# 4 AC INPUTS,

# 4 ANALOG OUTPUTS

# CAN CONTROLLER

## USER MANUAL

**P/N: AX130550**

# Table of Contents

**ACRONYMS**

| | | |
|---|---|---|
| ACK | Positive Acknowledgement | (from SAE J1939 standard) |
| ACIN | Alternative Current Input: Current or Voltage | |
| AOUT | Analog Output: Current or Voltage type | |
| DM | Diagnostic Message | (from SAE J1939 standard) |
| DOUT | Digital Output | |
| DTC | Diagnostic Trouble Code | (from SAE J1939 standard) |
| EA | Axiomatic Electronic Assistant (A Service Tool for Axiomatic ECUs) | |
| ECU | Electronic Control Unit | (from SAE J1939 standard) |
| 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 the Proprietary B PGN for broadcast communication | |
| PWM | Pulse Width Modulation | |
| SPN | Suspect Parameter Number | (from SAE J1939 standard) |

---

**Note:**

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

---

**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 |
| TDAX130550 | Technical Datasheet |
| UMAX07050x | User Manual, Axiomatic Electronic Assistant and USB-CAN, Axiomatic Technologies |

*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: 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: "**Output Type**" set to *'Analog Current'*

NOTE:

This product uses the Axiomatic Electronic Assistant, P/Ns: AX070502 or AX070506K, 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 AX130550 controllers over the CAN network.

# 1. OVERVIEW OF CONTROLLER

## 1.1. Description of 4 VAC or Current Inputs to 4 Analog Signal Output CAN Controller

This User Manual describes the architecture and functionality of the 4 AC Inputs to 4 Analog Outputs Controller (4ACIN-4AOUT). It accepts power supply voltages from 9 to 36 VDC. All logical function blocks on the unit are inherently independent from one another but can be configured to interact with each other. All parameters are configurable using the Axiomatic Electronic Assistant. Figure 1 below shows the hardware features. The J1939 CAN network can operate at standard 250 and 500kbit/s and non-standard 667kbit/s and 1Mbit/s baud rates. The required baud rate is detected automatically upon connection to the CAN network.
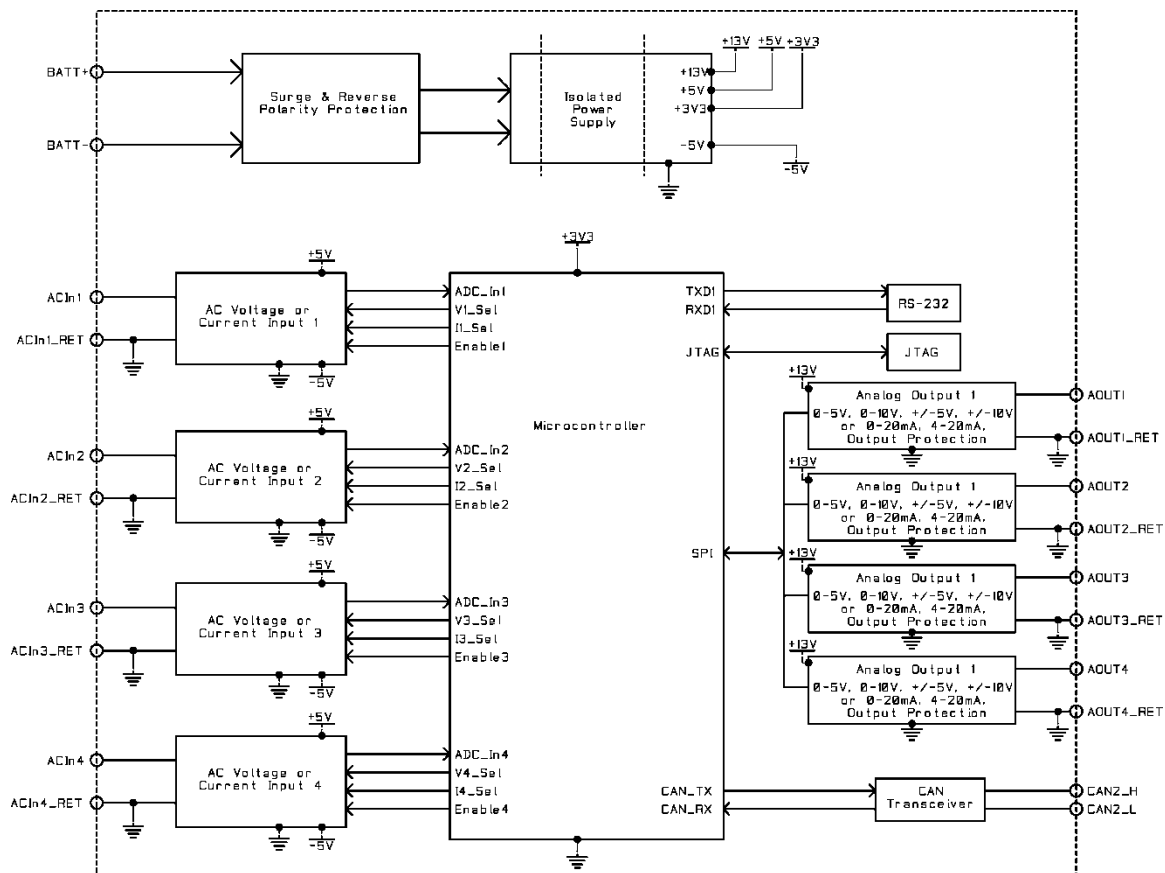


**Figure 1 - Hardware Functional Block Diagram**

This controller is designed for versatile control of CAN bus to 4 analog/digital outputs. 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 hardware design allows for the controller to have the following input types: *AC Current and AC Voltage.* Input Blocks are described in more details in section 1.2.

## 1.2.    Input Function Block

The 4ACIN-4AOUT controller has 4 configurable AC Inputs that work independently of each other. The **"Input Type"** parameter defines the type of measured signal. Changing this parameter will update all dependent parameters according to the type selected. For this reason, this parameter should be changed first. The supported input types by the controller are listed in Table 1 below. The default state of the **"Input Type"** parameter is '*1, AC Voltage*' type.

| Value | Meaning |
|-------|---------|
| 0 | *Disabled* |
| **1** | ***AC Voltage*** |
| 2 | *AC Current* |

**Table 1- Analog Input Type Options**

Units of measurements depend on the input type set in the ***"Input type"*** setpoint: voltage peak-to-peak [Vpp] for '*AC Voltage'*, and milliamps peak-to-peak [mApp] for '*AC Current'* type. Also, there is an additional functionality that allows to use the Root Mean Square (RMS) values as a control source.

To provide smooth measurements, the software filtering feature is provided. There are three software filtering options that can be set under **"Software Filter Type"**. They are listed in Table 2 below.

| Value | Meaning |
|-------|---------|
| **0** | ***No Filter*** |
| 1 | *Moving Average* |
| 2 | *Repeating Average* |

**Table 2 - Software Filter Type Options**

### 1.2.1 AC Current/AC Voltage

Current Inputs, as well as Voltage Inputs limits can be configured within the predefined ranges from 0 to 5.0 Vpp and 0 to 200.0 mApp for Voltage and Current minimum input and minimum input error respectively. To set these parameters user needs to provide required values under **"Minimum Input"** and **"Minimum Input Error"**. The limits for **"Maximum Input"** and **"Maximum Input Error"** are varying from 0 to 5Vpp and 0 to 200 mApp for voltage and current.

### 1.2.2 Common Parameters

There also are the following setpoints under AC Input section:
- **"Minimum Input Error"**: setpoint is configurable by the user and represents the minimum value to trigger the AC Input Under-Range Error.
- **"Minimum Input"**: setpoint is set by user to represent the value that clears the AC Input Under-Range Error.

- **"Maximum Input"**: this setpoint serves the similar purpose as the **"Minimum Input"**, but for the AC Input Over-Range Error.
- **"Maximum Input Error"**: setpoint that represent the value needs to be exceeded to trigger the AC Input Over-Range Error.

*Note: if the* **"Minimum Input Error"** *is set to the same value as* **"Minimum Input"** *and they are equal to 0, the AC Input Under-Range Error will not be detected. Additionally, if the* **"Maximum Input Error"** *is equal to* **"Maximum Input"** *and they equal to the maximum possible input (5.0Vpp or 200mApp), the AC Input Over-Range Error will not be detected.*

- **"Software Filter Constant"**: it represents the number of samplings being taken before calculating the average. This parameter is the same for both moving and repeating average.

Also, there are AC Input Fault diagnostic within each AC Input group. The detailed information about this part is stated in section 1.10.

### 1.2.3 Software Filtering

There are three options under **"Software Filter Type"** with two actual types of software filtering: *'Moving Average'* and *'Repeating Average'*. Both are used to smoothen the actual sampling and provide more precise data.

If customer use the *'Moving Average'* type under the **"Software Filter Type"** setpoint the unit will collect the number of samples equal to the value under the **"Software Filter Constant"** setpoint and then calculate the average. Each new sample will replace the oldest sample and the average will be recalculated during each new iteration. The algorithm is shown below:

Moving Average:

1. Calculating the first average:

$$N_{avg} = \frac{\sum_{i=0}^{C} N_i}{C} \text{ (1), where}$$

*C – Software Filter Constant value;*
*$N_i$ – the number of current value;*
*$N_{avg}$ – Average sample.*

2. Replacing the latest item and recalculating the average using formula (1):

$$N_{avg}^{(i)} = \frac{N_{avg}^{(i-1)} - N_l + N_{new}}{C}\text{(2), where}$$

*C – Software Filter Constant value;*
*$N_{avg}^{(i)}$ – the new average sample;*
*$N_{avg}^{(i-1)}$ – a previous average sample;*
*$N_l$ – the oldest sample;*
*$N_{new}$ – a current sample.*

After first average has been calculated by using formula (1), the second formula is applied during each iteration.

Repeating Average:

The repeating average method is similar to the moving average except that instead of recalculating the average after each new sample, the unit will gather the required number of samples each time instead.

## 1.3. LEDs Function Block

The AX130550 unit has two configurable three-state LEDs to represent the information required for the application. Each LED has 4 separate stages that can be configured independently, as well as a default state. The control of LEDs is carried out through a unique list of control sources shown below in Table 3.

| Value | Meaning | Control Number Range |
|---|---|---|
| **0** | ***Control Not Used*** | **[1]** |
| 1 | *AC Input* | *[1…4]* |
| 3 | *CAN Receive Message* | *[1…10]* |
| 4 | *Constant Discrete Data* | *[1…5]* |
| 5 | *Constant Continuous Data* | *[1…5]* |
| 6 | *Lookup Table Block* | *[1…10]* |
| 7 | *Math Function Block* | *[1…4]* |
| 8 | *Programmable Logic Block* | *[1…3]* |
| 9 | *Temperature Warning* | *[1]* |
| 9 | *Temperature Shut Down* | *[1]* |
| 9 | *AC Input Fault* | *[1…4]* |

**Table 3 – List of Control Sources for LEDs**

*"Stage Control Source"* setpoint should be depending on the severing/importance of what is needed to be displayed be the LEDs for the different stages. By default, all stage control sources are set to 0 - *'Control Not Used'*. In this case, all other setpoints correlated with the corresponding stage are not used.

Each stage provides a separate configuration for one LED with a different priority level: from the most urgent (Stage 1) to the least urgent (Stage 4). If all stages are inactive, the default stage will be active. In this case, LEDs can be configured to represent a status of needed parameters in a suitable manner.

Besides *"Stage Control Source"*, these are the setpoints for each stage:

- *"Stage Control Number".* This setpoint is used to connect the LED to required control parameter. For example, if under *"Stage Control Source"* *'AC Input'* is selected and the *"Stage Control Number"* is equal to 1, that means that the current stage is controlled by AC Input 1.

- *"Stage Output Type".* This setpoint is responsible for setting the color of the LED. All four options are listed in Table 4 below.

| Value | Meaning |
|---|---|
| 0 | *LEDs Disabled* |
| 1 | *Green* |
| **2** | ***Yellow*** |
| 3 | *Red* |

**Table 4– Stage Output Type Options**

- **_"Stage Output Response Type"._** This setpoint determines one of three possible LED response pattern. All of them are listed in the Table 5.

| Value | Meaning |
|-------|---------|
| **0** | **_Normal On/Off_** |
| 1 | _Blink Logic_ |
| 2 | _Dim On/Off_ |

**Table 5 – Stage Output Response Type Options**

- **_"Stage Blink Rate"._** Can be configured only if _'Blink Logic'_ is selected under the **_"Stage Output Response Type"_** setpoint. It determines the blinking rate in milliseconds.

All mentioned setpoints except **_"Stage Control Source"_** and **_"Stage Control Number"_** are applicable for a default stage. Initially the default stage is turned on, and it signals that all other control stages are inactive.

## 1.4. Output Function Block

The 4ACIN-4AOUT controller has 4 configurable analog outputs and they are inherently independent of one another. The **"Output Type"** parameter determines what kind of signal the output produces. Changing this parameter will update other parameters in the group to match the selected type. For this reason, it should be the first parameter changed. The supported controller output types are listed in Table 6 below. By default, outputs are configured as '_2, Analog Voltage_' type.

| Value | Meaning |
|-------|---------|
| 0 | _Output Not Used_ |
| 1 | _Analog Current_ |
| **2** | **_Analog Voltage_** |

**Table 6- Analog Output Type Options**

The output control signals will have associated minimum and maximum values. All output types always respond in a linear fashion to changes in the control source, per the calculations 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**

Where 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 for output to follow the control signal inversely.

By default, when the outputs are in Current/Voltage mode, outputs are driven with a high frequency 25kHz signal. This can be changed through the setpoint **"Fixed Frequency/Duty Cycle"**. However, this value should not be lowered under these two modes without taking into consideration how this affects the accuracy and stability of the outputs. Since all four outputs are connected to independent timers, this parameter can be changed at any time for each output without affecting the other.

### 1.4.1  Analog Current/Analog Voltage

Current Outputs can be configured to different ranges such as 0-20mA and 4-20mA and Voltage Outputs can be configured to 0-5V, 0-10V, +/-5V and +/-10V. Simply setting the **"Output at Minimum Command"** and **"Output at Maximum Command"** to the corresponding value in each range will drive the output to different range options. The unit of measurement for the current output variables is milliamps [mA], and the unit of measurement for voltage outputs is volts [V].

### 1.4.2  Common Parameters

In order to prevent abrupt changes to 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 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. However, these setpoints are set to zero by default, since in most signal conversion applications, fast response times are desired.

By default, the **"Control Source"** is set to be *'AC Input'.* In other words, all the outputs will respond in a linear fashion to the corresponding AC Input on the selected input channel.

The **"Control Source"** together with the **"Control Number"** parameter determine which signal is used to drive the output. For example, setting **"Control Source"** to *'AC Input'* and **"Control Number"** to *'1'* will connect signal measured from AC Input channel 1 to the output in question. The options for **"Control Sources"** and available **"Control Number"** are listed in Table 7

| Value | Meaning | Control Number Range |
|---|---|---|
| 0 | *Control Not Used* | [1] |
| **1** | ***AC Input*** | **[1…4]** |
| 2 | *AC Input RMS* | [1…4] |
| 3 | *CAN Receive Message* | [1…10] |
| 4 | *Constant Discrete Data* | [1…5] |
| 5 | *Constant Continuous Data* | [1…5] |
| 6 | *Lookup Table Block* | [1…10] |
| 7 | *Math Function Block* | [1…4] |
| 8 | *Programmable Logic Block* | [1…3] |
| 9 | *Temperature Measured* | [1] |

**Table 7– Control Sources and Numbers**

In general, when the control input is within its range, outputs will respond in a linear fashion to changes in the control source, as per the calculation in Figure 2, and when the control input is out of range, outputs will respond with the minimum/maximum value in the output range. However, in some cases not having the minimum offset applied when the value is outside of the range may be desirable, i.e. when using a joystick profile with a deadband. For this reason, setpoint "**Control Response**" has the options shown in Table 8.

| Value | Meaning |
|---|---|
| 0 | Single Output Profile |
| **1** | **Off Below Control Minimum** |
| 2 | Off Above Control Maximum |

**Table 8– Control Responses Options**

In addition to the Control input, the function block also supports an Enable input which can be set up as either an enable or disable signal.

When an Enable input is used, the output will shut off as per the **"Enable Response"** in Table 9. If the response is selected as a Disable signal (3 or 4), when the Enable input is ON, the output will be shut off.

| Value | Meaning |
|---|---|
| 0 | *Enable When On, Else Shutoff* |
| 1 | *Enable When On, Else Rampoff* |
| 2 | *Enable When On, Else Keep Last Value* |
| **3** | ***Enable When Off, Else Shutoff*** |
| 4 | *Enable When Off, Else Rampoff* |
| 5 | *Enable When Off, Else Keep Last Value* |

**Table 9 – Enable Response Options**

The Override option allows the user to choose whether to drive the output with the override input engaged/disengaged, depending on the logic selected in **"Override Response."** The options for **"Override Response"** are listed in Table 10. When override is active, the output will be driven to the value in "**Output at Override Command**" regardless of the value of the Control input.

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

**Table 10 – Override Response Options**

The options for both **"Enable Source"** and **"Override Source"** are the same as the sources listed in Table 7.

When an input to the output block goes into an error condition, setpoint "**Fault Response**" determines how the output will respond, per Table 11. By default, the output will be driven to the value defined in setpoint "**Output at Fault Command**" which is set to 0 by default.

| Value | Meaning |
|---|---|
| 0 | *Maintain Last State* |
| **1** | ***Output Fault Mode Value*** |

**Table 11 – Fault Response Options**

Another fault response which can be enabled is that of a microprocessor over-temperature fault automatically disabling the outputs until it has cooled back to the operating range, if the corresponding setpoint is enabled. This will be described in more detail in Section 1.10.

Fault conditions are checked for first, and the control signal will then be evaluated only if they are not present. 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 3 below.
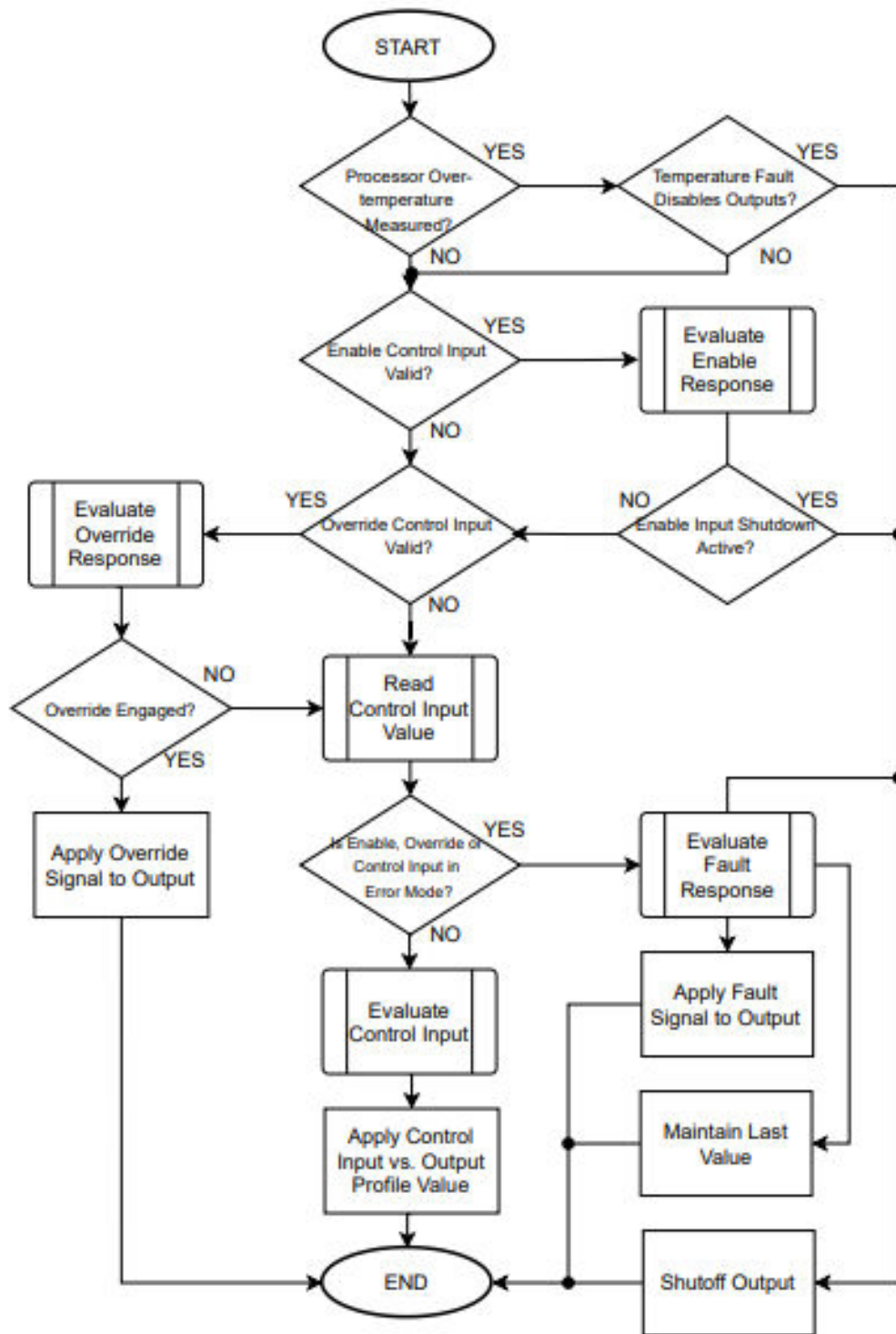
**Figure 3 – Output Logic Flowchart**
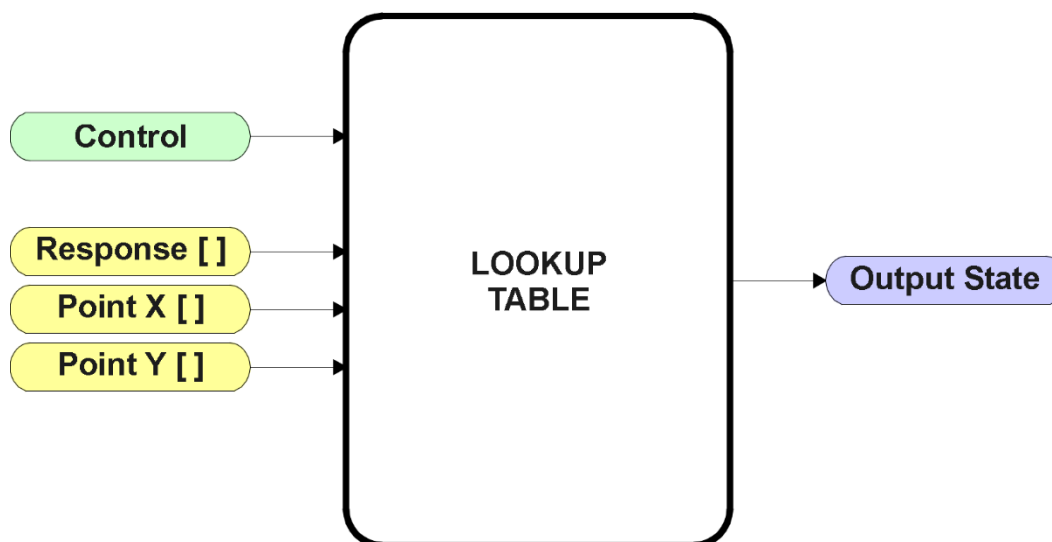
## 1.5. Lookup Table Function Block



**Figure 4 – 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 points are automatically updated with new defaults based on the X-Axis source selected. However, there are 2 more setpoints "**X Decimal Digits**" and "**Y Decimal Digits**" that might affect the point values. These 2 setpoints will change the resolution of the X Values or Y Values, respectively, with the decimal digits value user chooses. Keep in mind that any change on the decimal digits will make effect to the current X values or Y values.

> ⚠️ 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.

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 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.

**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, 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 another controller), 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.

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 Analog Output.

### 1.5.3.    Default Configuration, Data Response

By default, all Lookup Tables in the ECU 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 Analog Output 1. 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 Analog Output 1) 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 5) correspond to the minimum and maximum values of the table's Y-Axis (Xmin and Xmax in Figure 5).**

To control "Analog Output 1" by "CAN Received Message 1" modified by "Lookup Table 1", it is recommended to do so in the following order:
a) Change Analog Output 1 **"Output at Minimum Command"** and **"Output at Maximum Command"** to the desired limits.
b) Configure the desired Control Source (i.e. CAN Receive Message) and set the appropriate limits.
c) Change the Lookup Table 1 **"X-Axis Source"** setpoints. (If applicable)
   At this point, the X-Axis limits will match the control source, and the Y-Axis limits and the Y-Axis limits would correspond to the Analog Output 1 range, as a percentage.
d) Update the X and Y setpoints for the application

*Note: Order (b) to (d) holds true for all configuration done using any Lookup Table function block.*

All tables (1 to 10) are disabled by default (no control source selected). However, 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, a 100Ω to 54kΩ input (X-Axis) representing 120ºC to -30ºC (Y-Axis) would be setup as per figure (b) below. In each case, the user would have to adjust the table for the desired response.
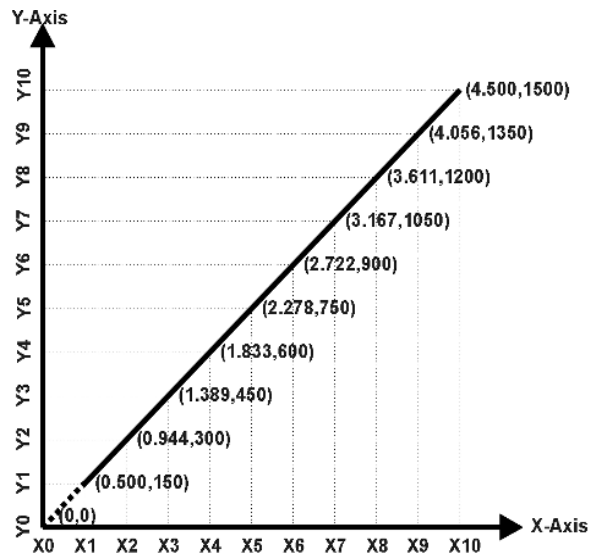


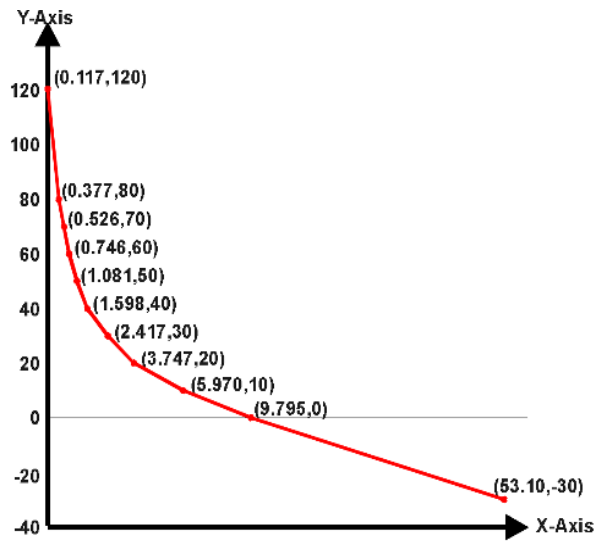Figure A - 0.5 to 4.5V Input, 0 to 1500mA Output    Figure B - 0.1 to 54kOhm Input, 120 to -30ºC Output

**Figure 5 – Lookup Table 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 2. 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 6.
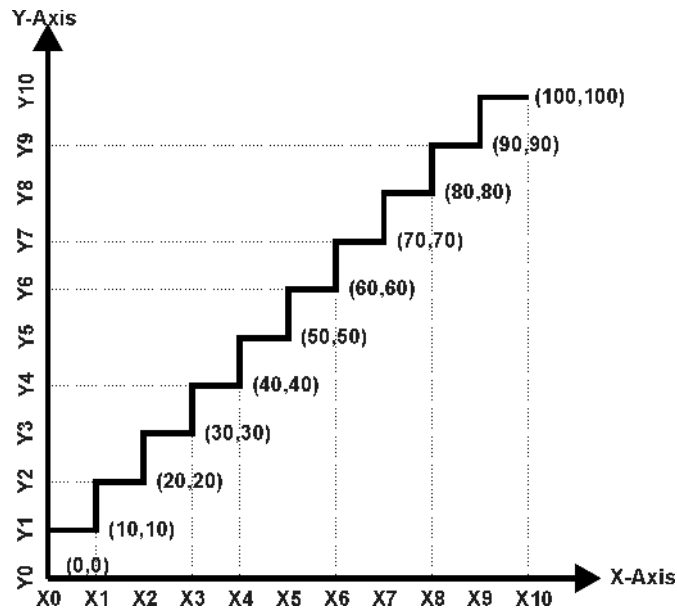
**Figure 6 – Lookup Table "Jump To" Data Response**

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 7. The example shows a dual slope output response for each side of the deadband, but additional slopes can be easily added as needed.
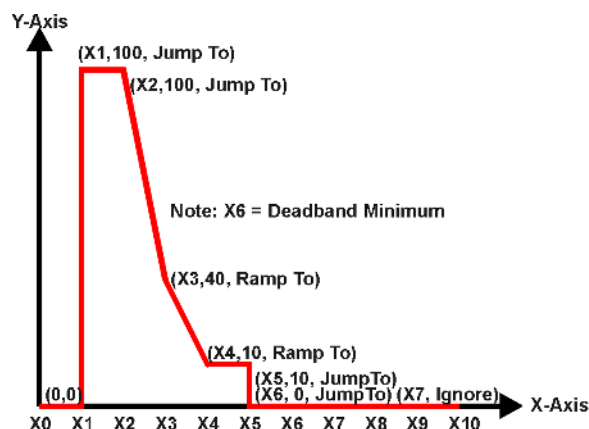


**Figure 7 – Lookup Table Examples to Setup for Joystick Deadband Response**

### 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 7. 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 Analog Output 1 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 Analog Output 1 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 8.
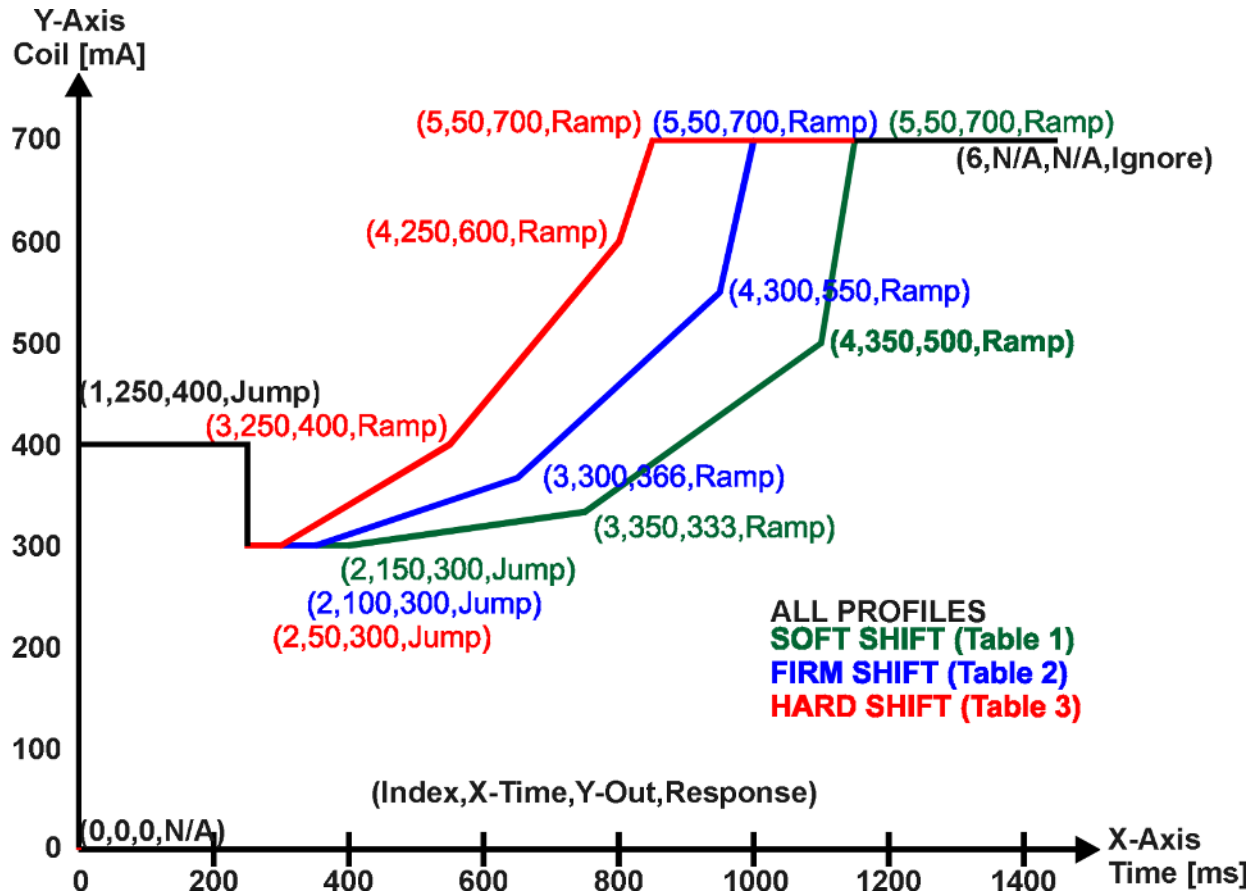


**Figure 8 – Lookup Table Time Response Clutch Fill Profiles**

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.

**Figure 9 – Lookup Table "Soft Shift" Axiomatic EA Configuration**

## 1.6. Programmable Logic Function Block
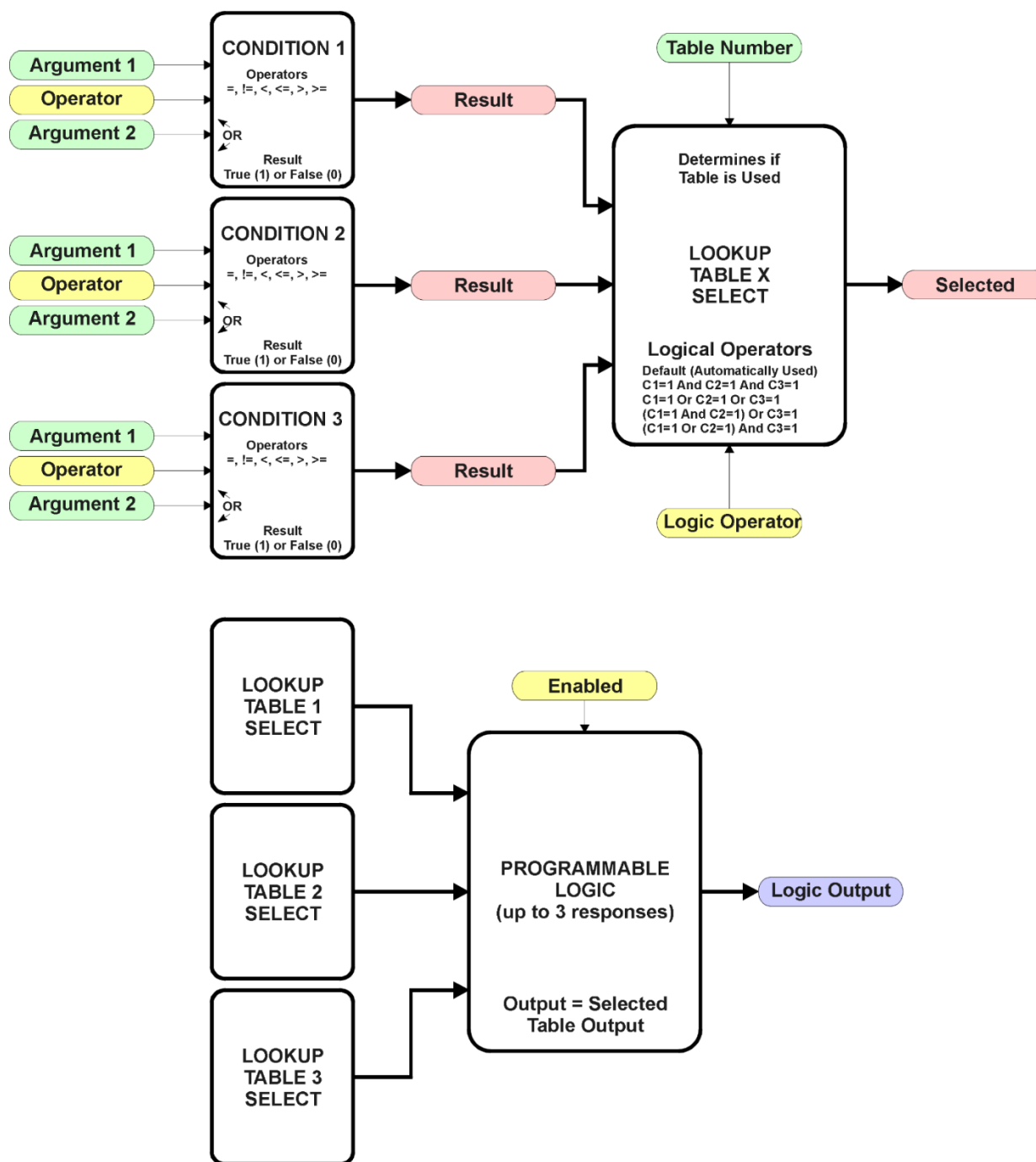


**Figure 10 – Programmable Logic Function Block**

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 10) 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 Analog Output 1. 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 11. 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.**
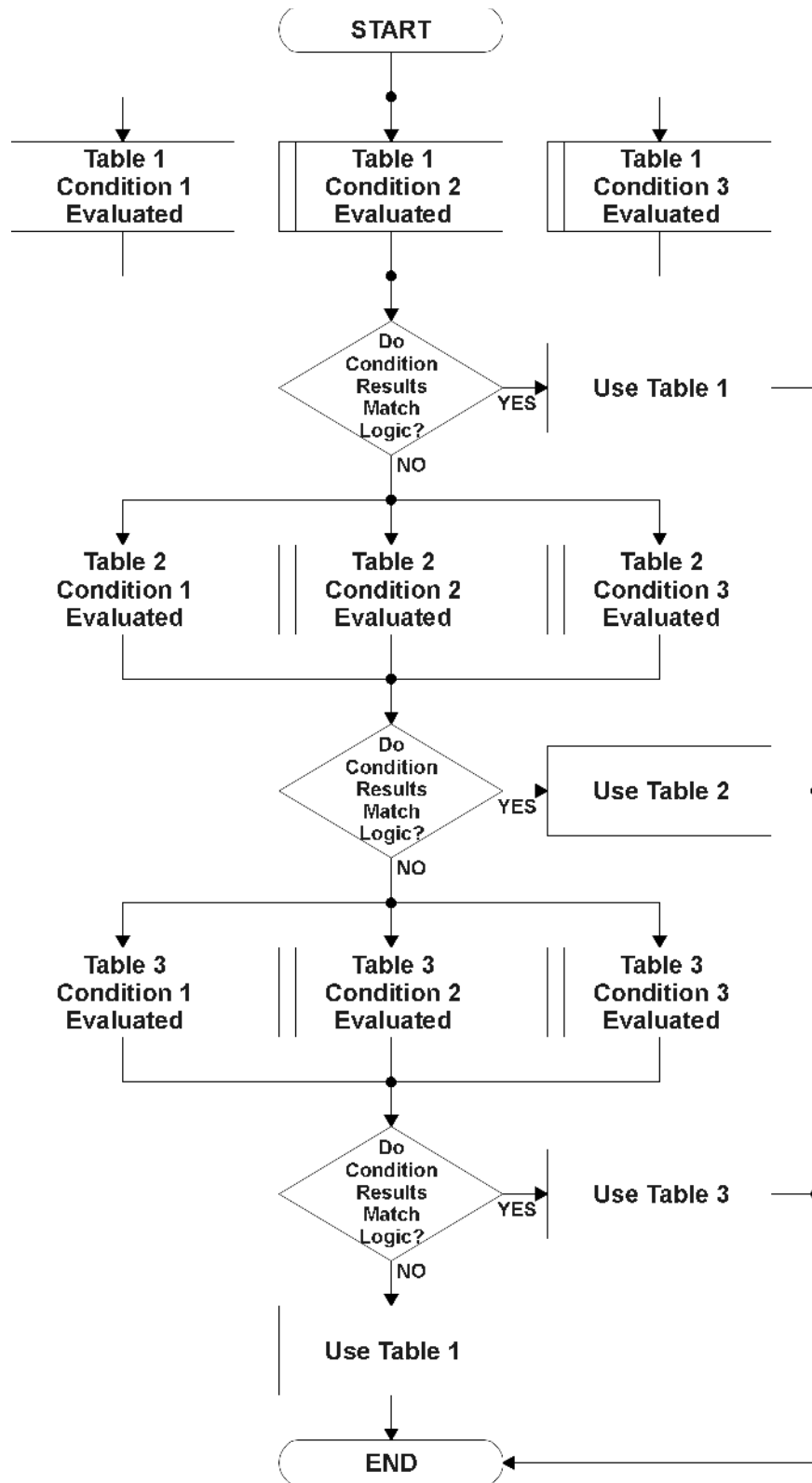
**Figure 11 – Programmable Logic Flowchart**

### 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 Z is always a logical output from another function block. 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.

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 12.

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

**Table 12 – Condition Operator Options**

For example, a condition for a transmission control shift selection, as shown in Figure 11 in the previous section, 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 *'CAN Receive Message 3',* **"…Argument 2 Source"** to *'Constant Continuous Data 1*, and the **"…Operator"** to *'<, Less Than.'*

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 11 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 13

| 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 13 – 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. There are several logical combinations that can be selected, as listed in Table 14.

| 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 14 – 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.**<br><br>If any condition equals False or Error, the table is not selected.<br>An N/A is treated like a True.<br>If all three conditions are True (or N/A), the table is selected.<br><br>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.**<br><br>If any condition is evaluated as True, the table is selected. Error or N/A results are treated as False<br><br>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.**<br><br>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<br><br>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.**<br><br>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<br><br>If(  ((Cnd1==True)||(Cnd2==True)) && (Cnd3==True) ) Then Use Table |

**Table 15 – 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 Table 16.

| 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 16 – 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.
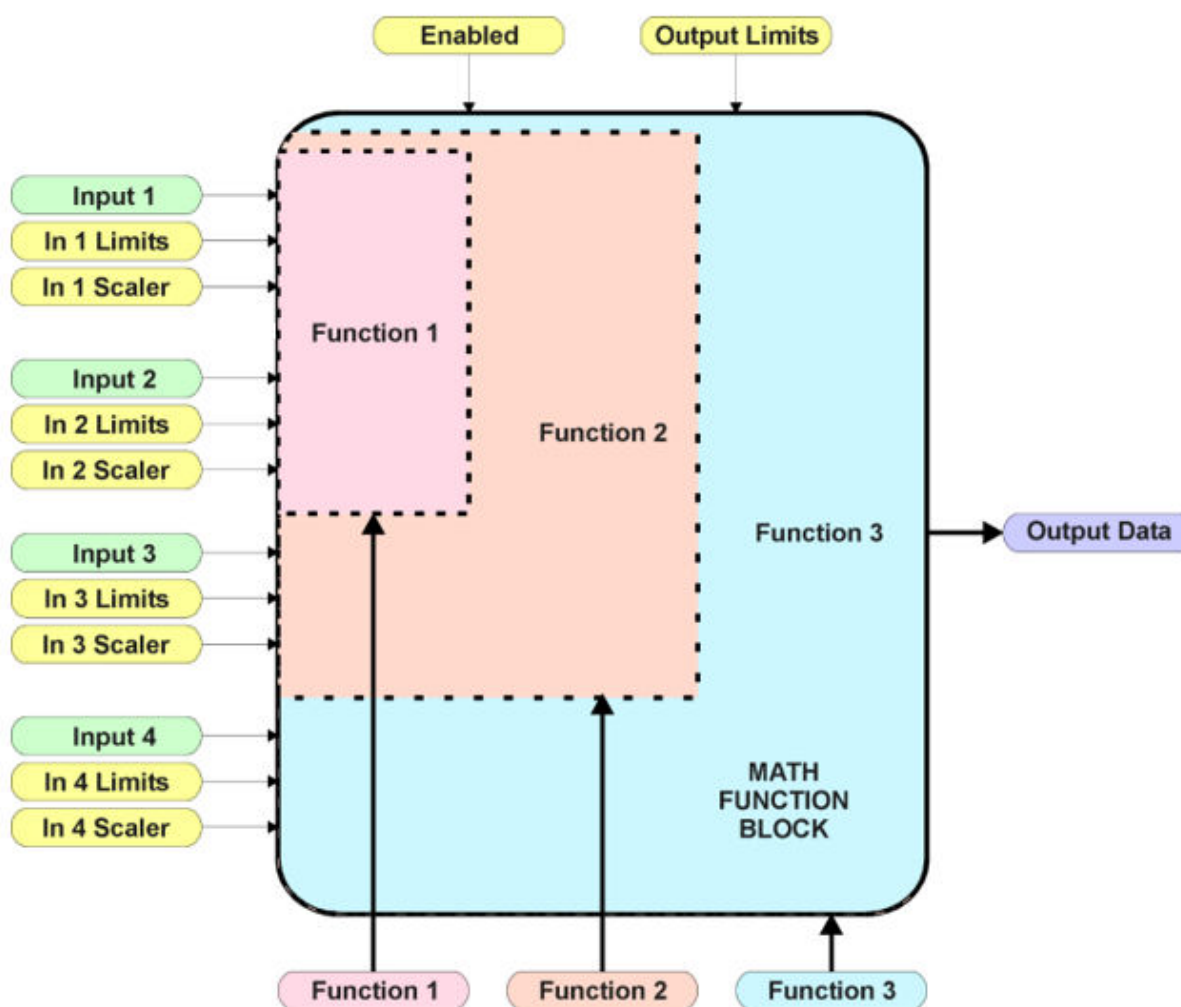
## 1.7. Math Function Block



**Figure 12 – 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 7 in Section 1.4.2. 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 17.

| 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 17 – 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.

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

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.
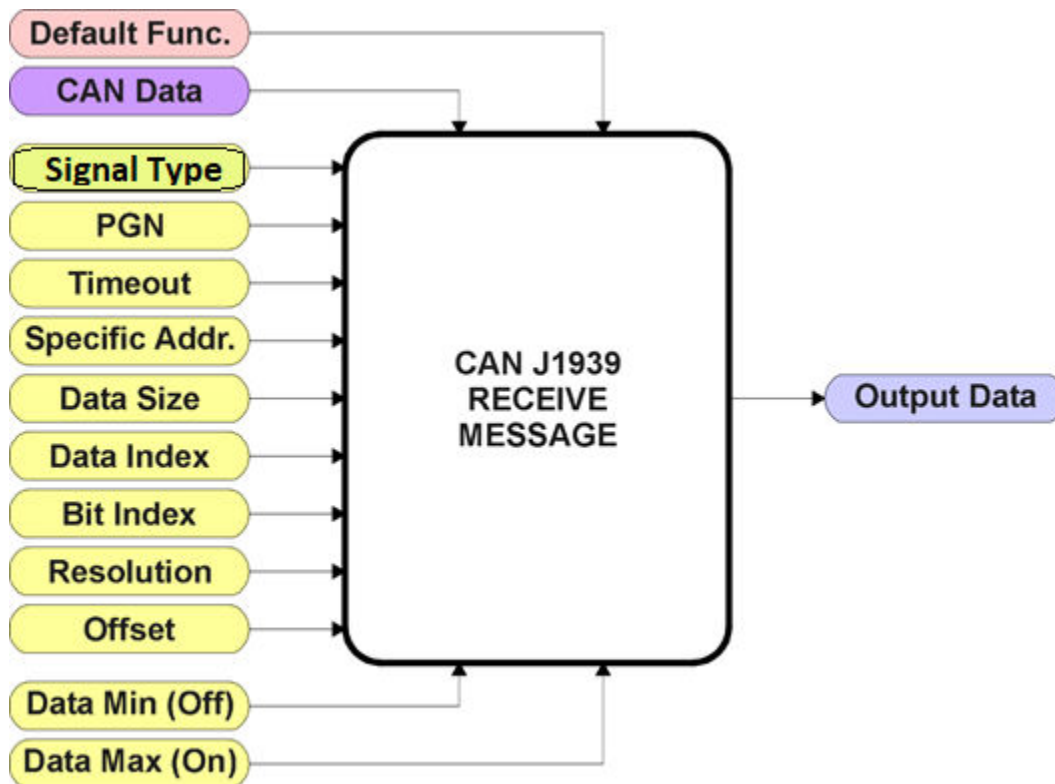
## 1.8. CAN Receive Function Block



**Figure 13 – 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. Analog Output).

The **"Received Signal Type"** is the most important setpoint associated with this function block, and it should be selected first. Changing it to '*Undefined*' will result in other setpoints being disabled. By default, only the first four 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 **"Enable Specific Address"** can be configured to TRUE then the unit can be setup to receive it from any ECU by setting the **"Specific Address that sends 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 10 unique CAN Receive Messages. By default, all messages are not used. The details are outlined in Section 3.3, and the default list is shown in Table 18 below.

| Block # | Default Receive Data |
|---------|---------------------|
| 1 | Message Not Used |
| 2 | Message Not Used |
| 3 | Message Not Used |
| 4 | Message Not Used |
| 5 | Message Not Used |
| 6 | Message Not Used |
| 7 | Message Not Used |
| 8 | Message Not Used |
| 9 | Message Not Used |
| 10 | Message Not Used |

**Table 18 – Default CAN Receive Messages**

## 1.9.    CAN Transmit Function Block

There are five CAN Transmit function blocks available to the user. The CAN Transmit function block is used to send any output from some other function block (i.e. Lookup Table 1) to the J1939 network.

"**Transmit Enabled**" setpoint will enable/disable the other setpoints in this block. Normally, to stop transmitting 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.

This Controller supports up to 5 unique CAN Transmit Messages, each CAN transmit message has four associated signals, all of which can be programmed to send any available data to the CAN network. "**Transmit Message Length**" can be used to set the length of a CAN message.

The **"Signal X Data Type"** is the most important setpoint associated with each single signal, and it should be selected first. Changing it to 'Discrete' or 'Continuous' will enable the other related setpoints, otherwise they will be disabled. **"Signal X Data Source"** setpoint together with **"Signal X Data Number"** setpoint define the signal source of the message. The control source options are same as the output drive which are listed in Table 4. Setting **"Signal X Data Source"** to '*Control Not Used'* disables the signal.

The **"Signal X Data Size"** setpoint selects the data size from 1-bit to 4-bytes. **"Signal X Data Index in Array (LSB)"** determines in which of 8 bytes of the CAN message LSB of the `signal is located`. Similarly, **"Signal X Bit Index in Byte (LSB)"** determines in which of 8 bits of a byte the LSB is located. These setpoints are freely configurable,

thus it is the User's responsibility to ensure that signals do not overlap and mask each other. **"Signal X Data Resolution"** setpoint determines the scaling done on the signal data before it is sent to the bus. and **"Signal X Data Offset"** setpoint determines the value that is subtracted from the signal data before it is scaled. Offset and Resolution are interpreted in units of the selected source signal. These setpoints can all be use to map the data to any SPN supported by the J1939 standard.

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

By default, none of the 5 messages is pre-configured and the default list is shown in Table 19 below.

| Block # | Default Transmit Data | (PGN) |
|---------|----------------------|-------|
| 1 | Control Source Not Used | (0xFF00) |
| 2 | Control Source Not Used | (0xFF01) |
| 3 | Control Source Not Used | (0xFF02) |
| 4 | Control Source Not Used | (0xFF03) |
| 5 | Control Source Not Used | (0xFF04) |

**Table 19 – Default CAN Transmit Messages**

## 1.10. Diagnostic Function Block

There are several types of diagnostics supported by the 4ACIN-4AOUT Controller. It can detect/react to input under-range and over-range errors, a processor over-temperature, or lost communication events.
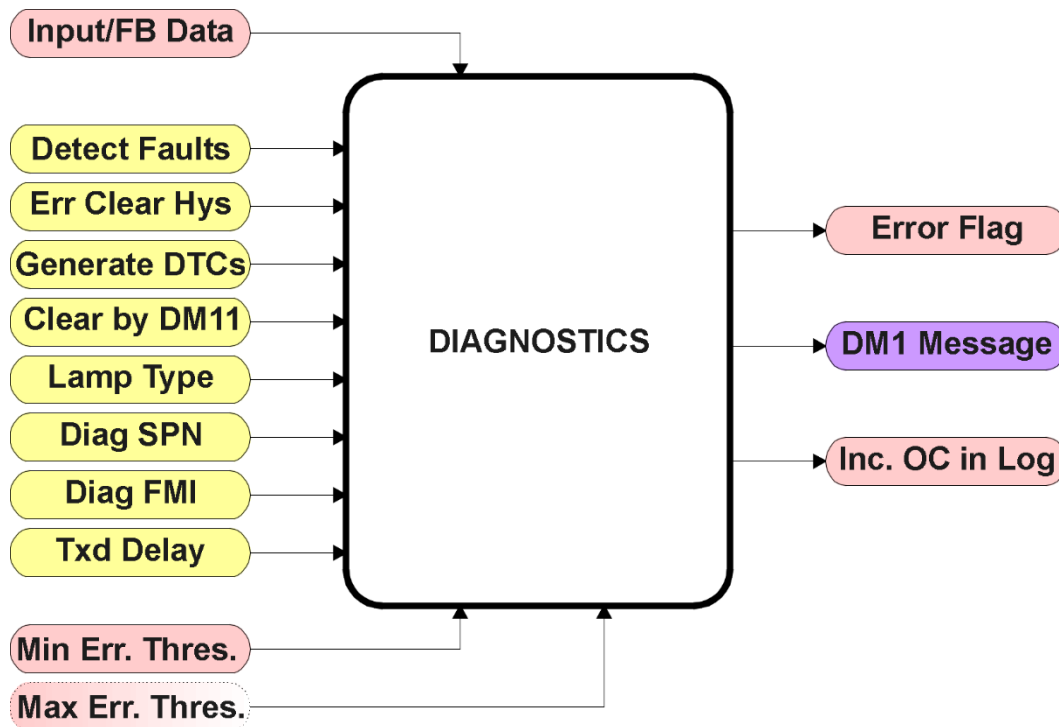


**Figure 14 – Diagnostics Function Block**

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 output 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 or temperature error, "**Fault Disables Outputs**" setpoint can be selected to disable all the outputs. By default, a power supply under/over voltage or over temperature condition will not shutoff the output.

In most cases, faults can be flagged as either a low or high occurrence. The min/max thresholds for all diagnostics supported by the 4ACIN-4AOUT 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 |
|---|---|---|
| AC Input Fault | **Minimum Error Range** | **Maximum Error Range** |
| Over Temperature | N/A | **Over Temperature Shutdown** |
| Lost Communication | N/A | **Received Message Timeout** (any) |

**Table 20 – Fault Detect Thresholds**

When applicable, a hysteresis setpoint is provided to prevent the rapid setting and clearing of the error flag when the measured 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 these function blocks 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 4ACIN-4AOUT 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 4ACIN-4AOUT 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 4ACIN-4AOUT 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).

> **ℹ** 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.

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. 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 use 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.

## 2. Installation Instructions

### 2.1 Dimensions and Pinout
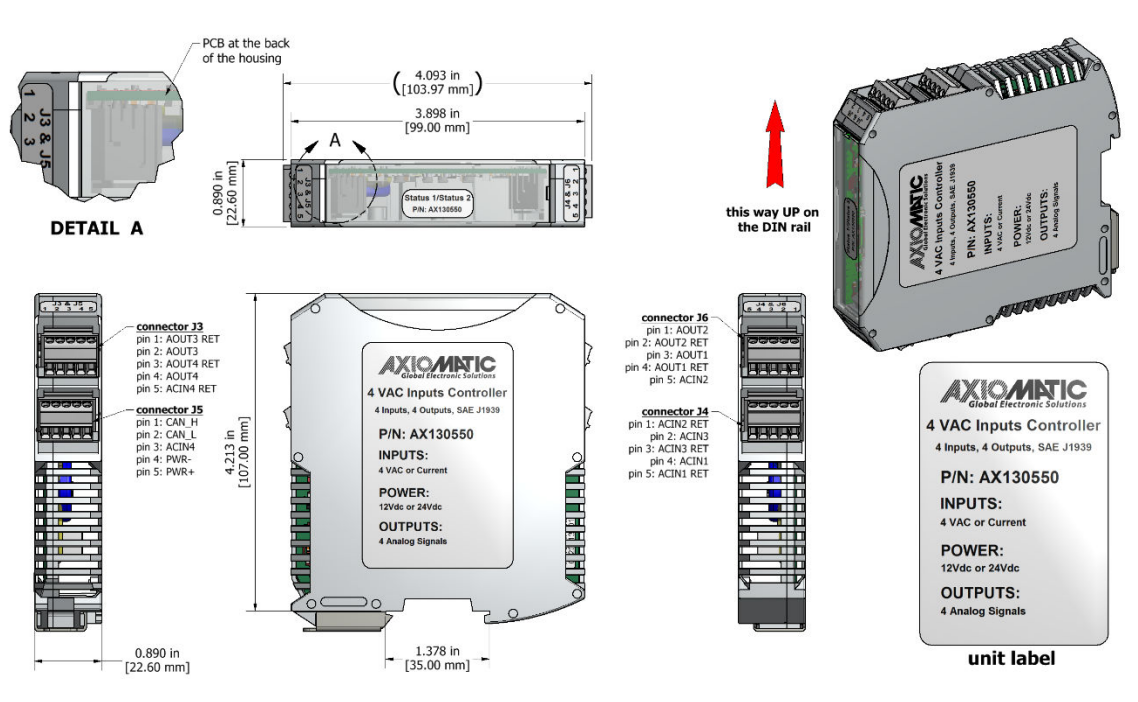


**Figure 15 – Dimensional Drawing**

| Outputs (J3) | | Power and CAN (J5) | | Inputs (J4) | | Inputs and Outputs (J6) | |
|---|---|---|---|---|---|---|---|
| PIN # | Function | PIN # | Function | PIN # | Function | PIN # | |
| 1 | Analog Output 3 return | 1 | CAN H | 1 | AC Voltage or Current Input 2 return | 1 | Analog Output 2 |
| 2 | Analog Output 3 | 2 | CAN L | 2 | AC Voltage or Current Input 3 | 2 | Analog Output 2_return |
| 3 | Analog Output 4 return | 3 | AC Voltage or Current Input 4 | 3 | AC Voltage or Current Input 3 return | 3 | Analog Output 1 |
| 4 | Analog Output 4 | 4 | BATT- | 4 | AC Voltage or Current Input 1 | 4 | Analog Output 1 return |
| 5 | AC Voltage or Current Input 4 return | 5 | BATT + | 5 | AC Voltage or Current Input 1_return | 5 | AC Voltage or Current Input 2 |

**Table 22  – Connector Pinout**

## 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 Receive, Default Output Control Data Message   65408 ($00FF80)

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**
a) DM1 – Active Diagnostic Trouble Codes                         65226 ($00FECA)
b) DM2 – Previously Active Diagnostic Trouble Codes       65227 ($00FECB)
c) DM3 – Diagnostic Data Clear/Reset for Previously Active DTCs 65228 ($00FECC)
d) DM11 - Diagnostic Data Clear/Reset for Active DTCs     65235 ($00FED3)
e) DM14 – Memory Access Request                                     55552 ($00D900)
f) DM15 – Memory Access Response                                   55296 ($00D800)
**g) 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 as shown in Table 23. 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 | 30, Axiomatic AX130550, 4 AC Inputs to 4 Analog Outputs Controller |
| **ECU Instance** | **0, First Instance** |
| Manufacture Code | 162, Axiomatic Technologies Corporation |
| Identity Number | Variable, uniquely assigned during factory programming for each ECU |

**Table 23 – Defaults for the J1939 NAME**

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 will allow 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:
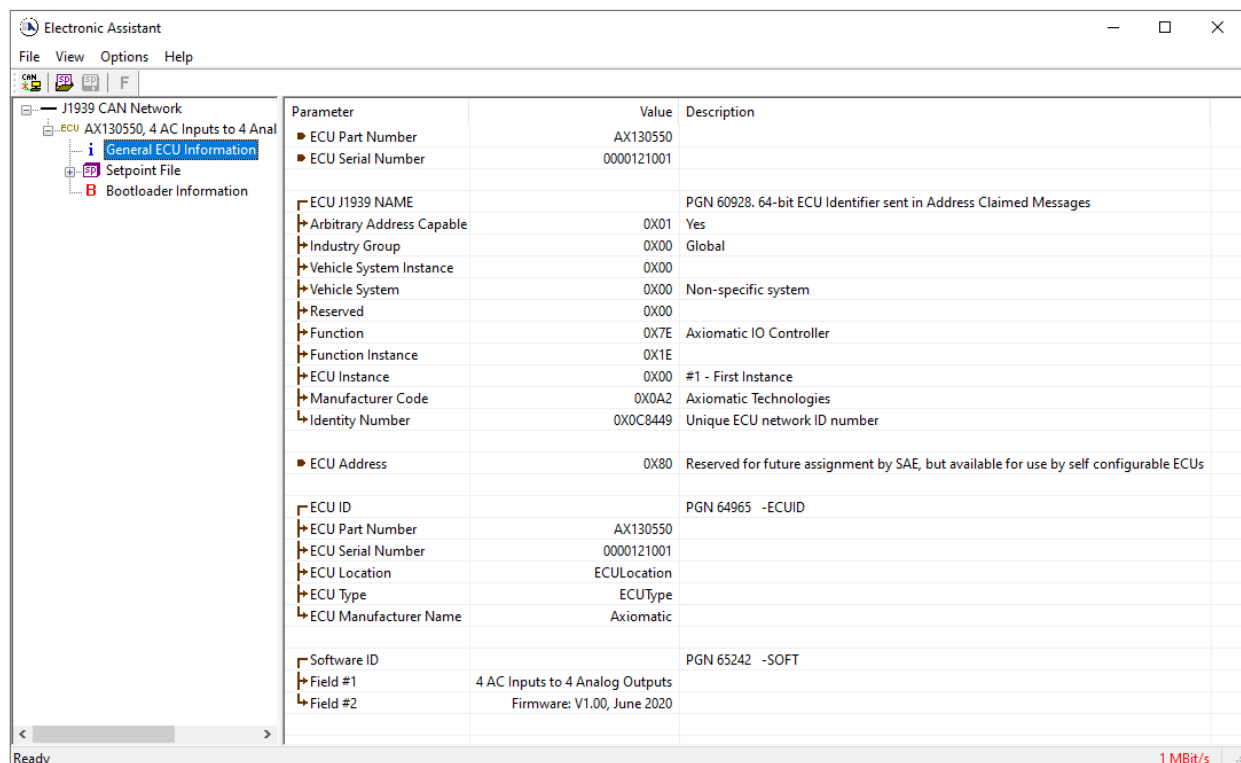


**Figure 16 – General ECU Information Caption**

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

## 3.3. 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 four CAN Receive Messages are disabled, as they are not part of the factory set logic. 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: | 0ms (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) (default) |
| Length (default): | 2 Bytes |
| Start Position: | First Byte |

# 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.
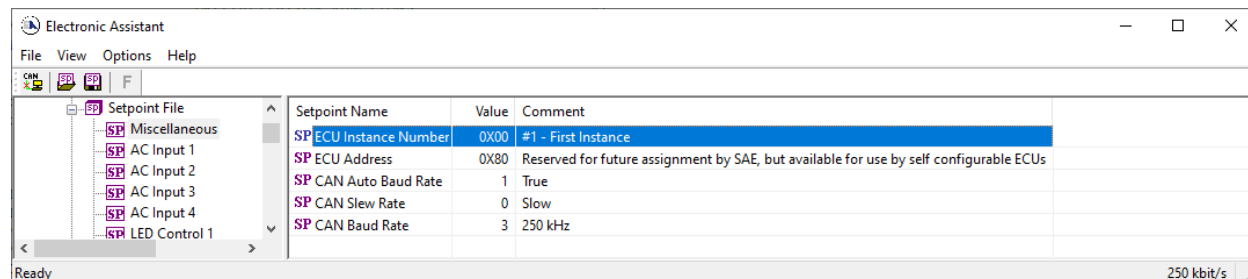


**Figure 17 - *Screen Capture of Default Miscellaneous Setpoints***

| 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 |
| CAN Auto Baud Rate | True/False | True | Baud rate automatic detection |
| CAN Slew Rate | Drop List | 0, Slow | Slew Rate selection |
| CAN Baud Rate | Drop List | 3, 250kHz | Baud rate manual  selection |

**Table 24 – Miscellaneous Setpoints Description**

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. AC Input Setpoints

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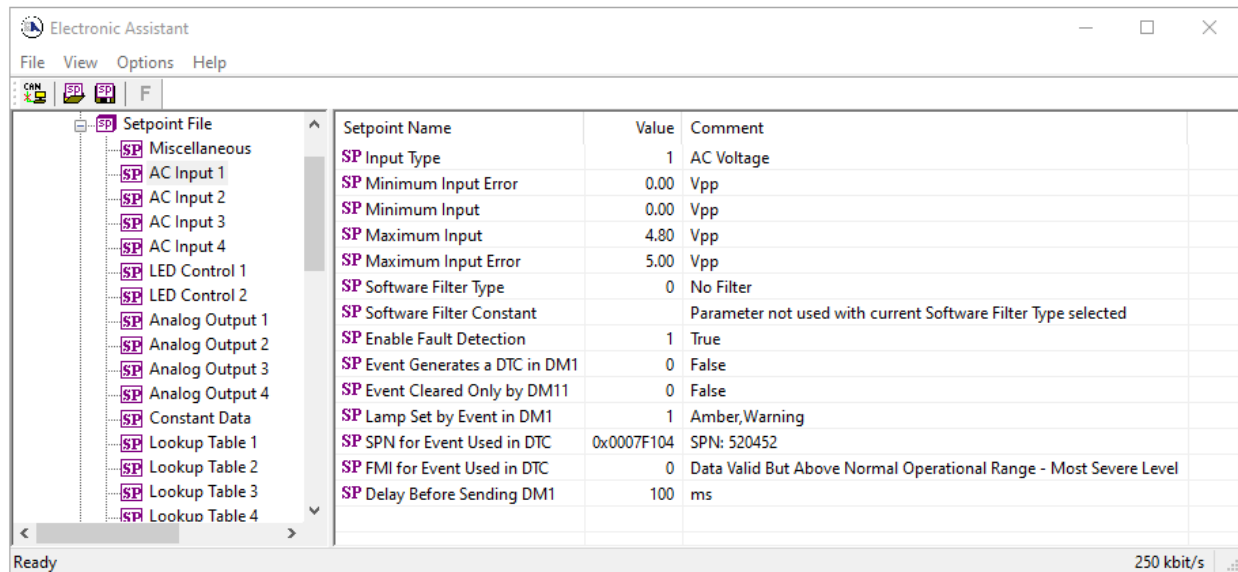


**Figure 18 - Screen Capture of Default AC Input Setpoints**

| Name | Range | Default | Notes |
|------|-------|---------|-------|
| Input Type | Drop List | 1, AC Voltage | See Table 1. |
| Minimum Input Error | Depends on Input Type | 0.00V | AC Voltage: [0…4.8] Analog Current: [0…192.0] |
| Minimum Input | Depends on Input Type | 0.00V | |
| Maximum Input | Depends on Input Type | 4.8V | AC Voltage: [0…4.8] Analog Current: [0…192.0] |
| Maximum Input Error | Depends on Input Type | 5V | AC Voltage: [0…5.0] Analog Current: [0…200.0] |
| Software Filter Type | Drop List | 0, No Filter | See Table 2. |
| Software Filter Constant | 0 to 60000 | 1000 | Not used by default |
| Enable Fault Detection | TRUE/FALSE | 1, True | See Section 4.11 |
| Event Generates a DTC in DM1 | TRUE/FALSE | 0, False | See Section 4.11 |
| Event Cleared Only by DM11 | TRUE/FALSE | 0, False | See Section 4.11 |
| Lamp Set by Event in DM1 | Drop List | 1, Amber, Warning | See Section 4.11 |
| SPN for Event Used in DTC | 1…524287 | 520452 ($7F104)  AC Input #1 - 520455 ($7F107)  AC Input #4 | See Section 4.11 |
| FMI for Event Used in DTC | Drop List | 0, Data Valid But Above Normal Operational Range – Most Severe Level | See Section 4.11 |
| Delay Before Sending DM1 | 0…60000 | 100 ms | See Section 4.11 |

**Table 25 - Default AC Input Setpoints**

## 4.3. LED Control

The LED Control function block is described in a Section 1.3. Please refer there for detailed information about how all these setpoints are used.
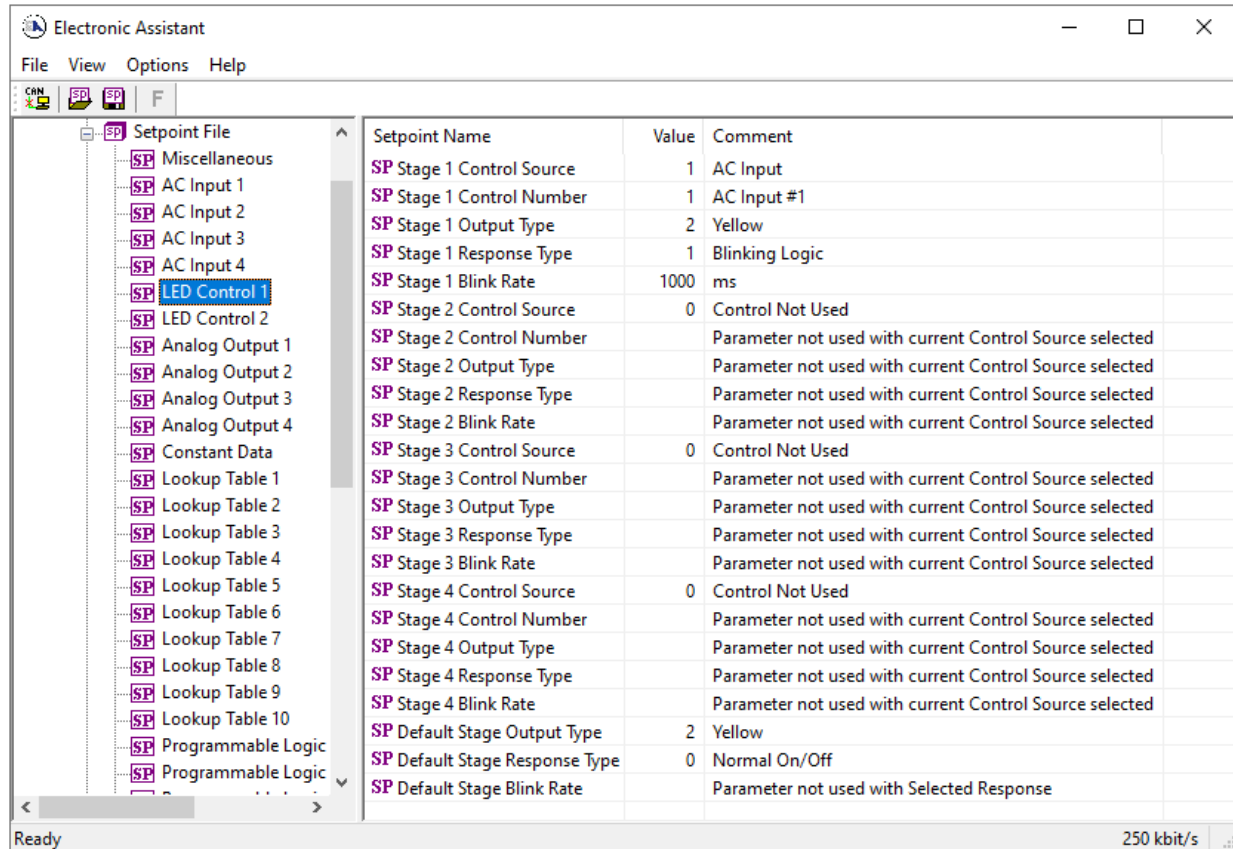


**Figure 19 - Screen Capture of Default LED Control Setpoints**

*Note: In the screen capture above the* ***"Stage 1 Control Source"*** *was changed to 'AC Input' as well as* ***"Stage 1 Response Type"*** *was changed to 'Blinking Logic' for demonstration purposes.*

| Name | Range | Default | Notes |
|------|-------|---------|-------|
| Stage 1...4 Control Source | Drop List | 0, Control Not Used | See Table 3. |
| Stage 1…4 Control Number | Depends on Control Source Type | N, where N is a number of stage | Not used in a default case |
| Stage 1…4 Output Type | Drop List | 2, Yellow | See Table 4 |
| Stage 1…4 Response Type | Drop List | 0, Normal On/Off | See Table 5. |
| Stage 1…4 Blink Rate | 0 to 60000 | 1000ms | Not used in a default case |
| Default Stage Output Type | Drop List | 2, Yellow | See Table 4 |
| Default Stage Response Type | Drop List | 0, Normal On/Off | See Table 5. |
| Default Stage Blink Rate | 0 to 60000 | 1000ms | Not used in a default case |

**Table 26 - Default LED Control Setpoints**

## 4.4. Analog Output Setpoints

The Analog Output function blocks are defined in Section 1.4. Please refer there for detailed information about how all these setpoints are used.
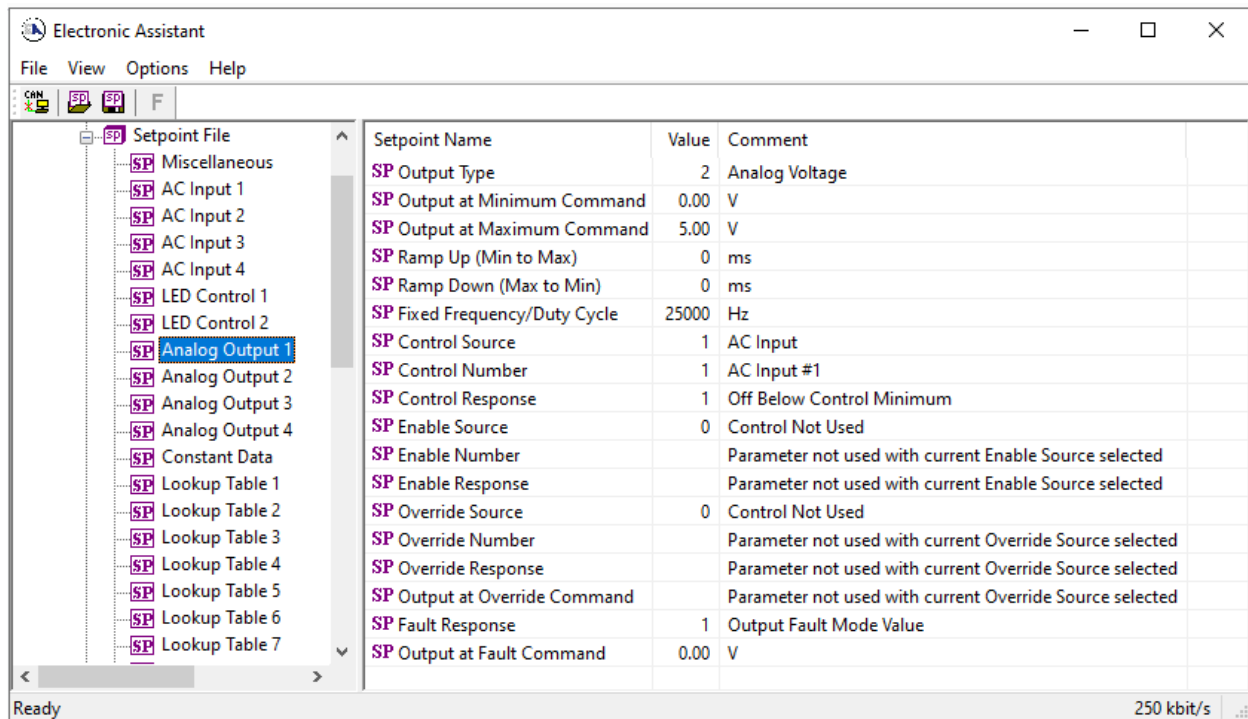


**Figure 20 - Screen Capture of Default Analog Output Setpoints**

| Name | Range | Default | Notes |
|---|---|---|---|
| Output Type | Drop List | 2, Voltage Output | See Table 6. |
| Output at Minimum Command | Depends on Output Type | 0.00V | Analog Voltage: [-10…10] |
| Output at Maximum Command | Depends on Output Type | 5V | Analog Current: [-20…20] |
| Ramp Up (Min to Max) | 0 to 10,000 | 0ms | Unit in milliseconds |
| Ramp Down (Max to Min) | 0 to 10,000 | 0ms | Unit in milliseconds |
| Control Source | Drop List | 1, AC Input | See Table 7. |
| Control Number | Per Source | 1, AC Input #1 (Analog Output 1) 2, AC Input #2 (Analog Output 2) 3, AC Input #3 (Analog Output 3) 4, AC Input #4 (Analog Output 4) | See Table 7. |
| Control Response | Drop List | 1, Off Below Control Minimum | See Table 8. |
| Enable Source | Drop List | 0, Control Not Used | See Table 7. |
| Enable Number | Per Source | N/A | See Table 7. |
| Enable Response | Drop List | N/A | See Table 8. |
| Override Source | Drop List | 0, Control Not Used | See Table 6. |

| Override Number | Per Source | N/A | See Table 6. |
| Override Response | Drop List | N/A | See Table 7. |
| Output at Override Command | Depends on Output Type | 1V | Same as minimum and maximum command |
| Fault Response | Drop List | 1, Output Fault Mode Value | See Table 8. |
| Output at Fault Command | Depends on Output Type | 0V | Same as minimum and maximum command |

**Table 27 - Screen Capture of Default Analog Output Setpoints**

## 4.5. Constant Data Setpoints

The Constant Data 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) Programmable Logic: Constant "**Table X = Condition Y, Argument 2**", where X and Y = 1 to 3
b) Math Function: Constant "**Math Input X**", where X = 1 to 4

All the constants are fully user programmable to any value. The default values (shown below) are arbitrary and should be configured by the user as appropriate for their application.
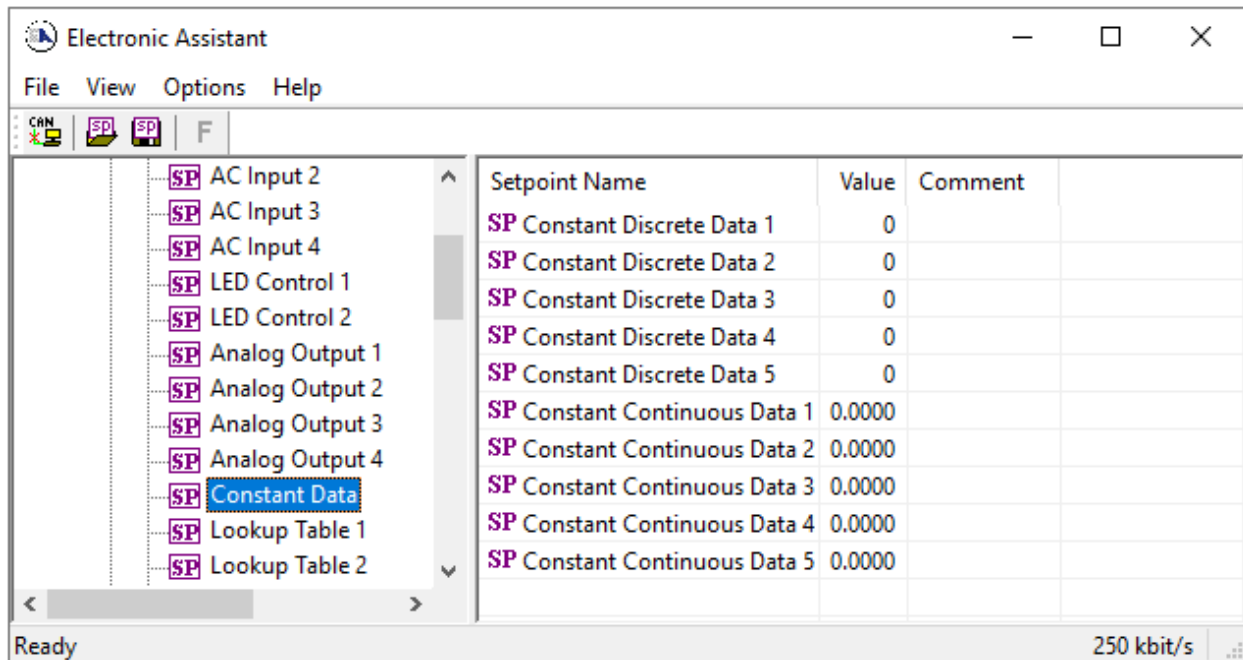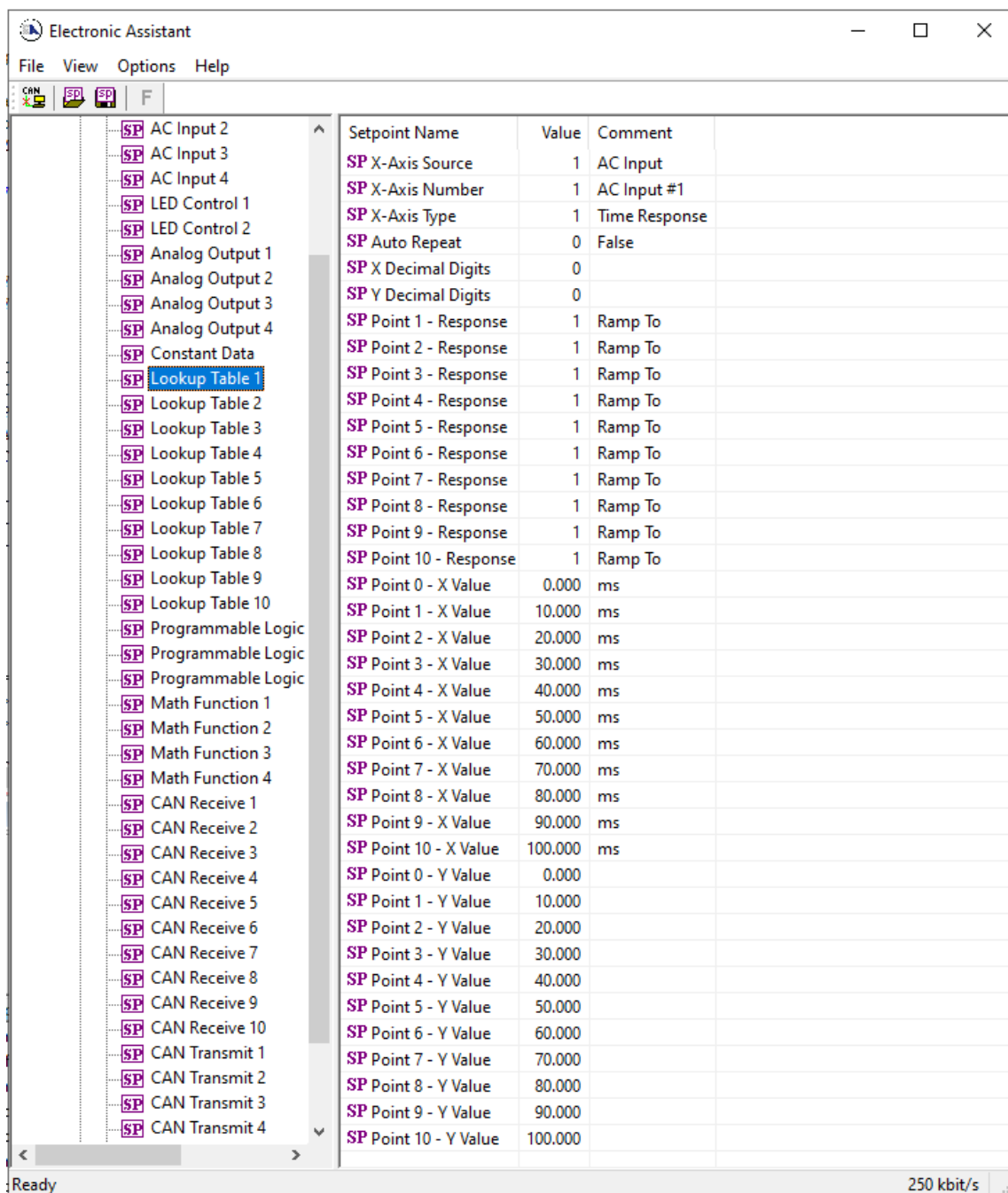


**Figure 21 - Screen Capture of Default Constant Data Setpoints**

## 4.6. 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 Table 7, 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.



**Figure 22 - 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.



**Figure 23 - 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 setpoints must 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.



**Figure 24 - 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 Output Minimum Range | $-10^4$ to $10^4$ | 0.0 | Converts calculation from a percentage value to the desired physical unit. |
| Math Output Maximum Range | $-10^4$ to $10^4$ | 100.0 | |
| Math Decimal Digits | 0 to 4 | 0 | Change the resolution of range values |
| Function X Input Y Source (X = 1 to 3, Y = A or B) | Drop List | 0, Control Source Not Used | See Table 7 |
| Function X Input Y Number | Per Source | 0 | See Table 7 |
| Function X Input Y Decimal Digits | 0 to 4 | 0 | Change the resolution for the range values |
| Function X Input Y Function Number | 0 to 3 | 0 | See Section 1.5 |
| Function X Input Y Minimum | $-10^4$ to $10^4$ | 0.0 | Converts input to a percentage before use in the calculation. |
| Function X Input Y Maximum | $-10^4$ to $10^4$ | 100.0 | |
| Function X Input Y Scaler | $-10^4$ to $10^4$ | 0.00 | See Section 1.5 |

**Table 28 – Default Math Function Setpoints**

## 4.9.    CAN Receive Setpoints

The CAN Receive 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 Receive 1 Setpoints*

*Note\*: If the "Received Signal Type" setpoint is Undefined, all the setpoints below it read "Parameter Not Used – Receive Message is Disabled"*

*Note\*\*: Each Block has different default values set to parameters. Refer to Section 3.3 for more details.*

| Name | Range | Default | Notes |
|---|---|---|---|
| Received Signal Type | Drop List | 2, Continuous (CAN Receive 1 to 4)<br>0, Undefined (CAN Receive 5 to 10) | See Note* above |
| Received PGN | 0 to 65535 | 65408 ($FF80) | See Section 3.3 for defaults |
| Enable Specific Address | False or True | 0, False | |
| Specific Address that sends PGN | 0 to 255 | 254 (0xFE, Null Addr) | Only configurable when specific address is enabled |
| Received Message Timeout | 0 to 60,000 ms | 0 ms | Expects all data at 100ms |
| Receive Data Size | Drop List | Different for each | 0 = Not Used (disabled)<br>1 = 1-Bit<br>2 = 2-Bits<br>3 = 4-Bits<br>4 = 1-Byte<br>5 = 2-Bytes<br>6 = 4-Bytes<br>See Section 3.3 for defaults |
| Receive Data Index in Array (LSB) | 0 to 7 Byte Position | Different for each | See Section 3.3 for defaults |
| Receive Bit Index in Byte (LSB) | 0 to 7 Bit Position | Different for each | Only used with Bit Data Types |
| Receive Data Resolution | $-10^5$ to $10^5$ | 1.0000000 | See Section 3.3 for defaults |
| Receive Data Offset | $-10^5$ to $10^5$ | 0.0000000 | See Section 3.3 for defaults |
| Received Data Min (Off Threshold) | $-10^5$ to $10^5$ | 0.0000000 | See Section 3.3 for defaults |
| Received Data Max (On Threshold) | $-10^5$ to $10^5$ | Different for each | See Section 3.3 for defaults |

**Table 29 - Default CAN Receive Setpoints**

## 4.10. CAN Transmit Setpoints

The CAN Transmit function block is defined in Section 1.9. Please refer there for detailed information about how all these setpoints are used.



**Figure 25 - Screen Capture of Default CAN Transmit 1 Setpoints**

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

| Name | Range | Default | Notes |
|---|---|---|---|
| Transmit PGN | 0 to 65535 | Different for each | See Section 1.9 for defaults |
| Transmit Enabled | True/False | 0, False | Enable the message |
| Transmit Repetition Rate | 0 to 65535 ms | 1000ms | 0ms disables transmit |
| Destination Address (for PDU1) | 0 to 255 | 254 (0xFE, Null Address) | Not used by default |
| Transmit Message Length | 1 to 8 | 8 | See Section 1.9 |
| Transmit Message Priority | 0 to 7 | 6 | Proprietary B Priority |
| Signal 1 Data Type | Drop List | 2, Continuous | Enable/disable the signal |
| Signal 1 Data Source | Drop List | AC Input | See Table 7 for defaults |
| Signal 1 Data Number | Per Source | 1 | See Table 7 for defaults |
| Signal 1 Data Size | Drop List | 5, 2-Bytes | 0 = Not Used (disabled)<br>1 = 1-Bit<br>2 = 2-Bits<br>3 = 4-Bits<br>4 = 1-Byte<br>5 = 2-Bytes<br>6 = 4-Bytes<br>See Section 1.9 for defaults |
| Signal 1 Data Index in Array (LSB) | 0 to 7 Byte Position | 0, 1$^{st}$ Byte Position | See Section 1.9 for defaults |
| Signal 1 Bit Index in Byte (LSB) | 0 to 7 Bit Position | Different for each | Only used with Bit Data Types |
| Signal 1 Data Resolution | -10$^5$ to 10$^5$ | 0.001 | See Section 1.9 for defaults |
| Signal 1 Data Offset | -10$^5$ to 10$^5$ | 0.0 | See Section 1.9 for defaults |
| Signal 2 Data Type | Drop List | 2, Continuous | Enable/disable the signal |
| Signal 2 Data Source | Drop List | AC Input | See Table 7 for defaults |
| Signal 2 Data Number | Per Source | 2 | See Table 7 for defaults |
| Signal 2 Data Size | Drop List | 5, 2-Bytes | 0 = Not Used (disabled)<br>1 = 1-Bit<br>2 = 2-Bits<br>3 = 4-Bits<br>4 = 1-Byte<br>5 = 2-Bytes<br>6 = 4-Bytes<br>See Section 3.3 for defaults |
| Signal 2 Data Index in Array (LSB) | 0 to 7 Byte Position | 2, 3$^{rd}$ Byte Position | See Section 1.9 for defaults |
| Signal 2 Bit Index in Byte (LSB) | 0 to 7 Bit Position | Different for each | Only used with Bit Data Types |
| Signal 2 Data Resolution | -10$^5$ to 10$^5$ | 0.001 | See Section 1.9 for defaults |
| Signal 2 Data Offset | -10$^5$ to 10$^5$ | 0.0 | See Section 1.9 for defaults |
| Signal 3 Data Type | Drop List | 2, Continuous | Enable/disable the signal |
| Signal 3 Data Source | Drop List | AC Input | See Table 7 for defaults |
| Signal 3 Data Number | Per Source | Different for each | See Table 7 for defaults |
| Signal 3 Data Size | Drop List | 5, 2-Bytes | 0 = Not Used (disabled) |

| | | | 1 = 1-Bit<br>2 = 2-Bits<br>3 = 4-Bits<br>4 = 1-Byte<br>5 = 2-Bytes<br>6 = 4-Bytes<br>See Section 1.9 for defaults |
|---|---|---|---|
| Signal 3 Data Index in Array (LSB) | 0 to 7 Byte Position | 4, 5th Byte Position | See Section 1.9 for defaults |
| Signal 3 Bit Index in Byte (LSB) | 0 to 7 Bit Position | Different for each | Only used with Bit Data Types |
| Signal 3 Data Resolution | $-10^5$ to $10^5$ | 0.001 | See Section 1.9 for defaults |
| Signal 3 Data Offset | $-10^5$ to $10^5$ | 0.0 | See Section 1.9 for defaults |
| Signal 4 Data Type | Drop List | 2, Continuous | Enable/disable the signal |
| Signal 4 Data Source | Drop List | AC Input | See Table 4 for defaults |
| Signal 4 Data Number | Per Source | Different for each | See Table 4 for defaults |
| Signal 4 Data Size | Drop List | 5, 2-Bytes | 0 = Not Used (disabled)<br>1 = 1-Bit<br>2 = 2-Bits<br>3 = 4-Bits<br>4 = 1-Byte<br>5 = 2-Bytes<br>6 = 4-Bytes<br>See Section 1.9 for defaults |
| Signal 4 Data Index in Array (LSB) | 0 to 7 Byte Position | 6, 7th Byte Position | See Section 1.7 for defaults |
| Signal 4 Bit Index in Byte (LSB) | 0 to 7 Bit Position | Different for each | Only used with Bit Data Types |
| Signal 4 Data Resolution | $-10^5$ to $10^5$ | 0.001 | See Section 1.9 for defaults |
| Signal 4 Data Offset | $-10^5$ to $10^5$ | 0.0 | See Section 1.9 for defaults |

**Table 30 - Default CAN Transmit Setpoints**

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.

## 4.11. Diagnostics Setpoints

Here, the diagnostics are AC Input Fault, Over Temperature and Lost Communication. The Diagnostic function block is defined in Section 1.10. Please refer there for detailed information about how all these setpoints are used.
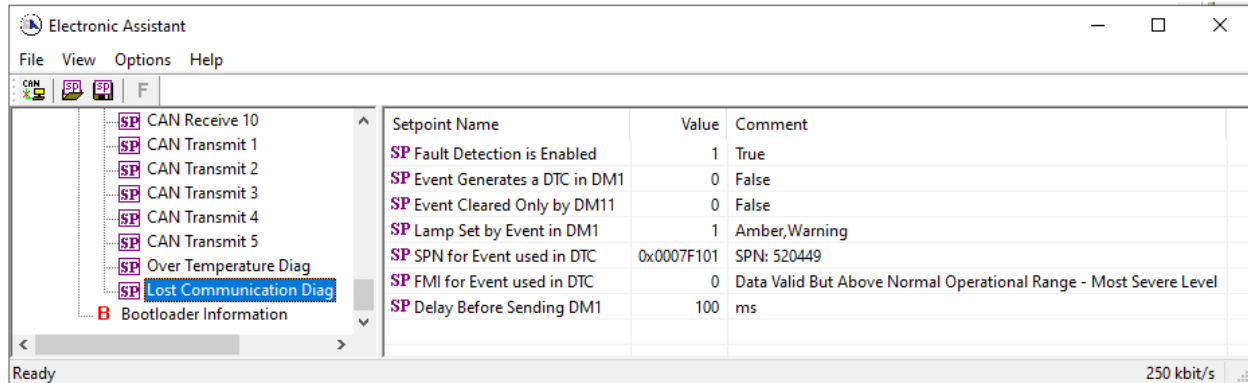


**Figure 26 -** *Screen Capture of Lost Communication Diag Setpoints*

| Name | Range | Default | | Notes |
|---|---|---|---|---|
| Fault Detection is Enabled | False or True | True | | See Section 1.10 |
| Input Under-Range Error * | 0V to Over V | 0.00 V | | |
| Input Over-Range Error * | Under V to 5 V | 5.00 V | | |
| Over Temperature Shutdown ** | 0 to 125 DegC | 75.00 Deg C | | |
| Hysteresis to Clear Fault | 0.01 to 250.00 | N/A 2.00 DegC N/A | Input Temp Comm | See Section 1.10 |
| Power Fault Disables Outputs ** | False or True | False | | |
| Event Generates a DTC in DM1 | False or True | False | | See Section 1.10 |
| Event Cleared Only by DM11 | False or True | False | | See Section 1.10 |
| Lamp Set by Event in DM1 | Drop List | 1, Amber, Warning | | See Section 1.10 |
| SPN for Event used in DTC | 1 to 524287 | 520448 ($7F100)    Over Temperature<br>520449 ($7F101)    Lost Communication<br>520452 ($7F104)    AC Input #1<br>520453 ($7F105)    AC Input #2<br>520454 ($7F106)    AC Input #3<br>520455 ($7F107)    AC Input #4 | | See Section 1.10 and Section 3.1 |
| FMI for Event used in DTC | Drop List | 0, Data Above Normal–Most Sever | | See Section 1.10 |
| Delay Before Sending DM1 | 0 to 60,000 ms | 100 ms | | See Section 1.10 |

**Table 31 – Diagnostic Setpoints**

*  Only used with AC Input Diag group
** Only used with Over Temperature Diag group

## 5. REFLASHING OVER CAN WITH THE AXIOMATIC EA BOOTLOADER

The AX130550 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.
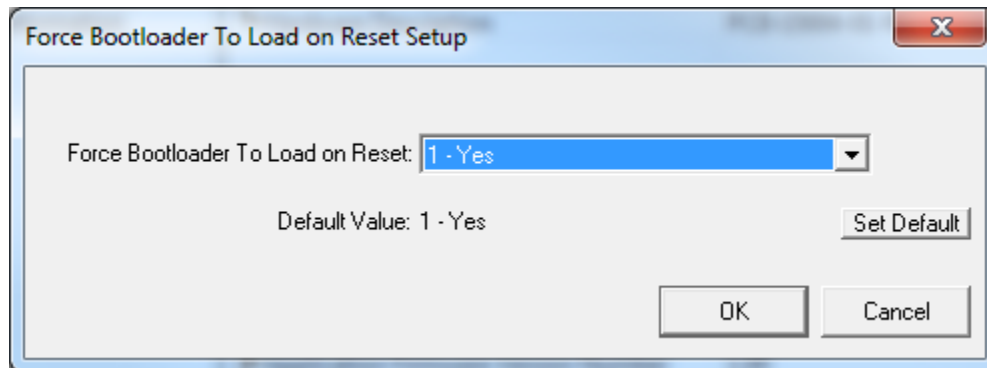
1. When the Axiomatic EA first connects to the ECU, the **Bootloader Information** section will display the following information.

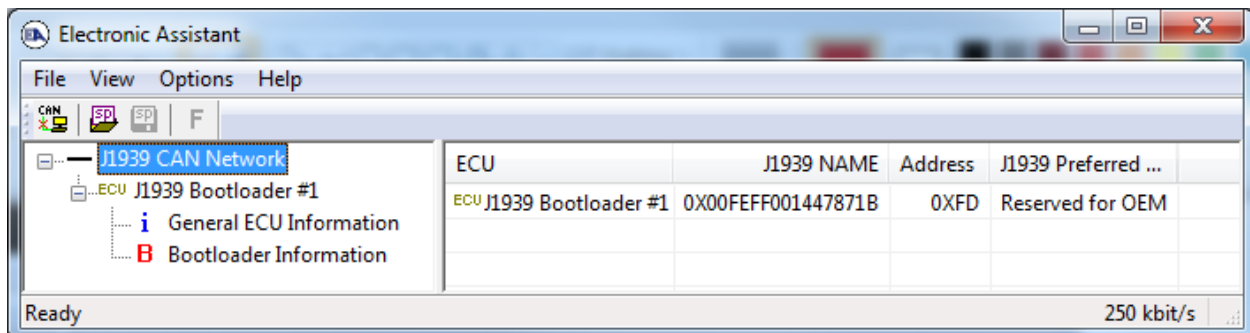2. To use the bootloader to upgrade the firmware running on the ECU, change the variable **"Force Bootloader To Load on Reset**" to *Yes*.
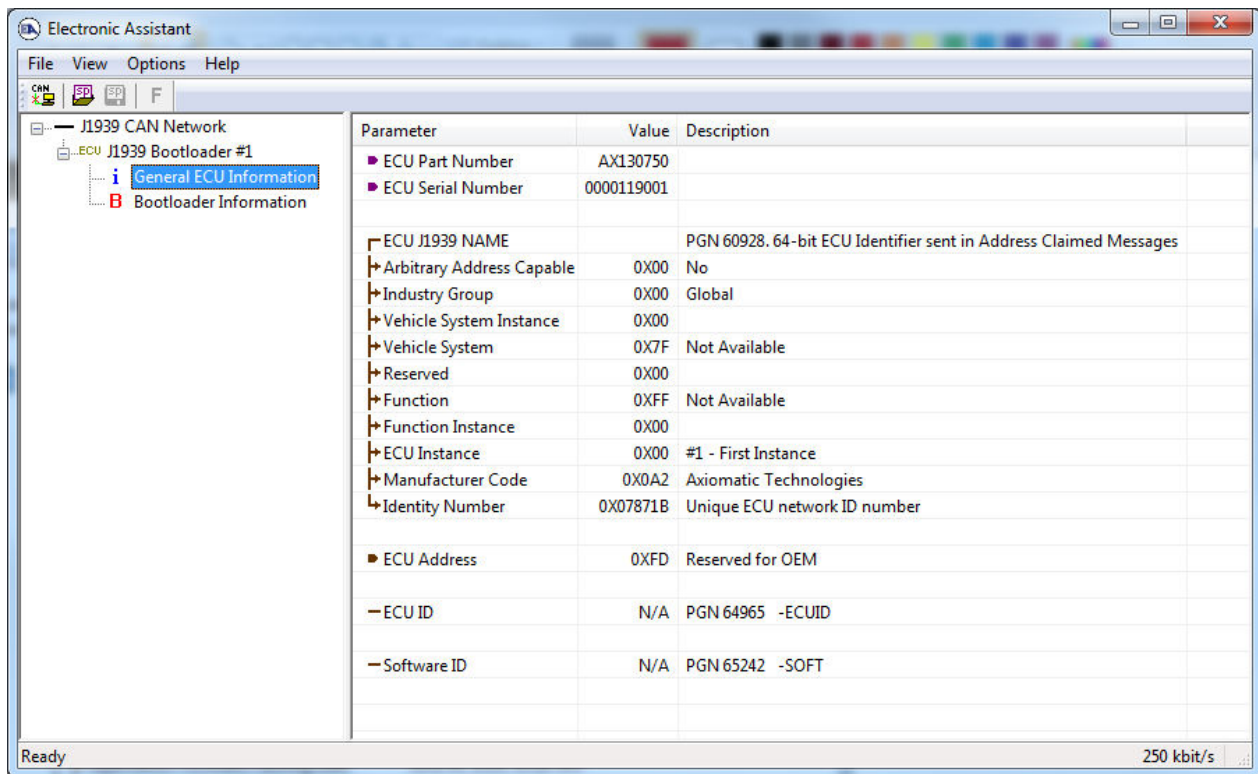


3. When the prompt box asks if you want to reset the ECU, select *Yes*.
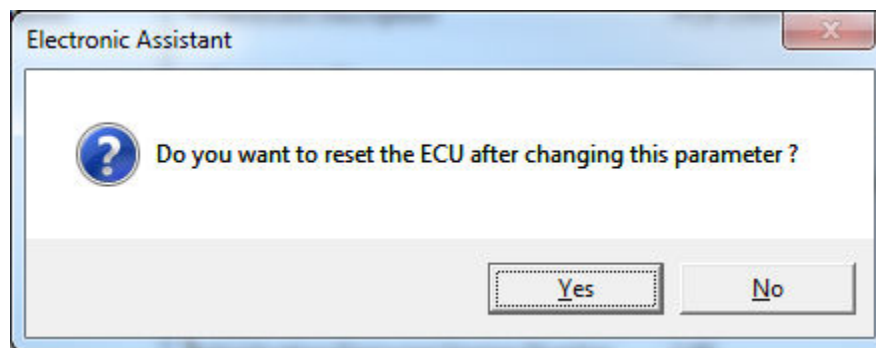


4. Upon reset, the ECU will no longer show up on the J1939 network as an AX130550 but rather as **J1939 Bootloader #1**.

The screenshot shows the Electronic Assistant software window with the following information:

| Parameter | Value | Description |
|---|---|---|
| ● ECU Part Number | AX130750 | |
| ● 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 | 0X07871B | 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 |

*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 AX130550 firmware, but in this case the <u>F</u>lashing feature has been enabled.
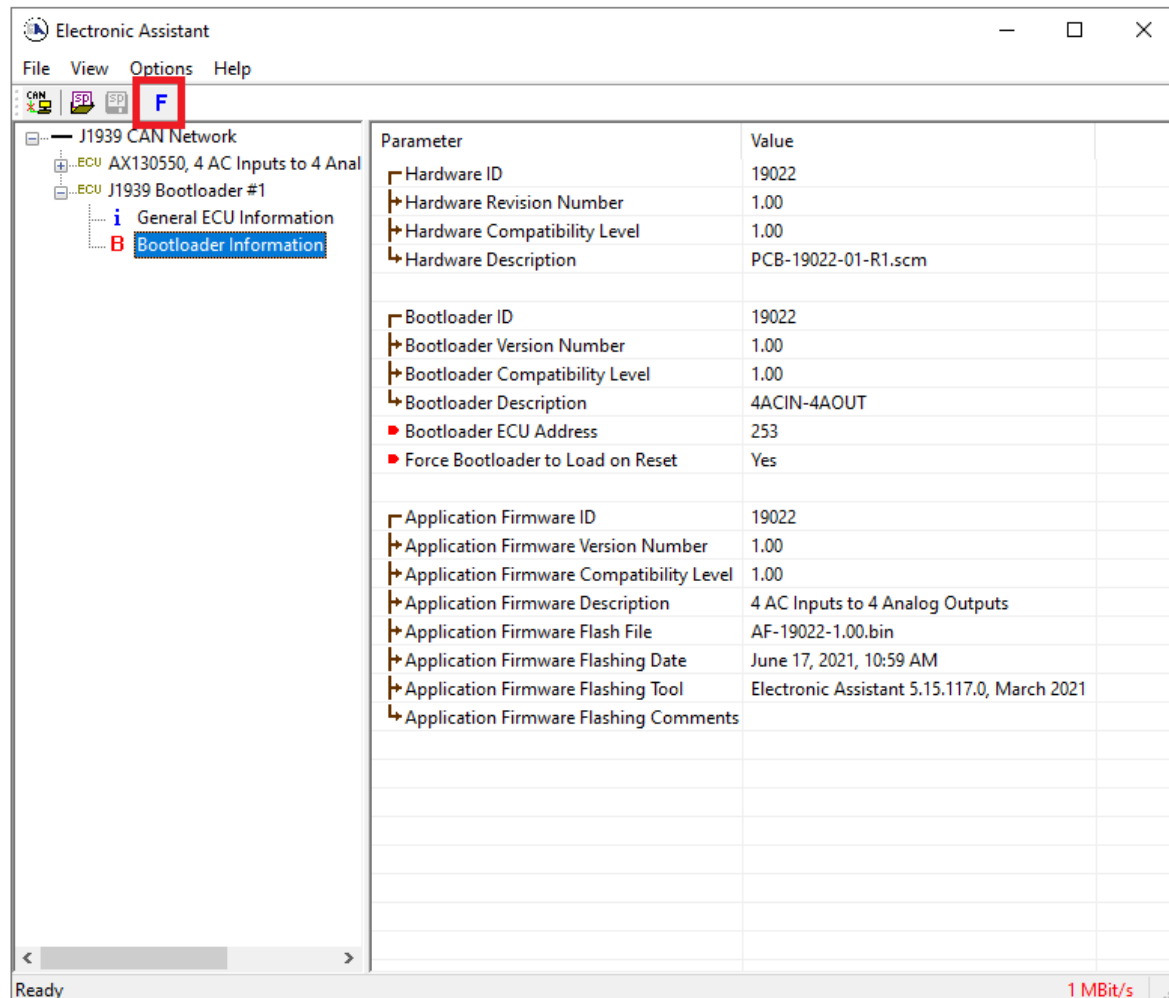


6. Select the <u>F</u>lashing button and navigate to where you had saved the **AF-19022_x.yy.bin** file sent from Axiomatic. (Note: only binary (.bin) files can be flashed using the Axiomatic EA tool)

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/time-stamp the file, as this is done automatically by the Axiomatic EA tool when you upload the new firmware.
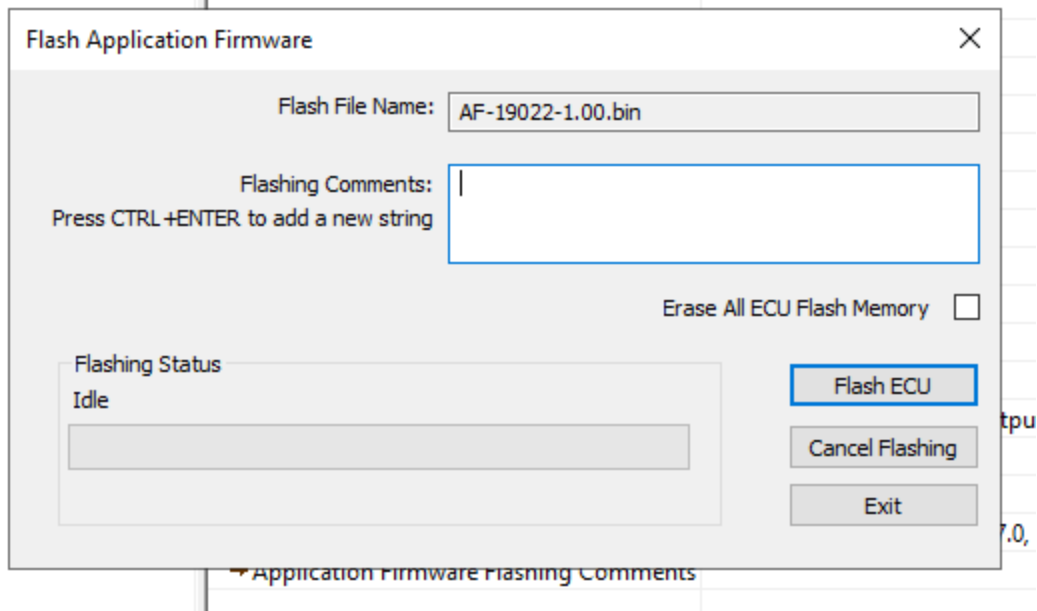
> ⚠ **WARNING:** Do not check the "Erase All ECU Flash Memory" box unless instructed to do so by your Axiomatic contact. Selecting this will erased ALL data stored in non-volatile flash, including the calibration done by Axiomatic during factory testing. It will also erase any configuration of the setpoints 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.
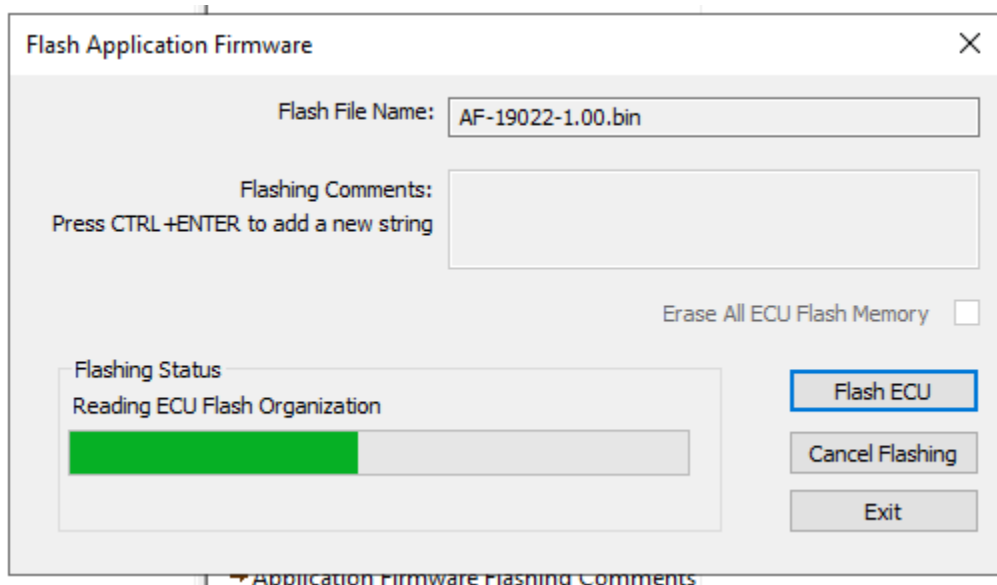
9. Once the firmware has finished uploading, a message will popup indicating the successful operation. If you select to reset the ECU, the new version of the AX130550 application will start running, and the ECU will be identified as such by the Axiomatic EA. Otherwise, the next time the ECU is power-cycled, the AX130550 application will run rather than the bootloader function.
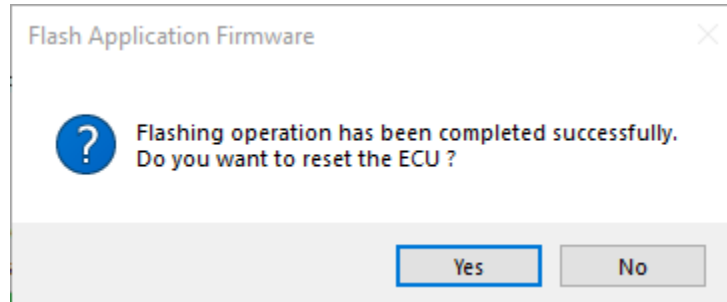


**Figure 27 – 35 – Firmware Installation Instructions**

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. VERSION HISTORY

| Version | Date | Author | Modifications |
|---------|------|--------|---------------|
| V1 | June 22nd, 2021 | Dmytro Tsebrii | Initial Draft |
| - | July 7, 2021 | Amanda Wilkins | Marketing Review |
| V1.01 | July 27, 2023 | Kiril Mojsov | Legacy Updates |

## APPENDIX - 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 Specifications

| | |
|---|---|
| Power Supply Input | 12 Vdc or 24 Vdc nominal<br>9…36 Vdc power supply range<br>Shutdown voltage is 7.0 Vdc. |
| Protection | Surge and transient protection<br>Reverse polarity protection<br>Overvoltage protection is up to 40 V. |

### Input Specifications

| | |
|---|---|
| Inputs | 4 AC Inputs, user selectable as:<br>• 0-5VAC peak<br>• 0-200 mA peak<br>Frequency range: 20-300Hz |
| Input Load | Voltage Input: 250 kΩ<br>Current Input: 10 Ω |
| Input Ground | 4 provided |
| Protection | All inputs are protected against short to GND and shorts to Nominal Vps (36Vdc). |
| Input Accuracy and Resolution | <table><tr><td>Input Type</td><td>Input Range</td><td>Accuracy</td><td>Resolution</td></tr><tr><td>Voltage</td><td>0-5V</td><td>+/-2%</td><td>1 mV</td></tr><tr><td>Current</td><td>0-200mA</td><td>+/-2%</td><td>24 µA</td></tr></table> |

### Output Specifications

| | |
|---|---|
| Outputs | 4 Analog Signal Outputs<br>Voltage or Current<br>The outputs are user selectable as follows. Refer to Table 1.0.<br><br>**Table 1.0: Programmable Outputs**<br><table><tr><td>Analog Voltage or Current Outputs:</td><td>Voltage Output:<br>0-5 Vdc, 0-10 Vdc, +/- 5Vdc or +/- 10Vdc<br>Maximum load is 50 mA.<br><br>Current Output:<br>0-20 mA or 4-20 mA<br>Maximum load resistance is < 500 Ohms.<br>Compliance Voltage is 10V.</td></tr></table> |
| Output Accuracy and Resolution | <table><tr><td>Output Type</td><td>Output Range</td><td>Output Accuracy</td><td>Output Resolution</td></tr><tr><td>Voltage</td><td>0-5V</td><td>+/- 0.5%</td><td>1 mV</td></tr><tr><td></td><td>0-10V</td><td>+/- 0.5%</td><td>1 mV</td></tr><tr><td></td><td>+/- 5Vdc</td><td>+/- 0.5%</td><td>1 mV</td></tr><tr><td></td><td>+/- 10Vdc</td><td>+/- 0.5%</td><td>1 mV</td></tr><tr><td>Current</td><td>0(4)-20mA</td><td>+/- 0.5%</td><td>0.5 µA</td></tr></table> |
| Output Grounds | All 4 output grounds are connected together internally. |
| Protection for Output Terminals | Fully protected against short circuit to output ground or +Vcc.<br>Unit will fail safe in the case of a short circuit condition, self-recovering when the short is removed. |

### General Specifications

| | |
|---|---|
| Microprocessor | STM32F405RGT7<br>32-bit, 1MByte flash memory |
| Isolation | 1-way Isolation (Power)<br>850 Vrms |
| Typical Quiescent Current | 88 mA@ 12Vdc; 44 mA @ 24Vdc typical |
| Response Time | 1ms |
| LED Indicators | 2 bicolour LED's (Red and Green)<br>Heartbeat, input fault and CAN fault indication |
| CAN Communications | 1 Isolated CAN port (SAE J1939) Auto baud rate detection<br>(CANopen® model is AX130551.) |

| Control Logic | Standard embedded software is provided and is configurable using the Axiomatic Electronic Assistant (EA). Any of the outputs can be configured to use any of the inputs either as a control signal or an enable signal as well as use the CAN network data. The user can configure the control logic using the following Function Blocks.<br><br>For more details on control logic, refer to the User Manual. |
|---|---|
| User Interface | The Axiomatic Electronic Assistant KIT, P/Ns: **AX070502** or **AX070506K** |
| Software Reflashing | The Axiomatic Electronic Assistant KIT, P/Ns: **AX070502** or **AX070506K** |
| Operating Conditions | -40 to 85 °C (-40 to 185 °F) |
| Storage Temperature | -55 to 125 °C (-67 to 257°F) |
| Protection | IP20 |
| Weight | 0.25 lb. (0.113 kg) |
| Enclosure and Dimensions | Phoenix Contact: ME MAX 22,5 G 2-2 KMGY – 2713625 (vented)<br>Polyamide, UL94V0, cULus recognized, China RoHS<br>DIN rail TH 35-7.5<br><br>99 x 114.5 x 22.5 x 99 mm (L x H x W x D)<br>Refer to the Installation Instructions, Figure 15. |
| Electrical Connections | `4 sets of 5 Phoenix Contact MC 1,5/ 5-ST-3,5 GY7035 3.5 mm screw terminal`<br>Accepts 24-14 AWG wire.<br>Refer to Table 22 and Figure 15 for pin out. |
| Installation | DIN rail mount<br>TH 35-7.5 |
| Network Termination | It is necessary to terminate the network with external termination resistors. The resistors are 120 Ohm, 0.25W minimum, metal film or similar type. They should be placed between CAN_H and CAN_L terminals at both ends of the network. |

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

# 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