BIPOLAR INPUT, ANALOG OUTPUT, RELAY OUTPUT, CONTROLLER with NFC

USER MANUAL

P/N: AX031950 (250kbps)
P/N: AX031950-01 (500kbps)
P/N: AX031950-02 (1Mbps)

Contents

1. OVERVIEW OF CONTROLLER

1.1. Description of Bipolar Input to Analog Output and Relay Output Controller with NFC

This User Manual describes the architecture and functionality of the Bipolar Input to Analog Output and Relay Output Controller with Near Field Communication (NFC). All input, outputs and logical function blocks on the unit are inherently independent from one another but can be configured to interact with each other. All parameters are configurable using the Axiomatic Electronic Assistant.

Most of the parameters are configurable using the mobile **E-Write NFC** configuration tool available on the Google Play Store. **E-Write NFC** allows the user to configure the module as well as to assign each of the AX031950 controllers a unique alias to easily distinguish between the controllers within a large system.

The controller's NFC technology provides users the ability to configure the controllers without the need the to be powered on. This feature proves especially useful in cases, for example, in which the unit is installed in a system requiring tuning and does not need to be isolated from the system and powered on externally to perform the tuning; instead the unit can be configured with the system off.

This controller is designed for versatile control of a bipolar input to an analog output and a relay output. The hardware design allows for the controller to have a wide range of input and output types. The control algorithms/function blocks allow the user to configure the controller for a wide range of applications without the need for custom firmware. The various function blocks supported by this controller are outlined in the following sections.

The bipolar input can be configured to read analog signals: *Voltage (Bipolar/Unipolar) and Current* as well as digital signals: *Frequency, PWM, and Digital types*. The inputs are described in more details in section 1.2.

Similarly, the analog output can be configured to different types: *Current, Voltage and PWM.* The analog output is described in more details in section 1.3.

It also supports a normally-open/normally-closed relay output. The relay output is described in more details in section 1.4.

1.2. Bipolar Input Function Block

The controller consists of a single input and can be configured to measure voltage (bipolar/unipolar), current, frequency, pulse width modulation (PWM) and digital signals. The subsections below detail the features/functionalities of the input.

1.2.1. Input Types

Table 1 lists the supported input types by the controller. The **Input Type** parameter provides a dropdown list with the input types described in Table 1. Changing the **Input Type** affects other parameters within the same parameter group such as Minimum/Maximum Range by refreshing them to new input type and thus should be changed first.

Value	Meaning
0	*Input Disabled*
1	*-5V to +5V Voltage*
2	*-10V to +10V Voltage*
3	*0V to 5V Voltage*
4	*0V to 10V Voltage*
5	*0mA to 20mA Current*
6	*4mA to 20mA Current*
7	*Frequency*
8	*PWM Duty Cycle*
9	*Digital (Normal)*
10	*Digital (Inverse)*
11	*Digital (Latched)*

Table 1 – Bipolar Input Type Options

There are a few choices of ranges for both voltage and current types. For the voltage types, the input can be configured to bipolar input (-5V to 5V/-10V to 10V) or unipolar input (0V to 5V/0V to 10V). For the current types, the input can be configured to be in range (0mA to 20mA) or (4mA to 20mA)

Frequency and PWM Duty Cycle **Input Types** are connected to the microcontroller timers.

Digital **Input Types** offers three modes: Normal, Inverse, and Latched. The measurements taken with digital input types are 1 (ON) or 0 (OFF).

1.2.2. Minimum and Maximum Ranges

The **Minimum Range** and **Maximum Range** parameters are used to create the overall useful range of the inputs. For example, if **Minimum Range** is set to 0.5V and **Maximum Range** is set to 4.5V, the overall useful range (0-100%) is between 0.5V to 4.5V. Anything below the **Minimum Range** will saturate at **Minimum Range**. Similarly, anything above the **Maximum Range** will saturate at **Maximum Range.**

1.2.3. Digital Debounce Time

This parameter is used in Digital (Normal), Digital (Inverse) and Digital (Latched) Input Types. It is the time the controller waits until processing and propagating the state of the input when an edge

is triggered. Figure 1 shows how the Debounce time helps filter out noisy push-buttons or switches in order to read a clean signal/state.
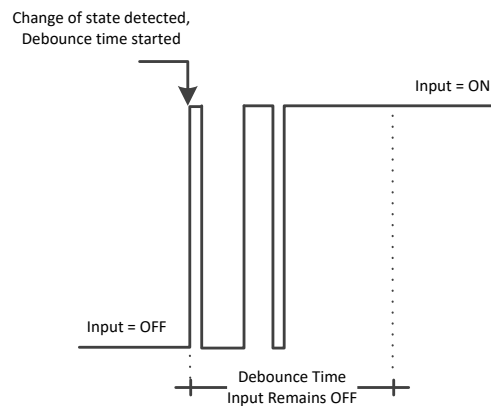
Figure 1: Digital Input Debounce Time

1.2.4. Digital Input Functionality

The Active High/Low parameter allows the user to select how the controller responds to the behavior of the digital input. Table 2 shows the different Active High/Low options with the default being highlighted.

Value	Meaning
0	***Active High***
1	*Active Low*

Table 2 – Active High/Low Options

The input is configured to *Active High* by default which means the input drive will give an ON response when the input signal is high and OFF when the signal is low. When the input is configured to *Active Low,* the responses will be the opposite.

1.2.5. Pullup / Pulldown Resistor Options

With **Input Types:** Frequency, PWM, Digital, the user has the option of three (3) different pull up/pull down options as listed in Table 3.

Value	Meaning
0	*Pull Up/Pull Down Not Used*
1	*Pullup*
2	*Pulldown*

Table 3 – Pullup/Pulldown Resistor Options

1.2.6. Software Filter

All input types except for Digital (Normal), Digital (Inverse), Digital (Latched) can be filtered using **Software Filter Type** and **Software Filter Constant** parameters. There are three (3) filter types available as listed in Table 4.

Value	Meaning
0	*Not Used*
1	*Moving Average*
2	*Repeating Average*

Table 4 – Input Filtering Types

The first filter option *No Filter*, provides no filtering to the measured data. Thus, the measured data will be directly used to the any function block which uses this data.

The second option, *Moving Average*, applies the 'Equation 1' below to measured input data, where $Value_N$ represents the current input measured data, while $Value_{N-1}$ represents the previous filtered data. The Filter Constant is the **Input Filter Constant** parameter.

Equation 1 - Moving Average Filter Function:

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

The third option, Repeating Average, applies the 'Equation 2' below to measured input data, where N is the value of **Input Filter Constant** parameter. The filtered input, Value, is the average of all input measurements taken in N (**Input Filter Constant)** number of reads. When the average is taken, the filtered input will remain until the next average is ready.

Equation 2 - Repeating Average Transfer Function:

$$Value = \frac{\sum_0^N Input_N}{N}$$

1.3. Analog Output Drive Function Block

The controller has an analog output can be configured. The **Analog Output Type** parameter determines what kind of signal the output produces. Changing this parameter causes other parameters in the group to update to match selected type. For this reason, the first parameter that should be changed prior to configuring other parameters is the Analog Output Type parameter. The supported output types by the controller are listed in Table 5 below:

Value	Meaning
0	Output Not Used
1	Current Output
2	PWM Duty Cycle
3	Voltage Output

Table 5– Analog Output Type Options

Regardless of what type of analog output type is selected, the output will always respond in a linear fashion to changes in the control source per the calculation in Figure 2.

$$y = mx + a$$

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

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

Figure 2 – Linear Slope Calculations

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. Or it allows output to follow control signal inversely.

In *PWM Duty Cycle* **Output Type**, the controller outputs a signal (0-VPS amplitude) on a fixed output frequency set by **PWM Output Frequency** with varying PWM Duty Cycle based on commanded input. Since both outputs are connected to independent timers, the **PWM Output Frequency** parameter can be changed at any time for each output without affecting the other.

For the analog output, the signal minimum and maximum values are configured with **Output At Minimum Command** and **Output At Maximum Command** parameters. Value ranges for both parameters are limited by selected **Output Type**.

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 (Min to Max)** and **Ramp Down (Max to Min)** parameters 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.

The **Control Source** parameter together with **Control Number** parameter determine which signal is used to drive the output. For example, setting **Control Source** to *Universal Input Measured* and **Control Number** to (*1)* will connect signal measured from Universal Input1 to the output in question. The input signal is scaled per input type range between 0 and 1 to form control signal. Outputs respond in a linear fashion to changes in control signal. If a non-digital signal is selected to drive digital output the command state will be 0 (OFF) at or below the "**Output At Minimum Command**", 1 (ON) at or above "**Output At Maximum Command**" and will not change in between those points. The options for Control Sources and available Control Number are listed in Table 5.

Value	Meaning	Number
0	*Control Not Used*	*1*
1	*Received CAN Message*	*8*
2	*Universal Input Measured*	*1*
3	*Lookup Table*	*9*
4	*Programmable Logic Block*	*3*
5	*Math Function Block*	*5*
6	*Control Constant Data*	
7	*Diagnostic Trouble Code*	
8	*Output Current Feedback*	
9	*Power Supply Measured*	*1*
10	*Processor Temperature Measured*	*1*
11	*Receive Message Timeout*	

Table 6: Internal Sources

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"** in Table 7. If the response is selected as a disable signal (2 or 3), when the input is ON, the output will be shutoff.

Value	Meaning
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 7 – 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."** The options for Override Response are listed in Table 8.

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

Table 8 – Override Response Options

The options for both Enable Sources and Override Sources are same as Control Sources listed in Table 6.

If a fault is detected in any of the active input the output will shut down until the input recovers. Besides the input faults shutting down the output, if an under-voltage/over-voltage measurement occurs on VPS, the output will also shut down.

The output is inherently protected against a short to GND or VPS by hardware. In 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 detects output hardware shutdown and commands off the output in question. It will continue to drive non-shorted outputs normally and periodically 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 any of the outputs. The processor will continue to attempt to drive the open load.

1.4. Relay Output Drive Function Block

There is a relay output available in this controller as well. The following sub-sections will explain in more detail the functionalities and available setpoints/parameters.

1.4.1. Relay Output Functionality

The relay output has 2 states: *Normally Open* and *Normally Closed*. It has 3 pins associated with it: Normally Closed (NC), Normally Open (NO), and Common (C). The **Digital Response** parameter allows for flexibility in the response of the output. Table 9 shows the options available for this parameter.

Value	Meaning
0	*Normal ON/OFF*
1	*Inverse Logic*
2	*Latched Logic*
3	*Blinking Logic*

Table 9: Relay Output Responses

By default, *Normal Logic* response is used for the relay outputs. In *Normal Logic* response, the Common pin is connected to the Normally Closed pin if the source of the respective relay output is triggered ON, the Common pin is connected to the Normally Open pin.

In the case of *Inverse Logic* response, the Common pin is connected to the Normally Open pin when the source of the respective relay output is triggered ON. When the source of the respective relay output is triggered OFF, the Common pin is connected to the Normally Closed pin.

In the case of *Latched Logic* response, the Common pin is toggled between Normally Closed and Normally Open pins every time the source of the respective relay output goes from OFF to ON.

The *Blinking Logic* lets the relay output toggle between Normally closed and Normally Open pins for a configured frequency. The time for switching from one state to the other state results the **Blink Rate** which is in milliseconds and by default 500ms.

1.4.2. Relay Output Control / Enable Sources / Override Source

The relay output can be configured to be commanded and/or enabled by the control sources listed in Table 6. This table also displays the number associated to the control sources which can be selected. The default control source is highlighted in Table 6 while the default Enable Source and Override Source is configured to *Source Not Used.*

The selected control source in the **Control Source** parameter is the main commanding source of the relay output based on **Relay Output Response** parameter.

1.4.3. Relay Output Enable

The **Enable Source** will determine whether or not the relay output will be commanded by the **Control Source**. There are six different **Enable Responses** in which the enable signal can be used. These responses are the same as for analog output listed in Table 7.

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

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

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

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

1.4.4.   Relay Output Override

The **Override Source** will determine whether or not the relay output will be commanded by the **Control Source**. This Source has a higher priority than the Enable Source.

There are two different **Override Responses** in which the Override signal can be used. These responses are listed in Table 8, where the default value is highlighted.

When the **Override Response** is configured to *Override When ON*, the relay output will be commanded according to the signal of the **Control Source/Number** by the **Override State**. If the Override Response is set to Override When OFF, the relay output will be commanded according to the signal of the Control Source/Number by the Override State. Table 10 shows the two possible states for the **Override Response**.

In case of **Override State OFF**, the relay output switches to Normally Open. If **Override State ON** is configured, the relay output changes to Normally closed.

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

Table 10 – Override State Options

1.5. Lookup Table Function Block

The Lookup Table is used to give an output response of up to 10 slopes per Lookup Table. There are two types of Lookup Table response based on Lookup Table **X-Axis Type**: *Data Response* and *Time Response* Sections 1.5.1 through 1.5.6 will describe these two types of **Responses** in more detail.

When the Lookup Table **Response** is *Data Response,* the **X-Axis Point x** values are always in percentage which reflects the percentage of the **Control Source** used in the Lookup Table. Changing the **Control Source** will not change the values of the **X-Axis Point x** or **X-Axis Point y.**

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 CAN Receive message, 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.

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 X10 is changed first, then lower indexes in descending order.

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

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 X4 and higher are ignored, the formula becomes *Xmin <= X0 <= X1 <= X2 <= X3 <= Xmax* 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.

In all cases, the controller looks at the **entire range** of the data in the Y-Axis parameters, and selects the lowest value as the *Ymin* and the highest value as the *Ymax*. 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.)

1.5.3. Default Configuration, Data Response

By default, the Lookup Table is disabled (Lookup Table **Control Source** is set to *Control Not Used).* The Lookup Table can be used to create the desired response profiles. When the Universal Input is used as the **Control Source**, the output of the Lookup Table will be what the user enters in **Y-Values** parameters.

Recall, any controlled function block which uses the Lookup Table as an input source 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 correspond to the minimum and maximum values of the table's Y-Axis.**

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.

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 3. To get the linearization, each "**Point N – Response**", where N = 1 to 10, is setup for a *'Ramp To'* output response.
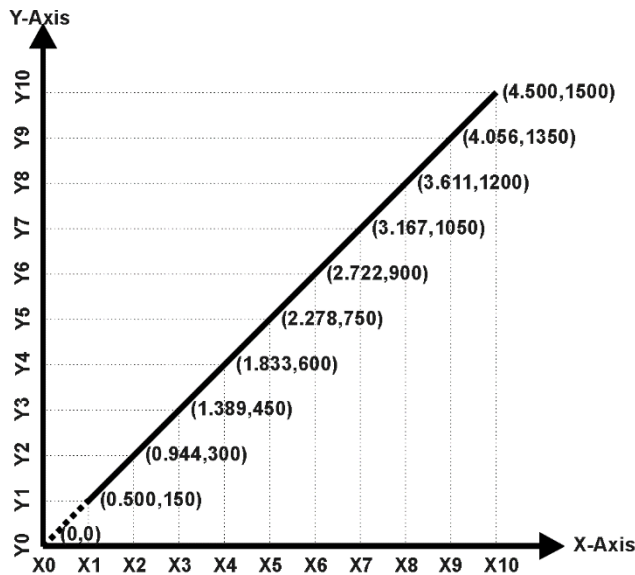

Figure 3 – Lookup Table with "Ramp To" Data Response

Alternatively, the user could select a *'Jump To'* response for "**Point N – Response**", where N = 1 to 10. In this case, the output of the Lookup Table will not change in between **X-Axis Points** rather it will only change when it is >**X-Axis Point n** and < **X-Axis Point (n+1)**

A combination of *Ramp To*, *Jump To* and *Ignore* responses can be used to create an application specific output profile.

1.5.5. X-Axis, Time Response

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

In this case, the **Control Source** is treated as a digital input. When the control input is ON, the output will be changed over a period of time based on the profile in the Lookup Table.

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.

In a time response, the interval time between each point on the X-axis can be set anywhere from 1ms to 1day [86400 s]

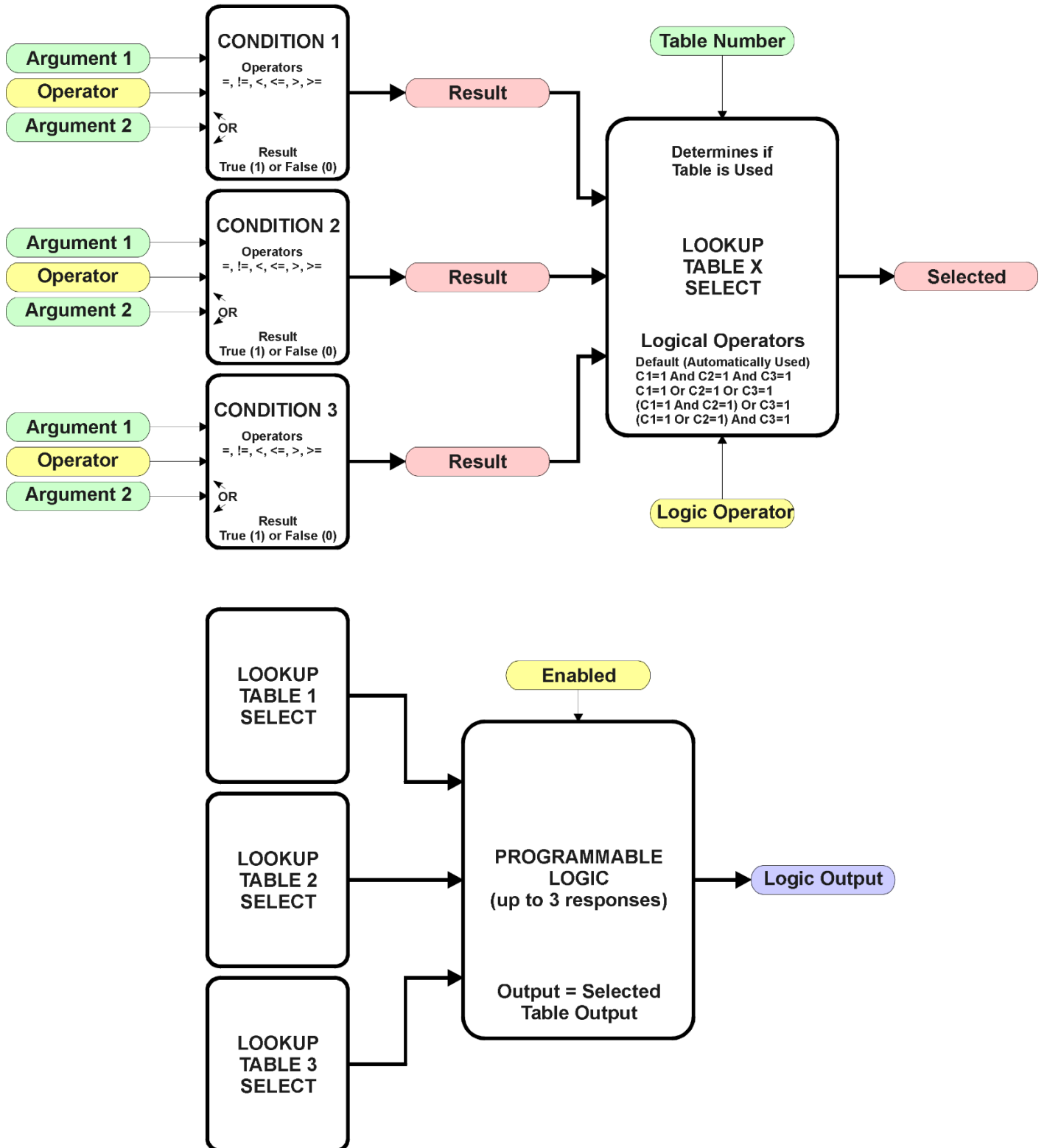## 1.6. Programmable Logic Function Block

Figure 4 – 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 9) 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 Relay Output. 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 17. 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.**
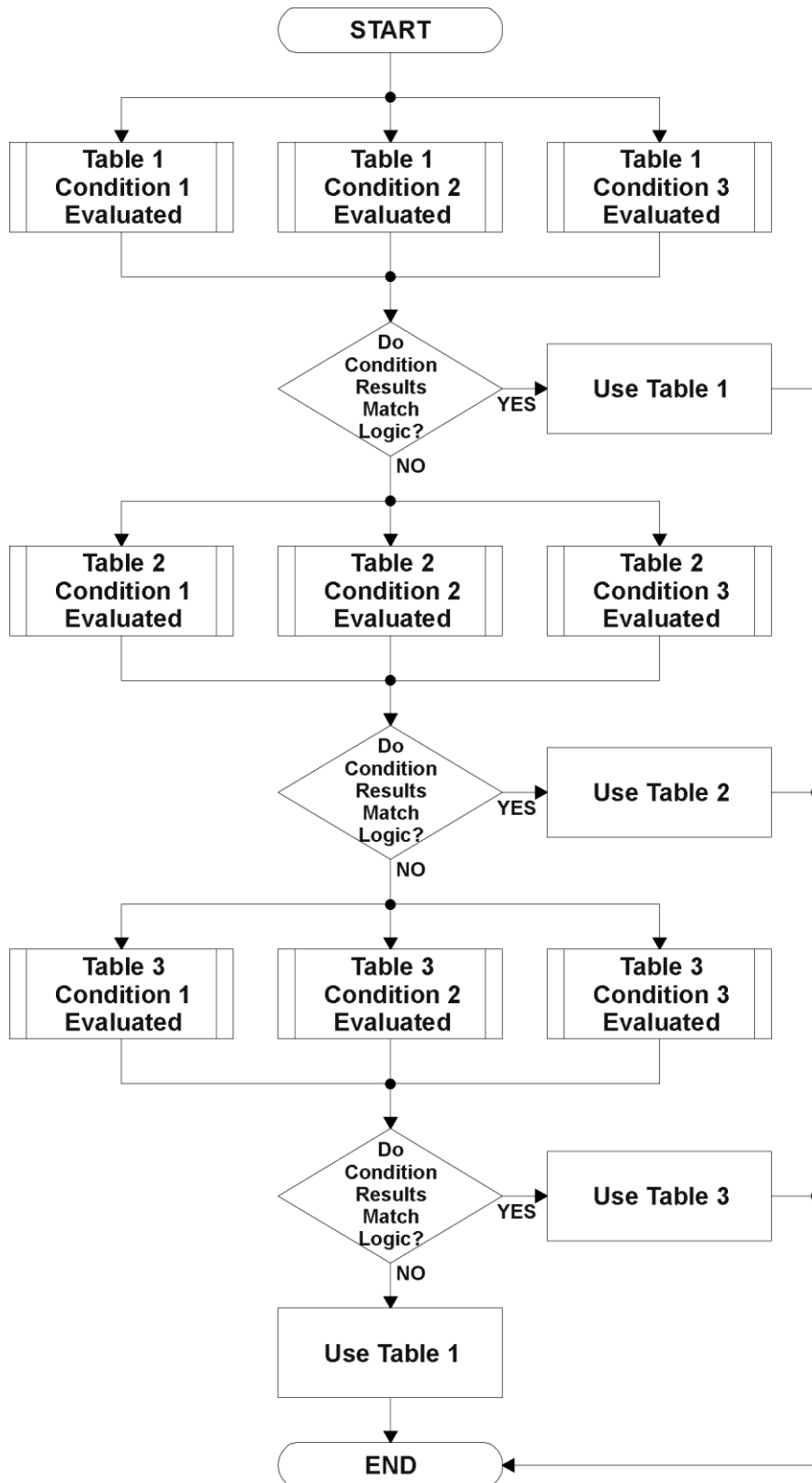
Figure 5 – Programmable Logic Flowchart

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 1.11. 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 11.

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

Table 11 – Condition Operator Options

For example, a condition for a transmission control shift selection could be that the Engine RPM received on CAN message 3 be less than a certain value to select a Soft Fill profile. In this case, **"…Argument 1 Source"** would be set to *'Received CAN J1939 Message 3'*, **"…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. There could be four possible results evaluation, as described in Table 12.

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 12 – 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 13.

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 13 – 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 14 – 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 15.

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	7	8	9

Table 15 – 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.
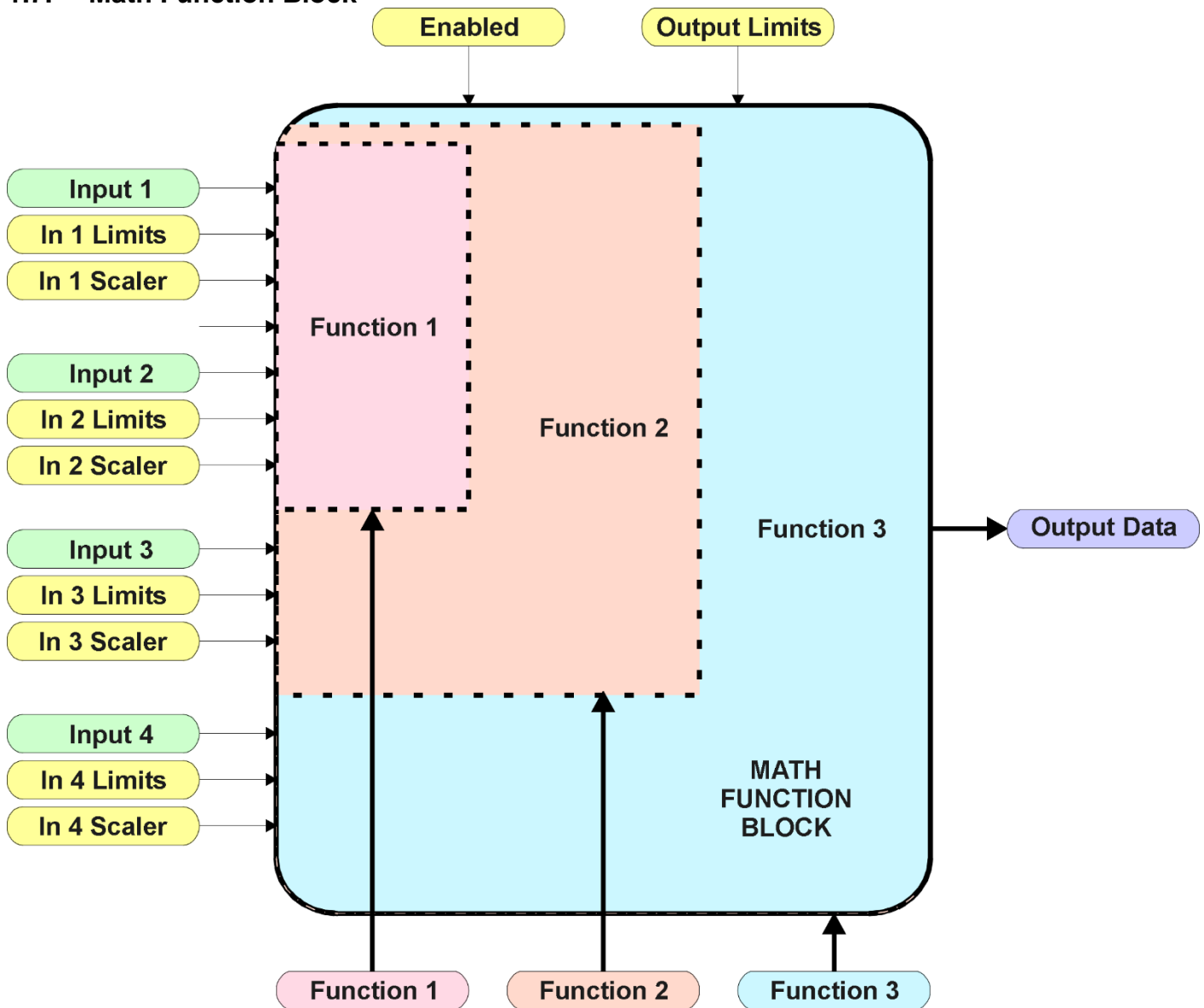
1.7. Math Function Block

Figure 6 – Math Function Block

There are five mathematic function blocks that allow the user to define basic algorithms. A math function block can take up to four input signals. 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 16.

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 16 – 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.
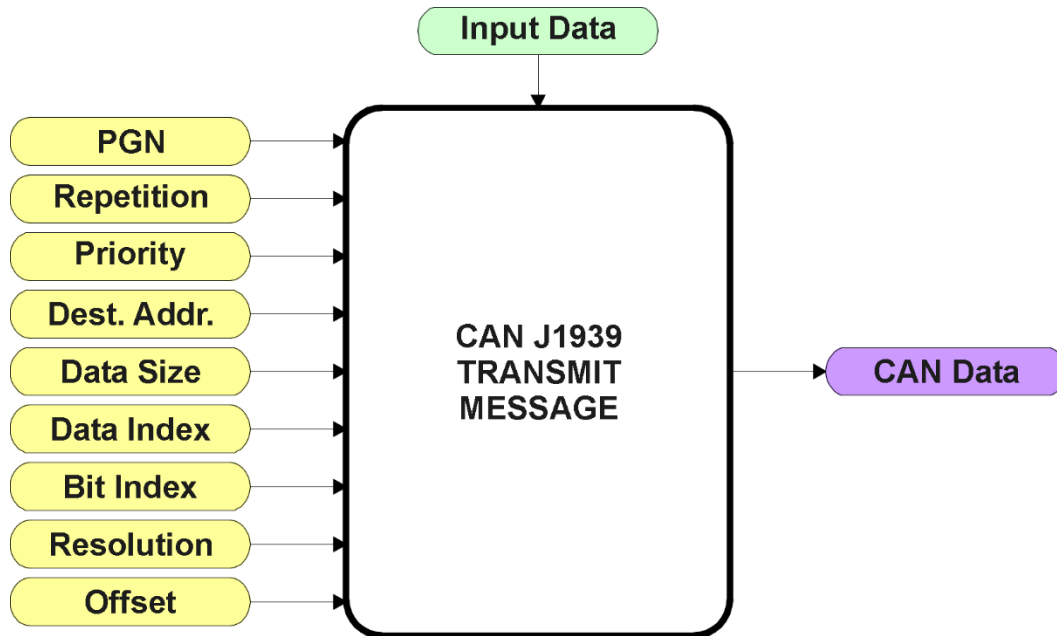
1.8. CAN Transmit Function Block

Figure 7 – 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 controller are all for proprietary SPNs, and are defined in detail in Section 3.3.

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

This 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, the first message is pre-configured. The details are outlined in Section 3.3, and the default list is shown in Table 17 below.

Block #	Default Transmit Data	(PGN)
1	Universal Input Measured	(0xFF00)
2	Control Source Not Used	(0xFFFF)
3	Control Source Not Used	(0xFFFF)
4	Control Source Not Used	(0xFFFF)
5	Control Source Not Used	(0xFFFF)
6	Control Source Not Used	(0xFFFF)
7	Control Source Not Used	(0xFFFF)
8	Control Source Not Used	(0xFFFF)

Table 17 – Default CAN Transmit Messages
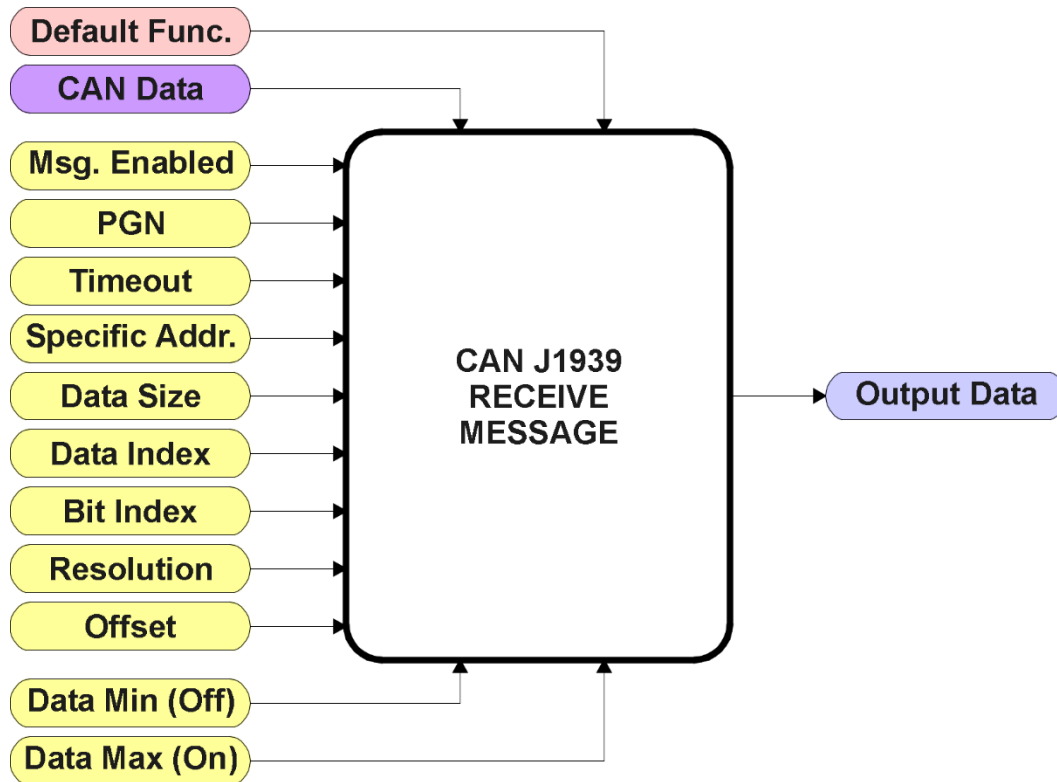
1.9. CAN Receive Function Block

Figure 8 – 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 only the first two received messages are enabled.

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 unit on Proprietary B PGNs. However, should a PDU1 message be selected, the unit 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 unit 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.

This Controller supports up to 8 unique CAN Receive Messages. By default, the first two messages are 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 18 below.

| Block #	Default Receive Data
1	Analog Output Command Input Data
2	Relay Output Command Input Data
3	Message Not Used
4	Message Not Used
5	Message Not Used
6	Message Not Used
7	Message Not Used
8	Message Not Used

Table 18 – Default CAN Receive Messages

2. Installation Instructions

2.1. Dimensions and Pinout

The control has 12 screw terminals with pin out shown b. Refer to dimensional drawing below for terminal orientation.

Screw Terminal #	Description
1	Relay Output COM 1
2	Relay Output NC 1
3	Relay Output NO 1
4	+5V Reference
5	Analog Output
6	Analog Output Return
7	Analog Input
8	CAN_H
9	CAN_L
10	Power +
11	Power -
12	Analog Input GND
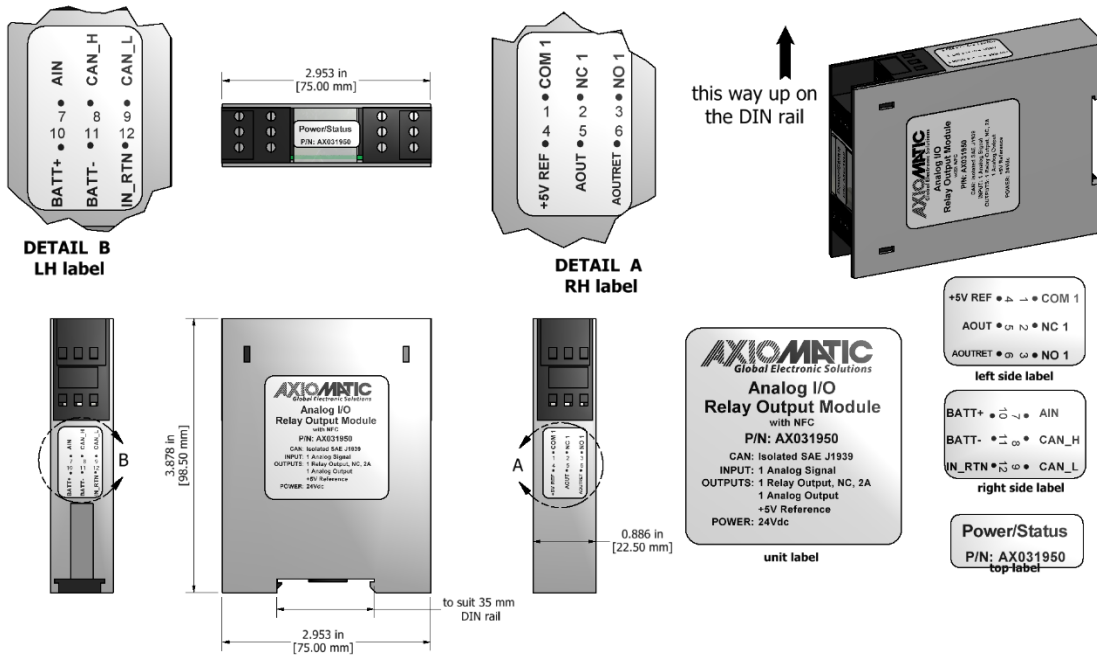
3. OVERVIEW OF J1939 FEATURES

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

3.2.1. J1939 Name

The unit 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	126, Axiomatic I/O Controller
Function Instance	11, Axiomatic AX031950, Bipolar Input, Relay Output and Analog Output with NFC
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.

3.2.2. 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 unit will continue select the next highest address until it find one that it can claim. See J1939/81 for more details about address claiming.

3.2.3. 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 this unit, Byte 1 is set to 1, 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:

Parameter	Value	Description
Electronic Assistant		
File View Options Help		
J1939 CAN Network		
ECU AX031950, Bipolar Input, Relay O		
i General ECU Information		
Setpoint File		
B Bootloader Information		
▶ ECU Part Number	AX031950	
▶ ECU Serial Number	0000119001	
┌ ECU J1939 NAME		PGN 60928. 64-bit ECU Identifier sent in Address Claimed Messages
┼ Arbitrary Address Capable	0X01	Yes
┼ Industry Group	0X00	Global
┼ Vehicle System Instance	0X00	
┼ Vehicle System	0X00	Non-specific system
┼ Reserved	0X00	
┼ Function	0X7E	Axiomatic IO Controller
┼ Function Instance	0X0B	
┼ ECU Instance	0X00	#1 - First Instance
┼ Manufacturer Code	0X0A2	Axiomatic Technologies
└ Identity Number	0X10ACB9	Unique ECU network ID number
▶ ECU Address	0X80	Reserved for future assignment by SAE, but available for use by self configurable ECUs
─ ECU ID	N/A	PGN 64965 -ECUID
┌ Software ID		PGN 65242 -SOFT
┼ Field #1	AX031950	
┼ Field #2	V1.00	
┼ Field #3	FEB 01, 2019	
┼ Field #4	Axiomatic Technologies Corp.	
└ Field #5	Bipolar Input, Relay Output and Analog Output with NFC	
Ready		250 kbit/s

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 unit 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 "Universal Input Measured Message" has the following default configuration.

```
PGN 65280                    Universal Input Measured

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
1-2	2 byte	Universal Input Measured	520448
3-5	3 byte	Not Used	N/A

Universal Input Measured

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 (0x7F100, proprietary SPNs)
Parameter Group Number:	65280

3.4. CAN Receive Message Defaults

This section outlines the **default** settings of this unit 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 unit react to a DTC sent by another ECU on the network on a DM1, use the DTC React Function block instead.

By default, all but the first CAN Receive Messages are disabled, as they are not part of the factory set logic shown in Figure 24. However, should any 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	Analog Output Command Input Data	N/A
2	1 byte	Relay Output Command Input Data	N/A
3	1 byte	Not Used	N/A
4	1 byte	Not Used	N/A
5	1 byte	Not Used	N/A
6	1 byte	Not Used	N/A
7	1 byte	Not Used	N/A
8	1 byte	Not Used	N/A

Analog Output Command Input Data

Default value used when a *'Received CAN J1939 Message'* is used as the control source for the Analog 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

Relay Output Command Input Data

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

4. ECU SETPOINTS ACCESSED WITH THE AXIOMATIC ELECTRONIC ASSISTANT

Many setpoints have been reference 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 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.
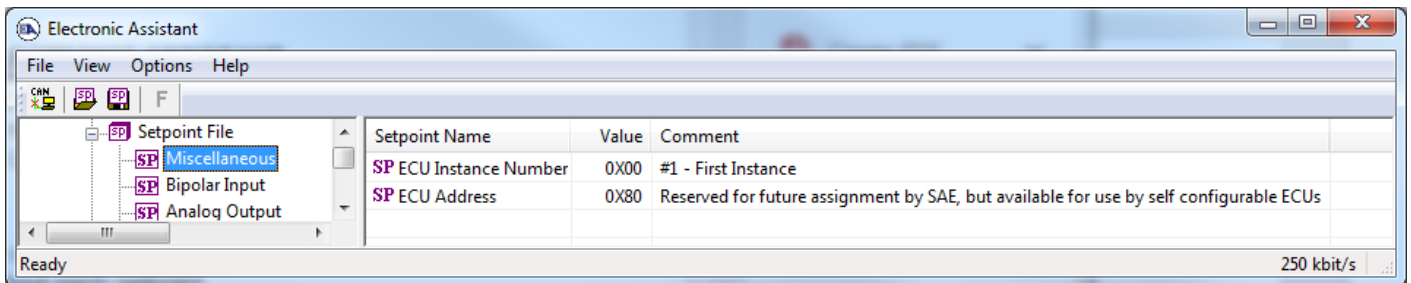
Screen Capture of Default Miscellaneous Setpoints

Name	Range	Default	Notes
ECU Instance Number	Drop List	0, #1 – First Instance	Per J1939-81
ECU Address	0 to 253	128 (0x80)	Preferred address for a self-configurable ECU

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. Bipolar Input Setpoints

The Bipolar Input function block is defined in Section 1.2. Please refer to that section for detailed information on how these setpoints are used.
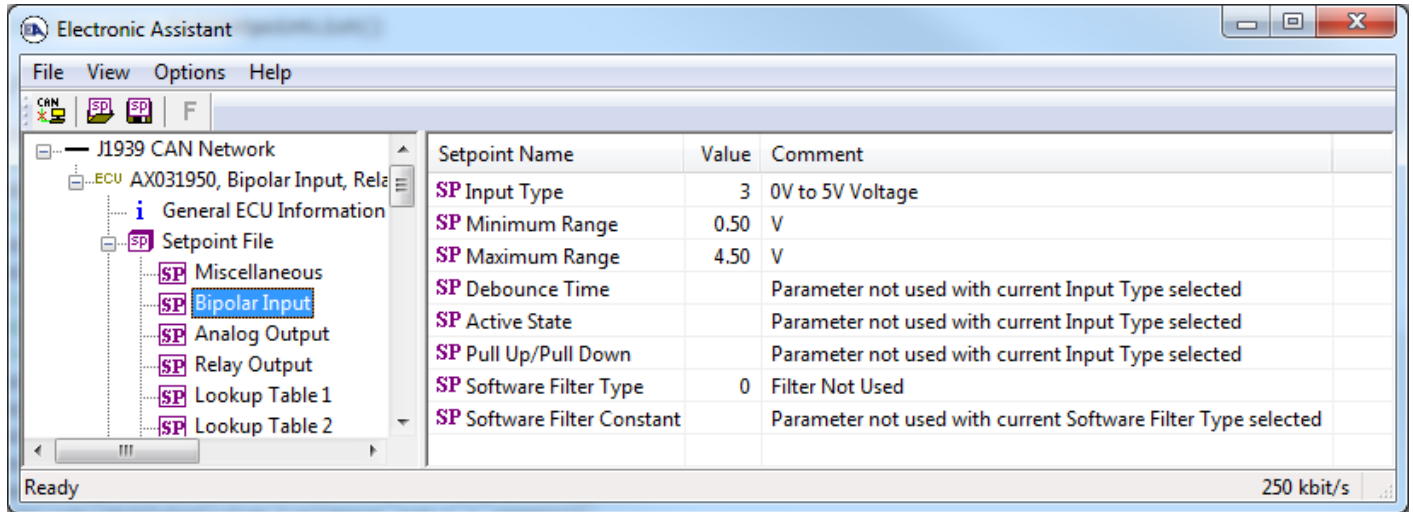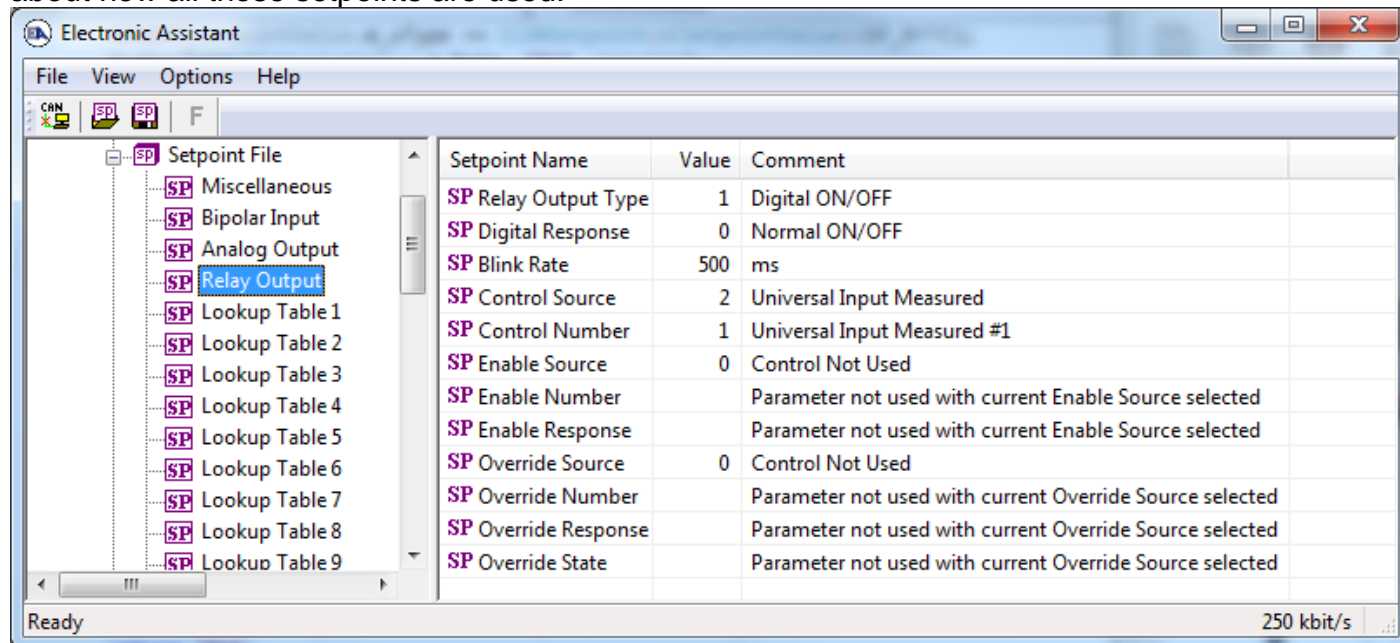

Screen Capture of Default Bipolar Input Setpoints

Name	Range	Default	Notes
Input Type	Drop List	3 – 0V to 5V Voltage	Refer to Section 1.2.1
Minimum Range	Depends on Sensor Type	0.5V	Refer to Section 1.2.2
Maximum Range	Depends on Sensor Type	4.5V	Refer to Section 1.2.2
Debounce Time	[0…5000]	10	Units in [milliseconds]
Active State	Drop List	Active High	Refer to Section 1.2.4
Pull Up/Pull Down	Drop List	Pull Up/Pull Down Not Used	Refer to Section 1.2.5
Software Filter Type	Drop List	Filter Not Used	Refer to Section 1.2.6
Software Filter Constant	[0…60000]	10	Refer to Section 1.2.6

4.3. Analog Output Setpoints

The Analog Output function block is defined in Section 1.3. Please refer there for detailed information about how all these setpoints are used.
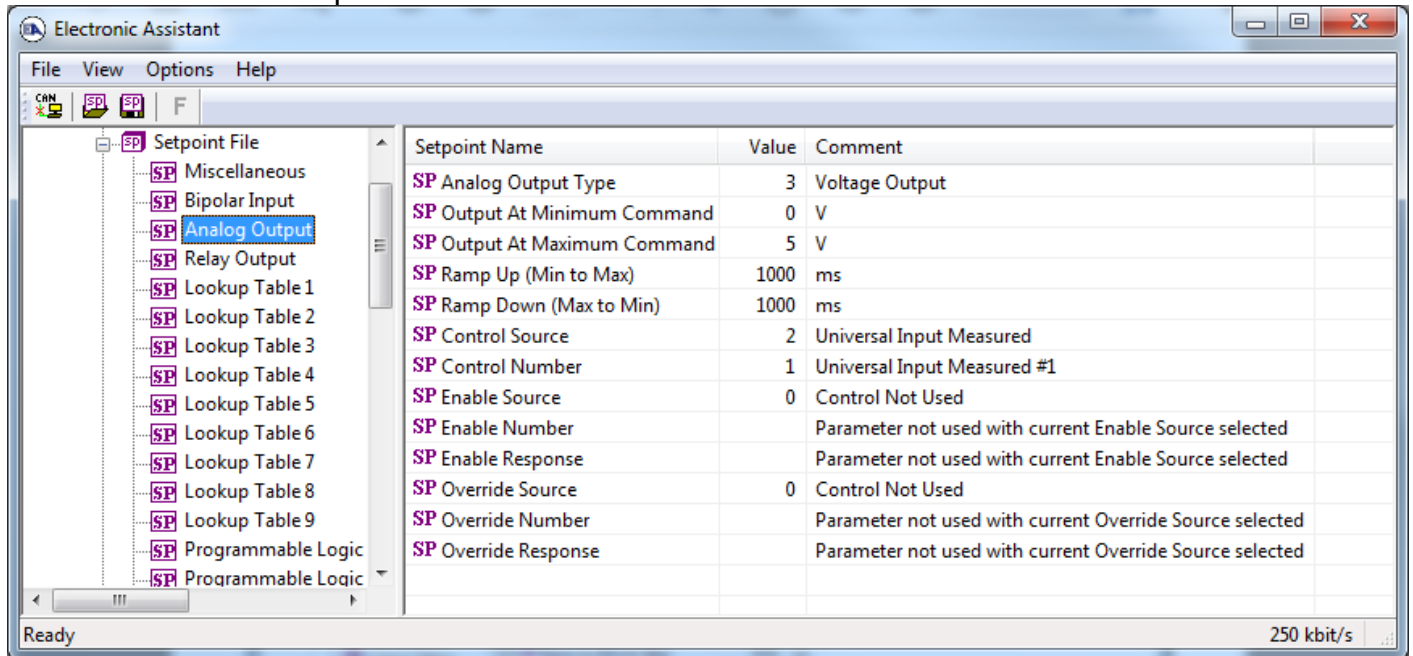
Screen Capture of Default Analog Output Setpoints

Name	Range	Default	Notes
Analog Output Type	Drop List	3, Voltage Output	See Table 5
Output at Minimum Command	Depends on Output Type	0V	Refer to Section 1.3
Output at Maximum Command	Depends on Output Type	5V	
Ramp Up (Min to Max)	0 to 10,000	1000	Unit in milliseconds
Ramp Down (Max to Min)	0 to 10,000	1000	Unit in milliseconds
Control Source	Drop List	2, Universal Input Measured	See Table 6
Control Number	Per Source	1	See Table 6
Enable Source	Drop List	0, Control Source Not Used	See Table 6
Enable Number	Per Source	N/A	See Table 6
Enable Response	Drop List	N/A	See Table 7
Override Source	Drop List	0, Control Source Not Used	See Table 6
Override Number	Per Source	N/A	See Table 6
Override Response	Drop List	N/A	See Table 8

4.4.   Relay Output Setpoints

The Analog Output 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 Analog Output Setpoints

Name	Range	Default	Notes
Relay Output Type	Drop List	1, Digital ON/OFF	Output Not Used, Digital ON/OFF
Digital Response	Drop List	0, Normal ON/OFF	See Table 9
Blink Rate	100 to 5000	500	Only usede with "Blinking" response
Control Source	Drop List	2, Universal Input Measured	See Table 6
Control Number	Per Source	1	See Table 6
Enable Source	Drop List	0, Control Source Not Used	See Table 6
Enable Number	Per Source	N/A	See Table 6
Enable Response	Drop List	N/A	See Table 7
Override Source	Drop List	0, Control Source Not Used	See Table 6
Override Number	Per Source	N/A	See Table 6
Override Response	Drop List	N/A	See Table 8
Override State	Drop List	Override State OFF	See Table 10

4.5. Lookup Table Setpoints

The Lookup Table 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 Table6, 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.

4.6. 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.
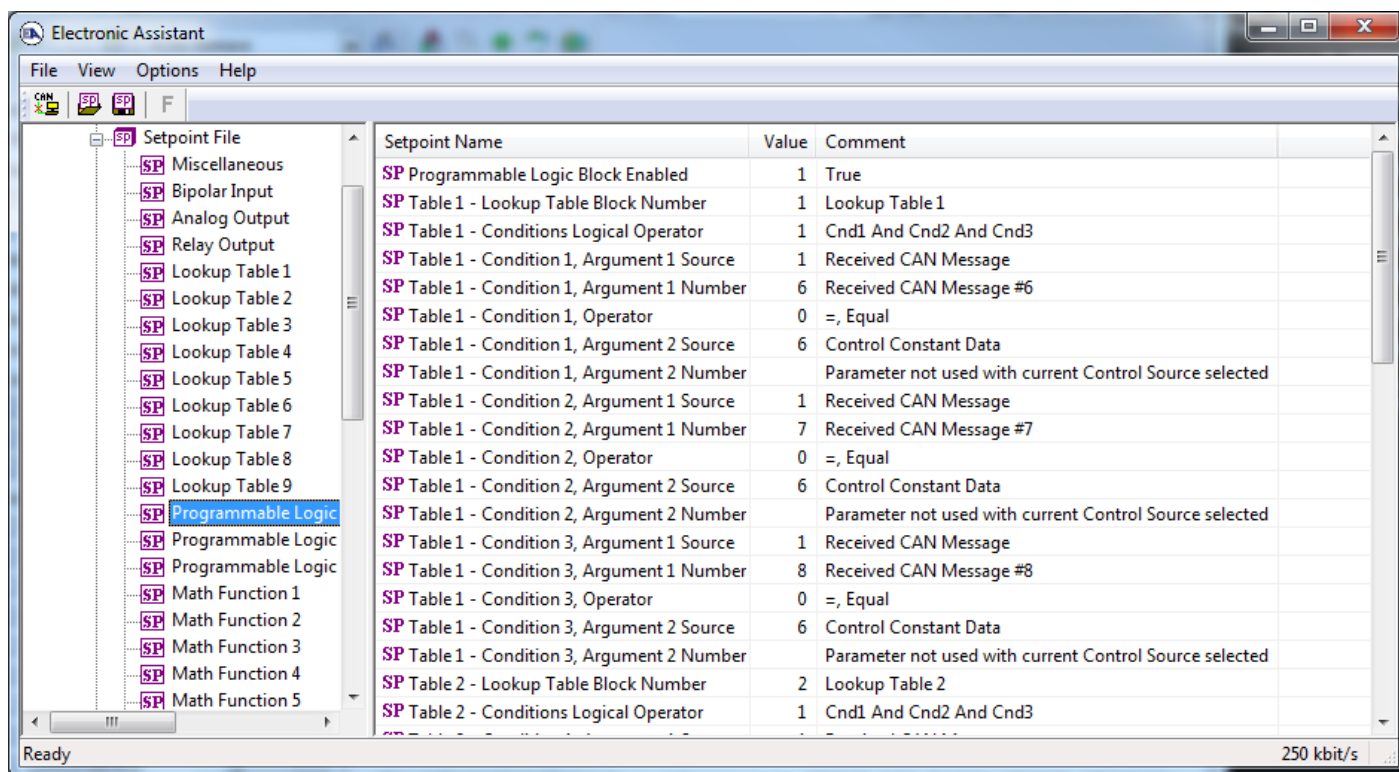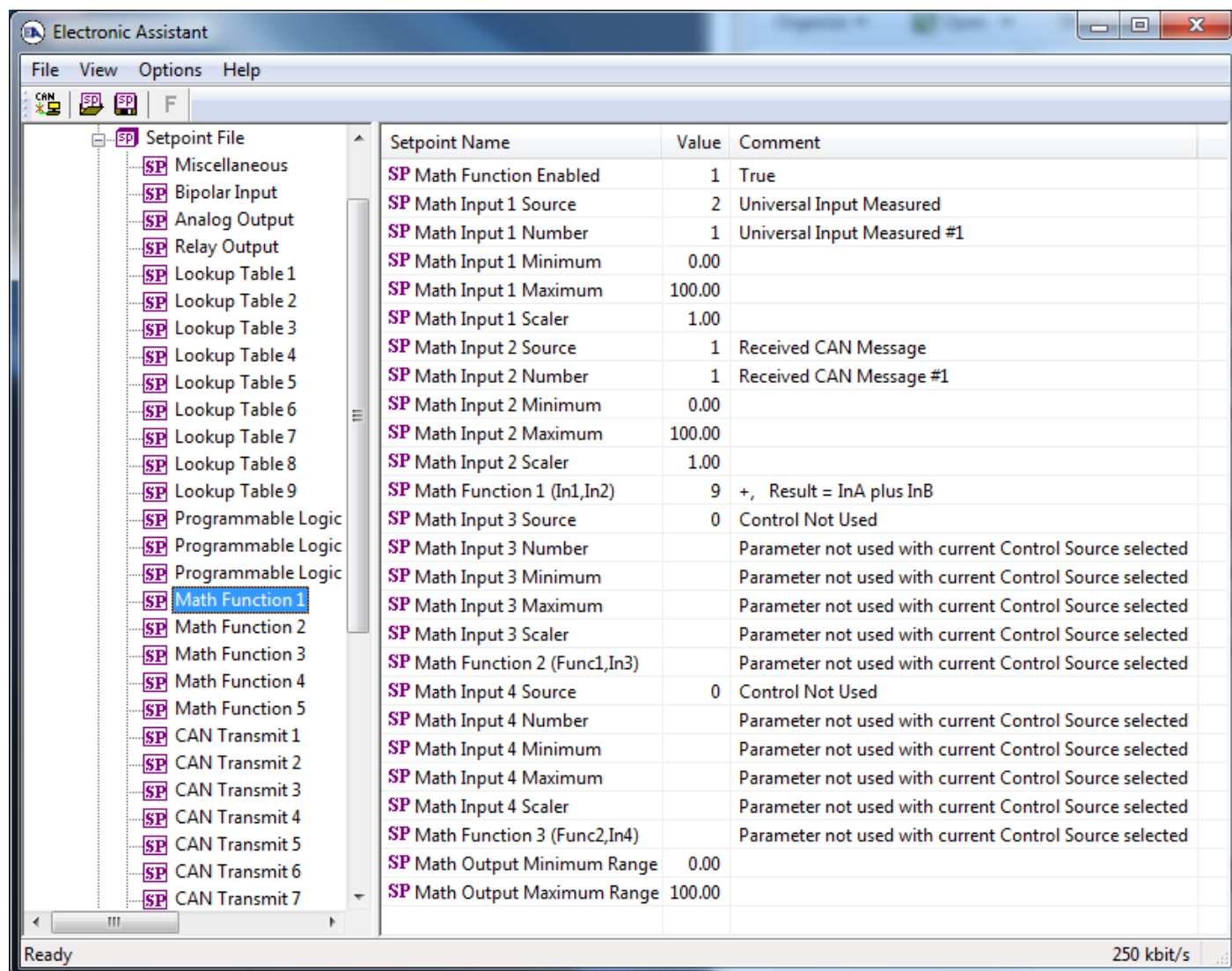

Screen Capture of Default Programmable Logic 1 Setpoints

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.

4.7. 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 Not Used	See Table 6
Math Input X Number	Per Source	0	See Table 6
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	

4.8. CAN Transmit Setpoints

The CAN Transmit function block is defined in Section 1.8, with addition information in Section 3.3. Please refer there for detailed information about how all these setpoints are used.
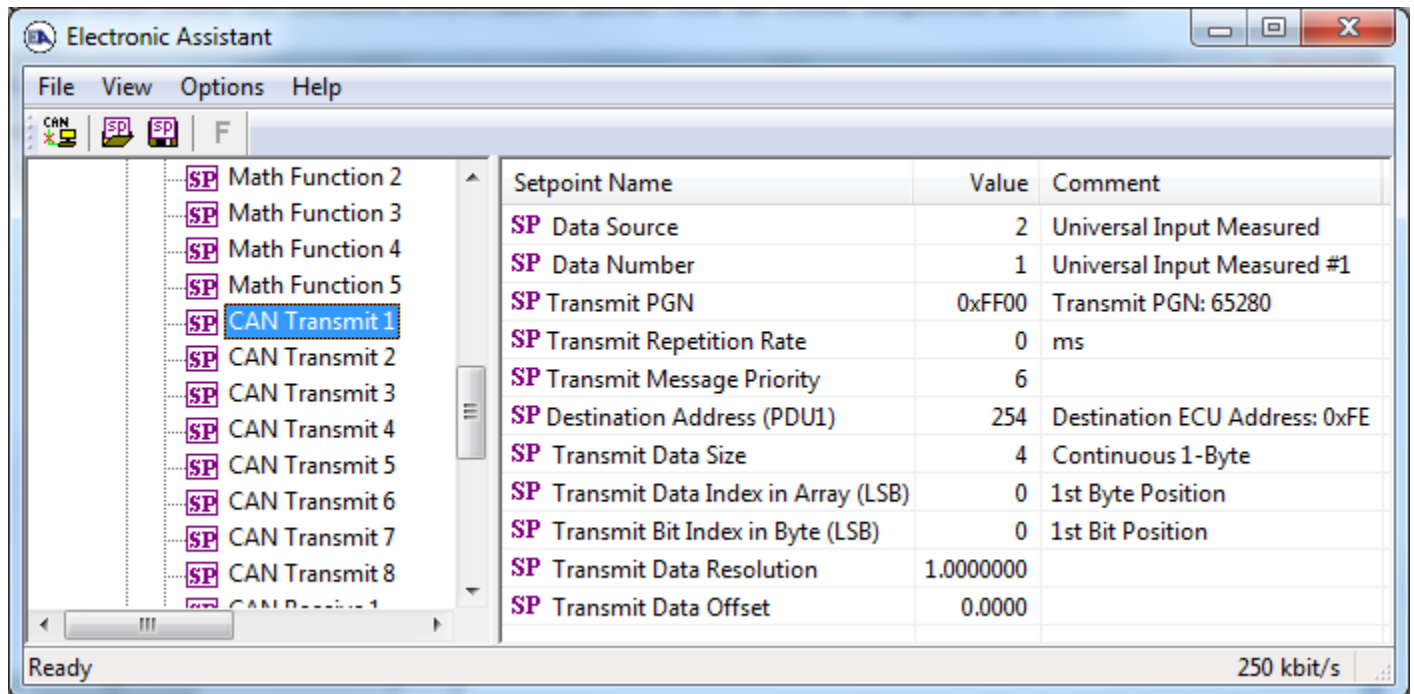
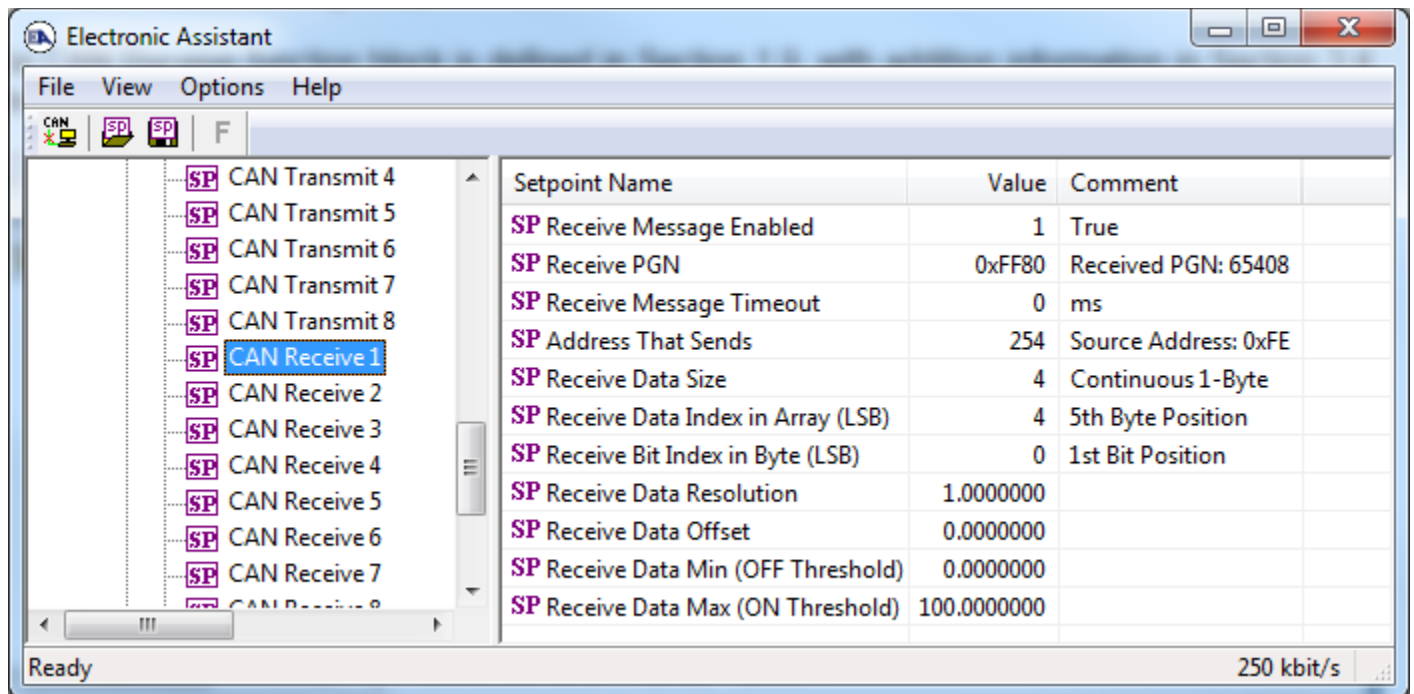*Screen Capture of Default CAN Transmit 1 Setpoints*

Name	Range	Default	Notes
Data Source	Drop List	Different for each	See Table 17 for defaults
Data Number	Per Source	Different for each	See Table 17 for defaults
Transmit PGN	0 to 65535	65280(0xFF80)	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.

4.9.   CAN Receive Setpoints

The CAN Receive function block is defined in Section 1.9, 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: It the "Received Message Enabled" is setpoint is False all other setpoints read "No Control Source, Function Block Disabled"

| Name	Range	Default	Notes
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	0 ms	Expects all data at 100ms
Specific Address that sends PGN	0 to 255	254	(0xFE, Null Addr)
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

5. REFLASHING OVER CAN WITH THE AXIOMATIC EA BOOTLOADER

The AX031950 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.

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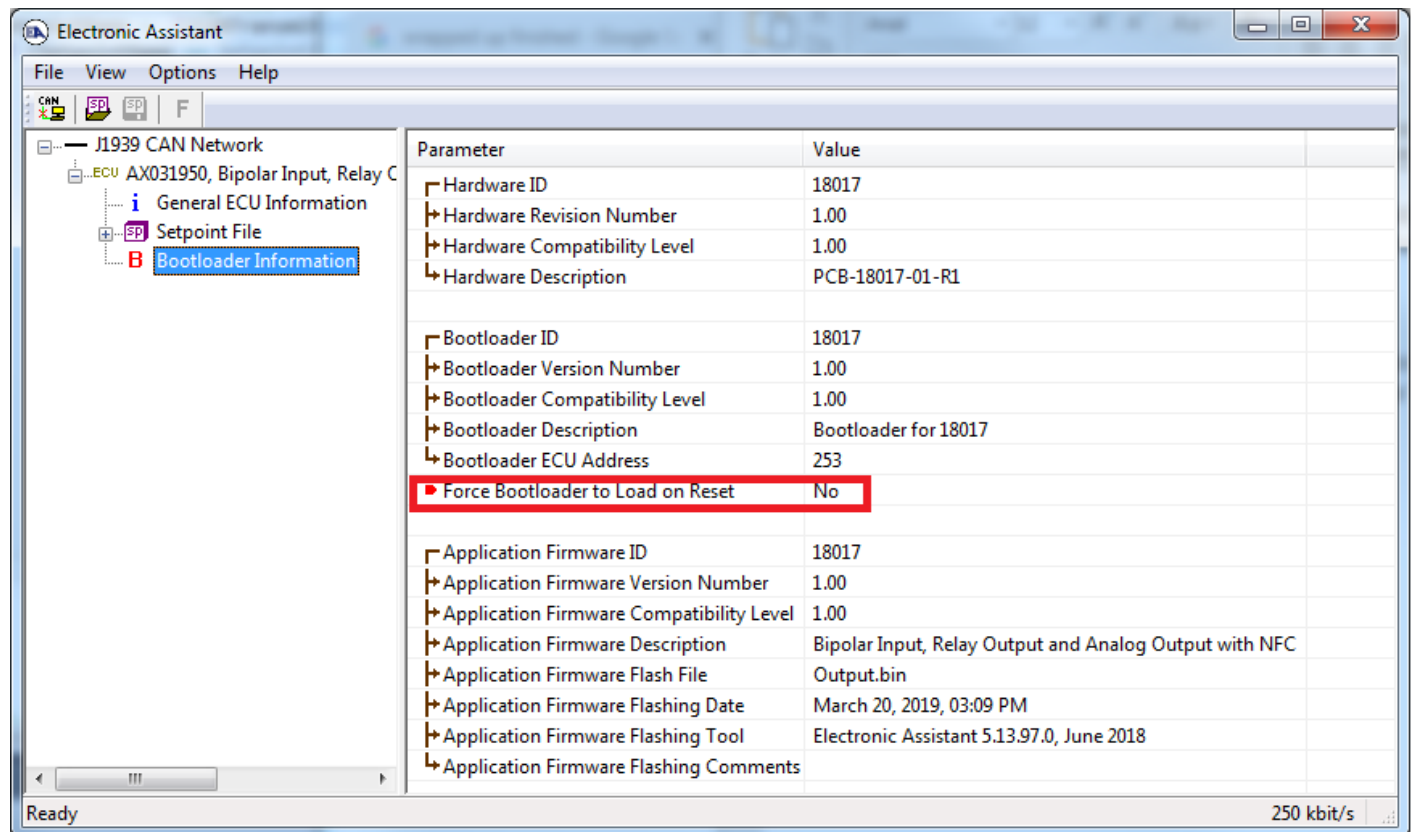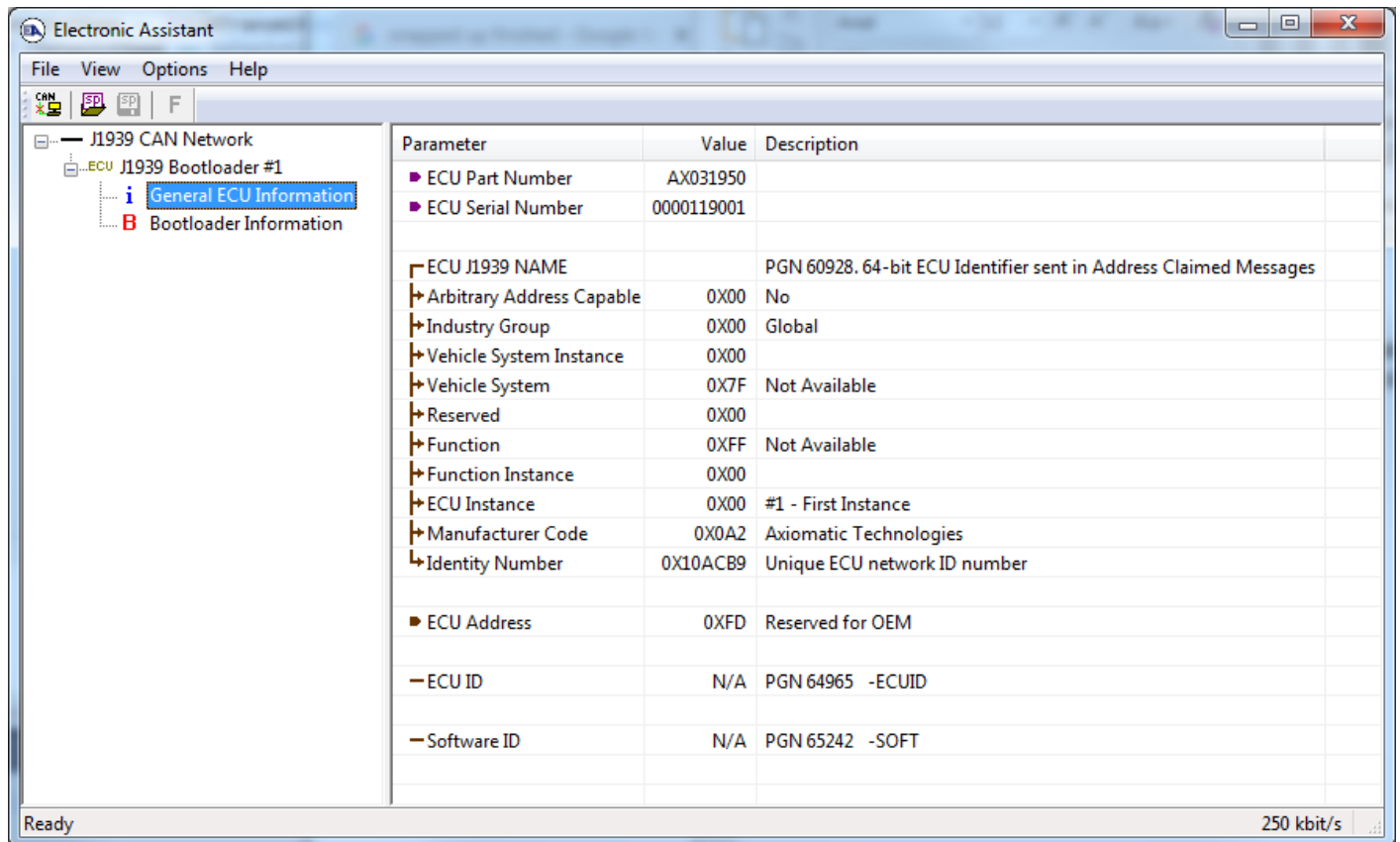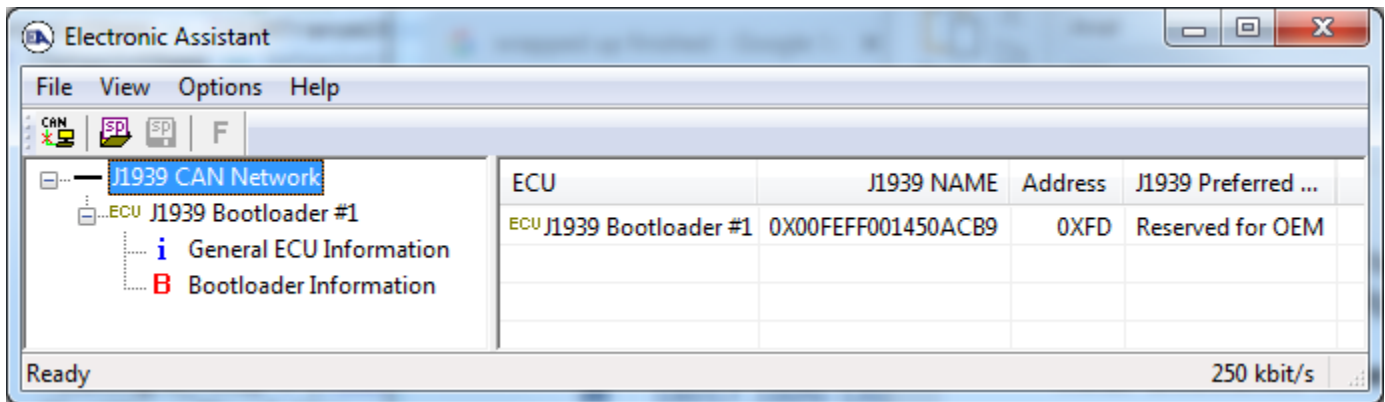

2. To use the bootloader to upgrade the firmware running on the ECU, change the variable **"Force Bootloader To Load on Reset"** to *Yes.*
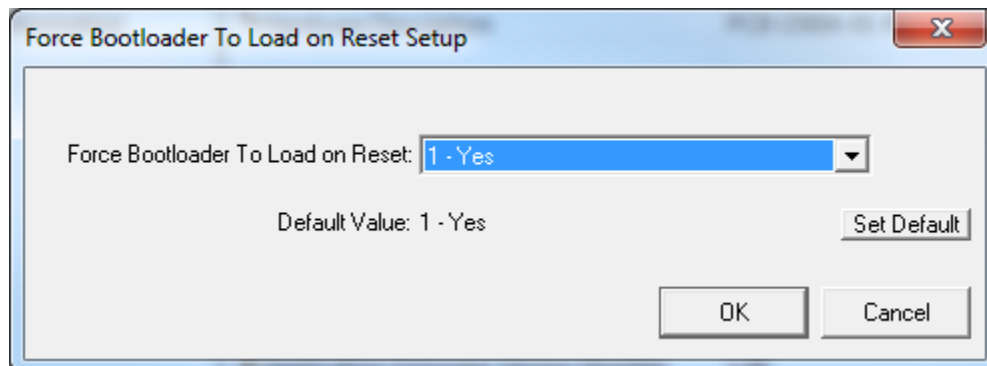
3. When the prompt box asks if you want to reset the ECU, select *Yes.*

Electronic Assistant

❓ Do you want to reset the ECU after changing this parameter ?

[Yes]   [No]
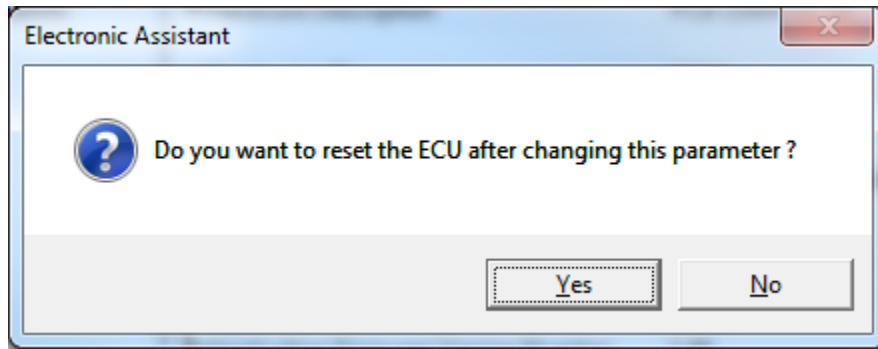
4. Upon reset, the ECU will no longer show up on the J1939 network as an AX031950 but rather as **J1939 Bootloader #1**.

ECU	J1939 NAME	Address	J1939 Preferred ...
ECU J1939 Bootloader #1	0X00FEFF001450ACB9	0XFD	Reserved for OEM

Ready — 250 kbit/s

Parameter	Value	Description
▸ ECU Part Number	AX031950	
▸ ECU Serial Number	0000119001	
⌐ECU J1939 NAME		PGN 60928. 64-bit ECU Identifier sent in Address Claimed Messages
┝ Arbitrary Address Capable	0X00	No
┝ Industry Group	0X00	Global
┝ Vehicle System Instance	0X00	
┝ Vehicle System	0X7F	Not Available
┝ Reserved	0X00	
┝ Function	0XFF	Not Available
┝ Function Instance	0X00	
┝ ECU Instance	0X00	#1 - First Instance
┝ Manufacturer Code	0X0A2	Axiomatic Technologies
┕ Identity Number	0X10ACB9	Unique ECU network ID number
▸ ECU Address	0XFD	Reserved for OEM
─ ECU ID	N/A	PGN 64965 -ECUID
─ Software ID	N/A	PGN 65242 -SOFT

Ready — 250 kbit/s

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 AX031950 firmware, but in this case the <u>F</u>lashing feature has been enabled.
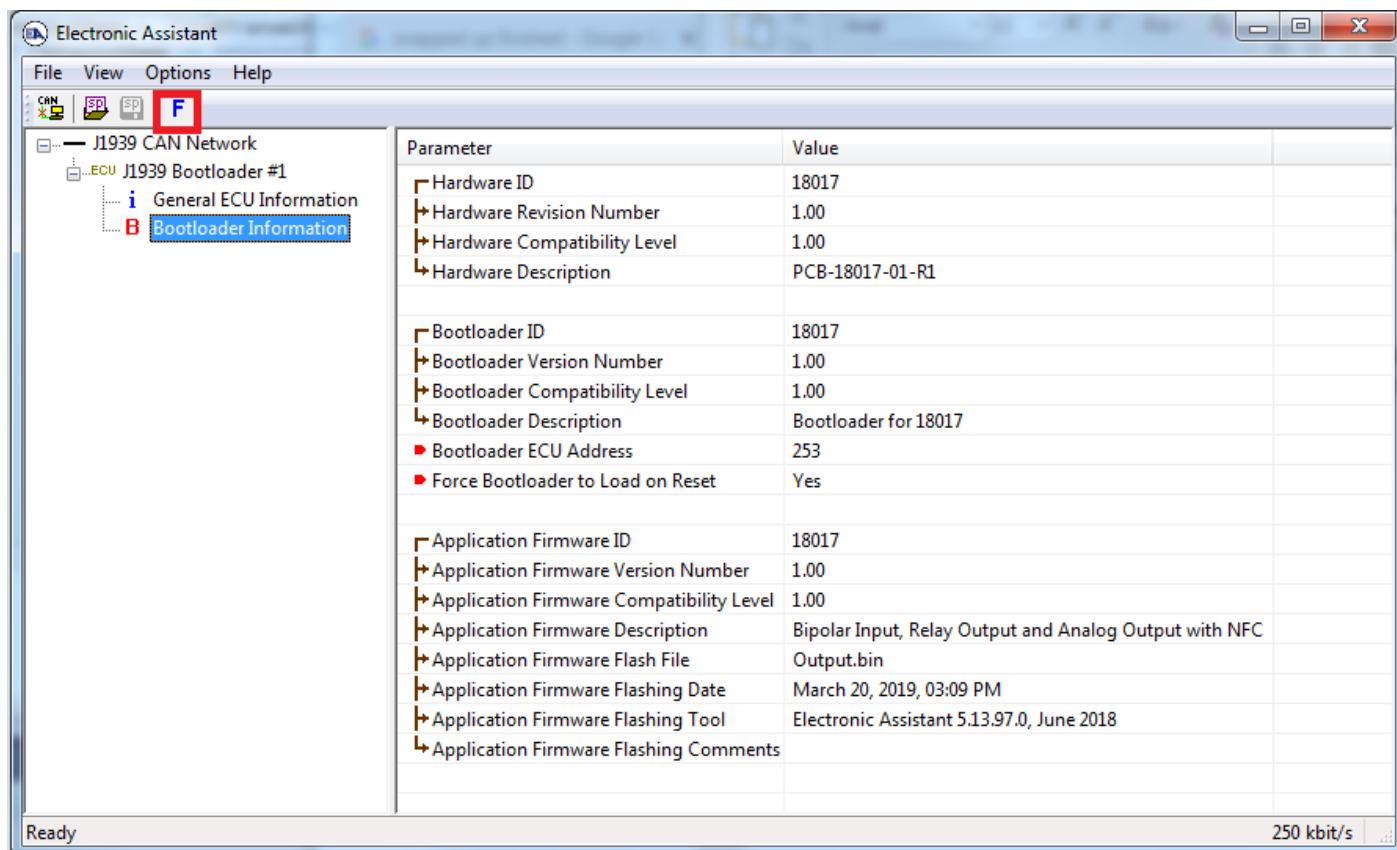
6. Select the <u>F</u>lashing button and navigate to where you had saved the **AF-18017_x.yy.bin** file sent from Axiomatic. (Note: only binary (.bin) files can be flashed using the Axiomatic EA tool)

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 all done automatically by the Axiomatic EA tool when you upload the new firmware.
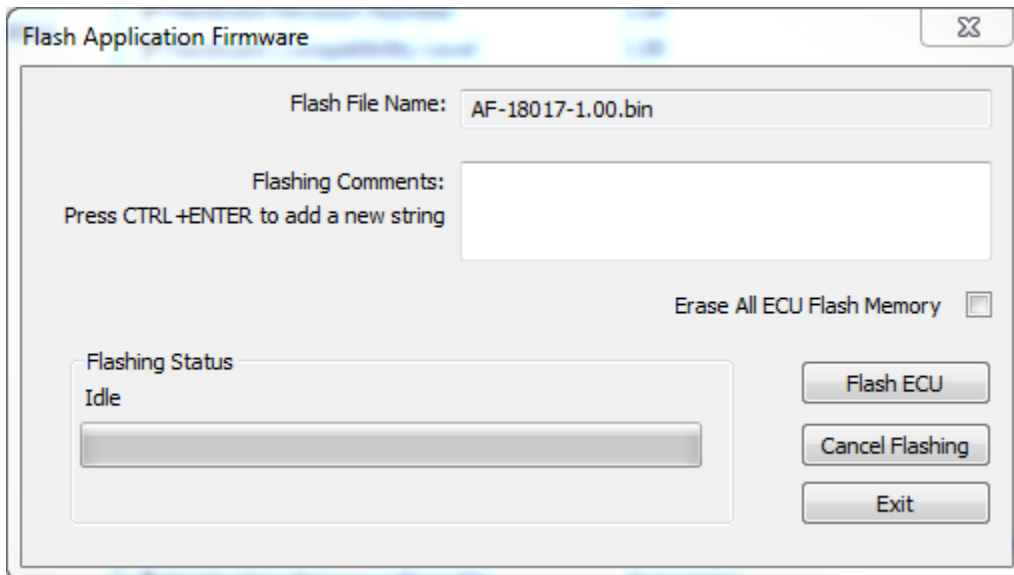

⚠	WARNING: Do not check the "Erase All ECU Flash Memory" box unless instructed to do so by your Axiomatic contact. Selecting this will erased ALL data stored in non-volatile flash, including the calibration done by Axiomatic during factory testing. It will also erase any configuration of the setpoints 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.
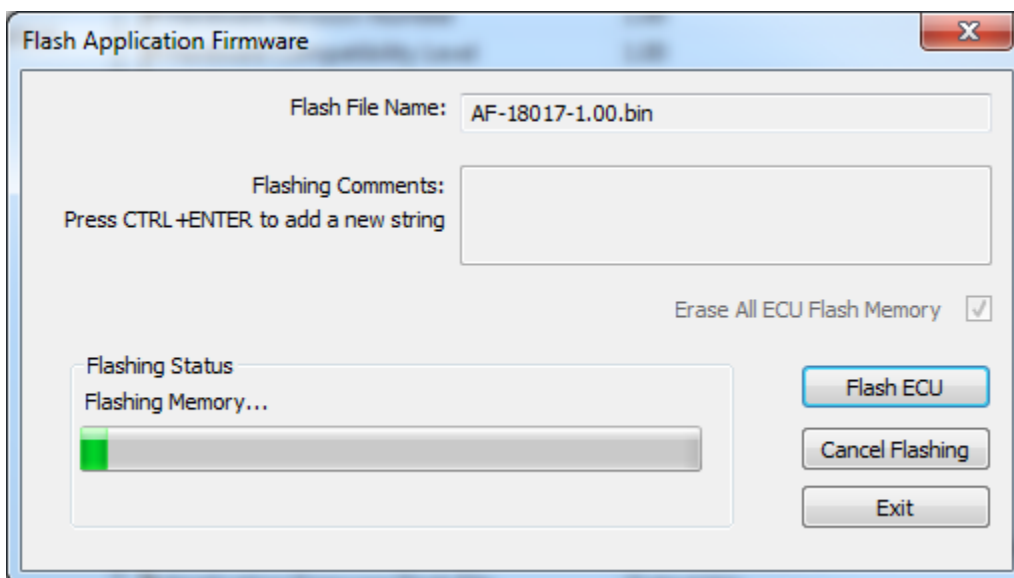
9. Once the firmware has finished uploading, a message will popup indicating the successful operation. If you select to reset the ECU, the new version of the AX031950 application will start running, and the ECU will be identified as such by the Axiomatic EA. Otherwise, the next time the ECU is power-cycled, the AX031950 application will run rather than the bootloader function.
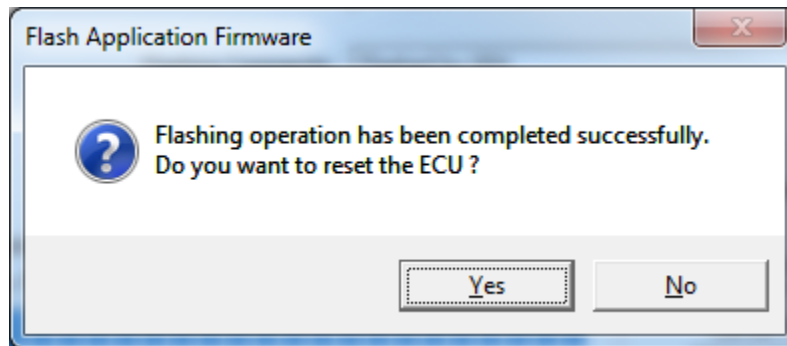
Note: If at any time during the upload the process is interrupted, the data is corrupted (bad checksum) or for any other reason the new firmware is not correct, i.e. bootloader detects that the file loaded was not designed to run on the hardware platform, the bad or corrupted application will not run. 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.

6. Technical Specifications

Specifications are indicative and subject to change. Actual performance will vary depending on the application and operating conditions. Users should satisfy themselves that the product is suitable for use in the intended application. All our products carry a limited warranty against defects in material and workmanship. Please refer to our Warranty, Application Approvals/Limitations and Return Materials Process as described on https://www.axiomatic.com/service/.

Power

Power Supply Input	12 or 24 VDC nominal (4 to 36 VDC range)
Typical Quiescent Current	20 mA @ 12 VDC; 35 mA @ 24VDC
Protection	Reverse polarity protection up to 60 V Under-voltage protection is down to 4.5 V Overvoltage protection is up to 40 V
Reference Voltage	1 +5 V, ±1%, 100 mA

Input

Input	1 universal signal input User selectable as bipolar or unipolar voltage, current, digital, frequency, or PWM input types. Refer to Table 1.0.
Table 1.0 - User Programmable Universal Signal Inputs	
Analog & Digital Input Functions	Voltage input, current input, or digital input
Voltage Input	0-5 V (impedance 100 kΩ) 0-10 V (impedance 100 kΩ) ±5 V (impedance 100 kΩ) ±10 V (impedance 100 kΩ) Resolution: ±1 mV Accuracy: ±0.5%
Current Input	0-20 mA (impedance 124 Ω) 4-20 mA (impedance 124 Ω) Resolution is +/- 1 µA. Accuracy is +/- 0.2%
Digital Input Level	Accepts 5 V TTL Accepts up to Vps Threshold: Low <1 V 　　　　　High >2.2 V
Digital Input	1 MΩ impedance or active high or active low with 10 kΩ pull-up or pull-down
Timer Input Functions	PWM input or frequency input
PWM Input	Low frequency (0.50 Hz to 1 kHz) High frequency (100 Hz to 10 kHz) 0 to 100% D.C. Resolution: 0.01% Accuracy: ±0.1%
Frequency/RPM Input	0.5 Hz to 50 Hz; 10 Hz to 1 kHz; or 100 Hz to 20 kHz Resolution: 0.01% Accuracy: ±0.1%

Maximum and Minimum Ratings	Characteristic	Min	Max	Units
	Power supply	9	36	VDC
	Voltage input	0	36	VDC
	Current input 0(4)-20 mA	0	12	VDC
	Digital input	0	36	VDC
	PWM duty cycle	0	100	%
	PWM low frequency	10	1 000	Hz
	PWM high frequency	100	10 000	Hz
	PWM voltage pk-pk	0	36	VDC
	Frequency	0	20 000	Hz

GND	1 analog GND connection is provided.

Outputs

Output	1 Form C relay NC 3 contact pins per output Maximum 2 A @ 250 VAC or 2 A @ 30VDC per contact

Analog Output	1 analog output, user selectable as: • 0-5 V, 0-10 V (±0.2% accuracy) • 0-20 mA, 4-20 mA (±0.2% accuracy)
Analog Output GND	1 GND connection is provided.

General Specifications

Microcontroller	STM32F205RE, 32-bit, 512 KB flash program memory
Isolation	300 Vrms CAN bus isolation
Control Logic	Standard embedded software is provided.
Communications	1 isolated CAN port (SAE J1939) Model AX031950: 250 kbps Model AX031950-01: 500 kbps Model AX031950-02: 1 Mbps
NFC Communications	Near Field Communication (NFC) Full-duplex Data rate: 106 kbit/s Complies with ISO1443 (RF protocol), ISO13239, and ISO7816 Protected and secure configuration
User Interface via NFC	E-Write NFC Application is available from the Google Play Store for simple configurations.
Software Reflashing	Via the Axiomatic Electronic Assistant, P/Ns: AX070502 or AX070506K
User Interface via CAN	The Axiomatic Electronic Assistant, P/Ns: AX070502 or AX070506K
LED Indicator	1 RED and GREEN LED with application-specific responses
Operating Conditions	-40°C to 85°C (-40°F to 185°F)
Storage Temperature	-55°C to 125°C (-67°F to 257°F)
Weight	Contact Axiomatic.
Protection	IP40 (enclosure), IP20 (terminals)
Enclosure	DR12, 35 mm DIN rail mount Polycarbonate 75 mm x 98.5 mm x 22.5 mm (W x H x D) 2.95 in x 3.88 in x 0.89 in Refer to the dimensional drawing.
Electrical Connections	12 screw terminal connections Refer to dimensional drawing for pin out orientation.

Screw Terminal #	Description
1	Relay Output COM 1
2	Relay Output NC 1
3	Relay Output NO 1
4	+5V Reference
5	Analog Output
6	Analog Output Return
7	Analog Input
8	CAN_H
9	CAN_L
10	Power +
11	Power -
12	Analog Input GND

7. VERSION HISTORY

Version	Date	Author	Modifications
V1	April 25th, 2019	Jessica Chen	Initial Draft
-	September 17, 2019	Amanda Wilkins	Marketing Review, Added Technical Specifications, Added Dimensions and Pin out
-	April 12, 2021	Jessica Chen	Modified Table of Contents
V1.0.1	May 25, 2023	M Ejaz	Fixed legacy issues Copied specifications from datasheet into user manual
V1.0.2	August 2, 2023	Kiril Mojsov	Performed Further Legacy Updates

OUR PRODUCTS

AC/DC Power Supplies

Actuator Controls/Interfaces

Automotive Ethernet Interfaces

Battery Chargers

CAN Controls, Routers, Repeaters

CAN/WiFi, CAN/Bluetooth, Routers

Current/Voltage/PWM Converters

DC/DC Power Converters

Engine Temperature Scanners

Ethernet/CAN Converters,
Gateways, Switches

Fan Drive Controllers

Gateways, CAN/Modbus, RS-232

Gyroscopes, Inclinometers

Hydraulic Valve Controllers

Inclinometers, Triaxial

I/O Controls

LVDT Signal Converters

Machine Controls

Modbus, RS-422, RS-485 Controls

Motor Controls, Inverters

Power Supplies, DC/DC, AC/DC

PWM Signal Converters/Isolators

Resolver Signal Conditioners

Service Tools

Signal Conditioners, Converters

Strain Gauge CAN Controls

Surge Suppressors

OUR COMPANY

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

QUALITY DESIGN AND MANUFACTURING
We have an ISO9001:2015 registered design/manufacturing facility in Canada.

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

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

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

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

SERVICE
All products to be returned to Axiomatic require a Return Materials Authorization Number (RMA#) from sales@axiomatic.com. Please provide the following information when requesting an RMA number:
- Serial number, part number
- Runtime hours, description of problem
- Wiring set up diagram, application and other comments as needed

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

CONTACTS

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

Axiomatic Technologies Oy
Höytämöntie 6
33880 Lempäälä
FINLAND
TEL: +358 103 375 750

www.axiomatic.com
salesfinland@axiomatic.com