USER MANUAL UMAX1307x1
Version 1B

CAN TO
2 ANALOG/DIGITAL OUTPUTS
AND 1 RELAY
CONVERTER

CANopen®

USER MANUAL

P/N: AX130751
P/N: AX130771

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

ACRONYMS

AO	Analog Output (Universal)
CAN	Controller Area Network
CANopen®	CANopen® is a registered community trademark of CAN in Automation e.V.
CAN-ID	CAN 11-bit Identifier
COB	Communication Object
CTRL	Control
DO	Digital Output
EDS	Electronic Data Sheet
EMCY	Emergency
LSB	Least Significant Byte (or Bit)
LSS	Layer Settling Service
MSB	Most Significant Byte (or Bit)
NMT	Network Management
PID	Proportional-Integral-Derivative Control
PWM	Pulse Width Modulation
RO	Read Only Object
RPDO	Received Process Data Object
RPM	Rotations per Minute
RW	Read/Write Object
SDO	Service Data Object
TPDO	Transmitted Process Data Object
WO	Write Only Object

REFERENCES

[DS-301] CiA DS-301 V4.1 – CANopen® Application Layer and Communication Profile. CAN in Automation 2005

[DS-305] CiA DS-305 V2.0 – Layer Setting Service (LSS) and Protocols. CAN in Automation 2006

[DS-404] CiA DS-404 V1.2 – CANopen® profile for Measurement Devices and Closed Loop Controllers. CAN in Automation 2002

These documents are available from the CAN in Automation e.V. website http://www.can-cia.org/

NOTE: When a description is **bolded**, this refers to the name of a user configurable object (variable). If it is *italicized*, it refers to an option for the associated object. For example: **DO Type** sets to *Relay Output Command*

Contents

1. OVERVIEW OF CONTROLLER

1.1. Description of CAN to 2 Analog/Digital Outputs and 1 Relay Converter

The following User Manual describes the architecture and functionality of a CAN to 2 Analog/Digital Outputs and 1 Relay CANopen® Converter.

This controller is designed for versatile control of CAN bus to 2 analog/digital outputs and a relay output (CAN-2AOUT-1RLYOUT). The hardware design allows for the controller to have a wide range of output types: *Analog Current, Analog Voltage, Digital PWM, Digital Frequency and Digital ON/OFF.*

The control algorithms/function blocks allow the user to configure the controller for a wide range of applications without the need for custom firmware. The various function blocks supported by this controller are outlined in the following sections. It accepts power supply voltages from 9 to 36 VDC. All logical function blocks on the unit are inherently independent from one another but can be configured to interact with each other.

The various function blocks supported by the CAN-2AOUT-1RLYOUT are outlined in the following sections. All objects are user configurable using standard commercially available tools that can interact with a CANopen® Object Dictionary via an .EDS file.

1.2. Digital Output Function Block

There are two types of the universal output function blocks, one for the digital outputs and the other for the analog outputs. While both types of function blocks are driving the same outputs, they are just treated as separate function blocks here for ease of explanation. The digital output (DO) function block only becomes applicable on the output when object 6310h, **AO Output Type**, is set to *Digital ON/OFF* output response (Type=1000).

Several other objects are associated with the Analog Output function block and will be explained in detail in section 1.3. Therefore, this section will only elaborate on the objects unique to the DO function block.

An output can be controlled either by an on-board control signal (such as the result from a lookup table function) or a CANopen® object that has mapped to an RPDO. In the case of an output configured for a digital response, when a CANopen® Message has been selected as the 2340h **AO Control Source** (see Table 4A), data from the appropriate sub-index from the write-mappable object 6220h **DO Write State** will be used as the control signal.

The Enable and Override Inputs and Responses for a digital output are the same as for an Analog Output and are evaluated in the same order. Therefore, what will be described in section 1.3 for these also applied for the DO function block.

As with the AO function, the output will respond to the Control Input if and only if the following conditions have been met:

a) No fault is detected for any of the control signals
b) The override command is either ignored or false (Override Input)
c) The output has not been disabled by a secondary signal (Enable Input)

When the output is being driven by the Control Input, the state is logically set to OFF while the Control Input is zero and is set to ON whenever a non-zero value is written. The resulting "DO Drive State" will depend on the object 2241h **DO Polarity**, as per Table 1. By default, *Normal On/Off* logic is used.

Value	Meaning	DO Logic State	DO Drive State
0	**Normal On/Off**	OFF	OFF
		ON	ON
1	Inverse On/Off	OFF	ON
		ON	OFF
2	Latched Logic	ON to OFF	No Change
		OFF to ON	State Change (i.e. OFF to ON)
3	Blinking Logic	OFF	OFF
		ON	Toggling OFF and ON at the rate defined in object 2223h **DO Blink Rate** (in ms)

Table 1 – DO Polarity Options

The read-only mappable object 2370h **AO Feedback FV** will be loaded with the "DO Drive State" (0=OFF, 1=ON) when the output is setup for a digital type.

There is an object 2240h **DO VPS range** determines if the output is at *+5V* or *+12V* when ON.

For *Digital ON/OFF*, if a non-digital control is selected for this type, the command state will be OFF at or below the minimum input, ON at or above the maximum input, and it will not change in between those points. In other words, the input will have built in hysteresis, as shown in Figure 1. This relationship is true for any function block that has a non-digital input mapped to a digital control.

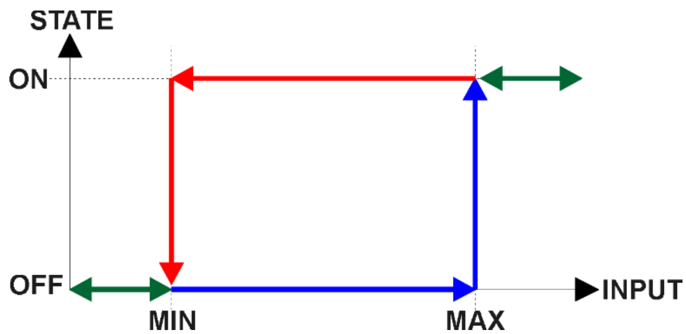


Figure 1 – Analog Input Read as Digital

As mentioned in the Controller Overview, any digital output can be setup to react to a fault detected on any of the ***inputs*** to the function block (control, enable or override.) Should any one of these inputs be showing an error condition, object 6250h **DO Fault Mode** determines how the output will respond, per Table 2. By default, the output will revert to the state defined in object 6260h **DO Fault State** (On/Off).

Value	Meaning
0	Maintain Last State
1	**Apply Pre-Defined State**

Table 2 – Object 6250h - DO Fault Mode Options

1.3. Analog Output Function Block

The analog output (AO) function block is the default logic driving the outputs. There are many objects associated with the analog output function block, but not all of them apply to all output types or control conditions. To start with, object 6310h **AO Output Type** defines how the output drive circuitry will be configures as per Table 3. This table also shows the output units and default ranges for each type. By default, analog outputs are configured as analog voltage types.

Value	Meaning	Range [Unit]
0	Output Not Used	N/A
10	**Output Voltage**	**-10000 to 10000 [mV]**
20	Output Current	0 to 24 [mA]
40	Output PWM	0 to 1000 [0.1%]
500	Output Frequency	50 to 30000 (Hz)
1020	Output Digital On/Off	0 (OFF) or 1 (ON)

Table 3 – Object 6310h - AO Output Type Options

When the output type is changed, all objects related to the output (scaling PV, Decimal Digits PV, etc.) are automatically updated by default. Object 5550h **Enable Auto Updates** enables/disables automatic updates. When disabled (set to False), the objects need to be manually configured.

The control signal of the outputs will have associated with it a minimum and maximum values. Besides type *Digital ON/OFF*, all the other output types are always responding in a linear fashion to changes in the control source per the calculation in Figure 2.

$$y = mx + a$$

$$m = \frac{Y\max - Y\min}{X\max - X\min}$$

$$a = Y\min - m * X\min$$

Figure 2 – Linear Slope Calculations

X and Y are defined as:

Xmin = **"AO Scaling 1 PV"**
Xmax = **"AO Scaling 2 PV"**

Ymin = **"AO Scaling 1 FV"**
Ymax = **"AO Scaling 2 FV"**

In all cases, while the X-axis has the constraint that Xmin < Xmax, there is no such limitation on the Y-axis. This allows for a negative slope so that as the control input signal increases, the target output value decreases. Or it allows output to follow control signal inversely.

Objects 7300h **AO Output PV** can be used to control the outputs. The relationship between the Process Value (input) and the Field Value (output) is a linear one, as shown in Figure 3. However, the output will use the 7321h **AO Scaling 1 FV** as minimum and 7323h **AO Scaling 2 FV** as maximum limits to the drive, such that the output will hold at the minimum and maximum FV points, as shown in the figure.

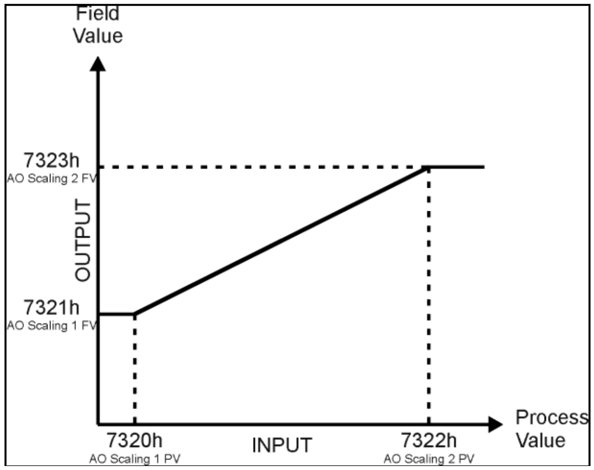


Figure 3 – Analog Output Linear Scaling PV to FV

The controller allows for the PV input can be selected from the list of the logical function blocks supported by the controller. As a result, any output from one function block can be selected as the control source for another. Keep in mind that not all options make sense in all cases, but the complete list of control sources is shown in Table 4A. By default, analog outputs are setup to respond to the corresponding CANopen® RPDO message.

Value	Meaning
0	Control Source Not Used (Ignored)
1	CANopen® Message (RPDO)
2	Relay Command Function Block
3	Lookup Table Function Block
4	Mathematical Function Block
5	Programmable Logic Function Block
6	Output Commanded Field Value
7	Output Feedback Field Value
8	Power Supply Measured
9	Processor Temperature Measured

Table 4A – Control Source Options

In addition to a source, each control also has a number which corresponds to the sub-index of the function block in question. Table 4B outlines the ranges supported for the number objects, depending on the source that had been selected.

Control Source	**Range**	**Object (Meaning)**
Control Source Not Used	0	Ignored
CANopen® Message (RPDO)	1	7300h sub-index 1 or 6220 sub-index 1*
	2	7300h sub-index 2 or 6220 sub-index 2*
	3	6220 sub-index 3 (Relay Output)
	4	2500h sub-index 1 (Extra Received PV 1)
	5	2500h sub-index 2 (Extra Received PV 2)
	6	2500h sub-index 3 (Extra Received PV 3)
	7	2500h sub-index 4 (Extra Received PV 4)
	8	2500h sub-index 5 (Extra Received PV 5)
	9	2500h sub-index 6 (Extra Received PV 6)
** Depending on AO Output Type selected, either* the *AO Output PV or DO Write State will be used*		
Relay Command Function Block	1	6220h sub-index 3
Lookup Table Function Block	1	3017h (Lookup Table 1 Output Y-Axis PV)
	2	3027h (Lookup Table 2 Output Y-Axis PV)
	3	3037h (Lookup Table 3 Output Y-Axis PV)
	4	3047h (Lookup Table 4 Output Y-Axis PV)
	5	3057h (Lookup Table 5 Output Y-Axis PV)
	6	3067h (Lookup Table 6 Output Y-Axis PV)
	7	3077h (Lookup Table 7 Output Y-Axis PV)
	8	3087h (Lookup Table 8 Output Y-Axis PV)
	9	3097h (Lookup Table 9 Output Y-Axis PV)
	10	3107h (Lookup Table 10 Output Y-Axis PV)
Mathematical Function Block	1	4350h sub-index 1 (Math Output PV 1)
	2	4350h sub-index 2 (Math Output PV 2)
	3	4350h sub-index 3 (Math Output PV 3)
	4	4350h sub-index 4 (Math Output PV 4)
Programmable Logic Function Block	1	3xy7h (Lookup Table Selected by Logic 1)
	2	3xy7h (Lookup Table Selected by Logic 2)
	3	3xy7h (Lookup Table Selected by Logic 3)
***NOTE**: The following options should be considered for diagnostic feedback, and should not be selected as a control source for logic inputs (i.e. output control or lookup table X-Axis)*		
Output Commanded Field Value	1	7330h sub-index 1 or “DO1 Logic State”
	2	7330h sub-index 2 or “DO2 Logic State”
Output Feedback Field Value	1	2370h sub-index 1 (AO1 Value or DO1 State)
	2	2370h sub-index 2 (AO2 Value or DO2 State)
Power Supply Measured	N/A	5020h (Power Supply FV)
Processor Temperature Measured	N/A	5030h (Processor Temperature FV)

Table 4B – Control Number Options Depending on Source Selected

There are three inputs to the output function block, each one with a unique source and number object. For the control function (PV axis in Figure 3), objects 2340h **AO Control Input Source** and 2341h **AO Control Input Number** are used. For the enable function, objects 2350h **AO Enable**

Input Source and 2351h **AO Enable Input Number** are used. Lastly, for the override function, objects 2360h **AO Override Input Source** and 2361h **AO Override Input Number** are used.

When using any control source as the X-Axis input to a function block, the corresponding scaling limits are defined as per Table 5. It is the responsibility of the user to make sure that the scaling objects for any function block are setup appropriately depending on the source selected for the X-Axis input.

Note that for the Outputs, the actual objects for the scaling (6302h, 7320h, 7322h) should be edited to match the objects defined in this table when the control source is changed.

Control Source	Scaling 1	Scaling 2	Dec Digits
CANopen® Message – Num 1 to 3	7320h	7322h	6302h
CANopen® Message – Num 4 to 7	2520h	2522h	2502h
Relay Command Function Block	0 (Low)	1 (High)	0 (fixed)
Lookup Table yz Function Block (where yz = 01 to 12)	0 or lowest from 3yz6h(*)	100 or highest from 3yz6h(**)	3yz3h
Mathematical Function	4021h	4023h	4032h
Programmable Logic Function	0%	100%	1 (fixed)
Output Commanded Field Value	7320h	7322h	6302h
Output Feedback Field Value	7320h	7322h	6302h
Power Supply Measured	5022h	5023h	1 (fixed)
Processor Temperature Measured	N/A	5032h	1 (fixed)

***(*) -** Whichever value is smaller; **(**) -** Whichever value is larger*

Table 5 – Scaling Limits per Control Source

As shown in Figure 3, the Output FV will be calculated based on the FV scaling selected. Since 7321h represents the value at or below the lowest control input received, it represents the minimum field value that will be applied at the output. Similarly, 7323h represents the maximum FV that will be applied.

While (7320h < 7322h) must always hold true, in order to get an inverse response (i.e. output decreases as the input increased), simply set 7321h higher than 7323h.

In general, the profile shown in Figure 3 holds true. However, in some cases it may be desired that the minimum offset is not applied when the value is outside of the range, i.e. when using a joystick profile with a deadband. For this reason, object 2342h **AO Control Response** has the options shown in Table 6.

Value	Meaning
0	Single Output Profile (Figure 3)
1	**Output OFF below Scaling 1 PV**
2	Output OFF above Scaling 2 PV

Table 6 – Object 2342 - AO Control Response Options

For the Enable and Override inputs, neither inputs are used by default (control sources are set to 0=Ignore), but they can be activated for safety interlocks or other more complex applications. Table 7 shows the options for object 2352h **AO Enable Response.**

Value	Meaning
0	Enable When ON, Else Shut OFF
1	Enable When ON, Else Ramp OFF
2	Enable When ON, Else Keep Last Value
3	**Enable When OFF, Else Shut OFF**
4	Enable When OFF, Else Ramp OFF
5	Enable When OFF, Else Keep Last Value

Table 7 – Object 2352h - AO Enable Response Options

Table 8 shows options for object 2362h **AO Override Response** respectively. In both cases, the default responses are bolded. When the override is applied, the output is driven to the value defined in object 2300h, **AO Override FV.**

Value	Meaning
0	**Override When ON**
1	Override When OFF

Table 8 – Object 2362h - AO Override Response Options

When the input to the output block goes into an error condition, object 6340 **AO Fault Mode** determines how the output will respond, per Table 9. By default, the output will be driven to the value defined in object 7341h **AO Fault FV**.

Value	Meaning
0	Maintain Last State
1	**Apply Pre-Defined FV**

Table 9 – Object 6340 - AO Fault Mode Options

The controller applies the logic shown in Figure 4 when evaluating what output FV to apply. Under normal conditions, i.e. when the control input is driving the output as shown in the green box, there are ramping objects that can be applied to soften the output response. Object 2330h **AO Ramp Up** and object 2331 **AO Ramp Down** are both millisecond numbers that define how long it will take to ramp from AO Scaling 1 FV to AO Scaling 2 FV. In addition to the read-only mappable object 7330h **AO Output Field Value** (as represented by the green box above), there is another object 2370h **AO Feedback FV**, also read-only mappable. This object reflects the actual value currently driving the output.

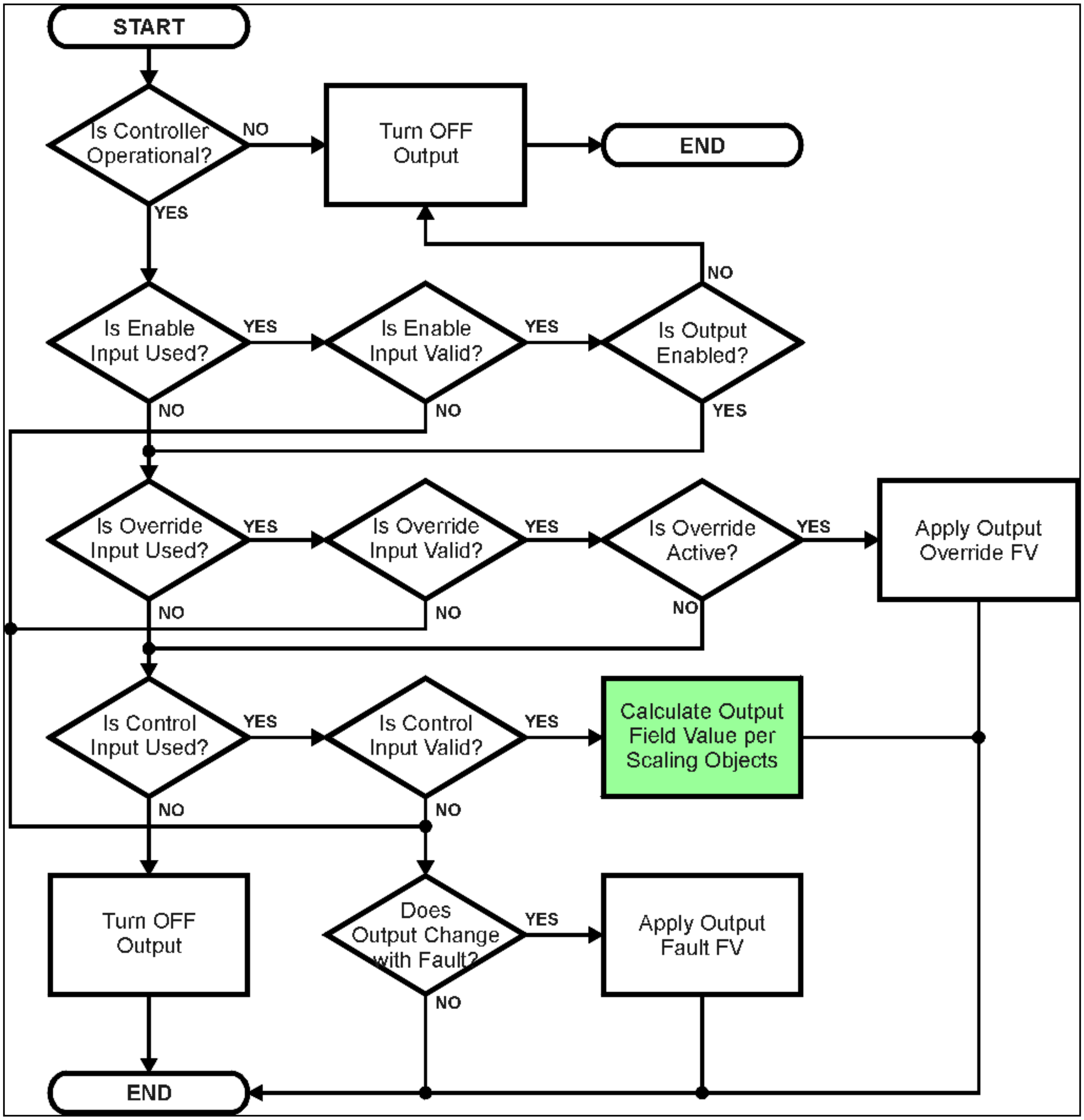


Figure 4 – Analog Output Logic Flowchart

By default, current and voltage outputs are driven with a high frequency 25kHz signal. This value should not be lowered in this mode without taking into consideration how this affects the accuracy and stability of the current through the load. However, in other output mode, Object 2380h **AO Output Frequency** allows for adjusting the frequency of the output.

There is also an object 2240h **DO VPS range** determines if the output is at *+5V* or *+12V* when ON. It does not apply to *Output Voltage* and *Output Current* type.

1.4. Relay Output Function Block

A relay output is also available in the CAN-2AOUT-1RLYOUT converter. It has 3 pins associated with it: Normally Closed (NC), Normally Open (NO), and Common (C) which gives 2 states: Normally Open and Normally Closed. The object 2242h **DO Polarity** allows for flexibility in the response of the output. Table 10 gives the options available for this parameter.

Value	Meaning
0	*Output Not Implemented*
1	***Normal Logic***
2	*Inverse Logic*
3	*Latched Logic*
4	*Inverse Latched Logic*
5	*Toggle Logic*

Table 10 – Relay Output Polarity

By default, Normal Logic response is used for the relay output. the Common pin is connected to the Normally Closed pin when the source of the respective relay output is triggered ON. When the source of the respective relay output is triggered OFF, the Common pin is connected to the Normally Open pin. The relay state will be written to the fourth sub-index of object 6220h DO Write State, while the second and the third sub-indices are for the digital outputs.

In the case of Inverse Logic response, the relay state will be OFF when the source of the respective relay output is triggered ON. When the source of the respective relay output is triggered OFF, the state will be ON.

In the case of *Latched Logic* response, the Common pin is toggled between Normally Closed and Normally Open pins every time the source of the respective relay output goes from OFF to ON, which means the relay state will change every time the source state goes from OFF to ON. The *Inverse Latched Logic* response will respond the opposite way.

The *Toggle Logic* lets the relay output toggle between Normally closed and Normally Open pins for a configured frequency. The time for switching from one state to the other state can be configured with the object 2250h Relay Blink Rate which is in milliseconds and by default 500ms.

The relay output can be configured to be commanded/ enabled/ overridden/ unlatched by the source selected from the list of the logical function blocks supported by the controller in Table. In addition to a source, each control also has a number which corresponds to the sub-index of the function block in question. Table 5 outlines the ranges supported for the number objects, depending on the source that had been selected.

For the command function, objects 2260h Relay Control Source and 2261h Relay Control Number are used. The default Control Source is setup to *CANopen® RPDO message* while the default Unlatch /Enable/Override Source are configured to *Source Not Used*.

For the unlatch function, objects 2270h Relay Unlatch Source and 2271h Relay Unlatch Number are used. When the relay polarity is set to *Latched Logic* or *Inverse Latched Logic*, unlatch function can be activated. When the polarity is set *Latched Logic*, the output state will be turned to OFF if the state of the Unlatch Source is high. If Unlatch Source state turns low afterwards, the output state will still be OFF in case the output state is high before. Only if the state of Control Source is changed back to low, the output state works like before in *Latched Logic*. The opposite behavior is valid for the *Inverse Latched Logic*. If the state of the Unlatch Source triggers and the output state is low, the output state will change to high. If the Unlatch Source state turns low afterwards, the output state will still be ON. Only if the state of the Control Source is changed back to high, the Output state works in *Inverse Latched Logic* mode again.

For the enable function, objects 2280h **Relay Enable Source** and 2281h **Relay Enable Number** are used. The Relay Enable Source is used to determine whether the relay output will be commanded by the Relay Control Source. There are six different options for the object 2282h **Relay Enable Response** in which the enable signal can be used. These responses are listed in Table 11, where the default value is highlighted.

Value	Meaning
0	***Enable When ON***
1	*Enable When OFF*
2	*Disable When ON*
3	*Disable When OFF*
4	*Enable When ON Else Keep State*
5	*Enable When OFF Else Keep State*

Table 11 – Relay Enable Response

When it is set to *Enable When ON* or *Disable When OFF***,** the relay output will be commanded according to the signal of the Control Source/Number and the relay polarity only when the signal of the Enable Source/Number is ON. Otherwise, the relay output is commanded to the OFF state (output type selected).

Similarly, when the Enable Response is set to *Enable When OFF* or *Disable When ON*, the relay output will be commanded according to the Control Source/Number and the relay polarity only when the signal of the Enable Source/ Number is OFF. Otherwise, the relay output is commanded to the OFF state (relay polarity selected).

In case the Enable Response is *Enable When ON Else Keep State*, the relay output will be commanded according to the signal of the Control Source/Number and the relay polarity only when the signal of the Enable Source/Number is ON. If the Enable Source is OFF, the relay output will keep the previous state.

Likewise, when the Enable Response is configured to *Enable When OFF Else Keep State*, the relay output will be commanded according to the Control Source/Number and the relay polarity only when the Enable Source/Number is OFF. Otherwise, the relay output holds the previous state.

Lastly, for the override function, objects 2290h **Relay Override Source** and 2291h **Relay Override Number** are used. Override Source has a higher priority than the Enable Source. There are two different options for object 2292h **Relay Override Response** in which the override signal can be used. These responses are listed in Table 12, where the default value is highlighted.

Value	Meaning
0	**Override When ON**
1	Override When OFF

Table 12 – Relay Override Responses

When the Override Response is configured to *Override When ON*, the relay output will be commanded when the signal of the Control Source/Number is OFF. If the Override Response is set to Override When OFF, the relay output will be commanded when the signal of the Control Source/Number is ON. Otherwise, the relay will be commanded by the object 2293h **Relay Override State**.

The object 2293h **Relay Override State** is used to configure the state when override is used. Table 6 shows the two options for it and the default value is highlighted. In case of *Override State OFF*, the relay output switches to Normally Open (state low) when override source is high and override response is selected as *Override When ON*. And if *Override State ON* is configured, the relay output changes to Normally closed (state high) when the override source is high and *Override When OFF* response is selected.

Value	Meaning
0	**Override State OFF**
1	Override State ON

Table 11 – Relay Override State

A time delay for both states (ON, OFF) can be configured by setting the object 22A0h **Enable Response Delay** to 1 (true). The values of these time delays can be set with the objects 22A1h **Turn ON Delay** and 22A2h **Turn OFF Delay**. In this case, the delays are valid for the enable state and the control state.

1.5. Lookup Table Function Block

The lookup table (LTz) function blocks are not used by default.

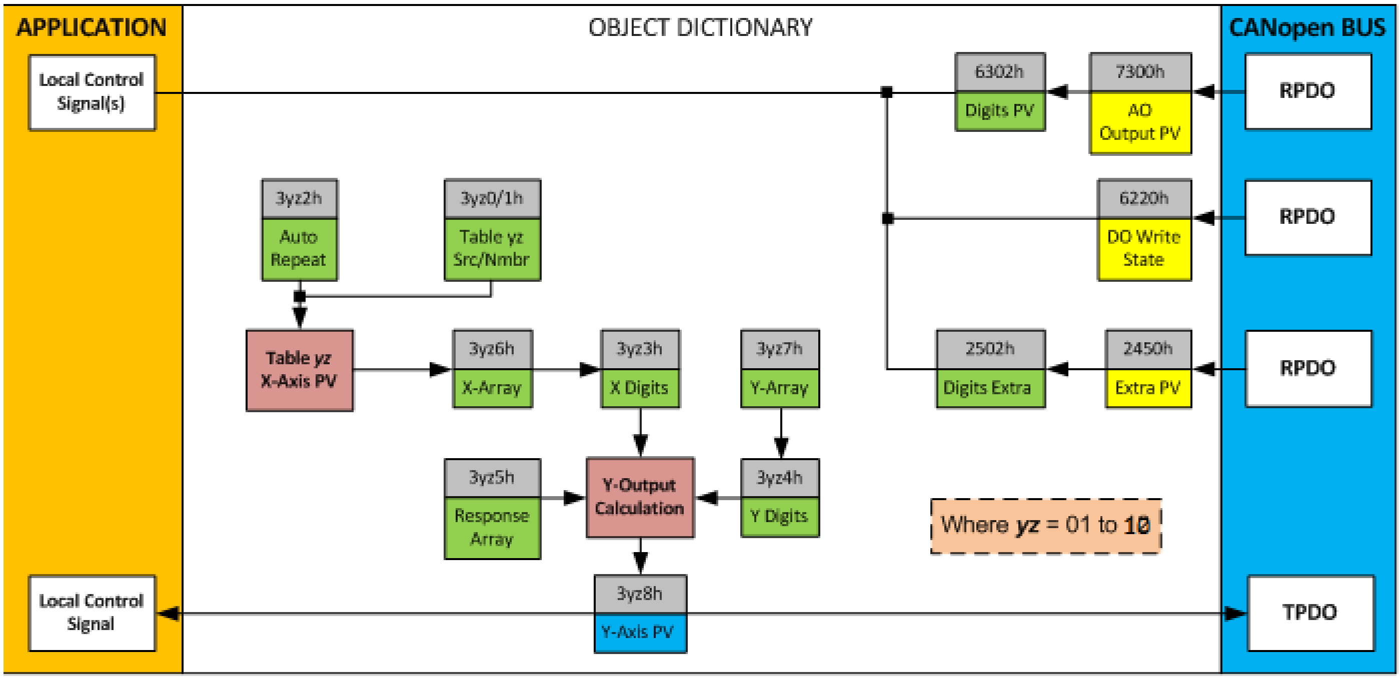


Figure 5 – Lookup Table Objects

Lookup tables are used to give an output response of up to 10 slopes per input. The array size of the objects 3yz5h **LTyz Point Response**, 3yz6h **LTyz Point X-Axis PV** and 3yz7h **Point Y-Axis PV** shown in the block diagram above is therefore 11.

Note: If more than 10 slopes are required, a Programmable Logic Block can be used to combine up to three tables to get 30 slopes, as is described in Section 1.6.

A parameter that will affect the function block is object **3yz5h sub-index 1** which defines the "**X-Axis Type**". By default, the tables have a *'Data Response'* output (0). Alternatively, it can be selected as a *'Time Response'* (1).

There are two (or three) other key parameters that will affect how this function block will behave depending on the "**X-Axis Type**" chosen. If chosen *'Data Response'*, then the objects 3yz0h **Lookup Table yz Input X-Axis Source** and 3yz1h **Lookup Table yz Input X-Axis Number** together define the control source for the function block. When it is changed, the table values in object 3yz6h need to be updated with new defaults based on the X-Axis source selected as described in Tables 4A and 4B. If however, the "**X-Axis Type**" is chosen to be *'Time Response'*, an additional parameter is taken into consideration - object 3yz2h, **Lookup Table yz Auto Repeat**. These will be described in more detail in Section 1.5.4.

1.5.1. X-Axis, Input Data Response

In the case where the "**X-Axis Type**" = *'Data Response',* the points on the X-Axis represents the data of the control source.

However, should the minimum input be less than zero, for example a resistive input that is reflecting temperature in the range of -40ºC to 210ºC, then the “**LTz Point X-Axis PV sub-index 1**” will be set to the minimum instead, in this case -40ºC.

The constraint on the X-Axis data is that the next index value is greater than or equal to the one below it, as shown in the equation below. Therefore, when adjusting the X-Axis data, it is recommended that X_{11} is changed first, then lower indexes in descending order.

MinInputRange <= X_1<= X_2<= X_3<= X_4<= X_5<= X_6<= X_7<= X_8<= X_9<= X_{10}<= X_{11}<= MaxInputRange

As stated earlier, MinInputRange and MaxInputRange will be determined by the scaling objects associated with X-Axis Source that has been selected, as outlined in Table 5.

1.5.2. Y-Axis, Lookup Table Output

By default, it is assumed that the output from the lookup table function block will be a percentage value in the range of 0 to 100.

In fact, so long as all the data in the Y-Axis is 0<=Y[i]<=100 (where i = 1 to 11) then other function blocks using the lookup table as a control source will have 0 and 100 as the Scaling 1 and Scaling 2 values used in linear calculations shown in Table 5.

However, the Y-Axis has no constraints on the data that it represents. This means that inverse, or increasing/decreasing or other responses can be easily established. **The Y-Axis does not have to be a percentage output but could represent full scale process values instead.**

In all cases, the controller looks at the **entire range** of the data in the Y-Axis sub-indexes, and selects the lowest value as the MinOutRange and the highest value as the MaxOutRange. So long as they are not both within the 0 to 100 range, they are passed directly to other function blocks as the limits on the lookup table output. (i.e. Scaling 1 and Scaling 2 values in linear calculations.)

Even if some of the data points are *‘Ignored’* as described in Section 1.5.3, they are still used in the Y-Axis range determination. If not all the data points are going to be used, it is recommended that Y10 be set to the minimum end of the range, and Y11 to the maximum first. This way, the user can get predictable results when using the table to drive another function block, such as an analog output.

1.5.3. Point to Point Response

By default, all six lookup tables have a simple linear response from 0 to 100 in steps of 10 for both the X and Y axes. For a smooth linear response, each point in the 30z5h **LTz Point Response** array is setup for a *‘Ramp To’* output.

Alternatively, the user could select a *‘Step To’* response for 30z4h, where N = 2 to 11. In this case, any input value between X_{N-1} to X_N will result in an output from the lookup table function block of Y_N. *(Recall: LTz Point Response sub-index 1 defines the X-Axis type)*

Figure 6 shows the difference between these two response profiles with the default settings.

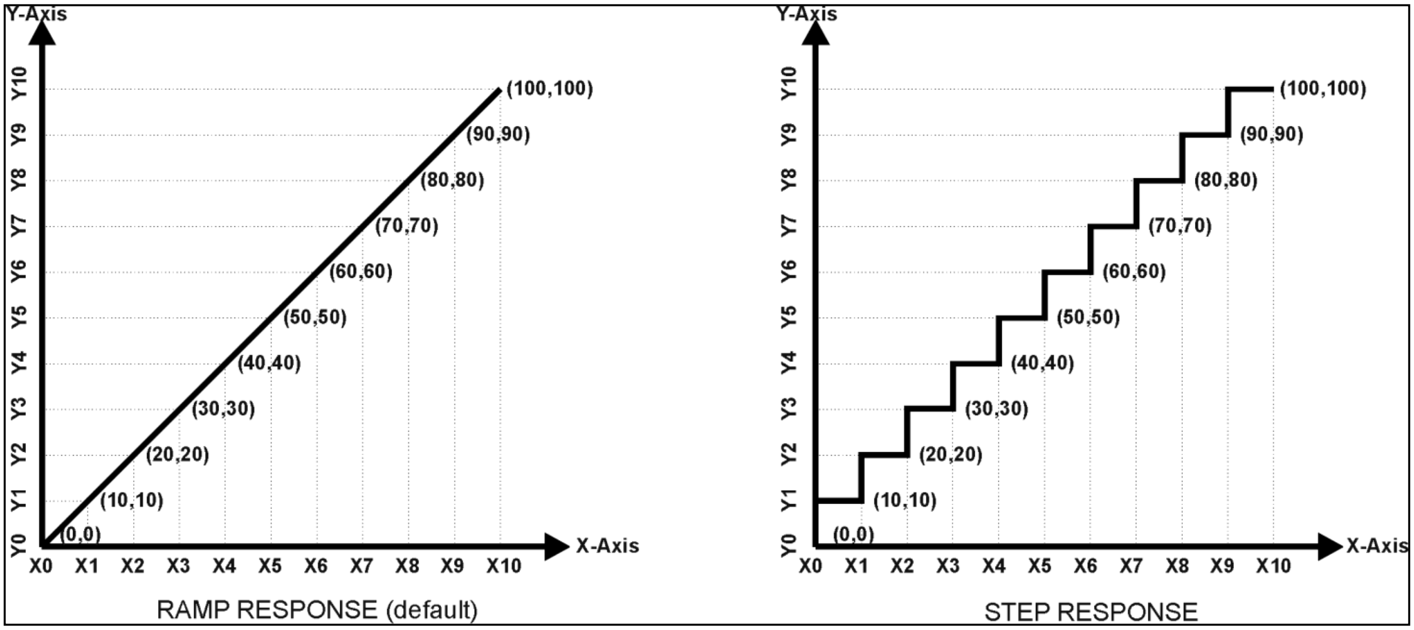


Figure 6 – Lookup Table Defaults with Ramp and Step Responses

Lastly, any point except (1,1) can be selected for an *'Ignore'* response. If **LTz Point Response sub-index N** is set to ignore, then all points from (X_N, Y_N) to (X_{11}, Y_{11}) will also be ignored. For all data greater than X_{N-1}, the output from the lookup table function block will be Y_{N-1}.

A combination of *'Ramp To'*, *'Jump To'* and *'Ignore'* responses can be used to create an application specific output profile. An example of where the same input is used as the X-Axis for two tables, but where the output profiles 'mirror' each other for a deadband joystick response is shown in Figure 7. The example shows a dual slope percentage output response for each side of the deadband, but additional slopes can be easily added as needed. *(Note: In this case, since the analog outputs are responding directly to the profile from the lookup tables, both would have object 2342h AO Control Response set to a 'Single Output Profile.')*

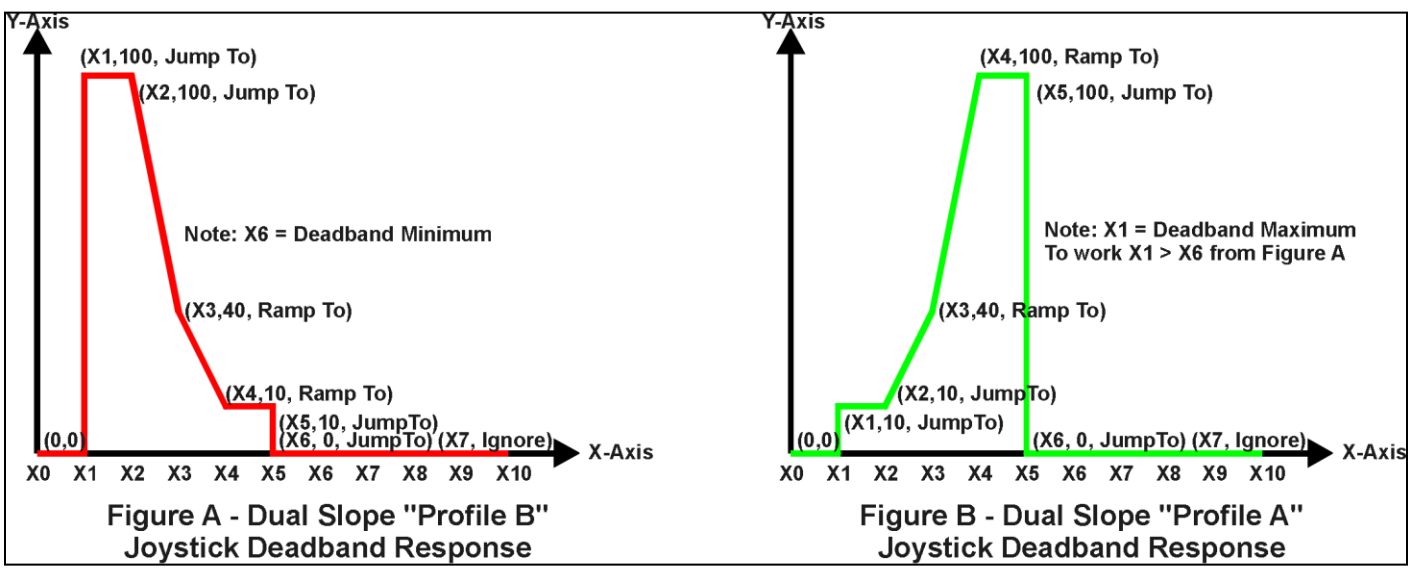


Figure 7 – Lookup Table Examples to Setup for Dual-Slope Joystick Deadband Response

To summarize, Table 14 outlines the different responses that can be selected for object 30z4h, both for the X-Axis type and for each point in the table.

Sub-Index	Value	Meaning
1	0	Data Response (X-Axis Type)
2 to 11		Ignore (this point and all following it)
1	1	Time Response (X-Axis Type)
2 to 11		Ramp To (this point)
1	2	N/A (not an allowed option)
2 to 11		Jump To (this point)

Table 14 – LTyz Point Response Options

1.5.4. X-Axis, Time Response

A lookup table can also be used to get a custom output response where the "**X-Axis Type**" is a *'Time Response'.* When this is selected, the X-Axis now represents time, in units of milliseconds, while the Y-Axis still represents the output of the function block.

With this response, the sequence will start depending on two parameters:

- **Lookup Table yz Input X-Axis Source** Object 3yz0h and;
- **Lookup Table yz Auto Repeat** Object 3yz2h

By default, the "Auto Repeat" object is set to FALSE (0). In this case, the lookup table will react in the following way:

> The X-Axis control source is treated as a digital input. When the control input is ON, the output will be changed over a period of time based on the profile in the lookup table. Once the profile has finished (i.e. reached index 11, or an 'Ignored' response), the output will remain at the last output at the end of the profile until the control input turns OFF.

However, when the "Auto Repeat" object is set to TRUE (1), the lookup table will react in the following way:

> When the control input is ON, the output will be changed over a period of time based on the profile in the lookup table. Once the profile has finished (i.e. reached index 11, or an 'Ignored' response), the lookup table will revert back to the first point in the table and Auto Repeat the sequence. This will continue for as long as the input remains ON. Once the input turns OFF, the lookup table sequence will stop and the output of the lookup table is zero.

Note: When the control input is OFF, the output is always at zero. When the input comes ON, the profile will **ALWAYS** start at position (X_1, Y_1) which is 0 output for 0ms.

When using the lookup table to drive an output based on **time**, it is mandatory that objects 2330h **Ramp Up** and 2331h **Ramp Down** in the analog output function block be set to **zero**. Otherwise, the output result will not match the profile as expected. Recall, also, that the AO scaling should be set to match the Y-Axis scaling of the table in order to get a 1:1 response of AO Output FV versus LTyz Output Y-Axis PV.

For a time response, the data in object 30z6h **LTyz Point X-Axis PV** is measured in milliseconds, and object 3yz3h **LTyz X-Axis Decimal Digits PV** is automatically set to 0. A minimum value of 1ms must be selected for all points other than sub-index 1 which is automatically set to [0,0]. The interval time between each point on the X-axis can be set anywhere from 1ms to 24 hours. [86,400,000 ms]

1.6. Programmable Logic Function Block

The programmable logic blocks (LB(x-3)) functions are not used by default.

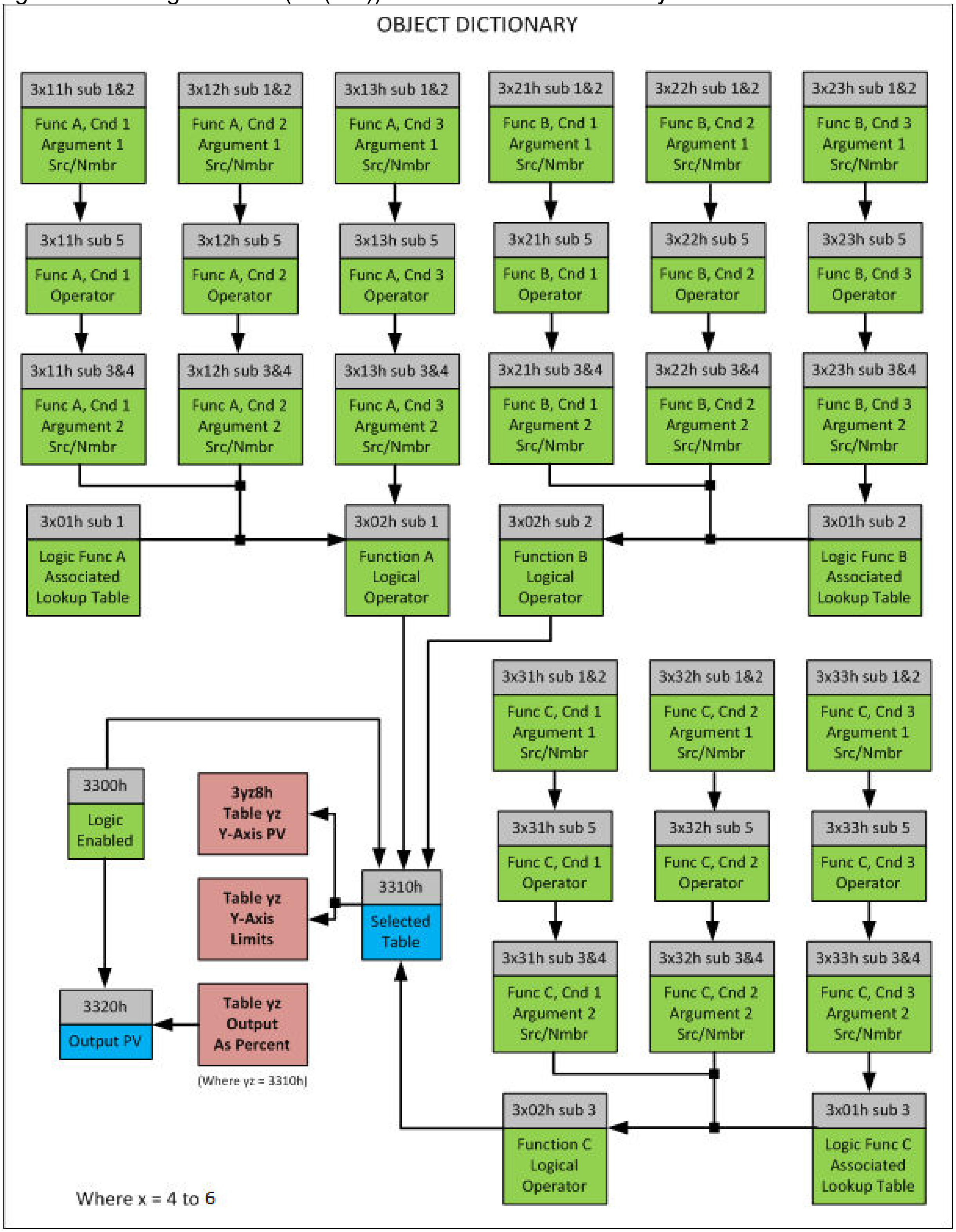


Figure 8 – Logic Block Objects

This function block is obviously the most complicated of them all, but very powerful. Any LB(x-3) (where X= 4 to 6) can be linked with up to three lookup tables, any one of which would be selected only under given conditions. Any three tables (of the available 10) can be associated with the logic, and which ones are used is fully configurable on object 3x01 **LB(x-3) Lookup Table Number**.

Should the conditions be such that a particular table (A, B or C) has been selected as described in Section 1.6.2, then the output from the selected table, at any given time, will be passed directly to LB(x-3)'s corresponding sub-index X in read-only mappable object 3320h **Logic Block Output PV**. The active table number can read from read-only object 3310h **Logic Block Selected Table**.

Note: In this document, the term *LB(x-3)* refers to Logic Blocks 1 to 3. Due to the CANopen® Object indices, Logic Block 1 begins at 3401h where x, in this case, is 4.

Therefore, an LB(x-3) allows up to three different responses to the same input, or three different responses to different inputs, to become the control for another function block, such as an analog output. Here, the "**Control Source**" for the reactive block would be selected to be the *'Programmable Logic Function Block,'* as described in Section 1.3.

In order to enable any one of logic blocks, the corresponding sub-index in object 3300h **Logic Block Enable** must be set to TRUE. They are all disabled by default.

Logic is evaluated in the order shown in Figure 9. Only if a lower indexed table (A, B, C) has not been selected will the conditions for the next table be looked at. **The default table is always selected as soon as it is evaluated. It is therefore required that the default table always be the highest index in any configuration.**

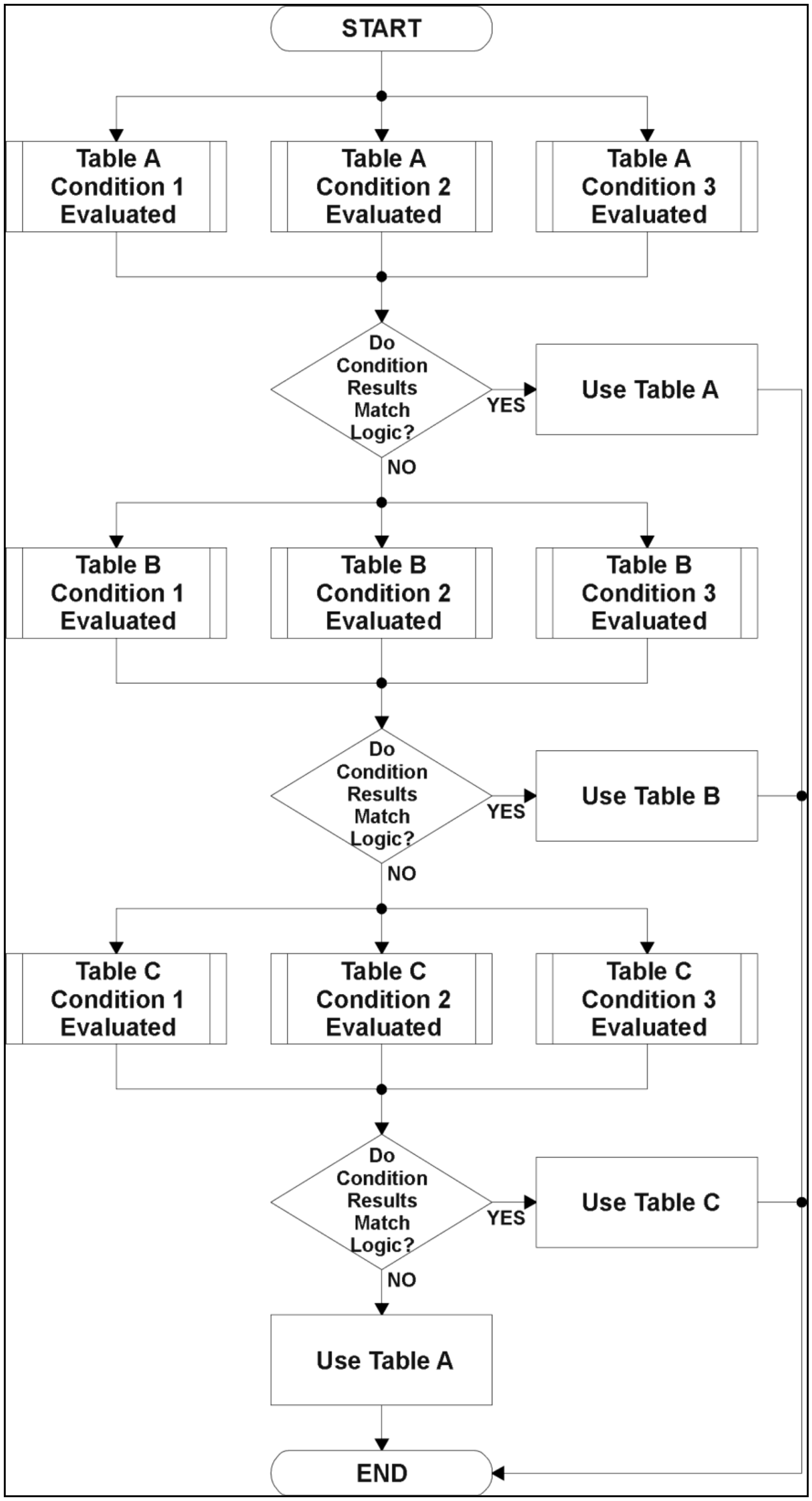


Figure 9 – Logic Block Flowchart

1.6.1. Conditions Evaluation

The first step in determining which table will be selected as the active table is to first evaluate the conditions associated with a given table. Each table has associated with it up to three conditions that can be evaluated. Conditional objects are custom DEFSTRUCT objects defined as shown in Table 15.

Index	Sub-Index	Name	Data Type
3xyz*	0	Highest sub-index supported	UNSIGNED8
	1	Argument 1 Source	UNSIGNED8
	2	Argument 1 Number	UNSIGNED8
	3	Argument 2 Source	UNSIGNED8
	4	Argument 2 Number	UNSIGNED8
	5	Operator	UNSIGNED8

** Logic Block X Function Y Condition Z, where X = 4 to 6, Y = A, B or C, and Z = 1 to 3*

Table 15 – LB(x-3) Condition Structure Definition

Objects 3x11h, 3x12h and 3x13h are the conditions evaluated for selecting Table A.
Objects 3x21h, 3x22h and 3x23h are the conditions evaluated for selecting Table B.
Objects 3x31h, 3x32h and 3x33h are the conditions evaluated for selecting Table C.

Argument 1 is always a logical output from another function block, as listed in Table 4A. As always, the input is a combination of the functional block objects 3xyzh sub-index 1 **"Argument 1 Source"** and **"Argument 1 Number."**

Argument 2 on the other hand, could either be another logical output such as with Argument 1, OR a constant value set by the user. To use a constant as the second argument in the operation, set **"Argument 2 Source"** to *'Constant Function Block',* and **"Argument 2 Number"** to the desired sub-index. When defining the constant, make sure it uses the same resolution (decimal digits) as the Argument 1 input.

Argument 1 is evaluated against Argument 2 based on the **"Operator"** selected in sub-index 5 of the condition object. The options for the operator are listed in Table 16, and the default value is always *'Equal'* for all condition objects.

Value	Meaning
0	**=, Equal**
1	!=, Not Equal
2	>, Greater Than
3	>=, Greater Than or Equal
4	<, Less Than
5	<=, Less Than or Equal

Table 16 – LB(x-3) Condition Operator Options

By default, both arguments are set to *'Control Source Not Used'* which disables the condition, and automatically results in a value of N/A as the result. Although it is generally considered that each condition will be evaluated as either TRUE or FALSE, the reality is that there could be four possible results, as described in Table 17.

Value	Meaning	Reason
0	False	(Argument 1) Operator (Argument 2) = False
1	True	(Argument 1) Operator (Argument 2) = True
2	Error	Argument 1 or 2 output was reported as being in an error state
3	Not Applicable	Argument 1 or 2 is not available (i.e. set to *'Control Source Not Used'*)

Table 17 – LB(x-3) Condition Evaluation Results

1.6.2. Table Selection

In order to determine if a particular table will be selected, logical operations are performed on the results of the conditions as determined by the logic in Section 1.6.1. There are several logical combinations that can be selected, as listed in Table 18. The default value for object 3x02h **LB(x-3) Function Logical Operator** is dependent on the sub-index. For sub-index 1 (Table A) and 2 (Table B), the *'Cnd1 And Cnd2 And Cnd3'* operator is used, whereas sub-index 3 (Table C) is setup as the *'Default Table''* response.

Value	Meaning
0	Default Table
1	Cnd1 And Cnd2 And Cnd3
2	Cnd1 Or Cnd2 Or Cnd3
3	(Cnd1 And Cnd2) Or Cnd3
4	(Cnd1 Or Cnd2) And Cnd3

Table 18 – LB(x-3) Function Logical Operator Options

Not every evaluation is going to need all three conditions. The case given in the earlier section, for example, only has one condition listed, i.e. that the Engine RPM be below a certain value. Therefore, it is important to understand how the logical operators would evaluate an Error or N/A result for a condition, as outlined in Table 19.

<table>
<tr><th>Logical Operator</th><th>Select Conditions Criteria</th></tr>
<tr><td>Default Table</td><td>Associated table is automatically selected as soon as it is evaluated.</td></tr>
<tr><td>Cnd1 And Cnd2 And Cnd3</td><td>Should be used when two or three conditions are relevant, and all must be True to select the table.

If any condition equals False or Error, the table is not selected.
An N/A is treated like a True.
If all three conditions are True (or N/A), the table is selected.

If((Cnd1==True) &&(Cnd2==True)&&(Cnd3==True)) Then Use Table</td></tr>
<tr><td>Cnd1 Or Cnd2 Or Cnd3</td><td>Should be used when only one condition is relevant. Can also be used with two or three relevant conditions.

If any condition is evaluated as True, the table is selected.
Error or N/A results are treated as False

If((Cnd1==True) || (Cnd2==True) || (Cnd3==True)) Then Use Table</td></tr>
<tr><td>(Cnd1 And Cnd2) Or Cnd3</td><td>To be used only when all three conditions are relevant.

If both Condition 1 and Condition 2 are True, OR Condition 3 is True, the table is selected. Error or N/A results are treated as False

If(((Cnd1==True)&&(Cnd2==True)) || (Cnd3==True)) Then Use Table</td></tr>
<tr><td>(Cnd1 Or Cnd2) And Cnd3</td><td>To be used only when all three conditions are relevant.

If Condition 1 And Condition 3 are True, OR Condition 2 And Condition 3 are True, the table is selected. Error or N/A results are treated as False

If(((Cnd1==True)||(Cnd2==True)) && (Cnd3==True)) Then Use Table</td></tr>
</table>

Table 19 – LB(x-3) Conditions Evaluation Based on Selected Logical Operator

If the result of the function logic is TRUE, then the associated lookup table (see object 4x01h) is immediately selected as the source for the logic output. No further conditions for other tables are evaluated. For this reason, the *'Default Table'* should always be setup as the highest letter table being used (A, B or C) If no default response has been setup, the Table A automatically becomes the default when no conditions are true for any table to be selected. This scenario should be avoided whenever possible so as to not result in unpredictable output responses.

The table number that has been selected as the output source is written to sub-index X of read-only object 4010h **Logic Block Selected Table.** This will change as different conditions result in different tables being used.

1.6.3. Logic Block Output

Recall that Table Y, where Y = A, B or C in the LB(x-3) function block does NOT mean lookup table 1 to 3. Each table has object 3x01h LB(x-3) **Lookup Table Number** which allows the user to select which lookup tables they want associated with a particular logic block. The default tables associated with each logic block are listed in Table 20.

Programmable Logic Block Number	Table A – Lookup Table Block Number	Table B – Lookup Table Block Number	Table C – Lookup Table Block Number
1	1	2	3
2	4	5	6
3	7	8	9

Table 20 – LB(x-3) Default Lookup Tables

If the associated Lookup Table YZ (where YZ equals 3310h sub-index X) does not have an **"X-Axis Source"** selected, then the output of LB(x-3) will always be "Not Available" so long as that table is selected. However, should LTyz be configured for a valid response to an input, be it Data or Time, the output of the LTyz function block (i.e. the Y-Axis data that has been selected based on the X-Axis value) will become the output of the LB(x-3) function block so long as that table is selected.

The LB(x-3) output is always setup as a percentage, based on the range of the Y-Axis for the associated table (see Section 1.5.2) It is written to sub-index X of read-only object 3320h **Logic Block Output PV** with a resolution of 1 decimal place.

1.7. Math Function Block

There are 4 mathematic function blocks that allow the user to define basic algorithms. Math function block Z = 1 to 4 will be enabled based on sub-index Z in object 4000h **Math Enable**.

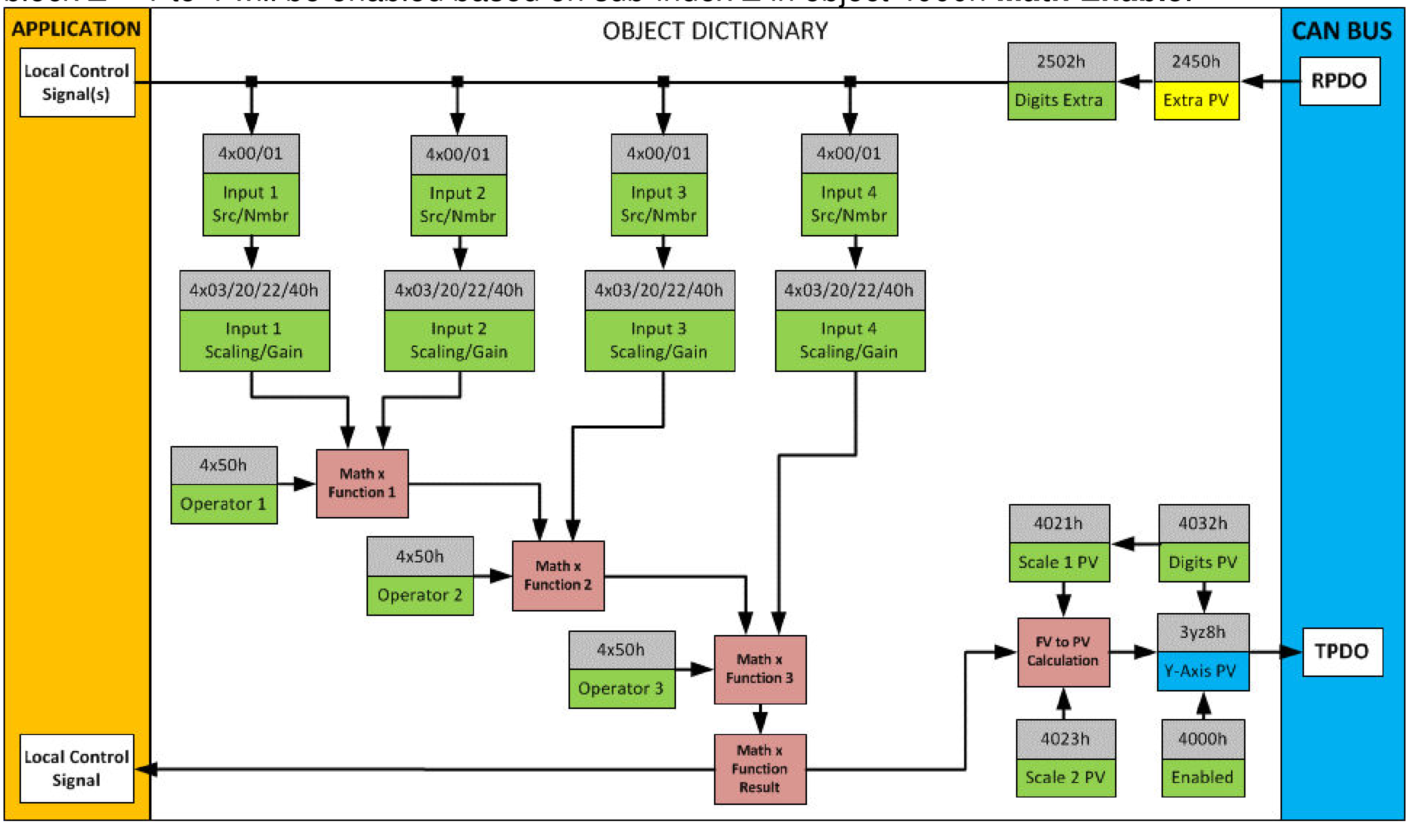


Figure 10 – Math Function Block Objects

A math function block can take up to four input signals, as listed in Table 4A in Section 1.3. Each input is then scaled according the associated scaling and gain objects. A "Math Input X" is determined by the corresponding sub-index X = 1 to 4 of the objects 4y00h **Math Y Input Source** and 4y01h **Math Y Input Number.** Here, y = 1 to 4; corresponding the Math 1- Math 4.

Inputs are converted into a percentage value based on objects 4y20h **Math Y Scaling 1 FV** and 4y22h **Math Y Scaling 2 FV**. Before being used in the calculation, these objects apply the resolution shift defined by object 4y02h **Math Y Decimal Digits FV**. As with any other function block using a control source for the X-Axis in a conversion, the scaling objects should be selected to match the values in the control's corresponding objects as per Table 5.

For additional flexibility, the user can also adjust object 4y40h **Math Y Input Gain.** This object has a fixed decimal digit resolution of 2, and a range of -100 to 100. By default, each input has a gain of 1.0.

For example, in the case where the user may want to combine two inputs such that a joystick (Input 1) is the primary control of an output, but the speed can be incremented or decremented based on a potentiometer (Input 2), it may be desired that 75% of the scale is controlled by the joystick position, while the potentiometer can increase or decrease the min/max output by up to 25%. In this case, Input 1 would have a gain of 0.75, while Input 2 uses 0.25. The resulting addition will give a command from 0 to 100% based on the combined positions of both inputs.

For each input pair, the appropriate arithmetic or logical operation is performed on the two inputs, InA and InB, according the associated function in sub-index of InB in object 4y50h **Math Y Operator**. The list of selectable function operations is defined in Table 21.

0	=	True when InA Equals InB
1	!=	True when InA Not Equal InB
2	>	True when InA Greater Than InB
3	>=	True when InA Greater Than or Equal InB
4	<	True when InA Less Than InB
5	<=	True when InA Less Than or Equal InB
6	OR	True when InA or InB is True
7	NOR	True when InA and InB are False
8	AND	True when InA and InB are True
9	NAND	True when InA and InB are not both True
10	XOR	True when InA/InB is True, but not both
11	XNOR	True when InA and InB are both True or False
12	+	Result = InA plus InB
13	-	Result = InA minus InB
14	x	Result = InA times InB
15	/	Result = InA divided by InB
16	MIN	Result = Smallest of InA and InB
17	MAX	Result = Largest of InA and InB

Table 21 – Object 4y50h Math Function Operators

For Function 1, InA and InB are Math Inputs 1 and 2, respectively.
For Function 2, InA is the result of Function 2 and InB is Math Input 3, respectively.
For Function 3, InA is the result of Function 3 and InB is Math Input 4, respectively.

Exclusively **within** a Math Block, there is a third control parameter: Object 4y02h, **Math Y Function Number.** This parameter allows for the result of any Function (1, 2 or 3) to be the input to any **Math Input Y** within the same Math Block. Therefore, **Math Y Input Source** must be a Math Block and **Math Y Input Number** must be the same number as being configured. When these four parameters match, if **Math Y Function Number** is set to 1, 2, or 3, the respective input will be the result of the Function selected. By default, **Math Y Function Number** is set to 0 – in which case this parameter is ignored and uses the Math Block output result. These functions can only be used **within** the Math Block. They can not be used for other Math Blocks or logic blocks.

This allows for more versatility within the Math Block. For a valid result in each Function, both inputs must be non-zero value (other than *'Control Source Not Used'*). Otherwise, the corresponding Function is ignored. Furthermore, for a valid/expected output result in each Math Block, it is necessary to keep in mind how the Functions link to one another within the Math Block. As an example, consider all 6 inputs to be CANopen® Messages 1 to 6 (thus using all 3 Functions). Since all 3 functions are used but Function 3 has no relation to Function 1 or 2, the result of the Math Block will be the result of Function 3, thus, ignoring Functions 1 and 2.

For logical operators (6 to 11), any SCALED input greater than or equal to 0.5 is treated as a TRUE input. For logic output operators (0 to 11), the result of the calculation for the function will always be 0 (FALSE) or 1 (TRUE).

Error data (i.e. input measured out of range) is always treated as a 0.0 input into the function.

For the arithmetic functions (12 to 17), it is recommended to scale the data such that the resulting operation will not exceed full scale (0 to 100%) and saturate the output result.

When dividing, a zero InB value will always result is a zero output value for the associated function. When subtracting, a negative result will always be treated as a zero, unless the function is multiplied by a negative one, or the inputs are scaled with a negative coefficient first.

The resulting final mathematical output calculation is in the appropriate physical units using object 4021h **Math Output Scaling 1 PV** and 4023h **Math Output Scaling 2 PV**. These objects are also considered the Min and Max values of the Math Block output and apply the resolution shift defined by object 4032h **Math Output Decimal Digits PV**. The result is written to read-only object 4030h **Math Output PV**. These scaling objects should also be taken into account when the Math Function is selected as the input source for another function block, as outlined in Table 5.

1.8. Miscellaneous Function Block

There are some other objects available which have not yet been discussed or mentioned briefly in passing (i.e. constants.) These objects are not necessarily associated with one another but are all discussed here.

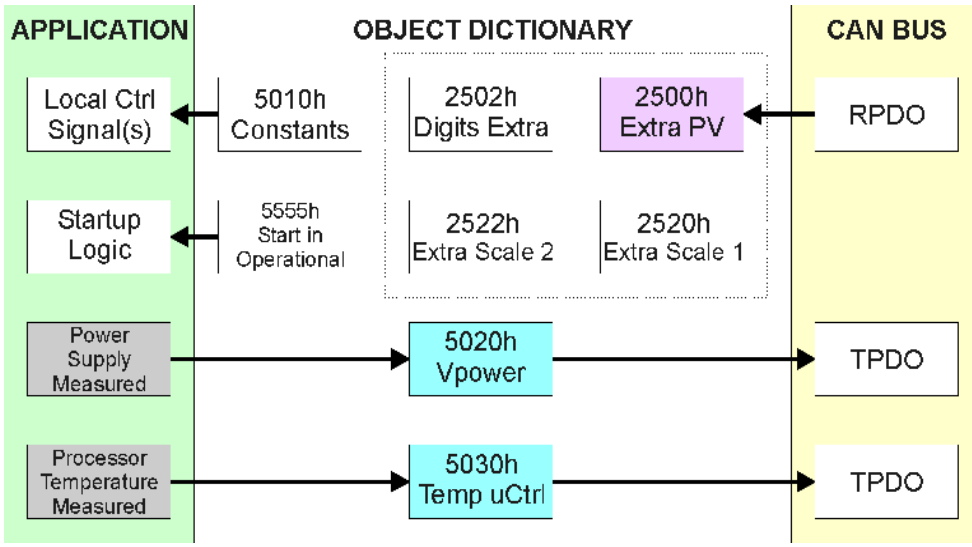


Figure 11 – Miscellaneous Objects

Objects 2500h **Extra Control Received PV**, 2502h **EC Decimal Digits PV**, 2520h **EC Scaling 1 PV** and 2522h **EC Scaling 2 PV** have been mentioned in Section 1.3, Table 5. These objects allow for additional data received on a CANopen® RPDO to be mapped independently to various function blocks as a control source. For example, a lookup table can use signal from CAN bus. The scaling objects are provided to define the limits of the data when it is used by another function block, as shown in Table 5.

Power Supply and Processor Temperature

Objects 5020h **Power Supply FV** and 5030h **Processor Temperature FV** are available as read-only feedback for additional diagnostics.

Object 5021h is used to enable or disable Power Supply faults. If object 5021h is enabled (1), then objects 5022h, **Under Voltage Threshold**, and 5023h, **Over Voltage Threshold** are used by the controller. These two configurable objects hold the desired values to trigger a supply fault. Object 5024h, **Hysteresis To Clear Fault**, is the number of Volts required for the supply to increase above or decrease below from the voltage entered in objects 5022h and 5023h, respectively.

Similarly, object 5031h enables or disables faults by the Processor Temperature measured. An **Over Temperature value** can be entered in object 5032h which, when reached and exceeded, will trigger a temperature fault. The temperature will need to decrease a value of **Hysteresis to Clear Temperature Fault,** object 5033h, below the temperature set in object 5032h.

Both diagnostics, Power Supply and Temperature, have an object to disable all outputs (including relay). In case all outputs should be disabled when an overvoltage or undervoltage is measured, the object 5025h **Power Fault Shutdown Outputs** must be set to true. Whereas the object 5034h **Over Temperature Shutdown** disables the outputs when the unit is overheating.

Automatic Update of Objects

Object 5500h Enable Automatic Updates allows for the controller to automatically update the objects related to the output to defaults when it is changed. By default, this object is set to TRUE, in which case the objects are set to their default values depending on the type selected.

On the other hand, when this object is FALSE, the objects are not set to defaults and are left with the same values previous to changing the type. In this case, these are to be configured manually.

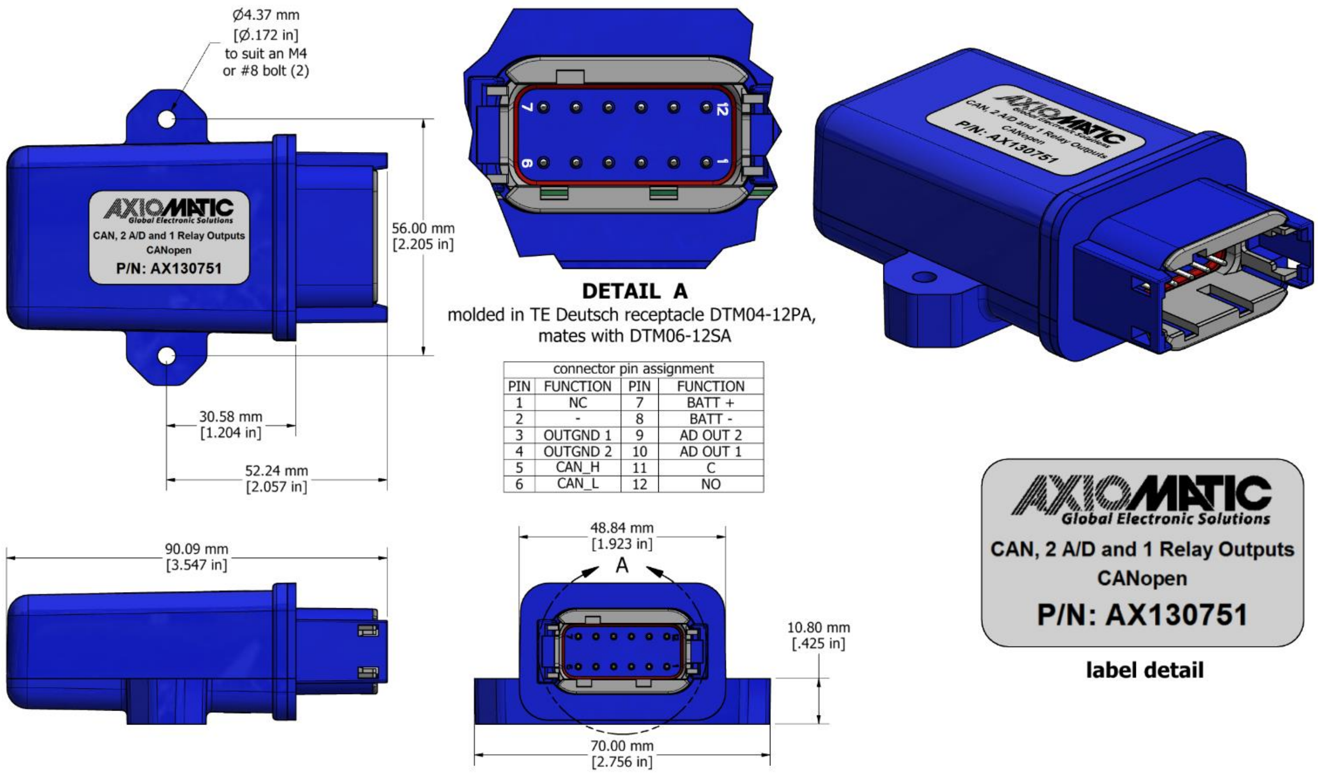
Start in Operational Mode

The last object 5555h **Start in Operational** is provided as a ‘cheat’ when the unit is not intended to work with a CANopen® network (i.e. a stand-alone control) or is working on a network comprised solely as slaves so the OPERATION command will never be received from a master. By default, this object is disabled (FALSE).

When using the unit as a stand-alone controller where 5555h is set to *TRUE*, it is recommended to disable all TPDOs (set the Event Timer to zero) so that it does not run with a continuous CAN error when not connected to a bus.

2. INSTALLATION INSTRUCTIONS

2.1. AX130751 Dimensions and Pinout



AX130751 Dimension Drawing.

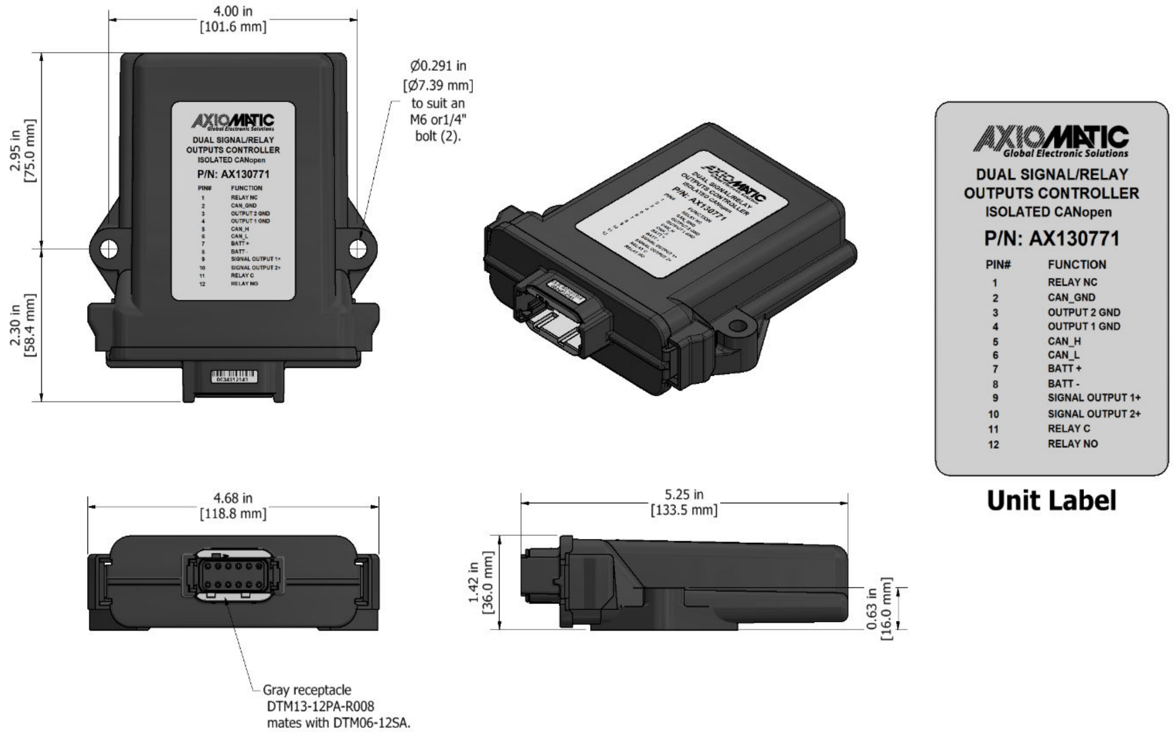
CANopen® is a registered community trade mark of CAN in Automation e.V.

Integral 12-pin receptacle (equivalent to TE Deutsch P/N: DTM04-12PA)

CAN and I/O Connector	
Pin #	**Description**
1	Relay Output (NC)
2	Not Used
3	Output 1 GND
4	Output 2 GND
5	CAN_H
6	CAN_L
7	BATT +
8	BATT-
9	A/D Output 2
10	A/D Output 1
11	Relay Output (C)
12	Relay Output (NO)

AX130751 Connector Pinout

2.2. AX130771 Dimensions and Pinout



Unit Label

CAN and I/O Connector	
Pin #	**Description**
1	Relay Output (NC)
2	CAN_GND
3	Output 2 GND
4	Output 1 GND
5	CAN_H
6	CAN_L
7	BATT +
8	BATT-
9	A/D Output 1
10	A/D Output 2
11	Relay Output (C)
12	Relay Output (NO)

AX130771 Connector Pinout

3. CANOPEN® OBJECT DICTIONARY

The CANopen® object dictionary of the CAN-2AOUT-1RLYOUT Controller is based on CiA device profile DS-404 V1.2 (device profile for Closed Loop Controllers). The object dictionary includes Communication Objects beyond the minimum requirements in the profile, as well as several manufacturer-specific objects for extended functionality.

3.1. NODE ID and BAUDRATE

By default, the CAN-2AOUT-1RLYOUT Controller ships factory programmed with a Node ID = 127 (0x7F) and with Baudrate = 125 kbps.

3.1.1. LSS Protocol to Update

The only means by which the Node-ID and Baudrate can be changed is to use Layer Settling Services (LSS) and protocols as defined by CANopen® standard DS-305.

Follow the steps below to configure either variable using LSS protocol. If required, please refer to the standard for more detailed information about how to use the protocol.

3.1.2. Setting Node-ID

- Set the module state to LSS-configuration by **sending** the following message:

Item	***Value***
COB-ID	0x7E5
Length	2
Data 0	0x04 (cs=4 for switch state global)
Data 1	0x01 (switches to configuration state)

- Set the Node-ID by **sending** the following message:

Item	***Value***
COB-ID	0x7E5
Length	2
Data 0	0x11 (cs=17 for configure node-id)
Data 1	Node-ID (set new Node-ID as a hexadecimal number)

- The module will send the following response (any other response is a failure):

Item	***Value***
COB-ID	0x7E4
Length	3
Data 0	0x11 (cs=17 for configure node-id)
Data 1	0x00
Data 2	0x00

- Save the configuration by **sending** the following message:

Item	***Value***
COB-ID	0x7E5
Length	1
Data 0	0x17 (cs=23 for store configuration)

- The module will send the following response (any other response is a failure):

Item	***Value***
COB-ID	0x7E4
Length	3
Data 0	0x17 (cs=23 for store configuration)
Data 1	0x00
Data 2	0x00

- Set the module state to LSS-operation by **sending** the following message:
  (Note, the module will reset itself back to the pre-operational state)

Item	***Value***
COB-ID	0x7E5
Length	2
Data 0	0x04 (cs=4 for switch state global)
Data 1	0x00 (switches to waiting state)

3.1.3. Setting Baudrate

- Set the module state to LSS-configuration by **sending** the following message:

Item	***Value***
COB-ID	0x7E5
Length	2
Data 0	0x04 (cs=4 for switch state global)
Data 1	0x01 (switches to configuration state)

- Set the baudrate by **sending** the following message:

Item	***Value***
COB-ID	0x7E5
Length	3
Data 0	0x13 (cs=19 for configure bit timing parameters)
Data 1	0x00 (switches to waiting state)
Data 2	Index (select baudrate index per table 22)

Index	***Bit Rate***
0	1 Mbit/s
1	666 kbit/s
2	500 kbit/s
3	250 kbit/s
4	125 kbit/s (default)
5	reserved (100 kbit/s)
6	50 kbit/s
7	20 kbit/s
8	10 kbit/s

Table 22 – LSS Baudrate Indices

- The module will send the following response (any other response is a failure):

Item	***Value***
COB-ID	0x7E4
Length	3
Data 0	0x13 (cs=19 for configure bit timing parameters)
Data 1	0x00
Data 2	0x00

- Activate bit timing parameters by **sending** the following message:

Item	***Value***
COB-ID	0x7E5
Length	3
Data 0	0x15 (cs=19 for activate bit timing parameters)
Data 1	<delay_lsb>
Data 2	<delay_msb>

The delay individually defines the duration of the two periods of time to wait until the bit timing parameters switch is done (first period) and before transmitting any CAN message with the new bit timing parameters after performing the switch (second period). The time unit of switch delay is 1 ms.

- Save the configuration by **sending** the following message (on the NEW baudrate):

Item	***Value***
COB-ID	0x7E5
Length	1
Data 0	0x17 (cs=23 for store configuration)

- The module will send the following response (any other response is a failure):

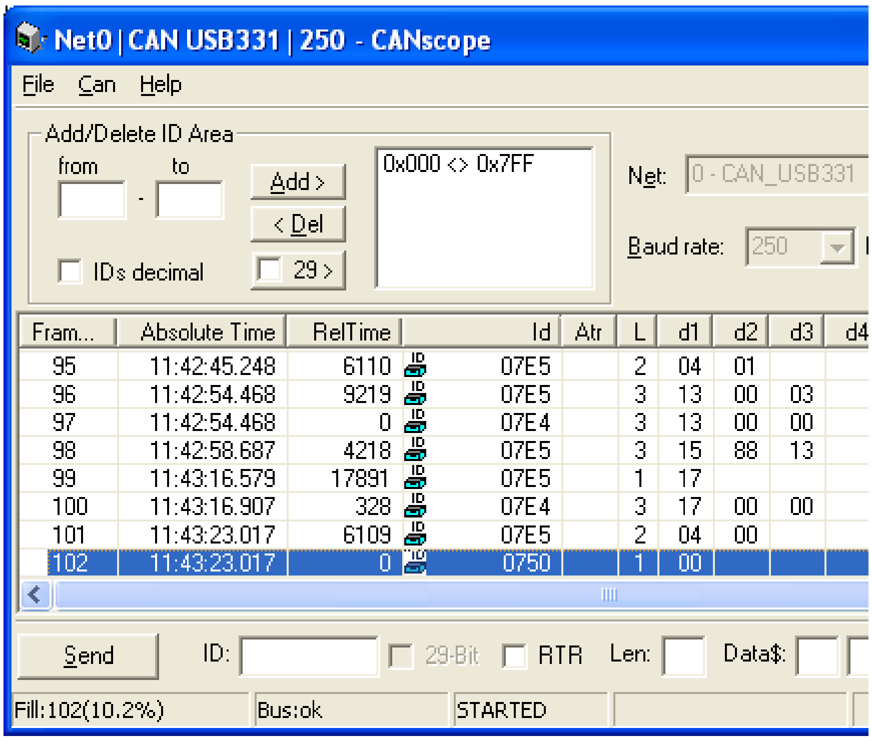
Item	***Value***
COB-ID	0x7E4
Length	3
Data 0	0x17 (cs=23 for store configuration)
Data 1	0x00
Data 2	0x00

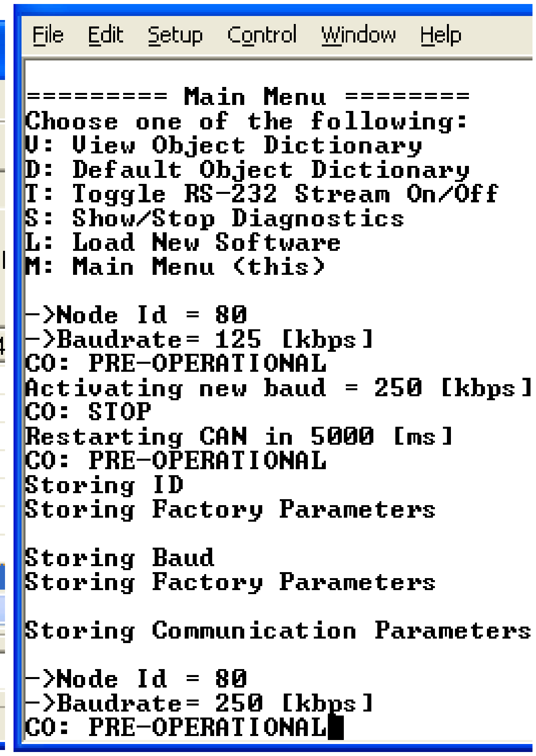
- Set the module state to LSS-operation by **sending** the following message:
(Note, the module will reset itself back to the pre-operational state)

Item	***Value***
COB-ID	0x7E5
Length	2
Data 0	0x04 (cs=4 for switch state global)
Data 1	0x00 (switches to waiting state)

The following screen capture (left) shows the CAN data was sent (7E5h) and received (7E4h) by the tool when the baudrate was changed to 250 kbps using the LSS protocol. The other image (right) shows what was printed on an example debug RS-232 menu while the operation took place.

Between CAN Frame 98 and 99, the baudrate on the CAN Scope tool was changed from 125 to 250 kbps.





3.2. COMMUNICATION OBJECTS (DS-301 and DS-404)

The communication objects supported by the CAN-2AOUT-1RLYOUT Controller 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-301 and 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
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 Behaviour	ARRAY	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		RO	No
1601	RPDO2 Mapping Parameter	RECORD		RO	No
1602	RPDO3 Mapping Parameter	RECORD		RO	No
1603	RPDO4 Mapping Parameter	RECORD		RO	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

3.2.1. Object 1000h: Device Type

This object contains information about the device type as per device profile DS-404. The 32-bit parameter is divided into two 16-bit values, showing General and Additional information as shown below.

MSB	LSB
Additional Information = 0xE01C	**General Information** = 0x0194 (404)

DS-404 defines the Additional Information field in the following manner:
0000h = reserved
0001h = digital input block
0002h = analog input block
0004h = digital output block
0008h = analog output block
0010h = controller block (aka PID)
0020h = alarm block
0040h … 0800h = reserved
1000h = reserved
2000h = lookup table block (manufacturer-specific)
4000h = programmable logic block (manufacturer-specific)
8000h = miscellaneous block (manufacturer-specific)

Object Description

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

Entry Description

Access	RO
PDO Mapping	No
Value Range	0xE01C0194
Default Value	0xE01C0194

3.2.2. Object 1001h: Error Register

This object is an error register for the device. Any time there is an error detected by the CAN-2AOUT-1RLYOUT Controller, the Generic Error Bit (bit 0) is set. Only if there are no errors in the module will this bit be cleared. No other bits in this register are used by the CAN-2AOUT-1RLYOUT Controller.

Object Description

Index	1001h
Name	Error Register
Object Type	VAR
Data Type	UNSIGNED8

Entry Description

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

3.2.3. Object 1002h: Manufacturer Status Register

This object is used for manufacturer debug purposes.

3.2.4. Object 1003h: Pre-Defined Error Field

This object 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 sub-index 1, with sub-index 0 containing the number of errors currently in the list. When the device is in an error-free state, the value of sub-index 0 is zero.

The error list may be cleared by writing a zero to sub-index 0, which will clear all errors from the list, regardless of whether they are still present or not. Clearing the list does NOT mean that the module will return to the error-free behavior state if at least one error is still active.

The CAN-2AOUT-1RLYOUT Controller has a limitation of a maximum of 4 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 on which the error occurred.

MSB LSB

Error Description	Channel-ID	EMCY Error Code

If node-guarding is used (not recommended per the latest standard) and a lifeguard event occurs, the manufacturer-specific field will be set to 0x1000. On the other hand, if a heartbeat consumer fails to be received within the expected timeframe, the Error Description will be set to 0x80 and the Channel-ID (nn) will reflect the Node-ID of the consumer channel that was not producing. In this case, the manufacturer-specific field will therefore be 0x80nn. In both cases, the corresponding EMCY Error Code will be the Guard Error 0x8130.

If an RPDO is not received within the expected “Event Timer” period, an RPDO timeout will be flagged. Table 23 outlines the resulting Error Field Codes and their meanings.

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

Table 23 – Pre-Defined Error Field Codes

Object Description

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

Entry Description

Sub-Index	0h
Description	Number of entries
Access	RW
PDO Mapping	No
Value Range	0 to 4
Default Value	0

Sub-Index	1h to 4
Description	Standard error field
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	0

3.2.5. Object 100Ch: Guard Time

The objects at index 100Ch and 100Dh shall indicate the configured guard time respective to the lifetime factor. The lifetime factor multiplied with the guard time gives the lifetime 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 the 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

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0 to 65535
Default Value	0

3.2.6. Object 100Dh: Lifetime Factor

The lifetime 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	Life time factor
Object Type	VAR
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0 to 255
Default Value	0

3.2.7. 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 sub-index. 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 sub-index, the CAN-2AOUT-1RLYOUT Controller will store the parameters in non-volatile memory, and then confirm the SDO transmission.

By read access, the object provides information about the module’s saving capabilities. For all sub-indexes, this value is 1h, indicating that the CAN-2AOUT-1RLYOUT Controller saves parameters on command. **This means that if power is removed before the Store object is written, changes to the Object Dictionary will NOT have been saved in the non-volatile memory and will be lost on the next power cycle.**

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h
Description	Save all parameters
Access	RW
PDO Mapping	No
Value Range	0x65766173 (write access) 1h (read access)
Default Value	1h

Sub-Index	2h
Description	Save communication parameters
Access	RW
PDO Mapping	No
Value Range	0x65766173 (write access) 1h (read access)
Default Value	1h

Sub-Index	3h
Description	Save application parameters
Access	RW
PDO Mapping	No
Value Range	0x65766173 (write access) 1h (read access)
Default Value	1h

Sub-Index	4h
Description	Save manufacturer parameters
Access	RW
PDO Mapping	No
Value Range	0x65766173 (write access) 1h (read access)
Default Value	1h

3.2.8. Object 1011h: Restore 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 sub-index. 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 sub-index, the CAN-2AOUT-1RLYOUT Controller will restore the defaults in non-volatile memory, and then confirm the SDO transmission. **The default values are set valid only after the device is reset or power-cycled.** This means that the CAN-2AOUT-1RLYOUT Controller will NOT start using the default values right away, but rather continue to run from whatever values were in the Object Dictionary prior to the restore operation.

By read access, the object provides information about the module's default parameter restoring capabilities. For all sub-indexes, this value is 1h, indicating that the CAN-2AOUT-1RLYOUT Controller restores defaults on command.

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h
Description	Restore all default parameters
Access	RW
PDO Mapping	No
Value Range	0x64616F6C (write access), 1h (read access)
Default Value	1h

Sub-Index	2h
Description	Restore default communication parameters
Access	RW
PDO Mapping	No
Value Range	0x64616F6C (write access), 1h (read access)
Default Value	1h

Sub-Index	3h
Description	Restore default application parameters
Access	RW
PDO Mapping	No
Value Range	0x64616F6C (write access), 1h (read access)
Default Value	1h

Sub-Index	4h
Description	Restore default manufacturer parameters
Access	RW
PDO Mapping	No
Value Range	0x64616F6C (write access), 1h (read access)
Default Value	1h

3.2.9. Object 1016h: Consumer Heartbeat Time

The CAN-2AOUT-1RLYOUT 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 CAN-2AOUT-1RLYOUT 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

Sub-Index	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

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

3.2.10. Object 1017h: Producer Heartbeat Time

The CAN-2AOUT-1RLYOUT 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

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	10 to 65535
Default Value	0

3.2.11. Object 1018h: Identity Object

The identity object indicates the data of the CAN-2AOUT-1RLYOUT Controller, including vendor id, device id, software and hardware version numbers, and the serial number.

In the Revision Number entry at sub-index 3, the format of the data is as shown below

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

Object Description

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

Entry Description

Sub-Index	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

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

Sub-Index	2h
Description	Product Code
Access	RO
PDO Mapping	No
Value Range	0xAA130751
Default Value	0xAA130751

Sub-Index	3h
Description	Revision Number
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	0x00010000

Sub-Index	4h
Description	Serial Number
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	No

3.2.12. Object 1020h: Verify Configuration

This object can be read to see what date the software (version identified in object 1018h) was compiled. The date is represented as a hexadecimal value showing day/month/year as per the format below. The time value at sub-index 2 is a hexadecimal value showing the time in a 24-hour clock

MSB LSB

Day (in 1-Byte Hex)	Month (in 1-Byte Hex)	Year (in 2-Byte Hex)
00	00	Time (in 2-Byte Hex)

For example, a value of 0x10082010 would indicate that the software was compiled on August 10th, 2010. A time value of 0x00001620 would indicate it was compiled at 4:20pm.

Object Description

Index	1020h
Name	Verify configuration
Object Type	ARRAY
Data Type	UNSIGNED32

Entry Description

Sub-Index	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h
Description	Configuration date
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	No

Sub-Index	2h
Description	Configuration time
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	No

3.2.13. Object 1029h: Error Behaviour

This object controls the state that the CAN-2AOUT-1RLYOUT Controller will be set into in case of an error of the type associated with the sub-index.

A network fault is flagged when an RPDO is not received within the expected time period defined in the "Event Timer" of the associated communication objects, (see Section 3.2.14 for more information) or if a lifeguard or heartbeat message is not received as expected.

For all sub-indexes, the following definitions hold true:

0 = Pre-Operational (node reverts to a pre-operational state when this fault is detected)
1 = No State Change (node remains in the same state it was in when the fault occurred)
2 = Stopped (node goes into stopped mode when the fault occurs)

Object Description

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

Entry Description

Sub-Index	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Sub-Index	1h
Description	Communication Fault
Access	RW
PDO Mapping	No
Value Range	See above
Default Value	1 (No State Change)

Sub-Index	2h
Description	Digital Input Fault (not used)
Access	RW
PDO Mapping	No
Value Range	See above
Default Value	1 (No State Change)

Sub-Index	3h
Description	Analog Input Fault (not used)
Access	RW
PDO Mapping	No
Value Range	See above
Default Value	1 (No State Change)

Sub-Index	4h
Description	Digital Output Fault (not used)
Access	RW
PDO Mapping	No
Value Range	See above
Default Value	1 (No State Change)

Sub-Index	5h
Description	Analog Output Fault (not used)
Access	RW
PDO Mapping	No
Value Range	See above
Default Value	1 (No State Change)

3.2.14. RPDO Behaviour

Per the CANopen® standard DS-301, the following procedure shall be used for re-mapping, and is the same for both RPDOs and TPDOs.

a) Destroy the PDO by setting bit **exists** (most significant bit) of sub-index 01h of the according PDO communication parameter to 1b

b) Disable mapping by setting sub-index 00h of the corresponding mapping object to 0

c) Modify the mapping by changing the values of the corresponding sub-indices

d) Enable mapping by setting sub-index 00h to the number of mapped objects

e) Create the PDO by setting bit **exists** (most significant bit) of sub-index 01h of the according PDO communication parameter to 0b

The CAN-2AOUT-1RLYOUT Controller can support up to four RPDO messages. All RPDOs on the CAN-2AOUT-1RLYOUT Controller use the similar default communication parameters, with the PDO IDs set according to the pre-defined connection set described in DS-301. Most RPDOs do not exist, there is no RTR allowed, they use 11-bit CAN-IDs (base frame valid) and they are all event-driven. While all four have valid default mappings defined (see below) only RPDO1 is enabled by default (i.e. RPDO exists).

RPDO1 Mapping at Object 1600h: Default ID 0x200 + Node ID

Sub-Index	***Value***	***Object***
0	3	Number of mapped application objects in PDO
1	0x73000110	Analog Output 1 Process Value
2	0x73000210	Analog Output 2 Process Value
3	0x62200308	Relay Write State
4	0	Not used by default

RTPDO2 Mapping at Object 1601h: Default ID 0x300 + Node ID

Sub-Index	***Value***	***Object***
0	1	Number of mapped application objects in PDO
1	0x25000110	Extra Received 1 PV (i.e. Lookup Table 1 X-Axis)
2	0x25000210	Extra Received 2 PV (i.e. Lookup Table 2 X-Axis)
3	0x25000310	Extra Received 3 PV (i.e. Lookup Table 3 X-Axis)
4	0	Not used by default

RPDO3 Mapping at Object 1602h: Default ID 0x400 + Node ID

Sub-Index	***Value***	***Object***
0	0	Number of mapped application objects in PDO
1	0x25000410	Extra Received 4 PV (i.e. Lookup Table 4 X-Axis)
2	0x25000510	Extra Received 5 PV (i.e. Lookup Table 5 X-Axis)
3	0x25000610	Extra Received 6 PV (i.e. Lookup Table 6 X-Axis)
4	0	Not used by default

RPDO4 Mapping at Object 1603h: Default ID 0x500 + Node ID

Sub-Index	Value	Object
0	2	Number of mapped application objects in PDO
1	0x62200108	Digital Output 1 Write State Output Line
2	0x62200208	Digital Output 2 Write State Output Line
3	0	Not used by default
4	0	Not used by default

None of them have the timeout feature enabled, i.e. the “Event Timer” on sub-index 5 is set to zero. When this is changed to a non-zero value, if the RPDO has not been received from another node within the time period defined (while in Operational mode), a network fault is activated, and the controller will go to the operational state define in Object 1029h sub-index 4.

Object Description

Index	1400h to 1403h
Name	RPDO communication parameter
Object Type	RECORD
Data Type	PDO Communication Record

Entry Description

Sub-Index	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Sub-Index	1h
Description	COB-ID used by RPDO
Access	RW
PDO Mapping	No
Value Range	See value definition in DS-301
Default Value	40000000h + RPDO1 + Node ID 40000000h + RPDO4 + Node ID C0000000h + RPDOx + Node-ID

X	***RPDOx ID***
1	0200h
2	0300h
3	0400h
4	0500h

Node-ID = Node-ID of the module. The RPDO COB-IDs are automatically updated if the Node-ID is changed by LSS protocol.
80000000h in the COB-ID indicates that the PDO does not exist (destroyed)
04000000h in the COB-ID indicates that there is no RTR allowed on the PDO

Sub-Index	2h
Description	Transmission type
Access	RO
PDO Mapping	No
Value Range	See value definition in DS-301
Default Value	255 (FFh) = Event Driven

Sub-Index	3h
Description	Inhibit Time
Access	RW
PDO Mapping	No
Value Range	See value definition in DS-301
Default Value	0

Sub-Index	4h
Description	Compatibility entry
Access	RW
PDO Mapping	No
Value Range	UNSIGNED8
Default Value	0

Sub-Index	5
Description	Event-timer
Access	RW
PDO Mapping	No
Value Range	See value definition in DS-301
Default Value	0

Recall: A non-zero event timer for an RPDO means that it will result in a network fault being flagged if it has not been received within this timeframe while in Operational mode.

3.2.15. TPDO Behaviour

The CAN-2AOUT-1RLYOUT Controller can support up to four TPDO messages. All TPDOs use the similar default communication parameters, with the PDO IDs set according to the pre-defined connection set described in DS-301. Most TPDOs do not exist, there is no RTR allowed, they use 11-bit CAN-IDs (base frame valid) and they are all time-driven. While all four have valid default mappings defined (see below) only TPDO1 is enabled by default (i.e. TPDO exists).

TPDO1 Mapping at Object 1A00h: Default ID 0x180 + Node ID

Sub-Index	***Value***	***Object***
0	2	Number of mapped application objects in PDO
1	0x23700110	Analog Output 1 Feedback Field Value
2	0x23700210	Analog Output 2 Feedback Field Value
3	0	Not used by default
4	0	Not used by default

TPDO2 Mapping at Object 1A01h: Default ID 0x280 + Node ID

Sub-Index	***Value***	***Object***
0	2	Number of mapped application objects in PDO
1	0x73300110	Analog Output 1 Field Value
2	0x73300210	Analog Output 2 Field Value
3	0	Not used by default
4	0	Not used by default

TPDO3 Mapping at Object 1A02h: Default ID 0x380 + Node ID

Sub-Index	***Value***	***Object***
0	0	Number of mapped application objects in PDO
1	0	Not used by default
2	0	Not used by default
3	0	Not used by default
4	0	Not used by default

TPDO4 Mapping at Object 1A03h: Default ID 0x480 + Node ID

Sub-Index	Value	Object
0	2	Number of mapped application objects in PDO
1	0x50200020	Power Supply Field Value (measured)
2	0x50300020	Processor Temperature Field Value (measured)
3	0	Not used by default
4	0	Not used by default

Since all but TPDO1 has a zero value transmission rate (i.e. Event Timer in sub-index 5 of communication object), only TPDO1 will be automatically broadcasted when the unit goes into OPERATIONAL mode.

Object Description

Index	1800h to 1803h
Name	TPDO communication parameter
Object Type	RECORD
Data Type	PDO Communication Record

Entry Description

Sub-Index	0h
Description	Number of entries
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Sub-Index	1h
Description	COB-ID used by TPDO
Access	RW
PDO Mapping	No
Value Range	See value definition in DS-301
Default Value	40000000h + TPDO1 + Node-ID C0000000h + TPDOx + Node-ID

X	***TPDOx ID***
1	0180h
2	0280h
3	0380h
4	0480h

Node-ID = Node-ID of the module. The TPDO COB-IDs are automatically updated if the Node-ID is changed by LSS protocol.
80000000h in the COB-ID indicates that the PDO does not exist (destroyed)
04000000h in the COB-ID indicates that there is no RTR allowed on the PDO

Sub-Index	2h
Description	Transmission type
Access	RO
PDO Mapping	No
Value Range	See value definition in DS-301
Default Value	254 (FEh) = Event Driven

Sub-Index	3h
Description	Inhibit Time
Access	RW
PDO Mapping	No
Value Range	See value definition in DS-301
Default Value	0

Sub-Index	4h
Description	Compatibility entry
Access	RW
PDO Mapping	No
Value Range	UNSIGNED8
Default Value	0

Sub-Index	5
Description	Event-timer
Access	RW
PDO Mapping	No
Value Range	See value definition in DS-301
Default Value	100ms (on TPDO1) 0ms (on TPDO2, TPDO3, TPDO4)

3.3. APPLICATION OBJECTS (DS-404)

Index (hex)	***Object***	***Object Type***	***Data Type***	***Access***	***PDO Mapping***
6220	DO Write State 1 Output Line	ARRAY	UNSIGNED8	RW	Yes
6250	DO Fault Mode 1 Output Line	ARRAY	UNSIGNED8	RW	No
6260	DO Fault State 1 Output Line	ARRAY	UNSIGNED8	RW	No
6302	AO Decimal Digits PV	ARRAY	UNSIGNED8	RW	No
6310	AO Output Type	ARRAY	UNSIGNED16	RW	No
6332	AO Decimal Digits FV	ARRAY	UNSIGNED8	RW	No
6340	AO Fault Mode	ARRAY	UNSIGNED8	RW	No
7300	AO Output Process Value	ARRAY	INTEGER16	RW	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	RO	Yes
7341	AO Fault Field Value	ARRAY	INTEGER16	RW	No

3.3.1. Object 6220h: DO Write State 1 Output Line

This object shall set a single digital output logic state when the corresponding DO is being controlled by a CANopen® Message (per Table 4B in Section 1.3).

Object Description

Index	6220h
Name	DO Write State 1 Output Line
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Sub-Index	1h to 2h (x=1 to 2)
Description	DOx Write State
Access	RW
PDO Mapping	Yes
Value Range	0 (OFF) or 1 (ON)
Default Value	0 (OFF)

Sub-Index	3h
Description	Relay Write State
Access	RW
PDO Mapping	Yes
Value Range	0 (OFF) or 1 (ON)
Default Value	0 (OFF)

3.3.2. Object 6250h: DO Fault Mode 1 Output Line

This object defines how a single digital output shall response when a fault condition is detected on any control input, as described in Table 2.

Object Description

Index	6250h
Name	DO Fault Mode 1 Output Line
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	DOx Fault Mode
Access	RW
PDO Mapping	No
Value Range	See Table 2
Default Value	1 (apply pre-defined state)

3.3.3. Object 6260h: DO Fault State 1 Output Line

This object defines the pre-defined state of a single digital output when a fault condition is present, and the corresponding sub-index in object 6250h is enabled.

Object Description

Index	6260h
Name	DO Fault State 1 Output Line
Object Type	ARRAY
Data Type	BOOLEAN

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	DOx Fault State
Access	RW
PDO Mapping	No
Value Range	0 (OFF) or 1 (ON)
Default Value	1 (ON)

3.3.4. Object 6302h: AO Decimal Digits PV

This object describes the number of digits following the decimal point (i.e. resolution) of the output control data, which is interpreted with data type Integer16 in the process value object.

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Decimal Digits PV
Access	RW
PDO Mapping	No
Value Range	0 to 4
Default Value	3 [Volt to mV]

3.3.5. Object 6310h: AO Output Type

This object specifies the type of analog output, as defined in Table 3.

Object Description

Index	6310h
Name	AO Output Type
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Type
Access	RW
PDO Mapping	No
Value Range	See Table 3
Default Value	10 (Voltage)

3.3.6. Object 6332h: AO Decimal Digits FV

This object describes the number of digits following the decimal point (i.e. resolution) of the output data, which is interpreted with data type Integer16 in the field value object.

Object Description

Index	6332h
Name	AO Decimal Digits FV
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Decimal Digits FV
Access	RW
PDO Mapping	No
Value Range	0 to 4
Default Value	3 [mV]

3.3.7. Object 6340h: AO Fault Mode

This object defines how an analog output shall response when a fault condition is detected on any control input, as described in Table 9.

Object Description

Index	6340h
Name	AO Fault Mode
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Fault Mode
Access	RW
PDO Mapping	No
Value Range	See Table 9
Default Value	1 (apply pre-defined FV)

3.3.8. Object 7300h: AO Output Process Value

This object represents the process value of the output. It can be used as an input to the analog output function block when the input has been selected as controlled by a CANopen® Message (per Table 4B in Section 1.3).

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Process Value
Access	RW
PDO Mapping	Yes
Value Range	Integer16
Default Value	No

3.3.9. Object 7320h: AO Output Scaling 1 PV

This object defines the minimum value of the input and should be specified to equal the corresponding scaling object of the control source, as outlined in Table 5. It will be scaled in the physical unit of the control source. The resolution will ALWAYS be dependent on object 6302h AO Decimal Digits PV, even when the output is not being controlled directly by the AO Output PV object 7300h. This object must always be smaller than object 7322h AO Output Scaling 2 PV.

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Scaling 1 PV
Access	RW
PDO Mapping	No
Value Range	See Table 5
Default Value	0 (mV)

3.3.10. Object 7321h: AO Output Scaling 1 FV

This object defines the output field value when the input data is at or below the AO Output Scaling 1 PV value as shown in Figure 3. It will be scaled in the physical unit of the output, dependent on type, with the resolution defined in object 6332h AO Decimal Digits FV. The value can be set anywhere within the allowable output range as outlined in Table 5. This value can be set higher than object 7323h AO Output Scaling 2 FV for an inverse response (i.e. decreasing) to an increasing input.

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Scaling 1 FV
Access	RW
PDO Mapping	No
Value Range	Dependent on type (see Table 5)
Default Value	0 [mV]

3.3.11. Object 7322h: AO Output Scaling 2 PV

This object defines the maximum value of the input and should be specified to equal the corresponding scaling object of the control source, as outlined in Table 5. It will be scaled in the physical unit of the control source. The resolution will ALWAYS be dependent on object 6302h AO Decimal Digits PV, even when the output is not being controlled directly by the AO Output PV object 7300h. This object must always be larger than object 7322h AO Output Scaling 2 PV.

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Scaling 2 PV
Access	RW
PDO Mapping	No
Value Range	See Table 5
Default Value	5000 [mV]

3.3.12. Object 7323h: AO Output Scaling 2 FV

This object defines the output field value when the input data is at or above the AO Output Scaling 2 PV value as shown in Figure 3. It will be scaled in the physical unit of the output, dependent on type, with the resolution defined in object 6332h AO Decimal Digits FV. The value can be set anywhere within the allowable output range as outlined in Table 3. This value can be set lower than object 7321h AO Output Scaling 1 FV for an inverse response (i.e. decreasing) to an increasing input.

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Scaling 2 FV
Access	RW
PDO Mapping	No
Value Range	Dependent on type (see Table 5)
Default Value	5000 [mV]

3.3.13. Object 7330h: AO Output Field Value

This object represents the target output drive field value as a result of the output logic described in Section 1.3, and the scaling applied as shown in Figure 3. It is defined in the physical unit of the output dependent on type, as outlined in Table 5. The resolution of the object is defined in object 6332h AO Decimal Digits FV.

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Field Value
Access	RO
PDO Mapping	Yes
Value Range	Integer16
Default Value	No

3.3.14. Object 7341h: AO Fault FV

This object contains the pre-defined field value of an analog output when a fault condition is present, and the corresponding sub-index in object 7341h is enabled. It will be scaled in the physical unit of the output, dependent on type, with the resolution defined in object 6332h AO Decimal Digits FV.

Object Description

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

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Fault FV
Access	RW
PDO Mapping	No
Value Range	Dependent on type (see Table 5)
Default Value	0 [mV]

3.4. MANUFACTURER OBJECTS

Index (hex)	Object	Object Type	Data Type	Access	PDO Mapping
2223	DO Blink Rate 1 Output Line	ARRAY	UNSIGNED16	RW	No
2230	DO Fixed DC Frequency	ARRAY	UNSIGNED16	RW	No
2240	DO Digital Range	ARRAY	UNSIGNED8	RW	No
2241	DO Polarity	ARRAY	UNSIGNED8	RW	No
2242	Relay Polarity	ARRAY	UNSIGNED8	RW	No
2250	Relay Blink Rate	ARRAY	UNSIGNED16	RW	No
2260	Relay Control Source	ARRAY	UNSIGNED8	RW	No
2261	Relay Control Number	ARRAY	UNSIGNED8	RW	No
2270	Relay Unlatch Source	ARRAY	UNSIGNED8	RW	No
2271	Relay Unlatch Number	ARRAY	UNSIGNED8	RW	No
2280	Relay Enable Source	ARRAY	UNSIGNED8	RW	No
2281	Relay Enable Number	ARRAY	UNSIGNED8	RW	No
2282	Relay Enable Response	ARRAY	UNSIGNED8	RW	No
2290	Relay Override Source	ARRAY	UNSIGNED8	RW	No
2291	Relay Override Number	ARRAY	UNSIGNED8	RW	No
2292	Relay Override response	ARRAY	UNSIGNED8	RW	No
2293	Relay Override State	ARRAY	UNSIGNED8	RW	No
22A0	Relay Enable Response Delay	ARRAY	UNSIGNED8	RW	No
22A1	Relay ON Delay	ARRAY	UNSIGNED32	RW	No
22A2	Relay OFF Delay	ARRAY	UNSIGNED32	RW	No
2300	AO Override Field Value	ARRAY	INTEGER16	RW	No
2330	AO Ramp Up	ARRAY	UNSIGNED16	RW	No
2331	AO Ramp Down	ARRAY	UNSIGNED16	RW	No
2340	AO Control Input Source	ARRAY	UNSIGNED8	RW	No
2341	AO Control Input Number	ARRAY	UNSIGNED8	RW	No
2342	AO Control Input Response	ARRAY	UNSIGNED8	RW	No
2350	AO Enable Input Source	ARRAY	UNSIGNED8	RW	No
2351	AO Enable Input Number	ARRAY	UNSIGNED8	RW	No
2352	AO Enable Input Response	ARRAY	UNSIGNED8	RW	No
2360	AO Override Input Source	ARRAY	UNSIGNED8	RW	No
2361	AO Override Input Number	ARRAY	UNSIGNED8	RW	No
2362	AO Override Input Response	ARRAY	UNSIGNED8	RW	No
2370	AO Feedback Field Value	ARRAY	INTEGER16	RO	Yes
2380	AO Output Frequency	ARRAY	UNSIGNED16	RW	No
2500	EC Extra Received Process Value	ARRAY	INTEGER16	RW	Yes
2502	EC Decimal Digits PV	ARRAY	UNSIGNED8	RW	No
2520	EC Scaling 1 PV	ARRAY	INTEGER16	RW	No
2522	EC Scaling 2 PV	ARRAY	INTEGER16	RW	No
3yz0	LTyz Input X-Axis Source	VAR	UNSIGNED8	RW	No
3yz1	LTyz Input X-Axis Number	VAR	UNSIGNED8	RW	No
3yz2	LTyz Auto Repeat	VAR	UNSIGNED8	RW	No
3yz3	LTyz X-Axis Decimal Digits PV	VAR	UNSIGNED8	RW	No
3yz4	LTyz Y-Axis Decimal Digits PV	VAR	UNSIGNED8	RW	No
3yz5	LTyz Point Response	ARRAY	UNSIGNED8	RW	No
3yz6	LTyz Point X-Axis PV	ARRAY	INTEGER32	RW	No
3yz7	LTyz Point Y-Axis PV	ARRAY	INTEGER16	RW	No
3yz8	LTyz Output Y-Axis PV	VAR	INTEGER16	RO	Yes

3300	Logic Block Enable	ARRAY	UNSIGNED8	RW	No
3310	Logic Block Selected Table	ARRAY	UNSIGNED8	RO	Yes
3320	Logic Output Process Value	ARRAY	INTEGER16	RO	Yes
3x01	LB(x-3) Lookup Table Number	ARRAY	UNSIGNED8	RW	No
3x02	LB(x-3) Function Logical Operator	ARRAY	UNSIGNED8	RW	No
3x11	LB(x-3) Function A Condition 1	RECORD	UNSIGNED8	RW	No
3x12	LB(x-3) Function A Condition 2	RECORD	UNSIGNED8	RW	No
3x13	LB(x-3) Function A Condition 3	RECORD	UNSIGNED8	RW	No
3x21	LB(x-3) Function B Condition 1	RECORD	UNSIGNED8	RW	No
3x22	LB(x-3) Function B Condition 2	RECORD	UNSIGNED8	RW	No
3x23	LB(x-3) Function B Condition 3	RECORD	UNSIGNED8	RW	No
3x31	LB(x-3) Function C Condition 1	RECORD	UNSIGNED8	RW	No
3x32	LB(x-3) Function C Condition 2	RECORD	UNSIGNED8	RW	No
3x33	LB(x-3) Function C Condition 3	RECORD	UNSIGNED8	RW	No
4000	Math Block Enable	ARRAY	UNSIGNED8	RW	No
4021	Math Output Scaling 1 PV	ARRAY	INTEGER16	RW	No
4023	Math Output Scaling 2 PV	ARRAY	INTEGER16	RW	No
4030	Math Output Process Value	ARRAY	INTEGER16	RO	Yes
4032	Math Output Decimal Digits PV	ARRAY	UNSIGNED8	RW	No
4y00	Math Y Input Source	ARRAY	UNSIGNED8	RW	No
4y01	Math Y Input Number	ARRAY	UNSIGNED8	RW	No
4y02	Math Y Function Number	ARRAY	UNSIGNED8	RW	No
4y03	Math Y Input Decimal Digits FV	ARRAY	UNSIGNED8	RW	No
4y20	Math Y Input Scaling 1 FV	ARRAY	INTEGER16	RW	No
4y22	Math Y Input Scaling 2 FV	ARRAY	INTEGER16	RW	No
4y40	Math Y Input Gain	ARRAY	INTEGER8	RW	No
4y50	Math Y Operator	ARRAY	UNSIGNED8	RW	No
5020	Power Supply Field Value	VAR	FLOAT32	RO	Yes
5021	Enable Error Detection on Power Supply	VAR	UNSIGNED8	RW	No
5022	Under Voltage Threshold	VAR	UNSIGNED8	RW	No
5023	Over Voltage Threshold	VAR	UNSIGNED8	RW	No
5024	Hysteresis to Clear Supply Fault	VAR	UNSIGNED8	RW	No
5025	Power Supply Fault Shutdown	VAR	UNSIGNED8	RW	No
5030	Processor Temperature Field Value	VAR	FLOAT32	RO	Yes
5031	Enable Error Detection on Temperature	VAR	UNSIGNED8	RW	No
5032	Over Temperature Threshold	VAR	UNSIGNED8	RW	No
5033	Hysteresis to Clear Temperature Fault	VAR	UNSIGNED8	RW	No
5034	Temperature Fault Shutdown	VAR	UNSIGNED8	RW	No
5550	Enable Auto Updates	VAR	UNSIGEND8	RW	No
5555	Start in Operational Mode	VAR	BOOLEAN	RW	No
5B50	Change Baud Rate	VAR	UNSIGNED8	RW	No
5B51	Change Node ID	VAR	UNSIGNED8	RW	No

*Where **yz** = 01 to 10 (LUT 1 to 10) and **x** = 4 to 6 (Logic 1 to 3) and **y** = 1 to 4 (Math 1 to 4)*

3.4.1. Object 2223h: DO Blink Rate 1 Output Line

This object is used only when an On/Off digital output has been specified for a blinking response by object 2241h DO Polarity. While the DO is commanded ON, it will blink on/off at the rate specified in this object. The physical unit is milliseconds.

Object Description

Index	2223h
Name	DO Blink Rate1 Output Line
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	DOx Blink Rate
Access	RW
PDO Mapping	Yes
Value Range	0 to 60,000
Default Value	500 [ms]

3.4.2. Object 2240h: DO Digital Range

This object determines if the output is at *+5V* or *+12V* when ON. It does not apply to *Output Voltage* and *Output Current* type.

Object Description

Index	2240h
Name	DO Digital Range
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	DOx Digital Range
Access	RW
PDO Mapping	No
Value Range	0 (5V) to 1 (12V)
Default Value	0 (5V)

3.4.3. Object 2241h: DO Polarity 1 Output Line

This object defines the relationship between the logic state and the drive state of a single digital output, as described in Table 1.

Object Description

Index	2241h
Name	DO Polarity 1 Output Line
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	DOx Polarity
Access	RW
PDO Mapping	No
Value Range	See Table 1
Default Value	0 (normal on/off)

3.4.4. Object 2242h: Relay Polarity

This object defines the relationship between the logic state and the drive state of the relay output, as described in Table 10.

Object Description

Index	2242h
Name	Relay Polarity 1 Output Line
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Polarity
Access	RW
PDO Mapping	No
Value Range	See Table 10
Default Value	0 (normal on/off)

3.4.5. Object 2250h: Relay Blink Rate

This object is used only when the relay output has been specified for a blinking response by object 2242h Relay Polarity. While the relay is commanded ON, it will blink on/off at the rate specified in this object. The physical unit is milliseconds.

Object Description

Index	2250h
Name	Relay Blink Rate Output Line
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Blink Rate
Access	RW
PDO Mapping	Yes
Value Range	0 to 60,000
Default Value	500 [ms]

3.4.6. Object 2260h: Relay Control Source

This object defines the type of input that will be used to control the relay output. The available control sources on the controller are listed in Table 4A. Not all sources would make sense to control the relay, it is the user's responsibility to select a source that makes sense for the application.

Object Description

Index	2260h
Name	Relay Control Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Number of Entries
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Control Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	1 (CANopen® Message)

3.4.7. Object 2261h: Relay Control Number

This object defines the number of the source that will be used to control the relay output. The available control numbers are dependent on the source selected, as shown in Table 4B. Once selected, the control state will drive the relay according to the selected Relay Polarity.

Object Description

Index	2261h
Name	Relay Control Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Number of Entries
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Control Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	3 (CANopen® Message 3)

3.4.8. Object 2270h: Relay Unlatch Source

This object defines the type of input that will be used to active the unlatch signal for the relay (when the relay polarity is set to Latched Logic or Inverse Latched Logic only). The available unlatch sources on the controller are listed in Table 4A. Not all sources would make sense to unlatch the relay, and it is the user's responsibility to select a source that makes sense for the application.

Object Description

Index	2270h
Name	Relay Unlatch Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Unlatch Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	0 (control not used)

3.4.9. Object 2271h: Relay Unlatch Number

This object defines the number of the source that will be used to unlatch the relay. The available unlatch numbers are dependent on the source selected, as shown in Table 4B. Once selected, the unlatch state will be interpreted as an input to the relay.

Object Description

Index	2271h
Name	Relay Unlatch Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Unlatch Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	0 (control not used)

3.4.10. Object 2280h: Relay Enable Source

This object defines the type of input that will be used to enable/disable the relay output. The available control sources on this controller are listed in Table 4A. Not all sources would make sense to enable the relay, and it is the user's responsibility to select a source that makes sense for the application.

Object Description

Index	2280h
Name	Relay Enable Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Number of Entries
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Enable Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	0 (control not used)

3.4.11. Object 2281h: Relay Enable Number

This object defines the number of the source that will be used to enable/disable the relay output. The available control numbers are dependent on the source selected, as shown in Table 4B. Once selected, the enable state will be interpreted as a digital input.

Object Description

Index	2281h
Name	Relay Enable Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Enable Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	0 (null source selected)

3.4.12. Object 2282h: Relay Enable Response

This object determines if the input will act as an enable or safety interlock (i.e. input must be ON to engage the output) or a disable signal (i.e. the output will shutoff when the input is ON.) The options for this object are listed in Table 11.

Object Description

Index	2282h
Name	Relay Enable Response
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Enable Response
Access	RW
PDO Mapping	No
Value Range	See Table 11
Default Value	0 (enable when input on)

3.4.13. Object 2290h: Relay Override Source

This object defines the type of input that will be used to active the override value for the relay output. The available override sources on the controller are listed in Table 4A. Not all sources would make sense to be used for override, it is the user's responsibility to select a source that makes sense for the application.

Object Description

Index	2290h
Name	Relay Override Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Override Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	0 (control not used)

3.4.14. Object 2291h: Relay Override Number

This object defines the number of the source that will be used to override the relay output. The available override numbers are dependent on the source selected, as shown in Table 4B. Once selected, the override state will be interpreted as an input to the relay.

Object Description

Index	2291h
Name	Relay Override Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Override Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	0 (null source selected)

3.4.15. Object 2292h: Relay Override Response

This object determines how the override command will respond to the input state. The options for this object are listed in Table 12.

Object Description

Index	2292h
Name	Relay Override Response
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Override Response
Access	RW
PDO Mapping	No
Value Range	See Table 12
Default Value	0 (override when input on)

3.4.16. Object 2293h: Relay Override State

This object determines if the override state is ON (Normally Closed) or OFF (Normally Open). The options for this object are listed in Table 13.

Object Description

Index	2293h
Name	DO Override State
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Override State
Access	RW
PDO Mapping	No
Value Range	See Table 13
Default Value	0 (override state off)

3.4.17. Object 22A0h: Relay Enable Response Delay

This object defines if a time delay for output states (ON/OFF) is enabled/disabled.

Object Description

Index	22A0h
Name	Relay Enable Response Delay
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay Enable Response Delay
Access	RW
PDO Mapping	No
Value Range	0 (Disable)/ 1 (Enable)
Default Value	0

3.4.18. Object 22A1h: Relay ON Delay

This object defines the length of time delay configured before turning ON the outputs. The unit is in milliseconds.

Object Description

Index	22A1h
Name	Relay ON Delay
Object Type	ARRAY
Data Type	UNSIGNED32

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay ON Delay
Access	RW
PDO Mapping	No
Value Range	0…86400000 (24hrs) [ms]
Default Value	0

3.4.19. Object 22A2h: Relay OFF Delay

This object defines the length of time delay configured before turning OFF the outputs. The unit is in milliseconds.

Object Description

Index	22A2h
Name	Relay OFF Delay
Object Type	ARRAY
Data Type	UNSIGNED32

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	1
Default Value	1

Sub-Index	1h
Description	Relay OFF Delay
Access	RW
PDO Mapping	No
Value Range	0…86400000 (24hrs) [ms]
Default Value	0

3.4.20. Object 2300h: AO Override Field Value

This object contains the pre-defined field value of an analog output when an override condition is active. It will be scaled in the physical unit of the output, dependent on type, with the resolution defined in object 6332h AO Decimal Digits FV.

Object Description

Index	2300h
Name	AO Override FV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Override FV
Access	RW
PDO Mapping	No
Value Range	Dependent on type (see Table 5)
Default Value	0 [mV]

3.4.21. Object 2330h: AO Ramp Up

This object defines the time it will take to ramp from the minimum output PV to the maximum as defined by objects 7321h and 7323h. It can be used to soften the response to a step change at the input. The physical unit for this object is milliseconds.

Object Description

Index	2330h
Name	AO Ramp Up
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Ramp Up
Access	RW
PDO Mapping	No
Value Range	0 to 10,000
Default Value	1000 [ms]

3.4.22. Object 2331h: AO Ramp Down

This object defines the time it will take to ramp from the maximum output PV to the minimum as defined by objects 7321h and 7323h. It can be used to soften the response to a step change at the input. The physical unit for this object is milliseconds.

Object Description

Index	2331h
Name	AO Ramp Down
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Ramp Down
Access	RW
PDO Mapping	No
Value Range	0 to 10,000
Default Value	1000 [ms]

3.4.23. Object 2340h: AO Control Input Source

This object defines the type of input that will be used to control the analog (or digital) output as shown in the logic flowchart in Figure 4A. The available control sources on the CAN-2AOUT-1RLYOUT controller are listed in Table 4A. Not all sources would make sense to control the AO, and it is the user's responsibility to select a source that makes sense for the application.

Object Description

Index	2340h
Name	AO Control Input Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Control Input Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	1 (CANopen® Message)

3.4.24. Object 2341h: AO Control Input Number

This object defines the number of the source that will be used to control the analog (or digital) output as shown in the logic flowchart in Figure 4B. The available control numbers are dependent on the source selected, as shown in Table 4B. Once selected, the control represents the process value (X-Axis input) in Figure 3. Objects 6302h, 7320h, 7322h should therefore be updated to match the scaling limits defined by the control source/number, as listed in Table 5.

Object Description

Index	2341h
Name	AO Control Input Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Control Input Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	1h = 1 (CANopen® Message 1) 2h = 2 (CANopen® Message 2)

3.4.25. Object 2342h: AO Control Response

This object defines the response profile of the analog output FV with respect to the input PV (as selected by objects 2340h/2341h.) Normally it will follow the profile shown in Figure 3. However, in some cases the offset will be disabled (i.e. output at 0) when the PV is below 7320h Scaling 1 PV or alternatively above the 7322h Scaling 2 PV. The options for this object are listed in Table 6.

Object Description

Index	2342h
Name	AO Control Response
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Control Response
Access	RW
PDO Mapping	No
Value Range	See Table 6
Default Value	1 (off below scaling 1 PV)

3.4.26. Object 2350h: AO Enable Input Source

This object defines the type of input that will be used to enable/disable the analog (or digital) output as shown in the logic flowchart in Figure 4A. The available control sources on the CAN-2AOUT-1RLYOUT controller are listed in Table 4A. Not all sources would make sense to enable the AO, and it is the user's responsibility to select a source that makes sense for the application.

Object Description

Index	2350h
Name	AO Enable Input Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Enable Input Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	0 (control not used)

3.4.27. Object 2351h: AO Enable Input Number

This object defines the number of the source that will be used to enable/disable the analog (or digital) output as shown in the logic flowchart in Figure 4B. The available control numbers are dependent on the source selected, as shown in Table 4B. Once selected, the control state will be interpreted as an input to AO.

Object Description

Index	2351h
Name	AO Enable Input Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Enable Input Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	0 (null source selected)

3.4.28. Object 2352h: AO Enable Response

This object determines if the input will act as an enable or safety interlock (i.e. input must be ON to engage the output) or a disable signal (i.e. the output will shutoff when the input is ON.) The options for this object are listed in Table 7.

Object Description

Index	2352h
Name	AO Enable Response
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Enable Response
Access	RW
PDO Mapping	No
Value Range	See Table 7
Default Value	3 (Enable when OFF else shutoff)

3.4.29. Object 2360h: AO Override Input Source

This object defines the type of input that will be used to active the override value for the analog (or digital) output as shown in the logic flowchart in Figure 4A. The available control sources on the CAN-2AOUT-1RLYOUT controller are listed in Table 4A. Not all sources would make sense to enable the AO, and it is the user's responsibility to select a source that makes sense for the application.

Object Description

Index	2360h
Name	AO Override Input Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Override Input Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	0 (control not used)

3.4.30. Object 2361h: AO Override Input Number

This object defines the number of the source that will be used to override the analog (or digital) output as shown in the logic flowchart in Figure 4. The available control numbers are dependent on the source selected, as shown in Table 4B. Once selected, the override state will be interpreted as an input to AO.

Object Description

Index	2361h
Name	AO Override Input Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2

Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Override Input Number
Access	RW
PDO Mapping	No
Value Range	See Table 4
Default Value	0 (null source selected)

3.4.31. Object 2362h: AO Override Response

This object determines how the override command will respond to the input state. The options for this object are listed in Table 8.

Object Description

Index	2362h
Name	AO Override Response
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AO Override Response
Access	RW
PDO Mapping	No
Value Range	See Table 8
Default Value	0 (override when input on)

3.4.32. Object 2370h: AO Feedback Field Value

This read-only object reflects the actual measured current feedback of an analog output. For other output types (i.e. voltage, PWM or digital,) it will reflect the target Output FV or State (for DO type) based on PV vs. FV calculations (see Figure 3) and applied ramps. It can be mapped to a PDO for diagnostic purposes. It will be scaled in the physical unit of the output, dependent on type, with the resolution defined in object 6332h AO Decimal Digits FV.

Object Description

Index	2370h
Name	AO Feedback FV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Feedback FV
Access	RW
PDO Mapping	No
Value Range	Dependent on type (see Table 5)
Default Value	No

3.4.33. Object 2380h: AO Output Frequency

This object is mostly meant to be used with analog output type *Output PWM* or *Output Frequency*. It defines the fixed output frequency that will be used, as the duty cycle will be the Output FV. It can be changed with other output types (i.e. current or voltage), but Axiomatic will no longer guarantee the accuracy or responsiveness of the output as the frequency decreases. The physical unit for this object is Hertz.

Object Description

Index	2380h
Name	AO Output Frequency
Object Type	ARRAY
Data Type	UNSIGNED16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1h to 2h (x=1 to 2)
Description	AOx Output Frequency
Access	RW
PDO Mapping	No
Value Range	0 to 30,000
Default Value	25000 [Hz]

3.4.34. Object 2500h: EC Extra Received Process Value

This object provides an extra control source in order to allow other function blocks to be controlled by data received from a CANopen® RPDO. It functions similarly to any other writeable, mappable PV object, such as 7300h AO Output PV.

Object Description

Index	2500h
Name	EC Extra Received PV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	6
Default Value	6

Sub-Index	1h to 6h (x = 1 to 6)
Description	ECx Received PV
Access	RW
PDO Mapping	Yes
Value Range	Integer16
Default Value	No

3.4.35. Object 2502h: EC Decimal Digits PV

This object describes the number of digits following the decimal point (i.e. resolution) of the extra control data, which is interpreted with data type Integer16 in the process value object.

Object Description

Index	2502h
Name	EC Decimal Digits PV
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	6
Default Value	6

Sub-Index	1h to 6h (x = 1 to 6)
Description	ECx Decimal Digits PV
Access	RW
PDO Mapping	No
Value Range	0 to 4
Default Value	1 (0.1 resolution)

3.4.36. Object 2520h: EC Scaling 1 PV

This object defines the minimum value of the extra control source. It used as the Scaling 1 value by other functions blocks when the EC has been selected as the source for the X-Axis data, i.e. as seen in Figure 5. There is no physical unit associate with the data, but it uses the same resolution as the received PV as defined in object 2502h, EC Decimal Digits PV. This object must always be smaller than object 2522h EC Scaling 2 PV.

Object Description

Index	2520h
Name	EC Scaling 1 PV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	6
Default Value	6

Sub-Index	1h to 6h (x = 1 to 6)
Description	ECx Scaling 1 PV
Access	RW
PDO Mapping	No
Value Range	-32768 to 2522h sub-index X
Default Value	0

3.4.37. Object 2522h: EC Scaling 2 PV

This object defines the maximum value of the extra control source. It used as the Scaling 2 value by other functions blocks when the EC has been selected as the source for the X-Axis data, i.e. as seen in Figure 5. There is no physical unit associate with the data, but it uses the same resolution as the received PV as defined in object 2502h, EC Decimal Digits PV. This object must always be larger than object 2520h EC Scaling 1 PV.

Object Description

Index	2522h
Name	EC Scaling 2 PV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	6
Default Value	6

Sub-Index	1h to 6h (x = 1 to 6)
Description	ECx Scaling 2 PV
Access	RW
PDO Mapping	No
Value Range	2520h sub-index X to 32767
Default Value	0.0

3.4.38. Object 3yz0h: LTyz Input X-Axis Source

This object defines the type of input that will be used to determine the X-Axis input process value for the lookup table function. The available control sources on the controller are listed in Table 4A. Not all sources would make sense to use as an X-Axis input, and it is the user's responsibility to select a source that makes sense for the application. A selection of "Control Source Not Used" disables the associated lookup table function block.

Object Description

Index	3yz0h (where yz = 01 to 10)
Name	LTyz Input X-Axis Source
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	0 (control not used)

3.4.39. Object 3yz1h: LTyz Input X-Axis Number

This object defines the number of the source that will be used as the X-Axis input PV for the lookup table function. The available control numbers are dependent on the source selected, as shown in Table 4B. Once selected, the limits for the points on the X-Axis will be constrained by the scaling objects of the control source/number as defined in Table 5.

Object Description

Index	3yz1h (where yz = 01 to 10)
Name	LTyz Input X-Axis Number
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	0 (null control source)

3.4.40. Object 3yz2h: LTyz Auto Repeat

This object determines whether the lookup table sequence will repeat automatically once the last point in the lookup table has been completed. This object is only taken into effect when the response is set to *'Time Response'*. For more details on the functionality of this object and its effect on the lookup table, please refer to section 1.5.4

Object Description

Index	3yz2h (where yz = 01 to 10)
Name	LTyz X-Axis Decimal Digits PV
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0 (OFF) to 1 (ON)
Default Value	0 [OFF]

3.4.41. Object 3yz3h: LTyz X-Axis Decimal Digits PV

This object describes the number of digits following the decimal point (i.e. resolution) of the X-Axis input data and the points in the lookup table. It should be set equal to the decimal digits used by the PV from the control source/number as defined in Table 5.

Object Description

Index	3yz3h (where yz = 01 to 10)
Name	LTyz X-Axis Decimal Digits PV
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0 to 4 (see Table 5)
Default Value	0

3.4.42. Object 3yz4h: LTyz Y-Axis Decimal Digits PV

This object describes the number of digits following the decimal point (i.e. resolution) of the Y-Axis points in the lookup table. When the Y-Axis output is going to be the input to another function block (i.e. an analog output), it is recommended that this value be set equal to the decimal digits used by the block that is using the lookup table as the control source/number.

Object Description

Index	3yz4h (where yz = 01 to 12)
Name	LTyz Y-Axis Decimal Digits PV
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0 to 4
Default Value	0

3.4.43. Object 3yz5h: LTyz Point Response

This object determines the Y-Axis output response to changes in the X-Axis input. The value set in sub-index 1 determines the X-Axis type (i.e. data or time), while all other sub-indexes determine the response (ramp, step, ignore) between two points on the curve. The options for this object are listed in Table 14. See Figure 6 for an example of the difference between a step and ramp response.

Object Description

Index	3yz5h (where yz = 01 to 10)
Name	LTyz Point Response
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	11
Default Value	11

Sub-Index	1h
Description	X-Axis Type
Access	RW
PDO Mapping	No
Value Range	0 (data) or 1 (time)
Default Value	0 (x-axis data response)

Sub-Index	2h to 11h (x = 2 to 11)
Description	LTyz Point X Response
Access	RW
PDO Mapping	No
Value Range	See Table 14 (0, 1 or 2)
Default Value	1 (ramp to response)

3.4.44. Object 3yz6h: LTyz Point X-Axis PV

This object defines the X-Axis data for the 11 calibration points on the lookup table, resulting in 10 different output slopes.

When a data response is selected for the X-Axis type (sub-index 1 of object 3yz5), this object is constrained such that X1 cannot be less than the Scaling 1 value of the selected control source/number, and X11 cannot be more than the Scaling 2 value. The rest of the points are constrained by the formula below. The physical unit associate with the data will be that of the selected input, and it will use the resolution defined in object 3yz3h, LTz X-Axis Decimal Digits PV.

$$\text{MinInt16} <= X_1 <= X_2 <= X_3 <= X_4 <= X_5 <= X_6 <= X_7 <= X_8 <= X_9 <= X_{10} <= X_{11} <= \text{MaxInt16}$$

When a time response has been selected, each point on the X-Axis can be set anywhere from 1 to 86,400,000ms.

Object Description

Index	3yz6h (where yz = 01 to 10)
Name	LTyz Point X-Axis PV
Object Type	ARRAY
Data Type	INTEGER32

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	11
Default Value	11

Sub-Index	1h to 11h (x = 1 to 11)	
Description	LTyz Point X-Axis PVx	
Access	RW	
PDO Mapping	No	
Value Range	See above (data)	1 to 86400000 (time)
Default Value	10*(x-1)	No

3.4.45. Object 3yz7h: LTyz Point Y-Axis PV

This object defines the Y-Axis data for the 11 calibration points on the lookup table, resulting in 10 different output slopes. The data is unconstrained and has no physical unit associate with it. It will use the resolution defined in object 3yz4h, LTyz Y-Axis Decimal Digits PV.

Object Description

Index	3yz7h (where yz = 01 to 10)
Name	LTz Point Y-Axis PV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	11
Default Value	11

Sub-Index	1h to 11h (x = 1 to 11)
Description	LTyz Point Y-Axis PVx
Access	RW
PDO Mapping	No
Value Range	Integer16
Default Value	10*(x-1) [i.e. 0, 10, 20, 30, … 100]

3.4.46. Object 3yz8h: LTyz Output Y-Axis PV

This read-only object contains the lookup table function block PV that can be used as the input source for another function block (i.e. analog output.) The physical unit for this object is undefined, and it will use the resolution defined in object 3yz4h, LTz Y-Axis Decimal Digits PV.

Object Description

Index	3yz8h (where yz = 01 to 12)
Name	LTyz Output Y-Axis PV
Object Type	VARIABLE
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Access	RO
PDO Mapping	Yes
Value Range	Integer16
Default Value	No

3.4.47. Object 3300h: Logic Block Enable

This object defines whether or not the logic shown in Figure 8 will be evaluated.

Object Description

Index	3300h
Name	Logic Block Enable
Object Type	ARRAY
Data Type	BOOLEAN

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Sub-Index	1h to 3h (x = 1 to 3)
Description	LBx Enable
Access	RW
PDO Mapping	No
Value Range	0 (FALSE) or 1 (TRUE)
Default Value	0 [FALSE]

3.4.48. Object 3310h: Logic Block Selected Table

This read-only object reflects what table has been selected as the output source for the logic block after the evaluation shown in Figure 9 has been performed.

Object Description

Index	3310h
Name	Logic Block Selected Table
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Sub-Index	1h to 3h (x = 1 to 3)
Description	LBx Selected Table
Access	RO
PDO Mapping	Yes
Value Range	1 to 10
Default Value	No

3.4.49. Object 3320h: Logic Block Output PV

This read-only object reflects the output from the selected table, interpreted as a percentage. The limits for the percentage conversion are based on the range of the lookup tables Y-Axis Output PV as shown in Table 5. This value has a fixed decimal digit value of 1 giving a resolution of 0.1%.

Object Description

Index	3320h
Name	Logic Block Output PV
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Sub-Index	1h to 3h (x = 1 to 3)
Description	LBx Output PV
Access	RO
PDO Mapping	Yes
Value Range	Dependent on Selected Table
Default Value	No

3.4.50. Object 3x01h: LB(x-3) Lookup Table Numbers

This object determines which of the six lookup tables supports on the CAN-2AOUT-1RLYOUT are associated with a particular function within the given logic block. Up to three tables can be linked to each logic function.

Object Description

Index	3x01h (where x = 4 to 6)
Name	LB(x-3) Lookup Table Numbers
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Sub-Index	1h to 3h (y = A to C)
Description	LB(x-3) Lookup Table Y Number
Access	RW
PDO Mapping	No
Value Range	1 to 10
Default Value	See Table 20

3.4.51. Object 3x02h: LB(x-3) Function Logical Operator

This object determines how the results of the three conditions for each function are to be compared to one another to determine the overall state of the function output. There are up to three functions that can be evaluated in each logic block. The options for this object are defined in Table 16. See Section 1.6 for more information about how this object is used.

Object Description

Index	3x02h (where x = 4 to 6)
Name	LB(x-3) Function Logical Operator
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Sub-Index	1h to 3h (y = A to C)
Description	LB(x-3) Function Y Logical Operator
Access	RW
PDO Mapping	No
Value Range	See Table 16
Default Value	Function A = 1 (and all) Function B = 1 (and all) Function C = 0 (default)

3.4.52. Object 3x11h: LB(x-3) Function A Condition 1
3.4.53. Object 3x12h: LB(x-3) Function A Condition 2
3.4.54. Object 3x13h: LB(x-3) Function A Condition 3
3.4.55. Object 3x21h: LB(x-3) Function B Condition 1
3.4.56. Object 3x22h: LB(x-3) Function B Condition 2
3.4.57. Object 3x23h: LB(x-3) Function B Condition 3
3.4.58. Object 3x31h: LB(x-3) Function C Condition 1
3.4.59. Object 3x32h: LB(x-3) Function C Condition 2
3.4.60. Object 3x33h: LB(x-3) Function C Condition 3

These objects, 3xyzh, represent Logic Block z, Function y, Condition z, where x = 4 to 6, y = 1 (A) to 3 (C), and z = 1 to 3. All of these objects are a special type of record, defined in Table 19. Information on how to use these objects is defined in Section 1.6.

Object Description

Index	3xyzh
Name	LB(x-3) Function y Condition z
Object Type	RECORD
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	5
Default Value	5

Sub-Index	1h
Description	Argument 1 Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	1 (CANopen® Message)

Sub-Index	2h
Description	Argument 1 Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	0

Sub-Index	3h
Description	Argument 2 Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
eDefault Value	1 (CANopen® Message)

Sub-Index	4h
Description	Argument 2 Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	0

Sub-Index	5h
Description	Operator
Access	RW
PDO Mapping	No
Value Range	See Table 16
Default Value	0 (Equals)

3.4.61. Object 4000h: Math Function Enable

The corresponding sub-index of object must be set to TRUE in order to enable a math function block. Otherwise, the output will always be 0.

Object Description

Index	4000h
Name	Math Function Enable
Object Type	ARRAY
Data Type	BOOLEAN

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (Y = 1 to 4)
Description	Math Y Enable
Access	RW
PDO Mapping	No
Value Range	0 (FALSE) or 1 (TRUE)
Default Value	0 [FALSE]

3.4.62. Object 4021h: Math Output Scaling 1 PV

This object defines the process value that would correspond to 0% output from the math calculation. The object would apply the resolution defined in object 4532h Math Output Decimal Digits PV. The physical unit is undefined.

Object Description

Index	4021h
Name	Math Output Scaling 1 PV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (Y = 1 to 4)
Description	Math Y Output Scaling 1 PV
Access	RW
PDO Mapping	No
Value Range	-32768 to 32767
Default Value	0

3.4.63. Object 4023h: Math Output Scaling 2 PV

This object defines the process value that would correspond to 100% output from the math calculation. The object would apply the resolution defined in object 4532h Math Output Decimal Digits PV. The physical unit is undefined.

Object Description

Index	4023h
Name	Math Output Scaling 2 PV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (Y = 1 to 4)
Description	Math Y Output Scaling 2 PV
Access	RW
PDO Mapping	No
Value Range	-32768 to 32767
Default Value	10000 (100.00)

3.4.64. Object 4030h: Math Output Process Value

This read-only object reflects the output from the math function block after it has been scaled by objects 4021h and 4023h. The object would apply the resolution defined in object 4032h Math Output Decimal Digits PV. The physical unit is undefined.

Object Description

Index	4030h
Name	Math Output Process Value
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (Y = 1 to 4)
Description	Math Y Output Process Value
Access	RO
PDO Mapping	Yes
Value Range	-32768 to 32767
Default Value	No

3.4.65. Object 4032h: Math Output Decimal Digits PV

This object describes the number of digits following the decimal point (i.e. resolution) of the output data, which is interpreted with data type Integer16 in the process value object.

Object Description

Index	4032h
Name	Math Output Decimal Digits PV
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (Y = 1 to 4)
Description	Math Y Decimal Digits PV
Access	RW
PDO Mapping	No
Value Range	0 to 4
Default Value	2 (0.01)

3.4.66. Object 4y00h: Math Y Input Source

This object defines the input sources that will be used in the mathematical calculations. Here, y = 1 to 4 – representing Math Block 1 to Math Block 4. If a control source is not used, the associate mathematical calculation would be ignored. The available control sources on the CAN-2AOUT-1RLYOUT controller are listed in Table 4A.

Object Description

Index	4y00h (y = 1 to 4)
Name	Math Y Input Source
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (X = 1 to 4)
Description	Math Y Input X Source
Access	RW
PDO Mapping	No
Value Range	See Table 4A
Default Value	0 (control source not used)

3.4.67. Object 4y01h: Math Y Input Number

This object defines the number of the input source that will be used in the math calculation. The available control numbers are dependent on the source selected, as shown in Table 4B. Once selected, the input value will be used in the corresponding calculation as described in Section 1.7.

Object Description

Index	4y01h (y = 1 to 4)
Name	Math Y Input Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (X = 1 to 4)
Description	Math Y Input X Number
Access	RW
PDO Mapping	No
Value Range	See Table 4B
Default Value	0 (null input)

3.4.68. Object 4y02h: Math Y Input Function Number

This object defines the number of the function within the Math Block will be used in the math calculation. This object is applicable when the Input Source together with the Input Number match the Math Block that is being configured. If Input Source and Input Number match the Math Block being configured and the Function Number is 0, this object is ignored. For more details, refer to Section 1.7.

Object Description

Index	4y02h (y = 1 to 4)
Name	Math Y Input Number
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (X = 1 to 4)
Description	Math Y Input X Function Number
Access	RW
PDO Mapping	No
Value Range	0 to 3
Default Value	0 (null input)

3.4.69. Object 4y03h: Math Y Input Decimal Digits FV

This object describes the number of digits following the decimal point (i.e. resolution) of the input data, which is interpreted with data type Integer16 in the field value object.

Object Description

Index	4y03h (y = 1 to 4)
Name	Math Y Input Decimal Digits FV
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (X = 1 to 4)
Description	Math Y Input X Decimal Digits PV
Access	RW
PDO Mapping	No
Value Range	0 to 4
Default Value	2 (0.01)

3.4.70. Object 4y20h: Math Y Input Scaling 1 FV

This object defines the input field value that would correspond to 0% when scaling the input for use in the math calculation. All inputs are normalized to a percentage before being used by the math function block. The object would apply the resolution defined in object 4y03h Math Y Input Decimal Digits FV. The physical unit would match that of the input source.

Object Description

Index	4y20h (y = 1 to 4)
Name	Math Y Input Scaling 1 FV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (X = 1 to 4)
Description	Math Y Input X Scaling 1 FV
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	0

3.4.71. Object 4y22h: Math Y Input Scaling 2 FV

This object defines the input field value that would correspond to 100% when scaling the input for use in the math calculation. All inputs are normalized to a percentage before being used by the math function block. The object would apply the resolution defined in object 4y03h Math Y Input Decimal Digits FV. The physical unit would match that of the input source.

Object Description

Index	4y22h (y = 1 to 4)
Name	Math Y Input Scaling 2 FV
Object Type	ARRAY
Data Type	INTEGER16

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (X = 1 to 4)
Description	Math Y Input X Scaling 2 FV
Access	RW
PDO Mapping	No
Value Range	INTEGER16
Default Value	10000 (100.00%)

3.4.72. Object 4y40h: Math Y Input Gain

This object can be used to adjust the 'weight' of the input in the math calculation. It is a multiplier of the input after it has been converted into a percentage, before it is used in the math calculation. This object has a fixed resolution of 2 decimal digits.

Object Description

Index	4y40h (y = 1 to 4)
Name	Math Y Input Gain
Object Type	ARRAY
Data Type	INTEGER8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	4
Default Value	4

Sub-Index	1h to 4h (X = 1 to 4)
Description	Math Y Input X Gain
Access	RW
PDO Mapping	No
Value Range	-100 to 100
Default Value	100 (1.0)

3.4.73. Object 4y50h: Math Y Operator

This object defines the actual operators that will be used in each stage of a math calculation, as described in Section 1.7. The options for this object are listed in Table 21.

Object Description

Index	4y50h (y = 1 to 4)
Name	Math Y Operator
Object Type	ARRAY
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Description	Largest sub-index supported
Access	RO
PDO Mapping	No
Value Range	3
Default Value	3

Sub-Index	1h to 3h (X = 1 to 3)
Description	Math Y Function X Operator
Access	RW
PDO Mapping	No
Value Range	See Table 21
Default Value	12 (Plus)

3.4.74. Object 5020h: Power Supply Field Value

This read-only object is available for diagnostic feedback purposes. It reflects the measured voltage powering the controller. The physical unit for this object is volts.

Object Description

Index	5020h
Name	Power Supply Field Value
Object Type	VARIABLE
Data Type	FLOAT32

Entry Description

Sub-Index	0h
Access	RO
PDO Mapping	Yes
Value Range	0 to 35 [V]
Default Value	No

3.4.75. Object 5021h: Enable Error Detection on Power Supply

This write-able object is available to enable faults on power supply measured in the system. When this object is enabled, objects 5022h-5024h are used to determine fault triggers and fault clear thresholds. For more information please refer to Section 1.8.1.

Object Description

Index	5021h
Name	Enable Error Detection on Power Supply
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	FALSE; TRUE
Default Value	TRUE (1)

3.4.76. Object 5022h: Under Voltage Threshold

This object is used to set the value at which a supply fault will trigger if the measured supply falls below it. If object 5021h is disabled, this value is ignored. For more information please refer to Section 1.8.1.

Object Description

Index	5022h
Name	Under Voltage Threshold
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0V… Object 5023h
Default Value	8 (V)

3.4.77. Object 5023h: Over Voltage Threshold

This object is used to set the value at which a supply fault will trigger if the measured supply falls above it. If object 5021h is disabled, this value is ignored. For more information please refer to Section 1.8.1.

Object Description

Index	5023h
Name	Over Voltage Threshold
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	Object 522h… 36V
Default Value	35 (V)

3.4.78. Object 5024h: Hysteresis to Clear Supply Fault

This object is used to set the value by which the supply voltage needs to increase or decrease to clear the fault set by an under voltage or over voltage, respectively. If object 5021h is disabled, this value is ignored. For more information please refer to Section 1.8.1.

Object Description

Index	5024h
Name	Hysteresis to Clear Supply Fault
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	1V…30V
Default Value	5 (V)

3.4.79. Object 5025h: Enable Power Supply Fault Disables Outputs

This write-able object is available to enable disable on all the outputs when there are faults on power supply measured in the system. For more information please refer to Section 1.8.1.

Object Description

Index	5025h
Name	Enable Power Supply Fault Disables Outputs
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	FALSE; TRUE
Default Value	TRUE (1)

3.4.80. Object 5030h: Processor Temperature Field Value

This read-only object is available for diagnostic feedback purposes. It reflects the measured temperature of the processor, which will always run approximately 10°C to 20°C above ambient. The physical unit for this object is degrees Celsius.

Object Description

Index	5030h
Name	Processor Temperature Field Value
Object Type	VARIABLE
Data Type	FLOAT32

Entry Description

Sub-Index	0h
Access	RO
PDO Mapping	Yes
Value Range	-50 to 150 [°C]
Default Value	No

3.4.81. Object 5031h: Enable Error Detection on Temperature

This write-able object is available to enable faults on temperature measured in the system. When this object is enabled, objects 5032h-5033h are used to determine fault triggers and fault clear thresholds. For more information please refer to Section 1.8.1.

Object Description

Index	5021h
Name	Enable Error Detection on Temperature
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	FALSE; TRUE
Default Value	TRUE (1)

3.4.82. Object 5032h: Over Temperature Threshold

This object is used to set the value at which a temperature fault will trigger if the measured microcontroller temperature falls above it. If object 5031h is disabled, this value is ignored. For more information please refer to Section 1.8.1.

Object Description

Index	5032h
Name	Over Temperature Threshold
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0…150 (DegC)
Default Value	85 (DegC)

3.4.83. Object 5033: Hysteresis to Clear Temperature Fault

This object is used to set the value by which the temperature needs to decrease to clear the fault set by an over temperature event. If object 5031h is disabled, this value is ignored. For more information please refer to Section 1.8.1.

Object Description

Index	5033h
Name	Hysteresis to Clear Temperature Fault
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0…150 (DegC)
Default Value	25 (DegC)

3.4.84. Object 5034h: Enable Temperature Fault Shutdown

This write-able object is available to enable shutdown when there are faults on temperature measured in the system. For more information please refer to Section 1.8.1.

Object Description

Index	5034h
Name	Enable Temperature Fault Shutdown
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	FALSE; TRUE
Default Value	TRUE (1)

3.4.85. Object 5550h: Enable Automatic Updates

This object allows the controller to update objects to defaults automatically when an output type is changed. Be default this object is TRUE.

Object Description

Index	5550h
Name	Enable Auto Updates
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0 (FALSE) or 1 (TRUE)
Default Value	1 [TRUE]

3.4.86. Object 5555h: Start in Operational Mode

This object allows the unit to start in Operational mode without requiring the presence of a CANopen® Master on the network. It is intended to be used only when running the CAN-2AOUT-1RLYOUT controller as a stand-alone module. This should always be set FALSE whenever it is connected to a standard master/slave network.

Object Description

Index	5555h
Name	Start in Operational Mode
Object Type	VARIABLE
Data Type	BOOLEAN

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0 (FALSE) or 1 (TRUE)
Default Value	0 [FALSE]

3.4.87. Object 5B50h: Change Baud Rate

The CAN-2AOUT-1RLYOUT Controller could be configured to different baud rates by changing the value in this object. The options for this object are defined in Table 22.

Object Description

Index	5B50h
Name	Change Baud Rate
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	0…8
Default Value	4 (125k)

3.4.88. Object 5B51h: Change Node ID

This object is used to change the node ID of the module.

Object Description

Index	5B51h
Name	Change Node ID
Object Type	VARIABLE
Data Type	UNSIGNED8

Entry Description

Sub-Index	0h
Access	RW
PDO Mapping	No
Value Range	Unsigned8
Default Value	32

4. TECHNICAL SPECIFICATIONS

Specifications are indicative and subject to change. Actual performance will vary depending on the application and operating conditions. 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 https://www.axiomatic.com/service/.

Power

Power Supply Input - Nominal	12 V or 24 Vdc nominal; 9…36 Vdc The minimum allowable supply voltage for the power pin is 7 Vdc.
Surge Protection	Meets the surge requirements of SAE J1445
Reverse Polarity Protection	Provided
Under and Over-Voltage Protections	Under-voltage shutdown at 7 V Over-voltage shutdown at 39 V

Outputs

Analog/Digital Output	**AX130751**: 2 signal outputs configurable as: 0-5V, 0-10V, -5V-5V, -10V-10V, 0-20mA, 4-20 mA or PWM/Frequency **AX130771**: 2 isolated signal outputs configurable as: 0-5V, 0-10V, -5V-5V, -10V-10V, 0-20mA, 4-20 mA, PWM/Frequency or Digital Analog Voltage or Current Outputs: Voltage Output: -5 to 5 Vdc, -10 to 10 Vdc 0-5 Vdc, 0-10 Vdc Maximum load is 30 mA. Current Output: 0-20 mA or 4-20 mA Maximum load resistance is < 500 Ohms. Compliance Voltage is 14V. PWM or Frequency Outputs 0.1 Hz to 20 kHz 0-100% D.C. 5 V or 12 V Amplitude Push pull output Maximum load is 50 mA. Over-current protection (50 mA)
Output Accuracy	Voltage: 0.2% Current: 0.2% PWM Signal: 0.1% Frequency Signal: 0.1%
Relay Output	Sets 1 Form C relay output Resistive load: • 5A (NO)/5 A (NC) at 220VDC/250VAC Dielectric strength: • 3,000 VAC, 50/60 Hz for 1 min between coil and contacts • 2,500 Vrms between open contacts There is no special overcurrent/overvoltage protection on the relay outputs. The user is advised to provide a fast acting 6A fuse or an adequate external protection if necessary.

Control Software

Software Platform	Pre-programmed with standard logic. Refer to the user manual.

General Specifications

Memory	STM32F405RGT7; 32-bit, 1024 Kbytes Flash Program Memory
CAN Port	**AX130751**: 1 CANopen® SAE J1939 model is PN AX130750 **AX130771**: 1 Isolated CAN (CANopen®) SAE J1939 model is PN AX130770
Isolation	300 Vrms isolation for CAN port
Quiescent Current Draw	**AX130751**: Typical 63mA @12Vdc; 33mA @ 24Vdc **AX130771**: Typical 65mA @12Vdc; 35mA @ 24Vdc
Response Time	<10 mSec. Typical
Operating Conditions	-40 to 85°C (-40 to 185°F)
Weight	**AX130751**: 0.15 lb. (0.068 kg) **AX130771**: 0.50 lb. (0.227 kg)
Protection Rating	IP67
Vibration	MIL-STD-202G, Test 204D and 214A (Sine and Random) 10 g peak (Sine); 7.86 Grms peak (Random)

Shock	MIL-STD-202G, Test 213B, 50 g
Enclosure and Dimensions	**AX130751**: Molded Enclosure, integral connector Nylon 6/6, 30% glass Ultrasonically welded 3.54 x 2.75 x 1.31 inches (90.09 x 70.00 x 33.35 mm) L x W x H including integral connector Refer to Figure 14, dimensional drawing. **AX130771**: High Temperature Nylon PCB Enclosure – (equivalent TE Deutsch P/N: EEC-325X4B) 4.677 x 5.236 x 1.417 inches 118.80 x 133.00 x 36.00 mm (W x L x H excluding mating plugs) Refer to Figure 15, dimensional drawing..
Mating Plug Kit	**AX130751**: Axiomatic P/N: **PL-DTM06-12SA** It is comprised of the following TE Deutsch P/N equivalents: 1 plug DTM06-12SA, 1 wedgelock WM-12S, 12 contacts 0462-201-20141, 6 sealing plugs 0413-204-2005. **AX130771**: Axiomatic P/N: **PL-DTM06-12SA** It is comprised of the following TE Deutsch P/N equivalents: 1 plug DTM06-12SA, 1 wedgelock WM-12S, 12 contacts 0462-201-20141, 6 sealing plugs 0413-204-2005.
User Interface	EDS File, Standard CANopen® tools (not supplied)
Reflashing	Axiomatic Electronic Assistant KIT, P/Ns: **AX070502** or **AX070506K**

CANopen® is a registered community trademark of CAN in Automation e.V.

5. VERSION HISTORY

User Manual Version	Date	Author	Modifications
1	Feb 10th, 2020	Jessica Chen	Initial Draft
-	Mar 25th, 2020	Jessica Chen	Updated Vibration information
1A	Jan 27th, 2022	Jordan Wilbur	Replaced 800 kbit/s baud rate with 666 kbit/s
-	Apr 24th, 2023	M Ejaz	Marketing review Corrected under and over-voltage thresholds and relay resistive load
1B	July 28th, 2023	Kiril Mojsov	Performed Legacy Updates

OUR PRODUCTS

AC/DC Power Supplies

Actuator Controls/Interfaces

Automotive Ethernet Interfaces

Battery Chargers

CAN Controls, Routers, Repeaters

CAN/WiFi, CAN/Bluetooth, Routers

Current/Voltage/PWM Converters

DC/DC Power Converters

Engine Temperature Scanners

Ethernet/CAN Converters, Gateways, Switches

Fan Drive Controllers

Gateways, CAN/Modbus, RS-232

Gyroscopes, Inclinometers

Hydraulic Valve Controllers

Inclinometers, Triaxial

I/O Controls

LVDT Signal Converters

Machine Controls

Modbus, RS-422, RS-485 Controls

Motor Controls, Inverters

Power Supplies, DC/DC, AC/DC

PWM Signal Converters/Isolators

Resolver Signal Conditioners

Service Tools

Signal Conditioners, Converters

Strain Gauge CAN Controls

Surge Suppressors

OUR COMPANY

Axiomatic provides electronic machine control components to the off-highway, commercial vehicle, electric vehicle, power generator set, material handling, renewable energy and industrial OEM markets. ***We innovate with engineered and off-the-shelf machine controls that add value for our customers.***

QUALITY DESIGN AND MANUFACTURING

We have an ISO9001:2015 registered design/manufacturing facility in Canada.

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 at https://www.axiomatic.com/service/.

COMPLIANCE

Product compliance details can be found in the product literature and/or on axiomatic.com. Any inquiries should be sent to sales@axiomatic.com.

SAFE USE

All products should be serviced by Axiomatic. Do not open the product and perform the service yourself.

This product can expose you to chemicals which are known in the State of California, USA to cause cancer and reproductive harm. For more information go to www.P65Warnings.ca.gov.

SERVICE

All products to be returned to Axiomatic require a Return Materials Authorization Number (RMA#) from sales@axiomatic.com. Please provide the following information when requesting an RMA number:

- Serial number, part number
- Runtime hours, description of problem
- Wiring set up diagram, application and other comments as needed

DISPOSAL

Axiomatic products are electronic waste. Please follow your local environmental waste and recycling laws, regulations and policies for safe disposal or recycling of electronic waste.

CONTACTS

Axiomatic Technologies Corporation
1445 Courtneypark Drive E.
Mississauga, ON
CANADA L5T 2E3
TEL: +1 905 602 9270
FAX: +1 905 602 9279
www.axiomatic.com
sales@axiomatic.com

Axiomatic Technologies Oy
Höytämöntie 6
33880 Lempäälä
FINLAND
TEL: +358 103 375 750
www.axiomatic.com
salesfinland@axiomatic.com

Copyright 2023