USER MANUAL UMAX022400
USER MANUAL UMAX022410

DUAL UNIVERSAL INPUT, DUAL VALVE CONTROLLER

With Programmable Logic

USER MANUAL

P/N: AX022400 – Top mount connector

P/N: AX022410 – Side mount connector

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

VERSION HISTORY

Version	Date	Author	Modifications
1.0.0	May 16, 2011	Anna Murray	Initial Draft, reviewed by Stuart Byma
1.0.1	August 5, 2011	Amanda Wilkins	Added upgrade of +5V reference to 50mA, added weight
2.0.0	December 5, 2011	Anna Murray	Added section explaining how to reflash the ECU via CAN with the Axiomatic EA Bootloader. Updated "General ECU Information" for EA V4.1.35.1+
2.1.0	March 5, 2012	Anna Murray	Updated PWM output to have a wider output frequency range (1Hz to 25KHz) Added information about the output mirroring feature for both outputs in PWM mode.
2.1.1	March 13, 2012	Anna Murray	Minor update to the PID section for clarification
2.2.0	May 25, 2012	Anna Murray	Added "Auto update when control changes" setpoint. Changes made to Lookup Table function block. Addition information about flashing "ECU Instance" and "ECU Address" when downloading a setpoint file, in Section 4.1
3.0.0	June 11, 2012	Anna Murray	Added the side mount connector part number AX022410. Updated the mounting instructions section 2.2.
3.1.0	July 5, 2012	Anna Murray	Updated the picture for the AX022410 part number in Figure 30. Updated the information for the mating plug kits for both part numbers. Changed +Vcc in pullup reference to +5V. Changed other references of +Vcc to +Vps. Added BATT+/-, GND and Vps to acronym list Updated technical specifications in Appendix B with additional information including response times.
3.1.1	July 27, 2012	Anna Murray	Added an additional note about the "Auto update when control changes" feature on the REFERENCES page, as well as a note about which version of the Axiomatic EA supports all the features for the AX0224x0 part numbers described in this manual.
3.2.0	August 10, 2023	Kiril Mojsov	Performed Legacy Updates

ACCRONYMS

ACK	Positive Acknowledgement	(from SAE J1939 standard)
BATT +/-	Battery positive (a.k.a Vps) or Battery Negative (a.k.a. GND)	
DIN	Digital Input used to measure active high or low signals	
DM	Diagnostic Message	(from SAE J1939 standard)
DOUT	Digital Output, sourcing (high-side) output up to 3A current	
DTC	Diagnostic Trouble Code	(from SAE J1939 standard)
EA	The Axiomatic Electronic Assistant (A Service Tool for Axiomatic ECUs)	
ECU	Electronic Control Unit	(from SAE J1939 standard)
EM	Electro-Magnetic	
EMC	Electro-Magnetic Compliance	
EMI	Electro-Magnetic Immunity	
FIN	Frequency Input used to measure Frequency, RPM or PWM signals	
GND	Ground reference (a.k.a. BATT-)	
I/O	Inputs and Outputs	
NAK	Negative Acknowledgement	(from SAE J1939 standard)
PDU1	A format for messages that are to be sent to a destination address, either specific or global	(from SAE J1939 standard)
PDU2	A format used to send information that has been labeled using the Group Extension technique, and does not contain a destination address.	
PGN	Parameter Group Number	(from SAE J1939 standard)
PropA	Message that uses the Proprietary A PGN for peer-to-peer communication	
PropB	Message that uses a Proprietary B PGN for broadcast communication	
PWM	Pulse Width Modulation	
RPM	Rotations per Minute	
SPN	Suspect Parameter Number	(from SAE J1939 standard)
UIN	Universal Input used to measure voltage, current, resistive, frequency or digital inputs	
UOUT	Universal Output, 0-3A current, digital, voltage or PWM type	
Vps	Voltage Power Supply a.k.a BATT+	

Note:

An Axiomatic Electronic Assistant KIT may be ordered as P/N: AX070502 or AX070506K

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

REFERENCES

J1939	Recommended Practice for a Serial Control and Communications Vehicle Network, SAE, April 2011
J1939/21	Data Link Layer, SAE, December 2010
J1939/71	Vehicle Application Layer, SAE, March 2011
J1939/73	Application Layer-Diagnostics, SAE, February 2010
J1939/81	Network Management, SAE, May 2003
TDAX022400	Technical Datasheet, Dual Input, Dual Output Valve Controller
UMAX07050x	User Manual V3.0.39, Axiomatic Electronic Assistant and USB-CAN, Axiomatic Technologies, August 2023

This document assumes the reader is familiar with the SAE J1939 standard. Terminology from the standard is used, but is not described in this document.

NOTE: This product is supported by Axiomatic Electronic Assistant V4.2.39.0 and higher

NOTE: When a description is in "**double-quotes**" and bolded, this refers to the name of a user configurable setpoint (variable). If it is in '*single-quotes*' and italicized, it refers to an option for the associated setpoint.

For example: "**Input Sensor Type**" set to *'Current'*

This product uses the Axiomatic Electronic Assistant to program the setpoints for application specific requirements. After configuration, the setpoints can be saved in a file which could then be flashed into other AX0224x0 controllers over the CAN network.

One point to note is that if the setpoint **"Auto update when control changes"** is set to TRUE in the file, the controller will automatically update some setpoints to new defaults when key setpoints are changed. This is a useful feature during configuration, but during a setpoint file upload it may be required to reflash the same file a second time if this feature is active. The second upload will ensure all setpoints are correctly updated, since some of them may have been automatically overwritten during the first reflashing process, and therefore don't match the values in the file.

To avoid this potential problem, it is HIGHLY recommended by Axiomatic to always set the **"Auto update when control changes"** setpoint to FALSE before saving a setpoint file, so that it all setpoints will be set as expected on the first upload.

1. OVERVIEW OF CONTROLLER

1.1. Description of Valve Controller

The Dual Input, Dual Output Valve Controller (2i2o) is designed for extremely versatile control of up to two proportional outputs to directly drive coils or other loads. Its flexible circuit design gives the user a wide range of configurable input or output types. The sophisticated control algorithms allow the user to program the controller for a wide range of applications without the need for custom software.

The controller has two fully programmable universal inputs that can be setup to read: voltage, current, resistive, frequency, or digital input signals. There are also two universal outputs that can be setup to drive: proportional current (up to 3A each); hotshot digital current; proportional voltage (up to supply); proportional PWM; or straight on/off digital loads. All I/O ports on the unit are independent from one another, unless specifically noted otherwise in this document.

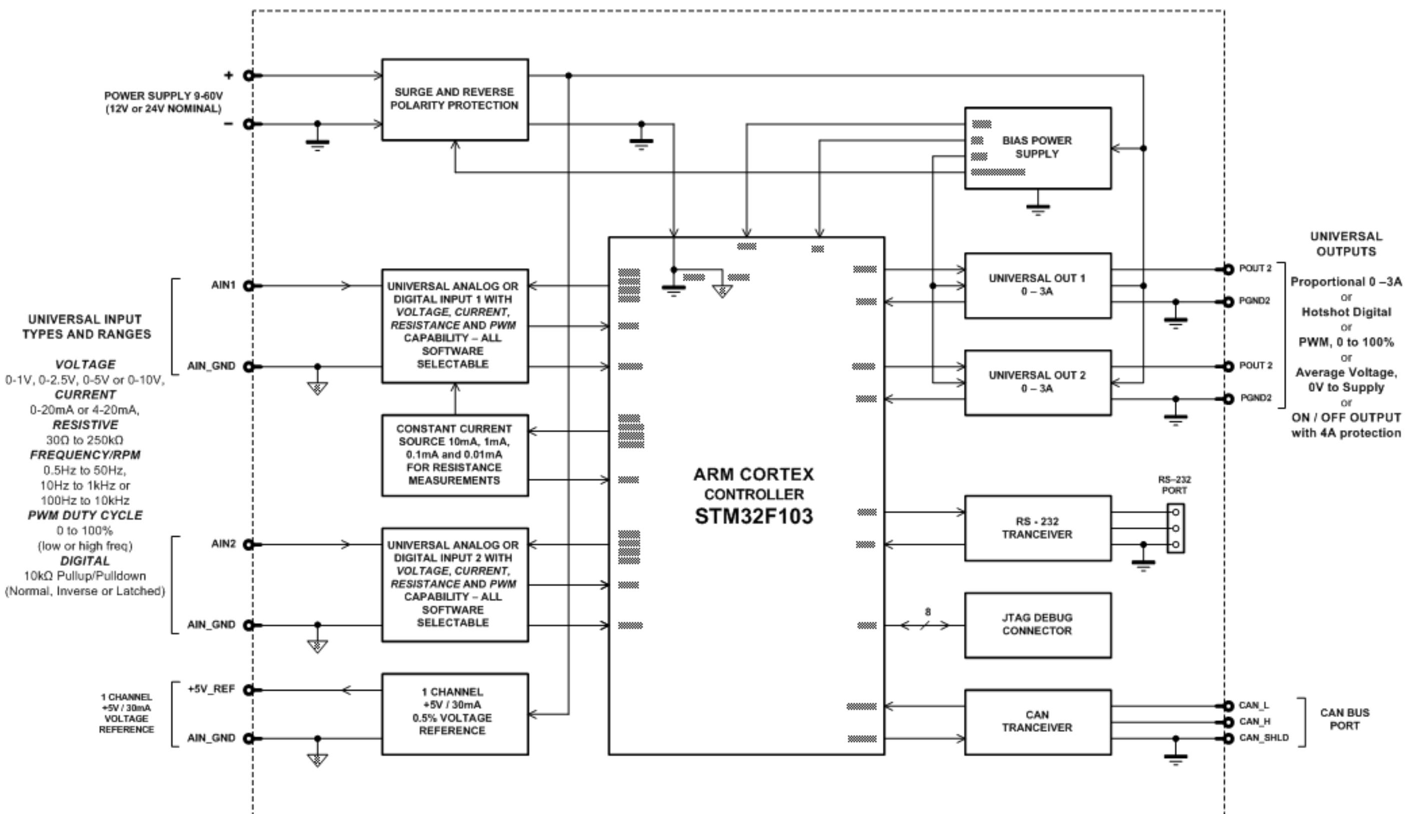


Figure 1A – Hardware Functional Block Diagram

The 2i2o is a highly programmable controller, allowing the user to configure it for their application. Its sophisticated control algorithms allow for open or closed loop drive of the proportional outputs. It can be operated as either a self-contained control system, driving the outputs directly from the on-board inputs, and/or it can be integrated into a CAN J1939 network of controllers. All I/O and logical function blocks on the unit are inherently independent from one another, but can be programmed to interact in a large number of ways. While Figure 1A shows the hardware features, Figure 1B shows the logical function blocks (software) available on the 2i2o.

The 2i2o has a number of built-in protection features that can shutoff the outputs in adverse conditions. These features are described in detail in section 1.9 Table 20 and they include hardware shutoffs to protect the circuits from being damaged as well as software shutdown features that can be enabled in safety critical systems.

The various function blocks supported by the 2i2o are outlined in the following sections. All setpoints are user configurable using the Axiomatic Electronic Assistant, as outlined in Section 4 of this document.

- Yellow bubbles represent setpoints that are unique to the function block in question
- Green bubbles represent the inputs to the function block
- Grey (light purple) bubbles represent outputs from the function block that can be mapped as inputs to other function blocks
- Blue bubbles represent another function block supported by the ECU which is intrinsically linked to the function block in question.
- Red bubbles represent outputs from the function block that are used internally in the controller by other function blocks, but are not accessible to the user.
- Dark purple bubbles represent messages that are sent to or received from the CAN bus.

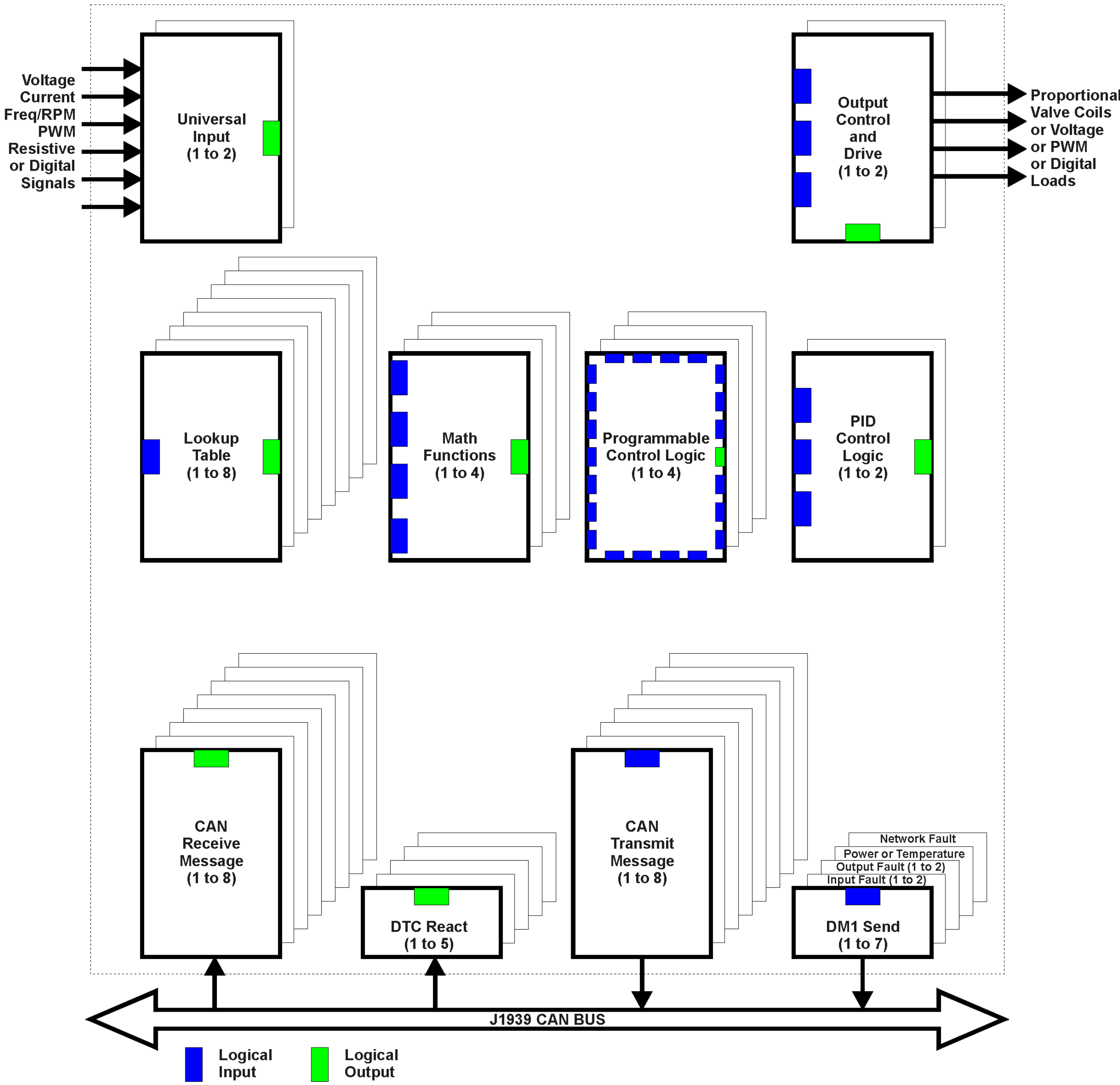


Figure 1B –Logical Functional Block Diagram

1.2. Input Function Blocks

Both inputs on the 2i2o controller are "universal" input types. ***However, since the timer used to measure PWM or Frequency types for Input 1 is also used by the analog-to-digital converter on the microcontroller, if one input has to be analog and the other digital, configure Input 2 as the digital type.*** Alternatively both inputs can be used to measure digital signals, in which case they are fully independent of one another.

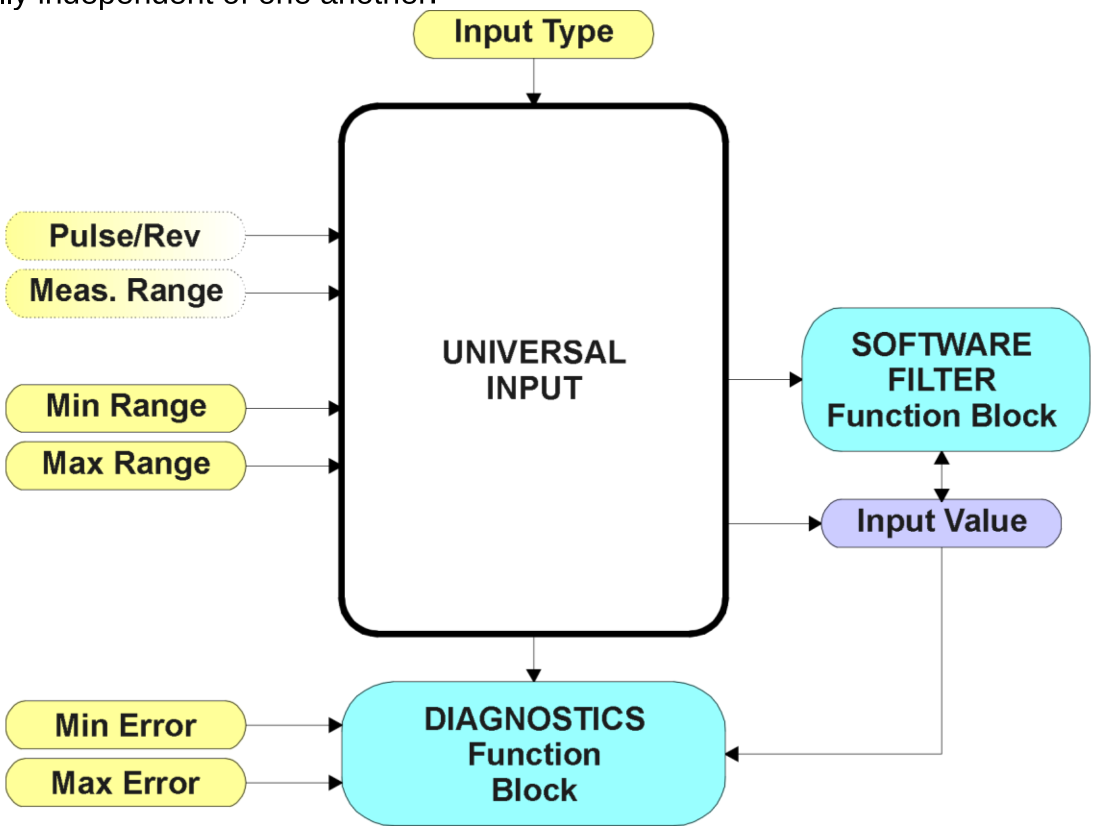


Figure 2 – Universal Input Function Block

The **"Input Sensor Type"** is the most important setpoint associated with this function block, and it should be selected first. Changing it will result in other setpoints being automatically updated to new default values by the 2i2o to match the new type.

0	*Disabled*
1	*Voltage (Default)*
2	*Current*
3	*Resistive*
4	*Freq/RPM*
5	*PWM*
6	*Digital*

Table 1 – Universal Input Type Options

If the input is used as a control input into another function block (i.e. PID, Table or Output Control), the corresponding function block will also be updated when the Input Type (or min/max) value is changed to reflect the new defaults if the **"Auto update when control changes"** setpoint in the Miscellaneous function block is set to *TRUE*.

Voltage (i.e. 0-5V) or Current (0-20mA) inputs have a hardware filter which then goes directly to a 12-bit analog-to-digital converter (ADC) on the processor. A voltage input is a high impedance (50kΩ) input protected against shorts to GND or Vcc. In current mode, a 250Ω resistor is used to measure the input signal. Analog Inputs should be connected to the AGND reference pin provided on the connector, per Table 27.

Frequency/RPM or Pulse Width Modulated (PWM) inputs are connected to 15-bit timer pins on the processor. The same pin can be used to reflect either an active low (connected to +5V through a 10kΩ pullup) or an active high (connected to GND through a 10kΩ pulldown) digital input.

Resistive inputs can accurately read a resistive value connected between the input pin and AGND. The 2i2o controller multiplexes a configurable current source (10uA, 100uA, 1mA or 10mA) to any pin setup as a resistive input, and measures the voltage created across the input. Depending on the value of the load on the input pin, it will self-calibrate to the appropriate sourcing current and adjust internal gain amplifiers. In this way, resistances from 30 Ω to 250 kΩ can be measured. Values outside of this range are automatically read as either a short or open circuit on the input.

The **"Pulse Per Revolution"** setpoint is only associated with the frequency input type. If a non-zero Pulse/Rev is selected, then the input data will be reported as in rotations-per-minute (RPM). Otherwise, frequencies inputs are measured in Hertz.

The **"Input Measuring Range"** setpoint is not used with resistive (self-calibrating) or digital inputs. For all other types, this setpoint is enabled to allow the user to optimize the hardware for the input signal.

For voltage inputs, there are actually four input ranges that can be selected. Typically, inputs will be in either the 0-5V or 0-10V range, but for more resolution at smaller voltages, a low range can be selected when appropriate.

0	*0 to 1V*
1	*0 to 2.5V*
2	*0 to 5V*
3	*0 to 10V*

Table 2 – Voltage Input Ranges

For current inputs, both ranges are the same at the hardware level. However, in 4-20mA, the minimum value cannot be set below 4mA.

0	*0 to 20mA*
1	*4 to 20mA*

Table 3 – Current Input Ranges

For frequency inputs, the user can select the following ranges shown in Table 4. Beside it, the overflow values are shown, showing how long it would take to flag an error once the signal is lost (i.e. no frequency measured.)

		Overflow [Sec]
0	*0.5Hz to 50Hz*	~4.000 (0.25Hz)
1	*10Hz to 1kHz*	~0.105 (9.5Hz)
2	*100Hz to 10kHz*	~0.010 (95Hz)

Table 4 – Frequency Input Ranges

The means by which the frequency is calculated is by measuring the time between pulses. The advantage of this type of measuring technique is that it gives an accurate reading of low frequency signals, with 2 decimal places of resolution. However, it has a drawback in that it will take several seconds to detect a loss of signal when the pulses go away, and may be read as a very high frequency signal during the transition phase. Therefore, it is highly recommended that diagnostics on low frequency inputs are not sent to the network for at least 5 seconds, to ensure that a low signal error is flagged, not a high error. If a signal will never go below 10Hz, it is highly recommended to use a more appropriate range for faster error reaction.

For PWM Inputs, there are only two types of measuring ranges, as shown below. To guarantee the rated accuracy of +/- 1% from 1% to 99%, ensure that the appropriate measuring range is selected.

0	*Low Freq (<1kHz)*
1	*High Freq (>100Hz)*

Table 5 – PWM Input Ranges

The **“Minimum Range”** and **“Maximum Range”** setpoints must not be confused with the measuring range. These setpoints are available with all but the digital input, and they are used when the input is selected as a control input for another function block. They become the Xmin and Xmax values used in the slope calculations (see Figure 6). When these values are changed, other function blocks using the input as a control source are automatically updated to reflect the new X-axis values.

The **“Minimum Error”** and **“Maximum Error”** setpoints are used with the Diagnostic function block. The values for these setpoints are constrained such that

*0 <= Minimum Error <= Minimum Range <= Maximum Range <= Maximum Error <= 1.1xMax**

* The maximum value for any input is dependent on type. The error range can be set up to 10% above this value.

Frequency: Max = 10,000 [Hz or RPM]
PWM: Max = 100.00 [%]
Voltage: Max = 1.00, 2.50, 5.00 or 10.00 [V]
Current: Max = 20.00 [mA]
Resistive Max = 250 [kΩ]

Lastly, the user can choose to use a Software Filter on the measured input prior to using it for diagnostic or control purposes.

More information about the Diagnostic or Software Filter function blocks can be found in Sections 1.9 and 1.8 respectively.

Since a Universal Input could be configured to act as an extra digital input, the Digital Input Function Block would take over when the **“Input Sensor Type”** is set to *‘Digital’*.

In contrast to the analog input types, the digital input function is very simple, as shown below.

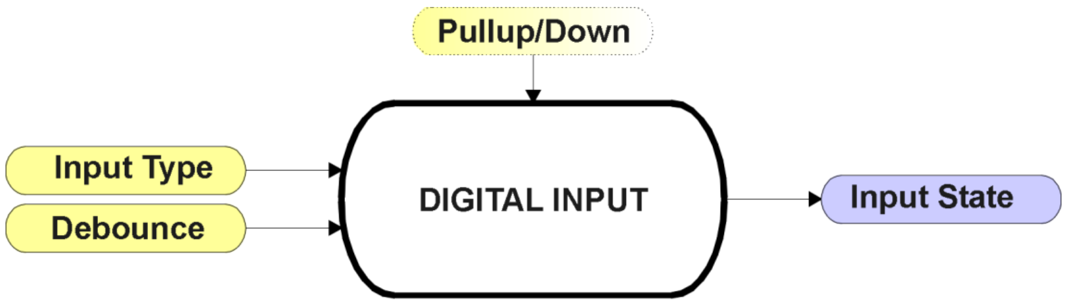


Figure 3 – Digital Input Function Block

The digital inputs connected to the Universal Inputs, Din1 to Din2, are set to active high by default. This means that when the input signal on the pin goes HIGH (>3V), the normal state response of the input is ON. When nothing or a low (GND) is connected to the pin, the input is OFF. With an active low input, the opposite is true, where a GND signal is considered to be ON, and open or a high is OFF.

To prevent spurious signals from affecting the logic of the controller, all digital inputs have associated with them a debounce time. If and only if the signal at the end of the debounce period has changed state will the change of state of the input be translated to the controller. The **"Debounce Time"** setpoint can be selected from a drop list, as shown in Table 6.

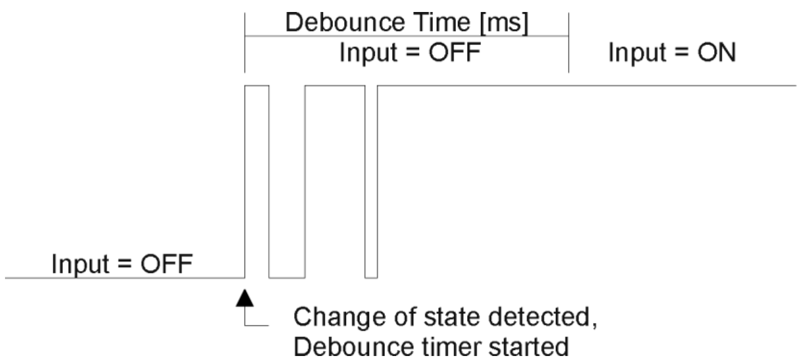


Figure 4 – Digital Input Debouncing

0	*None*
1	*111 ns*
2	*1.78 us*
3	*14.22 us*

Table 6 – Digital Debounce Filters

The **"Digital Input Type"** setpoint determines the Input State used by other function blocks based on the actual state of the active high input per the table below.

		Active High	Active Low	State
0	*Normal On/Off*	HIGH	LOW	ON
		LOW or Open	HIGH or Open	OFF
1	*Inverse Logic*	HIGH	LOW	OFF
		LOW or Open	HIGH or Open	ON
2	*Latched Logic*	HIGH to LOW	LOW to HIGH	No Change
		LOW to HIGH	HIGH to LOW	State Change (i.e. OFF to ON)

Table 7 – Digital Input Type versus Input State

1.3. Output Function Blocks

There are two types of output function blocks, one for proportional output drive and the other for output control. While both types of function blocks are found in the **"Output X Drive"** setpoint list, they are treated as separate function blocks here for ease of explanation.

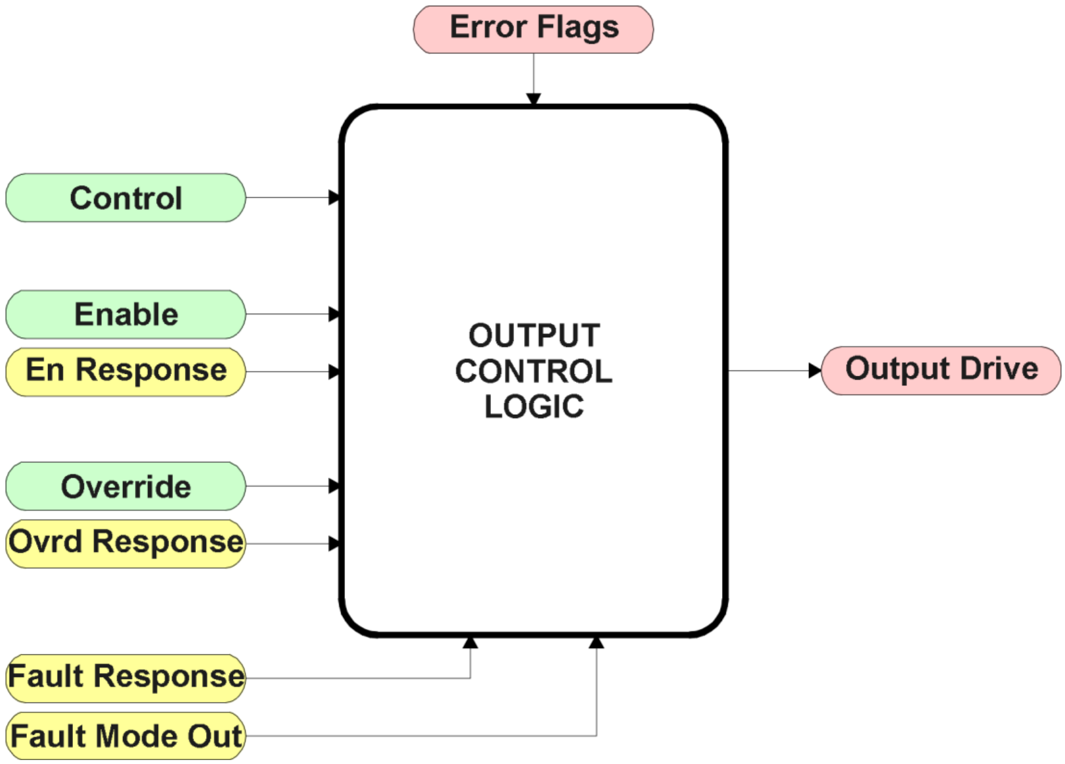


Figure 5 – Output Control Function Block

The Output Control Logic and the Output Drive function blocks are linked. The control signal, (whether an on-board input, a CAN command, a Lookup Table, or a Logic, Math or PID block) will have associate with it a minimum and maximum value. Regardless of what type of control input is selected, the output will always respond in a linear fashion to changes in the input per the calculation in Figure 6.

$$y = mx + a$$

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

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

Figure 6 – Linear Slope Calculations

In the case of the Output Control Logic function block, X and Y are defined as

Xmin = Control Input Minimum Ymin = **"Output at Minimum Command"**
Xmax = Control Input Maximum Ymax = **"Output at Maximum Command"**

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.

By default, the Output X **"Control Source"** is setup to be *'Universal Input X.'* In other words, all the proportional outputs will response in a linear fashion the corresponding input. (i.e. Input 1 controls Output 1) For a non-linear response, the Control should be mapped to *'Lookup Table X'* instead.

In addition to the Control input, the function block also supports an enable input which can be setup as either an enable or disable signal. If multiple enable/disable inputs are required, select the **"Enable Source"** to be the *'Math Function Block'*, and setup the desired logic response as described in Section 1.7.

When an Enable input is used, the output will be shutoff as per the **"Enable Response"**. If the response is selected as a disable signal (2 or 3), when the input is ON, the output will be shutoff.

0	*Enable When On, Else Shutoff*
1	*Enable When On, Else Rampoff*
2	*Enable When Off, Else Shutoff*
3	*Enable When Off, Else Rampoff*
4	*Enable When On, Else Ramp To Min*
5	*Enable When On, Else Ramp To Max*

Table 8 – Enable Response Options

A useful feature of the Enable input is that it can be mapped to react to a diagnostic trouble code (DTC) sent by another ECU on the network via a DM1. In this case, the **"Enable Source"** would be a *'Diagnostic Trouble Code (Rxd)'*, and the response would be setup as a disable signal.

The Override option allows the user to have the output drive go to a default value in the case of the override input being engaged/disengaged, depending on the logic selected in **"Override Response."** When active, the output will be driven to the value in **"Output at Override Command"** regardless of the value of the Control input.

The Enable and Override commands can be mapped to either a CAN message or any input. By default, though, the 2i2o assumes a digital input will be used for all of these inputs. If a non-digital control is selected, the input will be interpreted per the hysteresis logic shown in Figure 8.

If a fault is detected at any active input (Enable/Override/Control), then the output will respond per the **"Control Fault Response"** setpoint as outlined in Table 9. This is also true when controlled by a CAN message that is not received within the expected update timeframe (lost communication error). Should a fault response occur the output drive will remain in the corresponding state until the condition(s) causing the fault reaction has disappeared.

0	*Shutoff Output*
1	*Apply Fault Value*
2	*Hold Last Value*

Table 9 – Fault Response Options

Another fault response that can be enabled is that a power supply over voltage or under voltage will automatically disable ALL outputs. Note: this setpoint is associated with the **Power Supply Diag** function block. Also, if the **Over Temperature Diag** function block is enabled, then a microcontroller over-temperature reading disables all the outputs until it has cooled back to the operating range.

Fault conditions are checked for first, and only if they are not present are the control signal then evaluated. If Enable, Override and Control inputs are all used, the Enable logic is evaluated first, then the Override, and lastly the Control. The logic flow chart for evaluating the output drive is shown in Figure 7 below.

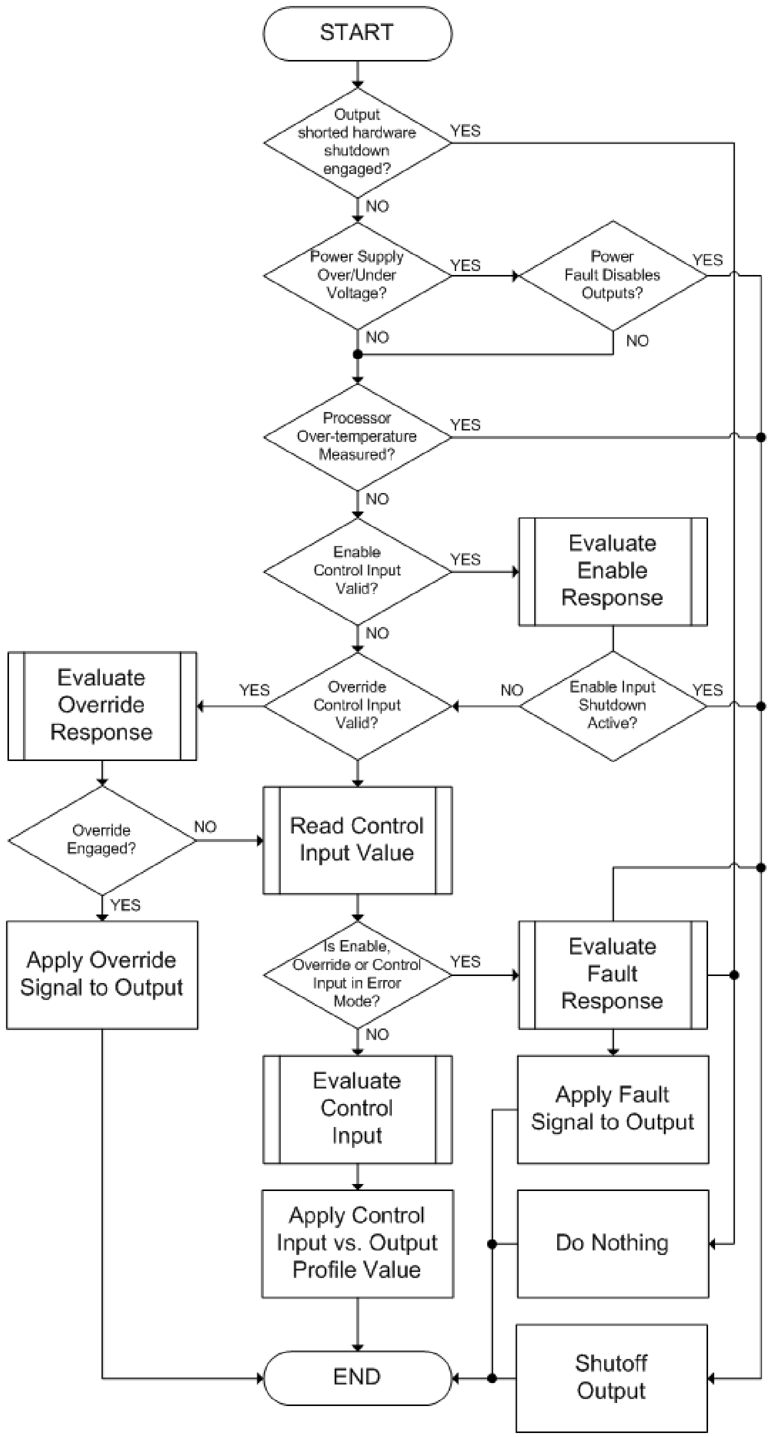


Figure 7 – Output Logic Flowchart

The other type of function block associated with an output is the drive block itself, as shown in Figure 8.

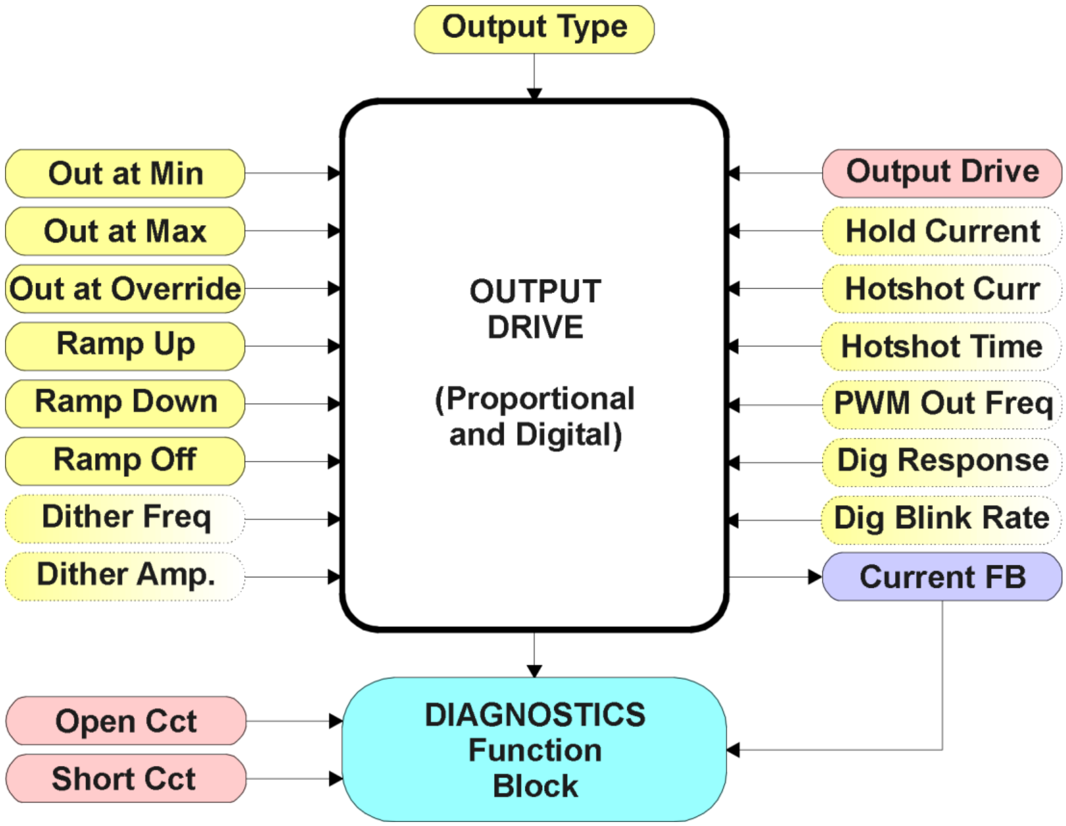


Figure 8 – Output Drive Function Block

The **"Output Type"** is the most important setpoint associated with this function block, and it should be selected first. Changing it will result in other setpoints being automatically updated by the 2i2o to match the new type. There are several output types supported, as outlined in Table 10.

0	*Disabled*
1	*Proportional Current*
2	*Digital Hotshot*
3	*PWM Duty Cycle*
4	*Proportional Voltage*
5	*Digital On/Off*

Table 10 – Output Type Options

For the *'Proportional Current'* or *'Hotshot Digital'* types, the unit for the output values is mA, and the Diagnostic function block, related to the current feedback, is used. For all other output types, the output fault detection/reaction setpoints are ignored.

Any of the proportional responses (current, duty cycle or voltage) react to the Control input in a linear fashion per the relationship in Figure 6.

In order to prevent abrupt changes at the output due to sudden changes in the command input, the user can choose to use the independent up or down ramps to smooth out the coil's response. The "**Ramp Up**" and "**Ramp Down**" setpoints are in milliseconds, and the step size of the output change will be determined by taking the absolute value of the output range and dividing it by the ramp time.

If the output is being controlled by a Lookup Table function block in the "Time Response" mode, the user should disable the output drive ramps (set them to zero) as the table will already ramp the command before it is applied to the output drive.

The *'Proportional Current'* type has associated with it two setpoints not used by the other types, which are the **"Dither Frequency"** and **"Dither Amplitude"** values. The output is controlled by a high frequency signal (25kHz), with the low frequency dither superimposed on top. ***The same dither frequency is applied to both outputs.***

The dither frequency will match exactly what is programmed into the setpoint, but the exact amplitude of the dither will depend on the properties of the coil. When adjusting the dither amplitude value, select one that is high enough to ensure an immediate response to the coil to small changes in the control inputs, but not so large as to effect the accuracy or stability of the output. Refer to the coil's datasheet for more information.

The *'Proportional Voltage'* type uses the measured value of the power supply, and adjusts the duty cycle of the output such that the average value will match the target output voltage. Since the output is normally running at a high frequency (25kHz), the voltage can be easily average using a simple low pass filter.

Normally, the *'PWM Duty Cycle'* option allows the user to run the output at a fixed frequency (i.e. 500Hz) while the duty cycle changes depending on the Control input. However, since the PWM peripheral for the drive is used by both outputs, the output frequency for this output type remains fixed at 25kHz if either output is setup for current control. In this case, there is not much difference between the Voltage or PWM outputs in the product.

If and only if both outputs are setup for PWM, Voltage or Digital On/Off will the setpoint **"PWM Output Frequency"** on be selectable in the Output 1 Drive group. ***In all case, the same output frequency is applied to both outputs.***

Another feature in PWM output mode is on Output 1, the setpoint **"PWM Outputs Linked"** can be set to *TRUE*. In this case, Output 2 automatically gets set as PWM output type, and it will be driven with the inverse duty cycled as that applied to Output 1. In this mode, the outputs are no longer independent, but rather become mirror images of one another, as shown in Figure 9.

WARNING: These outputs are not a true H-bridge drive, so there is no deadband between turning off one output, and engaging the next. Therefore, this mode must not be used in any application where the load will short if both outputs are on at the same time.

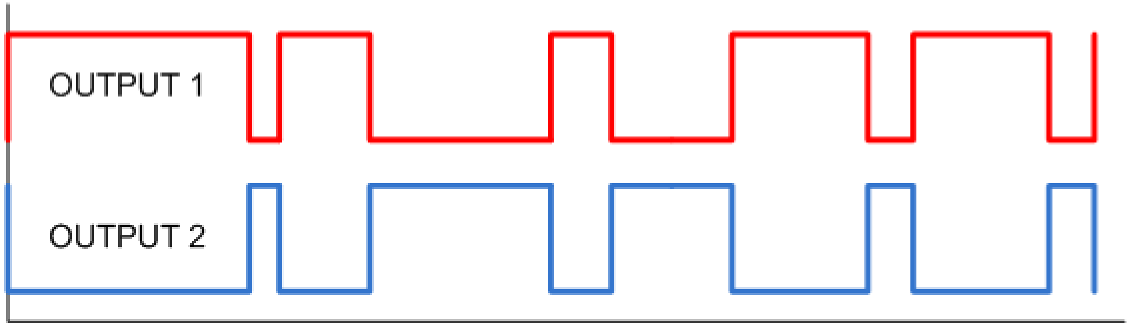


Figure 9 – PWM Outputs with Linked Response

Instead of proportional, there are also two types of digital responses possible as well. With the *'Digital On/Off'* type, should the control require the output to be on, it will be turned on at whatever the system power supply is. The output will source whatever current is required by the load, up to 3A.

If a non-digital control is selected, 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 10. This relationship is true for any function block that has a non-digital input mapped to a digital control.

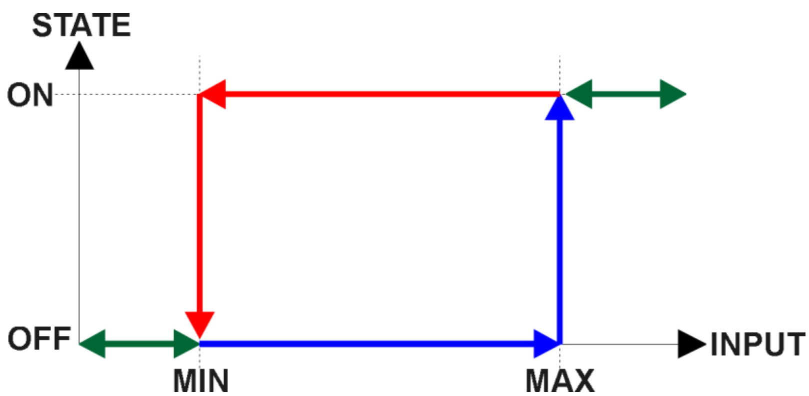


Figure 10 – Analog to Digital Input

If a digital **"Output Type"** has been selected the **"Digital Response"** setpoint will be enabled as shown in Table 11. Also, the **"Digital Override State"** setpoint will be enabled, and will follow the Override input the same way that the **"Output at Override Command"** would be used with a proportional output response.

0	*Normal On/Off*
1	*Inverse Logic*
2	*Latched Logic*
3	*Blinking Logic*

Table 11 – Digital Response Options

In a *'Normal'* response, when the Control input commands the output ON, then the output will be turned ON. However, in an *'Inverse'* response, the output will be ON unless the input commands the output ON, in which case it turns OFF.

If a *'Latched'* response is selected, when the input commands the state from OFF to ON, the output will change state.

If a *'Blinking'* response is selected, then while the input command the output ON, it will blink at the rate in the **"Digital Blink Rate"** setpoint. When commanded OFF, the output will stay off. A blinking response is only available with a *'Digital On/Off'* type of output (not a Hotshot type.)

The *'Hotshot Digital'* type is different from in simple *'Digital On/Off'* in that it still controls the current through the load. This type of output is used to turn on a coil then reduce the current so that the valve will remain open, as shown in Figure 11 below. Since less energy is used to keep the output engaged, this type of response is very useful to improve overall system efficiency.

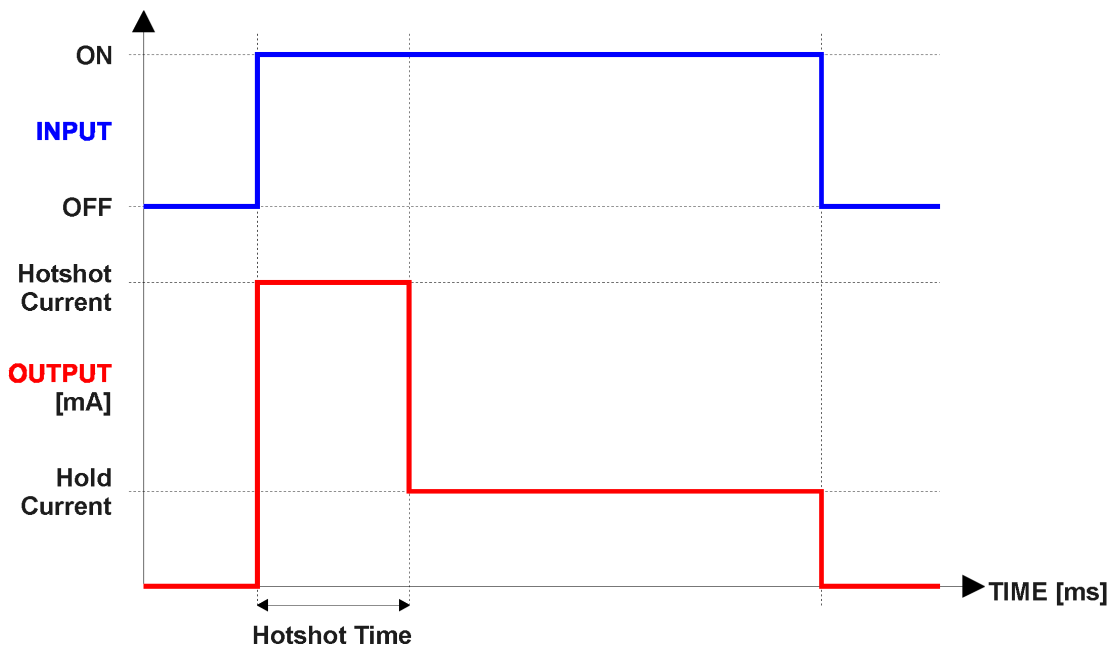


Figure 11 – Hotshot Digital Profile

<u>Open or Short Circuits</u>

The outputs are inherently protected against a short to GND or +Vps by circuitry. In the case of a dead short, the hardware will automatically disable the output drive, regardless of what the processor is commanding for the output. When this happens, the processor will go through a brief sub-routine to determine which output is shorted. It will then continue to drive the non-shorted output normally, and periodically (every 5 seconds) try to re-engage the short load, if still commanded to do so. If the fault has gone away since the last time the output was engaged while shorted, the controller will automatically resume normal operation.

In the case of an open circuit, there will be no interruption of the control for either output. The processor will continue to attempt to drive the open load based on the logic shown in Figure 7.

The measured current through the load is available to be broadcasted on a CAN message if desired. It is also used as the input to the diagnostic function block for each output, and an open or shorted output can be broadcasted in a DM1 message on the CAN network. See Section 1.9 for the complete description of the diagnostic functions available on this controller.

1.4. PID Control Function Block

The PID Control function block is an independent logic block, but it is normally intended to be associated with proportional output control blocks described earlier. When the **"Control Source"** for an output has been setup as a *'PID Function Block'*, the command from the selected PID block drives the physical output on the 2i2o.

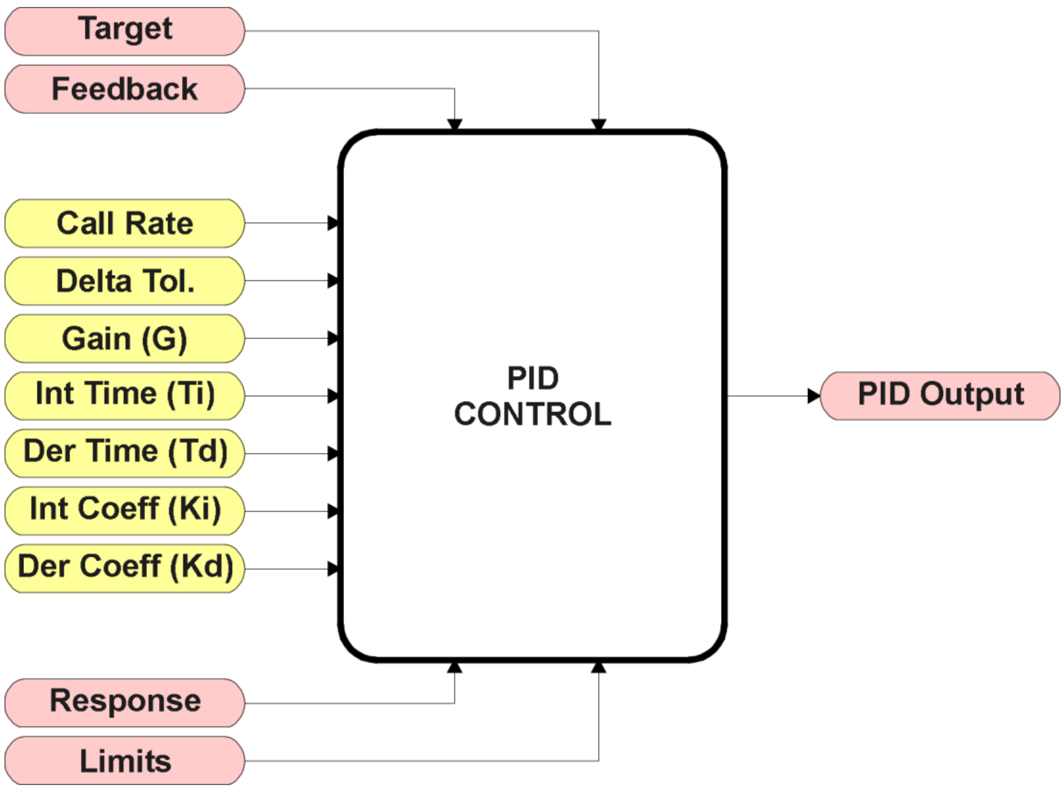


Figure 12 – PID Control Function Block

To use the PID calculations, the **"PID Target Command Source"** determines the control input, and the **"PID Feedback Input Source"** established the feedback signal to the PID function block. The **"PID Response Profile"** will use the selected inputs as per the options listed in Table 12. When active, the PID algorithm will be called every "**PID Loop Update Rate**" (in ms)

0	*Single Output*
1	*Setpoint Control*
2	*On When Over Target*
3	*On When Below Target*

Table 12 – PID Responses

When a *'Single Output'* response is selected, the Target and Feedback inputs do not have to share the same units. In both cases, the signals are converted to a percentage value based on the minimum and maximum values associated with the source function block.

For example, a CAN command could be used to set the target value, in which case it would be converted to a percentage value using the **"Received Data Min"** and **"Received Data Max"** setpoints in the appropriate *'CAN Receive X'* function block. The close-loop feedback signal (i.e. a

0-5V input) could be connected to *'Universal Input 1'* and selected as the feedback source. In this case the value of the input would be converted to a percentage based on the **"Minimum Range"** and **"Maximum Range"** setpoints in the input block. The output of the PID function would depend on the difference between the commanded target and the measured feedback as a percentage of each signals range. In this mode, the output of the block would be a value from -100% to 100%.

When a *'Setpoint Control'* response is selected, the **"PID Target Command Source"** automatically gets updated to *'Control Constant Data'* and cannot be changed. The value set in the associated constant in the Constant Data List function block becomes the desired target value. In this case, both the target and feedback values are assumed to be in the same units and range. The minimum and maximum values for the feedback automatically become the constraints on the constant target. In this mode, the output of the block would be a value from 0% to 100%.

For example, if the feedback was setup as a 4-20mA input, a "Constant Value X" setpoint set to 14.2 would automatically be converted to 63.75%. The PID function would adjust the output as needed to have the measured feedback to maintain that target value.

The last two response options, '*On When Over Target*' and '*On When Below Target'*, are designed to allow the user to combine the two proportional outputs as a push-pull drive for a system. Both outputs must be setup to use the same control input (linear response) and feedback signal in order to get the expected output response. In this mode, the output would be between 0% to 100%.

In order to allow the output to stabilize, the user can select a non-zero value for "**PID Delta Tolerance**". If the absolute value of $Error_K$ is less than this value, $Error_K$ in the formula below will be set to zero.

The PID algorithm used is shown below, with names in red being the configurable parameters.

```
T = Loop_Update_Rate*0.001

P_Gain = G
I_Gain = G*Ki*T/Ti
D_Gain = G*Kd*Td/T
Note: If Ti is zero, I_Gain = 0

Error_k     = Target – Feedback
ErrorSum_k  = ErrorSum_k-1 + Error_k

P_k = Error_k * P_Gain

I_k = ErrorSum_k * I_Gain

D_k = (Error_k-Error_k-1) * D_Gain

PIDOutput_k = P_k + I_k + D_k
```

Figure 13 – PID Control Algorithm

Each system will have to be tuned for the optimum output response. Response times, overshoots and other variables will have to be decided by the customer using an appropriate PID tuning strategy. Axiomatic is not responsible for tuning the control system.

1.5. Lookup Table Function Block

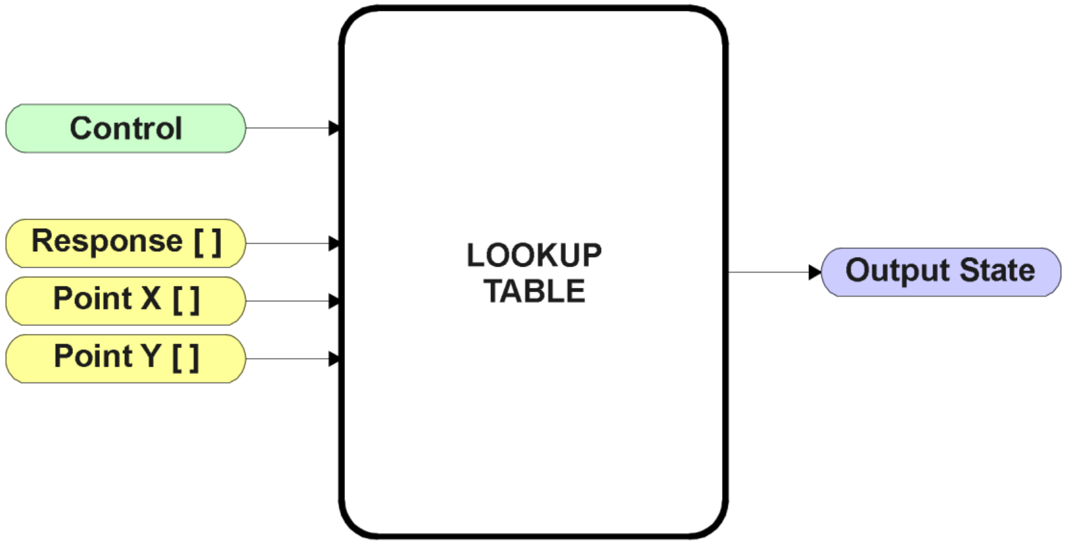


Figure 14 – Lookup Table Function Block

Lookup Tables are used to give an output response of up to 10 slopes per input. The array size of the Response [ ], Point X [ ] and Point Y [ ] setpoints 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.

There are two key setpoints that will affect this function block. The first is the "**X-Axis Source**" and **"X-Axis Number"** which together define the Control Source for the function block. When it is changed, the table is automatically updated with new defaults based on the X-Axis source selected if "**Auto update when control changes"** in the Miscellaneous block is *TRUE.*

As stated earlier if **Auto update when control changes"** is *TRUE,* should the selected Control Source change (i.e. the Min or Max values of the function block are updated), the associated table will also be automatically updated with default settings, based on the new X-Axis limits.

> Initialize the Control Source of a Lookup Table BEFORE changing the table values, as the new settings WILL get erased when the control is updated if the **"Auto update when control changes"** in the Miscellaneous function block is set to *TRUE*.

The second setpoint that will affect the function block (i.e. reset to defaults), is the "**X-Axis Type**". By default, the tables have a *'Data Response'* output. Alternatively, it can be selected as a *'Time Response',* which is described later in Section 1.5.5.

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.

For example, if the control source is a Universal Input, setup as a 0-5V type, with an operating range of 0.5V to 4.5V, the X-Axis will be setup to have a default “**Point 1 – X Value**” of 0.5V, and setpoint “**Point 10 – X Value**” will be set to 4.5V. The “**Point 0 – X Value**” will be set to the default value of 0.0V.

For most *‘Data Responses’*, the default value at point (0,0) is [0,0].

However, should the minimum input be less than zero, for example a CAN message that is reflecting temperature in the range of -40ºC to 210ºC, then the “**Point 0 – X Value**” 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_{10} is changed first, then lower indexes in descending order.

$$MinInputRange <= X_0 <= X_1 <= X_2 <= X_3 <= X_4 <= X_5 <= X_6 <= X_7 <= X_8 <= X_9 <= X_{10} <= MaxInputRange$$

As stated earlier, MinInputRange and MaxInputRange will be determined by the X-Axis Source that has been selected.

If some of the data points are *‘Ignored’* as described in Section 1.5.4, they will not be used in the X-Axis calculation shown above. For example, if points X_4 and higher are ignored, the formula becomes MinInputRange <= X_0 <= X_1 <= X_2<= X_3<= MaxInputRange instead.

1.5.2. Y-Axis, Lookup Table Output

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.

For example, should the X-Axis of a table be a resistive value (as read from a Universal Input), the output of the table could be temperature from an NTC sensor in the range Y_0=125ºC to Y_{10}= -20ºC. If this table is used as the control source for another function block (i.e. transmitted over CAN), then Xmin would be -20 and Xmax would be 125 when used the linear formula shown in Figure 6.

In all cases, the controller looks at the **entire range** of the data in the Y-Axis setpoints, and selects the lowest value as the MinOutRange and the highest value as the MaxOutRange. They are passed directly to other function blocks as the limits on the Lookup Table output. (i.e used as Xmin and Xmax values in linear calculations.)

However, if some of the data points are *‘Ignored’* as described in Section 1.5.4, they will not be used in the Y-Axis range determination. Only the Y-Axis values shown on the Axiomatic EA will be considered when establishing the limits of the table when it is used to drive another function block, such as an Output Drive.

1.5.3. Default Configuration, Data Response

By default, all Lookup Tables in the 2i2o are disabled (**"X-Axis Source"** equals *'Control Source Not Used'.*) If they were to use the default settings for Inputs 1 and 2 instead as the X-Axis, and output current (in mA) they could be used to control the corresponding Output Drives. If a non-linear response for one or more of the outputs is required, the user can easily use the table(s) to create the desired response profiles.

Recall, any controlled function block which uses the Lookup Table as an input source (not only the Output Drive) will also apply a linearization to the data. **Therefore, for a 1:1 control response, ensure that the minimum and maximum values of the output (Ymin and Ymax in Figure 6) correspond to the minimum and maximum values of the table's Y-Axis (Xmin and Xmax in Figure 6).**

To control "Output N Drive" by "Universal Input N" modified by "Lookup Table N" (N = 1 to 2) it is recommended to do so in the following order:

a) Change Output Drive **"Output at Minimum Command"** and **"Output at Maximum Command"** to the desired limits.
b) Set "**Auto update when control changes"** in the Miscellaneous block to *TRUE* (optional)
c) Configure the desired Control Source (i.e. Universal Input, or CAN Receive Message), and set the appropriate limits. (Note: If the Control Source is the Universal Input, the table will automatically be updated with new defaults when the **"Input Sensor Type"**, **"Minimum Range"** or **"Maximum Range"** setpoints are updated)
d) Change the Lookup Table X **"X-Axis Source"** setpoints. (If applicable)
   At this point, the X-Axis limits will match the control source, and the Y-Axis limits would correspond to the Output Drive range, as a percentage.
e) Update the X and Y setpoints for the application

Note: Order (c) to (e) holds true for all configuration done using any Lookup Table function block.

In all cases, should an **"X-Axis Source"** be selected, the Y-Axis defaults will be in the range of 0 to 100% as described in the "Y-Axis, Lookup Table Output" section above. X-Axis minimum and maximum defaults will be set as described in the "X-Axis, Data Response" section above.

By default, the X and Y axes data is setup for an equal value between each point from the minimum to maximum in each case.

For example, with a 0.5 to 4.5V input (X-Axis) driving a 0 to 1500mA output (Y-Axis), the default points would be setup as per figure (a) below. However, the 100Ω to 200kΩ input (X-Axis) representing 125ºC to -20ºC (Y-Axis) would be given defaults as per figure (b) below. In each case, the user would have to adjust the table for the desired response.

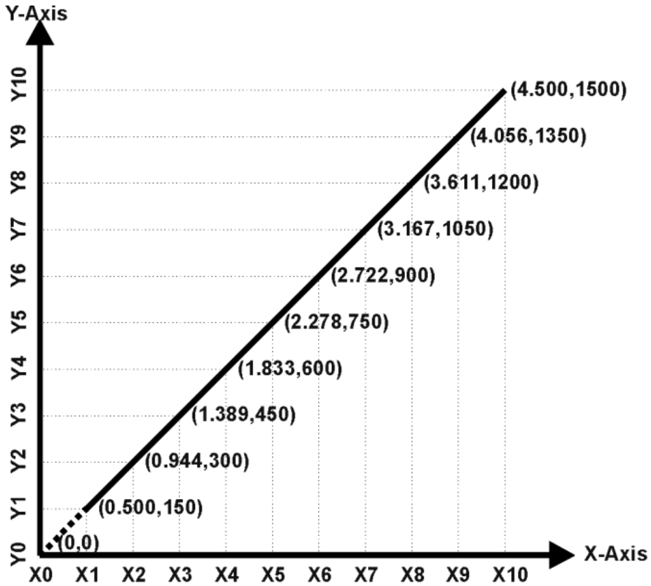


Figure A - 0.5 to 4.5V Input, 0 to 1500mA Output

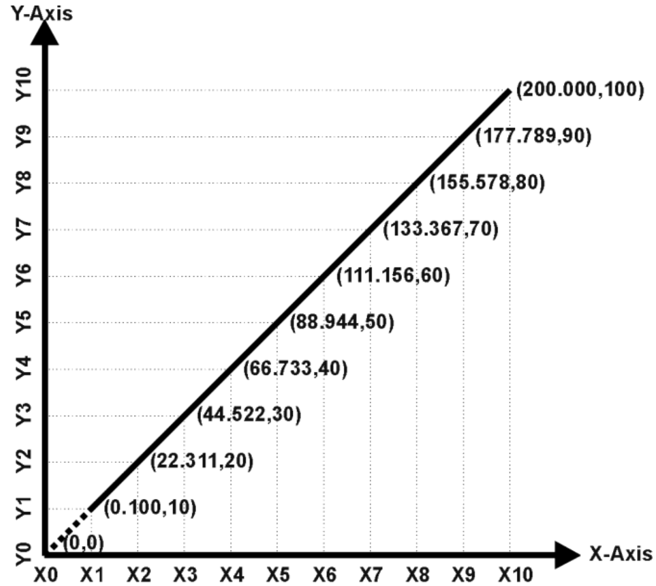


Figure B - 0.1 to 200kOhm Input, 0 to 100% Output

Figure 15 – Lookup Table Default Initialization Examples

1.5.4. Point To Point Response

By default, the X and Y axes are setup for a linear response from point (0,0) to (10,10), where the output will use linearization between each point, as shown in Figure 15. To get the linearization, each "**Point N – Response**", where N = 1 to 10, is setup for a *'Ramp To'* output response.

Alternatively, the user could select a *'Jump To'* response for "**Point N – Response**", where N = 1 to 10. 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.

An example of a CAN message (0 to 100) used to control a default table (0 to 100) but with a *'Jump To'* response instead of the default *'Ramp To'* is shown in Figure 16.

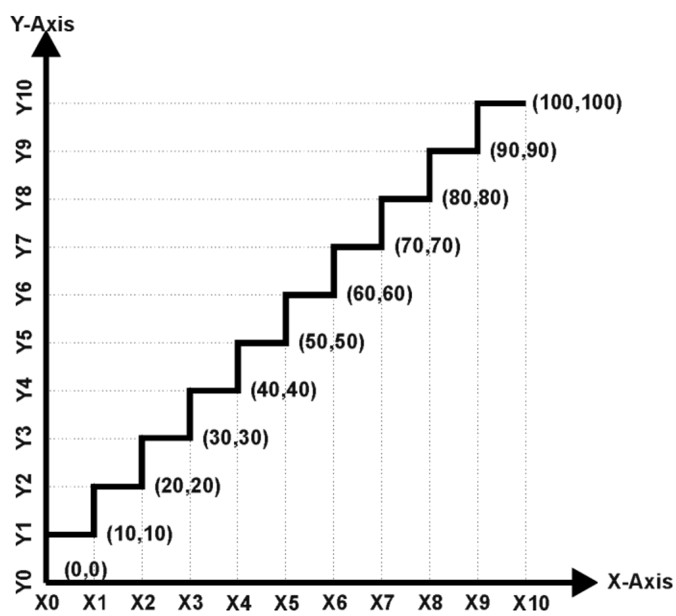


Figure 16 – Lookup Table "Jump To" Data Response

Lastly, any point except (0,0) can be selected for an *'Ignore'* response. If "**Point N – Response**" is set to ignore, then all points from (X_N, Y_N) to (X_{10}, Y_{10}) 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 (i.e. a CAN Message) 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 17. The example shows a dual slope output response for each side of the deadband, but additional slopes can be easily added as needed.

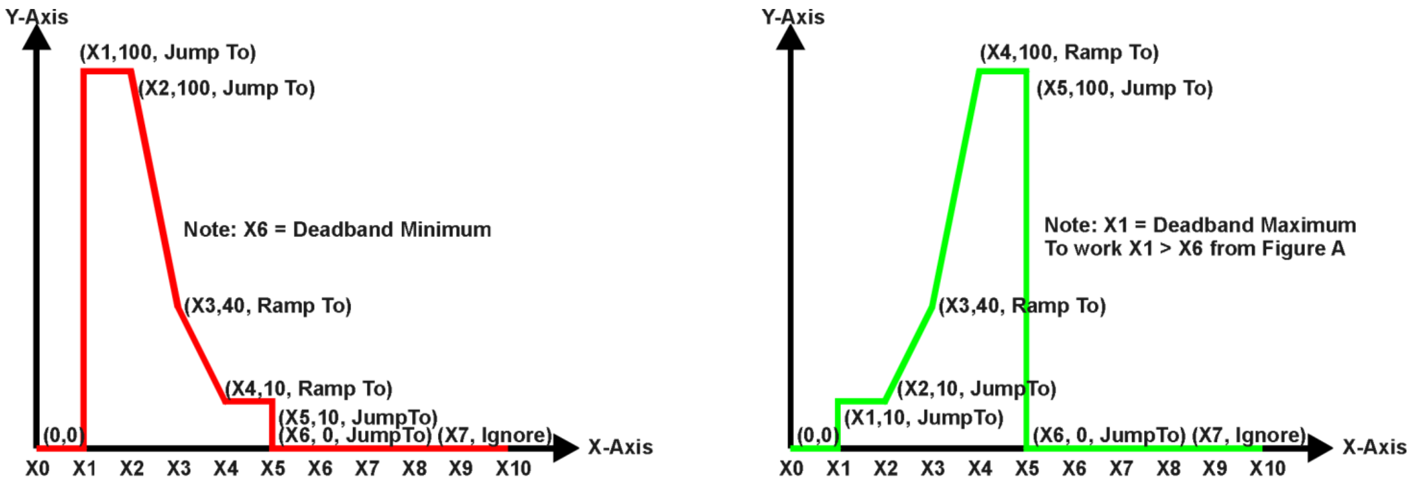


Figure A - Dual Slope "Profile B" Joystick Deadband Response

Figure B - Dual Slope "Profile A" Joystick Deadband Response

Figure 17 – Lookup Table Examples to Setup for Joystick Deadband Response

1.5.5. X-Axis, Time Response

As mentioned in Section 1.5, 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.

In this case, the **"X-Axis Source"** is treated as a digital input. If the signal is actually an analog input, it is interpreted like a digital input per Figure 10. 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. index 10, or *'Ignored'* response), the output will remain at the last output at the end of the profile until the control input turns OFF.

When the control input is OFF, the output is always at zero. When the input comes ON, the profile ALWAYS starts at position (X_0, Y_0) which is 0 output for 0ms.

When using the Lookup Table to drive an output based on **time**, it is mandatory that setpoints **"Ramp Up (min to max)"** and **"Ramp Down (max to min)"** in the Output X Drive function block be set to **zero**. Otherwise, the output result will not match the profile as expected. Recall, also, that the Y-Axis range of the table should be set to match the Output Drive range in order to get a 1:1 response of table output versus drive output.

An application where this feature would be useful is filling a clutch when a transmission is engaged. An example of some fill profiles is shown in Figure 18.

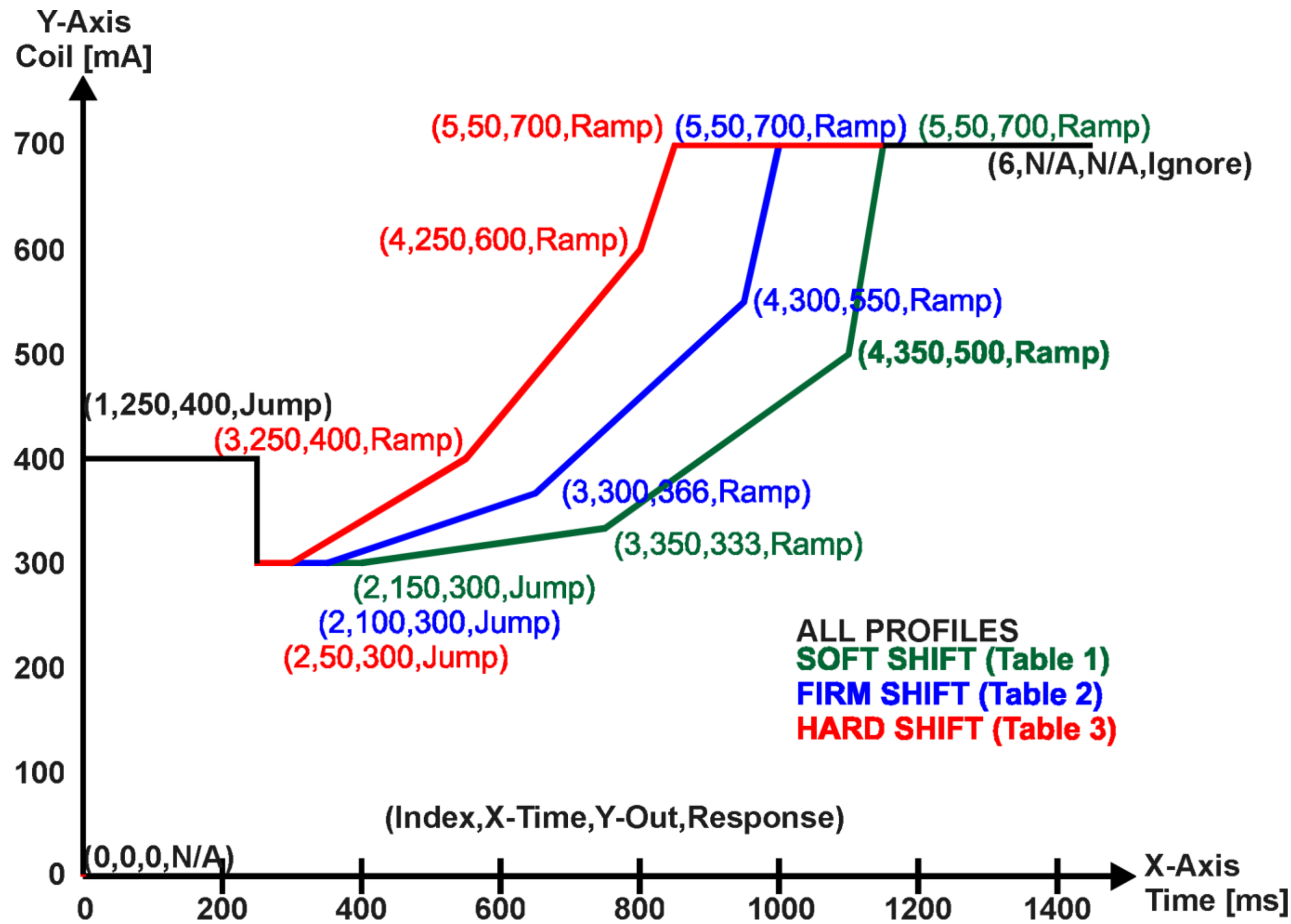
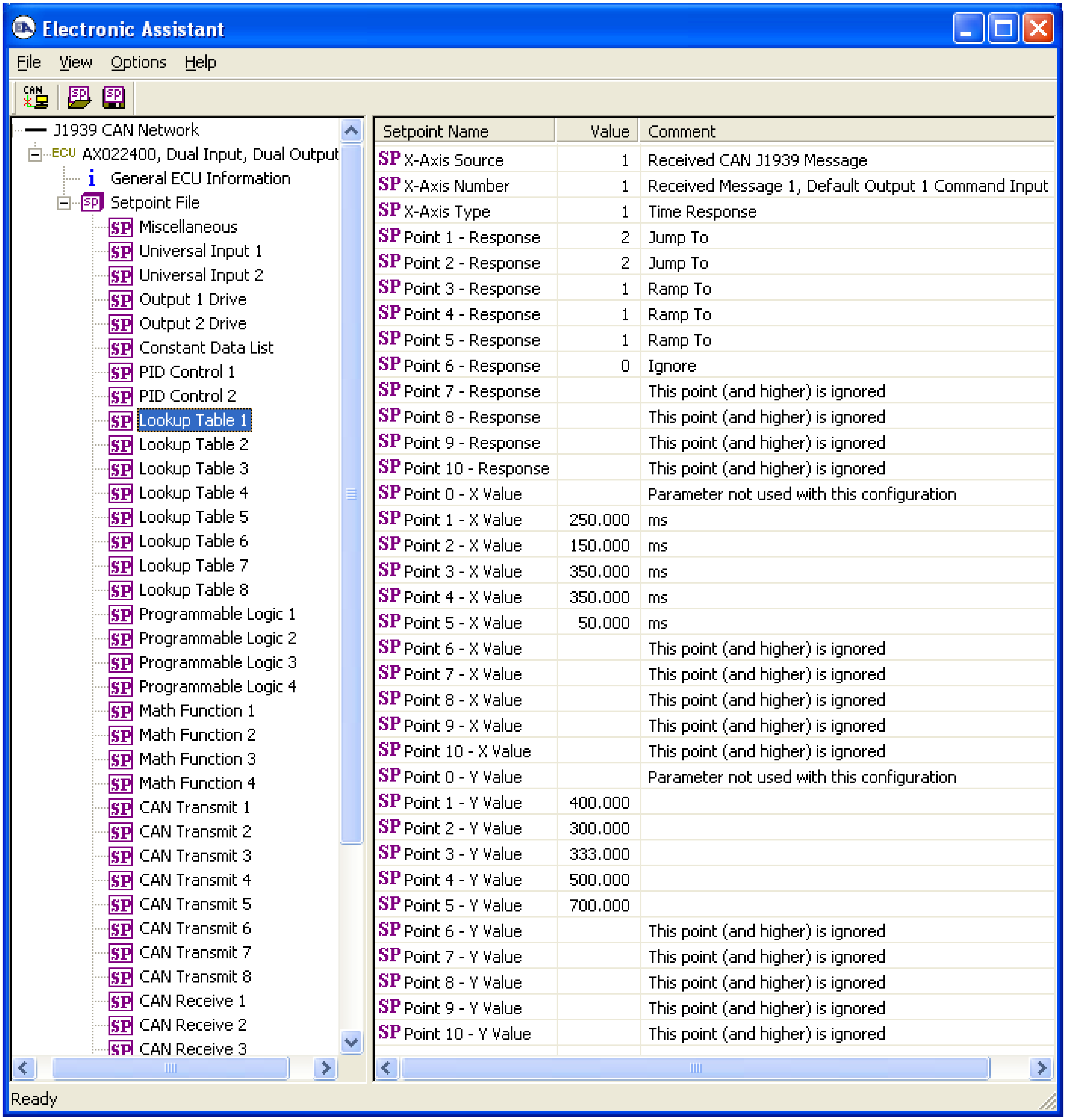


Figure 18 – Lookup Table Time Response Clutch Fill Profiles

In a time response, the interval time between each point on the X-axis can be set anywhere from 1ms to 24 hours. [86,400,000 ms]

One final note about the Lookup Tables is that if a digital input is selected as the control source for the X-Axis, only a 0 (Off) or 1 (On) will be measured. Ensure that the data range for the X-Axis on the table is updated appropriately in this condition.



Setpoint Name	Value	Comment
X-Axis Source	1	Received CAN J1939 Message
X-Axis Number	1	Received Message 1, Default Output 1 Command Input
X-Axis Type	1	Time Response
Point 1 - Response	2	Jump To
Point 2 - Response	2	Jump To
Point 3 - Response	1	Ramp To
Point 4 - Response	1	Ramp To
Point 5 - Response	1	Ramp To
Point 6 - Response	0	Ignore
Point 7 - Response		This point (and higher) is ignored
Point 8 - Response		This point (and higher) is ignored
Point 9 - Response		This point (and higher) is ignored
Point 10 - Response		This point (and higher) is ignored
Point 0 - X Value		Parameter not used with this configuration
Point 1 - X Value	250.000	ms
Point 2 - X Value	150.000	ms
Point 3 - X Value	350.000	ms
Point 4 - X Value	350.000	ms
Point 5 - X Value	50.000	ms
Point 6 - X Value		This point (and higher) is ignored
Point 7 - X Value		This point (and higher) is ignored
Point 8 - X Value		This point (and higher) is ignored
Point 9 - X Value		This point (and higher) is ignored
Point 10 - X Value		This point (and higher) is ignored
Point 0 - Y Value		Parameter not used with this configuration
Point 1 - Y Value	400.000	
Point 2 - Y Value	300.000	
Point 3 - Y Value	333.000	
Point 4 - Y Value	500.000	
Point 5 - Y Value	700.000	
Point 6 - Y Value		This point (and higher) is ignored
Point 7 - Y Value		This point (and higher) is ignored
Point 8 - Y Value		This point (and higher) is ignored
Point 9 - Y Value		This point (and higher) is ignored
Point 10 - Y Value		This point (and higher) is ignored

Figure 19 – Lookup Table “Soft Shift” Axiomatic EA Configuration

1.6. Programmable Logic Function Block

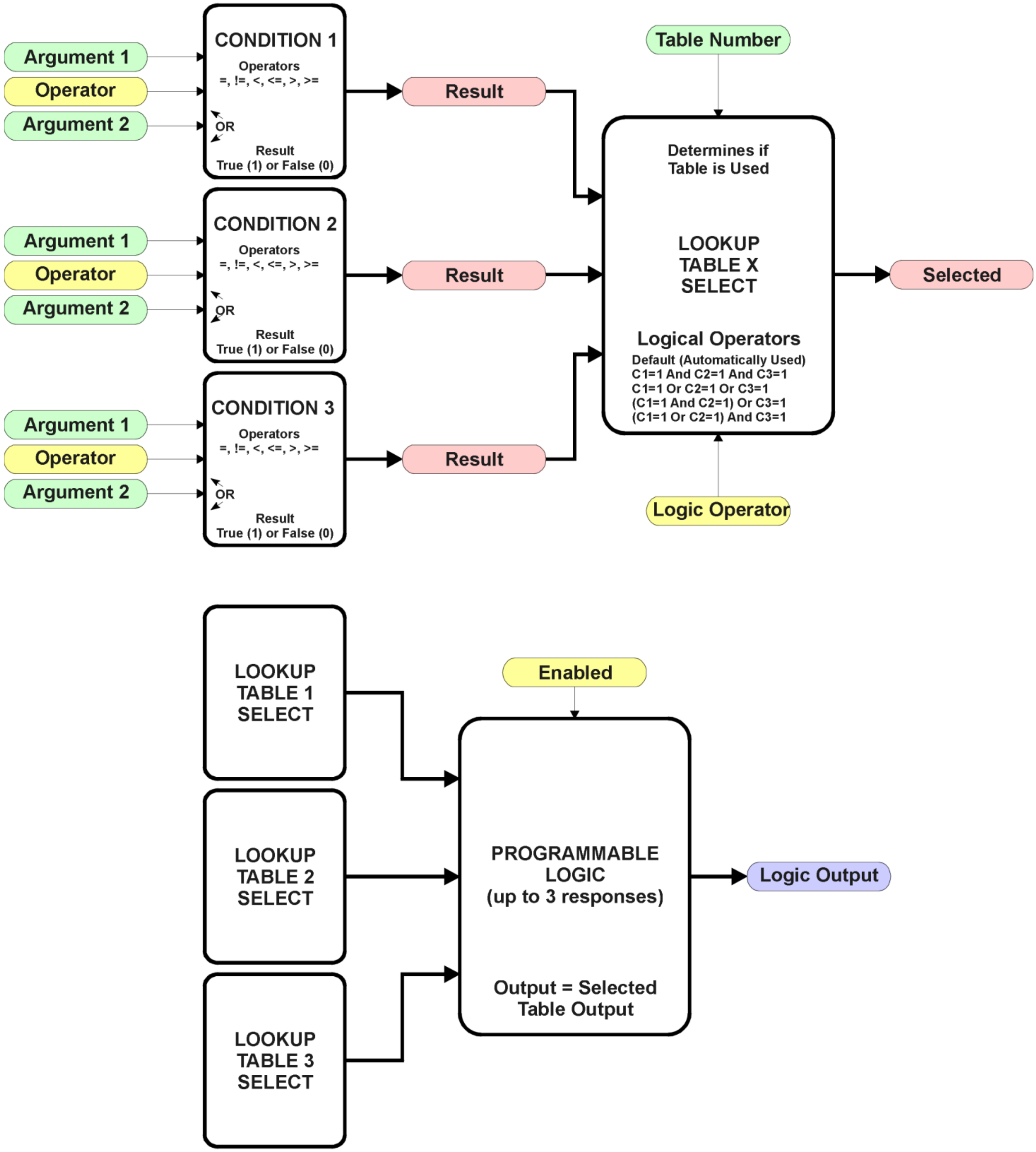


Figure 20 – Programmable Logic Function Block

This function block is obviously the most complicated of them all, but very powerful. The Programmable Logic can be linked to up to three tables, any one of which would be selected only under given conditions. Any three tables (of the available 16) can be associated with the logic, and which ones are used is fully configurable.

Should the conditions be such that a particular table (1, 2 or 3) 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 the Logic Output.

Therefore, up to three different responses to the same input, or three different responses to different inputs, can become the input to another function block, such as an Output X Drive. To do this, the "**Control Source**" for the reactive block would be selected to be the *'Programmable Logic Function Block.'*

In order to enable any one of Programmable Logic blocks, the "**Programmable Logic Block Enabled**" setpoint must be set to True. They are all disabled by default.

Logic is evaluated in the order shown in Figure 21. Only if a lower number table 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 number in any configuration.**

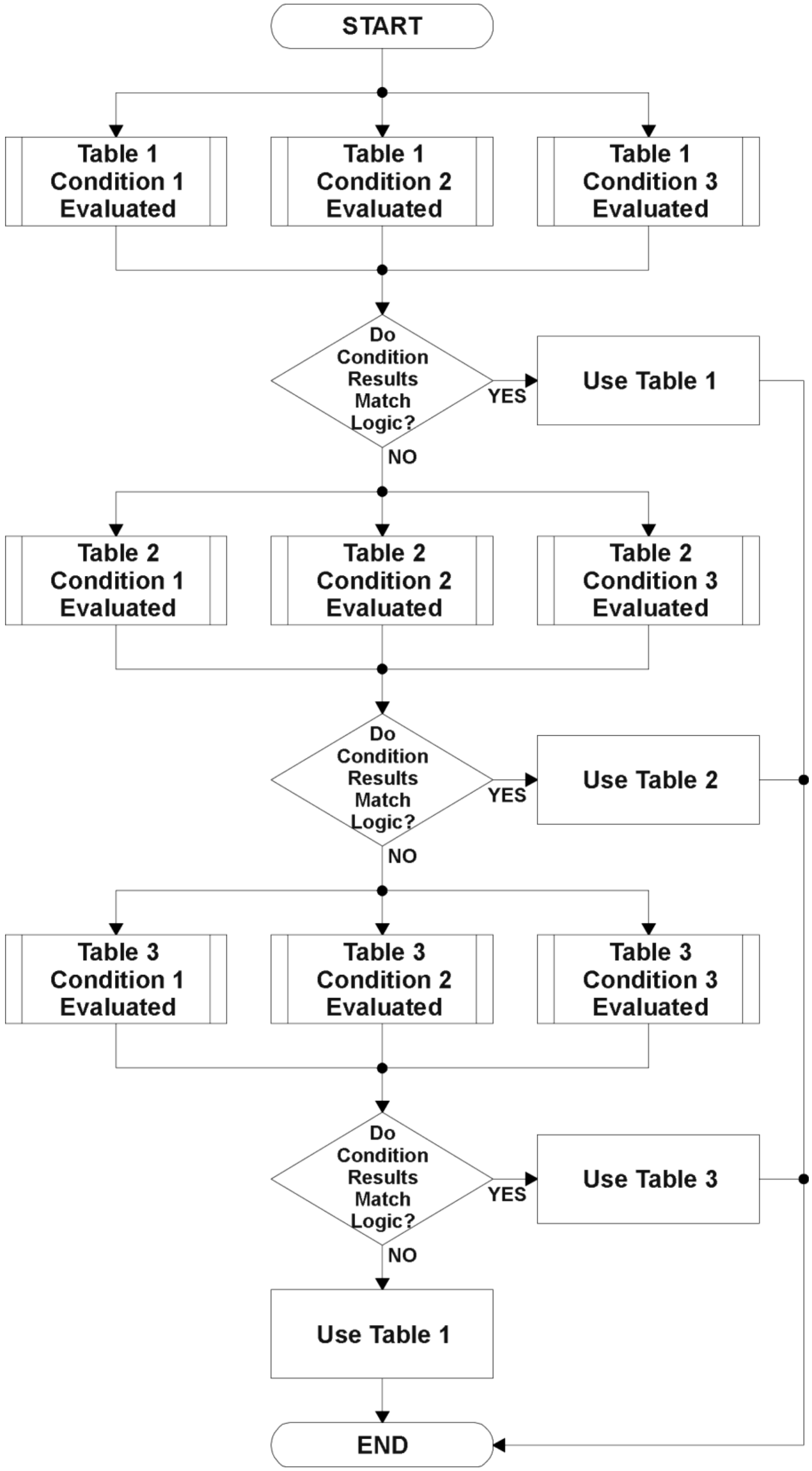


Figure 21 – Programmable Logic Flowchart

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.

Argument 1 is always a logical output from another function block, as listed in Section 1.14. As always, the source is a combination of the functional block type and number, setpoints **"Table X, Condition Y, Argument 1 Source"** and **"Table X, Condition Y, Argument 1 Number"**, where both X = 1 to 3 and Y = 1 to 3.

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 **"Table X, Condition Y, Argument 2 Source"** to *'Control Constant Data.'* Note that the constant value has no unit associated with it in the Axiomatic EA, so the user must set it as needed for the application.

The condition is evaluated based on the **"Table X, Condition Y Operator"** selected by the user. It is always *'=, Equal'* by default. The only way to change this is to have two valid arguments selected for any given condition. Options for the operator are listed in Table 13.

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

Table 13 – Condition Operator Options

For example, a condition for a transmission control shift selection, as shown in Figure 18 in the previous section, could be that the Engine RPM be less than a certain value to select a Soft Fill profile. In this case, **"…Argument 1 Source"** would be set to *'Magnetic Pickup Frequency/RPM',* **"…Argument 2 Source"** to *'Control Constant Data'*, and the **"…Operator"** to *'<, Less Than.'* The **"Constant Value X"** in the Constant Data List would be set to whatever cutoff RPM the application required.

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 Figure 21 shows only True or False as a result of a condition evaluation, the reality is that there could be four possible results, as described in Table 14.

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 14 – 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 15.

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 15 – Conditions 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.

Logical Operator	Select Conditions Criteria
Default Table	Associated table is automatically selected as soon as it is evaluated.
Cnd1 And Cnd2 And Cnd3	**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
Cnd1 Or Cnd2 Or Cnd3	**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
(Cnd1 And Cnd2) Or Cnd3	**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
(Cnd1 Or Cnd2) And Cnd3	**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

Table 16 – Conditions Evaluation Based on Selected Logical Operator

The default **"Table X, Conditions Logical Operator"** for Table 1 and Table 2 is *'Cnd1 And Cnd2 And Cnd3,'* while Table 3 is set to be the *'Default Table.'*

1.6.3. Logic Block Output

Recall that Table X, where X = 1 to 3 in the Programmable Logic function block does NOT mean Lookup Table 1 to 3. Each table has a setpoint **“Table X – Lookup Table Block Number”** which allows the user to select which Lookup Tables they want associated with a particular Programmable Logic Block. The default tables associated with each logic block are listed in Table 17.

Programmable Logic Block Number	Table 1 – Lookup Table Block Number	Table 2 – Lookup Table Block Number	Table 3 – Lookup Table Block Number
1	1	2	3
2	4	5	6
3	1	2	3
4	4	5	6

Table 17 – Programmable Logic Block Default Lookup Tables

If the associated Lookup Table does not have an **“X-Axis Source”** selected, then the output of the Programmable Logic block will always be “Not Available” so long as that table is selected. However, should the Lookup Table be configured for a valid response to an input, be it Data or Time, the output of the Lookup Table function block (i.e. the Y-Axis data that has been selected based on the X-Axis value) will become the output of the Programmable Logic function block so long as that table is selected.

Unlike all other function blocks, the Programmable Logic does NOT perform any linearization calculations between the input and the output data. Instead, it mirrors exactly the input (Lookup Table) data. Therefore, when using the Programmable Logic as a control source for another function block, it is HIGHLY recommended that all the associated Lookup Table Y-Axes either be (a) Set between the 0 to 100% output range or (b) all set to the same scale.

1.6.4. Application Ideas

This section is not meant to be a comprehensive list of all the possibilities that the Programmable Logic offers. Rather, it is meant to show how some common, but widely diversified functions can be achieved by using it.

a) Dual Speed Application – Under certain conditions, a valve could be driven between Min_A to Max_A while under others, the speed is limited by having the output respond to changes at the input between Min_B and Max _B.

b) Multi-Speed Transmission Control – By using a Forward input as the enable of one Output X Drive, and the Reverse input as the other, different clutch fill profiles could be selected based on Engine Speed as discussed in earlier examples.

c) Getting better resolution (i.e. up to 30 slopes) on a resistive to temperature curve for an NTC sensor. The condition for Table 1 would be input resistance <= R1, Table 2 is input <= R2 and Table 3 as the default for high resistance values.

1.7. Math Function Block

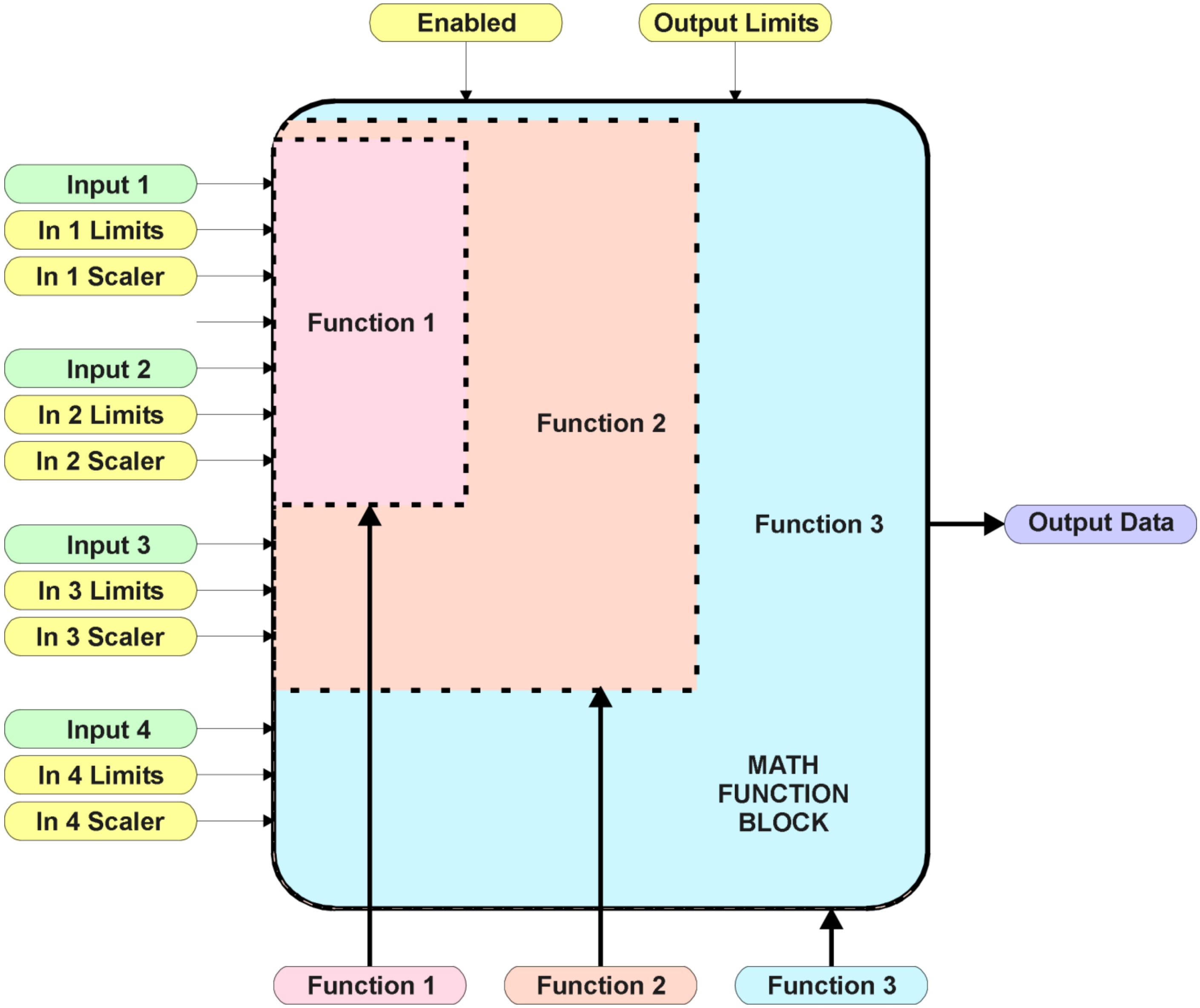


Figure 22 – Math Function Block

There are four mathematic function blocks that allow the user to define basic algorithms. A math function block can take up to four input signals, as listed in Table 24 in Section 1.13. Each input is then scaled according the associated limit and scaling setpoints.

Inputs are converted into a percentage value based on the **"Math Input X Minimum"** and **"Math Input X Maximum"** values selected, where X = 1 to 4. For additional control, the user can also adjust the **"Math Input X Scaler."** By default, each input has a scaling 'weight' of 1.0. However, each input can be scaled from -1.0 to 1.0 as necessary before it is applied in the function.

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 be scaled with 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.

The appropriate arithmetic or logical operation is performed on the two inputs, InA and InB, according the associated function. The list of selectable function operations is defined in Table 18.

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	AND	True when InA and InB are True
8	XOR	True when InA/InB is True, but not both
9	+	Result = InA plus InB
10	-	Result = InA minus InB
11	x	Result = InA times InB
12	/	Result = InA divided by InB
13	MIN	Result = Smallest of InA and InB
14	MAX	Result = Largest of InA and InB

Table 18 – Math Function Operators

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

For a valid result, the control source for an input must be a non-zero value, i.e. something other than *'Control Source Not Used.'* Otherwise, the corresponding function is ignored, and the "Output Data" for the math function block is the result of the earlier function scaled according to the output limit setpoints. For example, if Input 4 is not used, the math output would be the result of the Function 2 operation.

For logical operators (6, 7 or 8), any SCALED input greater than or equal to 0.5 is treated as a TRUE input. For logic output operators (0 to 8), 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 (9 to 14), 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 mathematical calculation, represented as a percentage value, can be scaled into the appropriate physical units using the **"Math Output Minimum Range"** and **"Math Output Maximum Range"** setpoints. These values are also used as the limits when the Math Function is selected as the input source for another function block.

1.8. Filtering Function Block

The filtering function block is associated with the universal input function block.

When used with an input, the software filter is applied to the measured signal (from 12-bit analog-to-digital converter in the case of a voltage/current/resistive input, or from the 15-bit timer in the case of a frequency/PWM input) before the value is moved to the output of the function block, and used as an input to other function blocks (i.e. control or feedback input)

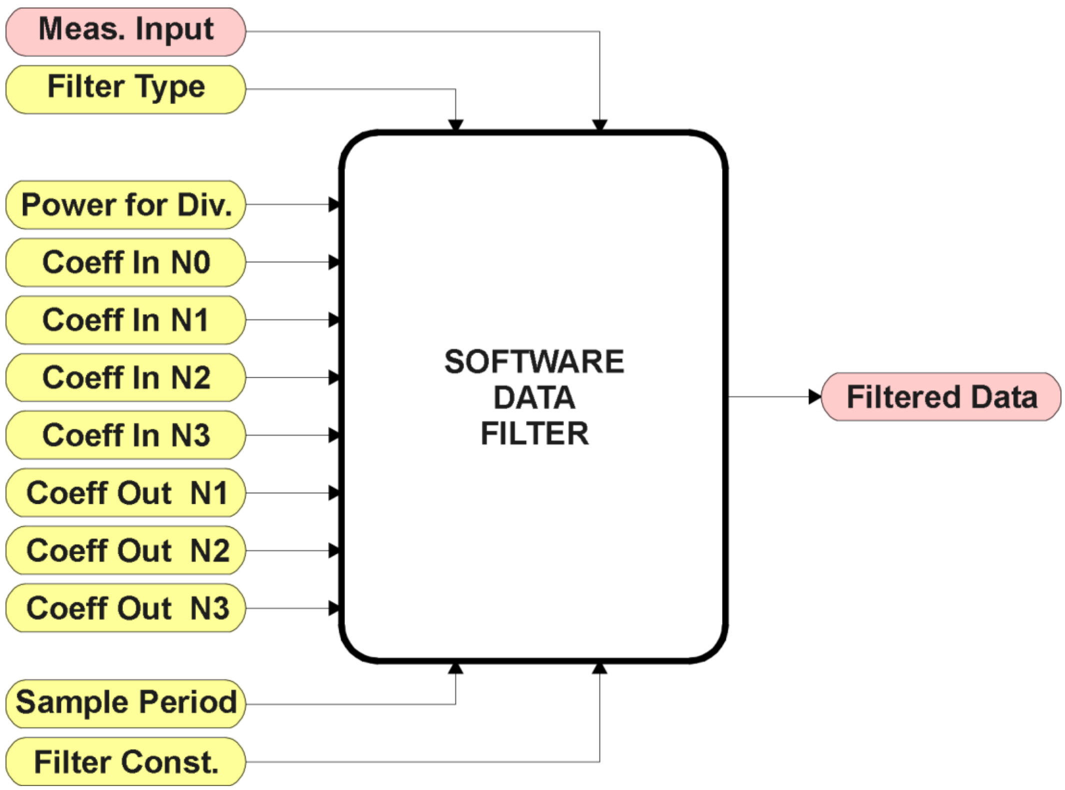


Figure 23 – Software Data Filter Function Block

The **"Software Filter Type"** is the most important setpoint associated with this function block, and it should be selected first. Changing it will result in other setpoints being enabled/disabled as appropriate for the selected filter. The available filters types are shown in Table 19.

0	*No Filter*
1	*Moving Average*
2	*Repeating Average*
3	*3rd Order Low Pass*

Table 19 – Software Filter Types

Calculation with no filter:

Value = Input

The data is simply a ‘snapshot’ of the latest value measured by the ADC or timer.

Calculation with the moving average filter:

$$Value_N = Value_{N-1} + \frac{(Input - Value_{N-1})}{FilterConstant}$$

This filter is called every 1ms.

Calculation with the repeating average filter:

$$Value = \frac{\sum Input_N}{N}$$

At every reading of the input value, it is added to the sum. At every N^{th} read, the sum is divided by N, and the result is the new input value. The value and counter will be set to zero for the next read. The value of N is stored in the “**Filter Constant**” setpoint. This filter is called every 1ms.

Calculation with the 3rd Order Low Pass filter:

$$Value_N = \frac{[C_{IN0}*Input_N + C_{IN1}*Input_{N-1} + C_{IN2}*Input_{N-2} + C_{IN3}*Input_{N-3} + C_{OUT1}*Value_{N-1} + C_{OUT2}*Value_{N-2} + C_{OUT3}*Value_{N-3}]}{2^{Power}}$$

C_{IN0}	Coefficient for Input(n)		
C_{IN1}	Coefficient for Input(n-1)	C_{OUT1}	Coefficient for Output(n-1)
C_{IN2}	Coefficient for Input(n-2)	C_{OUT2}	Coefficient for Output(n-2)
C_{IN3}	Coefficient for Input(n-3)	C_{OUT3}	Coefficient for Output(n-3)

This filter uses 16-bit fixed point math. The “**Filter Power for Divisor**” setpoint tells the controller the shift value used when the coefficients were selected.

The filter will be called every “**Filter Sampling Period**” (in ms). In between calls, the data of the input/feedback signal is the value which was calculated the last time the filter was called.

1.9. Diagnostic Function Block

There are several types of diagnostics supported by the 2i2o Valve Controller. Fault detection and reaction is associated with all universal inputs and output drives. In addition to I/O faults, the 2i2o can also detect/react to power supply over/under voltage measurements, a processor over-temperature, or lost communication events.

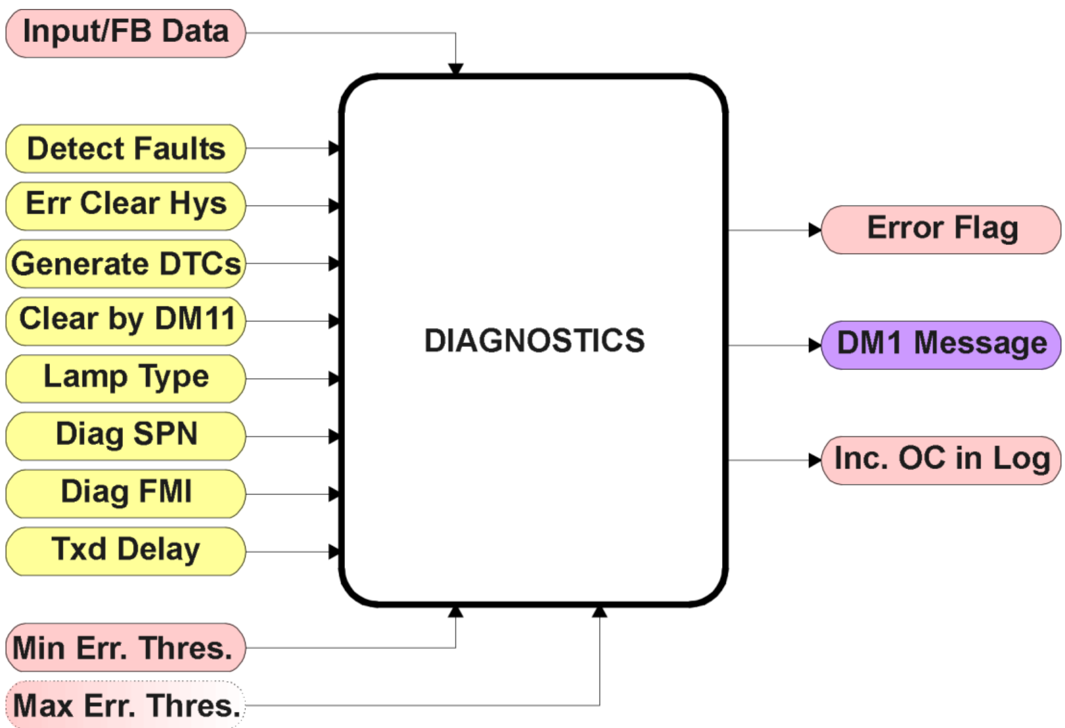


Figure 24 – Diagnostics Function Block

The **"Fault Detection is Enabled"** is the most important setpoint associated with this function block, and it should be selected first. Changing it will result in other setpoints being enabled/disabled as appropriate. When disabled, all diagnostic behaviour associated with the I/O or event in question is ignored (i.e. this type of fault will not disable the output)

In the case of a power supply error, it can be selected to disable the 3A proportional outputs. By default, a power supply under or over voltage condition will not shutoff the outputs.

In most cases, faults can be flagged as either a low or high occurrence. The min/max thresholds for all diagnostics supported by the 2i2o are listed in Table 20. Bolded values are user configurable setpoints. Some diagnostics react only to a single condition, in which case a N/A is listed in one of the columns.

Function Block	Minimum Threshold	Maximum Threshold
Universal Input	**Minimum Error**	**Maximum Error**
Output Drive	**Hysteresis to Clear Fault** *	~4.5A * (Hardware)
Power Supply	**Power Undervoltage Threshold**	**Power Overvoltage Threshold**
Over Temperature	N/A	**Over Temperature Shutdown**
Lost Communication	N/A	**Received Message Timeout** (any)

Table 20 – Fault Detect Thresholds

** The output current diagnostics will compare the desired current with the measured value, and if the difference between the two readings is greater the **"Hysteresis to Clear Fault"** setpoint, an open circuit will be flagged. A hardware shutdown will occur if the output is sourcing greater than 4.5A +/- 0.5A, most likely due to a short circuit on the load. Output diagnostics are not available with non-current output types.*

When applicable, a hysteresis setpoint is provided to prevent the rapid setting and clearing of the error flag when an input or feedback value is right near the fault detection threshold. For the low end, once a fault has been flagged, it will not be cleared until the measured value is greater than or equal to the Minimum Threshold **+ "Hysteresis to Clear Fault."** For the high end, it will not be cleared until the measured value is less than or equal to the Maximum Threshold **– "Hysteresis to Clear Fault."** The minimum, maximum and hysteresis values are always measured in the units of the fault in question.

The next setpoint in this function block is the **"Event Generates a DTC in DM1."** If and only if this is set to true will the other setpoints in the function block be enabled. They are all related to the data that is sent to the J1939 network as part of the DM1 message, Active Diagnostic Trouble Codes.

A Diagnostic Trouble Code (DTC) is defined by the J1939 standard as a four byte value which is a combination of:

SPN	Suspect Parameter Number	(first 19 bits of the DTC, LSB first)
FMI	Failure Mode Identifier	(next 5 bits of the DTC)
CM	Conversion Method	(1 bit, always set to 0)
OC	Occurrence Count	(7 bits, number of times the fault has happened)

In addition to supporting the DM1 message, the 2i2o Valve Controller also supports

DM2	Previously Active Diagnostic Trouble Codes	**Sent only on request**
DM3	Diagnostic Data Clear/Reset of Previously Active DTCs	**Done only on request**
DM11	Diagnostic Data Clear/Reset for Active DTCs	**Done only on request**

So long as even one Diagnostic function block has **"Event Generates a DTC in DM1"** set to True, the 2i2o Valve Controller will send the DM1 message every one second, regardless of whether or not there are any active faults, as recommended by the standard. While there are no active DTCs, the 2i2o will send the "No Active Faults" message. If a previously inactive DTC becomes active, a DM1 will be sent immediately to reflect this. As soon as the last active DTC goes inactive, it will send a DM1 indicating that there are no more active DTCs.

If there is more than one active DTC at any given time, the regular DM1 message will be sent using a multipacket Broadcast Announce Message (BAM). If the controller receives a request for a

DM1 while this is true, it will send the multipacket message to the Requester Address using the Transport Protocol (TP).

At power up, the DM1 message will not be broadcasted until after a 5 second delay. This is done to prevent any power up or initialization conditions from being flagged as an active error on the network.

When the fault is linked to a DTC, a non-volatile log of the occurrence count (OC) is kept. As soon as the controller detects a new (previously inactive) fault, it will start decrementing the "**Delay Before Sending DM1**" timer for that Diagnostic function block. If the fault has remained present during the delay time, then the controller will set the DTC to active, and will increment the OC in the log. A DM1 will immediately be generated that includes the new DTC. The timer is provided so that intermittent faults do not overwhelm the network as the fault comes and goes, since a DM1 message would be sent every time the fault shows up or goes away.

Previously active DTCs (any with a non-zero OC) are available upon request for a **DM2** message. If there is more than one previously active DTC, the multipacket DM2 will be sent to the Requester Address using the Transport Protocol (TP).

Should a **DM3** be requested, the occurrence count of all previously active DTCs will be reset to zero. The OC of currently active DTCs will not be changed.

The Diagnostic function block has a setpoint **"Event Cleared only by DM11."** By default, this is always set to False, which means that as soon as the condition that caused an error flag to be set goes away, the DTC is automatically made Previously Active, and is no longer included in the DM1 message. However, when this setpoint is set to True, even if the flag is cleared, the DTC will not be made inactive, so it will continue to be sent on the DM1 message. Only when a DM11 has been requested will the DTC go inactive. This feature may be useful in a system where a critical fault needs to be clearly identified as having happened, even if the conditions that caused it went away.

In addition to all the active DTCs, another part of the DM1 message is the first byte which reflects the Lamp Status. Each Diagnostic function block has the setpoint **"Lamp Set by Event in DM1"** which determines which lamp will be set in this byte while the DTC is active. The J1939 standard defines the lamps as *'Malfunction', 'Red, Stop', 'Amber, Warning'* or *'Protect'.* By default, the *'Amber, Warning'* lamp is typically the one set by any active fault.

By default, every Diagnostic function block has associated with it a proprietary SPN. However, this setpoint **"SPN for Event used in DTC"** is fully configurable by the user should they wish it to reflect a standard SPN define in J1939-71 instead. If the SPN is changed, the OC of the associate error log is automatically reset to zero.

Every Diagnostic function block also has associated with it a default FMI. The only setpoint for the user to change the FMI is **"FMI for Event used in DTC,"** even though some Diagnostic function blocks can have both high and low errors as shown in Table 21. In those cases, the FMI in the setpoint reflect that of the low end condition, and the FMI used by the high fault will be determined per Table 21. If the FMI is changed, the OC of the associate error log is automatically reset to zero.

FMI for Event used in DTC – Low Fault	Corresponding FMI used in DTC – High Fault
FMI=1, Data Valid But Below Normal Operational Range – Most Severe Level	FMI=0, Data Valid But Above Normal Operational Range – Most Severe Level
FMI=4, Voltage Below Normal, Or Shorted To Low Source	FMI=3, Voltage Above Normal, Or Shorted To High Source
FMI=5, Current Below Normal Or Open Circuit	FMI=6, Current Above Normal Or Grounded Circuit
FMI=17, Data Valid But Below Normal Operating Range – Least Severe Level	FMI=15, Data Valid But Above Normal Operating Range – Least Severe Level
FMI=18, Data Valid But Below Normal Operating Range – Moderately Severe Level	FMI=16, Data Valid But Above Normal Operating Range – Moderately Severe Level
FMI=21, Data Drifted Low	FMI=20, Data Drifted High

Table 21 – Low Fault FMI versus High Fault FMI

If the FMI used is anything other than one of those in Table 21, then both the low and high faults will be assigned the same FMI. This condition should be avoided, as the log will still used different OC for the two types of faults, even though they will be reported the same in the DTC. It is the user's responsibility to make sure this does not happen.

1.10. DTC React Function Block

The DTC React function block is a very simple function which will allow a received DTC, sent from another ECU on a DM1 message, to disable an output or be used as input to another type of logic block. Up to five SPN/FMI combinations can be selected.

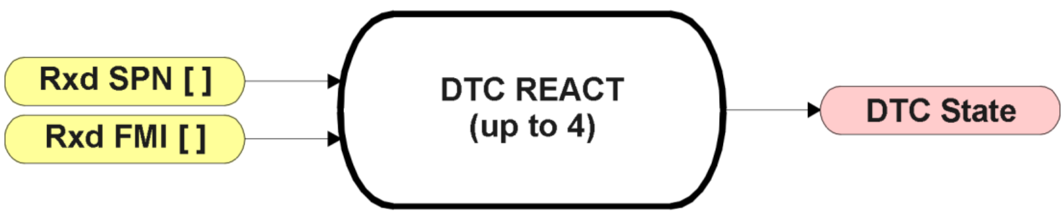


Figure 25 – DTC React Function Block

Should a DM1 message be received with the SPN/FMI combination defined, the corresponding DTC State will be set to ON. Once ON, if the same SPN/FMI combination has not been received again after 3 seconds, the DTC State will be reset to OFF.

The DTC could be used as a digital input for any function block as appropriate.

1.11. CAN Transmit Function Block

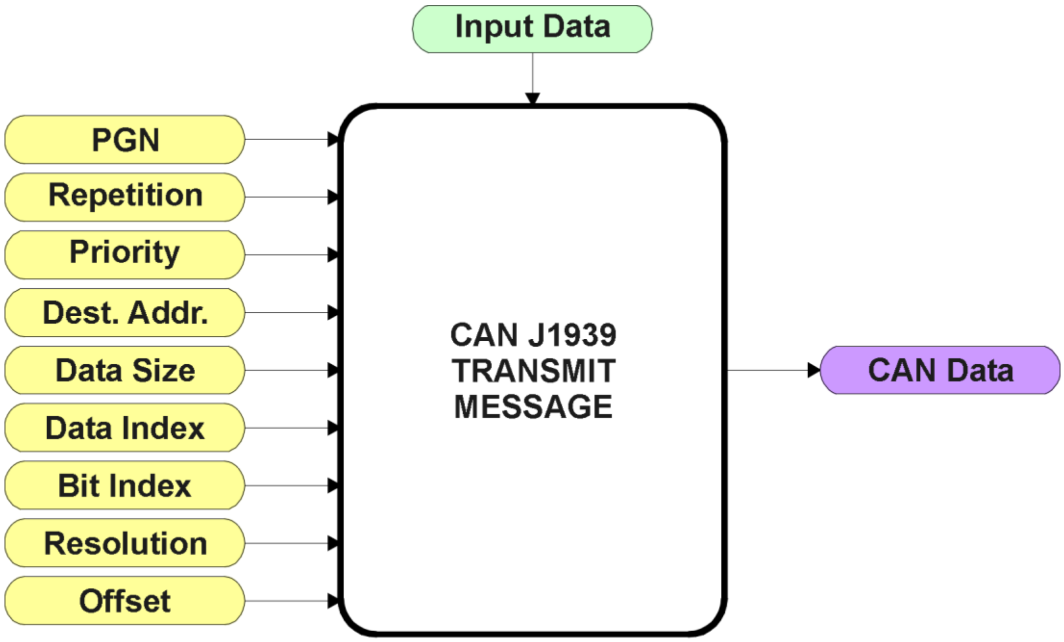


Figure 26 – CAN Transmit Function Block

The CAN Transmit function block is used to send any output from another function block (i.e. input, status or feedback signals) to the J1939 network.

Normally, to disable a transmit message, the **"Transmit Repetition Rate"** is set to zero. However, should message share its Parameter Group Number (PGN) with another message, this is not necessarily true. In the case where multiple messages share the same **"Transmit PGN"**, the repetition rate selected in the message with the LOWEST number will be used for ALL the messages that use that PGN.

By default, all messages are sent on Proprietary B PGNs as broadcast messages. The default settings do 'bundle' multiple messages onto a PGN, as outlined in Section 3. If all of the data is not necessary, disable the entire message by setting the lowest channel using that PGN to zero. If some of the data is not necessary, simply change the PGN of the superfluous channel(s) to an unused value in the Proprietary B range.

	At power up, transmitted message will not be broadcasted until after a 5 second delay. This is done to prevent any power up or initialization conditions from creating problems on the network.

Since the defaults are PropB messages, the **"Transmit Message Priority"** is always initialized to 6 (low priority) and the **"Destination Address (for PDU1)"** setpoint is not used. This setpoint is only valid when a PDU1 PGN has been select, and it can be set either to the Global Address (0xFF) for broadcasts, or sent to a specific address as setup by the user.

The **“Transmit Data Size”**, **“Transmit Data Index in Array (LSB)”**, **“Transmit Bit Index in Byte (LSB)”**, **“Transmit Resolution”** and **“Transmit Offset”** can all be use to map the data to any SPN supported by the J1939 standard. The defaults used by the 2i2o are all for proprietary SPNs, and are defined in detail in Section 3.3.

Note: CAN Data = (Input Data – Offset)/Resolution

The 2i2o Valve Controller supports up to 8 unique CAN Transmit Messages, all of which can be programmed to send any available data to the CAN network. By default, each message is pre-configured to send a particular type of data. The details are outlined in Section 3.3, and the default list is shown in Table 22 below.

Block #	Default Transmit Data	(PGN)
1	Universal Input 1 Measured Value	(0xFF00)
2	Universal Input 2 Measured Value	(0xFF00)
3	Power Supply Measured	(0xFF00)
4	Processor Temperature Measured	(0xFF00)
5	Proportional Output 1 Target Value	(0xFF10)
6	Proportional Output 1 Current Feedback	(0xFF10)
7	Proportional Output 2 Target Value	(0xFF10)
8	Proportional Output 2 Current Feedback	(0xFF10)

Table 22 – Default CAN Transmit Messages

1.12. CAN Receive Function Block

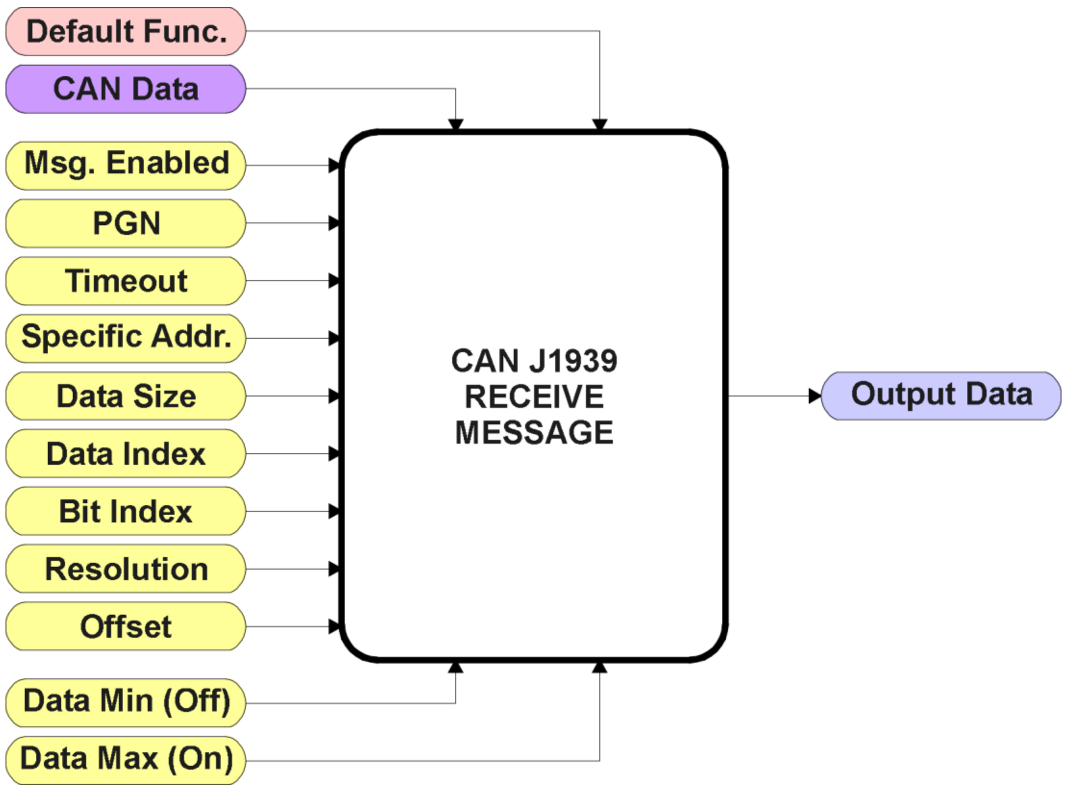


Figure 27 – CAN Receive Function Block

The CAN Receive function block is designed to take any SPN from the J1939 network, and use it as an input to any another function block (i.e. Lookup Table or Output Control).

The **"Received Message Enabled"** is the most important setpoint associated with this function block, and it should be selected first. Changing it will result in other setpoints being enabled/disabled as appropriate. By default ALL received messages are disabled.

Since the 2i2o can support up to 8 received messages, it can get confusing as to what message performs what function. Therefore, the intended **"Default Receive Function"** is available as a read-only value for each message. Change functions with CAUTION!

Once a message has been enabled, a Lost Communication fault will be flagged if that message is not received off of the bus within the **"Receive Message Timeout"** period. This could trigger a Lost Communication event as described in Section 1.9. In order to avoid timeouts on a heavily saturated network, it is recommended to set it at least three times larger than the expected update rate. To disable the timeout feature, simply set this value to zero, in which case the received message will never trigger a Lost Communication fault.

By default, all control messages are expected to be sent to the 2i2o on Proprietary B PGNs. However, should a PDU1 message be selected, the 2i2o can be setup to receive it from any ECU by setting the **"Specific Address that sends the PGN"** to the Global Address (0xFF). If a specific address is selected instead, then any other ECU sending data on the PGN will be ignored.

The **“Receive Data Size”**, **“Receive Data Index in Array (LSB)”**, **“Receive Bit Index in Byte (LSB)”**, **“Receive Resolution”** and **“Receive Offset”** can all be use to map any SPN supported by the J1939 standard to the output data of the Received function block. The defaults used by the 2i2o are all for proprietary SPNs, and are defined in detail in Section 3.4.

Note: Output Data = CAN Data * Resolution + Offset

As mentioned earlier, a CAN Receive function block can be selected as the source of the control input for the output function blocks. When this is the case, the **“Received Data Min (Off Threshold)”** and **“Received Data Max (On Threshold)”** setpoints become the minimum and maximum values of the X-axis used in the linear calculations. As the names imply, they are also used as the On/Off thresholds for digital input types. These values are in whatever units the output data is AFTER the resolution and offset is applied to the CAN data.

The 2i2o Valve Controller supports up to 8 unique CAN Receive Messages. By default, each message is pre-configured to read a particular type of data. The details are outlined in Section 3.4, and the default list is shown in Table 23 below.

Block #	Default Receive Data
1	Output 1 Command Input Data
2	Output 2 Command Input Data
3	Output 1 Closed Loop Feedback
4	Output 2 Closed Loop Feedback
5	Output 1 Enable Input Data
6	Output 2 Enable Input Data
7	Output 1 Override Input Data
8	Output 2 Override Input Data

Table 23 – Default CAN Receive Messages

1.13. Available Control/Input Sources

Throughout Section 1, there have been numerous mentions of 'mapping' an output of one function block to a control or input of another. Any time an input is programmable, as shown by the green bubbles in the figures above, it consists of two setpoints. One is the **"[Name] Source"** and the other the **"[Name] Number."** Together, these two setpoints uniquely select how the I/O of the various function blocks are linked together in the custom application.

Below is the full list of the programmable inputs supported by the 2i2o Valve Controller.

Function Block	Input Name
Output X Drive (where X =1 to 2)	Control
	Enable
	Override
PID Control X (where X =1 to 2)	PID Target Command
	PID Feedback Input
Lookup Table X (where X = 1 to 8)	X-Axis
Programmable Logic	Table 1 – Lookup Table Block Number
	Table 1 – Condition 1, Argument 1
	Table 1 – Condition 1, Argument 2
	Table 1 – Condition 2, Argument 1
	Table 1 – Condition 2, Argument 2
	Table 1 – Condition 3, Argument 1
	Table 1 – Condition 3, Argument 2
	Table 2 – Lookup Table Block Number
	Table 2 – Condition 1, Argument 1
	Table 2 – Condition 1, Argument 2
	Table 2 – Condition 2, Argument 1
	Table 2 – Condition 2, Argument 2
	Table 2 – Condition 3, Argument 1
	Table 2 – Condition 3, Argument 2
	Table 3 – Lookup Table Block Number
	Table 3 – Condition 1, Argument 1
	Table 3 – Condition 1, Argument 2
	Table 3 – Condition 2, Argument 1
	Table 3 – Condition 2, Argument 2
	Table 3 – Condition 3, Argument 1
	Table 3 – Condition 3, Argument 2
Math Function X (where X = 1 to 4)	Input 1, Input 2, Input 3 and Input 4
CAN Transmit X (where X = 1 to 8)	Transmit Data

Table 24 – Programmable Inputs

For each of the inputs listed in Table 24, the options for the Source and Number are listed below. Note, not all options would make sense for any particular input, and it is up to the user to program the controller in a logical and functional fashion.

Sources	Range	Notes
Control Source Not Used	N/A	When this is selected, it disables all other setpoints associated with the input in question.
Received CAN J1939 Message	1 to 8	The CAN Receive Messages each have a unique default assigned to them as outlined in Section 1.11
Universal Input Measured	1 to 2	
PID Function Block	1 to 2	User must enable the function block, as it is disabled by default
Lookup Table Function Block	1 to 8	User must enable the function block, as it is disabled by default
Programmable Logic Function Block	1 to 4	User must enable the function block, as it is disabled by default
Math Function Block	1 to 4	User must enable the function block, as it is disabled by default
Control Constant Data	1 to 14	1 = Always FALSE, 2 = Always TRUE, 3 to 14 = User Selectable
Diagnostic Trouble Code (Rxd)	1 to 4	Will only be valid if the corresponding DTC has a non-zero SPN
Note: Control sources below this line are meant for feedback and diagnostics, and are not suitable for use as controls for other logic blocks		
Universal Output Target Value	1 to 2	Can be mapped to a CAN Txd
Universal Output Feedback Value	1 to 2	Input to Output X Drive Diagnostics Can be mapped to a CAN Txd
Power Supply Measured	N/A	Input to Power Supply Diagnostics Can be mapped to a CAN Txd
Processor Temperature Measured	N/A	Input to Over Temperature Diagnostics Can be mapped to a CAN Txd

Table 25 – Input Sources and Numbers

1.14. Default Valve Controller Program Logic

Below is shown an **extremely** simplified block diagram illustrating the default control logic programmed into the 2i2o Valve Controller.

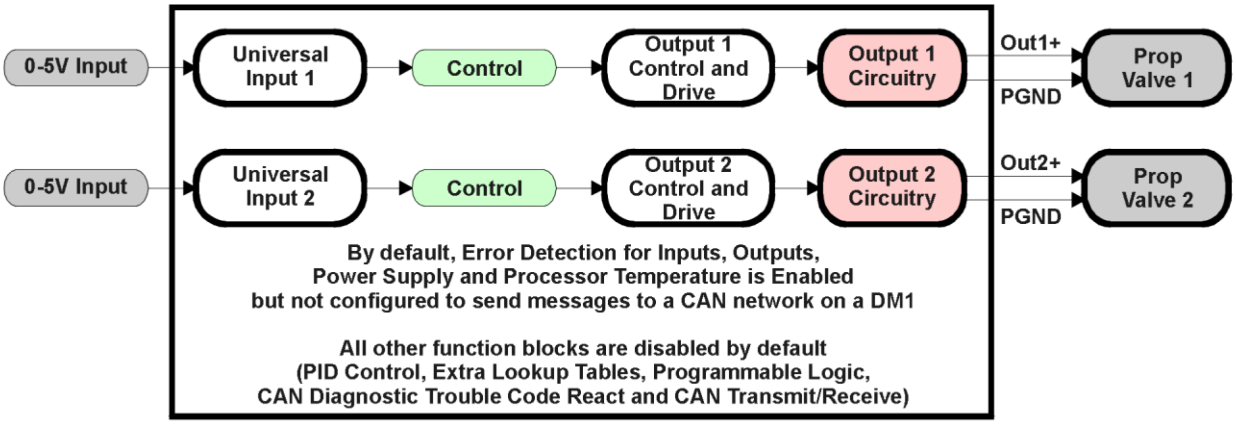


Figure 28 – Default Control Logic

By default, the controller is programmed to be fully self-contained, and does not require a connection to the CAN network to function.

2. INSTALLATION INSTRUCTIONS

2.1. Dimensions and Pinout

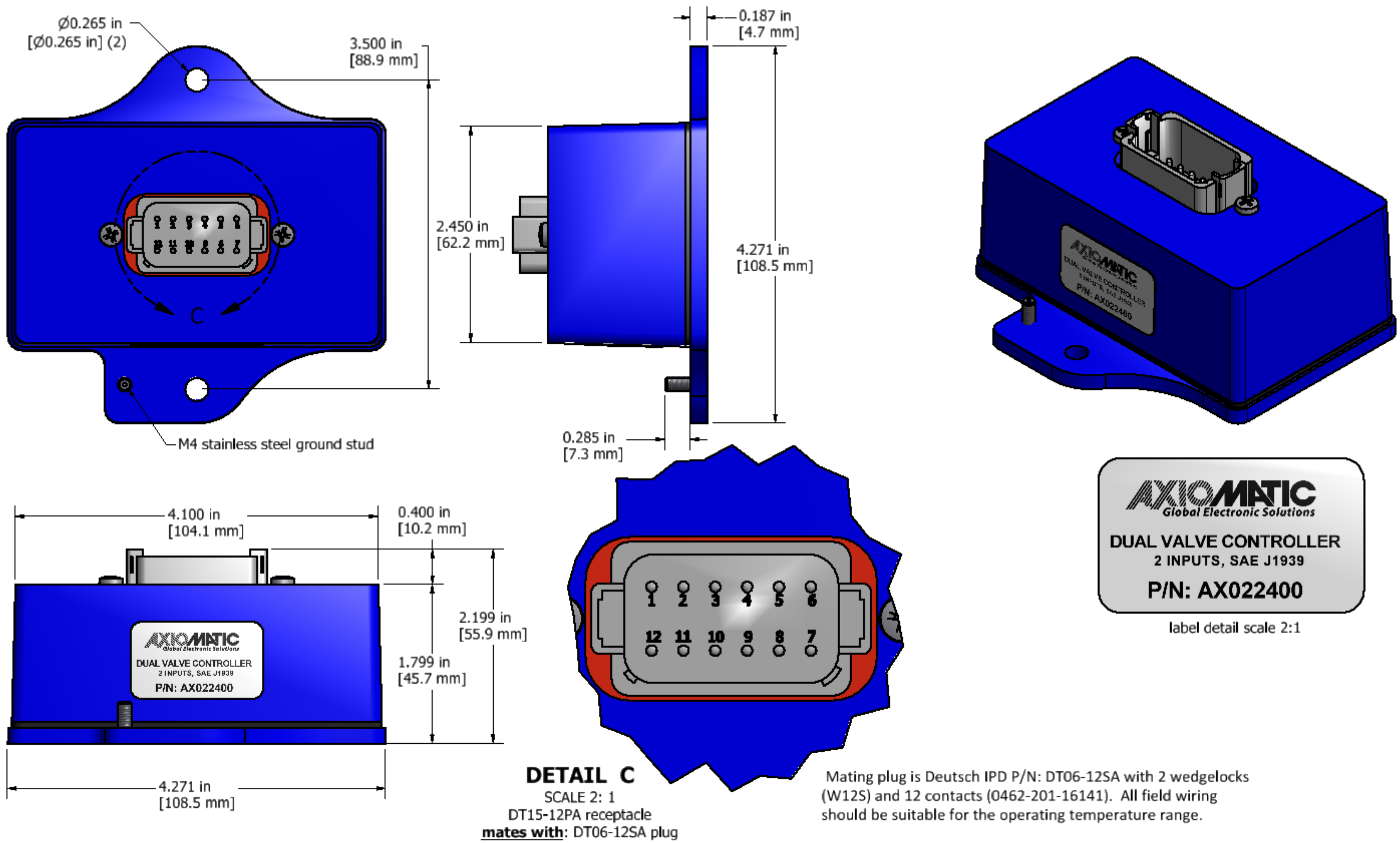


Figure 29 – AX022400 Dimension Drawing

CAN and I/O Connector		
Pin #	**Function**	**(Notes)**
1	Output 2 +	
2	Output 1 +	
3	P_GND	(Out1 and Out2)
4	A_GND	(In1 and In2)
5	CAN_L	
6	CAN_H	
7	CAN_Shield	
8	Input 2 +	
9	Input 1 +	
10	+5Vref	(up to 50mA)
11	BATT -	
12	BATT +	

Table 26 – AX022400 Connector Pinout

12-pin connector (equivalent TE Deutsch P/N: DT15-12PA)

A mating plug kit is available as Axiomatic P/N: **AX070105**.
It is equivalent to the following TE Deutsch P/N: 1 plug DT06-12SA; 1 wedge W12S; 12 sockets 0462-201-16141; and 6 plugs 114017 for any unused pins.

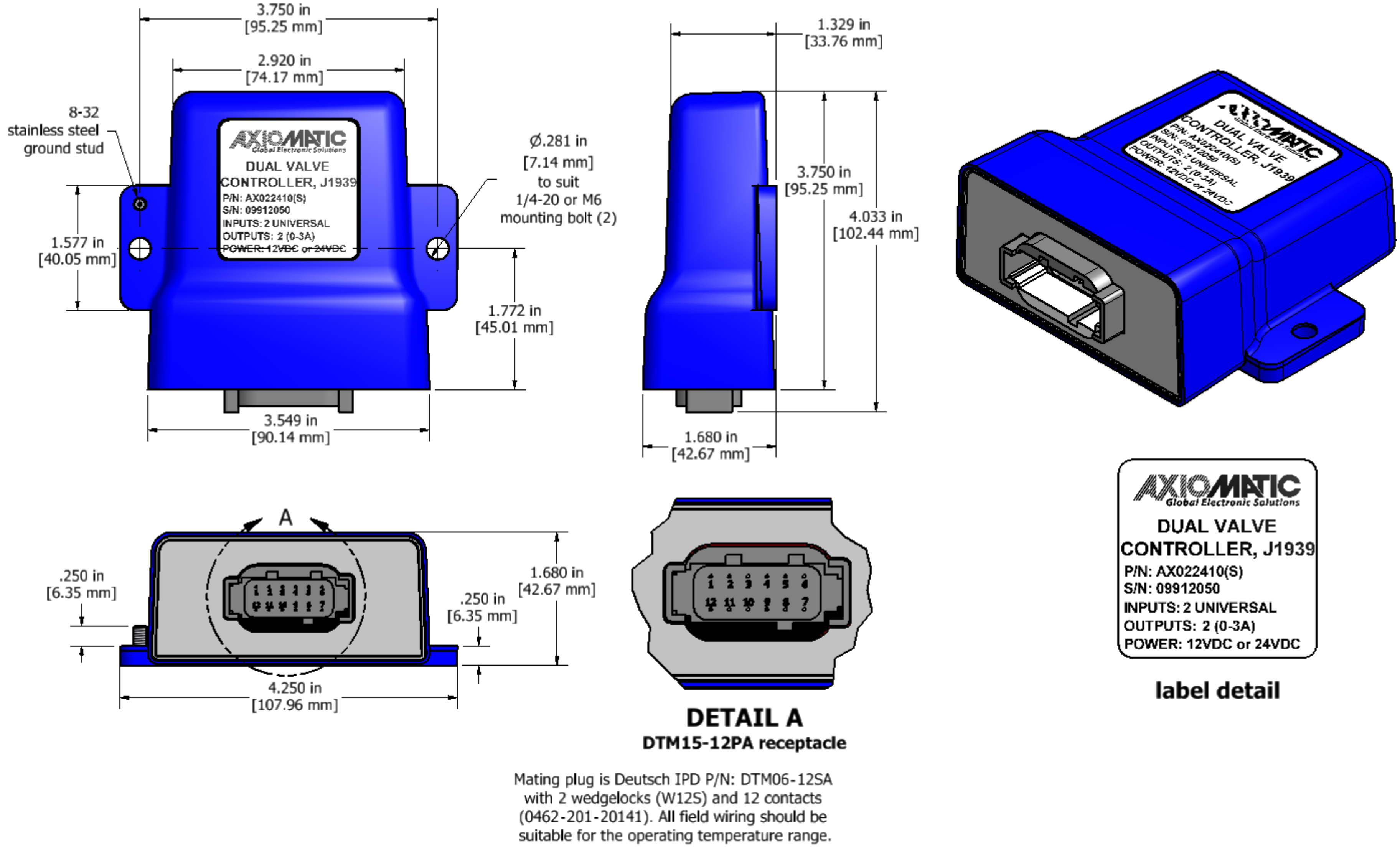


Figure 30 – AX022410 Dimension Drawing

CAN and I/O Connector		
Pin #	**Function**	**(Notes)**
1	Input 2 +	
2	Input 1 +	
3	A_GND	(In1 and In2)
4	P_GND	(Out1 and Out2)
5	Output 1 +	
6	BATT +	
7	BATT -	
8	Output 2 +	
9	CAN_Shield	
10	CAN_L	
11	CAN_H	
12	+5Vref	(up to 50mA)

Table 27 – AX022410 Connector Pinout

12-pin connector (equivalent TE Deutsch P/N: DTM15-12PA)

A mating plug kit is available as Axiomatic P/N: **PL-DTM06-12SA**.
It is equivalent to the following TE Deutsch P/Ns: 1 plug DTM06-12SA; 1 wedge WM12S; 12 sockets 0462-201-20141; and 6 plugs 0413-204-2005 for any unused pins.

2.2. Mounting Instructions

NOTES & WARNINGS

- Do not install near high-voltage or high-current devices.
- Ground the chassis for safety purposes and proper EMI shielding.
- Note the operating temperature range. All field wiring must be suitable for that temperature range.
- Install the unit with appropriate space available for servicing and for adequate wire harness access (15 cm) and strain relief (30 cm).
- Do not connect or disconnect the unit while the circuit is live, unless the area is known to be non-hazardous.

MOUNTING

The module is designed for mounting on the valve block. If it is mounted without an enclosure, the controller should be mounted horizontally with connectors facing left or right, or with the connectors facing down, to reduce likelihood of moisture entry.

Mask all labels if the unit is to be repainted, so label information remains visible.

Mounting legs include holes sized for up to ¼-20 or M6 bolts. The bolt length will be determined by the end-user's mounting plate thickness. Typically 20 mm (3/4 inch) is adequate.

If the module is mounted away from the valve block, no wire or cable in the harness should exceed 30 meters in length. The power input wiring should be limited to 10 meters.

CONNECTIONS

Use the following TE Deutsch mating plugs to connect to the integral receptacles. Wiring to these mating plugs must be in accordance with all applicable local codes. Suitable field wiring for the rated voltage and current must be used. The rating of the connecting cables must be at least 85°C. For ambient temperatures below –10°C and above +70°C, use field wiring suitable for both minimum and maximum ambient temperature.

AX022400 – Top mount Connection

Mating Connector	DT06-12SA and wedge W12S
Sockets	0462-201-16141 or acceptable alternate Refer to www.laddinc.com for more information on the contacts available for this mating plug.

AX022410 – Side mount Connection

Mating Connector	DTM06-12SA and wedge WM12S
Sockets	0462-201-20141 or acceptable alternate Refer to www.laddinc.com for more information on the contacts available for this mating plug.

NOISE – ELECTRICAL CONNECTIONS AND SHIELDING

To reduce noise, separate all power and output wires from those of the input and CAN. Shielded wires will protect against injected or emitted noise. Shield wires should be connected at the power or input source, or at the output load.

The CAN shield can be connected at the controller using the CAN Shield pin provide on the connector. However the other end should not be connected in this case.

Alternatively, all shielded cables can be tied to the ground stud on the housing. Refer to GROUNDING below for more information.

The decision on whether or not to use shield cables is ultimately up to the customer depending on the application. However, Axiomatic recommends independent two (or three) wire shield cables for the following I/O pairs in order to reduce EM noise and improve EMI performance: POWER, OUTPUT 1, OUTPUT 2, INPUT 1, INPUT 2, and CAN. Without proper shielding on the cables, Axiomatic cannot guarantee EMC performance in the system.

WIRING

All wires used must be 16 or 18 AWG for the AX022400. Axiomatic recommends using flexible stranded wires. The jacket diameter on the wire must be sized according to Deutsch specifications for the DT06 connector. Refer to www.laddinc.com for more information on the TE Deutsch plug.

All wires used must be 18 or 20 AWG for the AX022410. Axiomatic recommends using flexible stranded wires. The jacket diameter on the wire must be sized according to Deutsch specifications for the DTM06 connector. Refer to www.laddinc.com for more information on the TE Deutsch plug.

GROUNDING

Protective Earth (PE) must be connected to the module's grounding lug to reduce the risk of electric shock. The conductor providing the connection must have a ring lug and wire larger than or equal to 4 mm^2 (12 AWG). The ring lug should be placed between the nut and a star washer.

All chassis grounding should go to a single ground point designated for the machine and all related equipment. Axiomatic recommends that the ground strap that provides a low impedance path for EMI should be a ½ inch wide, flat, hollow braid, no more than 12 inches long .

CAN NETWORK CONSTRUCTION

Axiomatic recommends that multi-drop networks be constructed using a "daisy chain" or "backbone" configuration with short drop lines.

CAN TERMINATION

It is necessary to terminate the network; therefore an external CAN termination is required. No more than two network terminators should be used on any one single network. A terminator is a 121Ω, 0.25 W, 1% metal film resistor placed between CAN_H and CAN_L terminals at the end two nodes on a network.

3. OVERVIEW OF J1939 FEATURES

The software was designed to provide flexibility to the user with respect to messages sent to and from the ECU by providing:

- Configurable ECU Instance in the NAME (to allow multiple ECUs on the same network)
- Configurable Transmit PGN and SPN Parameters
- Configurable Receive PGN and SPN Parameters
- Sending DM1 Diagnostic Message Parameters
- Reading and reacting to DM1 messages sent by other ECUs
- Diagnostic Log, maintained in non-volatile memory, for sending DM2 messages

3.1. Introduction To Supported Messages

The ECU is compliant with the standard SAE J1939, and supports the following PGNs

From J1939-21 - Data Link Layer

- Request 59904 ($00EA00)
- Acknowledgment 59392 ($00E800)
- Transport Protocol – Connection Management 60416 ($00EC00)
- Transport Protocol – Data Transfer Message 60160 ($00EB00)
- PropB Transmit, Default Measured Inputs Feedback Message 65280 ($00FF00)
- PropB Transmit, Default Proportional Outputs Target Message 65296 ($00FF10)
- PropB Transmit, Default Proportional Outputs Feedback Message 65312 ($00FF20)
- PropB Transmit, Default Digital I/O State Feedback Message 65328 ($00FF30)
- PropB Receive, Default Output Control Data Message 65408 ($00FF80)
- PropB Receive, Default Output Enable Data Message 65424 ($00FF90)
- PropB Receive, Default Output Override Data Message 65440 ($00FFA0)
- PropB Receive, Default PID Feedback Data Message 65456 ($00FFB0)

Note: Any Proprietary B PGN in the range 65280 to 65535 ($00FF00 to $00FFFF) can be selected
Note: The Proprietary A PGN 61184 ($00EF00) can also be selected for any of the messages

From J1939-73 - Diagnostics

- DM1 – Active Diagnostic Trouble Codes 65226 ($00FECA)
- DM2 – Previously Active Diagnostic Trouble Codes 65227 ($00FECB)
- DM3 – Diagnostic Data Clear/Reset for Previously Active DTCs 65228 ($00FECC)
- DM11 - Diagnostic Data Clear/Reset for Active DTCs 65235 ($00FED3)
- DM14 – Memory Access Request 55552 ($00D900)
- DM15 – Memory Access Response 55296 ($00D800)
- DM16 – Binary Data Transfer 55040 ($00D700)

From J1939-81 - Network Management

- Address Claimed/Cannot Claim 60928 ($00EE00)
- Commanded Address 65240 ($00FED8)

From J1939-71 – Vehicle Application Layer

- Software Identification 65242 ($00FEDA)

None of the application layer PGNs are supported as part of the default configurations, but they can be selected as desired for either transmit or received function blocks.

Setpoints are accessed using standard Memory Access Protocol (MAP) with proprietary addresses. The Axiomatic Electronic Assistant (EA) allows for quick and easy configuration of the unit over the CAN network.

3.2. NAME, Address and Software ID

J1939 NAME

The 2i2o Valve Controller ECU has the following defaults for the J1939 NAME. The user should refer to the SAE J1939/81 standard for more information on these parameters and their ranges.

Arbitrary Address Capable	Yes
Industry Group	0, Global
Vehicle System Instance	0
Vehicle System	0, Non-specific system
Function	66, I/O Controller
Function Instance	22, Axiomatic AX022400, Dual Input, Dual Output, Top Connect 30, Axiomatic AX022410, Dual Input, Dual Output, Side Connect
ECU Instance	**0, First Instance**
Manufacture Code	162, Axiomatic Technologies Corporation
Identity Number	Variable, uniquely assigned during factory programming for each ECU

The ECU Instance is a configurable setpoint associated with the NAME. Changing this value will allow multiple ECUs of this type to be distinguishable by other ECUs (including the Axiomatic Electronic Assistant) when they are all connected on the same network.

ECU Address

The default value of this setpoint is 128 (0x80), which is the preferred starting address for self-configurable ECUs as set by the SAE in J1939 tables B3 to B7. The Axiomatic EA supports the selection of any address between 0 to 253, and ***it is the user's responsibility to select an address that complies with the standard***. The user must also be aware that since the unit is arbitrary address capable, if another ECU with a higher priority NAME contends for the selected address, the 2i2o will continue select the next highest address until it find one that it can claim. See J1939/81 for more details about address claiming.

Software Identifier

PGN 65242 Software Identification - SOFT

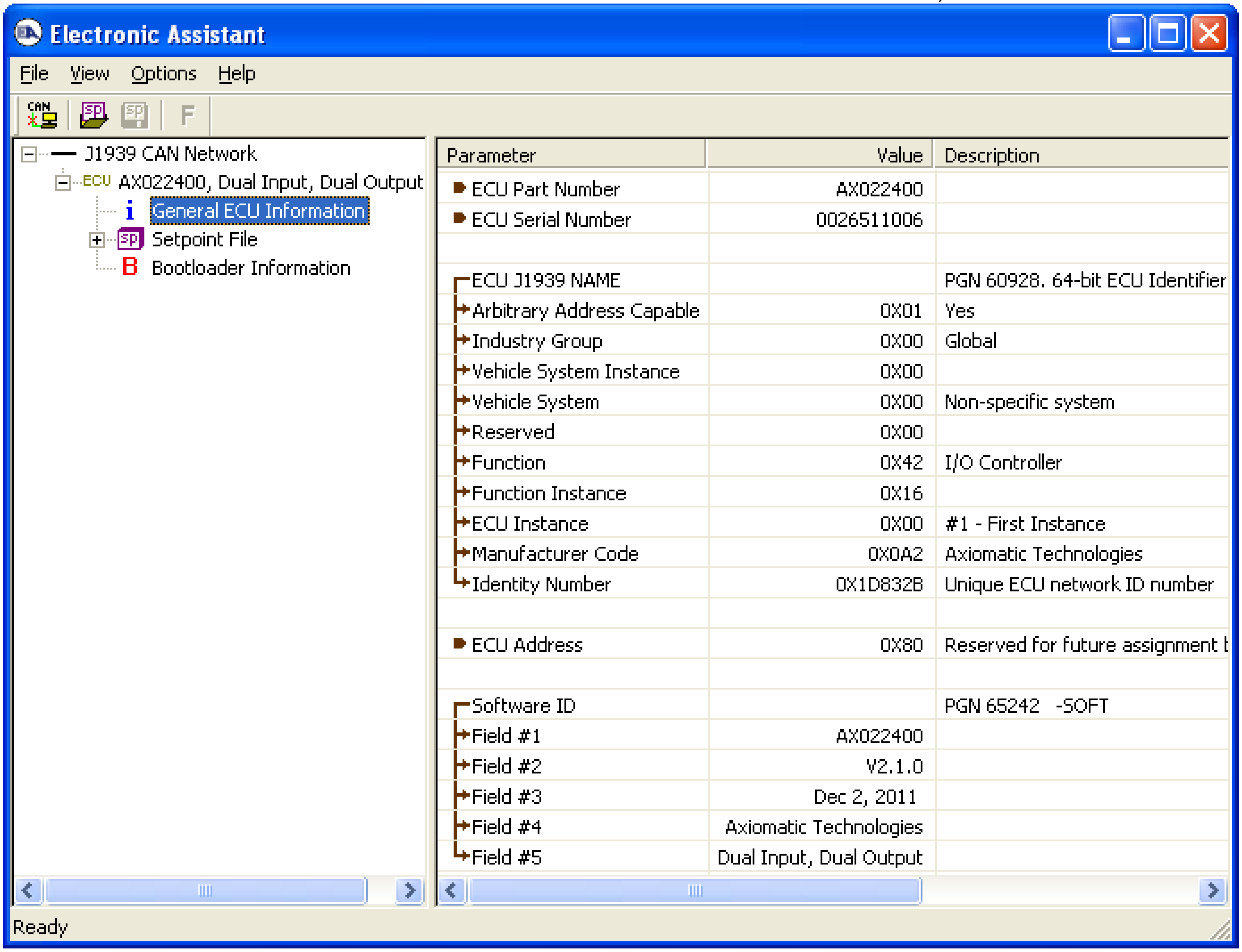
Transmission Repetition Rate:	On request
Data Length:	Variable
Extended Data Page:	0
Data Page:	0
PDU Format:	254
PDU Specific:	218 PGN Supporting Information:
Default Priority:	6
Parameter Group Number:	65242 (0xFEDA)

Start Position	Length	Parameter Name	SPN
1	1 Byte	Number of software identification fields	965
2-n	Variable	Software identification(s), Delimiter (ASCII "*")	234

For the 2i2o Valve Controller ECU, Byte 1 is set to 5, and the identification fields are as follows

(Part Number)*(Version)*(Date)*(Owner)*(Description)

The Axiomatic EA shows all this information in “General ECU Information”, as shown below:



Note: The information provided in the Software ID is available for any J1939 service tool which supports the PGN -SOFT.

3.3. CAN Transmit Message Defaults

This section outlines the **default** settings of the 2i2o Valve Controller CAN transmissions. Recall, however, that this is a fully programmable unit, such that all these SPNs can be sent on different PGNs if so desired.

In all the messages shown below, not all the transmitted values have an SPN assigned to them, as this ECU only uses the SPNs for diagnostic trouble codes. If the SPN is shown as N/A, this means that the associated value cannot be used to generate DTCs.

The “Measured Inputs Feedback Message” has the following default configuration.

PGN 65280 Measured Inputs Feedback

Transmission Repetition:	0s (not sent by default, configurable)
Data Length:	8
Data Page:	0
PDU Format:	254
PDU Specific:	GE PGN Supporting Information:
Default Priority:	6
Parameter Group Number:	65280 (0xFF00)

Start Position	Length	Parameter Name	SPN
1	1 byte	Universal Input 1 Measured Value	520192
2	1 byte	Universal Input 2 Measured Value	520193
3	1 byte	Power Supply Measured Value	520704
4	1 byte	Processor Temperature Measured Value	520960

The “Proportional Outputs Target Message” has the following default configuration.

PGN 65296 Proportional Outputs Target and Feedback

Transmission Repetition:	0s (not sent by default, configurable)
Data Length:	8
Data Page:	0
PDU Format:	254
PDU Specific:	GE PGN Supporting Information:
Default Priority:	6
Parameter Group Number:	65296 (0xFF10)

Start Position	Length	Parameter Name	SPN
1-2	2 byte	Universal Output 1 Target Value	N/A
3-4	2 byte	Universal Output 1 Current Feedback	520448
5-6	2 byte	Universal Output 2 Target Value	N/A
7-8	2 byte	Universal Output 2 Current Feedback	520449

Universal Input X Measured Value, X = 1 to 2

This value reflects the measured value (after filtering) of the Universal Input. Resolution and data size are automatically updated per the **"Input Sensor Type"** and **"Input Measuring Range"** setpoints of the associated input function block.

<u>0 to 1 Volt Data</u>

Data Length:	1 byte
Resolution:	0.005 V/bit, 0 offset
Data Range:	0 to 1 V

<u>0 to 2.5 Volt Data</u>

Data Length:	1 byte
Resolution:	0.01 V/bit, 0 offset
Data Range:	0 to 2.5 V

<u>0 to 5 Volt Data</u>

Data Length:	1 byte
Resolution:	0.02 V/bit, 0 offset
Data Range:	0 to 5 V

<u>0 to 10 Volt Data</u>

Data Length:	1 byte
Resolution:	0.04 V/bit, 0 offset
Data Range:	0 to 10 V

<u>0(4) to 20mA Current Data</u>

Data Length:	1 byte
Resolution:	0.08mA/bit, 0 offset
Data Range:	0 (4) to 20 mA

<u>Resistive Data</u>

Data Length:	1 byte
Resolution:	1 kΩ/bit, 0 offset
Data Range:	0 to 250 kΩ

Note: This type of input will likely be sent to a Lookup Table to convert it to another unit (i.e. Temperature) before sending it to the CAN bus.

<u>RPM Data</u>

Data Length:	2 byte
Resolution:	1 RPM/bit, 0 offset
Data Range:	0 to 64255 RPM

<u>0 to 50Hz Data:</u>

Data Length:	2 byte
Resolution:	0.01 Hz/bit, 0 offset
Data Range:	0 to 642.55 Hz

10Hz to 1kHz Data:
Data Length: 2 byte
Resolution: 0.1 Hz/bit, 0 offset
Data Range: 0 to 6425.5 Hz

100Hz to 10kHz Data:
Data Length: 2 byte
Resolution: 1 Hz/bit, 0 offset
Data Range: 0 to 64255 Hz

PWM Data
Data Length: 1 byte
Resolution: 0.4%/bit, 0 offset
Data Range: 0 to 100 %

Type: Measured
Suspect Parameter Number: 520192 to 520193 (0x7F000 to 0x7F001, all proprietary SPNs)
Parameter Group Number: 65280

Power Supply Measured Value

This value reflects the measured value of the supply voltage powering the unit

Data Length: 1 byte
Resolution: 0.4 V/bit, 0 offset
Data Range: 0 to 250 (0 to 100V)
Type: Measured
Suspect Parameter Number: 520704 (0x7F200, a proprietary SPN)
Parameter Group Number: 65280

Processor Temperature Measured Value

This value reflects the measured value of the processor by its internal temperature sensor

Data Length: 1 byte
Resolution: 1 °C/bit, -40°C offset
Data Range: 0 to 250 (-40 to 210°C)
Type: Measured
Suspect Parameter Number: 520960 (0x7F300, a proprietary SPN)
Parameter Group Number: 65280

Universal Output X Target Value, where X = 1 to 2

This value reflects the target value for the proportional output as determined by the control logic function block. The value will be in whatever unit is appropriate for the **"Output Type"** of the associated output drive block, and the CAN data size, resolution and offset is automatically updated when that setpoint is changed.

<u>Current Output</u>
Data Length: 2 byte
Resolution: 1 mA/bit, 0 offset
Data Range: 0 to 64255 mA

<u>Voltage Output</u>
Data Length: 2 byte
Resolution: 0.01 V/bit, 0 offset
Data Range: 0 to 642.55 V

<u>PWM Output</u>
Data Length: 1 byte
Resolution: 0.4 %/bit, 0 offset
Data Range: 0 to 100 %

<u>Digital Output</u>
Data Length: 1 byte
Resolution: 1 state/bit, 0 offset
Data Range:
- 00 Output is Commanded OFF
- 01 Output is Commanded ON
- 02 Error (Not applicable with this message)
- 03+ Ignored

Type: Calculated
Suspect Parameter Number: N/A
Parameter Group Number: 65296

Universal Output X Current Feedback, where X = 1 to 2

This value reflects the measured current sourced from the proportional output drive circuit. For non-current "Output Type" selections, this value will reflect the actual value being applied to the output port.

Data Length: 2 byte
Resolution: 1mA/bit, 0 offset
Data Range: 0 to 64255 mA
Type: Measured
Suspect Parameter Number: 520448 to 520449 (0x7F100 to 0x7F101, all proprietary SPNs)
Parameter Group Number: 65296

3.4. CAN Receive Message Defaults

This section outlines the **default** settings of the 2i2o Valve Controller CAN receive channels, used as inputs to the various function blocks supported by this ECU. Recall, however, that this is a fully programmable unit, such that all these SPNs can be received on different PGNs if so desired.

In all the messages shown below, none of the received values have an SPN assigned to them, as this ECU only uses the SPNs for diagnostic trouble codes. To have the 2i2o react to a DTC sent by another ECU on the network on a DM1, use the DTC React Function block instead.

By default, all CAN Receive Messages are disabled, as they are not part of the factory set logic shown in Figure 28. However, should one or all of them be enabled by the user, the default settings for each message are as outlined in this section.

The "Output Control Data Message" has the following default configuration.

PGN 65408 Output Control Data

Transmission Repetition:	100ms (default, configurable)
Data Length:	8
Data Page:	0
PDU Format:	254
PDU Specific:	GE PGN Supporting Information:
Default Priority:	6
Parameter Group Number:	65408 (0xFF80)

Start Position	Length	Parameter Name	SPN
1	1 byte	Output 1 Command Input Data	N/A
2	1 byte	Output 2 Command Input Data	N/A
3	1 byte	Output 1 Closed Loop Feedback	N/A
4	1 byte	Output 2 Closed Loop Feedback	N/A
5	1 byte	Output 1 Enable Input Data	N/A
6	1 byte	Output 2 Enable Input Data	N/A
7	1 byte	Output 1 Override Input Data	N/A
8	1 byte	Output 2 Override Input Data	N/A

Output X Command Input Data, where X = 1 to 2

Default value used when a *'Received CAN J1939 Message'* is used as the control source for the Output Control logic function block (or another block that is linked to the output control.)

Data Length:	1 byte
Resolution:	0.4%/bit, 0 offset
Data Range:	0 to 100 %
Type:	Input
Suspect Parameter Number:	N/A
Parameter Group Number:	65408

Output X Closed Loop Feedback, where X = 1 to 2

Default value used when a *'Received CAN J1939 Message'* is used as the feedback source for a proportional output PID control function block.

Data Length:	1 byte
Resolution:	0.4 [Data]/bit, 0 offset
Data Range:	0 to 100.0
Type:	Input
Suspect Parameter Number:	N/A
Parameter Group Number:	65408

Output X Enable Input Data, where X = 1 to 2

Default value used when a *'Received CAN J1939 Message'* is used as an enable source for the Output Control logic function block.

00	Enable Signal is OFF
01	Enable Signal is ON
02	Error (automatic disable)
03+	Ignored

Data Length:	2 bits
Resolution:	1 state/bit, 0 offset
Type:	Input
Suspect Parameter Number:	N/A
Parameter Group Number:	65408

Output X Override Input Data, where X = 1 to 2

Default value used when a *'Received CAN J1939 Message'* is used as the override source for the Output Control logic function block.

00	Override Signal is OFF
01	Override Signal is ON
02	Error (override disabled)
03+	Ignored

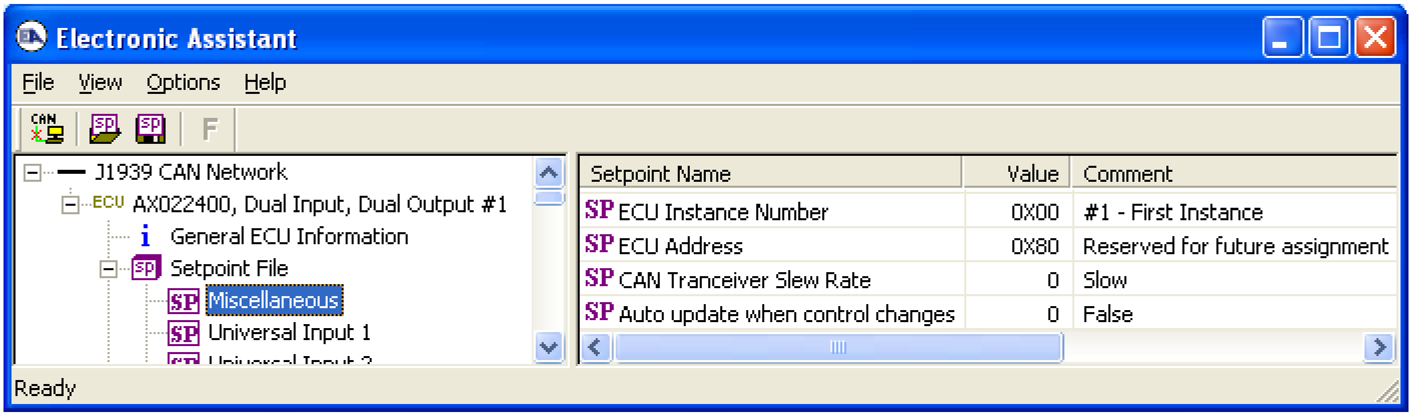
Data Length:	2 bits
Resolution:	1 state/bit, 0 offset
Type:	Input
Suspect Parameter Number:	N/A
Parameter Group Number:	65408

4. ECU SETPOINTS ACCESSED WITH THE AXIOMATIC ELECTRONIC ASSISTANT

Many setpoints have been referenced throughout this manual. This section describes in detail each setpoint, and their defaults and ranges. For more information on how each setpoint is used by the 2i2o Valve Controller, refer to the relevant section of the User Manual.

4.1. Miscellaneous Setpoints

The Miscellaneous setpoints primarily deal with the CAN Network. Refer to the notes for more information about each setpoint.



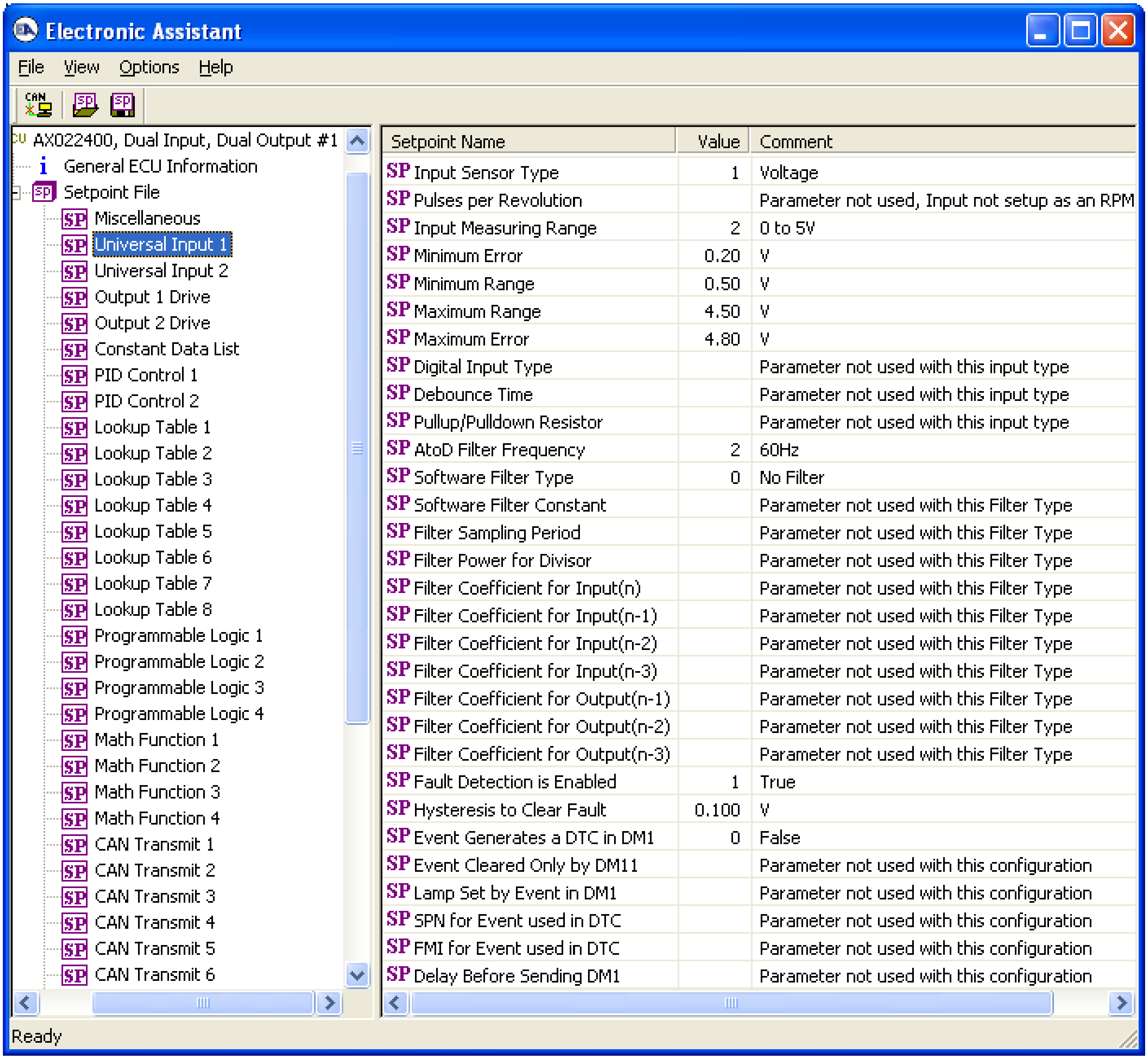
Screen Capture of Default Miscellaneous Setpoints

Name	Range	Default	Notes
ECU Instance Number	Drop List	0, #1 – First Instance	Per J1939-81
ECU Address	0 to 253	128	Preferred address for a self-configurable ECU
CAN Transceiver Slew Rate	0, Slow 1, Fast	0, Slow	A fast slew rate allows for better transmission over long CAN bus wires, but it can also increases the noise emitted on an unshielded network.
Auto update when control changes	0, False 1, True	0, False	When True, any change to an logic block that is used as a control source for another logic block will cause the block using the source as an input to automatically be updated with new the new range. Use this feature with caution as it will cause any previous changes to the controlled block to be overwritten.

If non-default values for the **"ECU Instance Number"** or **"ECU Address"** are used, they will be mirrored during a setpoint file flashing, and will only take effect once the entire file has been downloaded to the unit. After the setpoint flashing is complete, the unit will claim the new address and/or re-claim the address with the new NAME. If these setpoints are changing, it is recommended to close and re-open the CAN connection on the Axiomatic EA after the file is loaded, such that only the new NAME and address appear in the J1939 CAN Network ECU list.

4.2. Universal Input Setpoints

The Universal Input function block is defined in Section 1.2. Please refer there for detailed information about how all these setpoints are used.

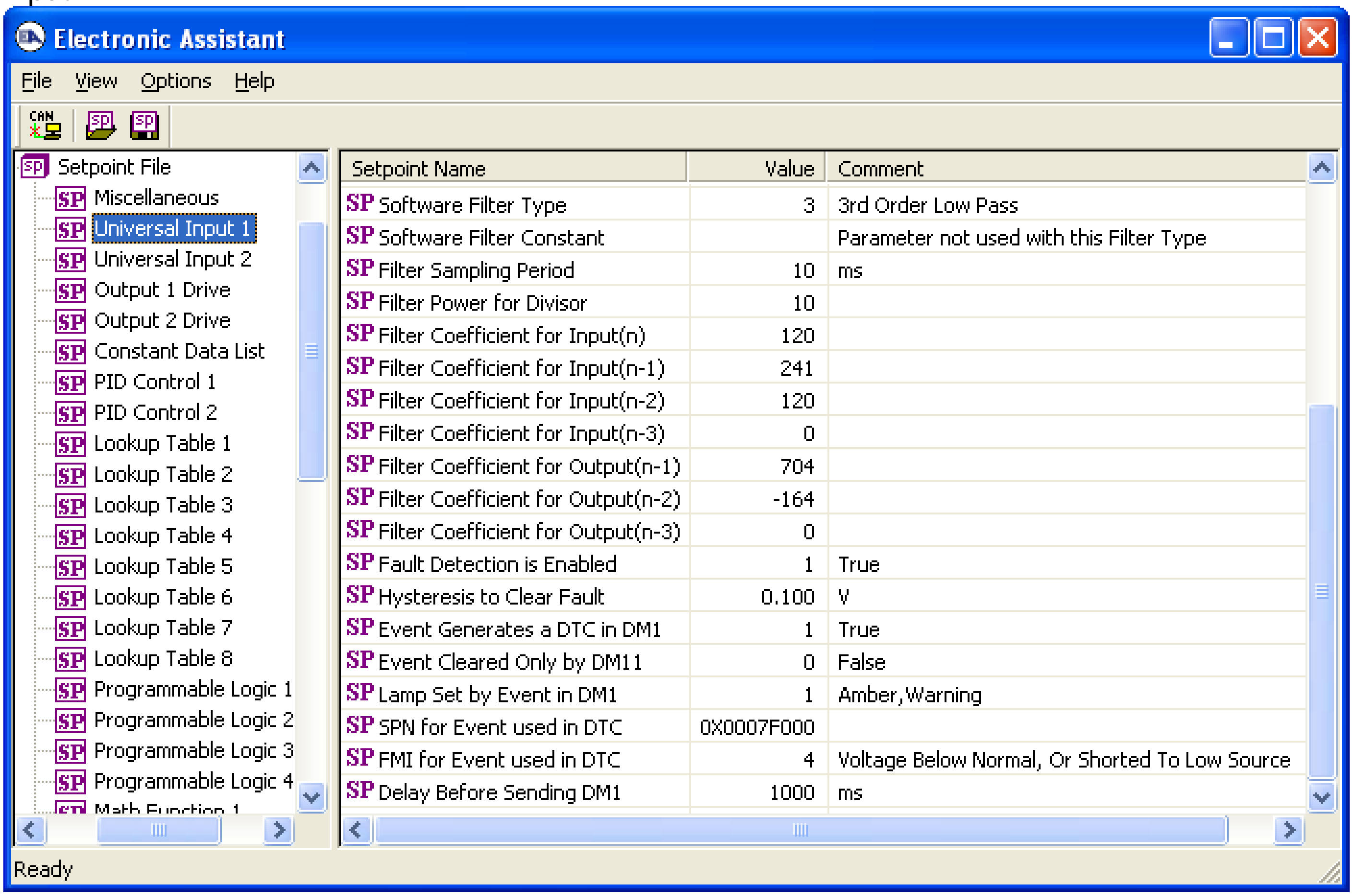


Setpoint Name	Value	Comment
SP Input Sensor Type	1	Voltage
SP Pulses per Revolution		Parameter not used, Input not setup as an RPM
SP Input Measuring Range	2	0 to 5V
SP Minimum Error	0.20	V
SP Minimum Range	0.50	V
SP Maximum Range	4.50	V
SP Maximum Error	4.80	V
SP Digital Input Type		Parameter not used with this input type
SP Debounce Time		Parameter not used with this input type
SP Pullup/Pulldown Resistor		Parameter not used with this input type
SP AtoD Filter Frequency	2	60Hz
SP Software Filter Type	0	No Filter
SP Software Filter Constant		Parameter not used with this Filter Type
SP Filter Sampling Period		Parameter not used with this Filter Type
SP Filter Power for Divisor		Parameter not used with this Filter Type
SP Filter Coefficient for Input(n)		Parameter not used with this Filter Type
SP Filter Coefficient for Input(n-1)		Parameter not used with this Filter Type
SP Filter Coefficient for Input(n-2)		Parameter not used with this Filter Type
SP Filter Coefficient for Input(n-3)		Parameter not used with this Filter Type
SP Filter Coefficient for Output(n-1)		Parameter not used with this Filter Type
SP Filter Coefficient for Output(n-2)		Parameter not used with this Filter Type
SP Filter Coefficient for Output(n-3)		Parameter not used with this Filter Type
SP Fault Detection is Enabled	1	True
SP Hysteresis to Clear Fault	0.100	V
SP Event Generates a DTC in DM1	0	False
SP Event Cleared Only by DM11		Parameter not used with this configuration
SP Lamp Set by Event in DM1		Parameter not used with this configuration
SP SPN for Event used in DTC		Parameter not used with this configuration
SP FMI for Event used in DTC		Parameter not used with this configuration
SP Delay Before Sending DM1		Parameter not used with this configuration

Screen Capture of Default Universal Input 1 Setpoints

By default, a 0-5V input type does not apply any software filtering, although the ADC does apply a 60Hz pre-filter. Also, as stated in Section 1.14, the controller is programmed to be self-contained, and does not send or receive CAN data, including diagnostic messages. While fault detection is enabled on Universal Inputs 1 to 2 (as they are used to control Outputs 1 to 2 respectively), faults will not generate DM1 messages.

Therefore, for completeness, a screen capture has been included to also show the default values of the Filtering Function Block (Section 1.8) and the Diagnostic Function Block (Section 1.9) for this input.



Screen Capture of Universal Input 1 with Active Software Filter, and Sending DM1 Messages

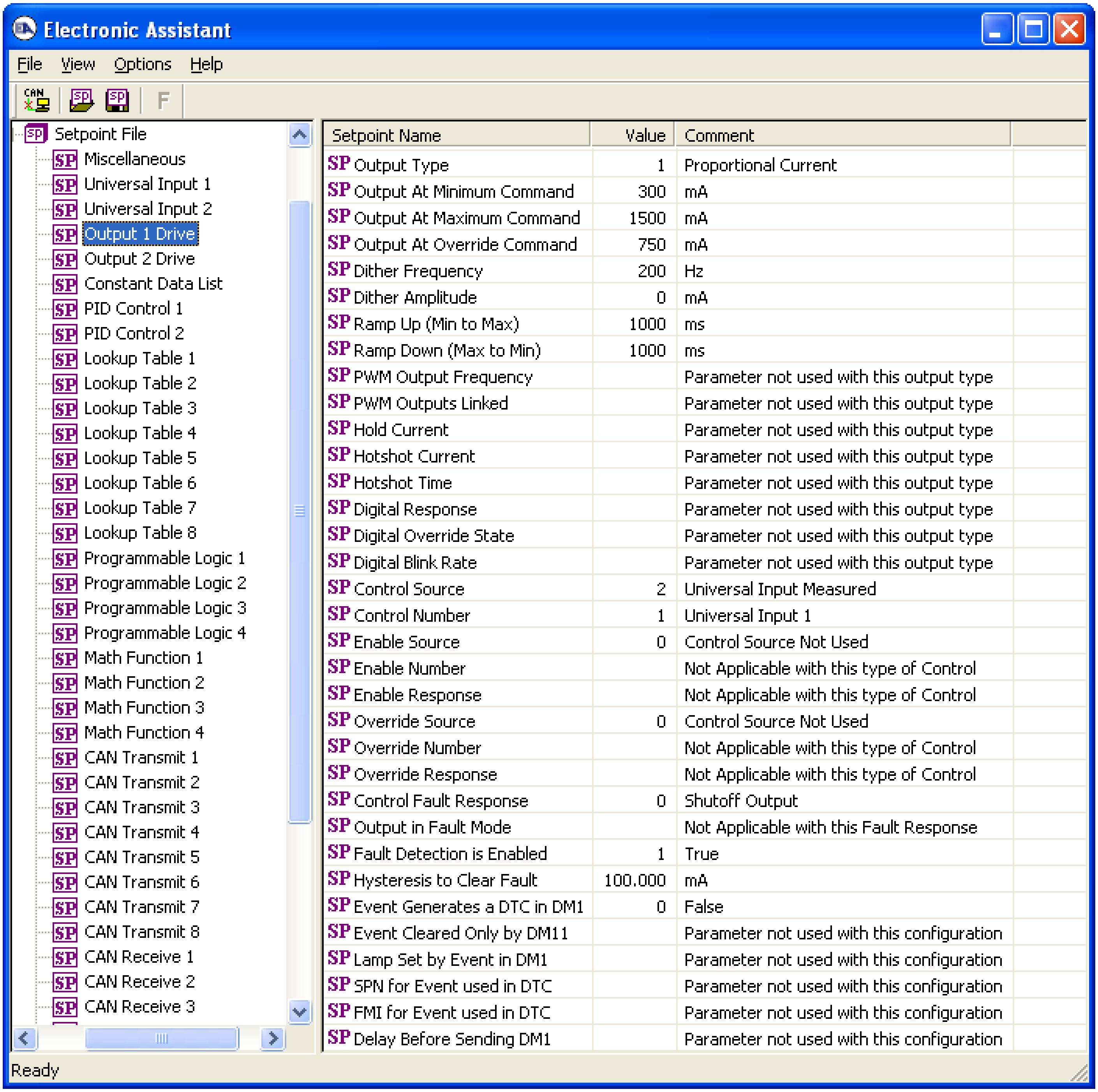
Type	Range	Minimum Error	Minimum Range	Maximum Range	Maximum Error	Hysteresis
Voltage	0-1V	0.05 [V]	0.10 [V]	1.00 [V]	1.10 [V]	0.025 [V]
	0-2.5V	0.10 [V]	0.25 [V]	2.50 [V]	2.75 [V]	0.050 [V]
	0-5V	0.20 [V]	0.50 [V]	4.50 [V]	4.80 [V]	0.100 [V]
	0-10V	0.20 [V]	0.50 [V]	9.50 [V]	9.80 [V]	0.200 [V]
Current	0-20mA	0.00 [mA]	0.00 [mA]	20.00 [mA]	22.00 [mA]	0.250 [mA]
	4-20mA	1.00 [mA]	4.00 [mA]	20.00 [mA]	22.00 [mA]	0.250 [mA]
Resistive	N/A	0.025 [kΩ]	0.100 [kΩ]	200.0 [kΩ]	250.0 [kΩ]	0.010 [kΩ]
Frequency/ RPM	RPM	50 [RPM]	100 [RPM]	3000 [RPM]	3300 [RPM]	10 [RPM]
	0.5 to 50 Hz	0.10 [Hz]	1.00 [Hz]	50.00 [Hz]	55.00 [Hz]	0.200 [Hz]
	10Hz to 1kHz	1.0 [Hz]	10.0 [Hz]	1000.0 [Hz]	1100.0 [Hz]	5.0 [Hz]
	100Hz to 10kHz	10 [Hz]	100 [Hz]	10000 [Hz]	10500 [Hz]	10 [Hz]
PWM	Low Freq [<1kHz]	1.00 [%]	5.00 [%]	95.00 [%]	99.00 [%]	1.00 [%]
	High Freq [>100Hz]	1.00 [%]	5.00 [%]	95.00 [%]	99.00 [%]	1.00 [%]
Digital	N/A	N/A	0.00	1.00	N/A	N/A

Input Setpoint Defaults Based on Type and Range

Name	Range	Default	Notes
Input Sensor Type	Drop List	1, Voltage	See Table 1
Pulses per Revolution	0 to 1,000	1	0 means measured in Hz
Input Measuring Range	Drop List	2, 0 to 5V	See Tables 2 thru 5 for ranges based on type
Minimum Error	0.00 to Minimum Range	0.20V	Default values for these setpoints are dependent on Type and Range. These values are automatically updated by the ECU when the Type/Range setpoints are changed. See Table above for the complete list.
Minimum Range	Minimum Error to Maximum Range	0.50V	
Maximum Range	Minimum Range to Maximum Error	4.50V	
Maximum Error	Maximum Range to Limit (see notes)	4.80V	
Digital Input Type	Drop List	0, Normal On/Off	See Table 7
Debounce Time	Drop List	2, 1.78us	See Table 6
Pullup/Pulldown Resistor	Drop List	2, 10kOhm Pulldown	See Table 7
AtoD Filter Frequency	Drop List	2, 60Hz	0 = None, 1 = 50Hz, 2 = 60Hz, 3 = Both This filter used on all values measured by the ADC
Software Filter Type	Drop List	0, No Filter	See Table 19
Software Filter Constant	1 to 1,000	10	Used with Moving or Repeating Average Filters
Filter Sampling Period	1 to 1,000 ms	10 ms	Used with 3rd Order Low Pass, otherwise called every 1ms
Filter Power for Divisor	0 to 16	10	2^Power is used in fixed point math for 3rd Order Low Pass
Filter Coefficient for Input(n)	-10,000 to 10,000	120	Used with 3rd Order Low Pass X = Y*2^Power
Filter Coefficient for Input(n-1)	-10,000 to 10,000	241	See above
Filter Coefficient for Input(n-2)	-10,000 to 10,000	120	See above
Filter Coefficient for Input(n-3)	-10,000 to 10,000	0	See above
Filter Coefficient for Ouput(n-1)	-10,000 to 10,000	704	See above
Filter Coefficient for Ouput(n-2)	-10,000 to 10,000	-164	See above
Filter Coefficient for Ouput(n-3)	-10,000 to 10,000	0	See above
Fault Detection is Enabled	False or True	1, True	See Section 1.9
Hysteresis to Clear Fault	0.01 to 250.00	0.100V	See Section 1.9
Event Generates a DTC in DM1	False or True	False	See Section 1.9
Event Cleared Only by DM11	False or True	False	See Section 1.9
Lamp Set by Event in DM1	Drop List	1, Amber, Warning	See Section 1.9
SPN for Event used in DTC	1 to 524287	520192 ($7F000) Uin1 520192 ($7F001) Uin2	See Section 1.9 and 3.3
FMI for Event used in DTC	Drop List	4, Voltage Below Normal	See Section 1.9 (low FMI)
Delay Before Sending DM1	0 to 60,000 ms	1000 ms	See Section 1.9

4.3. Output Drive Setpoints

The Output Drive function block is defined in Section 1.3. Please refer there for detailed information about how all these setpoints are used.



Setpoint Name	Value	Comment
SP Output Type	1	Proportional Current
SP Output At Minimum Command	300	mA
SP Output At Maximum Command	1500	mA
SP Output At Override Command	750	mA
SP Dither Frequency	200	Hz
SP Dither Amplitude	0	mA
SP Ramp Up (Min to Max)	1000	ms
SP Ramp Down (Max to Min)	1000	ms
SP PWM Output Frequency		Parameter not used with this output type
SP PWM Outputs Linked		Parameter not used with this output type
SP Hold Current		Parameter not used with this output type
SP Hotshot Current		Parameter not used with this output type
SP Hotshot Time		Parameter not used with this output type
SP Digital Response		Parameter not used with this output type
SP Digital Override State		Parameter not used with this output type
SP Digital Blink Rate		Parameter not used with this output type
SP Control Source	2	Universal Input Measured
SP Control Number	1	Universal Input 1
SP Enable Source	0	Control Source Not Used
SP Enable Number		Not Applicable with this type of Control
SP Enable Response		Not Applicable with this type of Control
SP Override Source	0	Control Source Not Used
SP Override Number		Not Applicable with this type of Control
SP Override Response		Not Applicable with this type of Control
SP Control Fault Response	0	Shutoff Output
SP Output in Fault Mode		Not Applicable with this Fault Response
SP Fault Detection is Enabled	1	True
SP Hysteresis to Clear Fault	100.000	mA
SP Event Generates a DTC in DM1	0	False
SP Event Cleared Only by DM11		Parameter not used with this configuration
SP Lamp Set by Event in DM1		Parameter not used with this configuration
SP SPN for Event used in DTC		Parameter not used with this configuration
SP FMI for Event used in DTC		Parameter not used with this configuration
SP Delay Before Sending DM1		Parameter not used with this configuration

Screen Capture of Default Output 1 Drive Setpoints

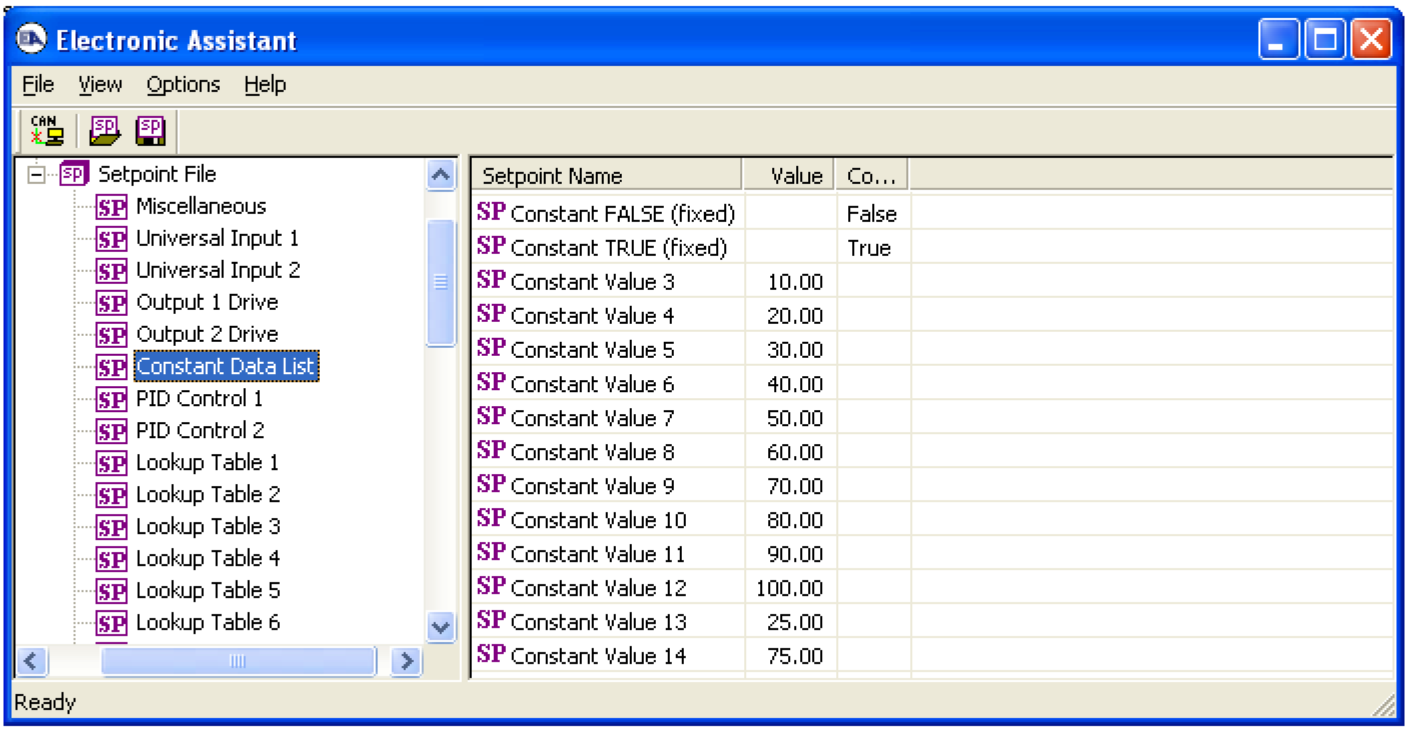
Name	Range	Default	Notes
Output Type	Drop List	1, Proportional Current	See Table 10
Output at Minimum Command	0 to Limit	300 mA	Current, Limit = 3000 mA Voltage, Limit = 48.0V PWM, Limit = 100.0 %
Output at Maximum Command	0 to Limit	1500 mA	
Output at Override Command	0 to Limit	750 mA	
Dither Frequency	50 to 400 Hz	200 Hz	
Dither Amplitude	0 to 500 mA	0 mA	0 = No Dithering
Ramp Up (Min to Max)	0 to 10,000 ms	1000 ms	
Ramp Down (Max to Min)	0 to 10,000 ms	1000 ms	
PWM Output Frequency	1Hz to 25,000 Hz	500Hz	Only configurable if no current output types are enabled for any output.
PWM Outputs Linked	False or True	False	Only configurable on Output 1 in PWM mode
Hold Current	0 to 2500 mA	500 mA	Only used with Hotshot output type
Hotshot Current	0 to 2500 mA	2000 mA	
Hotshot Time	500 to 10,000 ms	1000 ms	
Digital Response	Drop List	0, Normal On/Off	See Table 11
Digital Override State	0 or 1	1, ON	
Digital Blink Rate	100 to 5000 ms	1000 ms	Only used with 'Blinking' response
Control Source	Drop List	2 Universal Input Measured	See Table 25
Control Number	Per Source	Same as Output Drive Number	See Table 25
Enable Source	Drop List	0, Control Source Not Used	See Table 25
Enable Number	Per Source	N/A	See Table 25
Enable Response	Drop List	N/A	See Table 8
Override Source	Drop List	0, Control Source Not Used	See Table 25
Override Number	Per Source	N/A	See Table 25
Override Response	Drop List	N/A	0 = Override When On 1 = Override When Off
Control Fault Response	Drop List	0, Shutoff Output	See Table 9
Output in Fault Mode	Depends on Output Type	500 mA	See limits on ranges per output type outlined above
Fault Detection is Enabled	False or True	True	See Section 1.9
Hysteresis to Clear Fault	0.01 to 250.00	100.00 mA	See Section 1.9
Event Generates a DTC in DM1	False or True	False	See Section 1.9
Event Cleared Only by DM11	False or True	False	See Section 1.9
Lamp Set by Event in DM1	Drop List	1, Amber, Warning	See Section 1.9
SPN for Event used in DTC	1 to 524287	520448 ($7F100) Out1 520449 ($7F101) Out2	See Section 1.9 and 3.3
FMI for Event used in DTC	Drop List	5, Open Circuit	See Section 1.9 (low FMI)
Delay Before Sending DM1	0 to 60,000 ms	1000 ms	See Section 1.9

4.4. Constant Data List Setpoints

The Constant Data List function block is provided to allow the user to select values as desired for various logic block functions. Throughout this manual, various references have been made to constants, as summarized in the examples listed below.

a) PID Control: Constant **"PID Target Command"** in a *'Setpoint Control'* application
b) Programmable Logic: Constant **"Table X = Condition Y, Argument 2"**, where X and Y = 1 to 3
c) Math Function: Constant **"Math Input X"**, where X = 1 to 4

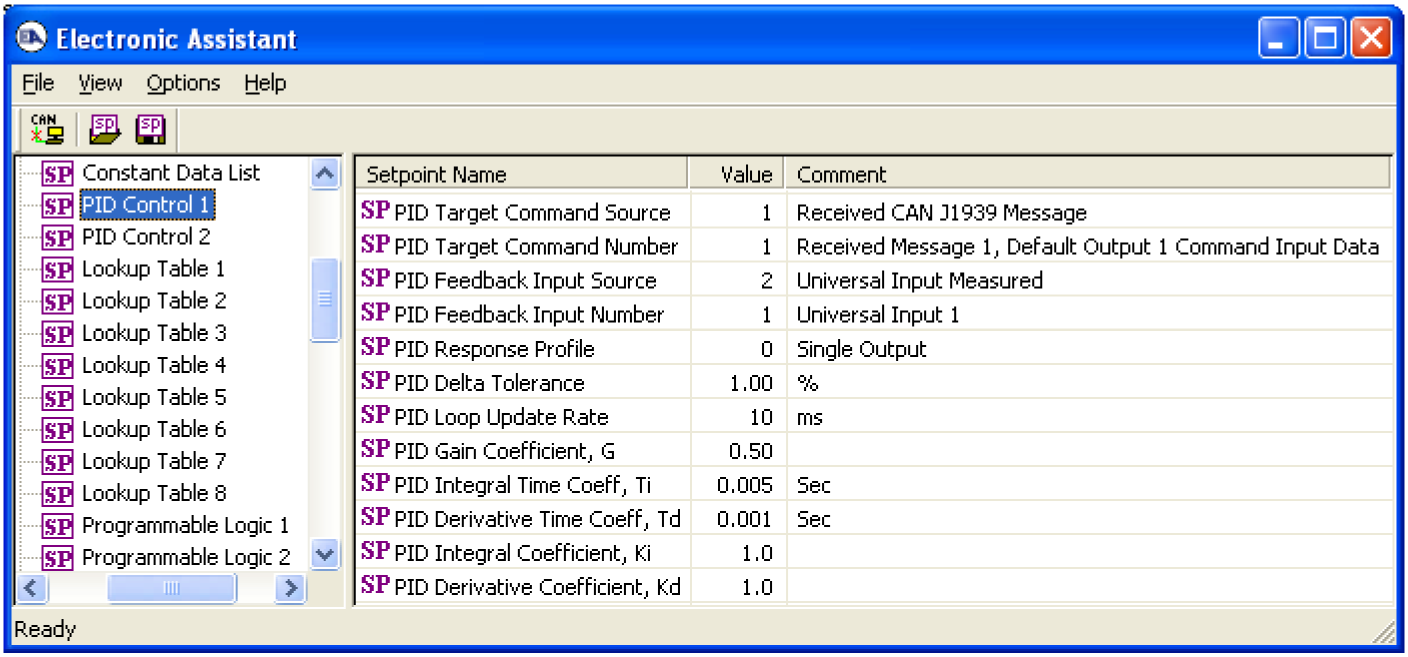
The first two constants are fixed values of 0 (False) and 1 (True) for use in binary logic. The remaining 12 constants are fully user programmable to any value between +/- 1,000,000. The default values (shown below) are arbitrary and should be configured by the user as appropriate for their application.



Screen Capture of Default Constant Data List Setpoints

4.5. PID Control

The PID Control function block is defined in Section 1.4. Please refer there for detailed information about how all these setpoints are used.



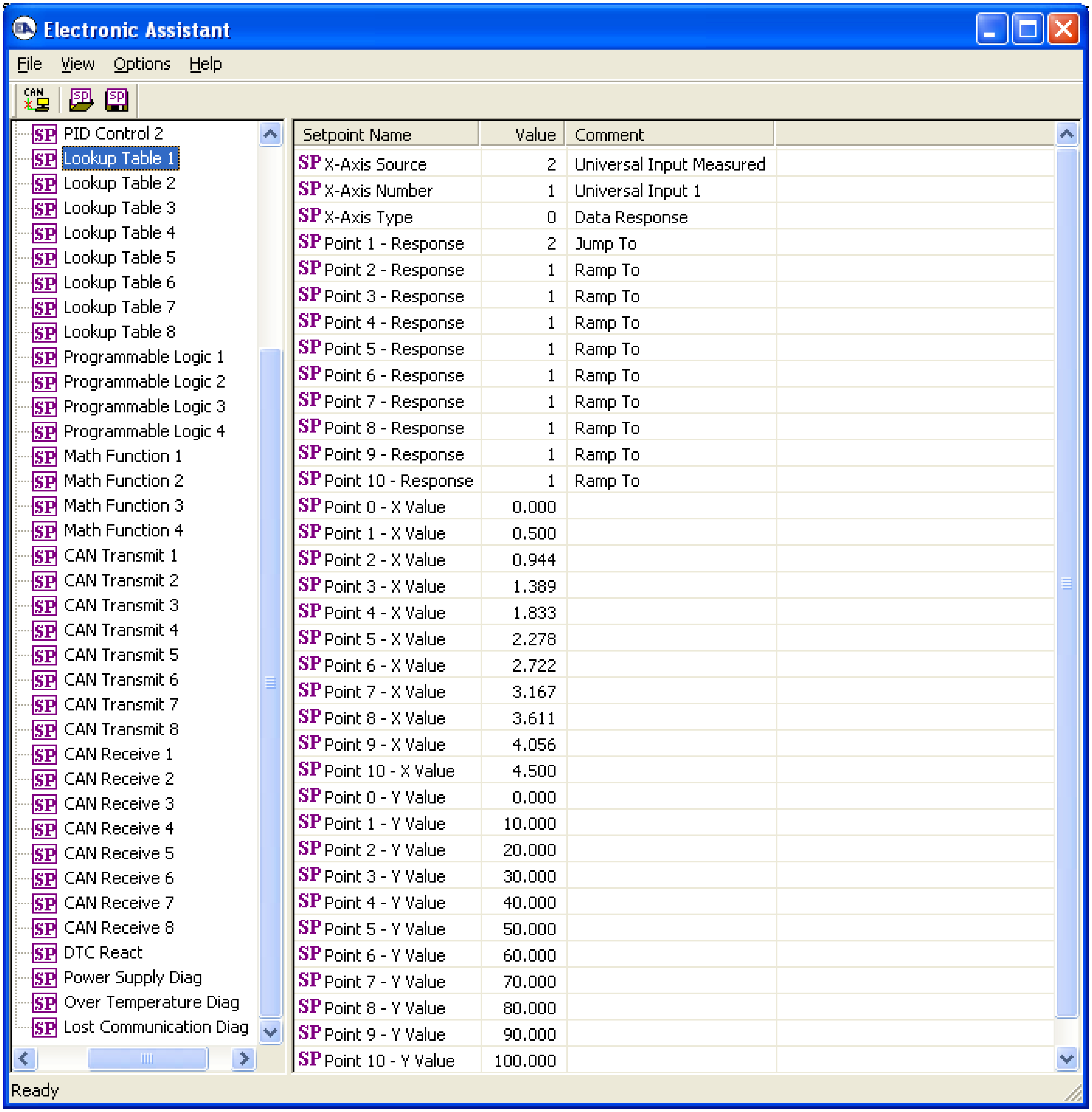
Screen Capture of Default PID Control 1 Setpoints

Note: In the screen capture shown above, the "PID Target Command Source" has been changed from its default value in order to enable the function block.

Name	Range	Default	Notes
PID Target Command Source	Drop List	0, Control Source Not Used	See Table 25
PID Target Command Number	Per Source	1, N/A with default value	See Table 25
PID Feedback Input Source	Drop List	1, Universal Input Measured	See Table 25
PID Feedback Input Number	Per Source	1, Universal Input 1 PID1 2, Universal Input 2 PID2	See Table 25
PID Response Profile	Drop List	0, Single Output	See Table 12
PID Delta Tolerance	0 to 100	1.00	%
PID Loop Update Rate	1 to 60,000 ms	10 ms	1 ms resolution
PID Gain Coefficient, G	0.1 to 10	0.5	See Figure 13
PID Integral Time Coeff, Ti	0.001 to 10 Sec	0.005 Sec	0.001 Sec (1ms) resolution
PID Derivative Time Coeff, Td	0.001 to 10 Sec	0.001 Sec	0.001 Sec (1ms) resolution
PID Integral Coefficient, Ki	0 to 10	1.0	0 disables integral, PD ctrl
PID Derivative Coefficient, Kd	0 to 10	1.0	0 disables derivative, PI ctrl

4.6. Lookup Table Setpoints

The Programmable Logic function block is defined in Section 1.5. Please refer there for detailed information about how all these setpoints are used. As this function block's X-Axis defaults are defined by the **"X-Axis Source"** selected from Table 25, there is nothing further to define in terms of defaults and ranges beyond that which is described in Section 1.5 Recall, the X-Axis values will be automatically updated if the min/max range of the selected source is changed.



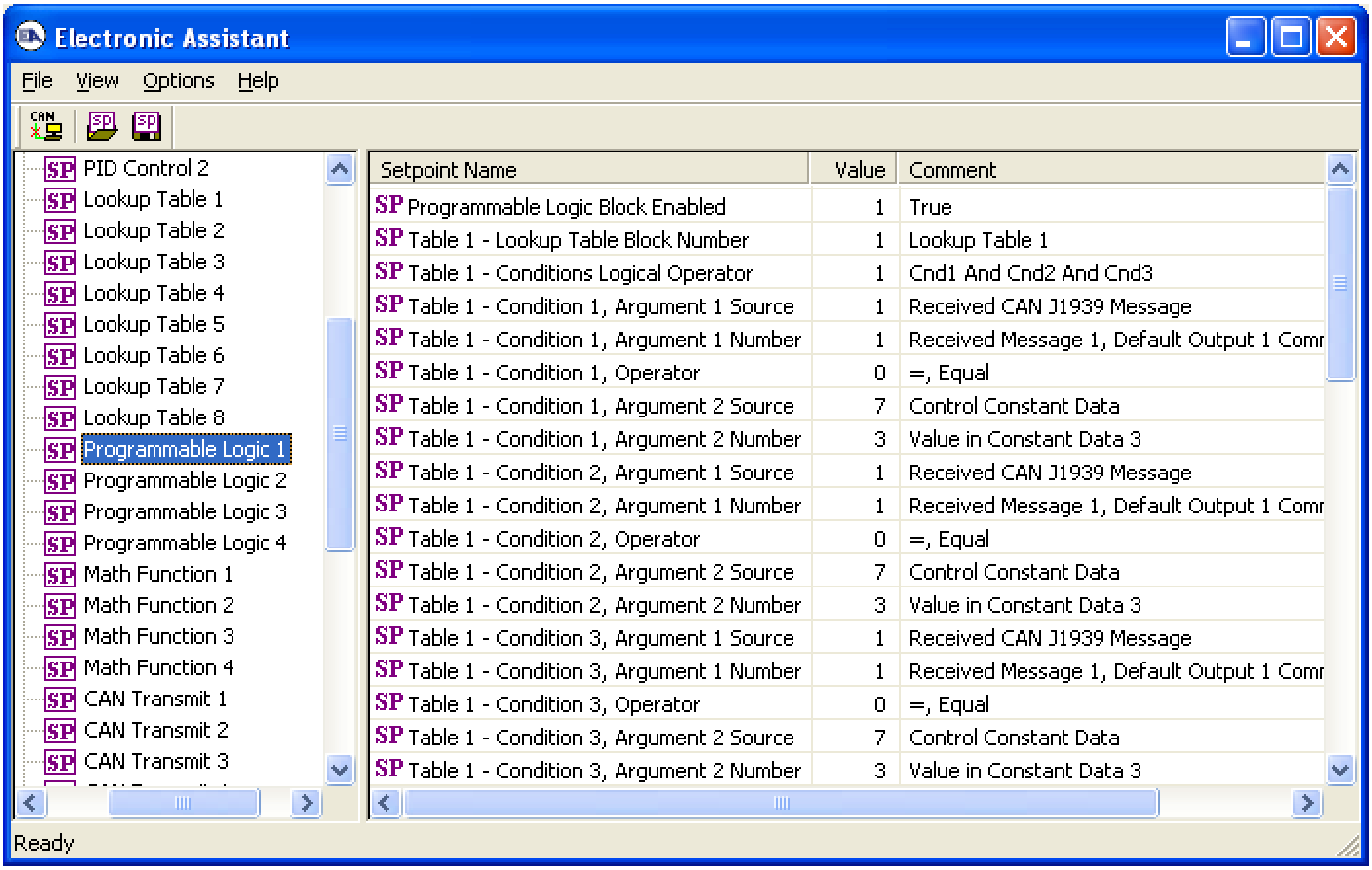
Screen Capture of Example Lookup Table 1 Setpoints

Note: In the screen capture shown above, the "X-Axis Source" has been changed from its default value in order to enable the function block.

4.7. Programmable Logic Setpoints

The Programmable Logic function block is defined in Section 1.6. Please refer there for detailed information about how all these setpoints are used.

As this function block is disabled by default, there is nothing further to define in terms of defaults and ranges beyond that which is described in Section 1.6. The screen capture below shows how the setpoints referenced in that section appear on the Axiomatic EA.



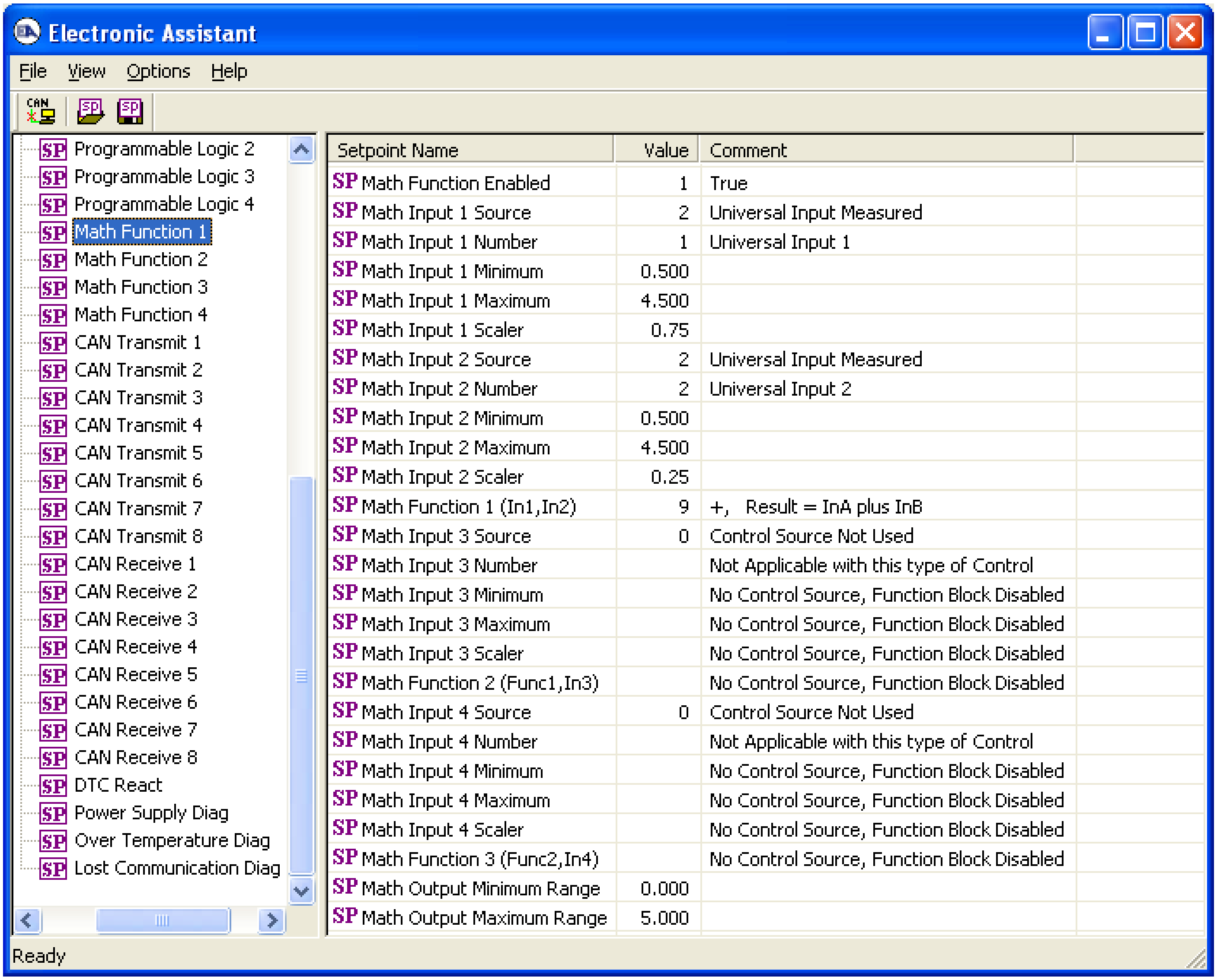
Screen Capture of Default Programmable Logic 1 Setpoints

Note: In the screen capture shown above, the "Programmable Logic Block Enabled" has been changed from its default value in order to enable the function block.

Note: The default values for the Argument1, Argument 2 and Operator are all the same across all the Programmable Logic function blocks, and must therefore be changed by the user as appropriate before this can be used.

4.8. Math Function Setpoints

The Math Function block is defined in Section 1.7. Please refer there for detailed information about how all these setpoints are used.



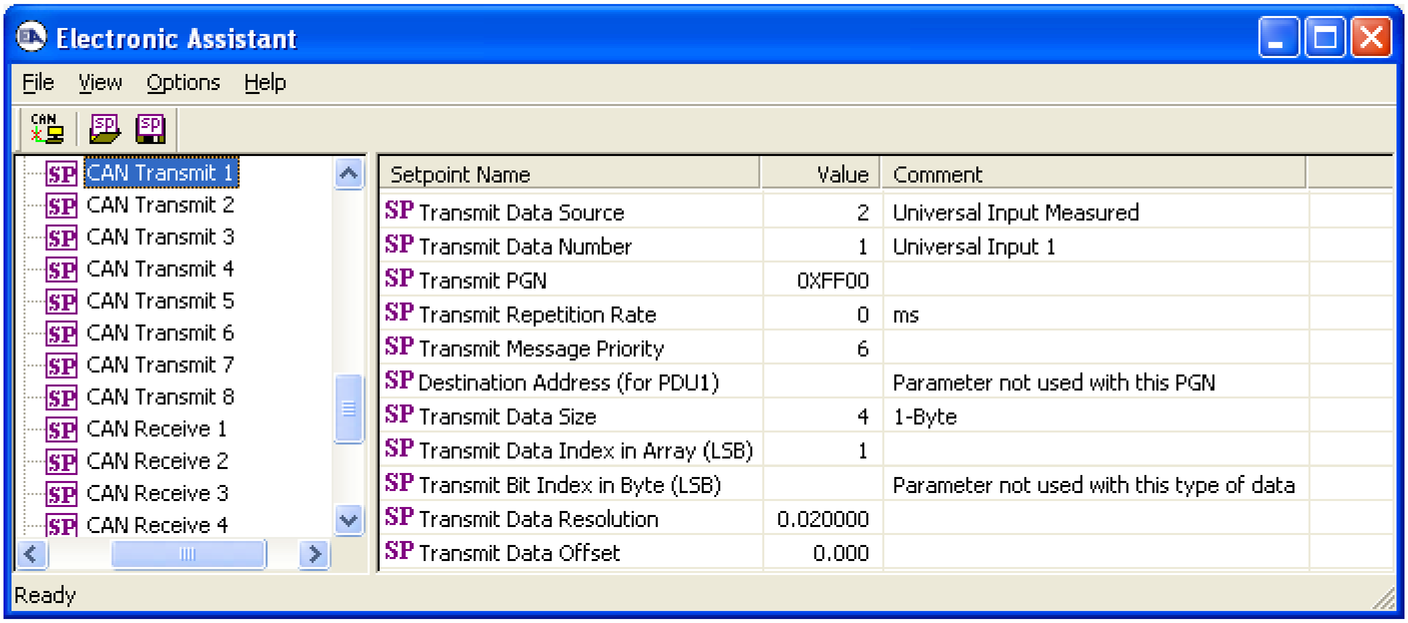
Screen Capture of Example Math Function 1 Setpoints

Note: In the screen capture shown above, the "Math Function Enabled" has been changed from its default value in order to enable the function block. Other setpoints have also been changed from default values in order to illustrate how the block might look when functional, as per the example outlined in Section 1.7.

Name	Range	Default	Notes
Math Function Enabled	Drop List	0, False	True or False
Math Input X Source (X = 1 to 4)	Drop List	0, Control Source Not Used	See Table 25
Math Input X Number	Per Source	0	See Table 25
Math Input X Minimum	-10^6 to 10^6	0.0	Converts input to a percentage before use in the calculation.
Math Input X Maximum	-10^6 to 10^6	100.0	
Math Input X Scaler	-1.00 to 1.00	1.00	See Section 1.7
Math Output Minimum Range	-10^6 to 10^6	0.0	Converts calculation from a percentage value to the desired physical unit.
Math Output Maximum Range	-10^6 to 10^6	100.0	

4.9. CAN Transmit Setpoints

The CAN Transmit function block is defined in Section 1.11, with addition information in Section 3.3. Please refer there for detailed information about how all these setpoints are used.



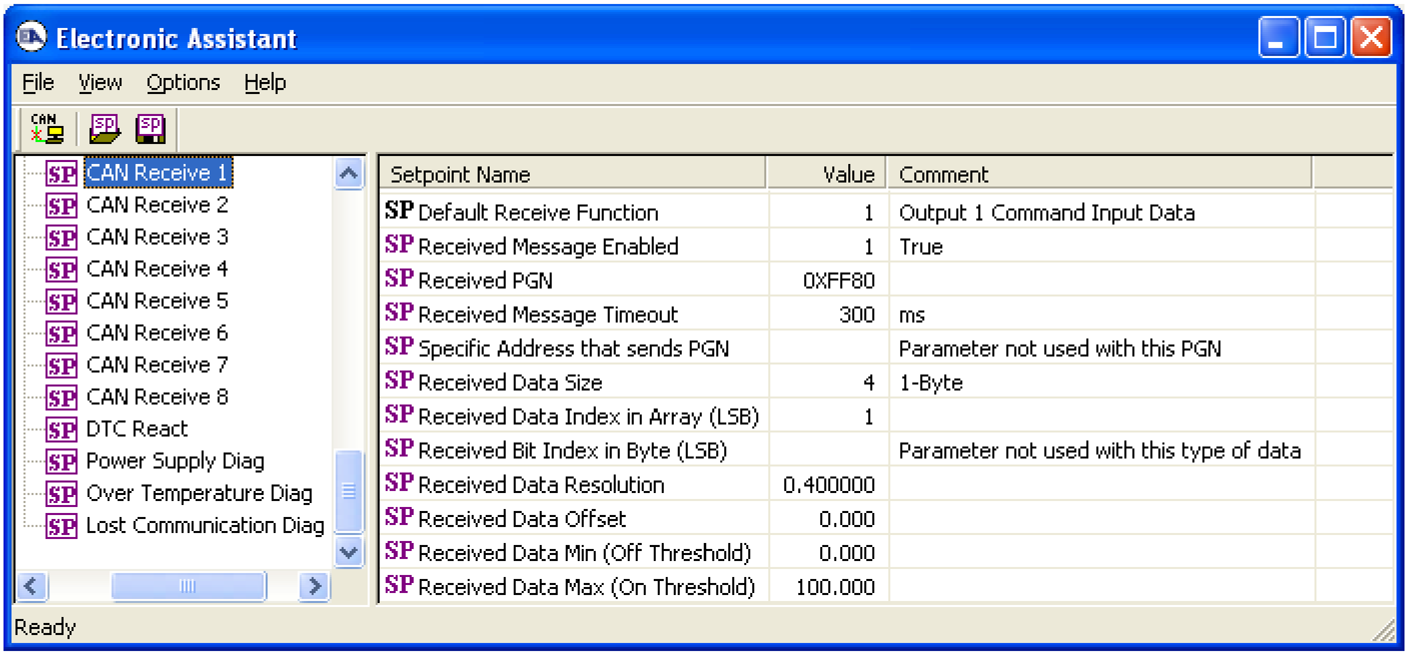
Screen Capture of Default CAN Transmit 1 Setpoints

Name	**Range**	**Default**	**Notes**
Transmit Data Source	Drop List	Different for each	See Table 22 for defaults
Transmit Data Number	Per Source	Different for each	See Table 22 for defaults
Transmit PGN	0 to 65535	65280 ($FF00) Txd1 to 4 65296 ($FF10) Txd5 to 8	See Section 3.3 for defaults
Transmit Repetition Rate	0 to 60,000 ms	0	0ms disables transmit
Transmit Message Priority	0 to 7	6	Proprietary B Priority
Destination Address (for PDU1)	0 to 255	254 (0xFE, Null Address)	Not used by default
Transmit Data Size	Drop List	Different for each	0 = Not Used (disabled) 1 = 1-Bit 2 = 2-Bits 3 = 4-Bits 4 = 1-Byte 5 = 2-Bytes 6 = 4-Bytes See Section 3.3 for defaults
Transmit Data Index in Array (LSB)	1 to 9-DataSize	Different for each	See Section 3.3 for defaults
Transmit Bit Index in Byte (LSB)	1 to 9-BitSize	Different for each	Only used with Bit Data Types
Transmit Data Resolution	-10^6 to 10^6	Different for each	See Section 3.3 for defaults
Transmit Data Offset	-10^4 to 10^4	Different for each	See Section 3.3 for defaults

Recall that when multiple messages are sent on the same PGN, only the LOWEST Indexed channel's *'Repetition Rate'* will be used. This means that even if a non-zero value is selected on a higher channel, but the lowest is still 0, no message will be sent.

4.10. CAN Receive Setpoints

The CAN Receive function block is defined in Section 1.12, with addition information in Section 3.4. Please refer there for detailed information about how all these setpoints are used.



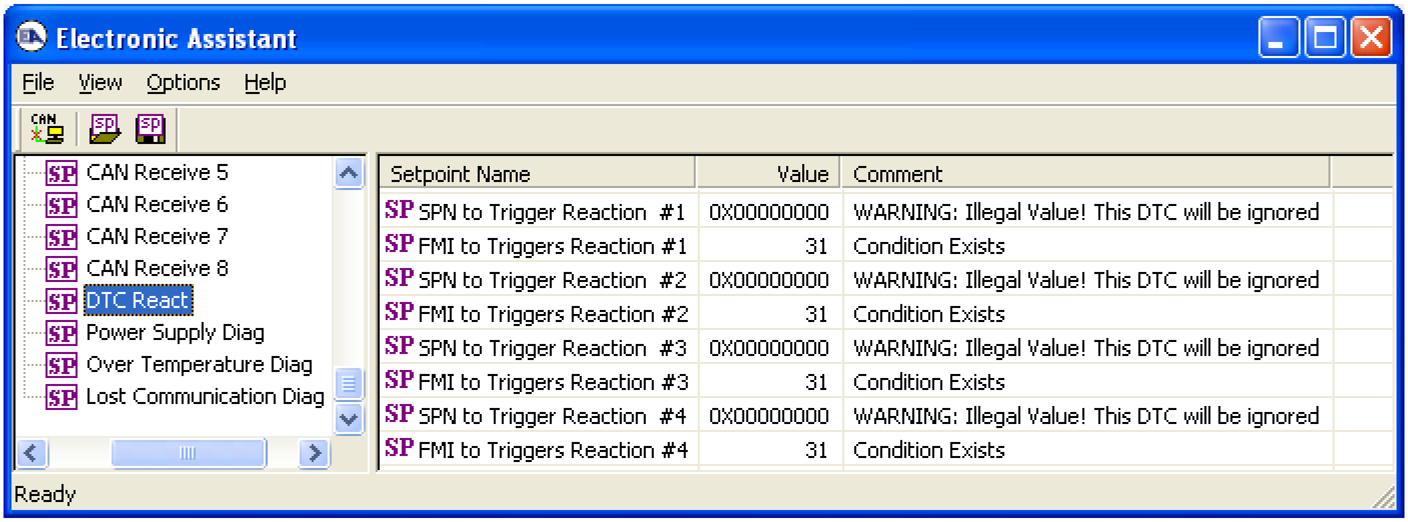
Screen Capture of Default CAN Receive 1 Setpoints

Note: By default, the "Received Message Enabled" is actually False, but it was changed to get the screen capture. Otherwise, all setpoints read "No Control Source, Function Block Disabled"

Name	**Range**	**Default**	**Notes**
Default Receive Function	Drop List	Different for each	Read-Only Setpoint See Table 23 for defaults
Received Message Enabled	False or True	False	See note above
Received PGN	0 to 65535	65408 ($FF80)	See Section 3.4 for defaults
Received Message Timeout	0 to 60,000 ms	300 ms	Expects all data at 100ms
Specific Address that sends PGN	0 to 255	254 (0xFE, Null Addr)	Not used by default
Receive Data Size	Drop List	Different for each	0 = Not Used (disabled) 1 = 1-Bit 2 = 2-Bits 3 = 4-Bits 4 = 1-Byte 5 = 2-Bytes 6 = 4-Bytes See Section 3.4 for defaults
Receive Data Index in Array (LSB)	1 to 9-DataSize	Different for each	See Section 3.4 for defaults
Receive Bit Index in Byte (LSB)	1 to 9-BitSize	Different for each	Only used with Bit Data Types
Receive Data Resolution	-10^6 to 10^6	Different for each	See Section 3.4 for defaults
Receive Data Offset	-10,000 to 10,000	Different for each	See Section 3.4 for defaults
Received Data Min (Off Threshold)	-10^6 to Max	0.0	See Section 3.4 for defaults
Received Data Max (On Threshold)	-10^4 to 10^4	100.0	See Section 3.4 for defaults

4.11. DTC React Setpoints

The DTC React function block is defined in Section 1.10. Please refer there for detailed information about how all these setpoints are used.



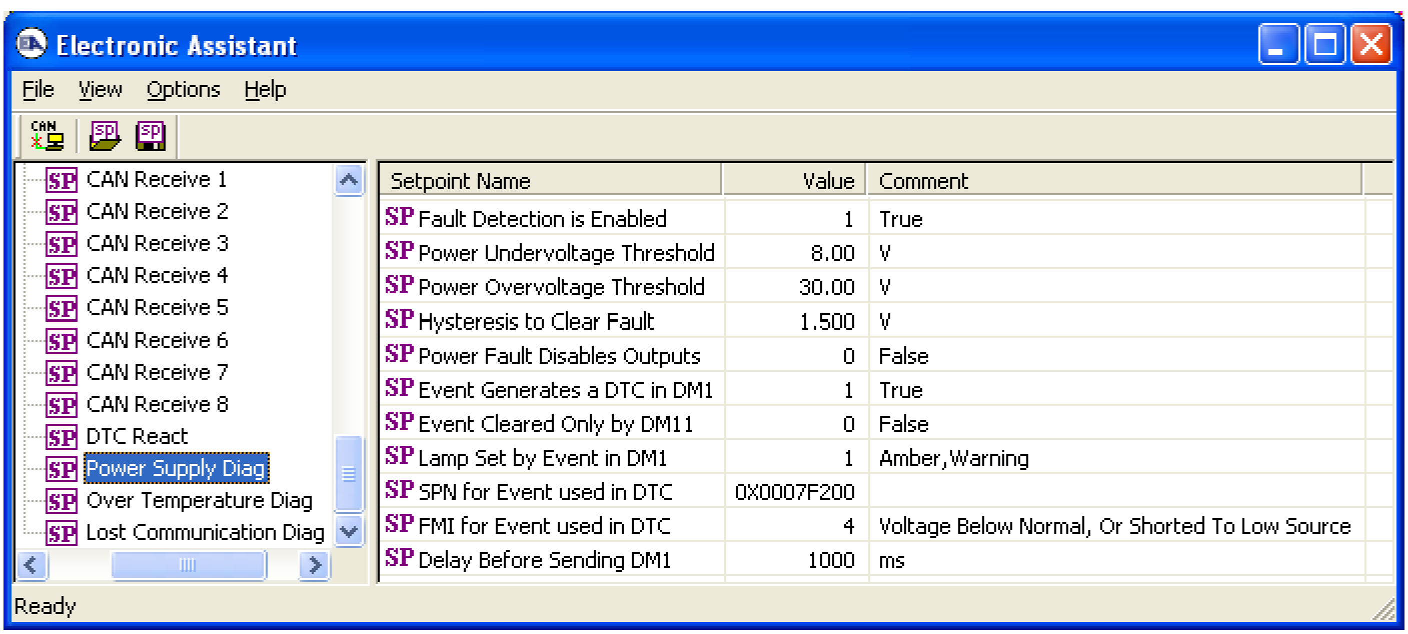
Screen Capture of Default DTC React Setpoints

Name	Range	Default	Notes
SPN to Trigger Reaction #X	0 to 524287	0	0 is an illegal value, and disables the DTC
FMI to Trigger Reaction #X	Drop List	31, Condition Exists	Supports all FMIs in the J1939 standard

Where X = 1 to 4

4.12. [Additional] Diagnostic Setpoints

Here, the [Additional] diagnostics are Power Supply, Over Temperature and Lost Communication. The Diagnostic function block is defined in Section 1.9. Please refer there for detailed information about how all these setpoints are used.



Screen Capture of Default Power Supply Diag Setpoints

Note: By default, the “Event Generates at DTC in DM1” is actually False, but it was changed to get the screen capture.

Name	Range	Default		Notes
Fault Detection is Enabled	False or True	True		See Section 1.9
Power Undervoltage Threshold *	7.5V to Over V	8.00 V		
Power Overvoltage Threshold *	Under V to 65V	30.00 V		
Over Temperature Shutdown **	0 to 125 DegC	125.00 Deg C		
Hysteresis to Clear Fault	0.01 to 250.00	1.50 V 5.00 DegC N/A	Power Temp Comm	
Power Fault Disables Outputs *	False or True	False		
Event Generates a DTC in DM1	False or True	False		See Section 1.9
Event Cleared Only by DM11	False or True	False		See Section 1.9
Lamp Set by Event in DM1	Drop List	1, Amber, Warning		See Section 1.9
SPN for Event used in DTC	1 to 524287	520704 ($7F200) 520960 ($7F300) 521216 ($7F400)	Power Temp Comm	See Section 1.9 and Section 3.3
FMI for Event used in DTC	Drop List	4, Voltage Below Normal 0, Data Above Normal–Most Sever 19, Received Data Error	Power Temp Comm	See Section 1.9
Delay Before Sending DM1	0 to 60,000 ms	1000 ms		See Section 1.9

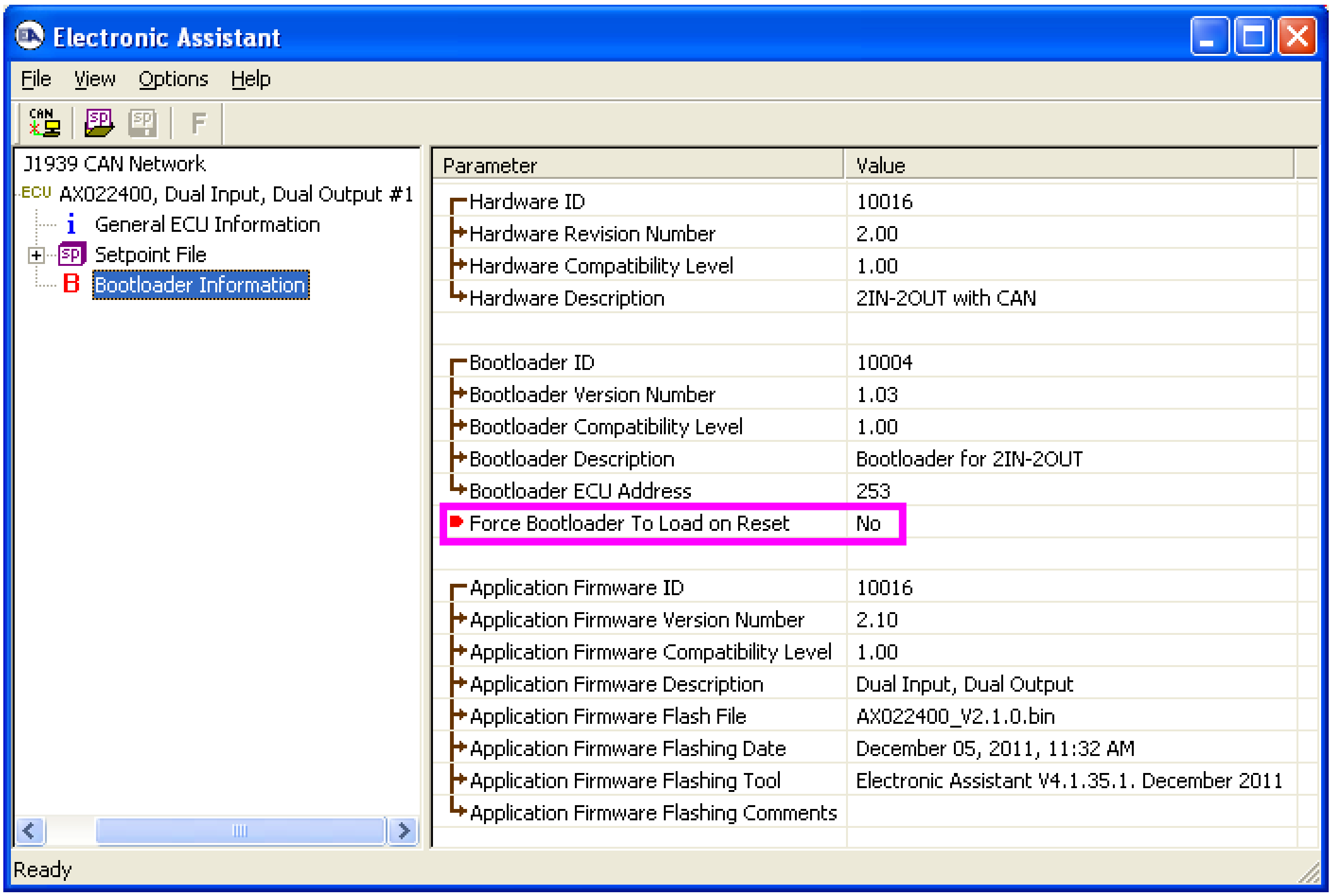
* *Only used with Power Supply Diag group*
** *Only used with Over Temperature Diag group*

5. REFLASHING OVER CAN WITH THE AXIOMATIC EA BOOTLOADER

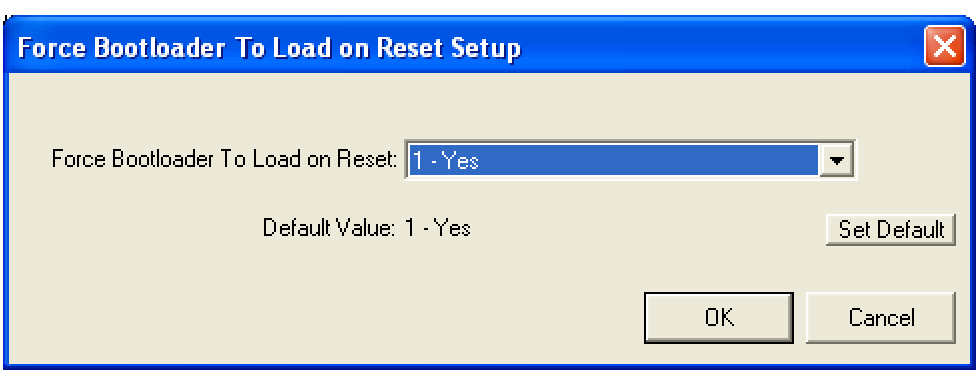
The AX0224x0 can be upgraded with new application firmware using the **Bootloader Information** section. This section details the simple step-by-step instructions to upload new firmware provided by Axiomatic onto the unit via CAN, without requiring it to be disconnected from the J1939 network.

Note: To upgrade the firmware use Axiomatic Electronic Assistant V4.2.39.0 or higher.

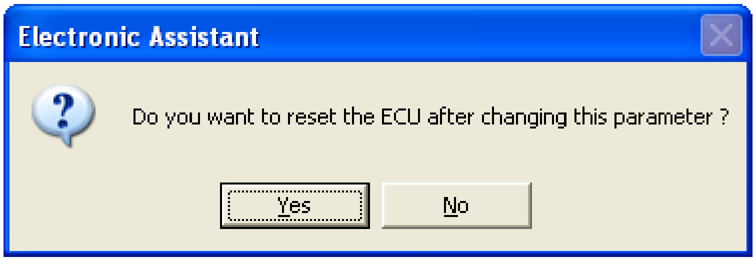
1. When the Axiomatic EA first connects to the ECU, the **Bootloader Information** section will display the following information.



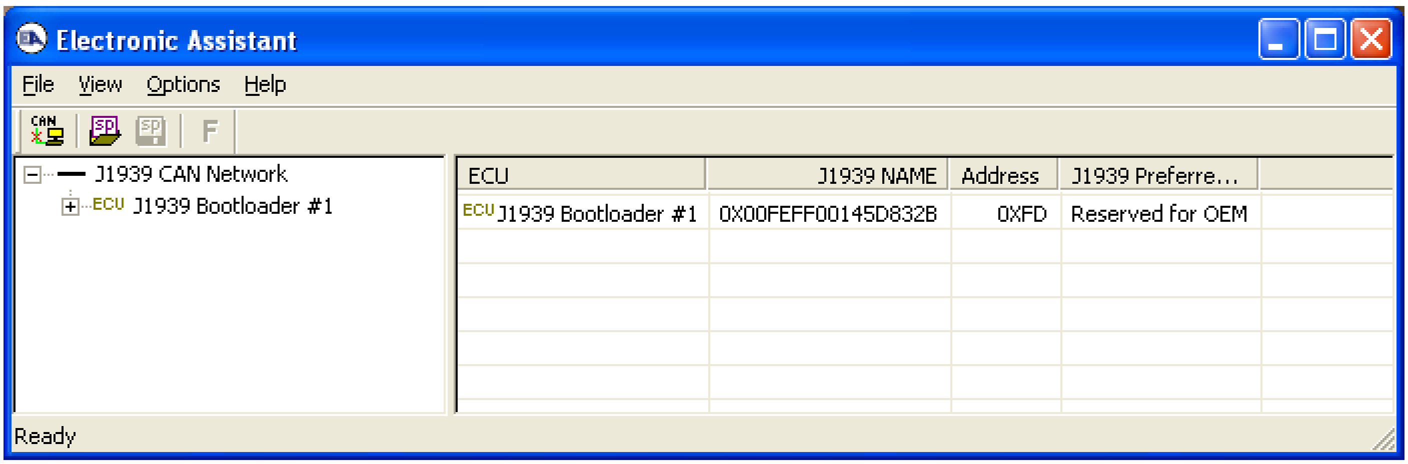
2. To use the bootloader to upgrade the firmware running on the ECU, change the variable **"Force Bootloader To Load on Reset"** to *Yes.*

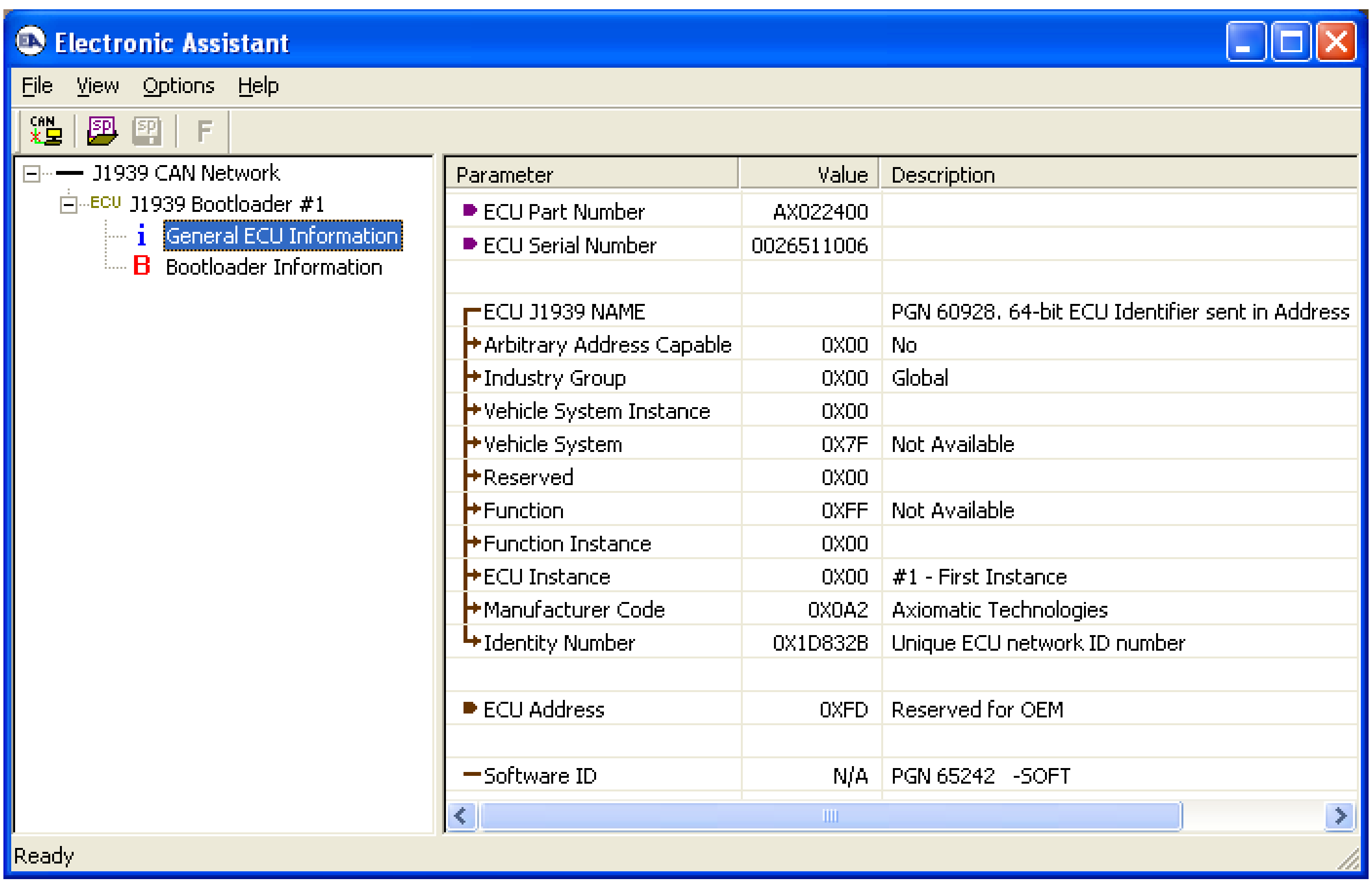


3. When the prompt box asks if you want to reset the ECU, select *Yes*.



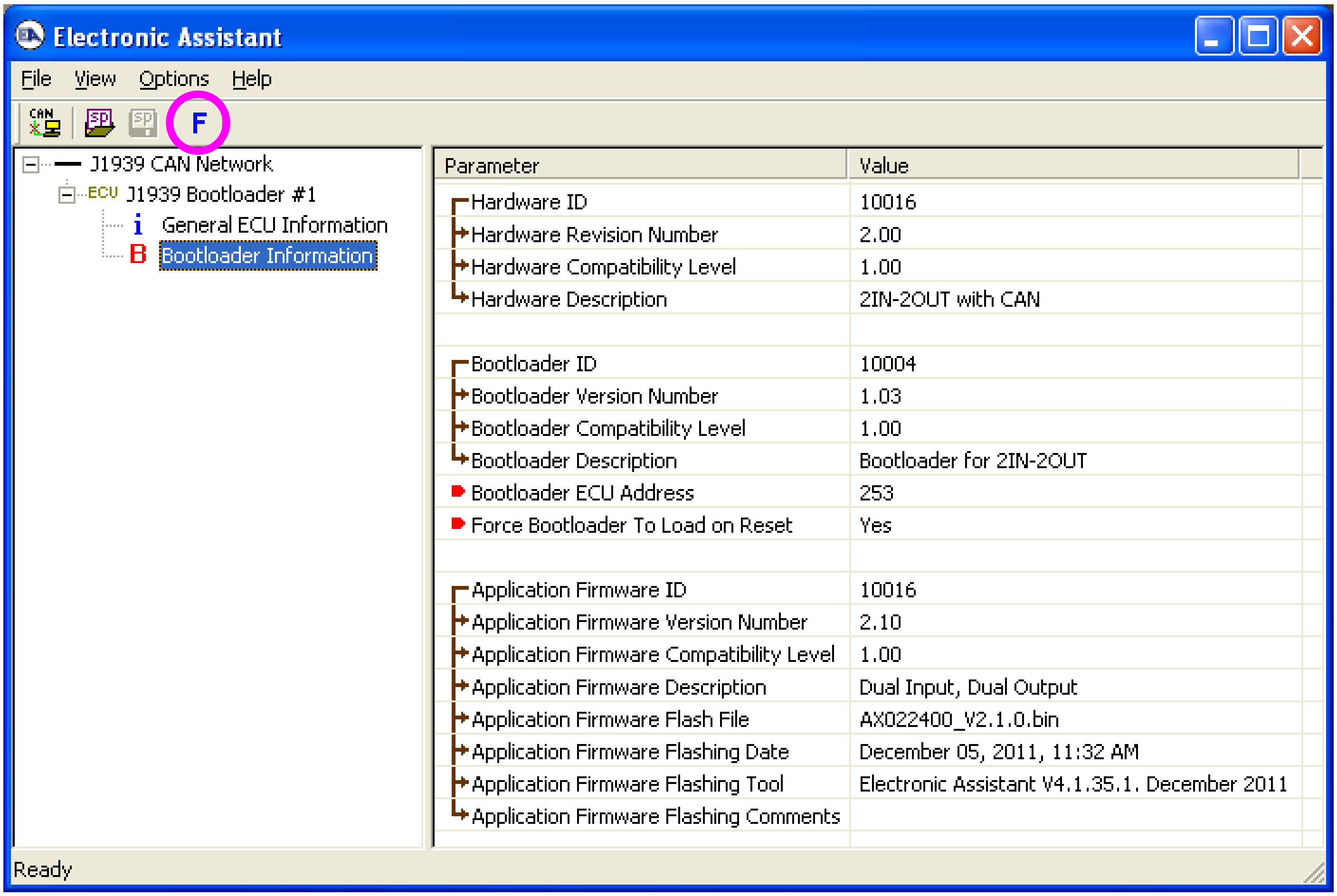
4. Upon reset, the ECU will no longer show up on the J1939 network as an AX0224x0 but rather as **J1939 Bootloader #1**.





Note that the bootloader is NOT Arbitrary Address Capable. This means that if you want to have multiple bootloaders running simultaneously (not recommended) you would have to manually change the address for each one before activating the next, or there will be address conflicts, and only one ECU would show up as the bootloader. Once the 'active' bootloader returns to regular functionality, the other ECU(s) would have to be power cycled to re-activate the bootloader feature.

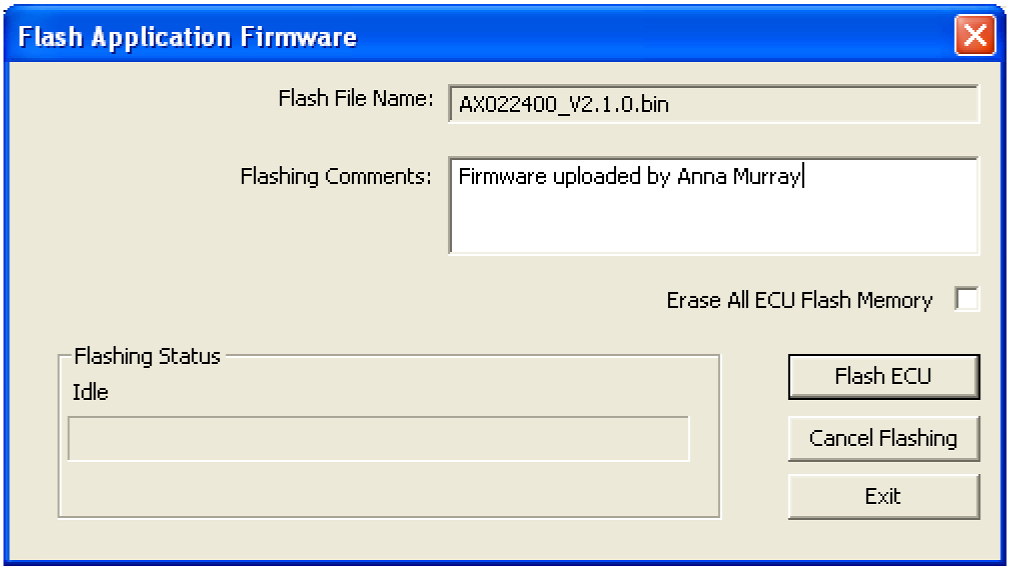
5. When the **Bootloader Information** section is selected, the same information is shown as when it was running the AX0224x0 firmware, but in this case the Flashing feature has been enabled.



6. Select the Flashing button and navigate to where you had saved the **AX0224x0_Vx.y.z.bin** file sent from Axiomatic. (Note: only binary (.bin) files can be flashed using the Axiomatic EA tool)

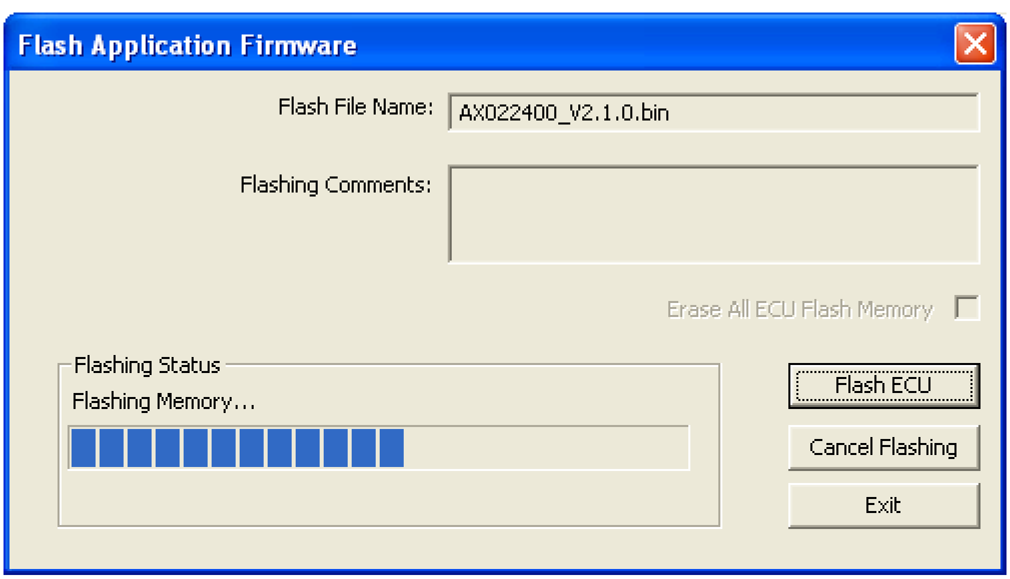
7. Once the Flash Application Firmware window opens, you can enter comments such as "Firmware upgraded by [Name]" if you so desire. This is not required, and you can leave the field blank if you do not want to use it.

   Note: You do not have to date-stamp or timestamp the file, as this is done automatically by the Axiomatic EA tool when you upload the new firmware.

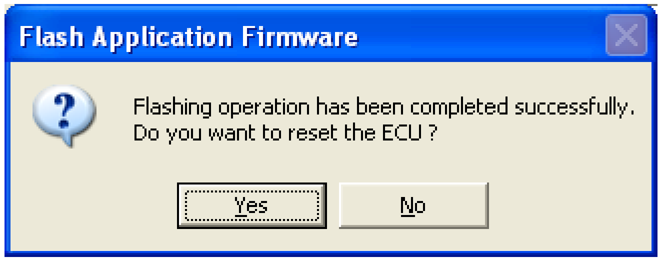


WARNING: Do not check the "Erase All ECU Flash Memory" box unless instructed to do so by your Axiomatic contact. Selecting this will erased ALL data stored in non-volatile flash, including the calibration done by Axiomatic during factory testing. It will also erase any configuration of the setpoints that might have been done to the ECU and reset all setpoints to their factory defaults. By leaving this box unchecked, none of the setpoints will be changed when the new firmware is uploaded.

8. A progress bar will show how much of the firmware has been sent as the upload progresses. The more traffic there is on the J1939 network, the longer the upload process will take.



9. Once the firmware has finished uploading, a message will popup indicating the successful operation. If you select to reset the ECU, the new version of the AX0224x0 application will start running, and the ECU will be identified as such by the Axiomatic EA. Otherwise, the next time the ECU is power-cycled, the AX0224x0 application will run rather than the bootloader function.



Note: If at any time during the upload the process is interrupted, the data is corrupted (bad checksum) or for any other reason the new firmware is not correct, i.e. bootloader detects that the file loaded was not designed to run on the hardware platform, the bad or corrupted application will not run. Rather, when the ECU is reset or power-cycled the **J1939 Bootloader** will continue to be the default application until valid firmware has been successfully uploaded into the unit.

APPENDIX A – Diagnostic Trouble Codes Table

J1939/73 DM1/2		Description	Detection conditions	Associated Setpoint Group	Restricted Function
SPN[1]/FMI	Lamp[1]				
520192-04	Amber	Universal Input 1 Minimum Error	Measured Input < Minimum Error	Universal Input 1	Note 2
520192-03	Amber	Universal Input 1 Maximum Error	Measured Input > Maximum Error	Universal Input 1	Note 2
520193-04	Amber	Universal Input 2 Minimum Error	Measured Input < Minimum Error	Universal Input 2	Note 2
520193-03	Amber	Universal Input 2 Maximum Error	Measured Input > Maximum Error	Universal Input 2	Note 2
520448-05	Amber	Output 1 Current Open Circuit	Output 1 Current FB error > 100mA (when ON)	Output 1 Drive	None
520448-06	Amber	Output 1 Current Short Circuit (Hardware)	Output 1 Current FB > ~4500mA	Output 1 Drive	Out 1 Off
520449-05	Amber	Output 2 Current Open Circuit	Output 2 Current FB error > 100mA (when ON)	Output 2 Drive	None
520449-06	Amber	Output 2 Current Short Circuit (Hardware)	Output 2 Current FB > ~4500mA	Output 2 Drive	Out 2 Off
520704-04	Amber	Power Supply Under Voltage	Measured Power Supply < Power Undervoltage Threshold	Power Supply Diag	Outs Off
520704-03	Amber	Power Supply Over Voltage	Measured Power Supply > Power Overvoltage Threshold	Power Supply Diag	Outs Off
520960-00	Amber	Processor Over Temperature	Did not receive an expected CAN message within timeout period	Over Temperature Diag	Outs Off
521216-19	Amber	Lost Communication	Did not receive an expected CAN message within timeout period	CAN Receive X	Out Off

Note 1: SPN, FMI and Lamp Type are user configurable parameters for all DTC. Default values are shown in this table.
Note 2: An input used as a control or feedback input for an output control block will disable the corresponding output when it is in an error state.
Note 3: All diagnostic trouble code (DTC) can be disabled at any time by setting the "XXX Fault Generates DTCs" to FALSE. If this setpoint for all diagnostics are FALSE, the DM1 message will not be sent every 1 second (i.e. in a standalone 2i2o mode where no CAN is connected)

APPENDIX B – Technical Specifications

Input Specifications

Universal Inputs	Two fully independent inputs selectable as : Voltage; Current; Resistive; Frequency; RPM; PWM; or Digital types 12-bit Analog to Digital (voltage, current, resistive) 15-bit Timer (frequency, RPM, PWM) Inputs are sampled multiple times per millisecond. Protected against shorts to GND or +Vps (up to 60V) Response time to change at the input 2ms +/- 1ms (without software filtering) unless otherwise noted. Voltage Types: 1mV resolution, accuracy +/- 1% error 0-1V or 0-2.5V have 1MΩ Impedance 0-5V has ~135 kΩ Impedance, 0-10V has ~127 kΩ Impedance Current Types: 1uA resolution, accuracy +/- 2% error Current sense resistor 249Ω Resistive Input: 1Ω resolution, accuracy +/- 1% error Self-calibrating for range of 30 Ω to 250 kΩ Slower response time due to auto-calibration feature. Could take up to ~2 sec for the input reading to stabilize after a large change (i.e. 50Ω to 200kΩ) at the input, or to detect an open circuit. *It is recommended to use software filtering type Moving Average with Filter Constant 100 for this input type.* Frequency Input: accuracy +/- 1% error 1MΩ Impedance, or 10kΩ Pullup/Pulldown 0.5 to 50Hz Range: 0.01Hz resolution, 10Hz to 1kHz Range: 0.1Hz resolution 100 Hz to 10kHz Range: 1Hz resolution *Response time is dependent on input frequency.* PWM Input: 0.01% resolution, accuracy +/- 1% error 1MΩ Impedance, or 10kΩ Pullup/Pulldown *Response time is dependent on input frequency.* Frequency/PWM Input fault detection time to flag an error (i.e. open) 0.5 to 50Hz Range: ~4 seconds (<0.25Hz) 10Hz to 1kHz Range: ~0.105 seconds (<9.5Hz) 100 Hz to 10kHz Range: ~0.01 seconds (<0.95Hz) Digital Input: Active High or Active Low Configurable 10kΩ pullup (to +5V) or pulldown (to GND) resistor which can also be disabled (floating input) Rising edge ON threshold 3.7V +/- 0.1V Falling edge OFF threshold 2.9V +/- 0.1V Input debouncing selectable, will slow response time
Voltage Reference	One provided 5V +/- 0.2% error Can source up to 50mA without derating
Analog GND Reference	One provided

Output Specifications

<table>
<tr><td>Universal Outputs</td><td>Two independent software controlled outputs selectable as: Proportional Current; Hotshot Digital; PWM Duty Cycle; Proportional Voltage; or On/Off Digital types

Half-bridge outputs, current sensing, grounded load.
High side sourcing up to 3A

All output types have configurable minimum and maximum output levels within the range for the type selected.

Current Outputs: 1mA resolution, accuracy +/- 2% error
Software controlled PID current
Range 0 to 3000mA
Fully configurable dither superimposed on top of output current configurable from 50 to 400Hz, 0 to 500mA amplitude
High frequency output drive at 25kHz
Note: Both Outputs use the same dither and drive frequencies

Voltage Outputs: 0.1V resolution, accuracy +/- 3% error
Range 0V to (Vps-0.5V)
Average voltage output based on unit power supply
High frequency output drive at 25kHz
Additional external filtering is required to create a DC voltage

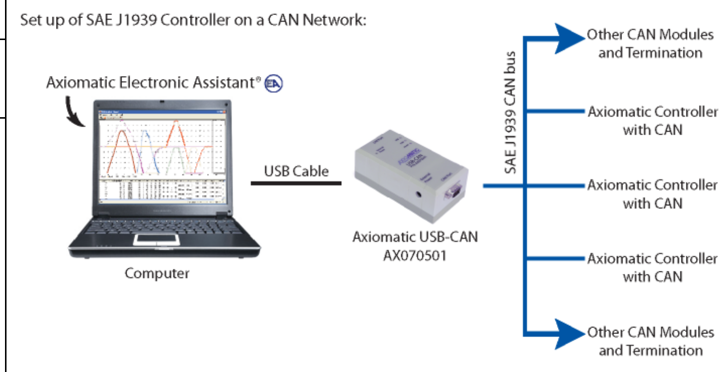
PWM Outputs: 0.1% resolution, accuracy +/- 1% error
Range 0% to 100%
Output Frequency: 1Hz to 25kHz
Configurable frequency ONLY if no current output types are used, otherwise default 25kHz is used
Outputs can be linked in this mode to provide mirrored output signals. (Note: not a true half H-bridge, no deadband provided)

Digital On/Off:
Load at supply voltage must not draw more than 3A.</td></tr>
<tr><td>Response time</td><td>All responses were tested using 0-5V input, full input step change commanding a full output range step change, with no ramps on the output.

Response times accurate to +/- 10ms
<table>
<tr><th>Type</th><th>Rising Edge</th><th>Falling Edge</th></tr>
<tr><td>Current *</td><td>150 ms</td><td>170 ms</td></tr>
<tr><td>Hotshot Digital *</td><td>160 ms</td><td>160 ms</td></tr>
<tr><td>PWM</td><td>30 ms</td><td>30 ms</td></tr>
<tr><td>Voltage</td><td>30 ms</td><td>30 ms</td></tr>
<tr><td>On/Off Digital</td><td><2ms</td><td><2ms</td></tr>
</table>
*Current response tested with 12V HydraForce coil 6507212. Response times will vary depending on the load inductance.</td></tr>
<tr><td>Protection</td><td>Fully protected against short circuit to ground or +Vps
Grounded short circuit protection will engage at 4.5A +/- 0.5A
Unit will fail safe in the case of a short-circuit condition, and is self-recovering when the short is removed.</td></tr>
<tr><td>Power GND Reference</td><td>One Provided</td></tr>
</table>

General Specifications

Power Supply Input - Nominal	12V. 24V or 48VDC nominal 9…60 VDC
Reverse Polarity Protection	Provided
Surge Protection	Provided (up to 65V)
Under-voltage Protection	Provided (software, hardware shutdown at 7.5V)
Over-voltage Protection	Provided (software)
Microcontroller	32-bit, 128 KByte flash program memory, ARM7 processor STM32103CB
Diagnostics	Each input and output channel can be configured to send diagnostic messages to the J1939 CAN network if the I/O goes out of range. Diagnostic data is stored in a non-volatile log. Refer to Section 1.9 for details
Additional Fault Feedback	There are several types of faults that the controller will detect and provide a response: unit power supply undervoltage and overvoltage, microcontroller over temperature and lost communication. They can be sent to the J1939 CAN bus.
CAN User Interface	Compliant to SAE CAN J1939 Standard Configurable via CAN using the Axiomatic Electronic Assistant for *Windows*. It comes with a royalty-free license for use. The Axiomatic Electronic Assistant requires an USB-CAN converter to link the device’s CAN port to a *Windows*-based PC for initial configuration. An Axiomatic USB-CAN Converter AX070501 is available. Order the Axiomatic EA and USB-CAN as a kit (P/N AX070502, or AX070506K), which includes all interconnecting cables.
CAN Response Time	Per the J1939 standard, the maximum recommended transmit rate for any message is 10ms. Response time of feedback on the CAN to changes at the I/O will be a combination of the I/O type’s response time and the configurable software filtering, ramps, delays, etc. that were selected in the application.
Reflashing over CAN	Yes, per J1939 standard using the Axiomatic Electronic Assistant. 29-bit IDs, 250 kbps baud rate
Electrical Connections	Refer to Section 2. Wires should be of the appropriate gauge to meet requirements of applicable electrical codes and suit the specifications of the connector(s).
Packaging and Dimensions	Rugged aluminum housing (AX022400 includes encapsulation material.) Can be mounted directly on the valve block or remotely Suitable for moist, high shock, vibrating and non-hazardous environments Refer to Figure 29 for dimensions of AX022400 (top mount connector) Refer to Figure 30 for dimensions of AX022410 (side mount connector)
Environmental Protection	IP67 rating for product assembly NOTE: Deutsch IPD connectors are rated for submersion (3 ft., 0.9m).
Temperature Rating	Operating: -40 to 85°C (-40 to 185°F) Storage: -50 to 105°C (-58 to 221°F)
Weight – AX022400	1.3 lbs. (0.59 kg) with encapsulation in potting material DOW SYLGARD SILICONE ELASTOMER 170
Weight – AX022410	0.65 lbs. (0.295 kg) without encapsulation



Note:

Technical 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/.

Global Electronic Solutions

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