Programming Manual UMAX030521
Firmware: V1.xx
Document Revision: A
Date: February 10, 2011

USER MANUAL

1 Analog Signal Output CANopen® Controller

P/N: AX030521

In Europe:
Axiomatic Technologies Oy
Höytämöntie 6
33880 Lempäälä - Finland
Tel. +358 3 3595 600
Fax. +358 3 3595 660
www.axiomatic.fi

In North America:
Axiomatic Technologies Corporation
5915 Wallace Street
Mississauga, ON Canada L4Z 1Z8
Tel. 1 905 602 9270
Fax. 1 905 602 9279
www.axiomatic.com

ACRONYMS

CAN	Controller Area Network
CANopen	CAN-based higher layer protocol supported by CAN in Automation (CiA)
DM	Diagnostic message. Defined in J1939/73 standard
EA	Electronic Assistant™. PC application software from Axiomatic, primary designed to view and program Axiomatic control setpoints through CAN bus using J1939 Memory Access Protocol
ECU	Electronic control unit
EMI	Electromagnetic Interference
LSB	Less Significant Byte
PC	Personal Computer
PGN	Parameter Group Number. Defined in J1939/73 standard
RS-232	PC serial port interface
SAE J1939	CAN-based higher level protocol designed and supported by Society of automobile Engineers (SAE)
USB	Universal Serial Bus
UTP	Un-shielded twisted pair

TABLE OF CONTENTS

1. INTRODUCTION

The following manual describes the controller software architecture, CANopen® object dictionary and firmware programming of the 1 Analog Signal Output CANopen® Controller. The manual is intended to provide users with all necessary information for programming of custom solutions on the base of this controller.

The programming is performed through CAN interface and does not require disconnection of the controller from the user system CAN bus after the controller is installed in the system.

The controller supports CANopen® interface. It is assumed, that the user is familiar with the CANopen standards; the terminology from these standards is widely used in this manual.

The programming is performed through CAN interface and does not require disconnection of the controller from the user system CAN bus after the controller is installed in the system.

2. CONTROLLER DESCRIPTION

The controller is designed to monitor application signals transmitted on the CAN bus by various ECUs using one universal analog signal output. The output can be individually programmed to output voltage or current in the user-defined output range. The ECU application signals can be pre-processed before being output in case an advanced logic is required. Embedded voltage and current monitoring circuits are available to monitor the actual output signal.

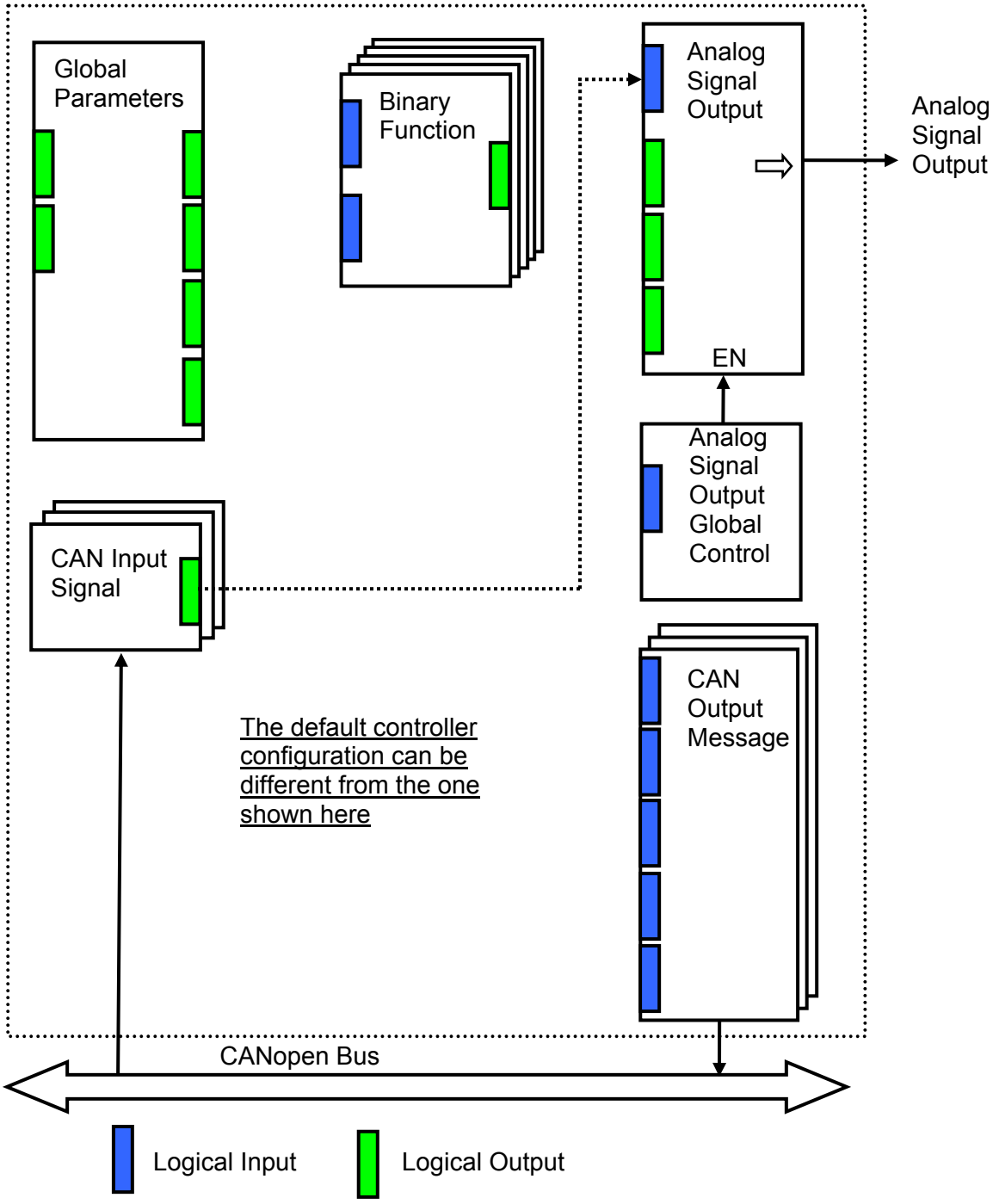
The 1 Analog Signal Output CAN Controller (1AOUT) belongs to a family of Axiomatic user-customizable smart controllers. The programmable internal architecture provides users with an ultimate flexibility, allowing them to build their own custom controller with a required functionality from a set of predefined internal functional blocks using any commercially available CANopen® tools.

All application programming is performed through CAN interface, without disconnecting the controller from the user’s system.

Besides reading data transmitted on the CANopen® bus, the controller can also transmit CANopen® PDO messages carrying data internally generated by the controller. This feature can be used for monitoring and debugging purposes

3. CONTROLLER FUNCTIONAL BLOCKS

From the software prospective, the controller consists of a set of internal functional blocks, which can be individually programmed and arbitrarily connected together to achieve the required system functionality, see Figure 1.



As an example, the logical output of the CAN Input Signal functional block is connected to the logical input of the Analog Signal Output functional block, providing a direct path for the CAN input signal to the controller signal output.

Figure 1.The Controller Internal Structure.

Each functional block is absolutely independent and has its own set of programmable parameters, or object dictionary entries. The parameters can be viewed and changed through CAN using any commercially available CANopen software.

There are two types of the controller functional blocks. One type represents the controller hardware resources, for example the analog signal output block. The other type is purely logical – these functional blocks are included to program the user defined functionality of the controller. The number and functional diversity of these functional blocks are only limited by the system resources of the internal microcontroller. They can be added or modified on the customer's request to accommodate user-specific requirements.

The user can build virtually any type of a custom control by logically connecting inputs and outputs of the functional blocks. This approach gives the user an absolute freedom of customization and an ability to fully utilize the controller hardware resources in a user's application.

Depending on the block functionality, a functional block can have: logical inputs, logical outputs or any combinations of them. The connection between logical inputs and outputs is defined by logical input objects. The following rules apply:

- A logical input can be connected to any logical output using a logical input object.
- Two or more logical inputs can be connected to one logical output.
- Logical outputs do not have their own setpoints controlling their connectivity. They can only be chosen as signal sources by logical inputs.

To provide data flow between logical inputs and outputs, all logical output signals are normalized to [0;1] data range using the following equation:

$$Yn = (Y - Ymin) / (Ymax - Ymin),$$

where: Yn – normalized output value,
Y – original output value,
Ymax – maximum output value,
Ymin – minimum output value.

The original output values are restored, if necessary, at the logical inputs using the following reverse linear transformation:

$$X = Xn \bullet (Xmax - Xmin) + Xmin,$$

where: X – original restored input value,
Xn – normalized input value, Xn=Yn,
Xmax – maximum input value, Xmax=Ymax,
Xmin – minimum input value, Xmin=Ymin.

All functional blocks have (Xmax, Xmin) and (Ymax, Ymin) object pairs controlling the normalization process. They will be called "normalization parameters" further in the object dictionary descriptions.

For discrete logical inputs and outputs the normalization parameters are not required, since the discrete signals can take only two values: {0,1}. When a regular logical output of a functional block is connected to a discrete logical input, it is assumed that the input values below 0.5 represent state 0 and above 0.5 – state 1:

Discrete Logical Input	Logical State
< 0.5	0
≥ 0.5	1

For additional flexibility, in a majority of functional blocks, logical input signals can be inverted using the following inversion function:

Inv(Xn,I), I∈ {Yes,No},

Inv(Xn,I)={1-Xn, if I=Yes; Xn, if I=No}

In addition to signal values in the range of [0;1], the logical inputs and outputs also carry information on the state of the data source. This information can show that the source is not available or there is an error in data, or the data source is in a special state.

When the data source does not carry a valid data, the output signal value is always set to 0 and the inversion operation on the signal in suppressed. In this case, instead of the signal value, the logical signal carries a signal state code, associated with its signal state, see the table below:

Signal State	Signal Value, Xn	Signal State Code	Inverted Signal Value	
			Xn'=Inv(Xn,Yes)	Xn'=Inv(Xn,No)
Valid Data	[0;1]	0	1-Xn	Xn
Special	0	0…4294967295 (0…0xFFFFFFFF) – Special State Code	0	0
Error	0	0…4294967295 (0…0xFFFFFFFF) – Error Code	0	0
Not Available	0	0	0	0

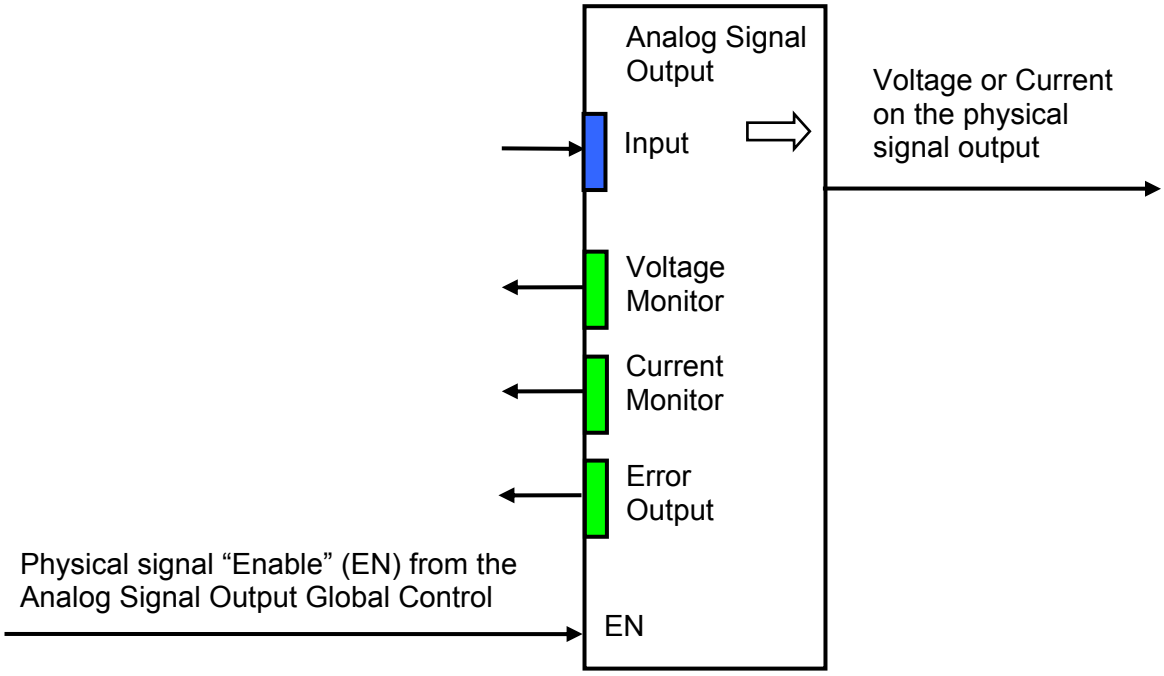
The states of the data source other than the "Valid Data" are primary used by CAN functional blocks to report that a CAN input signal is absent on the bus, is out of range, etc. Other functional blocks usually use only the "Error" state to show an error condition.

3.1 Analog Signal Output

There is one Analog Signal Output functional block representing analog signal output of the controller. The functional block can be programmed to output voltage or current in the user-defined range. The output signal can be enabled or disabled through the Analog Signal Output Global Control functional block.

The controller has internal hardware circuits measuring voltage and current on the physical signal output. These parameters are available as logical signals on the voltage and current monitor logical outputs. In addition, the functional block has a separate error logical output that flags an error condition when the output signal is beyond the specified error range.

The Analog Signal Output functional block has one logical input receiving a normalized output signal and three logical outputs monitoring the state of the physical signal output. The block is also internally connected to the Analog Signal Output Global Control functional block, which enables or disables the physical signal output.



The functional block setpoints are presented in the following table:

Name	Default Value	Range	Units	Description
Input Source	CAN Input Signal #1	Any logical output of any functional block or "Not Connected"	–	Defines an input signal source of the analog signal output.
Input Inversion	No	{Yes, No}	–	Defines whether the input signal from the Input Source is inverted.
Output Mode	Output Voltage	{Output Voltage, Output Current}	–	Specifies an output mode of the analog signal output.
Vmax – Maximum Output Voltage	5000	[-10000…10000], but Vmax>Vmin	mV	Normalization parameters for Output Voltage mode.
Vmin – Minimum Output Voltage	0	[-10000…10000], but Vmin<Vmax	mV	
Imax – Maximum Output Current	20	[-20…20], but Imax>Imin	mA	Normalization parameters for Output Current mode.
Imin – Minimum Output Current	4	[-20…20], but Imin<Imax	mA	
Vmax – Voltage Monitor	12.0	–	V	Normalization Parameters for Voltage Monitor output. Read only parameters.
Vmin – Voltage Monitor	-12.0	–	V	
Imax – Current Monitor[1]	25.0	–	mA	Normalization Parameters for Current Monitor output. Read only parameters.
Imin – Current Monitor[1]	-25.0	–	mA	
Output Error Threshold	5.0	[0…100]	%	Maximum relative error of the signal on the analog signal output. The Error Output is

Name	Default Value	Range	Units	Description
				switched from “0” to “1” when the output error exceeds this value.
Output Error Delay	0.02	[0…300]	s	Delay before the Error Output changes its condition.
Disable Output On Error, set using 1029h & 6340h	No	{Yes, No}	–	If "Yes", Signal Output will be disabled on an error condition until the next power-up.
Disable Output On Error Delay, set using 30D0h	1000	[0…10000]	ms	Delay before the Signal Output is disabled.

[1] Current Monitor logical output can be in an error state with an error code equal to 0 if voltage on the signal output is above or below the current monitor voltage range (±8 [V]).

The Output Error Threshold (Object 30D1h: AO Output Error Threshold) defines the output signal error relatively to the signal range specified by normalization parameters of the output signal. When the signal output is in the voltage output mode, these parameters are: Vmax – Maximum Output Voltage and Vmin – Minimum Output Voltage. In the current mode, they are: Imax – Maximum Output Current and Imin – Minimum Output Current.

The Error Output logical signal is set to “1” with the Output Error Delay time (Object 30D2h: AO Output Error Delay) when the output signal error exceeds the Output Error Threshold value. For example, for the voltage output mode:

ErrorOutput = {1, if $|Vout'-Vmon|>\Delta$ for OutputErrorDelay time;
0, if $|Vout'-Vmon|\leq\Delta$ for OutputErrorDelay time },

Δ = (Vmax-Vmin)*OutputErrorThreshold/100.
Vout’ – Voltage expected on the output and defined by the logical input.
Vmon – Voltage measured on the output and available on the Voltage Monitor logical output.

The Output Error Delay time should allow the signal output and the monitor circuits to settle down after a new signal value is commanded to the output. This settling time is approximately 5…20 ms depending on the required monitoring accuracy. The application signal update rate and the slew rate should be also taken into consideration when defining this setpoint value.

When the Disable Output On Error functionality is set to “Yes”, the signal output will be disabled (set to zero) and the ErrorOutput will be set to “1”until the next power-up, if the error condition on the output stays continuously for the time defined by the Output Error Delay (Object 30D0h: AO Output Error Flag Delay). The Disable Output On Error functionality is defined using Object 6340h: AO Fault Mode, Object 7341h: AO Fault FV and Object 1029h: Error Behavior. For the voltage output mode:

Vout’ = 0, ErrorOutput=1, if $|Vout'-Vmon|>\Delta$ for DisableOutputOnErrorDelay time.

For the current output, the error condition triggering the Error Output logical signal is either the output current error or the current monitor error. The current monitor error appears when the

voltage on the signal output is too high for the current monitor to function properly. It is usually due to an open circuit on the output.

When the current monitor is in the error condition, the Current Monitor logical output is in the error state with an error code equal to 0. It is different from the Voltage Monitor output, which always shows the valid data.

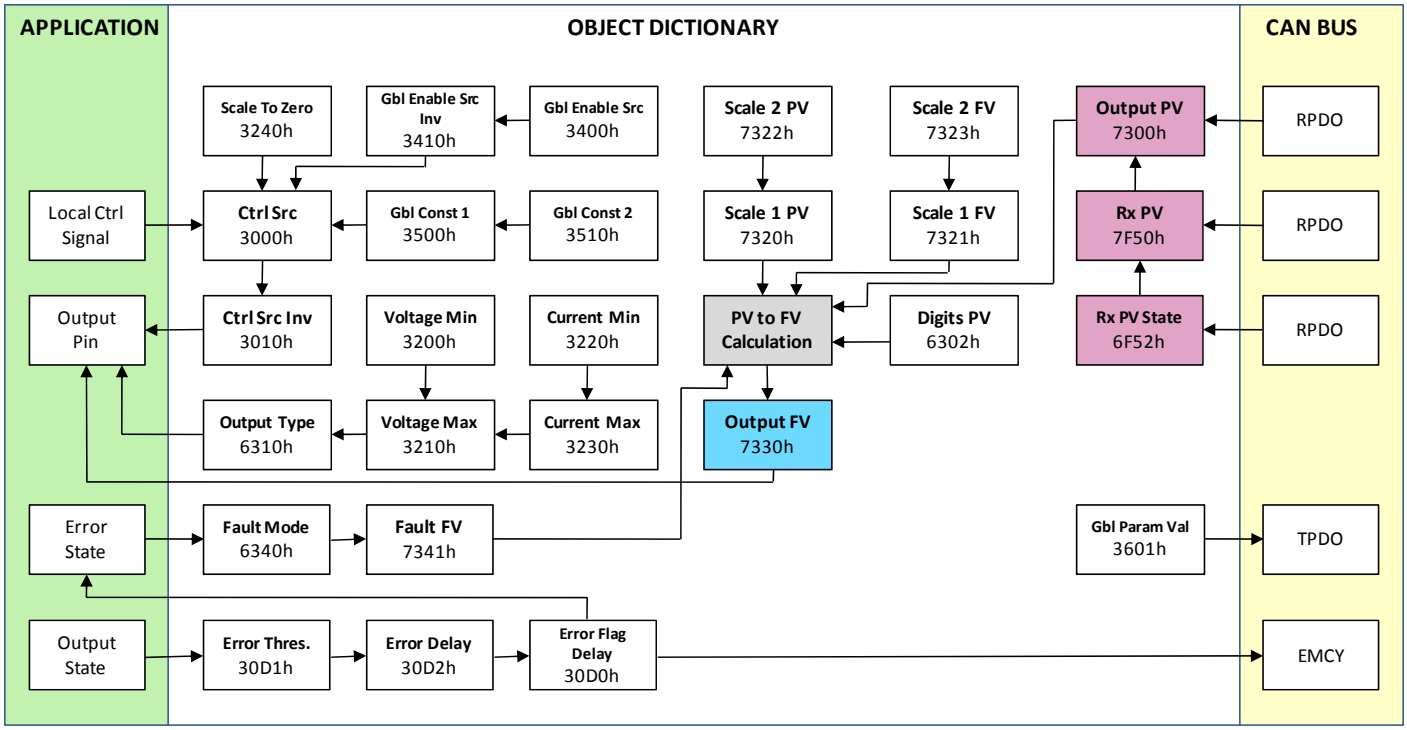


Figure 2: Analog Output Block

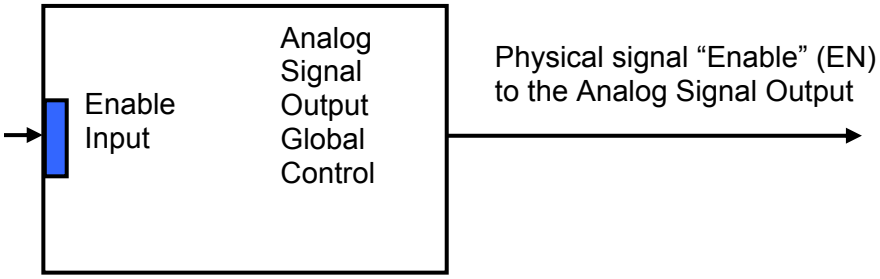
Table 1: Available control sources

Name	Value	Description
No Source	0	Output disabled
Binary Function 1	1	Binary function 1 output
Binary Function 2	2	Binary function 2 output
Binary Function 3	3	Binary function 3 output
Binary Function 4	4	Binary function 4 output
Binary Function 5	5	Binary function 5 output
CAN Input 1	6	CAN FV/PV, subindex 1
CAN Input 2	7	CAN FV/PV, subindex 2
CAN Input 3	8	CAN FV/PV, subindex 3
Global VS	9	Measured operating voltage
Global Temp	10	Measured chip temperature
Global Const 1	11	Constant value 1, 3500h
Global Const 2	12	Constant value 2, 3510h
Constant ‘0’	13	0
Constant ‘1’	14	1
Output 1 Voltage	15	Measured voltage on output 1
Output 1 Current	16	Measured current on output 1

Name	Value	Description
Output 1 Error	17	Measured error of output 1

3.2 Analog Signal Output Global Control

The Analog Signal Output Global Control functional block is used to globally enable or disable the analog signal output of the controller. It has one logical input to control the analog signal output.



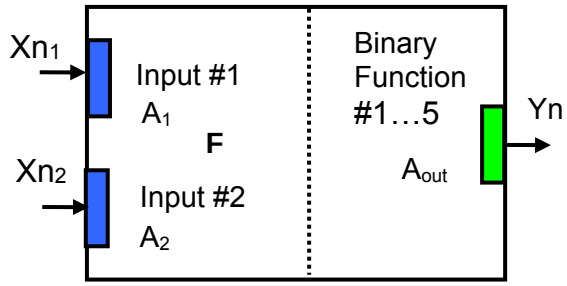
The functional block setpoints are defined as follows:

Name	Default Value	Range	Units	Description
Enable Input Source	Constant Output = 1.0	Any logical output of any functional block or “Not Connected”	–	Defines an input signal source to enable the signal output.
Enable Input Inversion (EA alias: Enable Input Source Inversion)	No	{Yes, No}	–	Defines whether the input signal from the Enable Input Source is inverted.

The Enable Input is connected to the Constant Output = 1.0 to enable the analog signal output by default.

3.3 Binary Function

There are five Binary Function functional blocks added to the controller to support advanced CAN signal monitoring algorithms. Each Binary Function functional block takes two logical input signals, scales them, and performs an arithmetic or logical operation on the scaled inputs. Then it outputs the result, which can be scaled as well.



The normalized output signal Yn of the Binary Function functional block can be presented by the following formula:

$Yn=Clip(Y)$,

$Y = A_{out} \bullet F[A_1 \bullet Xn_1, A_2 \bullet Xn_2]$,

where:

$Clip(Y)=\{Y, \text{ if } 0 \le Y \le 1; 0, \text{ if } Y<0; 1, \text{ if } Y>1\}$ – clipping function;

Xn_1, Xn_2 – normalized signal values of the input sources (can be inverted);

A_1, A_2 – input scale coefficients;

A_{out} – output scale coefficient;

$F[x, y]$ – binary function of the scaled input signals: $x=A_1 \bullet Xn_1$, $y=A_2 \bullet Xn_2$.

In case one of the input sources is not connected, the output signal of the functional block is not available and its signal value is equal to Yn=0.

The Binary Function functional block has the following set of setpoints:

Name	Default Value	Range	Units	Description
Input #1 Source	Not Connected	Any logical output of any functional block or "Not Connected"	–	Source of the input #1 signal
Input #1 Inversion	No	{Yes, No}	–	Specifies, whether to invert the input #1 signal
Input #1 Scale	1.0	Any value	–	Input #1 signal scale coefficient
Input #2 Source	Not Connected	Any logical output of any functional block or "Not Connected"	–	Source of the input #2 signal
Input #2 Inversion	No	{Yes, No}	–	Specifies, whether to invert the input #2 signal
Input #2 Scale	1.0	Any value	–	Input #2 signal scale coefficient
Function	+	{+, ∗,÷, Max, Min, OR, AND, XOR, <, ≤, =, >, ≥}	–	Binary function of the input #1 scaled signal and the input #2 scaled signal
Output Scale	1.0	Any value	–	Output signal scale coefficient

The binary functions F[x,y] have the following implementation specifics.

In the division function, to avoid ambiguity in dividing by 0, the dividend and the divisor are not allowed to be less than δ:

$F^{(\div)}[x,y] = \max(x,\delta)/ \max(y,\delta)$,

where: δ =1.0E-6 is a specially introduced computational constant.

For logical functions {OR, AND, XOR} values $X_i \ge 0.5$ (i=1,2) are treated as 1 (true) and $X_i<0.5$ – as 0 (false).

To minimize influence of computational errors during normalization, comparison functions {≤, =, ≥ } are defined the following way:

$F^{(\leq)}$ [x,y] = {1, if $x \leq y+\delta$; 0, if $x > y+\delta$ },

$F^{(=)}$ [x,y] = {1, if $|x-y| \leq \delta$; 0, if $|x-y| > \delta$},

$F^{(\geq)}$ [x,y] = {1, if $x \geq y-\delta$; 0, if $x < y-\delta$ }.

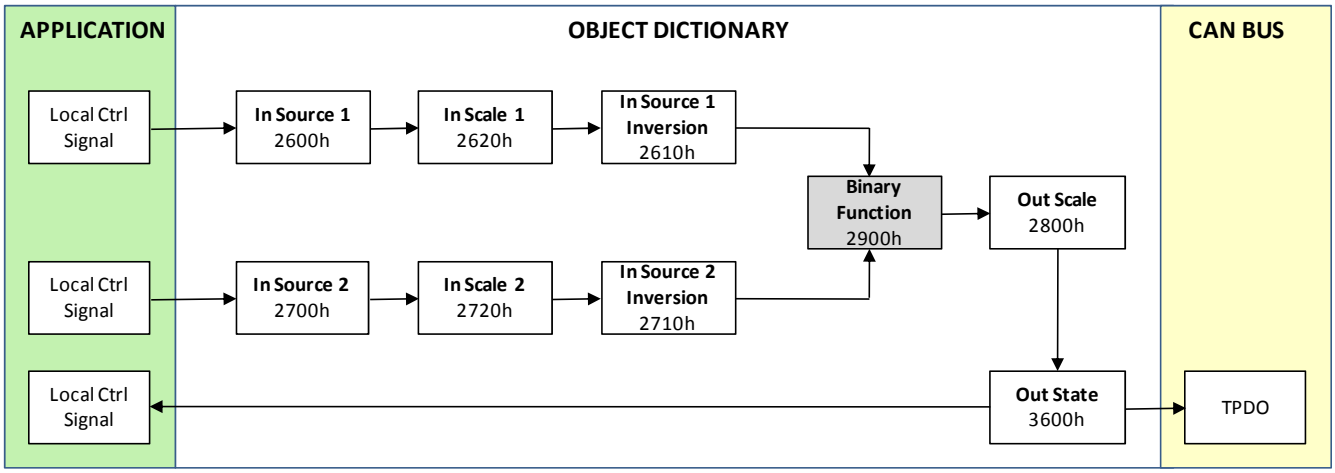


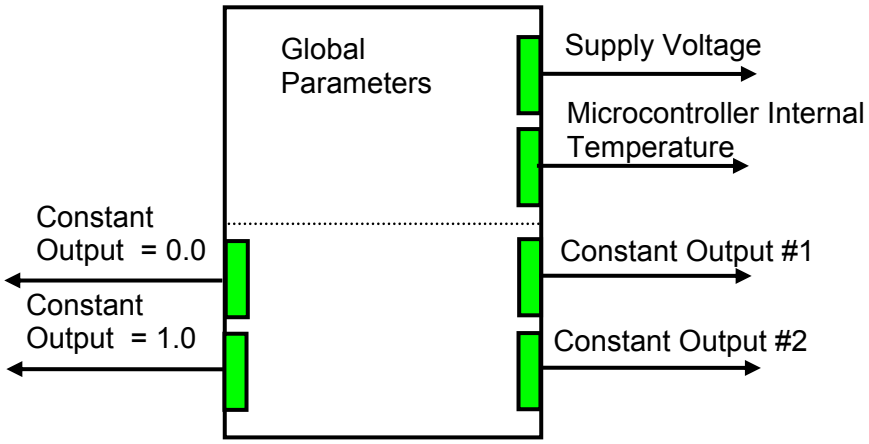
Figure 3: Binary Function block

Table 2: Available binary functions (2900h)

Name	Value	Description
Add	0	<In 1> **+** <In 2>
Multiplication	1	<In 1> ***** <In 2>
Division	2	<In 1> **/** <In 2>
Maximum	3	Max(<In 1>,<In 2>)
Minimum	4	Min(<In 1>,<In 2>)
OR (binary)	5	<In 1> **||** <In 2>
AND (binary)	6	<In 1> **&&** <In 2>
XOR (binary)	7	<In 1> **^** <In 2>
Less Than	8	<In 1> **<** <In 2>
Less or Equal	9	<In 1> **<=** <In 2>
Equal	10	<In 1> **==** <In 2>
Greater Then	11	<In 1> **>** <In 2>
Greater or Equal	12	<In 1> **>=** <In 2>

3.4 Global Parameters

The Global Parameters functional block gives the user access to the controller supply voltage and the microcontroller internal temperature as well as to a set of four constant logical outputs. These outputs can be used by other functional blocks as constant input sources. For example, they can be used to set up threshold values in Binary Function functional blocks.



Two out of four constant logical outputs are user programmable. Other two represent logical one and logical zero outputs.

Please note, that the supply voltage, provided by the Global Parameters functional block, is not the voltage on the controller power supply pins. It is an internal voltage measured after the reverse polarity protection and filtering circuit. It is always less than the actual power supply voltage by approximately 0.4…0.8 V.

The parameters for the Global Parameters functional block are presented in the following table:

Name	Default Value	Range	Units	Description
Constant Output #1	0.0	[0…1]	–	Logical output with a constant value.
Constant Output #2	0.0	[0…1]	–	Logical output with a constant value.
Vsmax – Max Supply Voltage	70	–	V	Normalization parameters for the inclinometer supply voltage. Read only parameters.
Vsmin – Min Supply Voltage	0	–	V	
Tmax – Max Microcontroller Temperature	150	–	°C	Normalization parameters for the microcontroller embedded temperature sensor. Read only parameters.
Tmin – Min Microcontroller Temperature	-50	–	°C	

3.5 CAN Input and Output Signals

There are three CAN Input Signal functional blocks supported by the controller. Each functional block can be programmed to receive PDOs and extract CAN data presented in virtually any user-defined signal data format. The functional block then outputs the signal data to its logical output for processing by other functional blocks of the controller.

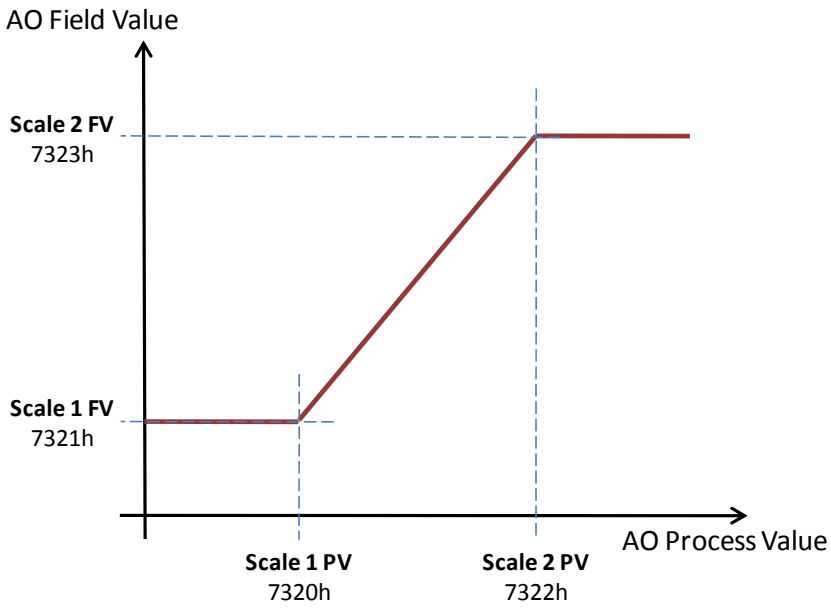


Figure 4: Analog Output PV/FV Characteristics

CAN Input Signal Blocks can be supplied with data using the 3 subindices of various Analog Output related objects in the Application –part of the object dictionary. By default the subindex 1 (namely CAN Input Signal Blocks 1) map directly to the only output on board (see Object 3000h: AO Output Control Source), however this can be changed. The subindexes 2 and 3 are included to provide closer match to the J1939 version of the product (AX030520).

See Object 7330h: AO Output Field Value, Object 7320h: AO Output Scaling 1 PV, Object 7321h: AO Output Scaling 1 FV, Object 7322h: AO Output Scaling 2 PV, Object 7323h: AO Output Scaling 2 FV, Object 7330h: AO Output Field Value, Object 7F50h: Received PV 16 and Object 6F52h: Received PV Status for more detailed description of the CAN Input Signal functional block parameters.

Figure 4 shows the relation between Process Value and Field Value of the received CAN data. The controller receives Process Values as PDOs (or SDOs) and uses Field value to set the value of the Analog Output Functional Block. The scaling can be changed by modifying the values of objects 732xh.

4. CONTROLLER ARCHITECTURE

The CANopen object dictionary of the 1 Analog Signal Output CAN Controller (1AOUT) is based on CiA device profile DS-404 (Device profile for measurement devices and closed-loop controllers). The object dictionary also includes some manufacturer-specific objects for extended functionality.

4.1 COMMUNICATION OBJECTS

The communication objects supported by the 1AOUT are listed in the following table. A more detailed description of some of the objects is given in the following subchapters. Only those objects that have device-profile specific information are described. For more information on the other objects, refer to the generic CANopen protocol specification DS-404.

Index (hex)	***Object***	***Object Type***	***Data Type***	***Access***	***PDO Mapping***
1000	Device Type	VAR	UNSIGNED32	RO	No
1001	Error Register	VAR	UNSIGNED8	RO	No
1002	Manufacturer Status Register	VAR	UNSIGNED32	RO	No
1003	Pre-Defined Error Field	ARRAY	UNSIGNED32	RO	No
100B	Node-ID	VAR	UNSIGNED8	RO	No
100C	Guard Time	VAR	UNSIGNED16	RW	No
100D	Life Time Factor	VAR	UNSIGNED8	RW	No
1010	Store Parameters	ARRAY	UNSIGNED32	RW	No
1011	Restore Default Parameters	ARRAY	UNSIGNED32	RW	No
1016	Consumer Heartbeat Time	ARRAY	UNSIGNED32	RW	No
1017	Producer Heartbeat Time	VAR	UNSIGNED16	RW	No
1018	Identity Object	RECORD		RO	No
1020	Verify Configuration	ARRAY	UNSIGNED32	RW	No
1029	Error Behavior	ARRAY	UNSIGNED8	RW	No
102A	Slew Rate	VAR	UNSIGNED8	RW	No
1400	RPDO1 Communication Parameter	RECORD		RW	No
1401	RPDO2 Communication Parameter	RECORD		RW	No
1402	RPDO3 Communication Parameter	RECORD		RW	No
1403	RPDO4 Communication Parameter	RECORD		RW	No
1600	RPDO1 Mapping Parameter	RECORD		RW	No
1601	RPDO2 Mapping Parameter	RECORD		RW	No
1602	RPDO3 Mapping Parameter	RECORD		RW	No
1603	RPDO4 Mapping Parameter	RECORD		RW	No
1800	TPDO1 Communication Parameter	RECORD		RW	No
1801	TPDO2 Communication Parameter	RECORD		RW	No
1802	TPDO3 Communication Parameter	RECORD		RW	No
1803	TPDO4 Communication Parameter	RECORD		RW	No
1A00	TPDO1 Mapping Parameter	RECORD		RW	No
1A01	TPDO2 Mapping Parameter	RECORD		RW	No
1A02	TPDO3 Mapping Parameter	RECORD		RW	No
1A03	TPDO4 Mapping Parameter	RECORD		RW	No

4.1.1 PDO Parameters

All RPDOs and TPDOs in the 1AOUT use the same default communication parameters, respectively. The PDO IDs are set according to the pre-defined connection set described in [DS-301]. All receive PDOs are set to transmission type 255, and all transmit PDOs to transmission type 254, with the event timer (subindex 5) set to 100 (100ms).

All PDOs are dynamically mappable, and the user can therefore change the mapping of the PDOs. The granularity is 8-bits, so the objects can be mapped with byte offsets. The mapping parameter records include 4 subindexes for receive and for transmit for the PDO mapping. The default PDO mappings are listed in the following tables.

RPDO1: default ID 0x200 + node ID

Subindex	***Value***	***Object***
0	4	Number of mapped application objects in PDO
1	0x73000110	Process Value, SW Output 1
2	0x73000210	Process Value, SW Output 2
3	0x73000310	Process Value, SW Output 3
4	0	

RPDO2: default ID 0x300 + node ID

Subindex	***Value***	***Object***
0	0	Number of mapped application objects in PDO
1	0x7F500110	Received PV, SW Output 1
2	0x7F500210	Received PV, SW Output 2
3	0x7F500310	Received PV, SW Output 3
4	0	

RPDO3: default ID 0x400 + node ID

Subindex	***Value***	***Object***
0	0	Number of mapped application objects in PDO
1	0x6F520108	Received PV Status, SW Output 1
2	0x6F520208	Received PV Status, SW Output 2
3	0x6F520308	Received PV Status, SW Output 3
4	0	

RPDO4: default ID 0x500 + node ID

Subindex	***Value***	***Object***
0	0	Number of mapped application objects in PDO
1	0	
2	0	
3	0	
4	0	

TPDO1: default ID 0x180 + node ID

Subindex	***Value***	***Object***
0	2	Number of mapped application objects in PDO
1	0x36010120	Board operating voltage
2	0x36010220	Board temperature
3	0	
4	0	

TPDO2: default ID 0x280 + node ID

Subindex	***Value***	***Object***
0	0	Number of mapped application objects in PDO
1	0x36000120	Binary function 1 output
2	0x36000220	Binary function 2 output
3	0	
4	0	

TPDO3: default ID 0x380 + node ID

Subindex	***Value***	***Object***
0	0	Number of mapped application objects in PDO
1	0x36000320	Binary function 3 output
2	0x36000420	Binary function 4 output
3	0	
4	0	

TPDO4: default ID 0x480 + node ID

Subindex	***Value***	***Object***
0	0	Number of mapped application objects in PDO
1	0x36000520	Binary function 5 output
2	0	
3	0	
4	0	

4.1.2 Object 1000h: Device Type

This object contains information about the device type as per device profile DS-404 (measurement devices and closed-loop controllers). The value stored in this object is 0x00080194, indicating that the 1AOUT includes the following function blocks defined in the device profile.

- Analog Output (AO)

Object Description

Index	1000h
Name	Device Type
Object Type	VAR
Data Type	UNSIGNED32

Entry Description

Access	RO
PDO Mapping	No
Value Range	0x00080194
Default Value	0x00080194

4.1.3 Object 1001h: Error Register

This object is an error register for the device. Any time there is an error detected by the 1 Analog Signal Output CAN Controller, the Generic Error Bit (bit 0) is set. Only if there are no errors in the module will this bit be cleared. The Single Channel CAN Controller uses no other bits in this register.

Object Description

Index	1001h
Name	Error Register
Object Type	VAR
Data Type	UNSIGNED16

Entry Description

Access	RO
PDO Mapping	No
Value Range	00h or 01h
Default Value	0

4.1.4 Object 1002h: Manufacturer Status Register

This object is used for manufacturer debug purposes.

4.1.5 Object 1003h: Pre-Defined Error Field

The object 1003h provides an error history by listing the errors in the order that they have occurred. An error is added to the top of the list when it occurs, and is immediately removed when the error condition has been cleared. The latest error is always at subindex 1, with subindex 0 containing the number of errors currently in the list. When the device is in an error-free state, the value of subindex 0 is zero.

The error list may be cleared by writing a zero to subindex 0, which will clear all errors from the list, regardless of whether or not they are still present. Clearing the list does NOT mean that the module will return to the error-free behaviour state if at least one error is still active.
The 1 Analog Signal Output CAN Controller has a limitation of a maximum of 8 errors in the list. If the device registers more errors, the list will be truncated, and the oldest entries will be lost.

The error codes stored in the list are 32-bit unsigned numbers, consisting of two 16-bit fields. The lower 16-bit field is the EMCY error code, and the higher 16-bit field is a manufacturer-specific code. The manufacturer-specific code is divided into two 8-bit fields, with the higher byte indicating the error description, and the lower byte indicating the channel number where the error occurred.

Note, that because of the hardware implementation specific details of the Analog Outputs, the only errors detected by the 1AOUT are CANopen communications specific.

Table 3: 1AOUT EMCY codes

Error Field Code	Error Description	Meaning	ID	Meaning	EMCY Code	Meaning
00000000h	EMCY Error Reset (fault no longer active)					
00008100h	00h	RPDO Timeout	00h	Unspecified	8100h	Communication - generic
10008130h	10h	Lifeguard Event	00h	Unspecified	8130h	Lifeguard/Heartbeat Error
80nn8130h	80h	Heartbeat Timeout	nn	Node-ID	8130h	Lifeguard/Heartbeat Error
1401F002h	14h	Output Overload	01h	Output 1	F002h	Open Circuit
1401F003h	14h	Output Overload	01h	Output 1	F003h	Short Circuit

Object Description

Index	1003h
Name	Pre-Defined Error Field
Object Type	VAR
Data Type	UNSIGNED32

Entry Description

Subindex	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	0 to 8
Default Value	0

Subindex	1h to 8h
Description	Standard error field
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	0

4.1.6 Object 100Bh: Node-ID

This object contains the CANopen node-ID of the 1 Analog Signal Output CAN Controller.

Object Description

Index	100Bh

Name	Node-ID
Object Type	VAR
Data Type	UNSIGNED8

Subindex	0h
Description	Node-ID
Access	RO
PDO Mapping	No
Value Range	0x7F
Default Value	0x7F

4.1.7 Object 100Ch: Guard Time

The objects at index 100Ch and 10Dh shall indicate the configured guard time respective to the life time factor. The life time factor multiplied with the guard time gives the life time for the life guarding protocol described in DS-301. The Guard Time value shall be given in multiples of ms, and a value of 0000h shall disable the life guarding.

It should be noted that this object, and that of 100Dh are only supported for backwards compatibility. The standard recommends that newer networks do not use life guarding protocol, but rather heartbeat monitoring instead. Both life guarding and heartbeats can NOT be active simultaneously.

Object Description

Index	100Ch
Name	Guard Time
Object Type	VAR
Data Type	UNSIGNED16

Subindex	0h
Access	RW
PDO Mapping	No
Value Range	0 to 65536
Default Value	0

4.1.8 Object 100Dh: Lifetime Factor

The life time factor multiplied with the guard time gives the life time for the life guarding protocol. A value of 00h shall disable life guarding.

Object Description

Index	100Dh
Name	Lifetime factor
Object Type	VAR
Data Type	UNSIGNED8

Subindex	0h
Access	RW
PDO Mapping	No
Value Range	0 to 255

Default Value	0

4.1.9 Object 1010h: Store Parameters

This object supports the saving of parameters in non-volatile memory. In order to avoid storage of parameters by mistake, storage is only executed when a specific signature is written to the appropriate subindex. The signature is “save”.

The signature is a 32-bit unsigned number, composed of the ASCII codes of the signature characters, according to the following table:

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

On reception of the correct signature to an appropriate subindex, the 1 Analog Signal Output CAN Controller will store the parameters in non-volatile memory, and then confirm the SDO transmission.

By read access, the object provides information about the 1 Analog Signal Output CAN Controller’s saving capabilities.

Object Description

Index	1010h
Name	Store Parameters
Object Type	ARRAY
Data Type	UNSIGNED32

Entry Description

Subindex	0h
Description	Largest subindex supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Subindex	1h
Description	Save all parameters
Access	RW
PDO Mapping	No
Value Range	0x65766173 (write access)/ 1h (read access)
Default Value	1h (saves parameters on command)

Subindex	2h
Description	Save communication parameters
Access	RW
PDO Mapping	No
Value Range	0x65766173 (write access) 1h (read access)
Default Value	1h (saves parameters on command)

Subindex	3h
Description	Save application parameters
Access	RW
PDO Mapping	No
Value Range	0x65766173 (write access) 1h (read access)
Default Value	1h (saves parameters on command)

Subindex	4h
Description	Save manufacturer parameters
Access	RW
PDO Mapping	No
Value Range	0x65766173 (write access) 1h (read access)
Default Value	1h (saves parameters on command)

4.1.10 Object 1011h: Restore Default Parameters

This object supports the restoring of the default values for the object dictionary in non-volatile memory. In order to avoid restoring of parameters by mistake, the device restores the defaults only when a specific signature is written to the appropriate subindex. The signature is “load”.

The signature is a 32-bit unsigned number, composed of the ASCII codes of the signature characters, according to the following table:

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

On reception of the correct signature to an appropriate subindex, the 1 Analog Signal Output CAN Controller will restore the defaults in non-volatile memory, and then confirm the SDO transmission. The default values are set valid after the device is reset or power-cycled.

By read access, the object provides information about the 1 Analog Signal Output CAN Controller’s default parameter restoring capabilities.

Object Description

Index	1011h
Name	Restore Default Parameters
Object Type	ARRAY
Data Type	UNSIGNED32

Entry Description

Subindex	0h
Description	Largest subindex supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Subindex	1h

Description	Restore all default parameters
Access	RW
PDO Mapping	No
Value Range	0x64616F6C (write access) 1h (read access)
Default Value	1h (restores defaults on command)

Subindex	2h
Description	Restore default communication parameters
Access	RW
PDO Mapping	No
Value Range	0x64616F6C (write access) 1h (read access)
Default Value	1h (restores defaults on command)

Subindex	3h
Description	Restore default application parameters
Access	RW
PDO Mapping	No
Value Range	0x64616F6C (write access) 1h (read access)
Default Value	1h (restores defaults on command)

Subindex	4h
Description	Restore default manufacturer parameters
Access	RW
PDO Mapping	No
Value Range	0x64616F6C (write access) 1h (read access)
Default Value	1h (restores defaults on command)

4.1.11 Object 1016h: Consumer Heartbeat Time

The 1 Analog Signal Output CAN Controller can be a consumer of heartbeat objects for up to four modules. This object defines the expected heartbeat cycle time for those modules, and if set to zero, it is not used. When non-zero, the time is a multiple of 1ms, and monitoring will start after the reception of the first heartbeat from the module. If the 1 Analog Signal Output CAN Controller fails to receive a heartbeat from a node in the expected timeframe, it will indicate a communication error, and respond as per object 1029h.

Bits	31-24	23-16	15-0
Value	Reserved 00h	Node-ID	Heartbeat time
Encoded as		UNSIGNED8	UNSIGNED16

Object Description

Index	1016h
Name	Consumer heartbeat time
Object Type	ARRAY
Data Type	UNSIGNED32

Entry Description

Subindex	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Subindex	1h to 4h
Description	Consumer heartbeat time
Access	RW
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	0

4.1.12 Object 1017h: Producer Heartbeat Time

The controller could be configured to produce a cyclical heartbeat by writing a non-zero value to this object. The value will be given in multiples of 1ms, and a value of 0 shall disable the heartbeat.

Object Description

Index	1017h
Name	Producer heartbeat time
Object Type	VAR
Data Type	UNSIGNED16

Subindex	0h
Access	RW
PDO Mapping	No
Value Range	10 to 65536
Default Value	0

4.1.13 Object 1018h: Identity Object

The identity object indicates the data of the 1 Analog Signal Output CAN Controller, including vendor id, device id, software and hardware version numbers, and the serial number.

In the Revision Number entry at subindex 3, the format of the data is as shown below

MSB			LSB
Major revision number (object dictionary)		Hardware Revision	Software Version

Object Description

Index	1018h
Name	Identity
Object Type	RECORD
Data Type	Identity Record

Entry Description

Subindex	0h
Description	Number of entries
Access	RO

PDO Mapping	No
Value Range	4
Default Value	4

Subindex	1h
Description	Vendor ID
Access	RO
PDO Mapping	No
Value Range	0x00000055
Default Value	0x00000055 (Axiomatic)

Subindex	2h
Description	Product Code
Access	RO
PDO Mapping	No
Value Range	0xAA030521
Default Value	0xAA030521

Subindex	3h
Description	Revision Number
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	00010300

Subindex	4h
Description	Serial Number
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	No

4.1.14 Object 1020h: Verify Configuration

This object can be read to see what date the software (version identified in object 1018h) was compiled.

The date and time are represented as a hexadecimal values showing day/month/year and hour/minute/second as per the format below. The time is shown in 24h format.

MSB LSB

Day (in 1-Byte Hex)	Month (in 1-Byte Hex)	Year (in 2-Byte Hex)	
0x00	Hour (in 1-Byte Hex)	Minute (in 1-Byte Hex)	Second (in 1-Byte Hex)

For example, a value of 0x02092011 would indicate that the software was compiled on February 9th, 2011. A value of 0x00001548 would indicate compilation took place at 15:48.

Object Description

Index	1020h
Name	Verify Configuration

Object Type	ARRAY
Data Type	UNSIGNED32

Entry Description

Subindex	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Subindex	1h
Description	Configuration date
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	No

Subindex	2h
Description	Configuration time
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	0

4.1.15 Object 1029h: Error Behavior

This object controls the state that the 1 Analog Signal Output CAN Controller will be set into in case of an error of the type associated with the subindex.

Object Description

Index	1029h
Name	Error Behavior
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	7
Default Value	7

Subindex	1h
Description	Communication Error
Access	RW
PDO Mapping	No
Value Range	0 = Pre-Operational 1 = No State Change 2 = Stopped
Default Value	0 (Pre-Operational)

Subindex	2h
Description	Digital Input Error
Access	RW
PDO Mapping	No
Value Range	0 = Pre-Operational 1 = No State Change 2 = Stopped
Default Value	1 (no state change)

Subindex	3h
Description	Analog Input Error
Access	RW
PDO Mapping	No
Value Range	0 = Pre-Operational 1 = No State Change 2 = Stopped
Default Value	1 (no state change)

Subindex	4h
Description	Digital Output Error
Access	RW
PDO Mapping	No
Value Range	0 = Pre-Operational 1 = No State Change 2 = Stopped
Default Value	1 (no state change)

Subindex	5h
Description	Analog Output Error
Access	RW
PDO Mapping	No
Value Range	0 = Pre-Operational 1 = No State Change 2 = Stopped
Default Value	1 (no state change)

Subindex	6h
Description	Controller Error (not implemented in firmware v1.00)
Access	RW
PDO Mapping	No
Value Range	0 = Pre-Operational 1 = No State Change 2 = Stopped
Default Value	1 (no state change)

Subindex	7h
Description	Alarm Error (not implemented in firmware v1.00)
Access	RW
PDO Mapping	No
Value Range	0 = Pre-Operational 1 = No State Change

	2 = Stopped
Default Value	1 (no state change)

4.1.16 Object 102Ah: Slew Rate

This object controls the slew rate of the CAN interface. Parameter save and reboot is needed in order to use the new value.

Object Description

Index	102Ah
Name	Slew Rate
Object Type	VAR
Data Type	UNSIGNED8

Entry Description

Access	RW
PDO Mapping	No
Value Range	0 (slow), 1 (fast)
Default Value	0

4.2 ANALOG OUTPUT FUNCTION BLOCK

The application and manufacturer objects supported by the 1 Analog Signal Output CAN Controller for the analog output block are listed in the following table.

Index (hex)	*Object*	*Object Type*	*Data Type*	*Access*	*PDO Mapping*
2600	Binary Funcs' Input 1 Source	ARRAY	UNSIGNED8	RW	No
2610	Binary Funcs' Input 1 Source Inv	ARRAY	UNSIGNED8	RW	No
2620	Binary Funcs' Input 1 Scale	ARRAY	FLOAT32	RW	No
2700	Binary Funcs' Input 2 Source	ARRAY	UNSIGNED8	RW	No
2710	Binary Funcs' Input 2 Source Inv	ARRAY	UNSIGNED8	RW	No
2720	Binary Funcs' Input 2 Scale	ARRAY	FLOAT32	RW	No
2800	Binary Funcs' Output Scale	ARRAY	FLOAT32	RW	No
2900	Binary Funcs' Name	ARRAY	UNSIGNED8	RW	No
3000	AO Output Control Source	ARRAY	UNSIGNED8	RW	No
3010	AO Output Control Source Inv	ARRAY	UNSIGNED8	RW	No
30D0	AO Output Error Flag Delay	ARRAY	UNSIGNED16	RW	No
30D1	AO Output Error Threshold	ARRAY	FLOAT32	RW	No
30D2	AO Output Error Delay	ARRAY	FLOAT32	RW	No
3200	AO Output Voltage Minimum	ARRAY	INTEGER16	RW	No
3210	AO Output Voltage Maximum	ARRAY	INTEGER16	RW	No
3220	AO Output Current Minimum	ARRAY	INTEGER16	RW	No
3230	AO Output Current Maximum	ARRAY	INTEGER16	RW	No
3240	Scale Output To Zero	ARRAY	UNSIGNED8	RW	No
3400	Global Enable Input Source	VARIABLE	UNSIGNED8	RW	No
3410	Global Enable Input Source Inv	VARIABLE	UNSIGNED8	RW	No
3500	Global Constant Output Source 1	VARIABLE	FLOAT32	RW	No
3510	Global Constant Output Source 2	VARIABLE	FLOAT32	RW	No
3600	Binary Funcs' Output Values	ARRAY	FLOAT32	RO	Yes
3601	Global Parameter Values	ARRAY	FLOAT32	RO	Yes
5555	Start in Operational	VARIABLE	UNSIGNED8	RW	No
6302	AO Decimal Digits PV	ARRAY	UNSIGNED8	RW	No
6310	AO Output Type	ARRAY	UNSIGNED8	RW	No
7300	AO Output Process Value	ARRAY	INTEGER16	RO	Yes
7320	AO Output Scaling 1 PV	ARRAY	INTEGER16	RW	No
7321	AO Output Scaling 1 FV	ARRAY	INTEGER16	RW	No
7322	AO Output Scaling 2 PV	ARRAY	INTEGER16	RW	No
7323	AO Output Scaling 2 FV	ARRAY	INTEGER16	RW	No
7330	AO Output Field Value	ARRAY	INTEGER16	RW	Yes
6340	AO Fault Mode	ARRAY	UNSIGNED16	RW	No
7341	AO Fault FV	ARRAY	INTEGER16	RW	No
7F50	Received PV 16	ARRAY	INTEGER16	RW	Yes
6F52	Received PV Status	ARRAY	UNSIGNED8	RW	Yes

4.2.1 Object 2600h: Binary Functions' Input 1 Source

This object is used to select the input source for binary function blocks' input 1. See **Error! Reference source not found.** for a list of all available input sources accepted by this object.

Object Description

Index	2600h
Name	Binary Function Input 1 Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Input 1 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

Subindex	2h
Description	Binary Function 2 Input 1 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

Subindex	3h
Description	Binary Function 3 Input 1 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

Subindex	4h
Description	Binary Function 4 Input 1 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

Subindex	5h
Description	Binary Function 5 Input 1 Source
Access	RW
PDO Mapping	No

Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

4.2.2 Object 2610h: Binary Functions' Input 1 Source Inversion

This object is used to select the input source inversion for binary function blocks' input 1.

Object Description

Index	2610h
Name	Binary Function Input 1 Source Inv
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Input 1 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

Subindex	2h
Description	Binary Function 2 Input 1 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

Subindex	3h
Description	Binary Function 3 Input 1 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

Subindex	4h
Description	Binary Function 4 Input 1 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

Subindex	5h
Description	Binary Function 5 Input 1 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

4.2.3 Object 2620h: Binary Functions' Input 1 Scale

This object is used to select the input scale for binary function blocks' input 1.

Object Description

Index	2620h
Name	Binary Function Input 1 Scale
Object Type	ARRAY
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Input 1 Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	2h
Description	Binary Function 2 Input 1 Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	3h
Description	Binary Function 3 Input 1 Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	4h
Description	Binary Function 4 Input 1 Scale
Access	RW
PDO Mapping	No

Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	5h
Description	Binary Function 5 Input 1 Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

4.2.4 Object 2700h: Binary Functions' Input 2 Source

This object is used to select the input source for binary function blocks' input 1. See **Error! Reference source not found.** for a list of all available input sources accepted by this object.

Object Description

Index	2700h
Name	Binary Function Input 2 Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Input 2 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

Subindex	2h
Description	Binary Function 2 Input 2 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

Subindex	3h
Description	Binary Function 3 Input 2 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference**

	source not found.)
Default Value	0 (No source)

Subindex	4h
Description	Binary Function 4 Input 2 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

Subindex	5h
Description	Binary Function 5 Input 2 Source
Access	RW
PDO Mapping	No
Value Range	0 … 16 (see **Error! Reference source not found.**)
Default Value	0 (No source)

4.2.5 Object 2710h: Binary Functions' Input 2 Source Inversion

This object is used to select the input source inversion for binary function blocks' input 2.

Object Description

Index	2710h
Name	Binary Function Input 2 Source Inv
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Input 2 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

Subindex	2h
Description	Binary Function 2 Input 2 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

Subindex	3h
Description	Binary Function 3 Input 2 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

Subindex	4h
Description	Binary Function 4 Input 2 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

Subindex	5h
Description	Binary Function 5 Input 2 Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

4.2.6 Object 2720h: Binary Functions' Input 2 Scale

This object is used to select the input scale for binary function blocks' input 2.

Object Description

Index	2720h
Name	Binary Function Input 2 Scale
Object Type	ARRAY
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Input 2 Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	2h
Description	Binary Function 2 Input 2 Scale
Access	RW

PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	3h
Description	Binary Function 3 Input 2 Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	4h
Description	Binary Function 4 Input 2 Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	5h
Description	Binary Function 5 Input 2 Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

4.2.7 Object 2800h: Binary Functions' Output Scale

This object is used to select the output scale for binary function blocks.

Object Description

Index	2800h
Name	Binary Functions' Output Scale
Object Type	ARRAY
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Output Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	2h

Description	Binary Function 2 Output Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	3h
Description	Binary Function 3 Output Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	4h
Description	Binary Function 4 Output Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

Subindex	5h
Description	Binary Function 5 Output Scale
Access	RW
PDO Mapping	No
Value Range	-1.0 … 1.0
Default Value	1.0

4.2.8 Object 2900h: Binary Functions' Names

This object is used to select the name and the operation for binary function blocks. See **Error! Reference source not found.** for a list of all available values for this object.

Object Description

Index	2900h
Name	Binary Functions' Name
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Name
Access	RW
PDO Mapping	No

Value Range	0 … 12 (see **Error! Reference source not found.**)
Default Value	0 (Addition)

Subindex	2h
Description	Binary Function 2 Name
Access	RW
PDO Mapping	No
Value Range	0 … 12 (see **Error! Reference source not found.**)
Default Value	0 (Addition)

Subindex	3h
Description	Binary Function 3 Name
Access	RW
PDO Mapping	No
Value Range	0 … 12 (see **Error! Reference source not found.**)
Default Value	0 (Addition)

Subindex	4h
Description	Binary Function 4 Name
Access	RW
PDO Mapping	No
Value Range	0 … 12 (see **Error! Reference source not found.**)
Default Value	0 (Addition)

Subindex	5h
Description	Binary Function 5 Name
Access	RW
PDO Mapping	No
Value Range	0 … 12 (see **Error! Reference source not found.**)
Default Value	0 (Addition)

4.2.9 Object 3000h: AO Output Control Source

This object is used to select the control signal for Analog Output Blocks. Valid values for this object are listed in **Error! Reference source not found.**.

Object Description

Index	3000h
Name	AO Output Control Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)

Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	AO Output 1 Control Source
Access	RW
PDO Mapping	No
Value Range	0 … 17 (see **Error! Reference source not found.**)
Default Value	6 (CAN FV/PV subindex 1)

4.2.10 Object 3010h: AO Output Control Source Inversion

This object is used to select whether the control signal for Analog Output Blocks should be inverted.

Object Description

Index	3010h
Name	AO Output Control Source Inv
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	AO Output 1 Control Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (No inversion)

4.2.11 Object 30D0h: AO Output Error Flag Delay

This object defines the delay from Output Error flag (set by analog output functional block) until the controller enters into operational mode defined by 1029h, 6341h and 7341h.

Object Description

Index	30D0h
Name	AO Output Error Flag Delay
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	AO Output 1 Error Flag Delay
Access	RW
PDO Mapping	No
Value Range	0 … 10000 [ms]
Default Value	1000

4.2.12 Object 30D1h: AO Output Error Threshold

This object defines the error threshold for the output.

Object Description

Index	30D1h
Name	AO Output Error Threshold
Object Type	ARRAY
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	AO Output 1 Error Threshold
Access	RW
PDO Mapping	No
Value Range	0.0 … 100.0 [%]
Default Value	5.0

4.2.13 Object 30D2h: AO Output Error Delay

This object defines the error delay for the analog output functional block. This is the first delay for detecting the error condition. The AO Output Error Flag Delay (30D0h) is in effect after this delay has elapsed.

Object Description

Index	30D2h
Name	AO Output Error Delay
Object Type	ARRAY
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	AO Output 1 Error Delay
Access	RW
PDO Mapping	No
Value Range	0.0 … 300.0 [s]
Default Value	0.02

4.2.14 Object 3200h: AO Voltage Minimum

This object defines the minimum voltage value for the output. This value is used in the control signal normalization process. Note, that this value is internal to the controller. The PV and FV Scaling coefficients (0x7320, 0x7321, 0x7322 and 0x7323) are the preferred objects for minimum and maximum values setting.

Object Description

Index	3200h
Name	AO Voltage Minimum
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	Minimum voltage, OUT1
Access	RW
PDO Mapping	No
Value Range	-10000 … 10000 [mV]
Default Value	0

4.2.15 Object 3210h: AO Voltage Maximum

This object defines the maximum voltage value for the output. This value is used in the control signal normalization process. Note, that this value is internal to the controller. The PV and FV Scaling coefficients (0x7320, 0x7321, 0x7322 and 0x7323) are the preferred objects for minimum and maximum values setting.

Object Description

Index	3210h
Name	AO Voltage Maximum

Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	Maximum voltage, OUT1
Access	RW
PDO Mapping	No
Value Range	-10000 … 10000 [mV]
Default Value	5000

4.2.16 Object 3220h: AO Current Minimum

This object defines the minimum current value for the output. This value is used in the control signal normalization process. Note, that this value is internal to the controller. The PV and FV Scaling coefficients (0x7320, 0x7321, 0x7322 and 0x7323) are the preferred objects for minimum and maximum values setting.

Object Description

Index	3220h
Name	AO Current Minimum
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	Minimum current, OUT1
Access	RW
PDO Mapping	No
Value Range	-20 … 20 [mA]
Default Value	4

4.2.17 Object 3230h: AO Current Maximum

This object defines the maximum current value for the output. This value is used in the control signal normalization process. Note, that this value is internal to the controller. The PV and FV Scaling coefficients (0x7320, 0x7321, 0x7322 and 0x7323) are the preferred objects for minimum and maximum values setting.

Object Description

Index	3230h
Name	AO Current Maximum
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	Maximum current, OUT1
Access	RW
PDO Mapping	No
Value Range	-20 … 20 [mA]
Default Value	20

4.2.18 Object 3240h: Scale Output To Zero

When using negative voltages as minimum output voltage, this object can be used to select whether the output is scaled to zero when the device is not operational. If the output is not scaled to zero, the output will be set equal to minimum output voltage when the device is not operational.

Object Description

Index	3240h
Name	Scale Output To Zero
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	Scale Output 1 To Zero
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	1

4.2.19 Object 3400h: Global Enable Input Source

This object can be used to enable or disable all Analog Output Blocks. See section **Error! Reference source not found.** for more details.

Object Description

Index	3400h
Name	Global Enable Input Source
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Global Enable Input Source
Access	RW
PDO Mapping	No
Value Range	0 … 17 (see **Error! Reference source not found.**)
Default Value	17 (all enabled)

4.2.20 Object 3410h: Global Enable Input Source Inversion

This object selects whether the inputs to all Analog Output Blocks should be inverted. See section **Error! Reference source not found.** for more details.

Object Description

Index	3410h
Name	Global Enable Input Source Inv
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Global Enable Input Source Inv
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0

4.2.21 Object 3500h: Global Configurable Output Constant 1

This object defines the global configurable output constant 1, which can be used as a source for Binary Function Blocks and Analog Output Blocks.

Object Description

Index	3500h
Name	Global Configurable Output Constant 1
Object Type	VARIABLE
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Global Configurable Output Constant 1
Access	RW

PDO Mapping	No
Value Range	0.0 … 1.0
Default Value	0.0

4.2.22 Object 3510h: Global Configurable Output Constant 2

This object defines the global configurable output constant 2, which can be used as a source for Binary Function Blocks and Analog Output Blocks.

Object Description

Index	3510h
Name	Global Configurable Output Constant 2
Object Type	VARIABLE
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Global Configurable Output Constant 2
Access	RW
PDO Mapping	No
Value Range	0.0 … 1.0
Default Value	0.0

4.2.23 Object 3600h: Binary Functions' Output Values

This object lists the current output values of the Binary Function Blocks. This object is read-only and targeted for debug purposes.

Object Description

Index	3600h
Name	Binary Functions' Output Values
Object Type	ARRAY
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Subindex	1h
Description	Binary Function 1 Output
Access	RO
PDO Mapping	Yes
Value Range	-1.0 … 1.0
Default Value	0.0

Subindex	2h
Description	Binary Function 2 Output
Access	RO

PDO Mapping	Yes
Value Range	-1.0 … 1.0
Default Value	0.0

Subindex	3h
Description	Binary Function 3 Output
Access	RO
PDO Mapping	Yes
Value Range	-1.0 … 1.0
Default Value	0.0

Subindex	4h
Description	Binary Function 4 Output
Access	RO
PDO Mapping	Yes
Value Range	-1.0 … 1.0
Default Value	0.0

Subindex	5h
Description	Binary Function 5 Output
Access	RO
PDO Mapping	Yes
Value Range	-1.0 … 1.0
Default Value	0.0

4.2.24 Object 3601h: Global Parameter Values

This object list the global parameters of the board, such as operating voltage, temperature and the two configurable coefficients' values.

Object Description

Index	3601h
Name	Global Parameter Values
Object Type	ARRAY
Data Type	FLOAT32

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Subindex	1h
Description	Operating voltage
Access	RO
PDO Mapping	Yes
Value Range	[V]
Default Value	

Subindex	2h
Description	Board temperature
Access	RO
PDO Mapping	Yes
Value Range	[°C]
Default Value	

Subindex	3h
Description	Global configurable coefficient 1
Access	RO
PDO Mapping	Yes
Value Range	0.0 … 1.0
Default Value	

Subindex	4h
Description	Global configurable coefficient 2
Access	RO
PDO Mapping	Yes
Value Range	0.0 … 1.0
Default Value	

4.2.25 Object 5555h: Start in Operational

This object defines the mode of operation in which the controller starts. The controller can be configured to start in operational mode, in which the controller will transmit all configured TX PDOs. The controller can also be configured to send a NMT message for starting also other devices on the network.

Object Description

Index	5555h
Name	Start in Operational
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Start in Operational
Access	RW
PDO Mapping	No
Value Range	0 – Start in PRE-OPERATIONAL 1 – Start in OPERATIONAL 2 – Start in OPERATIONAL and send NMT to other nodes
Default Value	0

4.2.26 Object 7300h: AO Output Process Value

This object is the process value that is fed into the analog output function block. Since this is a read-only object, the only way this object is written is if the value in object 7F50h [Received PV] is validated by object 6F52h. The process value can be in any physical unit (bar, rpm, etc.) or in any

custom format. The output value is scaled into the field value using the scaling coefficients defined in objects 7320h to 7323h.

Object Description

Index	7300h
Name	AO Output Process Value
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	Process Value, CAN input 1
Access	RO
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	0

Subindex	2h
Description	Process Value, CAN input 2
Access	RO
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	0

Subindex	3h
Description	Process Value, CAN input 3
Access	RO
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	0

4.2.27 Object 6302h: AO Decimal Digits PV

This object sets the number of decimal digits included in the process value.

Object Description

Index	6302h
Name	AO Decimal Digits PV
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)

Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	Decimal Digits PV, CAN input 1
Access	RW
PDO Mapping	No
Value Range	0 to 9
Default Value	0

Subindex	2h
Description	Decimal Digits PV, CAN input 2
Access	RW
PDO Mapping	No
Value Range	0 to 9
Default Value	0

Subindex	3h
Description	Decimal Digits PV, CAN input 3
Access	RW
PDO Mapping	No
Value Range	0 to 9
Default Value	0

4.2.28 Object 6310h: AO Output Type

This object defines the type of the output. The following values are allowed for this object.

- Output Type 0 = voltage output
- Output Type 1 = current output

Only the mentioned output types are supported. Writing a value other than those listed above will result in SDO abort download error, and the previous value will remain unchanged. The default value is 0, voltage output.

Object Description

Index	6310h
Name	AO Output Type
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	Output Type, OUT1
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0 (voltage)

4.2.29 Object 7320h: AO Output Scaling 1 PV

This object defines the process value for the first calibration point for the output channel. The value is scaled in the physical unit of the output process value, object 7300h.

Object Description

Index	7320h
Name	AO Output Scaling 1 PV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	Scaling 1 PV, CAN input 1
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	0

Subindex	2h
Description	Scaling 1 PV, CAN input 2
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	0

Subindex	3h
Description	Scaling 1 PV, CAN input 3
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	0

4.2.30 Object 7321h: AO Output Scaling 1 FV

This object defines the field value for the first calibration point for the output channel. The value is scaled in the physical unit of the output field value, object 7330h.

Object Description

Index	7321h
Name	AO Output Scaling 1 FV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	Scaling 1 FV, CAN input 1
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	0

Subindex	2h
Description	Scaling 1 FV, CAN input 2
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	0

Subindex	3h
Description	Scaling 1 FV, CAN input 3
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	0

4.2.31 Object 7322h: AO Output Scaling 2 PV

This object defines the process value for the second calibration point for the output channel. The value is scaled in the physical unit of the output process value, object 7300h.

Object Description

Index	7322h
Name	AO Output Scaling 2 PV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	Scaling 2 PV, CAN input 1
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	1000

Subindex	2h
Description	Scaling 2 PV, CAN input 2
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	1000

Subindex	3h
Description	Scaling 2 PV, CAN input 3
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	1000

4.2.32 Object 7323h: AO Output Scaling 2 FV

This object defines the field value for the second calibration point for the output channel. The value is scaled in the physical unit of the output field value, object 7330h.

Object Description

Index	7323h
Name	AO Output Scaling 2 FV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	Scaling 2 FV, CAN input 1

Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	1000

Subindex	2h
Description	Scaling 2 FV, CAN input 2
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	1000

Subindex	3h
Description	Scaling 2 FV, CAN input 3
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	1000

4.2.33 Object 7330h: AO Output Field Value

This object defines the field value of the proportional outputs. The FV is automatically updated if object 7300h [AO Output Process Value] is changed. Alternatively, it can be written to directly to set the output, as long as the corresponding bit in object 6F52h [Received PV status] is set to invalid. The value is defined as a multiple of 1mA / 1mV. The value range of this object depends on the selected output type.

Object Description

Index	7330h
Name	AO Output Field Value
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	Field Value, CAN input 1
Access	RW
PDO Mapping	Yes
Value Range	-20 to 20 [mA], -10000 to 10000 [mV]
Default Value	0

Subindex	2h

Description	Field Value, CAN input 2
Access	RW
PDO Mapping	Yes
Value Range	-20 to 20 [mA], -10000 to 10000 [mV]
Default Value	0

Subindex	3h
Description	Field Value, CAN input 3
Access	RW
PDO Mapping	Yes
Value Range	-20 to 20 [mA], -10000 to 10000 [mV]
Default Value	0

4.2.34 Object 6340h: AO Fault Mode

This object defines the fault mode response for proportional outputs. It determines whether the output shall continue to operate normally when the 1 Analog Output CAN Controller detects an error, or if the output shall be driven to the value defined in object 7341h.

- Value 0 = output continues to operate normally in controller fault state
- Value 1 = write the value defined in object 7341h in object 7330h, in controller fault state

Object Description

Index	6340h
Name	AO Fault Mode
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	Fault Mode, Output 1
Access	RW
PDO Mapping	No
Value Range	0, 1
Default Value	0

4.2.35 Object 7341h: AO Fault FV

This object defines the value that proportional outputs will be driven to in fault situations. Value range of this object depends on the selected output type.

Object Description

Index	7341h
Name	AO Fault FV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Subindex	1h
Description	Fault FV, OUT1
Access	RW
PDO Mapping	No
Value Range	-20 to 20 [mA], -10000 to 10000 [mV]
Default Value	0

4.2.36 Object 7F50h: Received PV 16

This object is a generic input process value that is used to write to the CAN input function block's process value, object 7300h. If and only if the value of object 6F52h at the same subindex is set to validate the PV will the data from 7F50h be copied to object 7300h. Note, that the device needs to be set to OPERATIONAL mode for writing successfully into this object.

Object Description

Index	7F50h
Name	Received PV 16
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	Received PV, CAN input 1
Access	RW
PDO Mapping	Yes
Value Range	INTERGER16
Default Value	0

Subindex	2h
Description	Received PV, CAN input 2
Access	RW
PDO Mapping	Yes
Value Range	INTERGER16
Default Value	0

Subindex	3h
Description	Received PV, CAN input 3
Access	RW
PDO Mapping	Yes
Value Range	INTERGER16
Default Value	0

4.2.37 Object 6F52h: Received PV Status

This object is used to validate the value in object 7F50h [Received PV] such that the value will be copied to object 7300h [AO output PV] and subsequently used as the output command. If this object is set to the non-validated value, the Received PV is not used. However, writes to object 7330h [AO Output FV] is allowed at any time, thus the field value can be used to control the CAN input blocks, even when the process values have not been validated.

The value of object 6F52h is invalidated after the value in object 7F50h is copied to 7300h. Therefore, if the object is not mapped into a PDO, the user must validate the Received PV by SDO writes before the Received PV will be used. Note, that the device needs to be set to OPERATIONAL mode for writing successfully into this object.

The following status values are allowed for object 6F52h.

- Value 0 = Received PV value not validated (writes to object 7F50h are ignored)
- Value 1 = Received PV value is validated (value in object 7F50h is copied to 7300h)

Object Description

Index	6F52h
Name	Received PV Status
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Subindex	0h
Description	Number of entries (NrOfObjects)
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Subindex	1h
Description	PV Status, CAN input 1
Access	RW
PDO Mapping	Yes
Value Range	0, 1

Default Value	0 (Received PV invalid)

Subindex	2h
Description	PV Status, CAN input 2
Access	RW
PDO Mapping	Yes
Value Range	0, 1
Default Value	0 (Received PV invalid)

Subindex	3h
Description	PV Status, CAN input 3
Access	RW
PDO Mapping	Yes
Value Range	0, 1
Default Value	0 (Received PV invalid)

5. FIRMWARE FLASHING

The controller does not support a simple in-application flashing procedure for reprogramming of the new firmware. It is assumed that in case the firmware upgrade is required, the unit is returned to the manufacturer for flashing.

In some special cases, however, the firmware can be reprogrammed through an internal service port in the field by a qualified technician. The flashing instructions, together with a firmware file, RS232 converter and a cable harness, can be obtained from Axiomatic on request.

6. TECHNICAL SPECIFICATIONS

Input Specifications

Power Supply Input - Nominal	12V, 24V or 48VDC nominal (9…60 VDC power supply range)
Protection	Surge and reverse polarity protection are provided.
Input	CAN Messages, SAE J1939 {CANopen® available on request} The CAN signal can be filtered to accept messages from a single address on the network permitting a link to a specific ECU. There are three CAN Input Signal functional blocks supported by the controller. Each functional block can be programmed to read single-frame CAN messages and extract CAN signal data presented in virtually any user-defined signal data format. The functional block then outputs the signal data to its logical output for processing by other functional blocks of the controller. (Refer to Figure 1.0.) By default, the output of the first CAN Input Signal functional block is connected to the input of the Analog Signal Output functional block. It provides the simplest controller configuration with a direct control of the signal output by the CAN input signal. The second and third CAN Input Signal functional blocks, not connected by default, can be engaged in more complicated CAN signal acquisition and processing algorithms involving Binary Function functional blocks and other controller resources. The Electronic Assistant® (EA) is used to set up CAN signal acquisition and processing algorithms.

Output Specifications

CAN	The controller can send a single frame application specific CAN message to the network continuously or on request. Using the EA, the user can configure this feature.
Analog Outputs	1 analog signal output Refer to Table 1.0.
Ground Connection	1 Analog GND connection is provided.
Protection for Output + Terminal	Fully protected against short circuit to ground and short circuit to power supply rail. Unit will fail safe in the case of a short circuit condition, self-recovering when the short is removed.

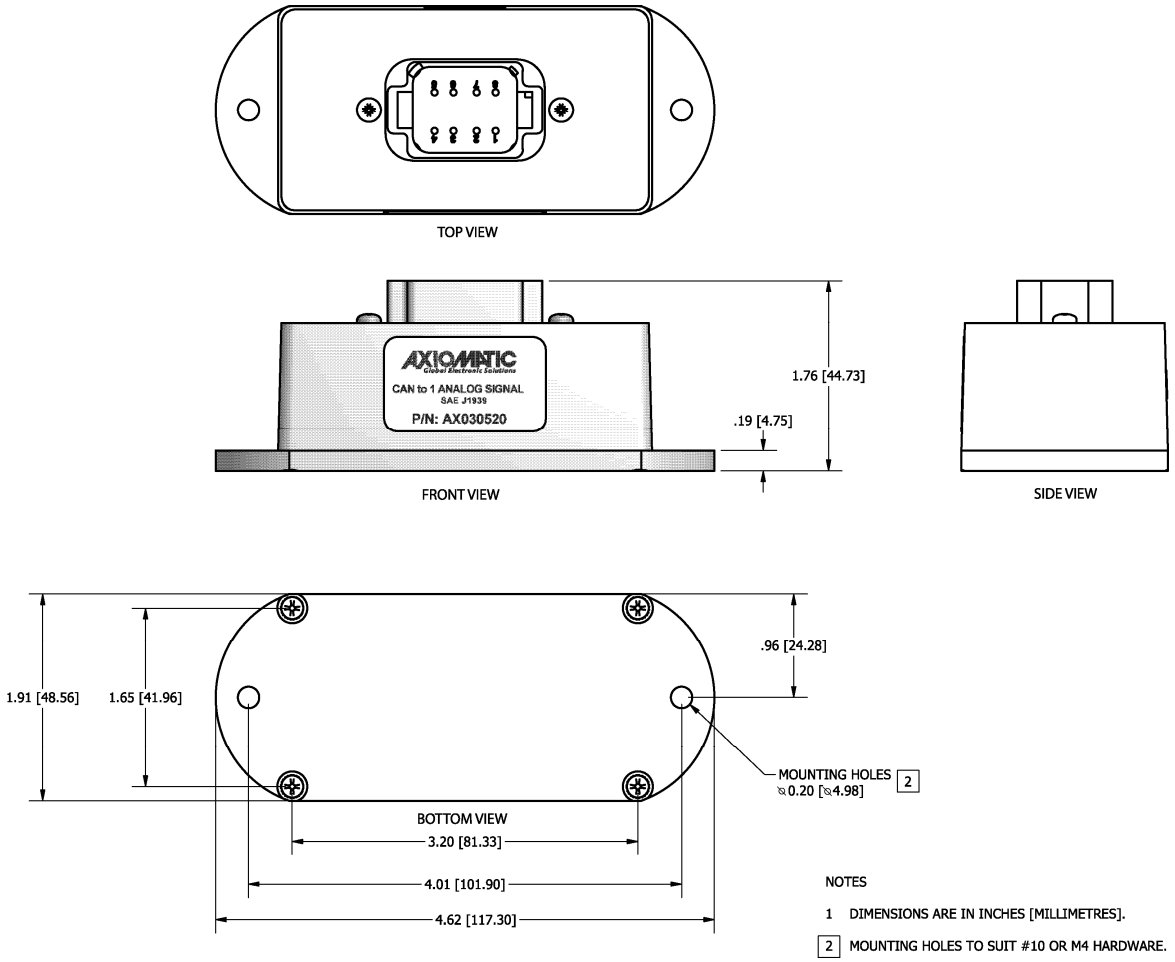
Table 1.0 - Outputs

Analog Output	1 analog signal output with embedded voltage and current monitoring circuits Using the Electronic Assistant®, the user selects: • the output mode (voltage or current); • the minimum and maximum values for the output signal from the +/-10V or +/-20 mA range. Standard analog signal ranges are supported, including: 0-5V; 0-10V; +/-5V; +/-10V; 0-20mA; or 4-20 mA. The output can be globally enabled or disabled.
Output Accuracy	≤0.5% at: +/-5V,+/-10V,+/-20mA Absolute accuracy is the same for other output signal ranges.
Output Resolution	0.015% (>12 bit)
Voltage Monitoring Range	+/-12V
Current Monitoring Range	+/-25mA Voltage on the load should be within +/-8V voltage range.
Voltage and Current Monitoring Accuracy	≤2.5% Please, contact Axiomatic if accuracy ≤1% is required.

General Specifications

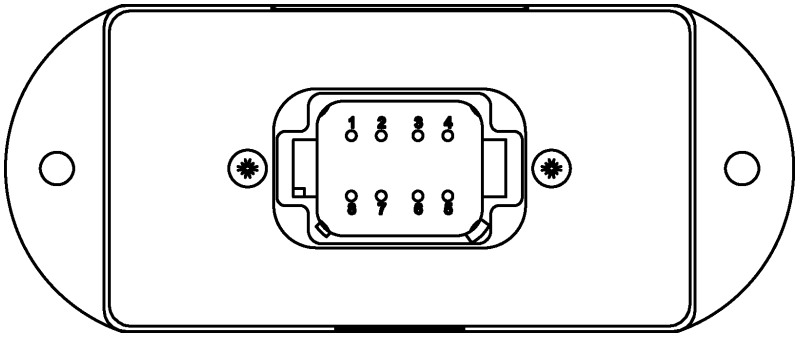
Microprocessor	32-bit, 128 KByte flash program memory
Control Logic	Standard embedded software is provided. (Application-specific control logic or factory programmed setpoints are available on request.) The controller belongs to a family of Axiomatic smart controllers with programmable internal architecture. This provides users with an ultimate flexibility, allowing them to build their own custom controller with a required functionality from a set of predefined internal functional blocks using the PC-based Axiomatic Electronic Assistant® software tool. Application programming is performed through CAN interface, without disconnecting the controller from the user's system.
CAN	1 CAN port (SAE J1939) (CANopen® on request)
Slew Rate	To adjust the controller to the CAN physical network, the slew rate can be configured as fast or slow.
Monitoring and Debugging	Besides reading application signals transmitted on the CAN bus, the controller can also transmit CAN application messages carrying signals internally generated by the controller. This feature can be used for monitoring the analog signal output and for debugging purposes.
User Interface (PC-based)	The controller setpoints can be viewed and programmed using the standard J1939 memory access protocol through the CAN port and the PC-based Axiomatic Electronic Assistant®. For default setpoints, refer to the User Manual. The EA can store all controller setpoints in one setpoint file and then flash them into the controller in one operation. The setpoint file is created and stored on disk using a command *Save Setpoint File* from the EA menu or toolbar. The user then can open the setpoint file, view or print it and flash the setpoint file into the controller. The Electronic Assistant® for *Windows* operating systems comes with a royalty-free license for use on multiple computers. It requires an USB-CAN converter to link the device's CAN port to a *Windows*-based PC. An Axiomatic USB-CAN Converter AX070501 is available as part of the Axiomatic Configuration KIT. **P/N: AX070502**, the Axiomatic Configuration KIT includes the following. USB-CAN Converter P/N: AX070501 1 ft. (0.3 m) USB Cable P/N: CBL-USB-AB-MM-1.5 12 in. (30 cm) CAN Cable with female DB-9 P/N: CAB-AX070501 AX070502IN CD P/N: CD-AX070502, includes: Electronic Assistant® software; EA & USB-CAN User Manual UMAX07050X; USB-CAN drivers & documentation; CAN Assistant (Scope and Visual) software & documentation; and the SDK Software Development Kit.
Typical Quiescent Current Draw	54 mA @ 12VDC, 29 mA @ 24VDC. 17 mA @ 48 VDC
Settling Time	≤ 5 mSec. (0…95%)
Weight	0.65 lbs. (0.29 kg)
Operating Conditions	-40 to 85 °C (-40 to 185 °F)
Storage Temperature	-55 to 125 °C (-67 to 257°F)
Protection	IP67 PCB is conformal coated and protected by the housing.
Packaging and Dimensions	Encapsulated Cast Aluminum housing with mounting holes 4.62 x 1.91 x 1.76 inches (117.30 x 48.56 x 44.73 mm) L x W x H including integral connector

DIMENSIONAL DRAWING



Mounting	Mounting holes – The controller accepts 2 #10 or M4 screws. The CAN wiring is considered intrinsically safe. The power wires are not considered intrinsically safe and so in hazardous locations, they need to be located in conduit or conduit trays at all times. The module must be mounted in an enclosure in hazardous locations for this purpose. All field wiring should be suitable for the operating temperature range. Install the unit with appropriate space available for servicing and for adequate wire harness access (6 inches or 15 cm) and strain relief (12 inches or 30 cm).
Network Termination	It is necessary to terminate the network with external termination resistors. The resistors are 120 Ohm, 0.25W minimum, metal film or similar type. They should be placed between CAN_H and CAN_L terminals at both ends of the network.

Electrical Connections



Deutsch DT series 8 pin plug (DT15-08PA)

Mating plug KIT: Axiomatic P/N AX070112
(Comprised of Deutsch IPD P/n's: DT06-08SA socket, wedge W8S, 7 solid contact sockets 0462-201-16141 and 1 sealing plug 114017.)

16-18 AWG wire is recommended for use with sockets 0462-201-16141.

Use dielectric grease on the pins when installing the controller.
Wiring to these mating plugs must be in accordance with all applicable local codes. Suitable field wiring for the rated voltage and current must be used. The rating of the connecting cables must be at least 70°C. Use field wiring suitable for both minimum and maximum ambient temperature.

PIN #	FUNCTION
1	POWER +
8	POWER -
2	NOT USED
7	CAN SHIELD
3	ANALOG SIGNAL OUTPUT
6	CAN_L
4	AGND
5	CAN_H

7. REVISION HISTORY

Firmware	Manual Revision	Date	Author	Changes
V1.00	Rev A	9. Feb. 2011	Antti Keränen	Initial release
-	-	22. March 2018	Amanda Wilkins	Updated connector p/n and mating plug p/n

OUR PRODUCTS

Actuator Controls

Battery Chargers

CAN bus Controls, Gateways

CAN/Wifi, CAN/Bluetooth

Current Converters

DC/DC Power Converters

DC Voltage/Current Signal Converters

Engine Temperature Scanners

Ethernet/CAN Converters

Fan Drive Controllers

Hydraulic Valve Controllers

I/O Controls

LVDT Simulators

Machine Controls

Motor Controls

PID Controls

Position Sensors, Angle Measurement Inclinometers

Power Supplies

PWM Signal Converters/Isolators

Resolver Signal Conditioners

Service Tools

Signal Conditioners

Strain Gauge CAN Controls

Surge Suppressors

OUR COMPANY

Axiomatic provides electronic machine controls, components, and systems to the off-highway, commercial vehicle, electric vehicle, power generator set, material handling, renewable energy and industrial OEM markets.

We provide efficient, innovative solutions that focus on adding value for our customers.

We emphasize service and partnership with our customers, suppliers, and employees to build long term relationships and mutual trust.

QUALITY DESIGN AND MANUFACTURING

Axiomatic is an ISO 9001:2008 registered facility.

SERVICE

All products to be returned to Axiomatic require a Return Materials Authorization Number (RMA#).

Please provide the following information when requesting an RMA number:

- Serial number, part number
- Axiomatic invoice number and date
- Hours of operation, description of problem
- Wiring set up diagram, application
- Other comments as needed

When preparing the return shipping paperwork, please note the following. The commercial invoice for customs (and packing slip) should state the harmonized international HS (tariff code), valuation and return goods terminology, as shown in italics below. The value of the units on the commercial invoice should be identical to their purchase price.

Goods Made In Canada (or Finland)
Returned Goods for Warranty Evaluation, HS: 9813.00
Valuation Identical Goods
Axiomatic RMA#

WARRANTY, APPLICATION APPROVALS/LIMITATIONS

Axiomatic Technologies Corporation reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. Users should satisfy themselves that the product is suitable for use in the intended application. All our products carry a limited warranty against defects in material and workmanship. Please refer to our Warranty, Application Approvals/Limitations and Return Materials Process as described on www.axiomatic.com/service.html.

CONTACTS

Axiomatic Technologies Corporation
5915 Wallace Street
Mississauga, ON
CANADA L4Z 1Z8
TEL: +1 905 602 9270
FAX: +1 905 602 9279
www.axiomatic.com

Axiomatic Technologies Oy
Höytämöntie 6
33880 Lempäälä
FINLAND
TEL: +358 3 3595 600
FAX: +358 3 3595 660
www.axiomatic.fi