | 6202h |
|---|---|
| Name | Polarity Output 8 Bit |
| Description | - |
| Data Type | Array |

| Index | Subindex 0 |
|---|---|
| Name | Nr of Subobjects |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | Number of digital output bytes |

| Index | Subindex 1 to Nr of input bytes |
|---|---|
| Name | Polarity Output 8 Bit Byte n |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 |

### Index 6206 : Error Mode Output 8 Bit

This object indicates, whether an output is forced to a predefined value (given in object 6207) in case of a device error.
1 = Output will be forced to the value selected in object 6207
0 = Output will be unchanged even in case of an error condition.

| | |
|---|---|
| Index | 6206h |
| Name | Error Mode Output 8 Bit |
| Description | - |
| Data Type | Array |

| | |
|---|---|
| Index | Subindex 0 |
| Name | Nr of Subobjects |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | Number of digital output bytes |

| | |
|---|---|
| Index | Subindex 1 to Nr of input bytes |
| Name | Error Mode Output 8 Bit Byte n |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0FFh<br>(Take error condition from object 6207) |

### Index 6207 : Error Value Output 8 Bit

This object selects the level the outputs are forced to in case of device error mode if the error mode (object 6206 is enabled)
1 = Output will be forced to active state
0 = Output will be forced to inactive state.

| | |
|---|---|
| Index | 6207h |
| Name | Error Value Output 8 Bit |
| Description | - |
| Data Type | Array |

| | |
|---|---|
| Index | Subindex 0 |
| Name | Nr of Subobjects |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RO |
| PDO Mapping | No |
| Value Range | - |
| Default Value | Number of digital output bytes |

| | |
|---|---|
| Index | Subindex 1 to Nr of input bytes |
| Name | Error Value Output 8 Bit Byte n |
| Description | |
| Data Type | Unsigned 8 |
| Access modes | RW |
| PDO Mapping | No |
| Value Range | - |
| Default Value | 0 (Inactive, high level) |

## DS401: LCD related Objects

The following objects are describing the functionality of the LCD functionality of the CO4031A.
The LCD is addressed in an x/y coordinates. X-direction means the display columns, Y-direction means the LCD line.

### Index 4200 : LCD-Command Register

This Object controls all access to the LCD display module. The LCD-Interface will process each writing to this object. So each SDO or PDO transfer to this object will cause a character or command cycle to the LCD.

| | |
|---|---|
| Index | 4200h |
| Name | LCD-Access |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | WO |
| PDO Mapping | YES |
| Value Range | - |
| Default Value | - |

LCD-Access Word

Bit

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | … |
|---|---|---|---|---|---|---|---|---|
| tog | res | res | res | C3 | C2 | C1 | C0 | … |

Bit

| … | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | A/C-Code | | | | | | | |

| | |
|---|---|
| Tog | Toggle Bit. This bit is not necessary for control of the CO4031s LCD interface. It is reserved to give a CANopen master the possibility to send the same CAN data once more in Event driven transmission mode (0xFF) |
| C3..C0 | Command Code for this cycle (See Command Table) |
| A/C-Code | ASCII or Command Code. In LCD character write mode, this byte keeps the ASCII data to write on the LCD. In command mode this byte keeps the command parameter. |
| Res | Reserved for future use |

Command Table

| C | | | | Command Type | A/C-Code keeps |
|---|---|---|---|---|---|
| 3 | 2 | 1 | 0 | | |
| 0 | 0 | 0 | 0 | ASCII-Write Mode | ASCII code of data to write to display (See ASCII-table) |
| 0 | 0 | 0 | 1 | Clear | Line-Code<br>00h: Clear all set Cursor X= 1, Y= 1<br>xxh: Line Nr to clear Set Cursor to first column of cleared line |
| 0 | 0 | 1 | 1 | Cursor Mode | New Cursor Mode<br>00h: cursor off<br>01h: cursor on<br>02h: cursor blink |
| 0 | 1 | 0 | 0 | Set X-Pos | New X-Position of Cursor (column) |
| 0 | 1 | 0 | 1 | Set Y-Pos | New Y-Position of Cursor (line) |
| 0 | 1 | 1 | 1 | Init | 3Ch: Init Display |

ASCII-Table of supported codes in ASCII-Write Mode

| ASCII (hex) | Explanation |
|---|---|
| 00h | Clear LCD display |
| 07h | Bell<br>(buzzer if supported by CANopen chip) |
| 0Ah | Line Feed<br>The Cursor is set to the next line. If the actual cursor position is in the lowest line, the cursor will be set to the first line. |
| 0Ch | Form Feed<br>The cursor is placed to the home position (X=1, Y=1) |
| 0Dh | Carriage Return<br>The cursor is set to the first column of the actual line. |
| 11h | Cursor one line up<br>If the cursor is in the first line, the command will be ignored. |
| 12h | Cursor one line down<br>If the cursor is in the lowest line, the command will be ignored. |
| 13h | Cursor one character (column) left.<br>If the cursor is in the first column, the command will be ignored. |
| 14h | Cursor one character (column) right.<br>If the cursor is in the last column, the command will be ignored. |
| 20h .. FFh | ASCII code of character<br>Depends on the LCD modules character set. |

### *Index 4201 : CHR7 Beep Time*

This Object controls the beep time if a character 7 is used to start a beep time. The time is given in milli seconds.

| | |
|---|---|
| Index | 4201h |
| Name | CHR7 Beep time |
| Description | - |
| Data Type | Unsigned 16 |
| Access modes | RW |
| PDO Mapping | NO |
| Value Range | - |
| Default Value | 500 |

## *Emergency Messages*

The CO4031 supports several emergency messages. For all emergencies the same structure is used:

| Byte | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| EMY-Code | | 1001 | 0 | CO4031-Code | | | |

EMY-Code: Emergency-Error-Code according to DS301
1001: Content of Object 1001
CO4031-Code: Emergency-Error-Code for CO4031 as unsigned 32 value

| CO4031-Code (hex) | May change | | Description |
|---|---|---|---|
| | NMT | I/O | |
| 8000 0000 | X | X | CAN bus bus off |
| 4000 0000 | | | CAN bus in error warning state |
| 2000 0000 | | | Node guarding warning |
| 3000 0000 | X | X | Life guarding error |
| 0000 4000 | | | LCD-FiFo Overload |
| 0000 2000 | | | LCD access error |
| | | | |

Emergency 2000 0000 (Node guarding warning) must be enabled with object 2103.

If more than one error is active at the same time, the bitmap of the CO4031-Codes for all active errors are combined with a logical or conjunction.
Some of the emergencies may cause a NMT state change. This behaviour depends on the setting of object 1029.

The ID for emergency transmission is fixed to: 0x80 + $NodeID.

List of emergency messages:

| Node-Guarding Warning | | | | | | | |
|---|---|---|---|---|---|---|---|
| 30 | 81 | 01 | 00 | 00 | 00 | 00 | 20 |

This warning occurs, if the masters fails to transmit the guarding remote frame within the specified Guard Time object 100C and if transmission is enabled in object 2103

| Life-Guarding Error | | | | | | | |
|---|---|---|---|---|---|---|---|
| 30 | 81 | 01 | 00 | 00 | 00 | 00 | 30 |

This error occurs, if the masters fails to transmit the guarding remote frame within the specified Life Time (Guard Time object 100C multiplied with Life Time Factor object 100D)

| CAN Bus in Error Warning state | | | | | | | |
|---|---|---|---|---|---|---|---|
| 00 | 81 | 01 | 00 | 00 | 00 | 00 | 40 |

This error occurs, if the chips internal CAN module is in error warning state.

| Return from CAN Bus OFF | | | | | | | |
|---|---|---|---|---|---|---|---|
| 40 | 81 | 01 | 00 | 00 | 00 | 00 | C0 |

This message indicates a return from Bus OFF state.

| LCD FIFO Overload Error | | | | | | | |
|---|---|---|---|---|---|---|---|
| 00 | 10 | 01 | 00 | 00 | 40 | 00 | 00 |

This error occurs, if the CO4031 fails to write all received LCD data to the display.

| LCD Access Error | | | | | | | |
|---|---|---|---|---|---|---|---|
| 00 | 10 | 01 | 00 | 00 | 20 | 00 | 00 |

This message indicates that the CO4031 cannot access the LCD.

## Mapping I/O to Object Dictionary

The mapping of the I/O channels is equal for all modes, except there is no LCD selected. In this case all LCD related objects are dropped.

### Summary of operation modes

The following table shows a summary of possible operation modes.

**LCD-Configuration**

With configuration bits CFG0 and CFG1 the number of LCD lines that are supported from by the LCD module is set.

| CFG 1 | CFG 0 | Nr of LCD-lines |
|---|---|---|
| 1 | 1 | No LCD module present |
| 1 | 0 | 1 |
| 0 | 1 | 2 |
| 0 | 0 | 4 |

If there is no LCD interface present, the CO4031 controller disables the corresponding PDO.

With configuration bits CFG2 and CFG3 the number of LCD characters per line that are supported from by the LCD module are set.

| CFG 3 | CFG 2 | Characters per line |
|---|---|---|
| 1 | 1 | 8 |
| 1 | 0 | 16 |
| 0 | 1 | 20 |
| 0 | 0 | 40 |

The CO4031 uses exact setting of this bits in order to enable correct calculation of the display RAM addresses.

CFG4 is reserved for future use and should be held at high level to keep compatibility to future versions.

| Operation mode 0<br>(No LCD selected)<br>CFG3 = 1 CFG 2 = 1 CFG1 = 1 CFG0 = 1<br>EDS-file: CO4031A0.EDS | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Data Mapping to Dictionary | | | | | | | | |
| Index.<br>SubIndex | Mapped I/O Signal bit/value | | | | | | | |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 6000.01 | Keyboard line 1 | | | | | | | |
| 6000.02 | Keyboard line 2 | | | | | | | |
| 6000.03 | Keyboard line 3 | | | | | | | |
| 6000.04 | Keyboard line 4 | | | | | | | |
| 6000.05 | Keyboard line 5 | | | | | | | |
| 6000.06 | Keyboard line 6 | | | | | | | |
| 6000.07 | Keyboard line 7 | | | | | | | |
| 6000.08 | Keyboard line 8 | | | | | | | |
| 4000 | Key-Code | | | | | | | |
| 6200.01 | LED Line 1 | | | | | | | |
| 6200.02 | LED Line 2 | | | | | | | |
| 6200.03 | LED Line 3 | | | | | | | |
| 6200.04 | LED Line 4 | | | | | | | |
| | | | | | | | | |

| Default PDO Mapping | |
|---|---|
| PDO | Mapped Data |
| RPDO1 | 6200.01 digital output LED Line 1<br>6200.02 digital output LED Line 2<br>6200.03 digital output LED Line 3<br>6200.04 digital output LED Line 4 |
| TPDO1 | 6000.01 digital input Key. Line 1<br>6000.02 digital input Key. Line 2<br>6000.03 digital input Key. Line 3<br>6000.04 digital input Key. Line 4<br>6000.05 digital input Key. Line 5<br>6000.06 digital input Key. Line 6<br>6000.07 digital input Key. Line 7<br>6000.08 digital input Key. Line 8 |
| TPDO5 | 4000 Key-Code |

## Absolute Maximum Ratings

Stresses greater than those listed parameters may cause permanent damage to the device. Functional operation should be restricted to recommended operation conditions. Exposure to absolute maximum rating conditions for extended times may affect reliability.

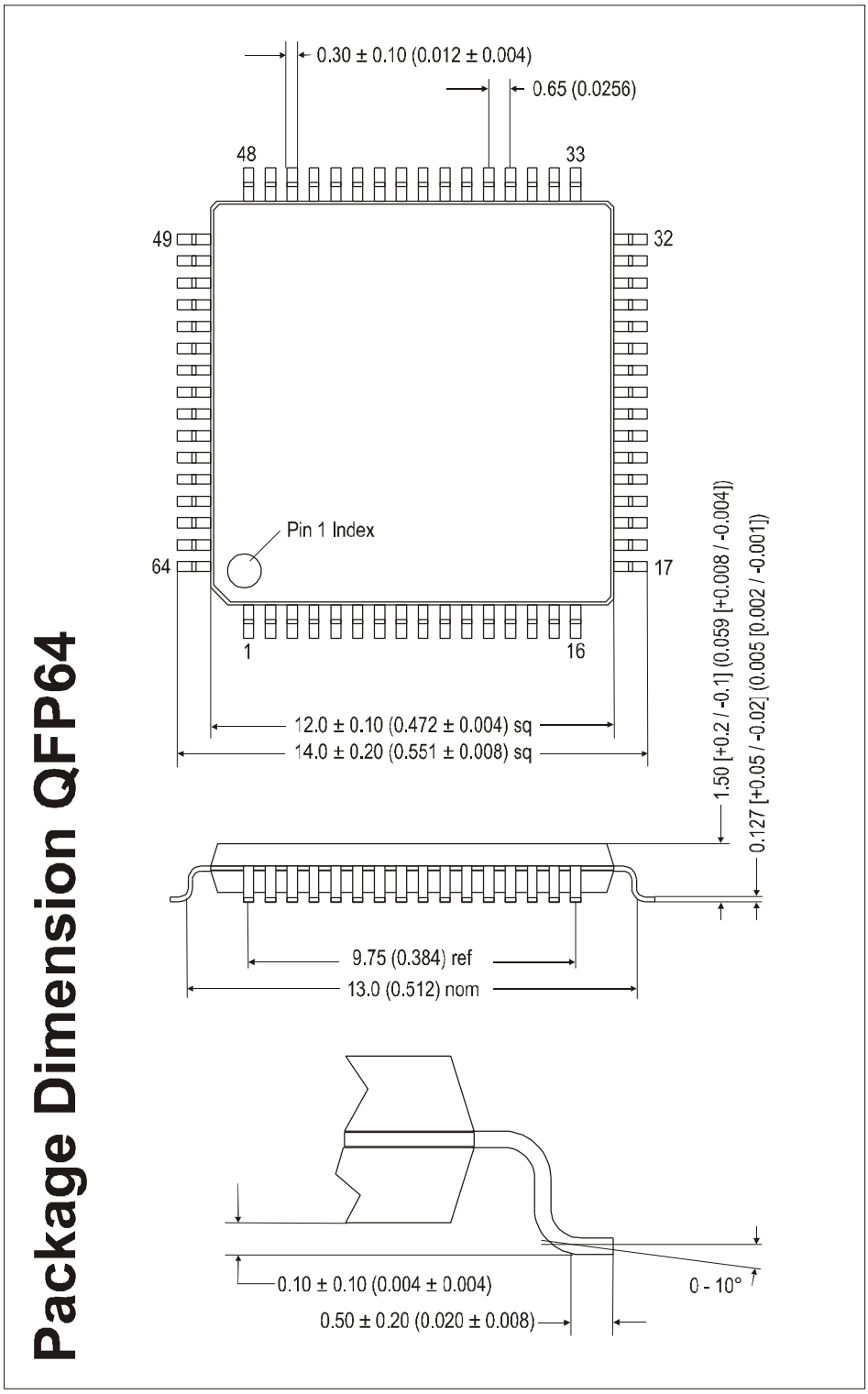
| Parameter | Symbol | Rated Value | | Units | Remarks |
|---|---|---|---|---|---|
| | | Min. | Max. | | |
| Power supply voltage | VCC | VSS – 0.3 | VSS + 6.0 | V | |
| Analog power supply voltage | AVCC | VSS – 0.3 | VCC | V | |
| Analog reference voltage | AVREF | VSS – 0.3 | VCC | V | |
| Input voltage | Vi | VSS – 0.3 | VSS + 6.0 | V | Vi < VCC + 0.3V |
| Output voltage | Vo | VSS – 0.3 | VSS + 6.0 | V | Vo < VCC + 0.3V |
| L level maximum output current | $I_{OLMAX}$ | | 15 | mA | Time < 20 msec |
| L level maximum output current | $I_{OL}$ | | 4 | mA | |
| H level maximum output current | $I_{OHMAX}$ | | 15 | mA | Time < 20 msec |
| H level maximum output current | $I_{OH}$ | | 4 | mA | |
| Maximum Power dissipation | $P_{MAX}$ | | 300 | mW | |
| Operating temperature | $T_A$ | -40 | +85 | $^oC$ | CO4031A |
| | $T_A$ | -40 | +105 | $^oC$ | CO4031AE |
| Storing temperature | $T_A$ | -55 | +150 | $^oC$ | |

## Recommended Operation Conditions and Characteristics

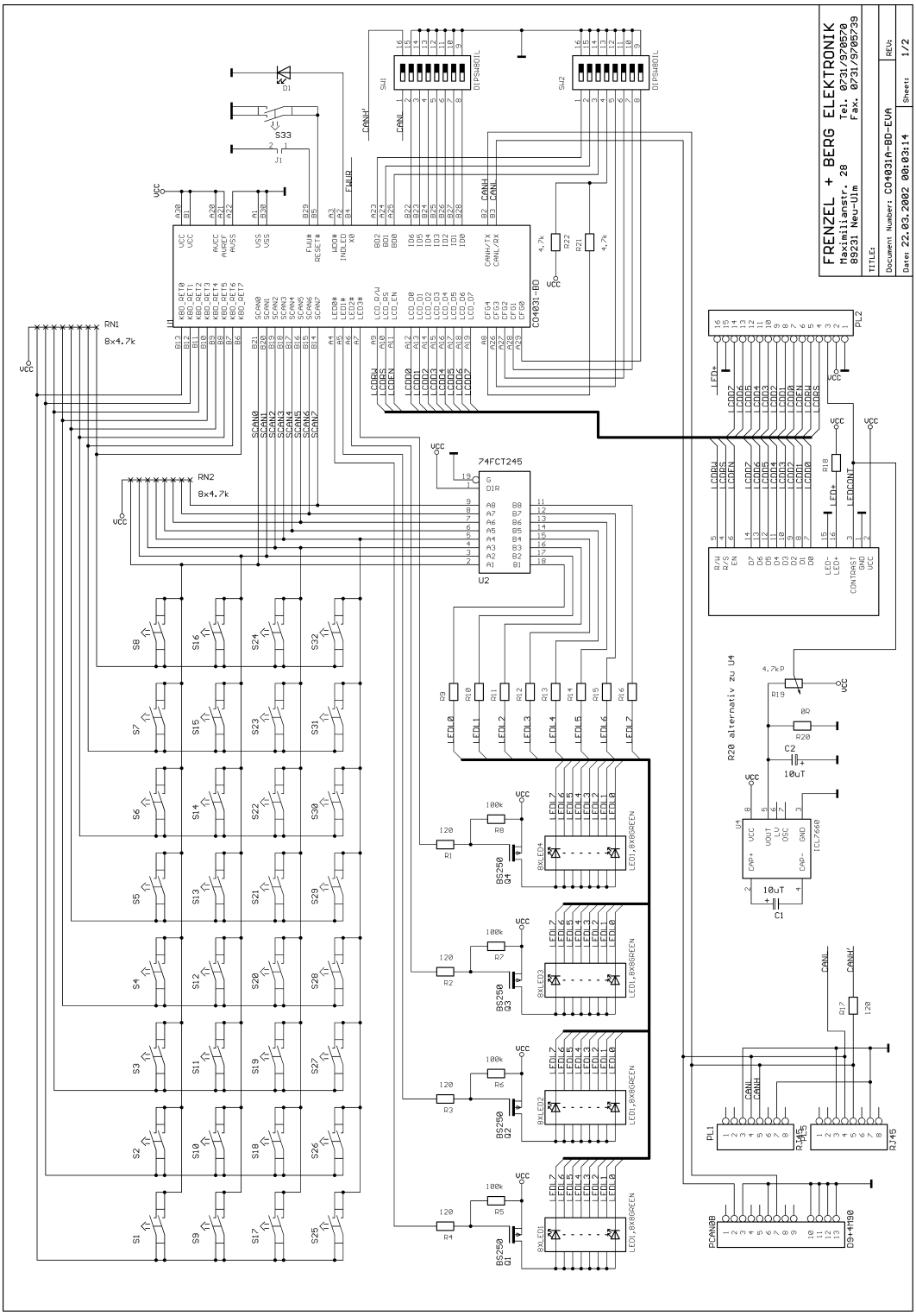
Functional operation should be restricted to recommended operation conditions.

| Parameter | Symbol | Rated Value | | | Units | Remarks |
|---|---|---|---|---|---|---|
| | | Min. | Typ. | Max. | | |
| Power supply voltage | VCC | 4.5 | 5.0 | 5.5 | V | |
| Analog power supply voltage | AVCC | | | VCC | V | |
| Analog reference voltage | AVREF | | | VCC | V | |
| Power supply current | $I_{CC}$ | | 35 | 50 | mA | All inputs $V_{IL}$ or $V_{IH}$ All outputs open |
| Input H voltage | $V_{IH}$ | 0.8 * VCC | | VCC + 0.3 | V | |
| Input L voltage | $V_{IL}$ | VSS – 0.3 | | 0.2 * VCC | V | |
| Output H voltage | $V_{OH}$ | VCC – 0.5 | | | V | $I_{OH}$ = -4.0 mA |
| Output L voltage | $V_{OL}$ | | | 0.4 | V | $I_{OL}$ = 4.0 mA |
| Input leakage current | $I_{LKC}$ | -5 | | 5 | uA | |
| Crystal frequency | $f_{osc}$ | | 4 | | MHz | |
| Reset pulse width | $T_{res}$ | 10 | | | us | |
| Power on rise time | $t_{RESLH}$ | 0.05 | | 30 | ms | |
| Maximum CANopen Receive-PDO frequency | $f_{RPDO}$ | | | 1 | kHz | |
| Watchdog trigger frequency | $f_{WDT}$ | | 500 | | Hz | Delayed max. 500 ms after reset |
| Maximum Power dissipation | $P_{MAX}$ | | | 300 | mW | |
| Operating temperature | $T_A$ | -40 | | +85 | $^oC$ | CO4031A |
| | $T_A$ | -40 | | +105 | $^oC$ | CO4031AE |

## *Package Dimension CO4031A*



## *Typical Application: EVA board for CO4031*



## *Version History and Notes*

| Version | Date | Changes | |
|---|---|---|---|
| 1.25 | April/26/2005 | BugFix | NMT Reset Communication resets node guarding to not started state. This avoids error condition after the following situation: Node guarding is active and in error state → Reset communication → Set node guarding objects 100C and 100D → Guarding is in error state. |
| | | New | Object 4006 added, Object 4000 rollover bit added |

The information herein is given to describe certain components and shall not be considered as warranted characteristics. Terms of delivery and all rights to technical changes are reserved.

We hereby disclaim any and all warranties, including but not limited to warranties of non-infringement, regarding circuits, descriptions and charts stated herein.

## Index