Gen 2.0 Joystick CANopen Modules

User Manual

Grayhill PN: 3JUM2020-1

Revisions

Revision	Description	Check / Approve
A	Original. JAA 10/26/2020	DW / JLF

Table of Contents

1. Overview

This document explains the usage and functionality of the 3J Gen 2 Series of CANopen joystick controller modules.

2. Relevant Documents

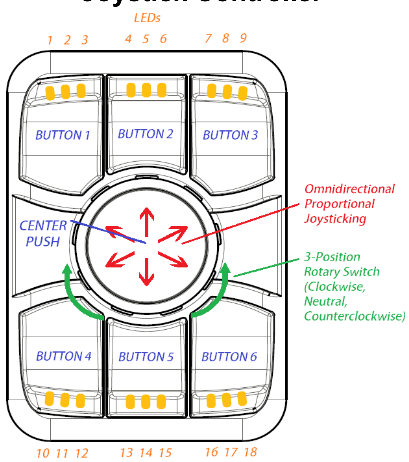
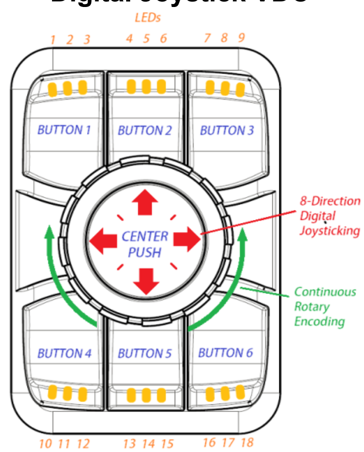
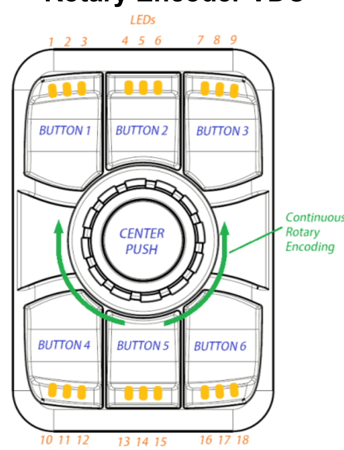
- **CiA Draft Standard 301 Version 4.2.0:** Application Layer and Communication Protocol
- **CiA Draft Standard 304 Version 1.0.1:** Framework for Safety-Relevant Communications
- **CiA Draft Standard 305 Version 3.0.0:** Layer Setting Services (LSS) and Protocols
- **CiA Draft Standard 320 Version 1.0.0:** Services and Protocols for Sleep and Wake-up Handling

3. Acronyms

- **CAN** Controller area network
- **CANID** CAN identifier
- **COB** Communication object
- **COBID** COB identifier
- **CRC** Cyclic redundancy check
- **LSB** Least significant bit/byte
- **LSS** Layer Setting Services
- **MSB** Most significant bit/byte
- **NMT** Network management
- **NodeID** Node identifier
- **OD** Object Dictionary
- **PDO** Process data object
- **RO** Read Only
- **RPDO** Receive-PDO
- **RTR** Remote transmission request
- **RW** Read/Write
- **SDO** Service data object
- **SCT** Safeguard cycle time
- **SRDO** Safety-relevant data object
- **SRVT** Safety-relevant object validation time
- **TPDO** Transmit-PDO
- **uC** Microcontroller

4. 3J Joystick Form Factors

Similar to the 3K, the buttons and indicators are numbered sequentially from left to right.

Joystick Controller	Digital Joystick VDC	Rotary Encoder VDC
		

5. Process Variables

Process variables are variables that are transmitted via PDOs (or SRDOs for CANopen Safety). These variables may be accessed via SDO requests and are located in the object dictionary range starting at 2000h. PDO mapping is used to map these variables within the PDO data bit-field that appears on the CAN bus. The following is a brief explanation of the process variables. Note that the process variables for the brightness control of the backlight and indicators do not exist in the CANopen Safety SRDOs. These will need to be controlled via the RxPDO2 and SDO Request.

5.1. Common to All Devices

Three process variables exist that are common to all devices which are indicated below:

- **Button Status**: Transmitted on TxPDO1 where each button corresponds to a bit.
- **Indicator Control**: Received via RxPDO1 were each indicator corresponds to a bit.
- **Backlight and Indicator Intensity**: Each is received via RxPDO2 and has a width of 8 bits.

5.2. Unique Variables and Associated Message Formats

The Encoder Only, Digital Joystick, and Proportional Joystick Single Sensor, and Proportional Joystick Dual Sensor each has unique process variables corresponding to its unique functionality. All process variables are mapped to TxPDO1. Though button data (not including the Center Push) is handled essentially the same between devices with the different joystick types, it is included below to show where all fields are placed in the associated TxPDO1 message.

- **Encoder Only (Rotary Encoder VDC)**
  - **Encoder Value**: Occupies the first 16 bits.
  - **Encoder Direction**: Occupies the next 8 bits.
  - **Button Data**: Occupies the next 6 bits.
  - **Center Push**: Occupies the bit following the 6 related to the buttons.

Standard TxPDO1 message data per default mapping:

Start	Length	Desc.	Values
1.1	16 bits	Encoder Data (Signed Integer)	Start value: 0000h Min: according to “encoder min value” parameter Max: according to “encoder max value” parameter
3.1	8 bits	Encoder Direction (Signed Integer)	01h (+1) - Clockwise FFh (-1) - Counter-Clockwise
4.1	1 bit	Button 1 Status	0 - Not pressed 1 - Pressed
4.2	1 bit	Button 2 Status	0 - Not pressed 1 - Pressed
4.3	1 bit	Button 3 Status	0 - Not pressed 1 - Pressed
4.4	1 bit	Button 4 Status	0 - Not pressed 1 - Pressed
4.5	1 bit	Button 5 Status	0 - Not pressed 1 - Pressed
4.6	1 bit	Button 6 Status	0 - Not pressed 1 - Pressed
4.7	1 bit	Center Push Status	0 - Not pressed 1 - Pressed
4.8	33 bits	Unused	All bits 0 - Ignore

- **Digital Joystick Encoder (Digital Joystick VDC)**
  - **Encoder Value**: Occupies the first 16 bits.
  - **Encoder Direction**: Occupies the next 8 bits.
  - **Button Data**: Occupies the next 6 bits.
  - **Center Push**: Occupies the bit following the 6 related to the buttons.
  - **Joystick Direction**: Occupies the next 4 bits where each cardinal direction has a corresponding bit.

Standard TxPDO1 message data per default mapping:

Start	Length	Desc.	Values
1.1	16 bits	Encoder Data (Signed Integer)	Start value: 0000h Min: according to “encoder min value” parameter Max: according to “encoder max value” parameter
3.1	8 bits	Encoder Direction (Signed Integer)	01h (+1) - Clockwise FFh (-1) - Counter-Clockwise
4.1	1 bit	Button 1 Status	0 - Not pressed 1 - Pressed
4.2	1 bit	Button 2 Status	0 - Not pressed 1 - Pressed
4.3	1 bit	Button 3 Status	0 - Not pressed 1 - Pressed
4.4	1 bit	Button 4 Status	0 - Not pressed 1 - Pressed
4.5	1 bit	Button 5 Status	0 - Not pressed 1 - Pressed
4.6	1 bit	Button 6 Status	0 - Not pressed 1 - Pressed
4.7	1 bit	Center Push Status	0 - Not pressed 1 - Pressed
4.8	1 bit	Joystick Y-Axis Down	0 - Not pushed 1 - Pushed
5.1	1 bit	Joystick Y-Axis Up	0 - Not pushed 1 - Pushed
5.2	1 bit	Joystick X-Axis Left	0 - Not pushed 1 - Pushed
5.3	1 bit	Joystick X-Axis Right	0 - Not pushed 1 - Pushed
5.4	29 bits	Unused	All bits 0 - Ignore

- **Proportional Joystick (Joystick Controller w/ Proportional Joystick Single Sensor)**
  - **X-Axis**: Occupies the first 8 bits.
  - **Y-Axis**: Occupies the next 8 bits.
  - **Center Twist**: Occupies the next 8 bits.
  - **Button Data**: Occupies the next 6 bits.
  - **Center Push**: Occupies the bit following the 6 related to the buttons.

Standard TxPDO1 message data per default mapping:

Start	Length	Desc.	Values
1.1	8 bits	Joystick X-Axis (Signed Integer)	0 - Neutral 1 to 32h (1 to 50) - Right FFh to CEh (-1 to -50) - Left
2.1	8 bits	Joystick Y-Axis (Signed Integer)	0 - Neutral 1 to 32h (1 to 50) - Up FFh to CEh (-1 to -50) - Down
3.1	8 bits	Center Twist	0 - Neutral 1 - Clockwise 2 - Counter-Clockwise
4.1	1 bit	Button 1 Status	0 - Not pressed 1 - Pressed
4.2	1 bit	Button 2 Status	0 - Not pressed 1 - Pressed
4.3	1 bit	Button 3 Status	0 - Not pressed 1 - Pressed
4.4	1 bit	Button 4 Status	0 - Not pressed 1 - Pressed
4.5	1 bit	Button 5 Status	0 - Not pressed 1 - Pressed
4.6	1 bit	Button 6 Status	0 - Not pressed 1 - Pressed
4.7	1 bit	Center Push Status	0 - Not pressed 1 - Pressed
4.8	33 bits	Unused	All bits 0 - Ignore

- **Dual Proportional Joystick (Joystick Controller w/ Prop. Joystick Dual Sensor)**
  - **X-Axis Primary**: Occupies the first 8 bits.
  - **Y-Axis Primary**: Occupies the next 8 bits.
  - **Center Twist**: Occupies the next 8 bits.
  - **Button Data**: Occupies the next 6 bits.
  - **X-Axis Secondary**: Occupies the 8 bits following the byte containing the 6 button bits and 2 unused bits (note there is no Center Push).
  - **Y-Axis Secondary**: Occupies the next 8 bits.

Standard TxPDO1 message data per default mapping:

Start	Length	Desc.	Values
1.1	8 bits	Joystick X-Axis Primary (Signed Integer)	0 - Neutral 1 to 32h (1 to 50) - Right FFh to CEh (-1 to -50) - Left
2.1	8 bits	Joystick Y-Axis Primary (Signed Integer)	0 - Neutral 1 to 32h (1 to 50) - Up FFh to CEh (-1 to -50) - Down
3.1	8 bits	Center Twist	0 - Neutral 1 - Clockwise 2 - Counter-Clockwise
4.1	1 bit	Button 1 Status	0 - Not pressed 1 - Pressed
4.2	1 bit	Button 2 Status	0 - Not pressed 1 - Pressed
4.3	1 bit	Button 3 Status	0 - Not pressed 1 - Pressed
4.4	1 bit	Button 4 Status	0 - Not pressed 1 - Pressed
4.5	1 bit	Button 5 Status	0 - Not pressed 1 - Pressed
4.6	1 bit	Button 6 Status	0 - Not pressed 1 - Pressed
4.7	2 bits	Unused	0b00 - Ignore
5.1	8 bits	Joystick X-Axis Secondary (Signed Integer)	0 - Neutral 1 to 32h (1 to 50) - Right FFh to CEh (-1 to -50) - Left
6.1	8 bits	Joystick Y-Axis Secondary (Signed Integer)	0 - Neutral 1 to 32h (1 to 50) - Up FFh to CEh (-1 to -50) - Down
7.1	16 bits	Unused	All bits 0

6. Indicator and Backlight Brightness

The intensity of both the indicator and backlight brightness can be controlled via:

- OD2010sub01 and 02 for the indicator and backlight respectively.
- Placing the device in NMT state Operational where RxPDO2 will be read and the brightness controlled accordingly.

The indicators and backlights have two process variables, Value and Scalar. Each has a range from 0 to 255. The value is typically what is set to control the intensity. The final 8-bit PWM value is realized by multiplying the Value and Scalar together then loading the PWM with the upper byte. The Scalar is used for the purposes of matching up the device's intensities with those of other devices on the instrument cluster. Ex. setting all devices to 50% brightness will look uniform.

7. CANopen Specific Implementation

The following list details features that are and aren't available or implemented on the 3J Gen 2 CANopen devices.

- Remote Transmit Request (RTR) is not implemented.
- All baud rates as defined in CiA 301 are supported. These are:
    - 1000 kbit/s (1 Mbit/s)
    - 800 kbit/s
    - 500 kbit/s
    - 250 kbit/s (Default)
    - 125 kbit/s
    - 50 kbit/s
    - 20 kbit/s
    - 10 kbit/s
- The following PDO's are implemented:
    - Default TxPDO1: Conveys button, joystick, and encoder status information.
    - Default RxPDO1: Controls indicator states.
    - Default RxPDO2: Controls the indicator and backlight intensities.
    - SRDO1 -> Safety Tx
    - SRDO2 -> Safety Rx
- PDO Dynamic Mapping is not supported.
- SDO Block transfer is not supported.
- Layer Setting Services are supported for changing the following:
    - Node ID of the device.
    - Baud Rate.
- Node Guarding is not supported.
- Emergency Object is not supported.
- Storing and Loading communication and application parameters is supported using the corresponding Object Dictionary entries as defined in CiA 301.
- All application process variables are realized within the Manufacturer Specific Object Dictionary Entries between 2000h and 5FFFh.

8. CANopen Safety

CANopen Safety requires that the process variables, in this case the button status and indicator control, have complementary OD entries. Note that the brightness is not controlled via the CANopen Safety protocol. The complementary values for the process variables have an offset of 200h from the non-inverted entries. Both the non-inverted and inverted values of the process variable are transmitted/received on the bus with two unique COBIDs that vary with at least two bits. The non-inverted message's COBID is always even, leaving the complemented message's COBID as odd. On these devices, SRDO1 can only be configured for Tx and SRDO2 can only be configured for Rx.

8.1. Button Transmission

When transmitting, both messages are sent with the non-inverted encoding first, followed by the inverted approximately 5ms later. This is repeated periodically according to the refresh time (SCT) of the SRDO.

8.2. Indicator Reception

When receiving, both messages are received with the non-inverted encoding first, followed by the inverted. This must be done within the validation time (SRVT). Consecutive sets of messages need to be received within the refresh time (SCT).

8.3. Parameter Modification

All of the OD entries related to CANopen Safety can only be modified when the device is in NMT Pre-Op. Like with the standard PDOs, dynamic mapping is not available. SRDO1 (OD Entry 1301h) can only be configured for transmission and SRDO2 (OD Entry 1302h) can only be configured for reception. Each can effectively be disabled by configuring the direction as Invalid. Once all parameters are set as desired, the corresponding CRC needs to be updated for each SRDO where the CRC for SRDO1 and 2 are at OD Entry OD13FFsub01 and OD13FFsub02 respectively. Once the CRCs have been updated, the configuration needs to be validated by setting OD13FEsub00 to A5h. Once the device is returned to Operational mode, the changes should take effect.

To summarize, a successful process to utilize the SRDOs is to take the following steps:

1. Write valid direction values to OD1301sub01 and/or OD1302sub01
2. Optionally modify values at the other subindexes of those OD entries
3. Read from the device the CRC value(s) at OD13FFsub01 and/or OD13FFsub02
4. Write those respective freshly computed CRC values to OD13FFsub01 and/or OD13FFsub02
5. Write the A5h validation value to OD13FEsub00
6. Reinitiate NMT Operational mode

9. OD Entries - Information and Parameter Saving/Loading

The following entries are related to information about the device and the saving/loading of configuration variables.

9.1. 1000h: Device Type

The Device Type indicates a Manufacturer Specific object.

- 3J Devices: 0x334A0000 -> ASCII "3J" in the upper two bytes. A lower two bytes other than "0000" would indicate a non-generic device profile.

9.2. 1001h: Error Reg

Currently not used. Always read as zero.

9.3. 1008h: Device Name

3J Devices (ASCII strings reported):

- "Device Unassigned" (indicates a setting that is invalid for proper operation)
- "Proportional Joystick" (indicates Joystick Controller with single proportional joystick)
- "Dual Proportional Joystick" (indicates Joystick Controller with dual proportional joystick)
- "Digital Joystick with Encoder" (indicates Digital Joystick VDC)
- "Encoder Only" (indicates Rotary Encoder VDC)

9.4. 1009h: Hardware Version

ASCII values reported, representing the standard product part number per Grayhill's 3JYY-G2-X print according to the joystick type used (variable characters in the part number that do not have their values determined by the given joystick type are reported with the "_" literal):

- "3J_ÿ_5-G2-N3A_", where "ÿ" is a FFh byte (for Device Unassigned)
- "3J_2_5-G2-N3A_" (for Proportional Joystick)
- "3J_305-G2-N3A_" (for Dual Proportional Joystick)
- "3J_115-G2-N3A_" (for Digital Joystick with Encoder)
- "3J_0_5-G2-N3A_" (for Encoder Only)

9.5. 100Ah: Software Version

ASCII String "3JPRxxxx-y,r, mmm dd yyyy, hh:mm:ss". Indicates the device's firmware part number, revision, and the date and time the firmware was compiled.

9.6. 1010h: Save Parameters

Subidx 01: Save All Parameters

SubIdx 02: Save Comm Parameters

SubIdx 03: Save Application Parameters

SubIdx 04..FF: Manufacturer Specific (not used)

When reading subidx 1, 2, or 3, the 32 bit value of 1 will be returned, indicating that the corresponding Save option is implemented. To save the corresponding set of parameters, write the 32 bit value of 0x65766173, ASCII string of "save" where the "s" character occupies the LSB.

9.7. 1011h: Load Parameters

Subidx 01: Load All Parameters

SubIdx 02: Load Comm Parameters

SubIdx 03: Load Application Parameters

SubIdx 04..FF: Manufacturer Specific (not used)

When reading subidx 1..3, the 32 bit value of 1 will be returned, indicating that the corresponding Load option is implemented. To load the corresponding set of parameters, write the 32 bit value of 0x64616F6C, ASCII string of “load” where the “d” character occupies the LSB.

9.8. 1017h: Producer Heartbeat

16 bit value indicating the period in ms at which the heartbeat message is transmitted. A value of zero specifies the heartbeat is not transmitted periodically.

Default Value: 0

9.9. 1018h: Identity Object

Subidx 01, Vendor ID

Default Value: 0x00307 -> Grayhill Inc.

Subidx 02, Product Code

Default Value: 0x0000334A -> “3J”

Subidx 03, Revision Number

Default Value: xxxxx

Subidx 04, Serial Number

Default Value: xxxxxx

10. OD Entries - PDO Relevant

The following entries are related to the standard PDOs.

10.1. 1400h: RxPDO1 Comm, Indicator Control

Subidx 01: COBID

Default: 0x00000200 + NodeID

Subidx 02: Type

Default: 0xFF, Event driven, MFG Specific.

10.2. 1401h: RxPDO2 Comm, Indicator and Backlight Brightness Control

Subidx 01: COBID

Default: 0x00000300 + NodeID

Subidx 02: Type

Default: 0xFF, Event driven, MFG Specific.

10.3. 1600h: RxPDO1 Mapping for the Indicators

Subidx 01: Mapping of Indicator 1

Default: 0x20030101

Subidx 02: Mapping of Indicator 2

Default: 0x20030201

Subidx n: Mapping of Indicator n up to 3*Button Count, where n is two hex digits

Default: 0x2003[n]01

10.4. 1601h: RxPDO2 Mapping, Indicator/Backlight Brightness Control

Subidx 01: Mapping of Indicator Brightness Setting

Default: 0x20100108

Subidx 02: Mapping of Backlight Brightness Setting

Default: 0x20100208

10.5. 1800h: TxPDO1 Comm, Buttons, Joystick, Encoder

Subidx 01: COBID

Default: 0x40000180 + NodeID

Subidx 02: Type

Default: 0xFF, Event driven, MFG Specific

Subidx 03: Inhibit Time

Units of 100us. Consecutive messages will not be transmitted faster than this period.

Default: 0us

*Note: Although the parameter is in units of 100us, the resolution is limited to 1ms

Subidx 04: Reserved

Not implemented

Subidx 05: Event Timer

Units of 1ms. Maximum interval for PDO Transmission. A value of zero specifies that the PDO will only transmit on a status change (Send on Event).

Default: 0ms

10.6. 1A00h: TxPDO1 Mapping **Keypad**

Subidx 00: Highest Index Count

This value is equal to the number of buttons on the device.

Subidx nn: Mapping of Button nn

Default: 0x2001nn01

10.7. 1A00h: TxPDO1 Mapping **Proportional Joystick**

Subidx 00: Highest Index Count

Default value: 0x0A

Subidx 01: Mapping of X-Axis

Default: 0x20040108

Subidx 02: Mapping of Y-Axis

Default: 0x20040208

Subidx 03: Mapping of Twist value

Default: 0x20040308

Subidx 04..09: Mapping of Button Status

Default: 0x20010[n]01 Where n has the value of 1..6. Subidx 04 corresponds to button 1 and so on.

Subidx 0A: Mapping of Joystick's Center Push

Default: 0x20040401

10.8. 1A00h: TxPDO1 Mapping **Dual Proportional Joystick**

Subidx 00: Highest Index Count

Default value: 0x0C

Subidx 01: Mapping of X-Axis

Default: 0x20040108

Subidx 02: Mapping of Y-Axis

Default: 0x20040208

Subidx 03: Mapping of Twist value

Default: 0x20040308

Subidx 04..09: Mapping of Button Status

Default: 0x20010[n]01 Where n has the value of 1..6. Subidx 04 corresponds to button 1 and so on.

Subidx 0A: Mapping of Joystick's Center Push (center push does not exist for dual version)

Default: 0x20040401

Subidx 0B: Mapping of X-Axis Secondary

Default: 0x20040508

Subidx 0C: Mapping of Y-Axis Secondary

Default: 0x20040608

10.9. 1A00h: TxPDO1 Mapping **Digital Joystick with Encoder**

Subidx 00: Highest Index Count

Default value: 0x0D

Subidx 01: Mapping 16 bit Encoder Value

Default: 0x20000110

Subidx 02: Encoder Turn Direction

Default: 0x20000208

Subidx 03..08: Mapping of Button Status

Default: 0x20010[n]01 Where n has the value of 1..6. Subidx 03 corresponds to button 1 and so on.

Subidx 09: Mapping of Encoder's Center Push Status

Default: 0x20000301

Subidx 0A..0D: Mapping of Digital Joysticks Direction.

Default: 0x20020[n]01 where n = {1:Up, 2:Down, 3:Right, 4:Left}

10.10. 1A00h: TxPDO1 Mapping **Encoder Only**

Subidx 00: Highest Index Count

Default value: 0x09

Subidx 01: Mapping 16 bit Encoder Value

Default: 0x20000110

Subidx 02: Encoder Turn Direction

Default: 0x20000208

Subidx 03..08: Mapping of Button Status

Default: 0x20010[n]01 Where n has the value of 1..6. Subidx 03 corresponds to button 1 and so on.

Subidx 09: Mapping of Encoder's Center Push Status

Default: 0x20000301

11. OD Entries - Safety Relevant

The following are related to the CANopen Safety protocol.

11.1. 1301h: SRDO Tx

Subidx 01: Information Direction

0: Does not exist / Invalid (Default)
1: Valid for Tx
2: Valid for Rx (Not Allowed)

Subidx 02: SCT, Refresh Time

This parameter specifies the period in ms with which the non-inverted message is transmitted.

Default Value: 25ms (when configured for Tx)

Subidx 03: SRVT

Not used when configured for Tx.

Default Value: 0ms

Subidx 04: Transmission Type

Default Value: 0xFE -> Event driven, mfg specific. RO

Subidx 05: COBID Non-inverted process variables

Default Value: 0x000000FF + (2 x NodeID)

Subidx 06: COBID Inverted process variables

Default Value: 0x00000100 + (2 x NodeID)

11.2. 1302h: SRDO Rx

Subidx 01: Information Direction

0: Does not exist / Invalid (Default)
1: Valid for Tx (Not Allowed)
2: Valid for Rx

Subidx 02: SCT, Refresh Time

This parameter specifies the period in ms within which consecutive message sets need to be received.

Default Value: 50ms (when configured for Rx)

Subidx 03: SRVT

This parameter specifies the period in ms within which the inverted message needs to be received after receiving the non-inverted message.

Default Value: 20ms

Subidx 04: Transmission Type

Default Value: 0xFE -> Event driven, mfg specific. RO

Subidx 05: COBID Non-inverted process variables

Default Value: 0x00000101 + (2 x NodeID)

Subidx 06: COBID Inverted process variables

Default Value: 0x00000102 + (2 x NodeID)

11.3. 1381h: Mapping Tx

The actual message data for the non-inverted message is identical to that of TxPDO1. All of the data is simply inverted for the second, inverted message. All process variables in the Object Dictionary range from 0x2000 to 0x2004 have a complement at 0x2200 to 0x2204 respectively.

Odd number indices correspond to the non-inverted process variables and even number indices correspond to the inverted counterpart. As an example, the mapping for the Encoder Only would be as follows:

Subidx 01.	0x20000110, Non-inverted encoder value
Subidx 02.	0x22000110, Inverted encoder value.
Subidx 03.	0x20000208, Non-inverted direction.
Subidx 04.	0x22000208, Inverted direction.

And so on.

11.4. 1382h: Mapping Rx

Odd number indices correspond to the non-inverted value OD2003 and even number of indices correspond to OD2203 where:

Subidx 01.	(# of buttons)*(3 indicators per button)*2
Subidx 02.	0x20030101
Subidx 03.	0x22030101
Subidx 04.	0x20030201
Subidx 05.	0x22030201

And so on.

11.5. 13FEh: Configuration Valid

This object indicates valid configurations. Whenever any parameter related to Safety is modified, this flag is cleared to zero. After making modifications, the 8-bit value of A5h needs to be written to subidx 00.

11.6. 13FFh: Safety Configuration Checksum

Each subidx corresponds to an SRDO where OD13FFsub01 corresponds to SRDO1 and OD13FFsub02 corresponds to SRDO2. Reading the subidx causes the CRC to be computed and returned. The CRC data of an SRDO consists of all of the communication parameters and mapping variables. The computed value is compared against the stored value. If the CRCs do not match, the SRDO is invalid and will not become active.

Subidx 00: Highest subidx supported

Default value: 0.

Subidx 01: CRC for SRDO1 Tx

Default value: 0

Subidx 02: CRC for SRDO2 Rx

Default Value: 0

12. OD Entries: Manufacturer Specific.

The following entries are Manufacturer Specific.

12.1. 2000h: Encoder

Subidx 00: Highest subidx value

Default: 7

Subidx 01: Encoder's Current Value

A 16 bit signed integer ranges between the minimum and maximum values specified in sub entries 05 and 06 respectively.

Subidx 02: Encoder Direction

An 8 bit signed character having the value of ether 1 or -1 (0xFF).

Subidx 03: Encoder Center Push Status

An 8 bit unsigned byte having the values of 1 or 0 for Pressed or not.

Subidx 04: Revolutions.

A 16 bit signed integer that increments when the data rolls over from Max to Min and decrements when going from Min to Max.

Subidx 05: Encoder's Max Value

A 16 bit signed integer. Note that negative values must be entered as the 2's complement.

Subidx 06: Encoder's Min Value

A 16 bit signed integer. Note that negative values must be entered as the 2's complement.

Subidx 07: Rollover Enabled.

0 : Rollover disabled. The data stops at the Min and Max without wrapping around.

1 : Rollover Enabled. The data rolls over from the Max to the Min and vice versa.

12.2. 2001h: Button Status.

Subidx 00: Highest subidx value.

Effectively the number of buttons.

Subidx n: The button number

An 8 bit number having the value of 0 or 1 for released or pressed respectively for button n.

12.3. 2002h: Digital Joystick Process Variables.

Subidx 00: Highest subidx value.

Default: 4

Subidx 01..04: Joystick Direction.

Each subidx has an 8 bit variable having a value of either 0 or 1 indicating if the joystick is pushed in the corresponding direction. Indices 1 through 4 represent Down, Up, Left, and Right, respectively.

12.4. 2003h: Indicator Status.

Subidx 00: Highest sub index value.

The value is equal to the number of Indicator LEDs on the device.

Subidx n: The Status of the nth Indicator.

Values can be 0 or 1 for unlit and illuminated respectively

12.5. 2004h: Proportional Joystick Process Variables.

Subidx 00: Highest sub index value.

Default: 4

Subidx 01: X-Axis

An 8 bit signed value indicating the movement in the X direction. Positive values indicate a right push.

Subidx 02: Y-Axis

An 8 bit signed value indicating the movement in the Y direction. Positive values indicate an upward push.

Subidx 03: Twist Direction

An 8 bit signed value indicating twist direction. Values range from -1 to +1 where the positive value indicates a clock-wise twist.

Subidx 04: Center Button Press

An 8 bit value indicating the state of the joysticks center push button where 1 indicates pressed and 0 released.

Subidx 05: X-Axis Secondary (Dual Proportional Joystick only)

An 8 bit signed value indicating the movement in the X direction. Positive values indicate a right push. This secondary value is a double-check of the value in Subidx 01, generated by a different sensor. Its value should be similar, but likely will not always be identical.

Subidx 06: Y-Axis Secondary (Dual Proportional Joystick only)

An 8 bit signed value indicating the movement in the Y direction. Positive values indicate an upward push. This secondary value is a double-check of the value in Subidx 02, generated by a different sensor. Its value should be similar, but likely will not always be identical.

12.6. 2200h to 2204h Process Variable Complements.

These are identical to corresponding values at 2000h to 2204h with the exception that the process variables are inverted. Only the sub-indices that are mapped to PDOs are implemented. For example, since the encoder's max value at OD2000sub05 is not transmitted via a PDO or SRDO, it has no complement.

12.7. 5FFEh: **Production Configuration and Control

The sub-indices here are used for production configuration and control. Each sub index corresponds to a configuration variable that can be read and written to. In order to write new values, the device must first be unlocked by writing the character string “conf” to sub index 0xFF. Some of the values in this table may be changed by the typical CANopen procedures for modifying Object Dictionary entries.

Parameter Description	OD Sub-Index	Default Value	Units/Value	Raw Data (hex)
Node ID	1 (0x01)	10	Node-ID	0A
Baud Rate	2 (0x02)	3	250Kbps	03
Button Count	3 (0x03)	6	6 Buttons	06
Device Type	4 (0x04)	0	Undefined	00
Encoder max value	5 (0x05)	10	+10 Detents	0A-00
Encoder min value	6 (0x06)	65526	-10 Detents	F6-FF
Encoder rollover	7 (0x07)	1	Rollover Enabled	01
Ind brightness	8 (0x08)	255	0.4%	FF
Bklt brightness	9 (0x09)	0	0.4%	00
Ind scalar	10 (0x0A)	255	0.4%	FF
Bklt scalar	11 (0x0B)	255	0.4%	FF
Heartbeat period	12 (0x0C)	0	1ms	00-00
TPDO1 COBID	13 (0x0D)	0*	COBID	00-00-00-00
RPDO1 COBID	14 (0x0E)	0*	COBID	00-00-00-00
RPDO2 COBID	15 (0x0F)	0*	COBID	00-00-00-00
TPDO inh time	16 (0x10)	0	x 100us	00
TPDO event time	17 (0x11)	0	x 1ms	00-00
Vendor ID	18 (0x12)	0x0307	Grayhill Inc.	07-03-00-00
Product code	19 (0x13)	0x334A	3J	4A-33-00-00
Revision	20 (0x14)	0	Revision	FF-FF-FF-FF
Serial number	21 (0x15)	0	Serial Number	FF-FF-FF-FF
Safety Tx dir	22 (0x16)	0	DISABLED	00
Safety Tx sct	23 (0x17)	25	x 1ms	19-00
Safety Tx cobid1	24 (0x18)	0*	COBID	00-00-00-00
Safety Tx cobid2	25 (0x19)	0*	COBID	00-00-00-00
Safety Tx crc	26 (0x1A)	65535	CRC	FF-FF
Safety Rx dir	27 (0x1B)	0	DISABLED	00
Safety Rx sct	28 (0x1C)	50	x 1ms	32-00
Safety Rx srvt	29 (0x1D)	20	x 1ms	14
Safety Rx cobid1	30 (0x1E)	0	COBID	00-00-00-00
Safety Rx cobid2	31 (0x1F)	0	COBID	00-00-00-00
Safety Rx crc	32 (0x20)	65535	CRC	FF-FF
Product Date Code	33 (0x21)	65535	DC	FF-FF
Grayhill id	34 (0x22)	0	mm/dd/yyyy hh:mm:ss	FF-FF-FF-FF
Diag blink period	35 (0x23)	0	100ms	00
GP flags	36 (0x24)	0	Auto Start Disabled	00-00-00-00
Last parameter	37 (0x25)	1437217740	EOL MARKER	CC-33-AA-55

****Note**: When the value is zero, the actual COBID is calculated by adding the Node ID to the default COBID. Ex. TPDO1 becomes (40000180h + 0Ah = 4000018Ah). Refer to CiA-301 for more detail.*

Grayed out rows are not used.

12.8. 5FFFh: **Memory Access

This object gives the ability to read either RAM, Flash or special chip ID memory. Memory is read by first performing an SDO Write of the 32 bit address to sub-index 0xFF. When an SDO Read of sub-index 0xFF is executed, the 32 bit value at the written address is returned. The address is restricted to the available Flash and RAM ranges of the uC. The following sub-indices are used to read special memory locations.

Subidx 01: uC's Electronic Signature.

This is a 12 byte long data field containing a unique number for each uC.

Subidx 02: uC's Flash Size.

This is a 16 bit value indicating the size of the uC's flash. The value returned is multiplied by 1024 to get the actual size.

Subidx 03: uC's Bootloader.

ASCII String "3JPRxxxx-y,r, mmm dd yyyy, hh:mm:ss". Indicates the device's bootloader part number, revision, and the date and time the bootloader was compiled.

Subidx FF: Memory Read.

- Writing a 32 bit address selects that address as the location at which to read.
- Reading returns the 32 bit value at the selected address.

13. Low Power Mode

OD117Fsub00 contains the number of objects, (2 by default).

OD117Fsub01 contains the wake reason code (for use in "sleep objection" service), (0 by default).

OD117Fsub02 contains the sleep reason code (for use in "request sleep" service), (0 by default).

Sending the following "set sleep" message will cause the module to initiate sleep (and enter a low power state).

691h 02h 00h 00h 00h 00h 00h 00h 00h

Sending the following "set alive" message will cause the module to initiate alive status if it was preparing for sleep.

691h 82h 00h 00h 00h 00h 00h 00h 00h

A button press or CAN message will wake the module from low power mode and cause it to enter Pre-Op mode and periodically send 690h 00h 00h 00h 00h 00h 00h 00h 00h, indicating it has woken up. Sending a command to enter Operational mode at this time will cause it to resume active operation.

14. How To's

The following are examples and explanations on the usage of the device. Note that unless indicated otherwise, all messages are standard with 11 bit identifiers.

14.1. Entering Operating Mode and Using the Device (non-Safety)

When the device is first powered, it transmits the Boot message. The CAN-ID is 0x700+NodeID and the data length is 1 with the byte value equal to zero (example with NodeID 0x0A):

```
ID=0x70A, DLC=1, Data={0x00}
```

Shortly afterwards the Heartbeat message is transmitted indicating the device is in Pre-Op mode.

```
ID=0x70A, DLC=1, Data={0x7F}
```

To place the device in Operational Mode, the NMT Master needs to send the command to do so.

```
ID=0x000, DLC=2, Data={0x01,0x00}
```

Now, pressing buttons will transmit the TxPDO with those buttons' corresponding bit each set to 1. For example, pressing the second button, assuming the COBID values are the defaults, will cause the following message to be transmitted, assuming an Encoder Only type module.

```
ID=0x18A, DLC=4, Data={Byte1, Byte2, Byte3, 0x02}
```

Controlling the indicators is done by sending the RxPDO where each bit in the data field corresponds to an indicator LED. The following message will turn on all three indicators on the second button.

```
ID=0x20A, DLC=3, Data={0x38,0x00,0x00}
```

To change the brightness of the backlights and indicators to 50% and 10% respectively, then both back to 100%, send the following messages.

```
ID=0x30A, DLC=2, Data={0x19,0x80} -> Dimmed
ID=0x30A, DLC=2, Data={0xFF,0xFF} -> Full
```

14.2. CANopen Safety

When using the CANopen Safety functionality of the device, it may be desired to disable the standard PDOs. This is done by invalidating the corresponding COBIDs for the TxPDO and RxPDO by writing a value with the most significant bit set (0x80000000) then executing the following steps with the device in Pre-Op mode:

1. Enter Pre-Op mode, if not already done.
2. Invalidate the standard PDOs.
3. Set up each SRDO as needed.
4. Validate the SRDO directions by writing the corresponding value to sub index 01.
5. Read the Configuration CRC for each of the SRDOs. The CRC value that is reported gets freshly computed from the SRDO's parameter values each time.
6. Write the CRC values back to the corresponding SRDOs. This is the value that is checked for validation.
7. Validate the entire module by writing 0xA5 to OD13FEsub00.
8. Put the device in Operational mode.

At this point when the device enters Operational mode, the Tx messages will transmit periodically according to the SCT value.

The CAN traffic shown below is an example of the steps listed above. SRDO1 (Tx) is configured to periodically send its messages every 100ms. SRDO2 (Rx) is configured to receive messages within 150ms of each other. Complementary messages need to be received within the default 20ms of each other. The messages follow the steps listed above and are sent from a device acting as the NMT Master. The example device has the default node-id of 10 (0x0A). All values are in hex:

```
ID=000, DLC=2, Data=80 00                   ; NMT Pre-Op
ID=60A, DLC=8, Data=23 00 14 01 00 00 00 80 ; Invalidate RxPDO
ID=60A, DLC=8, Data=23 00 18 01 00 00 00 80 ; Invalidate TxPDO
ID=60A, DLC=8, Data=2B 01 13 02 64 00 00 00 ; SRDO1->SCT=100ms
ID=60A, DLC=8, Data=2B 02 13 02 96 00 00 00 ; SRDO2->SCT=150ms
ID=60A, DLC=8, Data=2F 01 13 01 01 00 00 00 ; Validate SRDO1 for Tx
ID=60A, DLC=8, Data=2F 02 13 01 02 00 00 00 ; Validate SRDO2 for Rx
ID=60A, DLC=8, Data=40 FF 13 01 00 00 00 00 ; Read SRDO1 CRC
ID=58A, DLC=8, Data=4B FF 13 01 EA 05 00 00 : Response CRC=0x05EA
ID=60A, DLC=8, Data=40 FF 13 02 00 00 00 00 ; Read SRDO2 CRC
ID=58A, DLC=8, Data=4B FF 13 02 D4 A2 00 00 : Response CRC=0xA2D4
ID=60A, DLC=8, Data=2B FF 13 01 EA 05 00 00 ; Set SRDO1 CRC
ID=60A, DLC=8, Data=2B FF 13 02 D4 A2 00 00 ; set SRDO2 CRC
ID=60A, DLC=8, Data=2F FE 13 00 A5 00 00 00 ; Validate Config
ID=000, DLC=2, Data=01 00                   ; NMT Operational
```

If performed correctly, and if the example device is a Proportional Joystick type module, the following two messages should be transmitted every 100ms

```
ID=113, DLC=4, Data=00 00 00 00 ; no buttons pressed
ID=114, DLC=4, Data=FF FF FF FF ; 1's complement of the above data.
```

14.3. Change PDO COBIDs

The following messages are involved in changing the COBIDs of the PDOs. Note that by default the lower 7 bits of each COBID equals the device's NodeID. If the Node-ID changes via the LSS, the COBIDs will follow. However, once a value is written to the COBIDs, even if it matches the default, changing the NodeID will not cause the COBIDs to change. The following example assumes the default NodeID of 0x0A. TxPDO1's COBID will change to 0x123.

```
ID=60A, DLC=8, Data=23 00 18 01 00 00 00 80 ; COBID Invalidated
ID=58A, DLC=8, Data=60 00 18 01 00 00 00 00 ; Acknowledgement from device
ID=60A, DLC=8, Data=23 00 18 01 23 01 00 40 ; New COBID = 0x123, Note the RTR flag set.
ID=58A, DLC=8, Data=60 00 18 01 00 00 00 00 ; Acknowledgement from device
```

When changing a COBID for an RxPDO, the RTR flag (bit 30) does not need to be set and is ignored. In order for the changes to be permanent, the Save All or Save Comm must be sent by writing the ASCII string 'save' to OD1010sub01 or OD1010sub02 respectively.

```
ID=60A, DLC=8, Data=23 10 10 01 73 61 76 65; Save All
or
ID=60A, DLC=8, Data=23 10 10 02 73 61 76 65; Save Comm
```

14.4. Change NodeID Using LSS

The following steps are taken to change the NodeID of the device.

1. Put the device in NMT Stop
2. Send the LSS command Switch State Config.
3. Set the desired Node ID
4. Optionally save the Node ID to non-volatile memory.
5. Send the LSS command Switch State Wait
6. Send the NMT command Reset Comm.

The following CAN traffic is an example of the above steps.

```
ID=000, DLC=2, Data=02 00                      ; NMT Stop
ID=7E5, DLC=8, Data=04 01 00 00 00 00 00 00  ; Switch State Config
ID=7E5, DLC=8, Data=11 0B 00 00 00 00 00 00  ; Sets NodeID to 0x0B
ID=7E4, DLC=8, Data=11 00 00 00 00 00 00 00  ; Device Response
ID=7E5, DLC=8, Data=17 00 00 00 00 00 00 00  ; Save to Non-Volatile Memory
ID=7E4, DLC=8, Data=17 00 00 00 00 00 00 00  ; Device Response
ID=7E5, DLC=8, Data=04 00 00 00 00 00 00 00  ; Switch State Wait
ID=000, DLC=2, Data=82 00                     ; NMT Reset Comm
ID=70B, DLC=1, Data=00                         ; NMT Boot with new NodeID
```

14.5. Change Baud Rate Using LSS

The following steps are taken to change the baud rate.

1. Put the device in NMT Stop
2. Send the LSS command Switch State Config
3. Send the command with the value indicating the desired baud rate
4. Optionally send the command to store to non-volatile memory
5. Send the command to activate the new bit timing
6. Send LSS command Switch State Wait
7. Put the device in NMT Pre-Op

Note that once step 4 is executed, the device can be power cycled and the new baud rate will remain in effect. The following CAN traffic is an example of the above procedure. It assumes the current baud rate is 250 kbps and is being changed to 1000 kbps. The Switch Delay is configured for 3000 ms. So 3000 ms after executing Step 5, the device will switch to the new baud rate of 1000 kbps.

```
ID=000, DLC=2, Data=02 00                    ; NMT Stop
ID=7E5, DLC=8, Data=04 01 00 00 00 00 00 00 ; Switch State Config
ID=7E5, DLC=8, Data=13 00 00 00 00 00 00 00 ; Set to index 0, 1000kbps
ID=7E5, DLC=8, Data=17 00 00 00 00 00 00 00 ; Save to Non-Volatile Memory
ID=7E5, DLC=8, Data=15 B8 0B 00 00 00 00 00 ; Waits 3000 ms (0xBB8)
ID=7E5, DLC=8, Data=04 00 00 00 00 00 00 00 ; Switch State Wait
ID=000, DLC=2, Data=01 00                    ; NMT Pre-Op
```