AN408 – Configuring a GIM 140 with Protocol Converter

Introduction

Axiomatic Technologies' module AX140200 can be used in conjunction with the Baumer GIM 140 inclination sensor to convert messages from CANopen® to J1939 without a CANopen® master on the network. The module can be modified by the user to function as desired.

GIM 140 TPDO

TPDO1		GIM140 Angle Sensor A Tx COB ID 0x1DA
Transmission Rate		25 ms
Data Length		6 bytes
Byte	**Bits**	**Parameter**
1 & 2	1-8 1-8	Sensor Temperature
3 & 4	1-8 1-8	Slope X
5 & 6	1-8 1-8	Slope Y

TPDO1		GIM140 Angle Sensor B Tx COB ID 0x1DB
Transmission Rate		25 ms
Data Length		6 bytes
Byte	**Bits**	**Parameter**
1 & 2	1-8 1-8	Sensor Temperature
3 & 4	1-8 1-8	Slope X
5 & 6	1-8 1-8	Slope Y

It is configured to receive the GIM 140 slope sensor data from COB ID 0x1DAh and 0x1DBh on the CANopen® port and forward it to the J1939 port. The data is transmitted on PGNs 0xFF14h and 0xFF15h.

Axiomatic Technologies Oy
Höytämöntie 6
33880 LEMPÄÄLÄ, Finland
Tel. +358 103 375 750
salesfinland@axiomatic.com
www.axiomatic.fi

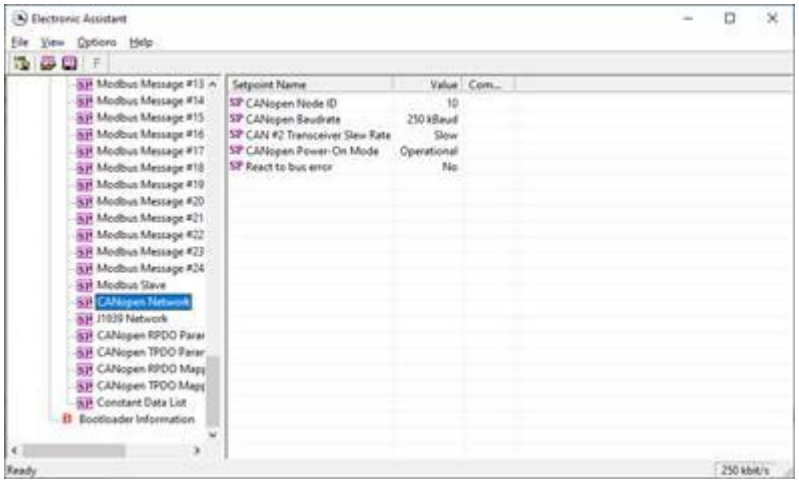
Axiomatic Technologies Corporation
1445 Courtneypark Dr. E.
Mississauga, ON Canada L5T 2E3
Tel. 1 905 602 9270
sales@axiomatic.com
www.axiomatic.com

Setting Up Transmission

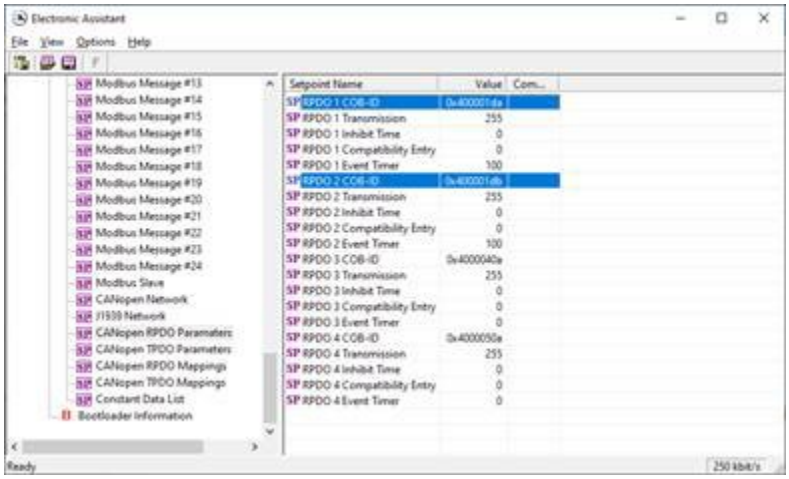
The setpoint will transmit the data exactly as it is received on the CANopen® port.

Here are the steps to set up the transmission:

1. In the CANopen® Network settings:
   a. Set the baud rate (250kbps).
   b. Set the Power on mode to 'Operational'.



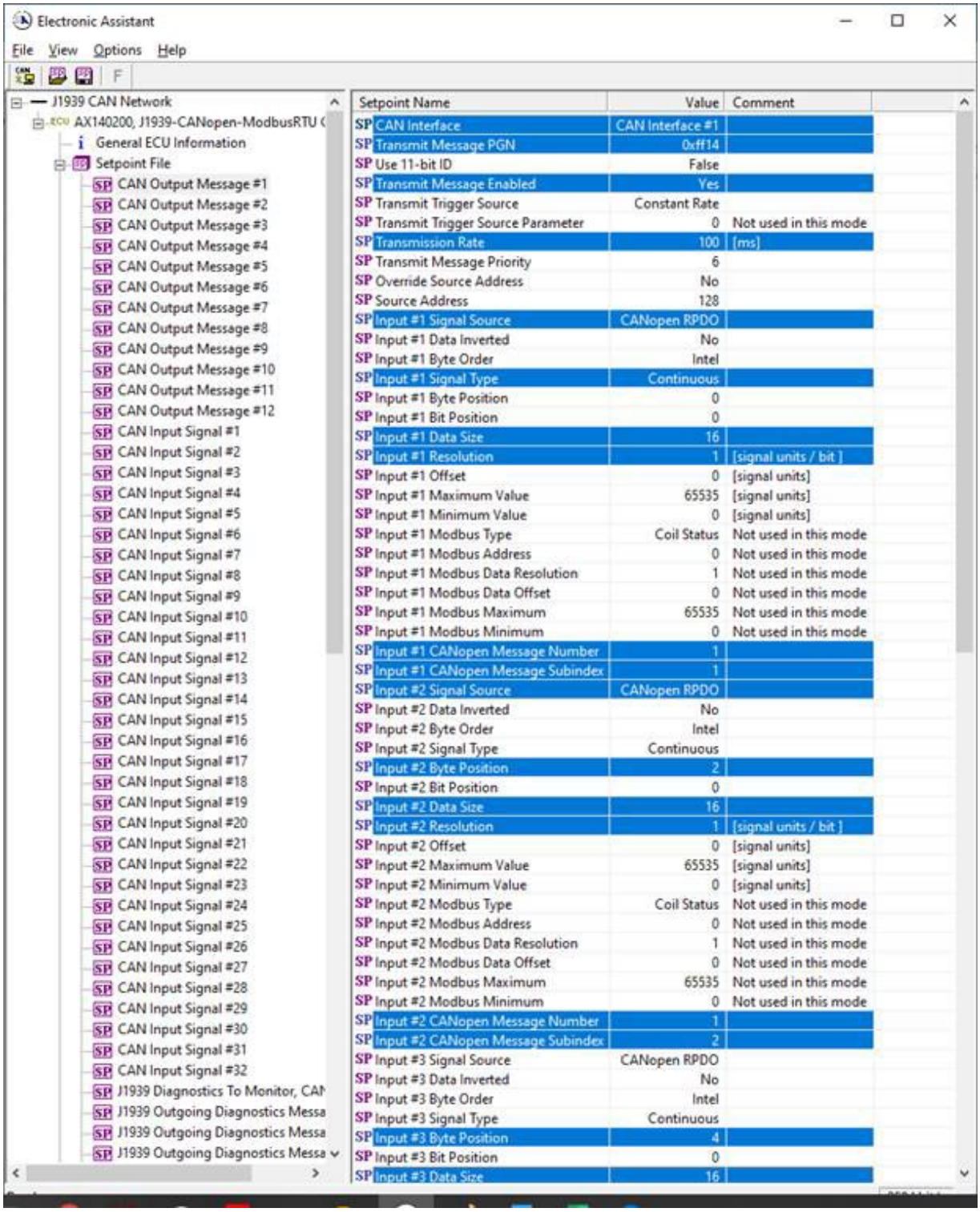
2. Go to the CANopen® RPDO parameters:
   a. Set the COB-ID for RPDO1 to "1DA" (Angle Sensor A TPDO1).
   b. Set the COB-ID for RPDO2 to "1DB" (Angle Sensor B TPDO1).
   c. Set the event timer for RPDO1 and 2 to 100 (ms).



3. Go to the CANopen® RPDO Mappings:
   a. Change the number of mapped objects in RPDO1 to three.
   b. Map three 2 byte objects (Sensor temp, Slope X, Slope Y)(Angle Sensor A) to Mapping entries 1 to 3 of RPDO1.
      Where in the Mapping entry 0x2000h (CAN RX data 1) ,0x01h (Sub index 1), 0x10h(2 bytes).
   c. Change the number of mapped objects in RPDO2 to three.
   d. Map three 2 byte objects (Sensor temp, Slope X, Slope Y)(Angle Sensor B) to Mapping entries 1 to 3 of RPDO2.

Setpoint Name	Value	Com...
SP RPDO 1 Number of Mapped Objects	3	
SP RPDO 1 Mapping Entry #1	0x20000110	
SP RPDO 1 Mapping Entry #2	0x20000210	
SP RPDO 1 Mapping Entry #3	0x20000310	
SP RPDO 1 Mapping Entry #4	0x00	
SP RPDO 2 Number of Mapped Objects	3	
SP RPDO 2 Mapping Entry #1	0x20010110	
SP RPDO 2 Mapping Entry #2	0x20010210	
SP RPDO 2 Mapping Entry #3	0x20010310	
SP RPDO 2 Mapping Entry #4	0x00	
SP RPDO 3 Number of Mapped Objects	1	
SP RPDO 3 Mapping Entry #1	0x20020120	
SP RPDO 3 Mapping Entry #2	0x00	
SP RPDO 3 Mapping Entry #3	0x00	
SP RPDO 3 Mapping Entry #4	0x00	
SP RPDO 4 Number of Mapped Objects	1	
SP RPDO 4 Mapping Entry #1	0x20030120	
SP RPDO 4 Mapping Entry #2	0x00	
SP RPDO 4 Mapping Entry #3	0x00	
SP RPDO 4 Mapping Entry #4	0x00	

4. Go to the CAN Output Message #1:
   a. Set the CAN interface to “#1”.
   b. Set the Transmit Message PGN.
   c. Set the Transmit Message Enabled to “Yes”.
   d. Set the transmission rate to “100ms”.
   e. Set the Input #1 Signal Source to “CANopen® RPDO”.
   f. Set the Signal Type to “Continuous”.
   g. Set the Input #1 byte position. “0”.
   h. Set the Data Size to “16” bits (2 bytes).
   i. Set the Data Resolution. Default = “1”.
   j. Set the CANopen® Message number “1” (Sensor Temp).
   k. Set the CANopen® Message subindex “1” (Sensor Temp).
   l. Repeat steps e to f to set up Slope X and Y for sensor A.
   m. For each new parameter, the byte position should be incremented by 2.
   n. For each new parameter, the CANopen® Message subindex should be increased by “1”.


Electronic Assistant
File View Options Help
J1939 CAN Network
AX140200, J1939-CANopen-ModbusRTU
General ECU Information
Setpoint File
CAN Output Message #1
CAN Output Message #2
CAN Output Message #3
CAN Output Message #4
CAN Output Message #5
CAN Output Message #6
CAN Output Message #7
CAN Output Message #8
CAN Output Message #9
CAN Output Message #10
CAN Output Message #11
CAN Output Message #12
CAN Input Signal #1
CAN Input Signal #2
CAN Input Signal #3
CAN Input Signal #4
CAN Input Signal #5
CAN Input Signal #6
CAN Input Signal #7
CAN Input Signal #8
CAN Input Signal #9
CAN Input Signal #10
CAN Input Signal #11
CAN Input Signal #12
CAN Input Signal #13
CAN Input Signal #14
CAN Input Signal #15
CAN Input Signal #16
CAN Input Signal #17
CAN Input Signal #18
CAN Input Signal #19
CAN Input Signal #20
CAN Input Signal #21
CAN Input Signal #22
CAN Input Signal #23
CAN Input Signal #24
CAN Input Signal #25
CAN Input Signal #26
CAN Input Signal #27
CAN Input Signal #28
CAN Input Signal #29
CAN Input Signal #30
CAN Input Signal #31
CAN Input Signal #32
J1939 Diagnostics To Monitor, CAN
J1939 Outgoing Diagnostics Messa
J1939 Outgoing Diagnostics Messa
J1939 Outgoing Diagnostics Messa
Setpoint Name | Value | Comment
CAN Interface | CAN Interface #1
Transmit Message PGN | 0xff14
Use 11-bit ID | False
Transmit Message Enabled | Yes
Transmit Trigger Source | Constant Rate
Transmit Trigger Source Parameter | 0 | Not used in this mode
Transmission Rate | 100 | (ms)
Transmit Message Priority | 6
Override Source Address | No
Source Address | 128
Input #1 Signal Source | CANopen RPDO
Input #1 Data Inverted | No
Input #1 Byte Order | Intel
Input #1 Signal Type | Continuous
Input #1 Byte Position | 0
Input #1 Bit Position | 0
Input #1 Data Size | 16
Input #1 Resolution | 1 | [signal units / bit]
Input #1 Offset | 0 | [signal units]
Input #1 Maximum Value | 65535 | [signal units]
Input #1 Minimum Value | 0 | [signal units]
Input #1 Modbus Type | Coil Status | Not used in this mode
Input #1 Modbus Address | 0 | Not used in this mode
Input #1 Modbus Data Resolution | 1 | Not used in this mode
Input #1 Modbus Data Offset | 0 | Not used in this mode
Input #1 Modbus Maximum | 65535 | Not used in this mode
Input #1 Modbus Minimum | 0 | Not used in this mode
Input #1 CANopen Message Number | 1
Input #1 CANopen Message Subindex | 1
Input #2 Signal Source | CANopen RPDO
Input #2 Data Inverted | No
Input #2 Byte Order | Intel
Input #2 Signal Type | Continuous
Input #2 Byte Position | 2
Input #2 Bit Position | 0
Input #2 Data Size | 16
Input #2 Resolution | 1 | [signal units / bit]
Input #2 Offset | 0 | [signal units]
Input #2 Maximum Value | 65535 | [signal units]
Input #2 Minimum Value | 0 | [signal units]
Input #2 Modbus Type | Coil Status | Not used in this mode
Input #2 Modbus Address | 0 | Not used in this mode
Input #2 Modbus Data Resolution | 1 | Not used in this mode
Input #2 Modbus Data Offset | 0 | Not used in this mode
Input #2 Modbus Maximum | 65535 | Not used in this mode
Input #2 Modbus Minimum | 0 | Not used in this mode
Input #2 CANopen Message Number | 1
Input #2 CANopen Message Subindex | 2
Input #3 Signal Source | CANopen RPDO
Input #3 Data Inverted | No
Input #3 Byte Order | Intel
Input #3 Signal Type | Continuous
Input #3 Byte Position | 4
Input #3 Bit Position | 0
Input #3 Data Size | 16

RPDO Mapping Parameters

To access the RPDO mapping parameters, users must first destroy the RPDO. Users can do this by writing a 'C' value as the leading byte of the COB-ID:

1. Set RPDO1 'COB-ID' = 0x**C**000020A (destroy RPDO1)

SP RPDO 1 COB-ID	0xc000020a
SP RPDO 1 Transmission	255
SP RPDO 1 Inhibit Time	0
SP RPDO 1 Compatibility Entry	0
SP RPDO 1 Event Timer	0

2. Set RPDO1 'Number of Mapped Objects' = 0 (this opens access to the mapping objects. Users should do this if they are configuring by CANopen® SDO writes. If the Axiomatic EA software is being used, then this is not needed)
3. Set RPDO1 'Mapping Entry #1' = 0x20000110 (this means: 'store a 16-bit data in object 0x2000_sub-index_1')
4. Set RPDO1 'Mapping Entry #2' = 0x20000210 (this means: 'store a 16-bit data in object 0x2000_sub-index_2')
5. Set RPDO1 'Mapping Entry #3' = 0x20000310 (this means: 'store a 16-bit data in object 0x2000_sub-index_3')
6. Set RPDO1 'Mapping Entry #2' = 0x20000410 (this means: 'store a 16-bit data in object 0x2000_sub-index_4')
7. Set RPDO1 'Number of Mapped Objects' = 4 (now there are four mapped entries)

SP RPDO 1 Number of Mapped Objects	4
SP RPDO 1 Mapping Entry #1	0x20000110
SP RPDO 1 Mapping Entry #2	0x20000210
SP RPDO 1 Mapping Entry #3	0x20000310
SP RPDO 1 Mapping Entry #4	0x20000410

- Set RPDO1 'COB-ID' = 0x**4**000020A (activate RPDO1)

SP RPDO 1 COB-ID	0x4000020a
SP RPDO 1 Transmission	255
SP RPDO 1 Inhibit Time	0
SP RPDO 1 Compatibility Entry	0
SP RPDO 1 Event Timer	0

So now with that done, RPDO1 should be sent on ID 0x20A and contain four 16-bit data signals.

Version	Date	Authors	Comments
1.00	September 17, 2019	Lawrence Durham / Sue Thomas	Initial Release
1.01	January 27, 2020	Lawrence Durham / Sue Thomas	To update the header from AN409 to AN408.
1.02	July 6, 2023	Kiril Mojsov	Legacy Update & Marketing Review