Project CANopen encoder
Device 5868

Additional extended features

Event mode programmable
default : time triggered PDO
alternate: event triggered mode

Start autonomously programmable
default: start to pre-operational
alternate: start to operational

Units for speed control selectable
default: revolutions / minute
alternate: measuring units /second

Receive PDO for Presetvalue
default: 1 PDO with predefined COB-ID
applications with a “realtime” preset-function

High Resolution Sync Protocol
default: SYNC PDO
alternate: time stamp mode

2 Working areas with 2 lower and 2 upper limits

Key-features

High Speed position capturing every 100µs

Extended Error Management for position controller and sensor with temperature control

- emergency messages generated on faults
- optical indication

User Interface with an optical error indication over 3 LED’s

Internal Diagnostic modes with built-in serial user interface

Preset value adaptable to a built-in preset-key

All “physicals” programmable -> Node address, Baudrate, CANbus termination

Manufacturer specific objects
Firmware flash-mode with Checksum in Object 2103h
Optical Sensor test data

Watchdog controlled device

All communication ,application and manufacturer parameters can be stored and are non-volatile

Special objects and protocols

Heartbeat Protocol

The Heartbeat Protocol defines an Error Control Service without need for remote frames. A Heartbeat Producer transmits a Heartbeat message cyclically. One or more Heartbeat Consumer receive the indication. The relationship between producer and consumer is configurable via the object dictionary. The Heartbeat Consumer guards the reception of the Heartbeat within the Heartbeat Consumer Time. If the Heartbeat is not received within the Heartbeat Consumer Time a Heartbeat Event will be generated.

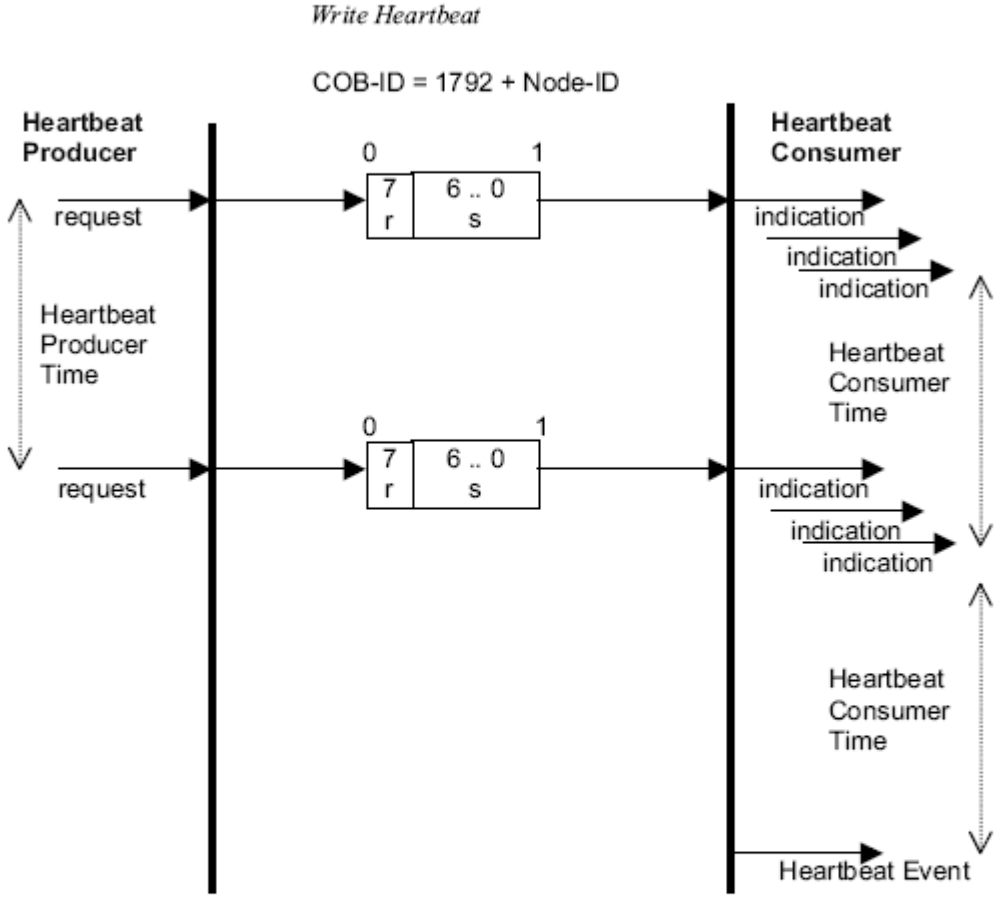


Figure 42: Heartbeat Protocol

r: reserved (always 0)
s: the state of the Heartbeat producer

0:	BOOTUP
4:	STOPPED
5:	OPERATIONAL
127:	PRE-OPERATIONAL

If the Heartbeat Producer Time is configured on a device the Heartbeat Protocol begins immediately. If a device starts with a value for the Heartbeat Producer Time unequal to 0 the Heartbeat Protocol starts on the state transition from INITIALISING to PRE-OPERATIONAL. In this case the Bootup Message is regarded as first heartbeat message. It is not allowed for one device to use both error control mechanisms Guarding Protocol and Heartbeat Protocol at the same time. If the heartbeat producer time is unequal 0 the heartbeat protocol is used.

Bootup protocol

This protocol is used to signal that a NMT slave has entered the node state PRE-OPERATIONAL after the state INITIALISING. The protocol uses the same identifier as the error control protocols.

Bootup Event

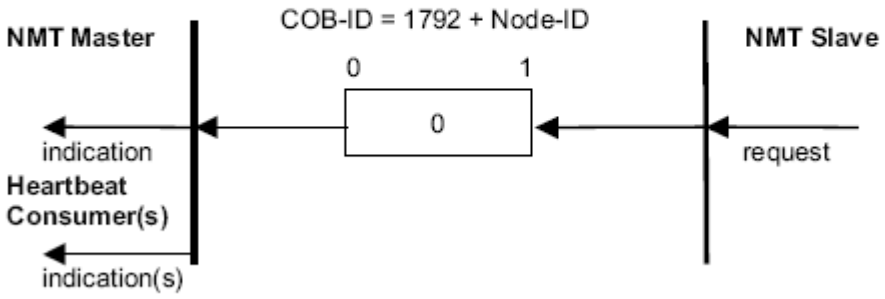
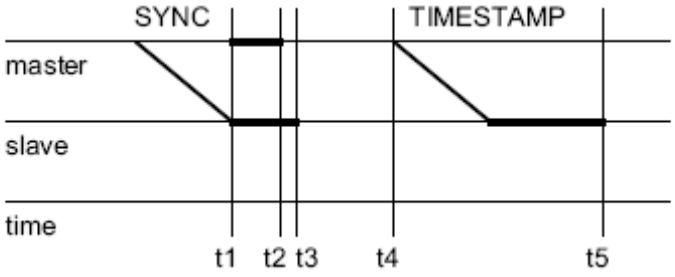


Figure 43: Bootup Protocol

High Resolution Sync Protocol

The synchronisation message carries no data and is easy to generate. However, the jitter of this SYNC depends on the bit rate of the bus as even the very high priority SYNC has to wait for the current message on the bus to be transmitted before it gains bus access.
Some time critical applications especially in large networks with reduced transmission rates require more accurate synchronisation; it may be necessary to synchronise the local clocks with an accuracy in the order of microseconds. This is achieved by using the optional high resolution synchronisation protocol which employs a special form of time stamp message (see Figure 46) to adjust the inevitable drift of the local clocks.
The SYNC producer time-stamps the interrupt generated at t1 by the successful transmission of the SYNC message (this takes until t2). After that (at t4) he sends a time-stamp message containing the corrected time-stamp (t1) for the SYNC transmission success indication. The SYNC consumer that have taken local time-stamps (t3) on the reception (t1) of the SYNC can now compare their corrected time-stamp (t1) with the one received in the time-stamp message from the SYNC producer. The difference between these values determines the amount of time to adjust the local clock. With this protocol only the local latencies (t2-t1 on the SYNC producer and t3-t1 on the SYNC consumer) are time critical. These latencies depend on local parameters (like interrupt processing times and hardware delays) on the nodes which have to be determined once. The accuracy of this determination is implementation specific, it forms the limiting factor of the synchronisation (or clock adjustment) accuracy. Note that each node only has to know its own latency time as the time-stamp message contains the corrected value t1 and not t2.
The time-stamp is encoded as UNSIGNED32 with a resolution of 1 microsecond which means that the time counter restarts every 72 minutes. It is configured by mapping the high resolution time-stamp into a PDO.
It is reasonable to repeat the clock adjustment only when the maximum drift of the local clock exceeds the synchronisation accuracy. For most implementations this means that it is sufficient to add this time-stamp message to the standard SYNC once every second.
This principle enables the best accuracy that can be achieved with bus-based synchronisation, especially when implemented on CAN controllers that support time-stamping. Note that the accuracy is widely independent of the transmission rate. Further improvement requires separate hardware (e.g. wiring).



Object 1013h: High Resolution Time Stamp

This object contains a time stamp with a resolution of 1 µs (see 9.3.2). It can be mapped into a PDO in order to define a high resolution time stamp message. (Note that the data type of the standard time stamp message (TIME) is fixed). Further application specific use is encouraged.

OBJECT DESCRIPTION

INDEX	1013h
Name	high resolution time stamp
Object Code	VAR
Data Type	UNSIGNED32
Category	Optional

ENTRY DESCRIPTION

Access	rw
PDO Mapping	Optional
Value Range	UNSIGNED32
Default Value	0

Object 1017h: Producer Heartbeat Time

The producer hartbeat 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.

OBJECT DESCRIPTION

INDEX	1017h
Name	Producer Heartbeat Time
Object Code	VAR
Data Type	UNSIGNED16
Category	Conditional; Mandatory if guarding not supported

ENTRY DESCRIPTION

Access	rw
PDO Mapping	No
Value Range	UNSIGNED16
Default Value	0

Object 1018h: Identity Object

The object at index 1018h contains general information about the device.
The Vendor ID (sub-index 1h) contains a unique value allocated to each manufacturer.
The manufacturer-specific Product code (sub-index 2h) identifies a specific device version.
The manufacturer-specific Revision number (sub-index 3h) consists of a major revision number and a minor revision number. The major revision number identifies a specific CANopen behaviour. If the CANopen functionality is expanded, the major revision has to be incremented. The minor revision number identifies different versions with the same CANopen behaviour.

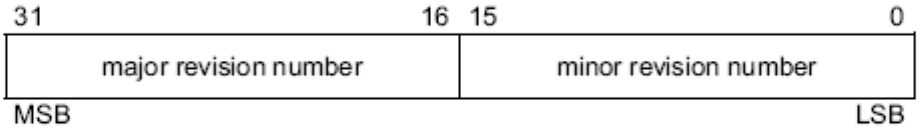


Figure 63: Structure of Revision number

The manufacturer-specific Serial number (sub-index 4h) identifies a specific device.

OBJECT DESCRIPTION

INDEX	1018h
Name	Identity Object
Object Code	RECORD
Data Type	Identity
Category	Mandatory

11.6.2 Error behaviour object

If a serious device failure is detected in Operational State, the module shall enter by default autonomously the pre-operational state. If object 1029h (Error Behaviour) is implemented, the device can be configured to enter alternatively 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 event with the state 'occurred'
- Heartbeat event with state 'occurred'

Severe 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

OBJECT DESCRIPTION

INDEX	1029h
Name	Error Behaviour
Object Code	ARRAY
Data Type	UNSIGNED8
Category	Optional

ENTRY DESCRIPTION

Sub-Index	0h
Description	No. of Error Classes
Entry Category	Mandatory
Access	ro
PDO Mapping	No
Value Range	1h to FEh
Default Value	No

Sub-Index	1h
Description	Communication Error
Entry Category	Mandatory
Access	rw
PDO Mapping	No
Value Range	UNSIGNED8
Default Value	0

Sub-Index	2h to FEh
Description	Device profile or manufacturer specific Error
Entry Category	Optional
Access	rw
PDO Mapping	No
Value Range	UNSIGNED8
Default Value	No

9.2.5 Emergency Object (EMCY)

9.2.5.1 Emergency Object Usage

Emergency objects are triggered by the occurrence of a device internal error situation and are transmitted from an emergency producer on the device. Emergency objects are suitable for interrupt type error alerts. An emergency object is transmitted only once per 'error event'. As long as no new errors occur on a device no further emergency objects must be transmitted.
The emergency object may be received by zero or more emergency consumers. The reaction on the emergency consumer(s) is not specified and does not fall in the scope of this document.
By means of this specification emergency error codes (Table 21) and the error register (Table 48) are specified. Device specific additional information and the emergency condition do not fall into the scope of this specification.

Table 21: Emergency Error Codes

Error Code (hex)	Meaning
00xx	Error Reset or No Error
10xx	Generic Error
20xx	Current
21xx	Current, device input side
22xx	Current inside the device
23xx	Current, device output side
30xx	Voltage
31xx	Mains Voltage
32xx	Voltage inside the device
33xx	Output Voltage
40xx	Temperature
41xx	Ambient Temperature
42xx	Device Temperature
50xx	Device Hardware
60xx	Device Software
61xx	Internal Software
62xx	User Software
63xx	Data Set
70xx	Additional Modules
80xx	Monitoring
81xx	Communication
8110	CAN Overrun (Objects lost)
8120	CAN in Error Passive Mode
8130	Life Guard Error or Heartbeat Error
8140	recovered from bus off
8150	Transmit COB-ID collision
82xx	Protocol Error
8210	PDO not processed due to length error
8220	PDO length exceeded
90xx	External Error
F0xx	Additional Functions
FFxx	Device specific

5.2 Error behavior

If a serious device failure is detected the module shall enter by default autonomously the pre-operational state. If object 1029_h is implemented, the device may be configured to enter alternatively 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 event with the state 'occurred'
- Heartbeat event with state 'occurred'

Severe device errors also may be caused by device internal failures.

5.3 Additional error code meanings

Error Code	Meaning
2110_h	Input current too high
3110_h	Input voltage out of range
5100_h	Hardware memory error

6.3 Error behavior (1029_h)

The object specifies to which state an encoder module shall be set, when a communication error or severe internal encoder error is detected.

0_h = pre-operational (only if the current state is operational)

1_h = no state change

2_h = stopped

In addition to the specification in /1/ the following sub-indices may be implemented.

Sub-Index	2_h
Description	Internal encoder error
Access	rw
Entry Category	Optional
PDO Mapping	No
Value Range	0_h to 2_h
Default Value	0_h

Note: If the object 1029_h is not implemented the device shall be set into the pre-operational state in the case a communication error is detected.

5.10.2 Internally created emergency messages

Internally the following events will lead to the transmission of an emergency message:

Table 20: internally generated EMCY messages

Value	Symbol	Meaning
0x8110	ECY_OVR_MON	The CAN controller has signalled a lost message.
0x8120	ECY_STATE_MON	The CAN controller has reached the warning limit due to error frames.
0x8210	ECY_LEN_EXC_PROT	A received PDO was smaller than specified by the valid mapping table
0x8220	ECY_LEN_PROT	The DLC of a received PDO exceeded the length of the valid mapping.
0x8130	ECY_ERR_MON	An error control event has been occurred. This can be a life guarding event or a heartbeat event.
0x8140	ECY_BOFF_MON	The CAN controller has recovered from BUS OFF.
0x8150	ECY_COB_COLL	A COB-ID collision was detected.
0x6261*		The object dictionary entries is not sorted in an ascending order. Some objects will not be found.
0x6262*		The length of a mapped object differs from the entry in the object dictionary.
0x6263*		According to the access type of the object it is not allowed to be mapped to this mapping table.
0x6264*		The physical address of the mapped object differs from the value defined in the object dictionary.
0x6265*		The mapped object does not exist.
0x6266*		The mapping exceeds the 64 bit limit of a single CAN message.

Variable PDO Mapping

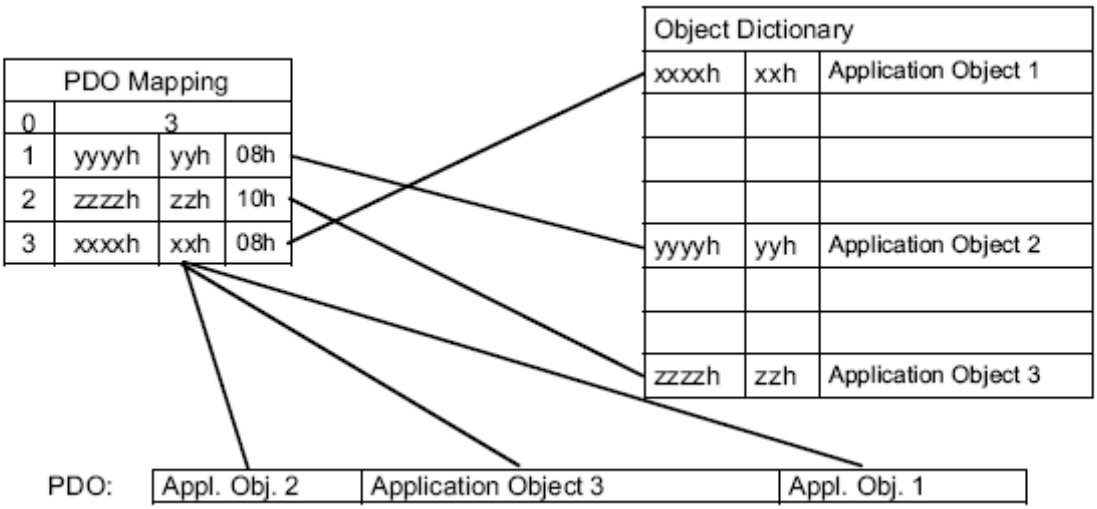


Figure 67: Principle of PDO mapping

6.4 Process data objects

Two PDOs to be transmitted shall be implemented in each encoder device by default. One is used for asynchronous transmission and the other one for the cyclic transmission functions.

6.4.1 1st Transmit PDO (asynchronous transmission)

This PDO transmits asynchronously the position value of the encoder. The event timer and the cyclic timer object (6200_h) are hard-wired, meaning that a SDO write access will cause changes in the event timer as well as object 6200_h. The 1st TPDO shall be transmitted when entering the Operational state.

Transmit PDO Communication Parameter

Index	Sub-Index	Comment	Default Value
1800_h	0	Largest sub-index supported	No
	1	COB-ID used by PDO	See /1/
	2	Transmission type	254
	3	Inhibit time	0
	4	reserved	See /1/
	5	Event timer	0

Transmit PDO Mapping Parameter

Index	Sub-Index	Comment	Default Value
$1A00_h$	0	Number of mapped objects	1
	1	Position value	6004 00 20_h*

* The default value for multi-sensor devices is 6020 01 20_h.

7.2.11 Speed value (6030_h)

This object shall define the output speed value(s). For linear encoders the speed-measuring step is defined in object 6005_h, sub-index 02_h. For rotary encoders the speed dimension shall be always measuring units per second:

speed = measuring units / second

This object is only mandatory for multi-sensor encoders (object 1000_h encoder type: code 10).

VALUE DEFINITION

Speed value	
Byte 0	Byte 1
2^7 to 2^0	2^{15} to 2^8

7.2.12 Acceleration value (6040_h)

This object shall define the output acceleration value(s). For linear encoders the acceleration-measuring step is defined in object 6005_h, sub-index 03_h. For rotary encoders the speed dimension shall be always measuring units per square second:

acceleration = measuring units / square second

VALUE DEFINITION

Acceleration value	
Byte 0	Byte 1
2^7 to 2^0	2^{15} to 2^8

7.2.14 Cyclic timer (6200_h)

This object shall define the transmission period for TPDO_1. It shall be hard-wired to the PDO's event timer meaning that a change in the event timer causes a change in object 6200_h and vice versa.

VALUE DEFINITION

A cyclic transmission of the position value shall be set, when the cyclic timer is programmed > 0. Values between 1 ms and 65,535 ms shall be selectable.

e.g.: 1 ms = 1_h

256 ms = 100_h

7.4 Work area supervision

It is possible for encoders to define a so-called user defined working area. The actual work area information with work area low limit and work area high limit may be stored in objects (6401_h and 6402_h), respectively. This way, the area state object (6400_h) may also be used as software limit switches.

7.4.1 Area state register (6400_h)

The object shall contain the actual area status of the encoder position. If the position is out of range, a bit shall be set in the related position line. If the position is lower than the position value set in object 6401_h „work area low limit" then bit 2 flags the underflow. If the position is higher than the position value set in object 6402_h „work area high limit" then bit 1 flags the overflow. If the manufacturer minimum position value or the manufacturer maximum position value (refer to module identification object, $650A_h$) is reached, bit 0 flags „out of range".

VALUE DEFINITION

Work_area_state							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
reserved	reserved	reserved	reserved	reserved	range underflow	range overflow	out of range

OBJECT DESCRIPTION

INDEX	6400_h
Name	Area_state_register
Object Code	ARRAY
Data Type	Unsigned8
Category	Optional

ENTRY DESCRIPTION

Sub-Index	00_h
Description	Number_of_available_channels
Entry Category	Mandatory
Access	ro
PDO Mapping	no
Value Range	1 - 254
Default Value	no

Sub-Index	01_h
Description	Work_area_state_channel_1
Entry Category	Mandatory
Access	ro
PDO Mapping	Optional
Value Range	Unsigned8
Default Value	no

2 Workareas with 2 lower and 2 upper limits

7.4.2 Work area low limit (6401_h)

This object shall contain the position value, at which bit 2 of the according work_area_state_channel in object 6400_h shall flag the underflow of the related work area.

VALUE DEFINITION

Work area low limit			
Byte 0	Byte 1	Byte 2	Byte 3
2^7 to 2^0	2^{15} to 2^8	2^{23} to 2^{16}	2^{31} to 2^{24}

7.4.3 Work area high limit (6402_h)

This object shall contain the position value, at which bit 1 of the according work_area_state_channel in object 6400_h shall flag the overflow of the related work area.

VALUE DEFINITION

Work area high limit			
Byte 0	Byte 1	Byte 2	Byte 3
2^7 to 2^0	2^{15} to 2^8	2^{23} to 2^{16}	2^{31} to 2^{24}