

Dual Stepper Drive, CAN J1939 with Additional I/O

USER MANUAL

P/N: AX100750

Table of Contents

LIST OF FIGURES.....	4
LIST OF TABLES.....	4
ACCRONYMS.....	5
1 OVERVIEW OF CONTROLLER	6
1.1 Description of Motor Drive.....	6
1.2 Motor Drive Function Blocks	8
1.2.1 Rotation Angle Per Step	9
1.2.2 Step Rate Data Unit	9
1.2.3 Motor Mode.....	9
1.2.4 Direction	11
1.2.5 Step Rate	11
1.2.6 Max. Phase Current.....	12
1.2.7 Number of Steps.....	12
1.2.8 Step Type.....	13
1.2.9 Percent Fast Decay	15
1.2.10 Acceleration Table	15
1.2.11 Put Drive to Sleep.....	17
1.2.12 Reset Motor Drive	17
1.2.13 E-Stop Motor Drive	18
1.2.14 Save Data to Defaults	18
1.2.15 PID Motor Control Function Block	19
1.3 Input Function Blocks.....	23
1.4 Encoder Input Function Block	27
1.4.1 Encoder Stationary Delay	28
1.5 Filtering Function Block	29
1.5.1 Calculation with no filter:.....	30
1.5.2 Calculation with the 3rd Order Low Pass filter:.....	30
1.6 Universal Output Function Block	31
1.7 PID Control Function Block	35
1.8 Lookup Table Function Block	36
1.9 Programmable Logic Function Block	37
1.10 Math Function Block.....	38
1.11 Set / Reset Latch Function Block.....	39
1.12 Diagnostic Function Blocks	40
1.13 CAN Receive Function Block	43
1.14 CAN Transmit Function Block.....	45

1.14.1	CAN Transmit Message Setpoints.....	45
1.14.2	CAN Transmit Signal Setpoints.....	45
1.15	Available Control/Input Sources	46
2	INSTALLATION INSTRUCTIONS	49
2.1	Dimensions and Pinout.....	49
3	OVERVIEW OF J1939 FEATURES	50
3.1	Introduction To Supported Messages	50
3.2	NAME, Address and Software ID	51
3.3	CAN Receive Message Defaults	53
3.4	CAN Transmit Message Defaults	58
4	ECU SETPOINTS ACCESSED WITH THE AXIOMATIC ELECTRONIC ASSISTANT	63
4.1	Voltage/Current Input Setpoints.....	63
4.2	Frequency/PWM Input Setpoints	64
4.3	Encoder Input Setpoints	65
4.4	Universal Output Setpoints	66
4.5	Stepper Drive Values Setpoints	68
4.6	Stepper Drive Controls Setpoints.....	71
4.7	Phase Current Filter Setpoints.....	72
4.8	Acceleration Table Setpoints.....	73
4.9	Global Parameters Setpoints	74
4.10	CAN Receive Setpoints	75
4.11	CAN Transmit Setpoints.....	76
4.12	Lookup Table Setpoints	77
4.13	Programmable Logic Setpoints.....	79
4.14	Math Function Block.....	80
4.15	Set-Reset Latch Block.....	81
4.16	PID Control Function Setpoints	82
4.17	Diagnostic Setpoints.....	84
4.18	CAN Network Setpoints	85
5	REFLASHING OVER CAN WITH THE AXIOMATIC EA BOOTLOADER.....	86
6	TECHNICAL SPECIFICATIONS:	91
6.1	Power Supply Specifications.....	91
6.2	Input Specifications.....	91
6.3	Output Specifications	92
6.4	General Specifications	92

LIST OF FIGURES

Figure 1. Dual Stepper Motor Drive Block Diagram	7
Figure 2. Stepper Drive Values Function Block	8
Figure 3. Stepper Drive Controls Function Block	8
Figure 4. Linear Slope Calculations	11
Figure 5. Full, Half and Quarter-Step Waveforms	13
Figure 6. Sixteenth-Step Waveforms	14
Figure 7. Example Acceleration Curve	16
Figure 8. PID Control Function Block	19
Figure 9. PID Control Algorithm	20
Figure 10. Input Function Block	23
Figure 11. Digital Input Function Block	26
Figure 12. Digital Input Debouncing	26
Figure 13: Incremental Decoder Signals (QA&QB) and resulting Direction and Step count	27
Figure 14. Software Data Filter Function Block	29
Figure 15. Hotshot Digital Profile	32
Figure 16. Function Block	40
Figure 17. Analog source to Digital input	48
Figure 18. Dimension Drawing	49
Figure 19. Connector Pinout	49

LIST OF TABLES

Table 1. Step Rate Data Unit Options	9
Table 2. Motor Mode Options	9
Table 3. Motor Direction Options	11
Table 4. Step Type Options	13
Table 5. Percent Fast Decay Options	15
Table 6. Acceleration Table Options	17
Table 7. PID Control Algorithm Options	19
Table 8. PID Logic Response Options	21
Table 9. PID Delta Check Response Options	22
Table 10. Analog Input Type Options	24
Table 11. Frequency Input Type Options	24
Table 12. Pullup/Pulldown Resistor Options	24
Table 13. Voltage Input Ranges	25
Table 14. Current Input Ranges	25
Table 15. Frequency Input Ranges	25
Table 16. PWM Input Ranges	25
Table 17. Debounce Time Options	27
Table 18. Digital Input Type versus Input State	27
Table 19. Input Type options for Encoder Inputs	27
Table 20. Software Filter Types	29
Table 21. Output Type Options for Proportional Output	31
Table 22. Digital Response Options	31
Table 23. High-Side/Low-Side Output Setting	32
Table 24. Delay Polarity	32
Table 25. Enable Response Options	33
Table 26. Override Response Options	34
Table 27. Fault Response Options	34
Table 28. PID Response Options	35
Table 29. X-Axis Type Options	36
Table 30. PointN – Response Options	36
Table 31. Table X – Condition Y Operator Options	37
Table 32. Table X – Conditions Logical Operator Options	37
Table 33. Math function X Operator Options	38
Table 34. Set-Reset Function block operation	39

Table 35. Diagnostic Blocks and Fault Detection Thresholds.....	41
Table 36. Low Fault FMI versus High Fault FMI	43
Table 37. Default CAN Receive Messages	44
Table 38. Programmable Inputs	46
Table 39. Input Sources and Numbers	47
Table 40. Input Setpoint Defaults Based on Type and Range.....	64

ACCRONYMS

ACK	Positive Acknowledgement (from SAE J1939 standard)
ADC	Analog-to-Digital Converter, 12-bit processor peripheral
AIN	Analog Input used to measure Voltage or Current signals
DIN	Digital Input used to measure active high or low signals
DM	Diagnostic Message (from SAE J1939 standard)
DTC	Diagnostic Trouble Code (from SAE J1939 standard)
EA	The Axiomatic Electronic Assistant (A Service Tool for Axiomatic ECUs)
ECU	Electronic Control Unit (from SAE J1939 standard)
FIN	Frequency Input used to measure Frequency, RPM or PWM signals
I/O	Inputs and Outputs
UOut	Universal Output
NAK	Negative Acknowledgement (from SAE J1939 standard)
PDU1	A format for messages that are to be sent to a destination address, either specific or global (from SAE J1939 standard)
PDU2	A format used to send information that has been labeled using the Group Extension technique and does not contain a destination address.
PGN	Parameter Group Number (from SAE J1939 standard)
PropA	Message that uses the Proprietary A PGN for peer-to-peer communication
PropB	Message that uses a Proprietary B PGN for broadcast communication
PWM	Pulse Width Modulation
RMS	Root Mean Squared
RPM	Rotations per Minute
SPN	Suspect Parameter Number (from SAE J1939 standard)

Note:

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

1 OVERVIEW OF CONTROLLER

The controller is internally organized as a set of function blocks, which can be individually configured and arbitrarily connected to achieve the required system functionality.

Measured input data can be sent to a SAE J1939 CAN Network as is or used in the 1 Encoder, 2 Universal Inputs, 2 Magnetic Pick-up Sensor Inputs controller function blocks for controlling 1 Universal Output and how the 2 motors are driven. The Windows-based Axiomatic Electronic Assistant (EA) is used to configure the controller via an USB-CAN (AX070501) device. Configurable properties, Axiomatic EA setpoints, are outlined in chapter 4. Setpoint configuration can be saved in a file which can be used to easily program the same configuration into another Controller.

1.1 Description of Motor Drive

The Stepper Motor Drive is designed to give the user full access to the 2A drive control chip A3979 via J1939 command messages. It allows for simple and accurate control of the levels, waveforms, timing, and phase-shifts of the current for a two-phase stepper motor.

The drive allows the user to set the default values for 2 stepper motors using the Axiomatic Electronic Assistant. It also allows the user to set the default targets and ranges for the run-time variables such as the step rate the motor is to run at, or the number of steps the motor is to take before holding.

A unique feature of the drive is that not only are the run-time variables accessible through the Axiomatic EA, but they can also be independently controlled by J1939 messages received while the unit is operational. This means that while the Axiomatic EA tool is used for initial setup and configuration, all the drive values can be changed as needed without it.

The default configuration for the drive is to be a purely reactive control based on the command data received from J1939 messages. However, it does have more sophisticated logic supported for more complex controls. In addition to the logic described in this user manual, other responses or function blocks could be added to the drive on request. For more information about this option, contact Application Engineering at support@axiomatic.com.

The “off-the-shelf” logic blocks supported by the drive, and selectable by the user, are:
Automatic PID control of the direction and step rate when the “**Motor Mode**” is selected to be controlled by the “**PID Control Function**” block.

Diagnostic detection, reaction, feedback and up to five acceleration/deceleration profiles that can be set up using the Axiomatic EA, then selected during operation via a CAN message.

Automatic resetting of the “Number of Steps” command so that the controller can simply send the same CAN message again to step the same number of steps.

The drive also has inputs that could be used as the command source for a run-time parameter, rather than a CAN message. Alternatively, the inputs could be used for feedback in a PID control, or as simple digital controls, for example to E-Stop the device. Lastly, they could independently be sent to the J1939 network as input data for another device on the CAN bus.

The Dual stepper drive is a highly programmable controller, allowing the user to configure it for their application. Its sophisticated control algorithms allow for open or closed loop drive of the motors. All I/O on the unit are inherently independent from one another but can be programmed to interact in many ways.

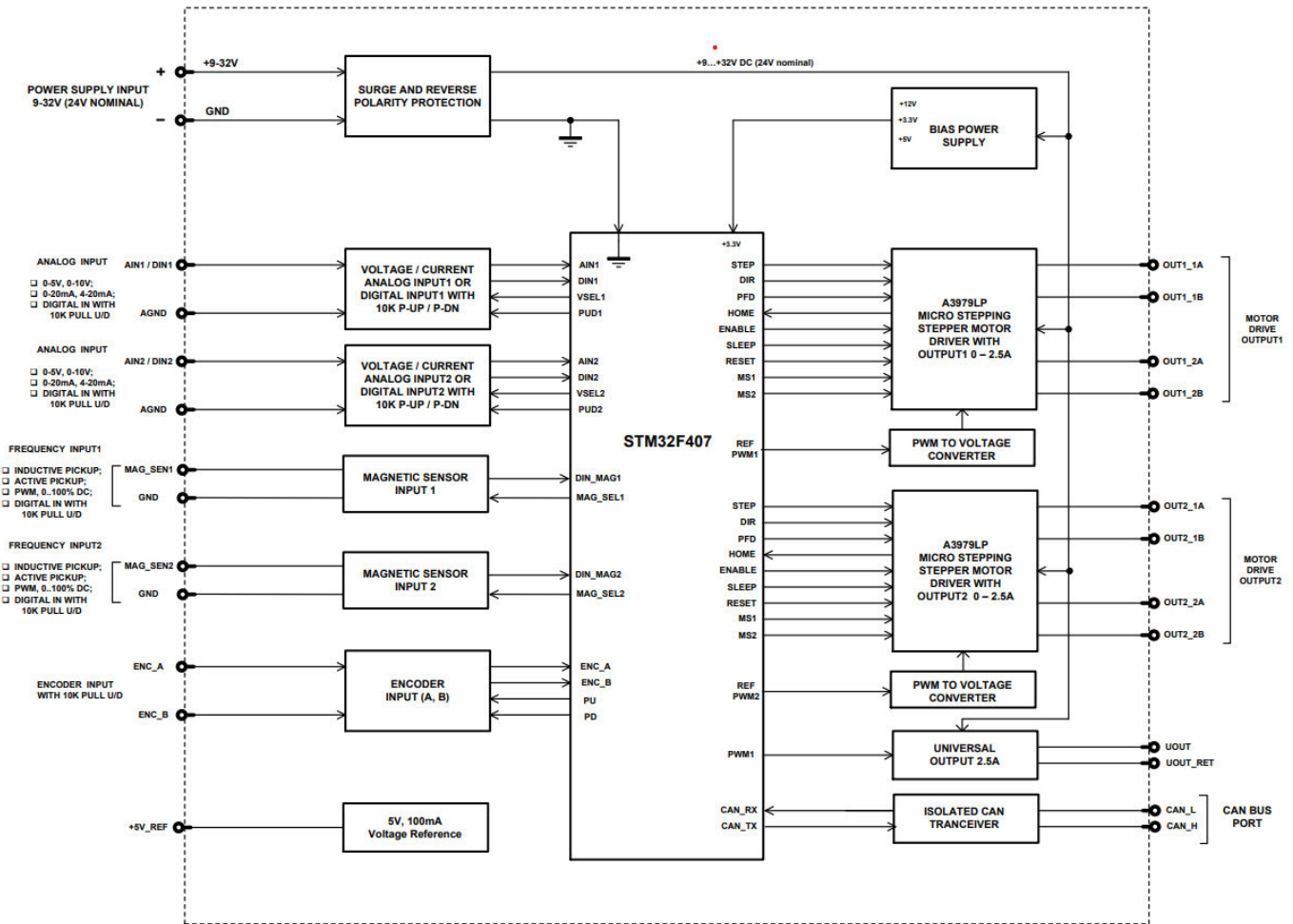


Figure 1. Dual Stepper Motor Drive Block Diagram

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

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

1.2 Motor Drive Function Blocks

There are two types of function blocks associated with the stepper drive, "Values" and "Controls". The "Values" function block contains the default drive setpoints, and the limits for the linear controls. The "Controls" function block tells the drive what it should use as the source of the command signal for each of the tasks that it supports.

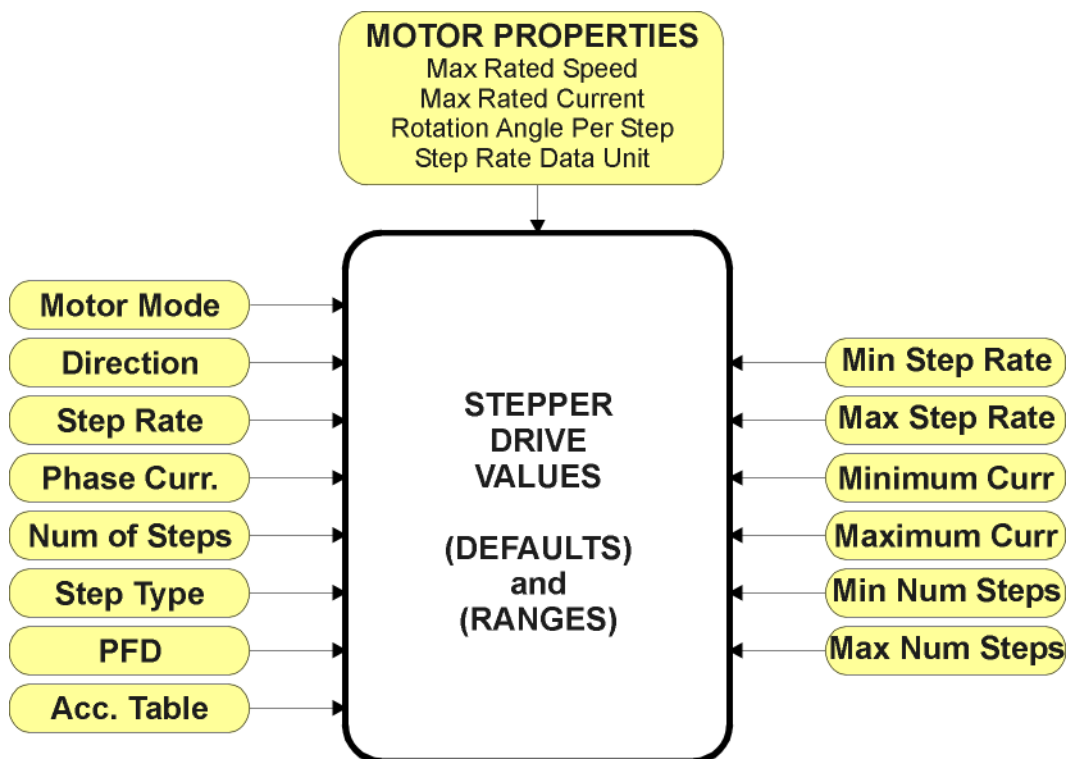


Figure 2. Stepper Drive Values Function Block

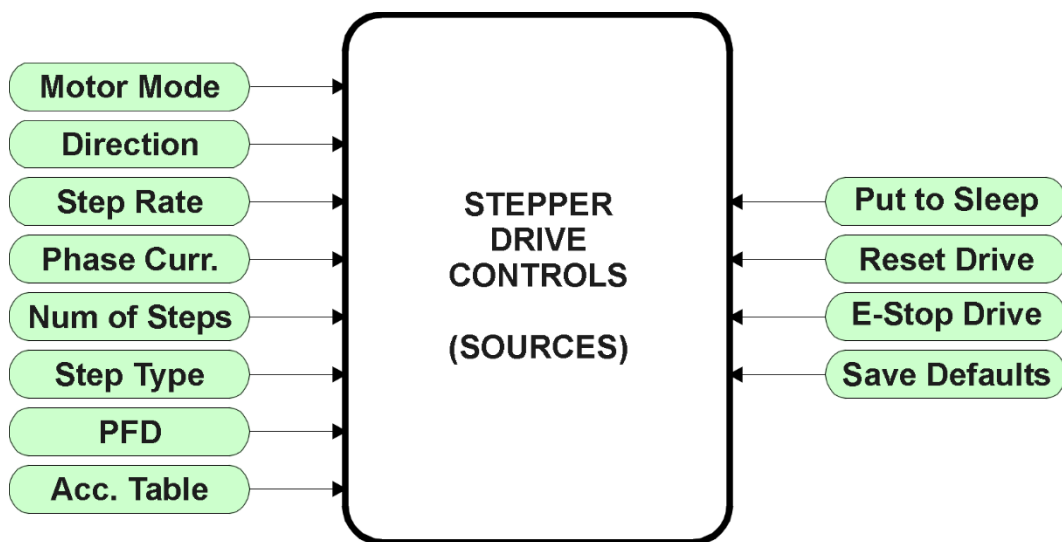


Figure 3. Stepper Drive Controls Function Block

1.2.1 Rotation Angle Per Step

This setpoint is only accessible through the Axiomatic EA in the “Stepper Drive Values” function block. It is to be set by the user to reflect the stepper motor’s step angle. This value is used by the drive when the step rate sent to the unit is in RPM. It uses it to calculate the corresponding Step/Sec signal to apply to the drive chip. By default, this value is set to 18°.

1.2.2 Step Rate Data Unit

This setpoint is only accessible through the Axiomatic EA in the “Stepper Drive Values” function block. It is used by the driver to interpret the data sent by the Step Rate control. By default, step rate data is read as *Steps Per Second*, and the available options for this setpoint are listed in the following table.

0	<i>Steps Per Second</i>
1	<i>Rotations Per Minute</i>

Table 1. Step Rate Data Unit Options

1.2.3 Motor Mode

This is the global control function for the drive. As a CAN command, it can be sent as one of three states, *Disabled*, *Stopped* or *Running*. As a digital input, it will either be commanded to the *Stopped* (input OFF) or *Running* (input ON) modes.

As a feedback signal, however, there are two additional modes that will be set by the drive, *Stepping* and *Holding*. These modes reflect how the drive is operating.

“Default Motor Mode” is a read-only setpoint, as the unit always starts up in *Stopped* mode after a power-up or reset. The available options for this setpoint are listed in the following table.

0	<i>Disabled</i>
1	<i>Stopped</i>
2	<i>Running</i>
3	<i>Stepping (read-only)</i>
4	<i>Holding (read-only)</i>

Table 2. Motor Mode Options

When commanded to go to the *Disabled* mode, the drive will immediately shutoff (no current through the phases) and reset.

When no acceleration table has been selected, and when the number of steps has been commanded to zero (continuous operation when running), the *Stopped* mode behaves the same as *Disabled*.

1.2.3.1 Motor Mode Used with an Acceleration Table

Should an acceleration table be engaged, when commanded to a *'Stopped'* mode while running or stepping, the motor will **decelerate** to zero from whatever step rate it was applying at the time the *'Stopped'* command was received. (Note: This does not apply if the unit is in a *'Holding'* mode, it will simply shutoff) More information about acceleration/deceleration profiles, refer to section 1.2.10.

If the motor has been *'Stopped'* (not holding) and is then commanded to *'Running'* while an acceleration table has been selected, the motor will **accelerate** from zero to the commanded step rate. If no acceleration table is used, then the motor will immediately start stepping at the commanded rate as soon as the motor mode is commanded to *'Running.'*

1.2.3.2 Motor Mode Used with Number of Steps

When the unit is commanded to the *'Running'* mode, it will drive the motor the commanded **"Number of Steps."** If this is a non-zero value, it steps the motor X steps, and will report its status as *'Stepping'* in a feedback message. Once it has reached X steps, it will stop spinning the motor, and hold the phases at the current level it was at when it finished stepping. In this case, it will report its status as *'Holding.'*

If a zero value has been commanded for the **"Number of Steps,"** the drive will run the motor continuously at the commanded **"Step Rate"** while the commanded **"Motor Mode"** remains as *'Running.'*

If the CAN message commanding the **"Number of Steps"** control is expected to be sent repeatedly (i.e., the **"Received Message Timeout"** is a non-zero value), then the **"Motor Mode"** signal can be used to re-activate the stepping action.

To clarify, every time the **"Number of Steps"** command changes in the CAN message, the unit will step the motor X steps. However, should it be desired to step again the same number of steps as used the last time the motor moved, the drive will not respond to the same data sent over and over, as it only reacts to changing data.

To work around this condition, when the CAN message is repeatedly broadcasted at some regular time interval, the **"Motor Mode"** data can be toggled from 1=Stopped to 2=Running and back again to trigger a new action to move X steps. Every time the mode changes, the drive will go into *'Stepping'* mode for X steps, then revert to *'Holding'* mode. To shut off the drive completely in this case, the mode would have to be commanded to *'Disabled.'*

Alternatively, the **"Number of Steps"** command can be sent only once each time the drive is to step the motor. In this case, the **"Received Message Timeout"** will be set to zero, and after the drive has finished stepping the motor, it will reset the CAN message data to "not available" such that the unit will react to the same data the next time it receives the CAN message.

In this case, the toggling action of the **"Motor Mode"** command is not required. Therefore, this control is used to establish how the drive will react after the stepping has completed. In *'Stopped'* mode, it returns to zero current through both phases. In *'Running'* mode it will go to the *'Holding'* mode after it has finished stepping.

1.2.3.3 Motor Mode Used with PID Control

The last point to mention about the **"Motor Mode"** is that it is the primary control source to change to operate the drive with the PID control function. When the **"Control Source, Motor Mode"** setpoint is set to *'PID Function Block Control'*, the drive will automatically update the **"Control Source, Direction"** and **"Control Source, Step Rate"** setpoints to also use the PID control.

When the PID control is used, the motor's mode, direction and speed are all controlled via the PID algorithm selected. All other parameters such as phase current or step type will run using their default value or can still be changed via CAN messages. For more information about the PID control, refer to section 1.2.15.

1.2.4 Direction

This setpoint is used by the driver to control the direction of rotation of the motor. By default, the motor will spin in the *'ClockWise'* direction, and the available options for this setpoint are listed in the following table.

0	<i>CounterCW</i>
1	<i>ClockWise</i>

Table 3. Motor Direction Options

1.2.5 Step Rate

This setpoint is used by the driver to control the speed of the motor. It is a linear control, and therefore the data that is used to control the speed does not necessarily have to be in the same units as the speed data.



NOTE: In the **"Step Rate"** setpoints, a 'Step' is interpreted as the full 360° cycle as shown in Figure 5 and Figure 6. Sixteenth-Step Waveforms (i.e., from Home position to Home position)

For example, a CAN message could be sent as a 0 to 100% value and the step rate will be calculated according to the formula shown in Figure 4. In this case, the Xmin and Xmax values would be 0 and 100, as set in **"Received Data Min"** and **"Received Data Max"**, while the Ymin and Ymax would be setpoints **"Limit Step Rate Min"** and **"Limit Step Rate Max"** respectively.

$$y = mx + a$$

$$m = \frac{Y_{\max} - Y_{\min}}{X_{\max} - X_{\min}}$$

$$a = Y_{\min} - m * X_{\min}$$

Figure 4. Linear Slope Calculations

Depending on the value of **"Step Rate Data Unit"**, the drive will interpret the calculated 'Y' value as either [Step/s] or [RPM]. It would then set the appropriate rate for the drive signal to the chip controlling the motor phases.

Therefore, all the users must do is set the minimum and maximum step rates according to the stepper motor specifications and their application. Defaults and ranges for the step rate setpoints are listed in section 4.5.

1.2.6 Max. Phase Current

This setpoint is used by the driver to control the amplitude of the phase current at the maximum point in the step. Refer to the figures in section 0 for more information. Please note, this value does NOT reflect the average RMS current through each phase.

This setpoint is also a linear control. Like the step rate, the maximum phase current set for the driver chip will be calculated using the formula in Figure 4. The X axis is represented by the received CAN message setpoints **"Received Data Min"** and **"Received Data Max"** while the Y axis is calculated based on the setpoints **"Limit Phase Current Min"** and **"Limit Phase Current Max."**

Note, however, that Ohm's law will still apply. If the calculated phase current setting is higher than what the stepper motor is capable of drawing at the applied voltage level, then the motor will simply draw whatever it is capable of.

Therefore, the user should set the minimum and maximum currents according to the stepper motor specifications and their application. Defaults and ranges for the phase current setpoints are listed in section 4.5.

1.2.7 Number of Steps

This setpoint is used by the driver step the motor a set number of steps, and then either stop or hold it. Please refer to section 1.2.3.2 for more information about how to use this control in conjunction with the motor mode.

This setpoint is also a linear control. Like the step rate, the number of steps will be calculated using the formula in Figure 4. The X axis is represented by the received CAN message setpoints **"Received Data Min"** and **"Received Data Max"** while the Y axis is calculated based on the setpoints **"Limit Number of Steps Min"** and **"Limit Number of Steps Max."**

"Limit Number of Steps Min" is a read-only setpoint and is set to zero representing continuous operation. The user should set the maximum according to their application. Defaults and ranges for the number of steps setpoints are listed in section 4.5.



NOTE: In the **"Number of Steps"** setpoints, a 'Step' is interpreted as a 90° shift of the rotor with respect to the driving phases. There are always 4 steps per 360° cycle as shown in Figure 5 and Figure 6. Sixteenth-Step Waveforms.

1.2.8 Step Type

This setpoint is used by the driver to control the micro steps used to get different phase current waveforms. By default, the step type used is '*SixteenStep*', and the available options for this setpoint, as supported by the driver chip A3979, are listed in the following table.

0	<i>FullStep</i>
1	<i>HalfStep</i>
2	<i>QuarterStep</i>
3	<i>SixteenStep</i>

Table 4. Step Type Options

The following figures, extracted from the Allegro datasheet for the driver chip, show the current waveforms on each phase depending on the step type selected.

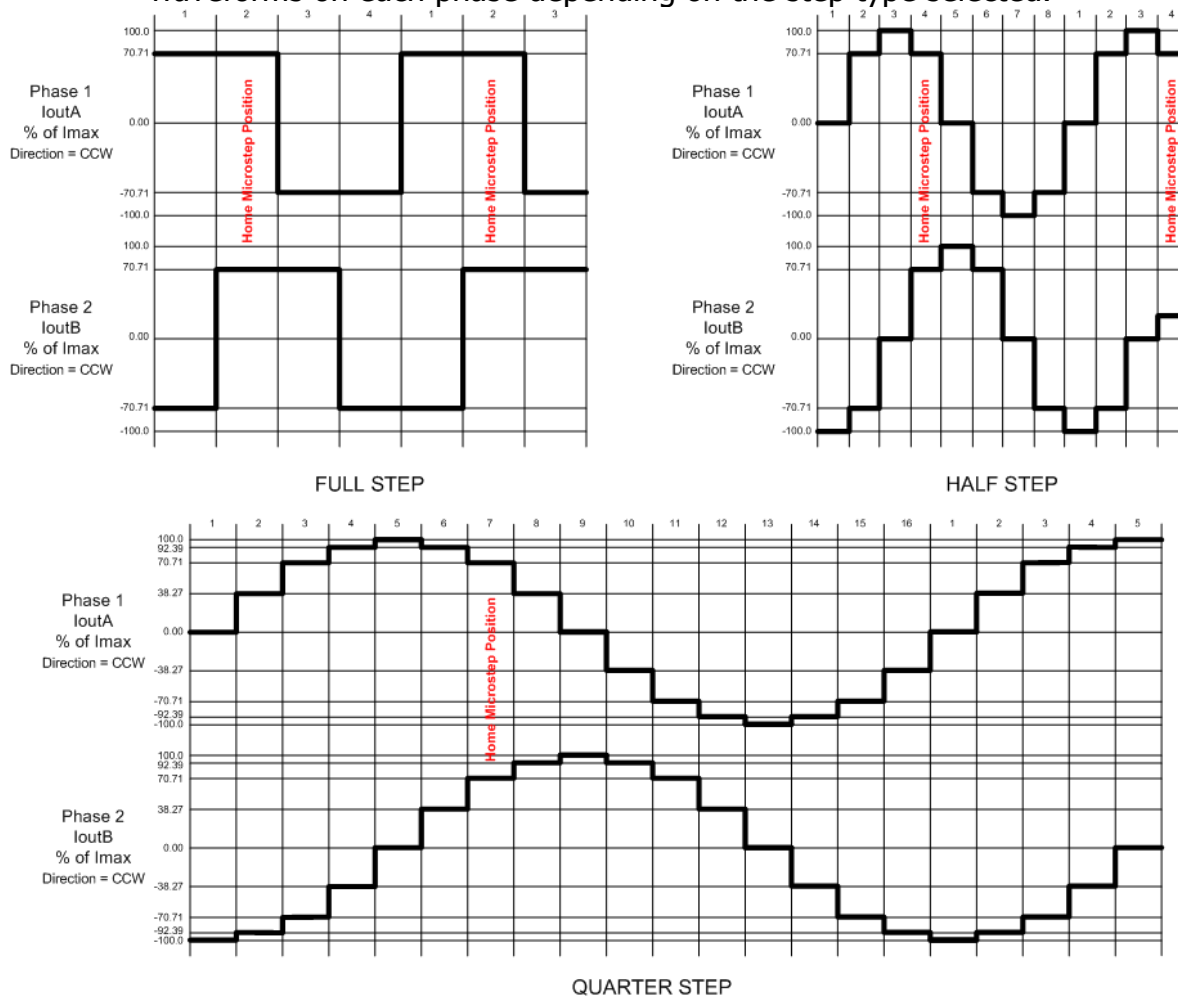


Figure 5. Full, Half and Quarter-Step Waveforms

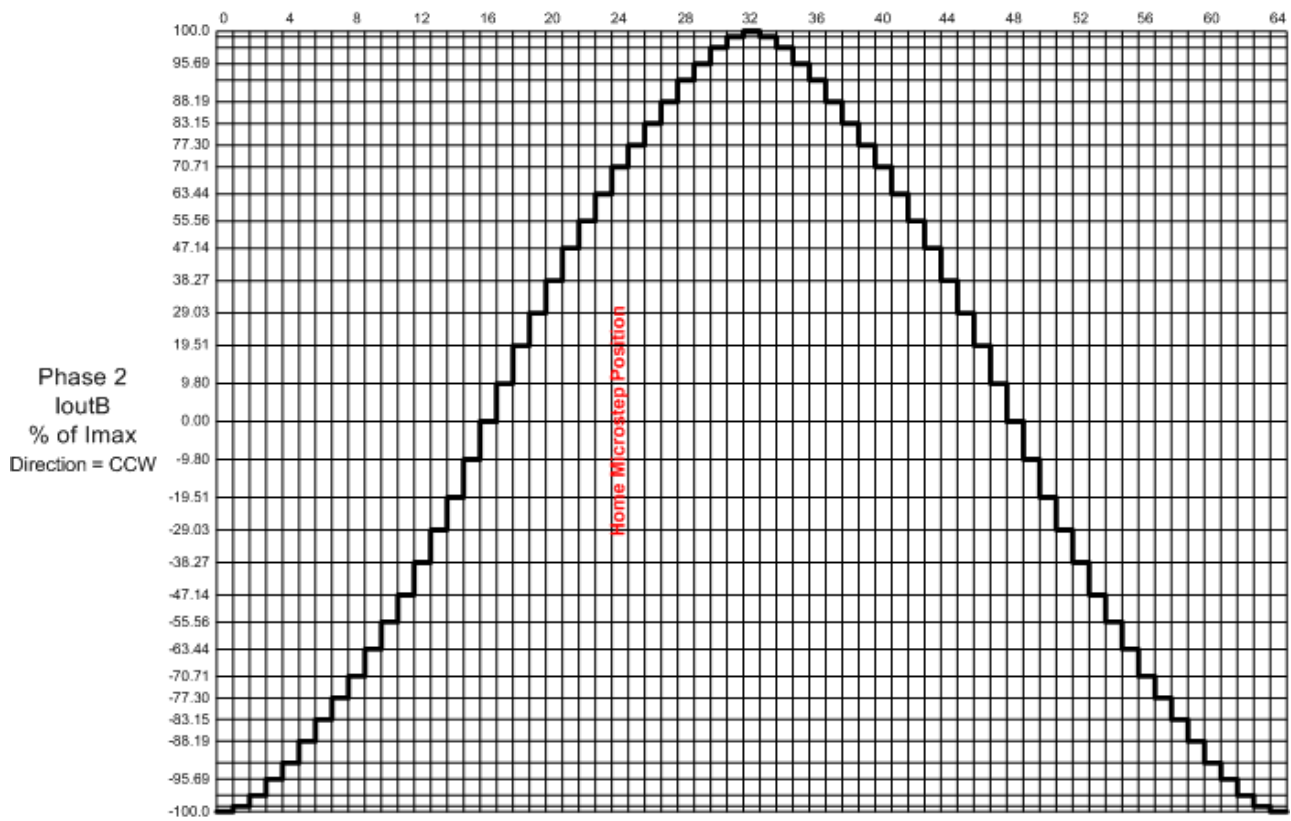
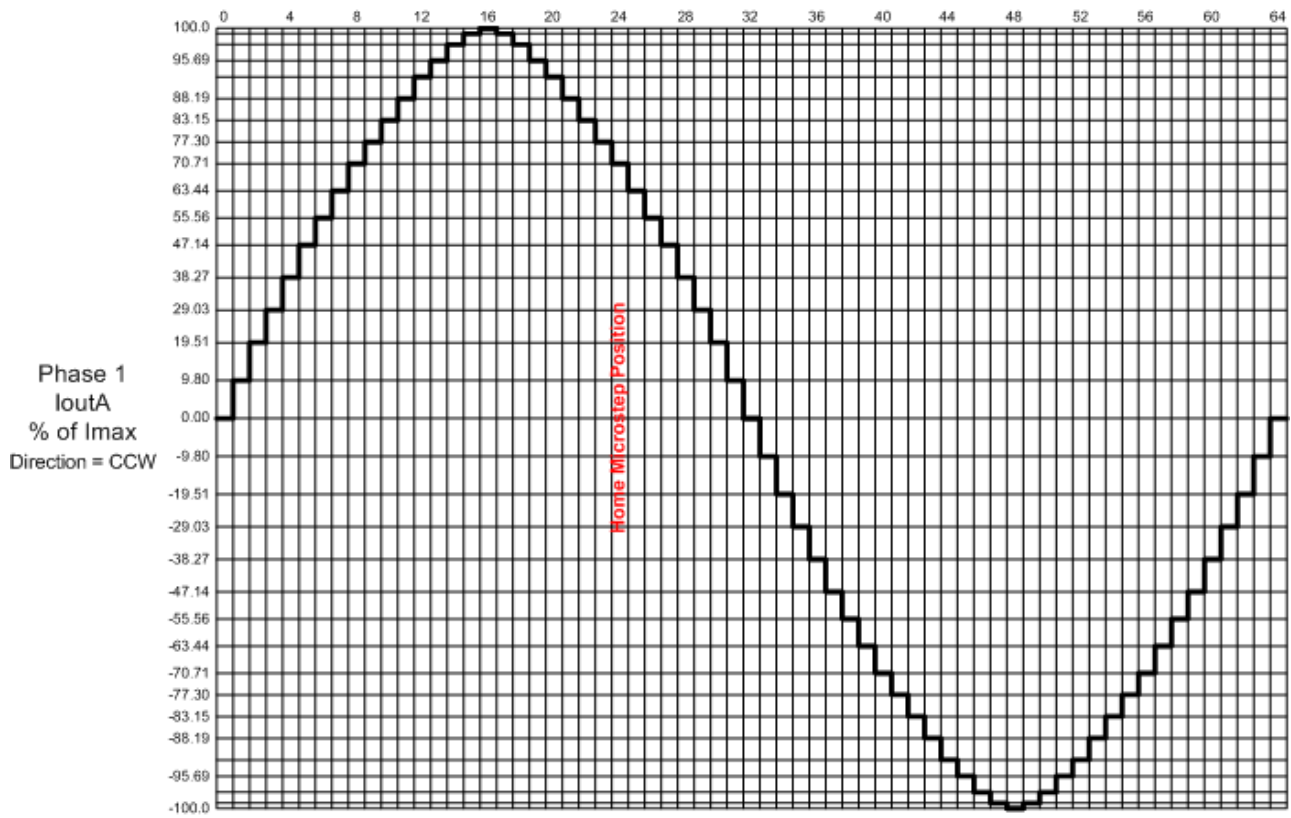


Figure 6. Sixteenth-Step Waveforms

1.2.9 Percent Fast Decay

This setpoint is used by the driver to control the percent fast decay (PFD), as supported by the driver chip A3979. The follow text was copied directly from the Allegro datasheet explaining how the PFD is used by the chip.

When a STEP input signal commands a lower output current than the previous step, it switches the output current decay to Slow, Fast, or Mixed decay mode, depending on the voltage level at the PFD input. If the voltage at the PFD input is greater than $0.6 \times V_{DD}$, then Slow decay mode is selected. If the voltage on the PFD input is less than $0.21 \times V_{DD}$, then Fast decay mode is selected. Mixed decay mode is selected when V_{PFD} is between these two levels, as described in the next section. This terminal should be decoupled with a $0.1 \mu F$ capacitor.

Mixed Decay Operation. If the voltage on the PFD input is between $0.6 \times V_{DD}$ and $0.21 \times V_{DD}$, the bridge operates in Mixed decay mode, as determined by the step sequence (shown in Figure 2-Figure 5). As the trip point is reached, the device goes into Fast decay mode until the voltage on the RCx terminal decays to the same level as voltage applied to the PFD terminal. The time that the device operates in fast decay is approximated by:

$$t_{FD} = R_T C_T \ln (0.6 V_{DD} / V_{PFD})$$

After this Fast decay portion, the device switches to Slow decay mode for the remainder of the fixed off-time period.

By default, the PFD used is 'Mixed', and the available options for this setpoint, as supported by the driver chip A3979, are listed in the following table.

0	<i>Slow</i>
1	<i>Fast</i>
2	<i>Mixed</i>

Table 5. Percent Fast Decay Options

1.2.10 Acceleration Table

As mentioned in section 0, the stepper drive supports acceleration/deceleration lookup tables. If selected, the table is used when the motor is started or stopped, to dynamically change the step/s/s. Up to five different acceleration/deceleration curves are supported.

Acceleration is only used when the motor is being started from a stopped position. Deceleration is only used when the motor is being fully stopped (i.e., to zero) while running. Changes to the step rate during operation, i.e., to make the motor faster or slower, are implemented immediately based on the new rate, without using any acceleration/deceleration.

When accelerating, if the commanded step rate is lower than the maximum step rate at the end of the acceleration table, then the motor will only accelerate to the commanded rate and then continue to spin at that speed.

For example, using the example table shown in Table 7

Figure 7, if the commanded step rate is 48 Steps/s, and the motor is started, then the acceleration table will be followed for the first 50ms. It will then stop at 48 Steps/s 60ms after the 'Running' motor mode command was received.

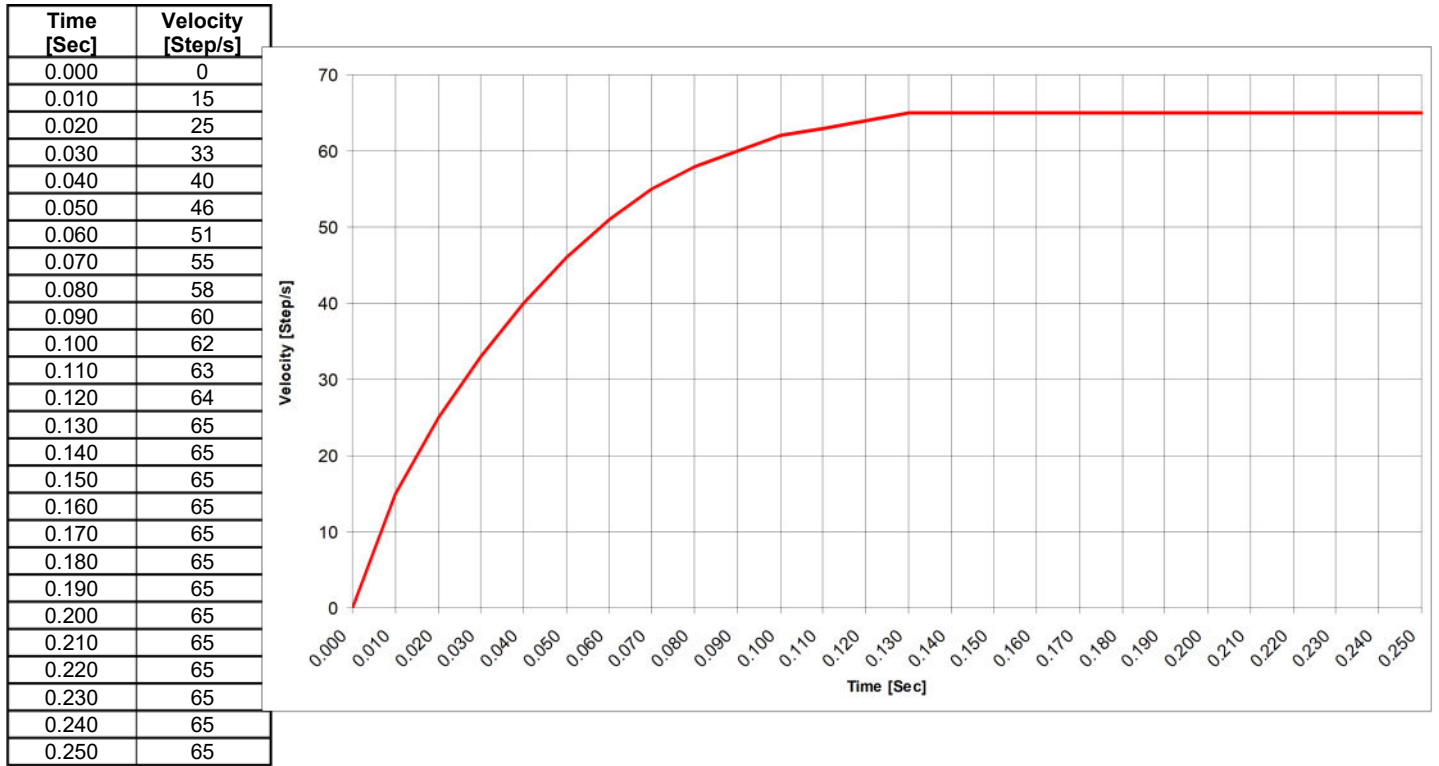


Figure 7. Example Acceleration Curve

Conversely, if the motor was running at 48 Steps/s when the 'Stopped' motor mode command was received, then the drive would establish where the motor was running on the deceleration curve and decelerate to zero from there. As with the starting operation, stopping would also take 60ms in the case of the example.

However, should the motor have been running at 70 Steps/s when the 'Stopped' command was received, it would immediately jump to the maximum rate on the curve, in this case 65 Steps/s and hold there for 120ms (130ms to 250ms). It would then decelerate to zero, changing the rate every 10ms as shown on the curve. To avoid the delay at the beginning, the last 12 entries on the table could have all been set to 130ms in the Time column.

To select an acceleration/deceleration profile, a non-zero **"Acceleration Table"** must be commanded. Otherwise, the drive will simply jump to or from the commanded **"Step Rate"** as the **"Motor Mode"** changes from 'Stopped' to 'Running' or vice versa. By default, no acceleration table is used, and the available options for this setpoint, as supported are listed in the following table.

0	No Profile
1	Profile1
2	Profile2
3	Profile3
4	Profile4
5	Profile5

Table 6. Acceleration Table Options

1.2.11 Put Drive to Sleep

A feature of driver chip A3979 is that it can be put into sleep mode to draw a low current when it is not operating. To utilize this feature, the drive can receive a command from the J1939 network to put the drive to sleep.

This input is treated as a Boolean value by the drive, so the data received is interpreted as either "False" (default, drive is awake) or "True" (drive is put to sleep). By default, the corresponding CAN Receive message has its default data set with **"Received Data Min (Off Threshold)"** as 0.0 and **"Received Data Max (On Threshold)"** as 1.0.

While the **"Put Drive to Sleep"** command is received as 1 (or greater), the drive will be disabled, and the chip is in low-power sleep mode. To reset the drive after the command has cleared (gone back to zero, or lower), the **"Motor Mode"** must transition from 'Stopped' to 'Running'. Alternatively, one of the other command data messages, such as the step rate, must change.

1.2.12 Reset Motor Drive

A feature of driver chip A3979 is that it reset the output current to the micro step "Home" position as shown in Figure 5 and Figure 6. Sixteenth-Step Waveforms. To utilize this feature, the drive can receive a command from the J1939 network to reset the drive to "Home." While in reset mode, the motor will not be running.

This input is treated as a Boolean value by the drive, so the data received is interpreted as either "False" (default, is not reset) or "True" (drive is reset to home). By default, the corresponding CAN Receive message has its default data set with **"Received Data Min (Off Threshold)"** as 0.0 and **"Received Data Max (On Threshold)"** as 1.0.

While the **"Reset Motor Drive"** command is received as 1 (or greater), the drive will be disabled, and the chip will begin micro-stepping from the home position when it restarts. To re-enable the drive after the command has cleared (gone back to zero, or lower), the **"Motor Mode"** must transition from 'Stopped' or 'Holding' to 'Running'. Alternatively, one of the other command data messages, such as the step rate, must change.

1.2.13 E-Stop Motor Drive

Another way to fully disable the motor drive (i.e. ensure that there is zero current running through both phases) is to use the Emergency Stop (E-Stop) input. By default, this input is mapped to a CAN Receive message, but it could also be linked to one of the on-board inputs configured as a digital input.

When controlled by a CAN message, this input is treated as a Boolean value by the drive, so the data received is interpreted as either "False" (default, E-Stop is OFF) or "True" (E-Stop ON, drive disabled). By default, the corresponding CAN Receive message has its default data set with **"Received Data Min (Off Threshold)"** as 0.0 and **"Received Data Max (On Threshold)"** as 1.0.

This signal is a true override signal, such that while **"E-Stop Motor Drive"** is active (ON), the drive will not run. To re-enable the drive after the E-Stop has cleared (gone back to zero, or lower), the **"Motor Mode"** must transition from *'Stopped'* to *'Running'*.

1.2.14 Save Data to Defaults

It may be desirable, in situations where the Axiomatic EA is not connected to the network, to be able to change the power-up default values used by the stepper drive by simply sending a CAN message. In order to support this feature, there is an option to save the current motor drive settings to non-volatile memory by simply setting a CAN bit.

This input is treated as a Boolean value by the drive, so the data received is interpreted as either "False" (no command to save) or "True" (save data). By default, the corresponding CAN Receive message has its default data set with **"Received Data Min (Off Threshold)"** as 0.0 and **"Received Data Max (On Threshold)"** as 1.0.

The save operation will only occur when the **"Save Data to Defaults"** bit transitions from a low (OFF) to a high (ON).

1.2.15 PID Motor Control Function Block

The PID Control Function block can be used to control the “Motor Mode”, “Direction” and “Step Rate” commands as described in Section 1.2.3.3.

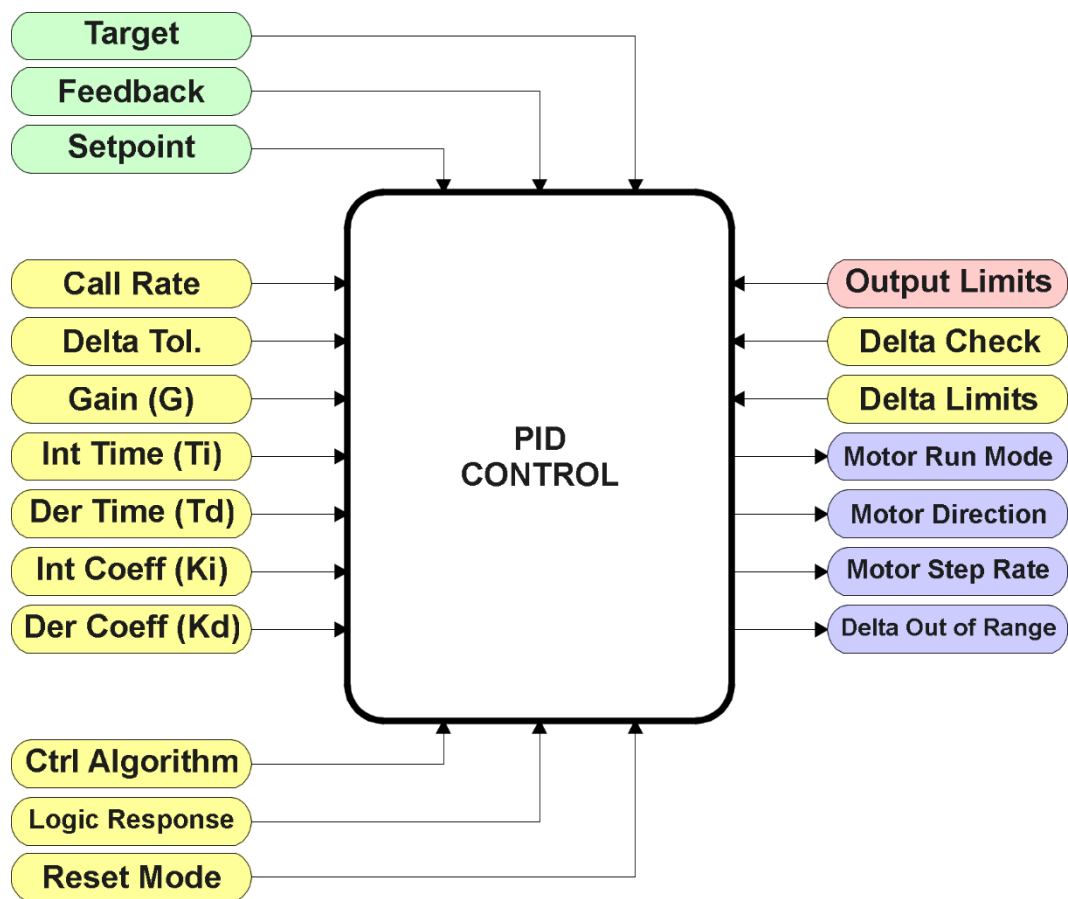


Figure 8. PID Control Function Block

When used, the “**Motor Mode**” and “**Direction**” will automatically be controlled by the PID function, depending on the “**PID Logic Response**” selected. The PID output will be some calculated value between 0 to 100% which will be used as the X-axis input to the “**Step Rate**” control (see Section 1.2.5)

1.2.15.1 Target versus Setpoint Control

There are two types of “**PID Control Algorithm**” that can be selected to establish how the PID function will response to the Feedback input, as listed in the following table.

0	Target Control
1	Setpoint Control

Table 7. PID Control Algorithm Options

When the ‘Target Control’ option is used, the PID loop will use the value of the Target input, as set by the “**PID Target Command Source**” and “**Number**” setpoints to compare to the Feedback input, as set by the “**PID Feedback Command Source**” and “**Number**” setpoints.

Both inputs will be normalized to 0 to 100%, using the input limits to set the range. For example, if the Target input is selected using a CAN Receive message, "**Received Data Min**" corresponds to 0% while "**Received Data Max**" is treated as 100% in the linear calculation defined in Figure 4. However, should the Target be controlled using the Frequency/PWM Input 1, then the input's "**Minimum Range**" corresponds to 0%, while the "**Maximum Range**" is 100%. Similarly, the same is true for the Feedback input. Because of the normalization to a percentage of the signal, it is not required that the Target and Feedback inputs be in the same units (i.e., both voltages.)

However, if the '*Setpoint Control*' option is used, it is assumed that the "**PID Setpoint Control Target**" value will be defined in the same unit as the Feedback input. It too will be normalized to a percentage, using the limits of the Feedback input to determine the range for the 0 to 100% linearization calculation. (Note: When setpoint control is used, the command source setpoints are not used in the algorithm, although they still show up as available in the Axiomatic EA.)

In order to allow the output to stabilize, the user can select a non-zero value for "**PID Delta Tolerance**". If the absolute value of $Error_k$ is less than this value, $Error_k$ in the formula shown in Figure 9 will be set to zero. The units for the tolerance are also assumed to be the same as the Feedback input. Like the setpoint value, the tolerance will be normalized to a percentage using the Feedback input range.

1.2.15.2 PID Control Algorithm

As described above, all the values used in the PID Control Algorithm are handled as a percentage value when inputted into the formula.

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

```

T = Loop_Update_Rate*0.001

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

Errork = Target - Feedback
ErrorSumk = ErrorSumk-1 + Errork

Pk = Errork * P_Gain

Ik = ErrorSumk * I_Gain

Dk = (Errork-Errork-1) * D_Gain

PIDOutputk = Pk + Ik + Dk

```

Figure 9. PID Control Algorithm

When active, the PID algorithm will be called every "**PID Loop Update Rate**" (in ms)

Each system will have to be tuned for the optimum motor response. Response times, overshoots and other variables will have to be decided by the customer using an appropriate PID tuning strategy.

1.2.15.3 PID Control Logic

It is recognized that in some systems, it could be desirable to have a push-pull type response. That is, the motor will run in one direction (pushed) when under target and runs in the other direction when over (pulled). In other systems, it may be that the motor is expected to run at some speed to maintain a target variable. For this reason, there is a setpoint **"PID Logic Response"** which will define how the motor will respond to changes in the Target or Feedback inputs. The options for this setpoint are listed in the following table.

0	<i>Run Motor Unidirectional</i>
1	<i>Motor CW when Under Target</i>
2	<i>Motor CW when Over Target</i>

Table 8. PID Logic Response Options

When running in the first mode '*Run Motor Unidirectional*', a traditional PID response will occur, where the motor will run faster the farther away the feedback is from the target and will slow down as the difference decreases. Once the feedback signal is within tolerance of the target/setpoint, the PID output will remain constant at whatever value it was at when the signals converged.

In this type of control, the direction of the motor will always be as per defined in the **"Default Direction."** The motor mode will always be '*Running*' (can still be stopped using the E-Stop input).

However, in a push-pull type system (selections 1 or 2 from Table 8); a purely proportional response at the output is likely more desirable than a PI or PID response. As the feedback signal goes farther away from the target, the motor should run faster to bring it back in alignment sooner. As the signals start to converge, however, the motor should slow down to not overshoot the target. For this reason, the **"PID Integral Coefficient, Ki"** and the **"PID Derivative Coefficient, Kd"** should be set to zero to get a P-only response. The **"PID Gain Coefficient, G"** can be adjusted to change the error versus speed response of the system.

In this type of control, the direction of the motor will be determined by the delta error (difference) between the target/setpoint value and the feedback signal.

1.2.15.4 Reset Response

Typically, the motor will be '*Running*' while the PID algorithm is being used. However, as explained earlier, when the error between the target and feedback values are within the **"PID Delta Tolerance"** value, the PID function is reset, and the Step Rate for the motor is set to 0.

In this instance, the **"PID Reset Response"** determines if the motor will be '*Disabled*', '*Stopped*' or '*Holding*' while the PID loop is idle. The default response is '*Holding*'.

1.2.15.5 Delta Checking

In some instances, it may be desired to have an indicator if the delta error, determined by the difference between the (*Target – Feedback*) inputs, goes beyond some threshold. When the Delta value goes out of range, a flag will be set (0 or 1), and this flag can be used as an input to another function block, i.e., the Digital Output, or a CAN Transmit message.

By default, this feature is disabled, however Table 9 outlines the options that can be selected for the **"PID Delta Check Response"** to activate the indicator.

0	<i>Delta Checking Disabled</i>
1	<i>Set when (Target-Feedback) more than limit</i>
2	<i>Set when (Target-Feedback) less than limit</i>
3	<i>Set when Absolute (Target-Feedback) more than limit</i>
4	<i>Set when Absolute (Target-Feedback) less than limit</i>

Table 9. PID Delta Check Response Options

The setpoints **"PID Delta Check Set Limit"** and **"PID Delta Check Clear Limit"** are used only when the response is set to a non-zero value. Like the **"PID Delta Tolerance"**, the units for these setpoints are also assumed to be the same as the Feedback input and they will be normalized to a percentage using the Feedback input range.

The two limit values are provided in order to provide some hysteresis for the *Delta Out of Range* flag. If option 1 or 3 is selected, **"PID Delta Check Set Limit"** must be higher than **"PID Delta Check Clear Limit."** When option 2 or 4 is used, the **Clear** limit must be higher than the **Set** limit. By default, both limits are set to zero.

1.2.15.6 Default Configuration

By default, the PID Control Function is not used. However, should it be selected, the default settings are that it will use the voltage input as the feedback signal, and be setup for a '*Setpoint Control*' response, targeting 2.5V. There will be a +/- 100mV allowable tolerance between setpoint and feedback. The motor will rotate clockwise when the feedback value is less than the setpoint, and counterclockwise when the feedback is too large. The output will respond proportionally by the gain value, G, as the feedback and target values diverge or converge. The Ki and Kd setpoints are set to 0.0 to disable the integral and derivative portions of the response.

See Section 4.5 for more information about the default settings.

1.3 Input Function Blocks

There are two types of input function blocks, one for frequency/PWM and the other for voltage/current/frequency/PWM inputs. Both input function blocks are virtually identical whether measuring an ADC or Timer input, as shown below.

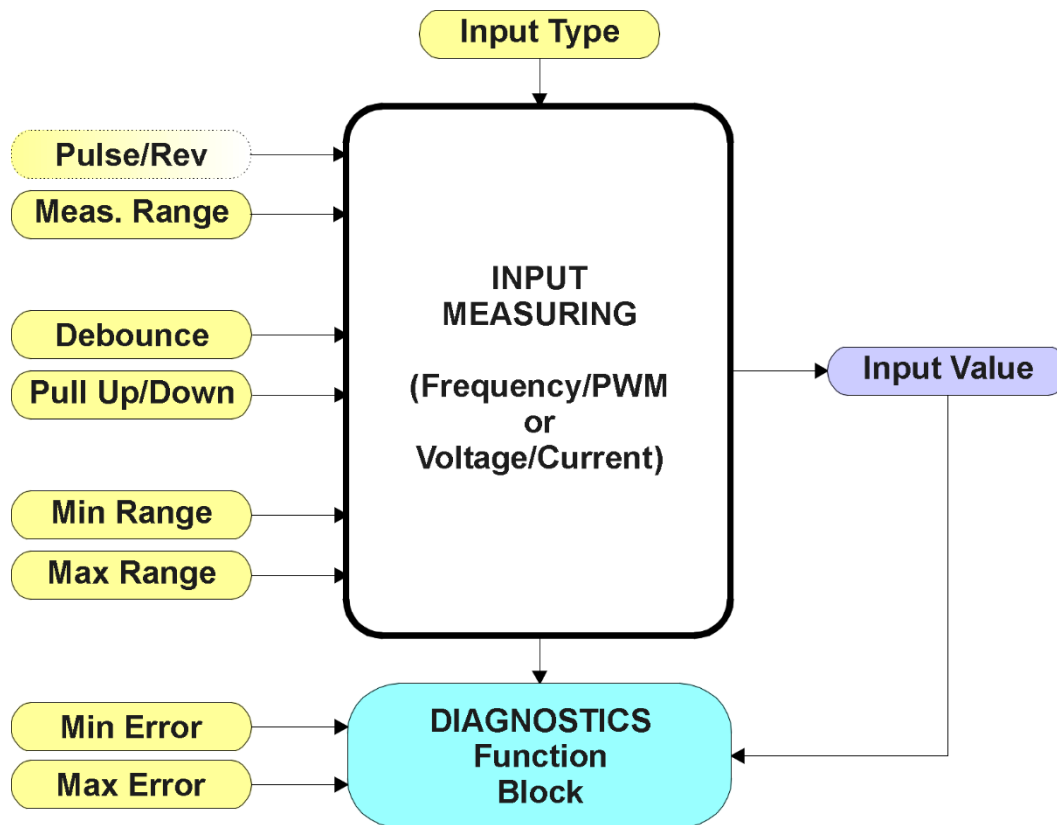


Figure 10. Input Function Block

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

1.3.1.1 Input Circuits

The Analog Input can be selected as either a Voltage or Current input. It has a hardware filter that eliminates the effects of noise on the input, which then goes directly to the pin of a 12-bit analog-to-digital converter (ADC.) A voltage input is a high impedance (1MΩ) input protected against shorts to GND or Vcc. In current mode, a 130Ω resistor is used to measure the input signal. The Analog Input should be connected to the AGND reference pin provided on the connector.

The following “**Analog Input Type**” options can be selected for Voltage/Current Input 1.

0	<i>Disabled</i>
1	<i>Voltage</i>
2	<i>Current</i>
3	<i>Digital</i>
4	<i>Active Freq/RPM</i>
5	<i>PWM Duty Cycle</i>

Table 10. Analog Input Type Options

The Frequency Input can read either a Frequency/RPM or Pulse Width Modulated (PWM) input. The circuit is designed to read either an AC input signal, such as one from an inductive pickup sensor, or an active signal swinging between +Vcc and GND, such as from a Hall Effect sensor. For active inputs, a 10kΩ pull-up or pull-down resistor could also be connected at the input to handle open-collector NPN (active low, pull-up) or PNP (active high, pull-down) signals. The conditioned input signal is then connected to a 16-bit timer pin on the processor. It is fully independent of the Analog Input.

The following “**Frequency Input Type**” options can be selected for Frequency/PWM Input 1.

0	<i>Input Disabled</i>
1	<i>Inductive Pickup</i>
2	<i>Active Freq/RPM</i>
3	<i>PWM Duty Cycle</i>
4	<i>Digital Signal</i>

Table 11. Frequency Input Type Options

As can be seen from Table 10 and Table 11, both input types could also be configured as a Digital Input. In either case, an active high or low signal could be read by selecting the appropriate pull-up or pull-down 10kΩ resistor, as shown in the following table.

0	<i>Pullup/down Off</i>
1	<i>10kOhm Pullup</i>
2	<i>10kOhm Pulldown</i>

Table 12. Pullup/Pulldown Resistor Options

A regulated +5V is provided to power an input such as a potentiometer. It is referenced with respect to the AGND used by the Analog Input and is capable of sourcing up to 10mA.

1.3.1.2 Measuring Data

Each input type has associated with it a configurable “**Input Measuring Range.**” This setpoint allows the user to optimize the hardware for the input signal.

For voltage inputs, there are four input ranges that can be selected. Typically, inputs will be in either the 0-5V or 0-10V range, but for more resolution at smaller voltages, a low range can be selected when appropriate. However, this hardware does not actually support the 0-1V input range, so if it selected, the drive would default back to a 0-5V range.

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

Table 13. Voltage Input Ranges

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

0	0 to 20mA
1	4 to 20mA

Table 14. Current Input Ranges

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

		Overflow [Sec]
0	0.5Hz to 50Hz	~4 [0.25Hz]
1	10Hz to 1kHz	~0.105 [9.5Hz]
2	100Hz to 10kHz	~0.010 [95Hz]

Table 15. Frequency Input Ranges

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

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

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

Table 16. PWM Input Ranges

The “**Minimum Range**” and “**Maximum Range**” setpoints must not be confused with the measuring range. These setpoints are available with all but the digital input, and they are used when the input is selected as a control input for another function block. They become the Xmin and Xmax values used in the slope calculations (see Figure 4). When these values are changed, other function blocks using the input as a control source are automatically updated to reflect the new X-axis values.

1.3.1.3 Diagnostic Detection

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

$$0 \leq \text{Minimum Error} \leq \text{Minimum Range} \leq \text{Maximum Range} \leq \text{Maximum Error} \leq 1.1 \times \text{Max}^*$$

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

Frequency: Max = 10,000 [Hz or RPM]

PWM: Max = 100.00 [%]

Voltage: Max = 1.00, 2.50, 5.00 or 10.00 [V]

Current: Max = 20.00 [mA]

More information about the Diagnostic function block can be found in Section 1.12.

1.3.1.4 Digital Inputs

Since either input could be configured to act as a digital input, they both have this as an option for “**Input Type.**” In this setup, when using it as an input to another function block (i.e., an E-Stop signal), map the control to the ‘*Digital Input Read State*’ type instead of the ‘*Input Measured Value.*’ Here, the analog input becomes Digital Input 1 and the Frequency input becomes Digital Input 2.

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

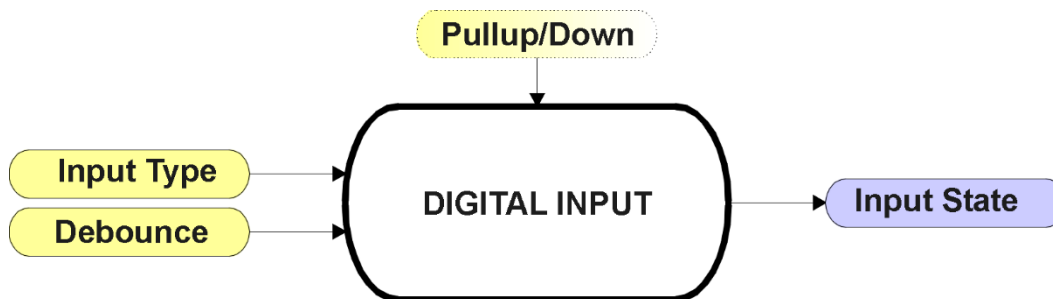


Figure 11. Digital Input Function Block

To prevent spurious signals from affecting the logic of the controller, all digital inputs have associated with them a debounce time. If and only if the signal at the end of the debounce period has changed state will the change of state of the input be translated to the controller. The “**Debounce Time**” setpoint can be selected from the table below.

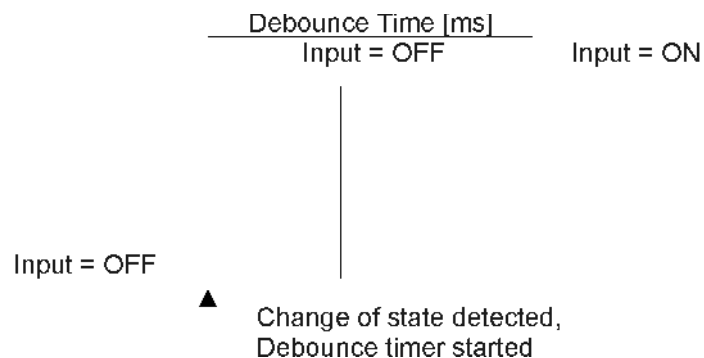


Figure 12. Digital Input Debouncing

0	None
1	111ns
2	1.78us
3	14.22us

Table 17. Debounce Time Options

In this case, the **"Input Measuring Range"** setpoint determines the Input State used by other function blocks based on the actual state of the input at the pin, per the table below.

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

Table 18. Digital Input Type versus Input State

1.4 Encoder Input Function Block

The controller provides three Encoder inputs, each of them has two physical inputs for connection A and B. These inputs can be configured to measure digital signals generated by up to three incremental encoders.

Moreover, it is possible to use all 6 inputs of the encoder function block as Digital Inputs. For example, if only one encoder Input (two physical inputs) is used and more than six digital inputs are required, the other four inputs of the encoder function block can be used as additional digital inputs. The input type of the Encoder can be changed with the setpoint **Input Type**. The options for this setpoint are shown in Table 19.

Value	Meaning
0	Input Not Implemented
1	Digital Input
2	Encoder Input

Table 19. Input Type options for Encoder Inputs

The controller has two inputs for detecting the pulses generated by an incremental encoder. The two measurements, step count and direction, are determined using these two input signals.

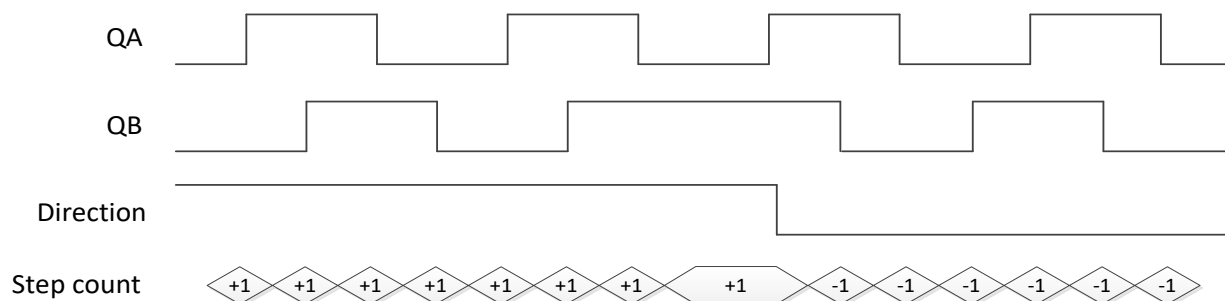


Figure 13: Incremental Decoder Signals (QA&QB) and resulting Direction and Step count

A positive direction is indicated when the edge of signal A rises before the edge of signal B is rising. If the encoder rotates in a positive direction, the **Encoder Counter value** increases. In case the signal changes and input B becomes the leading signal, the counter value decreases, and the direction is indicated as negative.

1.4.1 Encoder Stationary Delay

The **Encoder Stationary Delay** parameter decides after how many milliseconds the direction is set to stationary (0) if no rising edge or falling edge is detected on signal A or B on the encoder input. This way, the performance of the encoder input can be adjusted for slower and faster rotating encoders. The default value for this parameter is set to 100[ms].

1.5 Filtering Function Block

The filtering function block is associated with phase current feedback. When used, the software filter is applied to the measured current feedback value for each of the drive phases before it is used for error detection and as a feedback signal input to other function blocks (i.e. CAN transmission)

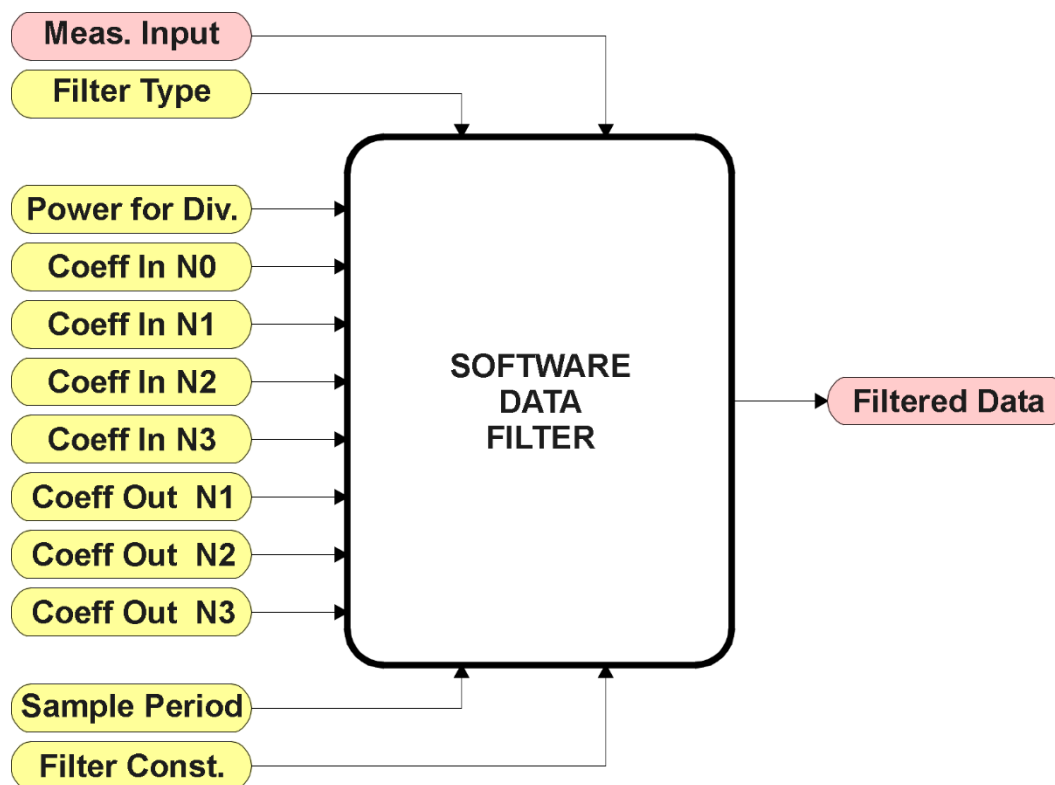


Figure 14. Software Data Filter Function Block

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

0	<i>No Filter</i>
1	<i>Moving Average</i>
2	<i>Repeating Average</i>
3	<i>3rd Order Low Pass</i>

Table 20. Software Filter Types

By default, the Phase Current Filter is setup as a '*Repeating Average*' type with a **"Software Filter Constant"** of 100. The output from the filter block does NOT represent the true RMS value of the phase currents. Rather, it is used solely for error detection to flag when a phase is open-circuited.

1.5.1 Calculation with no filter:

Value = Input

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

Calculation with the moving average filter:

$$\text{Value}_N = \text{Value}_{N-1} + \frac{(\text{Input} - \text{Value}_{N-1})}{\text{Filter Constant}}$$

This filter is called every 1ms.

Calculation with the repeating average filter:

$$\text{Value} = \frac{\sum \text{Input}_N}{N}$$

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

1.5.2 Calculation with the 3rd Order Low Pass filter:

$$\text{Value}_N = \frac{C_{IN0} * \text{Input}_N + C_{IN1} * \text{Input}_{N-1} + C_{IN2} * \text{Input}_{N-2} + C_{IN3} * \text{Input}_{N-3} + C_{OUT1} * \text{Value}_{N-1} + C_{OUT2} * \text{Value}_{N-2} + C_{OUT3} * \text{Value}_{N-3}}{2^{\text{Power}}}$$

C_{IN0} Coefficient for Input (n)

C_{IN1} Coefficient for Input (n-1)

C_{IN2} Coefficient for Input (n-2)

C_{IN3} Coefficient for Input (n-3)

C_{OUT1}

Coefficient for Output (n-1)

C_{OUT2}

Coefficient for Output (n-2)

C_{OUT3}

Coefficient for Output (n-3)

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

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

1.6 Universal Output Function Block

The controller has 1 Proportional output. The Proportional Output is half-bridge drive with either high side or low side sourcing up to 2.5A.

"Output Type" setpoint determines what kind of signal the output produces. Changing this setpoint causes other setpoints in the setpoint group to update to match selected type, thus the **"Output Type"** should be selected before configuring other setpoints within the setpoint group. **"Output Type"** setpoint options are listed in Table 21.

0	<i>Disabled</i>
1	<i>Proportional Current (0-2.5A)</i>
2	<i>Digital Hotshot (0-2.5A)</i>
3	<i>PWM Duty Cycle (0-100%)</i>
4	<i>Proportional Voltage (0-Vps)</i>
5	<i>Digital On/off (0-Vps)</i>

Table 21. Output Type Options for Proportional Output

'Proportional Current' type has associated with it two setpoints not used by other types, which are the **"Dither Frequency"** and **"Dither Amplitude"** values. The output is controlled by high frequency signal (25 kHz), with the low frequency dither superimposed on top. The dither frequency will match exactly what is programmed into the setpoint, but the exact amplitude of the dither will depend on the properties of the load coil. When adjusting the dither amplitude value, select one that is high enough to ensure an immediate response to the coil to small changes in the control inputs, but not so large as to affect the accuracy or stability of the output. Refer to the coil's datasheet for more information.

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

The *'PWM Duty Cycle'* option allows the user to run the output at fixed frequency configure with **"PWM Output Frequency"** setpoint, while the duty cycle changes depending on the control signal. **"PWM Output Frequency"** is editable only if none of the outputs in the output group is set to *'Proportional Current'* or *'Hotshot Digital'* type. Configuring output to *'Proportional Current'* or *'Hotshot Digital'* type changes frequency if the output group automatically to 25 kHz.

Instead of proportional output control, there are also two types of digital responses possible as well. With the *'Digital On/Off'* type, should the control require the output to be on; it will be turned on at whatever the system power supply is. The output will source whatever current is required by the load, up to 2.5A.

If a digital **"Output Type"** has been selected the **"Digital Response"** setpoint will be enabled as shown in Table 22.

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

Table 22. Digital Response Options

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

The **"High-Side/Low-Side Output Setting"** setpoint defines whether the output will have high or low side sourcing. When 'High' is selected there will be high side sourcing and selecting 'Low' will configure the output to have a low side source.

0	<i>Low</i>
1	<i>High</i>

Table 23. High-Side/Low-Side Output Setting

The **"Digital Out Delay"** and **"Delay Polarity"** setpoints are associated with *'Normal'* **"Digital Response"**. The **"Digital Out Delay"** defines output state change delay in milliseconds. Setting **"Digital Out Delay"** to 0ms applies no state change delay. The **"Delay Polarity"** setpoint defines which edge the state change delay will be applied. For example, selecting *'Rising Edge'* for **"Delay Polarity"** and setting **"Digital Out Delay"** value >0ms, applies defined delay when driving output signal state changes from OFF to ON.

0	<i>Rising Edge</i>
1	<i>Falling Edge</i>

Table 24. Delay Polarity

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

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

The *'Hotshot Digital'* type is different from in simple *'Digital On/Off'* in that it still controls the current through the load. This type of output is used to turn on a coil then reduce the current so that the valve will remain open, as shown in Figure 15. Since less energy is used to keep the output engaged, this type of response is very useful to improve overall system efficiency. With this output type there are associated three setpoints: **"Hold Current"**, **"Hotshot Current"** and **"Hotshot Time"** which are used to configure form of the output signal as shown in Figure 15.

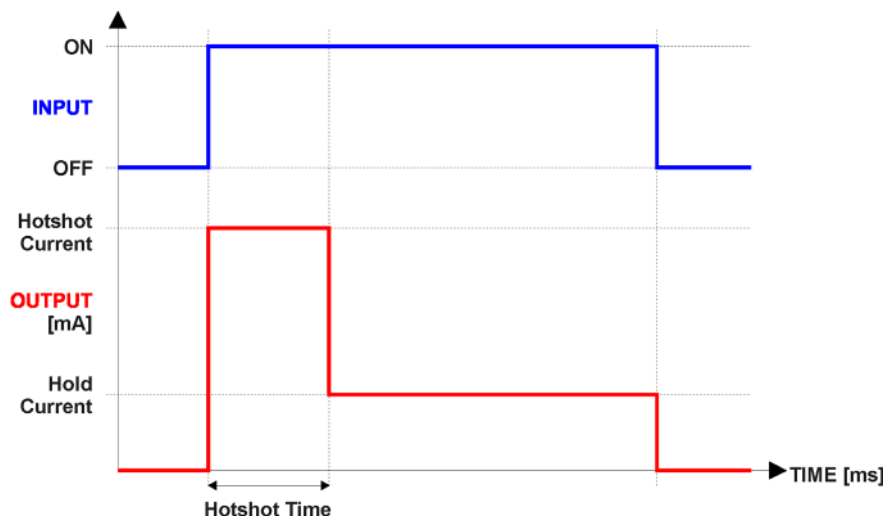


Figure 15. Hotshot Digital Profile

For Proportional output signal minimum and maximum values are configured with “**Output At Minimum Command**” and “**Output At Maximum Command**” setpoints. Value range for both is limited by selected “**Output Type**”.

Regardless of what type of control input is selected, the output will always respond in a linear fashion to changes in the input per Equation 1.

$$y = mx + a$$

$$m = \frac{Y_{max} - Y_{min}}{X_{max} - X_{min}}$$

$$a = Y_{min} - m * X_{min}$$

Equation 1. Linear Slope Calculations

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

Xmin = Control Input Minimum Ymin = “**Output at Minimum Command**”

Xmax = Control Input Maximum Ymax = “**Output at Maximum Command**”

In all cases, while X-axis has the constraint that Xmin < Xmax, there is no such limitation on the Y-axis. Thus configuring “**Output At Minimum Command**” to be greater than “**Output At Maximum Command**” allows the output to follow the control signal inversely.

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

The “**Control Source**” setpoint together with “**Control Number**” setpoint determine which signal is used to drive the output. For example, setting “**Control Source**” to ‘*Programmable Logic*’ and “**Control Number**” to ‘1’, connects output signal from Programmable Logic 1 to the output in question. Outputs respond in a linear fashion to changes in control signal. If a non-digital signal is selected to drive digital output the command state will be 0 (OFF) at or below the “**Output At Minimum Command**”, 1 (ON) at or above “**Output At Maximum Command**” and will not change in between those points.

In addition to the Control input, Proportional Outputs also support Enable and Override inputs. The “**Enable Source**” setpoint together with “**Enable Number**” setpoint determine the enable signal for the output in question. The “**Enable Response**” setpoint is used to select how output will respond to the selected Enable signal. “**Enable Response**” setpoint options are listed Table 25. If a non-digital signal is selected as Enable signal the signal is interpreted as shown below:

0	<i>Enable When On, Else Shutoff</i>
1	<i>Enable When On, Else Rampoff</i>
2	<i>Enable When Off, Else Shutoff</i>
3	<i>Enable When Off, Else Rampoff</i>
4	<i>Enable When On, Else Ramp To Min</i>
5	<i>Enable When On, Else Ramp To Max</i>

Table 25. Enable Response Options

Override input allows the output drive to be configured to go to a default value in the case of the override input being engaged/disengaged, depending on the logic selected in “**Override Response**”, presented in Table 26. When active, the output will be driven to the value in “**Output**

at **Override Command**” regardless of the value of the Control input. The “**Override Source**” and “**Override Number**” together determine the Override input signal.

0	Override When On
1	Override When Off

Table 26. Override Response Options

If a fault is detected in any of the active inputs (Control/Enable/Override) the output will respond per “**Control Fault Response**” setpoint as outlined in Table 27. Fault Value is defined by “**Output in Fault Mode**” setpoint value, which is interpreted in selected output units.

0	Shutoff Output
1	Apply Fault Value
2	Hold Last Value

Table 27. Fault Response Options

Regardless of a selected output type, current through the load is available as Output Current Feedback signal and can be selected as input for a Diagnostic Block as described in section 1.12. Another fault response that can be enabled is that a power supply over voltage or under voltage will automatically disable ALL outputs. Note: this setpoint is associated with the **Power Supply Diag** function block. Also, if the **Over Temperature Diag** function block is enabled, then a microprocessor over-temperature reading disables all the outputs until it has cooled back to the operating range.

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

In the case of an open circuit, there will be no interruption of the control for any of the outputs. The processor will continue to attempt to drive the open load.

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

1.7 PID Control Function Block

The PID Control function block is an independent logic block, but it is normally intended to be associated with proportional output control blocks described earlier. When the "Control Source" for an output has been setup as a 'PID Function Block', the command from the selected PID block drives the physical output on the 10 Output Valve Controller.

The "Target Source" and "Target Number" setpoints determine control input and the "Feedback Source" and "Feedback Number" setpoints determine the established the feedback signal to the PID function block. The "Control Response" will use the selected inputs as per the options listed in Table 28. When active, the PID algorithm will be called every "Cycle Time" in milliseconds.

0	Single Output
1	On When Over Target
2	On When Below Target

Table 28. PID Response Options

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

For example, output of a Math Function Block could be used to set the target value, in which case it would be converted to a percentage value using "Math Output Minimum Range" and "Math Output Maximum Range" setpoints. The closed-loop feedback signal could be connected a 'CAN Receive Message' and selected as the feedback source. In this case the value of the message data would be converted to a percentage based on the "Data Minimum" and "Data Maximum" setpoints in the CAN Receive block. The output of the PID function would depend on the difference between the commanded target and the measured feedback as a percentage of each signals range. In this mode, the output of the block would be a value from -100% to 100%.

The last two response options, 'On When Over Target' and 'On When Under Target', are designed to allow the user to combine the two proportional outputs as a push-pull drive for a system. Both outputs must be setup to use the same control input (linear response) and feedback signal in order to get the expected output response. In this mode, the output will be between 0% and 100%. In Order to allow the output to stabilize, the user can select a non-zero value for "Output Tolerance". If the absolute value of $Error_k$ is less than this value, $Error_k$ in the formula below will be set to zero.

The PID algorithm used is shown below, where G , K_i , T_i , K_d , T_d and $Cycle_Time$ are configurable parameters.

$$PIDOutput_k = P_k + I_k + D_k$$

$$P_k = P_Gain * Error_k$$

$$I_k = I_Gain * ErrorSum_k$$

$$D_k = D_Gain * (Error_k - Error_{k-1})$$

$$Error_k = Target - Feedback$$

$$ErrorSum_k = ErrorSum_{k-1} + Error_k$$

$$P_Gain = G$$

$$I_Gain = G * Ki * T/Ti \text{ (Note: If } Ti \text{ is zero, } I_Gain = 0)$$

$$D_Gain = G * Kd * Td/T$$

$$T = Cycle_Time * 0.001$$

Equation 2. PID Control Algorithm

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

1.8 Lookup Table Function Block

Lookup Tables are used to give output response up to 10 slopes per input. 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 described in Section 1.9.

Lookup tables have two differing modes defined by "X-Axis Type" setpoint, given in Table 29. Option '0 – Data Response' is the normal mode where block input signal is selected with the "X-Axis Source" and "X-Axis Number" setpoints and X values present directly input signal values. With option '1 – Time Response' the input signal is time and X values present time in milliseconds. And selected input signal is used as digital enable.

0	Data Response
1	Time Response

Table 29. X-Axis Type Options

The slopes are defined with (x, y) points and associated point response. X value presents input signal value and Y value corresponding Lookup Table output value. "PointN – Response" setpoint defines type of the slope from preceding point to the point in question. Response options are given in Table 30. 'Ramp To' gives a linearized slope between points, whereas 'Jump to' gives a point to point response, where any input value between XN-1 and XN will result Lookup Table output being YN. "Point0 – Response" is always 'Jump To' and cannot be edited. Choosing 'Ignored' response causes associated point and all the following points to be ignored.

0	Ignore
1	Ramp To
2	Jump To

Table 30. PointN – Response Options

The X values are limited by minimum and maximum range of the selected input source if the source is a Math Function Block. For the fore mentioned sources X-Axis data will be redefined when ranges are changed, therefore inputs should be adjusted before changing X-Axis values. For other sources Xmin and Xmax are -100000 and 1000000. The X-Axis is constraint to be in rising order, thus value of the next index is greater than or equal to preceding one. Therefore, when adjusting the X-Axis data, it is recommended that X10 is changed first, then lower indexes in descending order.

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

The Y-Axis has no constraints on the data it presents, thus inverse, decreasing, increasing or other response can be easily established. The Smallest of the Y-Axis values is used as Lookup Table output

min and the largest of the Y-Axis values is used as Lookup Table output max (i.e. used as Xmin and Xmax values in linear calculation, Section 1.6). Ignored points are not considered for min and max values.

1.9 Programmable Logic Function Block

The Programmable Logic Function Block is very powerful tool. A Programmable Logic can be linked to up to three Lookup Tables, any of which would be selected only under given conditions. Thus, output of a Programmable Logic at any given time will be the output of the Lookup Table selected by defined logic. 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.

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

The three associated tables are selected by setting "Table Number X" setpoint to desired Lookup Table number, for example selecting 1 would set Lookup Table 1 as TableX.

For each TableX there are three conditions that define the logic to select the associated Lookup Table as Logic output. Each condition implements function *Argument1 Operator Argument2* where Operator is logical operator defined by setpoint "Table X – Condition Y Operator". Setpoint options are listed in Table 31. Condition arguments are selected with "Table X – Condition Y Argument Z Source" and "Table X – Condition Y Argument Z Number" setpoints. If '0 – Control not Used' option is selected as "Table x – Condition Y Argument Z Source" the argument is interpreted as 0.

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

Table 31. Table X – Condition Y Operator Options

The three conditions are evaluated and if the result satisfies logical operation defined with "Logical Operator X" setpoint, given in Table 32, the associated Lookup Table is selected as output of the Logical block. Option '0 – Default Table' selects associated Lookup Table in all conditions.

0	Default Table (Table1)
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 32. Table X – Conditions Logical Operator Options

The three logical operations are evaluated in order and the first to satisfy gets selected, thus if Table1 logical operation is satisfied, the Lookup Table associated with Table1 gets selected regardless of two other logical operations. In addition, if none of the logical operations is satisfied the Lookup Table associated with Table1 gets selected.

1.10 Math Function Block

There are four mathematical function blocks that allow the user to define basic algorithms. A math function block can take up to six input signals. Each input is then scaled according to the associated limit and scaling setpoints.

Inputs are converted into percentage value based on the **"Input X Minimum"** and **"Input X Maximum"** values selected. For additional control the user can also adjust the **"Input X Decimal Digits"** setpoint to increase the resolution of the input data and the min and max values.

A mathematical function block includes three selectable functions, in which each implements equation A operator B, where A and B are function inputs and operator is function selected with a setpoint **"Math Function X"**. Setpoint options are presented in Table 33. The functions are connected, so that result of the preceding function goes into Input A of the next function. Thus Function 1 has both Input A and Input B selectable with setpoints, where Functions 2 to 4 have only Input B selectable. Input is selected by setting **"Function X Input Y Source"** and **"Function X Input Y Number"**. If **"Function X Input B Source"** is set to 0 *'Control not used'* signal goes through function unchanged.

$$\text{Math Block Output} = (((A1 \text{ op1 } B1) \text{ op2 } B2) \text{ op3 } B3) \text{ op4 } B4$$

0	<i>=, True when InA equals InB</i>
1	<i>!=, True when InA not equal InB</i>
2	<i>>, True when InA greater than InB</i>
3	<i>>=, True when InA greater than or equal InB</i>
4	<i><, True when InA less than InB</i>
5	<i><=, True when InA less than or equal InB</i>
6	<i>OR, True when InA or InB is True</i>
7	<i>AND, True when InA and InB are True</i>
8	<i>XOR, True when either InA or InB is True, but not both</i>
9	<i>+, Result = InA plus InB</i>
10	<i>-, Result = InA minus InB</i>
11	<i>x, Result = InA times InB</i>
12	<i>/, Result = InA divided by InB</i>
13	<i>MIN, Result = Smallest of InA and InB</i>
14	<i>MAX, Result = Largest of InA and InB</i>

Table 33. Math function X Operator Options

For logic operations (6, 7, and 8) scaled input greater than or equal to 1 is treated as TRUE. For logic operations (0 to 8), the result of the function will always be 0 (FALSE) or 1 (TRUE). 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 divider will always result in a 100% output value for the associated function. Lastly the resulting mathematical calculation, presented 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.

1.11 Set / Reset Latch Function Block

Set-Reset Block consists of only 2 control sources: **Reset Source** and **Set Source**. The purpose of these blocks is to simulate a modified latching function in which the 'Reset Signal' has more precedence. The 'latching' function works as per the Table 34 below.

'Set Signal'	'Reset Signal'	'Set-Reset Block Output' (Initial State: OFF)
OFF	OFF	Latched State
OFF	ON	OFF
ON	OFF	ON
ON	ON	OFF

Table 34. Set-Reset Function block operation

The **Reset** and **Set** sources have associated with them a minimum and maximum threshold values which determine the ON and OFF state. For the **Reset Source** are **Reset Minimum Threshold** and **Reset Maximum Threshold**. Similarly, for the **Set Source** are **Set Minimum Threshold** and **Set Maximum Threshold**. These setpoints also allow to have a dead band in between ON/OFF states and they are in terms of percentage of input selected.

As seen in Table 34 above, the 'Reset Signal' has more precedence over the 'Set Signal' - if the state of 'Reset Signal' is *ON*, the state of 'Set-Reset Block Output' will be *OFF*. To create an *ON* state in 'Set-Reset Block Output' the state of 'Reset Signal' must be *OFF* while the state of 'Set Signal' is *ON*. In this case, the state of 'Set-Reset Block Output' will remain *ON* even if 'Set Signal' turns *OFF* if 'Reset Signal' remains *OFF*. As soon as the 'Reset Signal' turns *ON* the 'Set-Reset Block Output' will turn *OFF* regardless of the state of 'Set Signal'.

1.12 Diagnostic Function Blocks

The Diagnostic Input function blocks are used to setup the diagnostic messages for the controller.

There are several types of diagnostics supported by the controller. Fault detection and reaction is associated with either input as well as the motor and output drives. In addition to I/O faults, the drive can also detect/react to power supply and lost communication events.

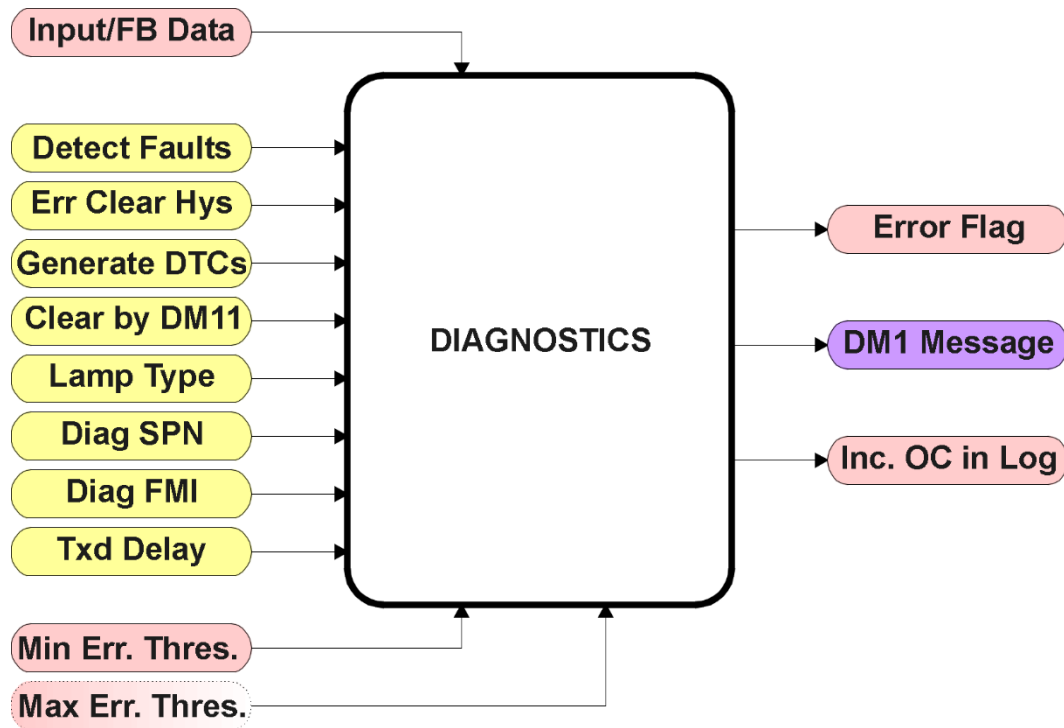


Figure 16. Function Block

The **"Event Generates a DTC in DM1"** is the most important setpoint associated with this function block, and it should be selected first. Changing it will result in other setpoints being enabled/disabled as appropriate. When disabled, all diagnostic behaviour associated with the I/O or event in question is ignored (i.e., this type of fault will not disable the motor)

In the case of a power supply error, it can be selected to disable the stepper motor when detected. By default, a power supply under or over voltage condition will shut off the motor.

Types of diagnostics supported by the controller are shown in Table 35.

Function Block	Minimum Threshold	Maximum Threshold	SPN
Phase X Winding Motor 1	Current Open Circuit Threshold	N/A	520193, 520194
Phase X Winding Motor 2	Current Open Circuit Threshold	N/A	520195, 520196
Voltage/current 1 out of range	Minimum error	Maximum error	520197 (0x7F005)
Voltage/current 2 out of range	Minimum error	Maximum error	520198 (0x7F006)
Frequency/PWM 1 out of range	Minimum error	Maximum error	520199 (0x7F007)
Frequency/PWM 2 out of range	Minimum error	Maximum error	520200 (0x7F008)
Output Open Circuit Fault	N/A	N/A	520448 (0x7F100)
Output Short Circuit Fault	N/A	N/A	520449 (0x7F101)
Over Temperature Fault	N/A	Temperature Shutdown	520450 (0x7F102)
VPS Undervoltage Fault	VPS Undervoltage	N/A	524273 (0x7FFF1)
VPS Overvoltage Fault	N/A	VPS Overvoltage	524273 (0x7FFF1)
Lost Communication Fault	N/A	Received Message Timeout	524287 (0x7FFFF)

Table 35. Diagnostic Blocks and Fault Detection Thresholds

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

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

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

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

In addition to supporting the DM1 message, the 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 controller will send the DM1 message every one second, regardless of whether there are any active faults, as recommended by the standard. While there are no active DTCs, the controller 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).



NOTE: 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.



NOTE: Since saving NVM data can affect the performance of the stepper motor, saving of the changes to the OC's in the log will only happen when the motor is not running.

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

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

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

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

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

Every Diagnostic function block also has associated with it a default FMI. The only setpoint for the user to change the FMI is “**FMI for Event used in DTC,**” even though some Diagnostic function blocks can have both high and low errors as shown in Table 36. 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 24. 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 36. Low Fault FMI versus High Fault FMI



If the FMI used is anything other than one of those in Table 24, 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.

1.13 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., motor or output control).

The "Signal Type" is the most important setpoint associated with this function block and it should be selected first. Setting it to 'Undefined' will result in other setpoints being disabled. By default, ALL receive messages are set to 'Continuous'.

Once a message has been enabled, a Lost Communication fault will be flagged if that message is not received off the bus within the "**Autoreset Time**" period. This will trigger a motor shutdown as described in Section 1.12. 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 drive on Proprietary B PGNs. However, should a PDU1 message be selected, the drive can be setup to receive it from any ECU by setting the "**PGN From Selected Address**" 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 "**Size**", "**Data Position Byte**", "**Data Position Bit**", "**Resolution**" and "**Offset**" can all be used to map any SPN supported by the J1939 standard to the output data of the Received function block. The defaults used by the drive are all for proprietary SPNs and are defined in detail in Section 3.3.

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 **"Data Min (Off Threshold)"** and **"Data Max (On Threshold)"** setpoints become the minimum and maximum values of the X-axis used in the linear calculations. As the names imply, they are also used as the On/Off thresholds for digital input types. These values are in whatever units the output data is AFTER the resolution and offset is applied to the CAN data.

The Stepper Drive supports up to 30 unique CAN Receive Messages. By default, each message is pre-configured to read a particular type of data. The details are outlined in Section 3.3, and the default list is shown in the table below.

Block #	Default Receive Data
1(16)	Command Motor Mode Motor 1(2)
2(17)	Command Direction Motor 1(2)
3(18)	Command Step Rate Motor 1(2)
4(19)	Command Max. Phase Current Motor 1(2)
5(20)	Command Number of Steps Motor 1(2)
6(21)	Set Step Type Motor 1(2)
7(22)	Set Percent Fast Decay Motor 1(2)
8(23)	Set Acceleration Table Motor 1(2)
9(24)	Put Drive to Sleep Motor 1(2)
10(25)	Reset Motor Drive Motor 1(2)
11(26)	E-Stop Motor Drive Motor 1(2)
12(27)	Save Data (Send Once) Motor 1(2)
13	Universal Output Control
14(29)	PID Target Input Data Motor 1(2)
15(30)	PID Feedback Input Data Motor 1(2)

Table 37. Default CAN Receive Messages

1.14 CAN Transmit Function Block

The CAN Transmit function block is used to send any output from another function block (i.e., input, CAN receive) to the J1939 network. The AX029000 ECU has six CAN Transmit Messages, and each message has four completely user defined signals.

1.14.1 CAN Transmit Message Setpoints

Each CAN Transmit Message setpoint group includes setpoints that effect the whole message and are thus mutual for all signals of the message. These setpoints are presented in this section. The setpoints that configure an individual signal are presented in next section.

The **"PGN"** setpoint sets PGN used with the message. **User should be familiar with the SAE J1939 standard and select values for PGN/SPN combinations as appropriate from section J1939/71.**

"Transfer Rate" setpoint defines the interval used to send the message to the J1939 network. If the **"Transfer Rate"** is set to zero, the message is disabled unless it shares its PGN with another message. In case of a shared PGN repetition rate of the LOWEST numbered message are used to send the message 'bundle'.

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 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.



NOTE: At power up, transmitted message will not be broadcasted until after a 1 second delay. This is done to prevent any power up or initialization conditions from creating problems.

Since the defaults are PropB messages, the **"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.

1.14.2 CAN Transmit Signal Setpoints

Each CAN transmit message has 10 associated signals, which define data inside the Transmit message. **"Signal X Source"** setpoint together with **"Signal X Number"** setpoint define the signal source of the message. **"Signal X Source"** and **"Signal X Number"** options are listed in Table 39. Setting **"Signal X Source"** to *Control Not Used* disables the signal.

"Signal X Size" setpoint determines how many bits signal reserves from the message. **"Signal X Byte Position"** determines in which of 8 bytes of the CAN message LSB of the signal is located. Similarly, **"Signal X Bit Position"** 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 Resolution” setpoint determines the scaling done on the signal data before it is sent to the bus. “Signal X 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.

Note: $\text{CAN Data} = (\text{Input Data} - \text{Offset}) / \text{Resolution}$

The Stepper Drive supports up to 10 unique CAN Transmit Messages, all of which can be programmed to send any available data to the CAN network. By default, each message is pre-configured to send a particular type of data. The details are outlined in Section 0.

1.15 Available Control/Input Sources

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

Below is the full list of the programmable inputs supported by the controller:

Function Block	Input Name
Stepper Drive Controls	Motor Mode
	Direction
	Step Rate
	Max. Phase Current
	Number of Steps
	Step Type
	Percent Fast Decay
	Acceleration Table
	Put Drive to Sleep
	Reset Motor Drive
	E-Stop Motor Drive
	Save Data to Defaults
PID Control Function	PID Target Command
	PID Feedback Input
Universal Output	Control, Enable, Override
CAN Transmit	Transmit Data
Lookup Table	X-Axis Source
Programmable Logic	Table X
Math Functions	Inputs
Set-Reset Latch	Set Sources, Reset Sources

Table 38. Programmable Inputs

For each of the inputs listed in Table 38, the options for the Source and Number are listed below. Note, not all options would make sense for any input, and it is up to the user to program the controller in a logical and functional fashion.

Sources	Range	Notes
Control Source Not Used	N/A	When this is selected, it disables all other setpoints associated with the input in question.
Received CAN J1939 Message	1 to 30	The CAN Receive Messages each have a unique default assigned to them as outlined in Section 1.13
Input Measured Value	1 or 4	1 = Voltage/Current Input 1 2 = Frequency/PWM Input 1 3 = Voltage/Current Input 2 4 = Frequency/PWM Input 2
Output Target Value	1	Can be mapped to a CAN Transmit Message.
Averaged UOut/Stepper Phase Current	1 or 5	Intended for error detection feedback only. Not a reflection of the true RMS value. Can be mapped to a CAN Transmit Message
Stepper Variable Feedback	1 to 24	1(13) = Save Operating Data Motor 1(2) 2(14) = Put Drive to Sleep Motor 1(2) 3(15) = Reset Motor Motor 1(2) 4(16) = E-Stop Motor Motor 1(2) 5(17) = Motor Drive Mode Motor 1(2) 6(18) = Drive Step Type Motor 1(2) 7(19) = Rotation Direction Motor 1(2) 8(20) = Selected Acc. Table Motor 1(2) 9(21) = Percent Fast Decay Motor 1(2) 10(22) = Maximum Phase Current Motor 1(2) 11(23) = Number of Steps Motor 1(2) 12(24) = Step Rate Motor 1(2)
Encoder Input	1 or 2	1 = Counter 2 = Direction
PID Motor Function Block Control	1 to 3	1 = PID Commanded Motor Mode 2 = PID Commanded Direction 3 = PID Commanded Step Rate
PID UOut Function Block Control	1 to 2	1 = PID Setpoint Value 2 = PID Target Value
Lookup Table	1 to 12	User must enable the function block, as it is disabled by default.
Math Function Block	1 to 6	User must enable the function block, as it is disabled by default.
Programmable Logic Block	1 to 4	User must enable the function block, as it is disabled by default.
Set/Reset Latch	1 to 5	User must enable the function block, as it is disabled by default.
Constant Continuous Data	1 to 15	
Constant Discrete Data	1 to 5	
Supply Voltage	N/A	Input to Power Supply Diagnostics Can be mapped to a CAN Transmit Message
Microcontroller Temperature	N/A	Input to Temperature Diagnostics Can be mapped to a CAN Transmit Message

Table 39. Input Sources and Numbers

If a non-digital signal is selected to drive a digital input, the signal is interpreted to be OFF at or below the minimum of selected source and ON at or above the maximum of the selected source, and it will not change in between those points. Thus, analog to digital interpretation has a built-in hysteresis defined by minimum and maximum of the selected source, as shown in Figure 17. For example, CAN Receive message signal is interpreted to be ON at or above "Maximum Range" and OFF at or below "Minimum Range".

Control Constant Data has no unit nor minimum and maximum assigned to it, thus user must assign appropriate constant values according to intended use.

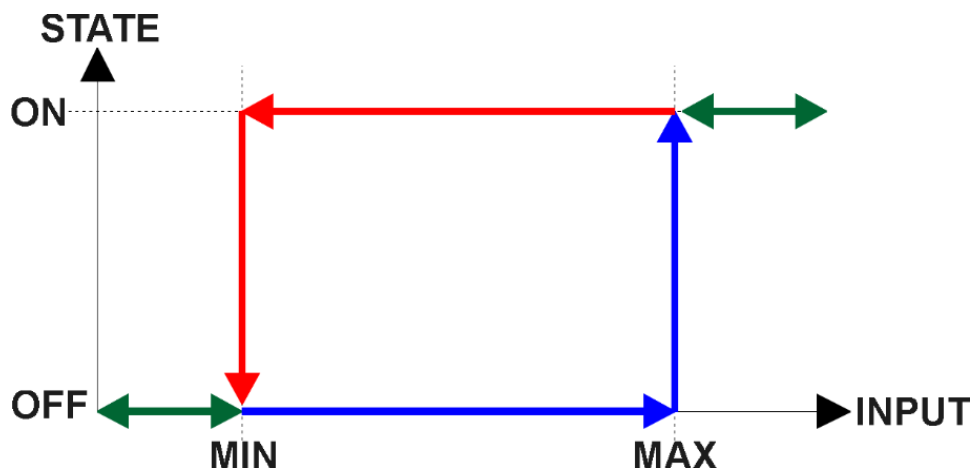


Figure 17. Analog source to Digital input

2 INSTALLATION INSTRUCTIONS

2.1 Dimensions and Pinout

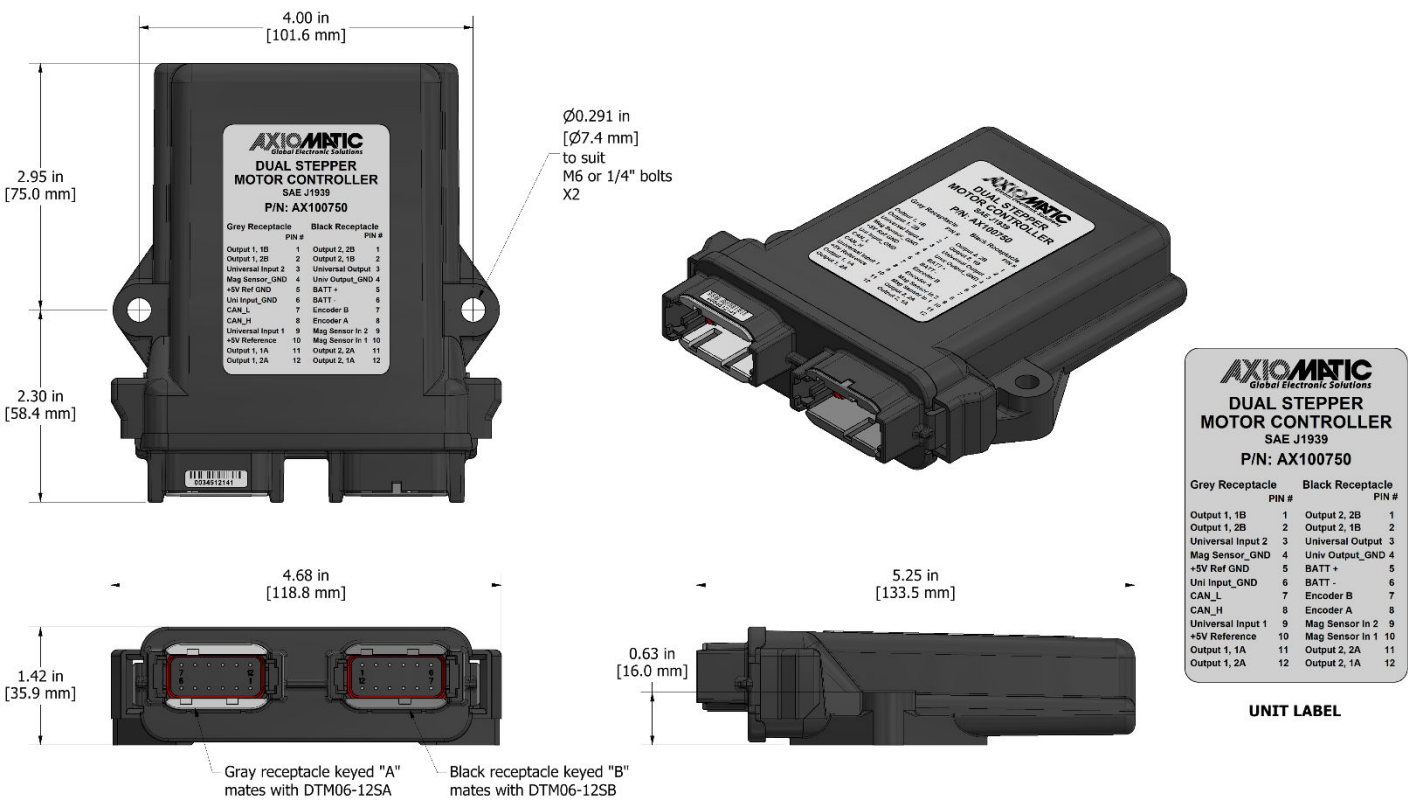


Figure 18. Dimension Drawing

Grey Receptacle		Black Receptacle	
	PIN #		PIN #
Output 1, 1B	1	Output 2, 2B	1
Output 1, 2B	2	Output 2, 1B	2
Universal Input 2	3	Universal Output 3	3
Mag Sensor_GND	4	Univ Output_GND 4	4
+5V Ref GND	5	BATT +	5
Uni Input_GND	6	BATT -	6
CAN_L	7	Encoder B	7
CAN_H	8	Encoder A	8
Universal Input 1	9	Mag Sensor In 2	9
+5V Reference	10	Mag Sensor In 1	10
Output 1, 1A	11	Output 2, 2A	11
Output 1, 2A	12	Output 2, 1A	12

Figure 19. 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 Receive PGN and SPN Parameters
- Configurable Transmit PGN and SPN Parameters
- Sending DM1 Diagnostic Message Parameters
- 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 Motor 1 Command Message	65280 (\$00FF00)
PropB Receive, Default Motor 1 Setup and Control Message	65296 (\$00FF10)
PropB Receive, Default Motor 2 Command Message	65312 (\$00FF20)
PropB Receive, Default Motor 2 Setup and Control Message	65328 (\$00FF30)
PropB Receive, Default Motor 1 PID Target and Feedback Messages	65408 (\$00FF80)
PropB Receive, Default Motor 2 PID Target and Feedback Messages	65440 (\$00FFA0)
PropB Transmit, Default Motor 1 Feedback Message	65360 (\$00FF50)
PropB Transmit, Default I/O Feedback Message 1	65365 (\$00FF55)
PropB Transmit, Default Motor 2 Feedback Message	65392 (\$00FF70)
PropB Transmit, Default I/O Feedback Message 2	65397 (\$00FF75)

Note: Any Proprietary B PGN in the range 65280 to 65535 (\$00FF00 to \$00FFFF) can be selected

Note: The Proprietary A PGN 61184 (\$00EF00) can also be selected for any of the messages

From J1939-73 - Diagnostics

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

From J1939-81 - Network Management

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

From J1939-71 – Vehicle Application Layer

Software Identification	65242 (\$00FEDA)
-------------------------	------------------

None of the application layer PGNs is 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

The NAME, Address and Software ID have the following default configurations.

J1939 NAME

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

Arbitrary Address Capable	Yes
Industry Group	0, Global
Vehicle System Instance	0
Vehicle System	0, Non-specific system
Function	132, Axiomatic Motor Control
Function Instance	16, Axiomatic Dual Stepper Motor Drive
ECU Instance	0, First Instance
Manufacture Code	162, Axiomatic Technologies Corporation
Identity Number	Variable, based on ECU Serial Number

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

ECU Address

The default value of this setpoint is 200 (0xC8), which is a Motor Controller preferred address as set by the SAE in J1939 tables B3 to B7. The Axiomatic EA supports the selection of any address between 0 to 253, and *it is the user's responsibility to select an address that complies with the standard*. The user must also be aware that since the unit is arbitrary address capable, if another ECU with a higher priority NAME contends for the selected address, the controller will continue select the next highest address until it finds one that it can claim. See J1939/81 for more details about address claiming.

Software Identifier

PGN 65242

Software Identification

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

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

For the controller ECU, Byte 1 is set to 5, and the identification fields are as follows

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

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

The screenshot shows the 'Electronic Assistant' window. On the left, a tree view shows the 'J1939 CAN Network' with 'ECU AX100750, Dual Stepper Motor Drive' selected. Under this, 'General ECU Information' is highlighted. The main pane displays a table of parameters and their values:

Parameter	Value	Description
ECU Part Number	AX100750	
ECU Serial Number	0000121001	
ECU J1939 NAME		PGN 60928. 64-bit ECU Identifier sent in Address Claimed Messages
Arbitrary Address Capable	0X01	Yes
Industry Group	0X00	Global
Vehicle System Instance	0X00	
Vehicle System	0X00	Non-specific system
Reserved	0X00	
Function	0X84	Axiomatic Motor Controller
Function Instance	0X10	
ECU Instance	0X00	#1 - First Instance
Manufacturer Code	0X0A2	Axiomatic Technologies
Identity Number	0X1A8AA4	Unique ECU network ID number
ECU Address	0X80	Reserved for future assignment by SAE, but available for use by self configurable ECUs
ECU ID		PGN 64965 -ECUID
ECU Part Number	AX100750	
ECU Serial Number	0000121001	
ECU Type	Controller	
ECU Manufacturer Name	Axiomatic	
Software ID		PGN 65242 -SOFT
Field #1	AX100750, Dual Stepper Motor, Proj_20006	
Field #2	Firmware: V1.00, August 2021	

The status bar at the bottom indicates 'Ready' and '250 kbit/s'.

3.3 CAN Receive Message Defaults

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

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

The “Motor Control Messages” has the following default configuration.

PGN 65280(65312)		Motor 1(2) Command Messages	
Transmission Repetition:	300ms (default, configurable)		
Data Length:	8		
Data Page:	0		
PDU Format:	254		
PDU Specific:	GE	PGN Supporting Information:	
Default Priority:	6		
Parameter Group Number:	65280(65312) / 0xFF00(0xFF20)		
Start Position	Length	Parameter Name	SPN
1	1 byte	Command Motor Mode	N/A
2	1 byte	Command Direction	N/A
3-4	2 bytes	Command Step Rate	N/A
5-6	2 bytes	Command Max. Phase Current	N/A
7-8	2 bytes	Command Number of Steps	N/A

The “Motor Setup and Control Message” has the following default configuration.

PGN 65296(65328)		Motor 1(2) Setup and Control Messages	
Transmission Repetition:	0s (only sent once, or on request by default, configurable)		
Data Length:	8		
Data Page:	0		
PDU Format:	254		
PDU Specific:	GE	PGN Supporting Information:	
Default Priority:	6		
Parameter Group Number:	65296(65328) / 0xFF10(0xFF30)		
Start Position	Length	Parameter Name	SPN
1	1 byte	Set Step Type	N/A
2	1 byte	Set Percent Fast Decay	N/A
3	1 byte	Set Acceleration Table	N/A
4	1 byte	Put Drive to Sleep	N/A
5	1 byte	Reset Motor Drive	N/A
6	1 byte	E-Step Motor Signal	N/A
7	1 byte	Save Data (Send Once)	N/A
8	1 byte	Universal Output Control	N/A

Command Motor Mode

Default value used when a '*Received CAN J1939 Message*' is selected as the control source for the motor mode.

00	Motor Disabled
01	Motor Stopped
02	Motor Running

Data Length: 1 byte
Resolution: 1 mode/bit, 0 offset
Data Range: 0 to 2 (Note: valid data >2 will be truncated to 2)
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65280(65312)

Command Direction

Default value used when a '*Received CAN J1939 Message*' is selected the control source for the direction of motor rotation.

00	Counterclockwise
01	Clockwise

Data Length: 1 byte
Resolution: 1 direction/bit, 0 offset
Data Range: 0 or 1 (Note: valid data >1 will be truncated to 1)
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65280(65312)

Command Step Rate

Default value used when a '*Received CAN J1939 Message*' is selected the control source for the step rate applied to the stepper motor.

Data Length: 2 bytes
Resolution: 1 [Step Unit]/bit, 0 offset
Data Range: 0 to 64255 [Step Unit], default 1 to 50
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65280(65312)

Command Max. Phase Current

Default value used when a '*Received CAN J1939 Message*' is selected the control source for the maximum current setting applied to the driver chip.

Data Length: 2 bytes
Resolution: 1 mA/bit, 0 offset
Data Range: 0 to 64255 [mA], default 100 to 2000
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65280(65312)

Command Number of Steps

Default value used when a 'Received CAN J1939 Message' is selected the control source for the number of steps the motor will run before holding or stopping.

Data Length: 2 bytes
Resolution: 1 step/bit, 0 offset
Data Range: 0 to 64255 [Step], default 0 to 1000
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65280(65312)

Set Step Type

Default value used when a 'Received CAN J1939 Message' is selected the control source for the selection of the micro-stepping type applied at the phase current outputs (see Figure 5, Figure 6).

00 Full Step
01 Half Step
02 Quarter Step
03 Sixteenth Step

Data Length: 1 byte
Resolution: 1 type/bit, 0 offset
Data Range: 0 to 3 (Note: valid data >3 will be truncated to 3)
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65296(65328)

Set Percent Fast Decay

Default value used when a 'Received CAN J1939 Message' is selected the control source for the PFD used by the driver chip, A3979.

00 Slow
01 Fast
02 Mixed

Data Length: 1 byte
Resolution: 1 PFD/bit, 0 offset
Data Range: 0 to 2 (Note: valid data >2 will be truncated to 2)
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65296(65328)

Set Acceleration Table

Default value used when a 'Received CAN J1939 Message' is selected the control source for the selection of the acceleration/deceleration profile used when the motor starts or stops.

00 No Profile
0X Profile X, where X = 1 to 5

Data Length: 1 byte
Resolution: 1 table/bit, 0 offset
Data Range: 0 to 5 (Note: valid data >5 will be truncated to 5)
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65296(65328)

Put Drive to Sleep

Default value used when a 'Received CAN J1939 Message' is selected the control source for putting the driver chip, A3979, to sleep

00	False	(Drive not asleep)
01	True	(Drive in sleep mode)

Data Length:	1 byte
Resolution:	1 state/bit, 0 offset
Data Range:	0 or 1 (Note: valid data >1 will be truncated to 1)
Type:	Input
Suspect Parameter Number:	N/A
Parameter Group Number:	65296(65328)

Reset Motor Drive

Default value used when a 'Received CAN J1939 Message' is selected the control source for resetting the driver chip, A3979, to the Home position when micro stepping

00	False	(Drive not reset)
01	True	(Drive reset to home position)

Data Length:	1 byte
Resolution:	1 state/bit, 0 offset
Data Range:	0 or 1 (Note: valid data >1 will be truncated to 1)
Type:	Input
Suspect Parameter Number:	N/A
Parameter Group Number:	65296(65328)

E-Stop Motor Signal

Default value used when a 'Received CAN J1939 Message' is selected the control source for emergency stopping the motor

00	False	(Drive not in E-Stop)
01	True	(E-Stop motor)

Data Length:	1 byte
Resolution:	1 state/bit, 0 offset
Data Range:	0 or 1 (Note: valid data >1 will be truncated to 1)
Type:	Input
Suspect Parameter Number:	N/A
Parameter Group Number:	65296(65328)

Save Data (Sent Once)

Default value used when a 'Received CAN J1939 Message' is selected the control source for saving the current settings of the motor variable to non-volatile memory to be used as the power-up and reset defaults. The flag to save is only set on the rising edge of the signal, i.e. a transition from False to True. The actual save operation will only take place once the motor is not running.

00	False	(Do not save current settings)
01	True	(Save current settings)

Data Length:	1 byte
Resolution:	1 state/bit, 0 offset
Data Range:	0 or 1 (Note: valid data >1 will be truncated to 1)
Type:	Input
Suspect Parameter Number:	N/A
Parameter Group Number:	65296(65328)

Universal Output Control

Default value used when a '*Received CAN J1939 Message*' is selected the control source for the universal output on the motor drive.

Data Length: 1 byte
Resolution: 1 %/bit, 0 offset
Data Range: 0 to 250, default 0% to 100%
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65296

The "PID Target and Feedback Messages" has the following default configuration.

PGN 65408(65440)		PID Target and Feedback Messages Motor 1(2)	
Transmission Repetition:	0s (default, configurable)		
Data Length:	8		
Data Page:	0		
PDU Format:	254		
PDU Specific:	GE	PGN Supporting Information:	
Default Priority:	6		
Parameter Group Number:	65408(65440) / 0xFF80(0xFFA0)		
Start Position	Length	Parameter Name	SPN
1	1 byte	PID Target Input Data	N/A
2	1 byte	PID Feedback Input Data	N/A

PID Target Input Data

Default value used when a '*Received CAN J1939 Message*' is selected the control source for the PID control function's target value.

Data Length: 1 byte
Resolution: 0.4 %/bit, 0 offset
Data Range: 0 to 250, default 0% to 100%
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65408(65440)

PID Feedback Input Data

Default value used when a '*Received CAN J1939 Message*' is selected the control source for the PID control function's feedback value.

Data Length: 1 byte
Resolution: 0.4 %/bit, 0 offset
Data Range: 0 to 250, default 0% to 100%
Type: Input
Suspect Parameter Number: N/A
Parameter Group Number: 65408(65440)

3.4 CAN Transmit Message Defaults

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

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

The “Motor Feedback Message 1” (CAN Output Message #1) has the following default configuration.

PGN 65360		Motor 1 Feedback Message	
Transmission Repetition:	1s (default, configurable)		
Data Length:	8		
Data Page :	0		
PDU Format :	254		
PDU Specific:	GE	PGN Supporting Information:	
Default Priority:	6		
Parameter Group Number:	65360 / 0xFF50		
Start Position	Length	Parameter Name	SPN
1-2	2 bytes	Averaged Stepper Phase 1 Current	520193 (\$7F001)
3-4	2 bytes	Averaged Stepper Phase 2 Current	520194 (\$7F002)
5	1 byte	Stepper Feedback, Motor Drive Mode	N/A
6	1 byte	Stepper Feedback, Rotation Direction	N/A
7-8	2 bytes	Stepper Feedback, Step Rate	N/A

The “Motor Feedback Message 2” (CAN Output Message #3) has the following default configuration.

PGN 65392		Motor 2 Feedback Message	
Transmission Repetition:	1s (default, configurable)		
Data Length:	8		
Data Page :	0		
PDU Format :	254		
PDU Specific:	GE	PGN Supporting Information:	
Default Priority:	6		
Parameter Group Number:	65392 / 0xFF70		
Start Position	Length	Parameter Name	SPN
1-2	2 bytes	Averaged Stepper Phase 1 Current	520195 (\$7F003)
3-4	2 bytes	Averaged Stepper Phase 2 Current	520196 (\$7F004)
5	1 byte	Stepper Feedback, Motor Drive Mode	N/A
6	1 byte	Stepper Feedback, Rotation Direction	N/A
7-8	2 bytes	Stepper Feedback, Step Rate	N/A

The "I/O Feedback Message 1" (CAN Output Message #2) has the following default configuration.

PGN 65365		I/O Feedback Message 1	
Transmission Repetition:	1s (default, configurable)		
Data Length:	8		
Data Page :	0		
PDU Format :	254		
PDU Specific:	GE	PGN Supporting Information:	
Default Priority:	6		
Parameter Group Number:	65365 / 0xFF55		
Start Position	Length	Parameter Name	SPN
1-2	2 byte	Input Measured Value, Voltage/Current 1	520197 (\$7F005)
3-4	2 byte	Input Measured Value, Frequency/PWM 1	520199 (\$7F007)
5-6	2 byte	Input Measured Value, Voltage/Current 2	520198 (\$7F006)
7-8	2 byte	Input Measured Value, Frequency/PWM 2	520200 (\$7F008)

The "I/O Feedback Message 2" (CAN Output Message #4) has the following default configuration.

PGN 65397		I/O Feedback Message 2	
Transmission Repetition:	1s (default, configurable)		
Data Length:	8		
Data Page :	0		
PDU Format :	254		
PDU Specific:	GE	PGN Supporting Information:	
Default Priority:	6		
Parameter Group Number:	65397 / 0xFF75		
Start Position	Length	Parameter Name	SPN
1-2	2 byte	Output Target Value	N/A
3-4	2 byte	Encoder Input Counter	N/A
5	1 byte	Encoder Input Direction	N/A
6-7	2 byte	Power Supply Measured	524273 (\$7FFF1)
8	1 byte	Temperature Measured	520450 (\$7F102)

Averaged Stepper Phase X Current, where X = 1 or 2 for Motor 1(2)

Reflects the averaged current measured from the current sense resistor on the corresponding phase. This is NOT a reflection of the true RMS current value, simply a feedback signal that can be used for diagnostic purposes to detect an open-circuit on the phase.

Data Length: 2 bytes
Resolution: 1 mA/bit, 0 offset
Type: Measured
Suspect Parameter Number: 520193(520195) - Phase 1 or 520194(520196) - Phase 2
Parameter Group Number: 65360(65392)

Stepper Feedback, Motor Drive Mode

Reflects the actual mode the stepper motor is operating in at the time. Typically, this signal will reflect the commanded mode, but it can also reflect when the motor is Stepping or Holding, or if it has been disabled because of an E-Stop signal or diagnostics.

00	Motor Disabled
01	Motor Stopped
02	Motor Running
03	Motor Stepping
04	Motor Holding

Data Length: 1 byte
Resolution: 1 mode/bit, 0 offset
Type: Status
Suspect Parameter Number: N/A
Parameter Group Number: 65360(65392)

Stepper Feedback, Rotation Direction

Reflects the direction of rotation of the motor. This feedback is primarily intended for use in conjunction with in the PID Control method of driving the motor. In CAN commanded mode, this byte will simply reflect the commanded direction that has been set.

00	Counter clockwise
01	Clockwise

Data Length: 1 byte
Resolution: 1 direction/bit, 0 offset
Type: Status
Suspect Parameter Number: N/A
Parameter Group Number: 65360(65392)

Stepper Feedback, Step Rate

Reflects the step rate of the motor. This feedback is primarily intended for use in conjunction with in the PID Control method of driving the motor. In CAN commanded mode, this byte will simply reflect the commanded step rate that has been set. The data will be in "**Step Rate Data Unit**"

Data Length: 2 bytes
Resolution: 1 [Step Unit]/bit, 0 offset
Type: Calculated
Suspect Parameter Number: N/A
Parameter Group Number: 65360(65392)

Input Measured Value, Voltage/Current 1,2

This value reflects the measured value of the Analog Input

Data Length: 2 bytes
Resolution: 0.001 [Unit]/bit, 0 offset
Data Range: 0 to 10V (10,000mV) or 0 to 20mA (20,000uA)
Type: Measured
Suspect Parameter Number: 520197,520198 (0x7F005, 0x7F006) - a proprietary SPN
Parameter Group Number: 65365

Input Measured Value, Frequency/PWM 1,2

This value reflects the measured value of the Frequency Input

Data Length: 2 bytes
Resolution: 0.1 [Unit]/bit, 0 offset
Data Range: 0 to 10,000Hz (max 100,000) or 0 to 100% (max 1,000)
Type: Measured
Suspect Parameter Number: 520199,520200 (0x7F007, 0x7F008) - a proprietary SPN
Parameter Group Number: 65365

Output Target Value

Data Length: 2 bytes
Resolution: 0.001 [Unit]/bit, 0 offset
Data Range: 0 to 10V (10,000mV)
Type: Measured
Suspect Parameter Number: N/A
Parameter Group Number: 65397

Encoder Feedback, Step Counter

Reflects the step counter of the encoder.

Data Length: 2 bytes
Resolution: 1 Step/bit, 0 offset
Type: Calculated
Suspect Parameter Number: N/A
Parameter Group Number: 65397

Encoder Feedback, Rotation Direction

Reflects the direction of rotation of the encoder.

00 Counter clockwise
01 Clockwise

Data Length: 1 byte
Resolution: 1 direction/bit, 0 offset
Type: Status
Suspect Parameter Number: N/A
Parameter Group Number: 65397

Power Supply Measured

This value reflects the measured supply voltage powering the controller.

Data Length: 2 bytes
Resolution: 0.01 V/bit, 0 offset
Data Range: 0 to 642.55 V
Type: Measured
Suspect Parameter Number: 524273 (0x7FFF1, a proprietary SPN)
Parameter Group Number: 65397

Temperature Measured

This value reflects the measured temperature of the controller.

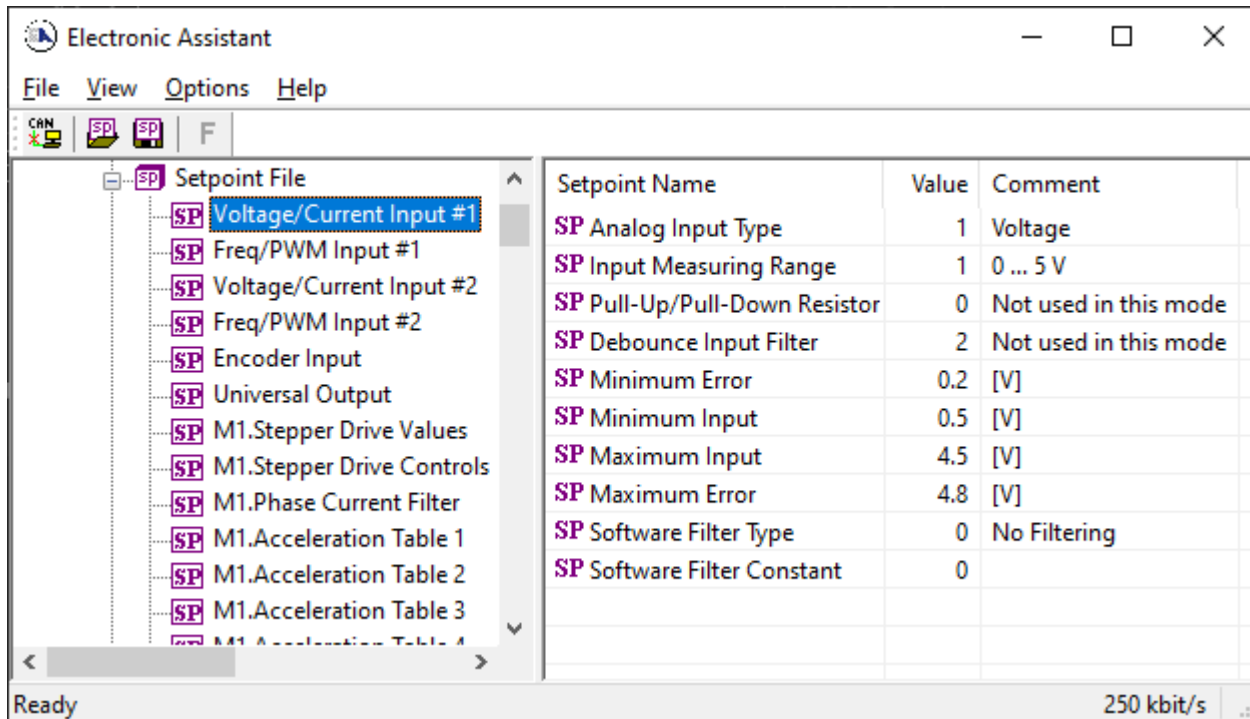
Data Length:	1 byte	
Resolution:	1 C/bit, 0 offset	
Data Range:	-125C to 125 V	
Type:	Measured	
Suspect Parameter Number:	520450	(0x7F102, a proprietary SPN)
Parameter Group Number:	65397	

4 ECU SETPOINTS ACCESSED WITH THE AXIOMATIC ELECTRONIC ASSISTANT

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

4.1 Voltage/Current Input Setpoints

The Input function blocks are defined in Section 0. Please refer there for detailed information about how all these setpoints are used.



Name	Range	Default	Notes
Analog Input Type	Drop List	1, Voltage	See Table 10
Input Measuring Range	Per Type	2, 0 to 5V	See Table 13, Table 14
Pullup/Pulldown Resistor	Drop List	0, Pullup/down Off	See Table 12
Debounce Input Filter	Drop List	2, 1.78us	See Table 17, Only used with "Digital" input type
Minimum Error	to Minimum Range	Per Type and Range	Default values for these setpoints are dependent on Type and Range. These values are automatically updated by the ECU when the Type/Range setpoints are changed. See Table 40 for the complete list.
Minimum Range	Minimum Error to Maximum Range	Per Type and Range	
Maximum Range	Minimum Range to Maximum Error	Per Type and Range	
Maximum Error	Maximum Range to Limit (see notes)	Per Type and Range	
Software Filter Type	Drop List	2, Repeating Average	See Table 40
Software Filter Constant	1 to 1,000	100	Used with Moving or Repeating Average Filters
Hysteresis to Clear Fault	0.01 to 50.00	Per Type and Range	See Table 27
Event Generates a DTC in DM1	False or True	True	See Section 1.12
Event Cleared Only by DM11	False or True	False	See Section 1.12
Lamp Set by Event in DM1	Drop List	1, Amber, Warning	See Section 1.12
SPN for Event used in DTC	1 to 524287	520208 (\$7F010)	See Section 1.12
FMI for Event used in DTC	Drop List	Per Type	See Section 1.12 (low FMI)

Delay Before Sending DM1	0 to 60,000 ms	1000 ms	See Section 1.12
--------------------------	----------------	---------	------------------

Type	Range	Minimum Error	Minimum Range	Maximum Range	Maximum Error*	Hysteresis
Voltage	0-2.5V	0.10 [V]	0.25 [V]	2.50 [V]	2.75 [V]	0.050 [V]
	0-5V	0.20 [V]	0.50 [V]	4.50 [V]	4.80 [V]	0.100 [V]
	0-10V	0.20 [V]	0.50 [V]	9.50 [V]	9.80 [V]	0.250 [V]
Current	0-20mA	0.00 [mA]	0.00 [mA]	20.00 [mA]	22.00 [mA]	0.250 [mA]
	4-20mA	1.00 [mA]	4.00 [mA]	20.00 [mA]	22.00 [mA]	0.250 [mA]
Frequency or RPM	RPM	50 [RPM]	100 [RPM]	2500 [RPM]	2600 [RPM]	10 [RPM]
	0.5 to 50 Hz	0.25 [Hz]	1.00 [Hz]	50.00 [Hz]	55.00 [Hz]	0.200 [Hz]
	10Hz to 1kHz	7.5 [Hz]	10.0 [Hz]	1000.0 [Hz]	1010.0 [Hz]	5.0 [Hz]
	100Hz to 10kHz	75 [Hz]	100 [Hz]	10000 [Hz]	10100 [Hz]	10 [Hz]
PWM	Low Freq [<1kHz]	1.00 [%]	5.00 [%]	95.00 [%]	99.00 [%]	1.00 [%]
	High Freq [>100Hz]	1.00 [%]	5.00 [%]	95.00 [%]	99.00 [%]	1.00 [%]
Digital	N/A	N/A	0.00	1.00	N/A	N/A

*The limit for the "Maximum Error" setpoint will be the upper end of the Range + 10%

Table 40. Input Setpoint Defaults Based on Type and Range

4.2 Frequency/PWM Input Setpoints

The Input function blocks are defined in Section 0. Please refer there for detailed information about how all these setpoints are used.

The screenshot shows the 'Electronic Assistant' software window. On the left, a tree view under 'Setpoint File' lists various input types, with 'Freq/PWM Input #1' selected. On the right, a table displays the configuration for this selected setpoint.

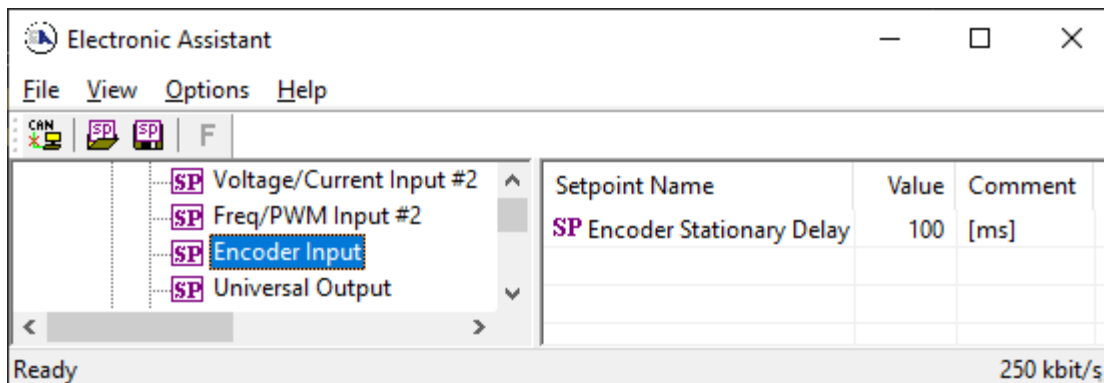
Setpoint Name	Value	Comment
SP Frequency Input Type	1	Inductive Pickup
SP Input Measuring Range	0	0.5Hz ... 50Hz
SP Pulses per Revolution	0	
SP Pull-Up/Pull-Down Resistor	0	Pullup/down Off
SP Debounce Input Filter	2	1.78 us
SP Minimum Error	0.25	[Hz]
SP Minimum Input	1	[Hz]
SP Maximum Input	50	[Hz]
SP Maximum Error	55	[Hz]

The status bar at the bottom indicates 'Ready' and a data rate of '250 kbit/s'.

Name	Range	Default	Notes
Frequency Input Type	Drop List	1, Inductive Pickup	See Table 11
Input Measuring Range	Per Type	0, 0.5Hz to 50Hz	See Table 15, Table 16
Pulse per Revolution	0 to 1000	0	Data measured in Hertz
Pullup/Pulldown Resistor	Drop List	0, Pullup/down Off	See Table 12
Debounce Time	Drop List	2, 1.78us	See Table 17, Only used with "Digital" input type
Minimum Error	0 to Minimum Range	Per Type and Range	Default values for these setpoints are dependent on Type and Range. These values are automatically updated by the ECU when the Type/Range setpoints are changed. See Table 40 for the complete list.
Minimum Range	Minimum Error to Maximum Range	Per Type and Range	
Maximum Range	Minimum Range to Maximum Error	Per Type and Range	
Maximum Error	Maximum Range to Limit (see notes)	Per Type and Range	
Fault Detection is Enabled	False or True	True	See Section 1.12
Hysteresis to Clear Fault	0.01 to 50.00	Per Type and Range	See Table 40
Event Generates a DTC in DM1	False or True	True	See Section 1.12
Event Cleared Only by DM11	False or True	False	See Section 1.12
Lamp Set by Event in DM1	Drop List	1, Amber, Warning	See Section 1.12
SPN for Event used in DTC	1 to 524287	520448 (\$7F100)	See Section 1.12
FMI for Event used in DTC	Drop List	Per Type	See Section 1.12 (low FMI)
Delay Before Sending DM1	0 to 60,000 ms	1000 ms	See Section 1.12

4.3 Encoder Input Setpoints

The Encoder Input setpoints are defined in Section 1.4. Refer to that section for detailed information on how these setpoints are used. The screen capture displays the available setpoints for the Encoder Input. The table below highlights the allowable ranges for each setpoint.



Name	Range	Default	Notes
Encoder Stationary Delay	0...60000	100	Refer to section 1.4.1

4.4 Universal Output Setpoints

The Universal Output Function Block is defined in Section 1.6. Please refer there for detailed information about how these setpoints are used. Outputs are disabled by default. In order to enable an output “**Output Type**” and “**Control Source**” must be chosen.

The screenshot shows the 'Electronic Assistant' software window. On the left, a tree view under 'Setpoint File' lists various configuration items, with 'Universal Output' selected and highlighted in blue. On the right, a table displays the configuration for the 'Universal Output' setpoint.

Setpoint Name	Value	Comment
SP Output type	1	Current
SP Output At Minimum Command	300	[mA]
SP Output At Maximum Command	1500	[mA]
SP Output At Override Command	750	[mA]
SP Dither Frequency	150	[Hz]
SP Dither Amplitude	0	[mA]
SP Ramp Up	1000	[ms]
SP Ramp Down	1000	[ms]
SP Output Frequency	25000	[Hz]
SP Hold Current	500	[mA]
SP Hotshot Current	2000	[mA]
SP Hotshot Time	1000	[ms]
SP Digital Response	0	Normal Logic
SP Digital Override State	On	
SP Digital Blink Rate	500	[ms]
SP Digital Out Delay	0	[ms]
SP Delay Polarity	On	
SP Control Source	1	Received CAN J1939 Message
SP Control Number	12	CAN Input Signal #13
SP Enable Source	0	Control Source Not Used
SP Enable Number	0	
SP Enable Response	3	Enable OFF=Output Enabled, otherwise shut off
SP Override Source	0	Control Source Not Used
SP Override Number	0	
SP Override Response	0	Override When ON
SP Output Fault Response	0	Output Maintain Value
SP Proportional Gain	1	
SP Interval Time	2	[s]
SP Derivative Time	0	[s]

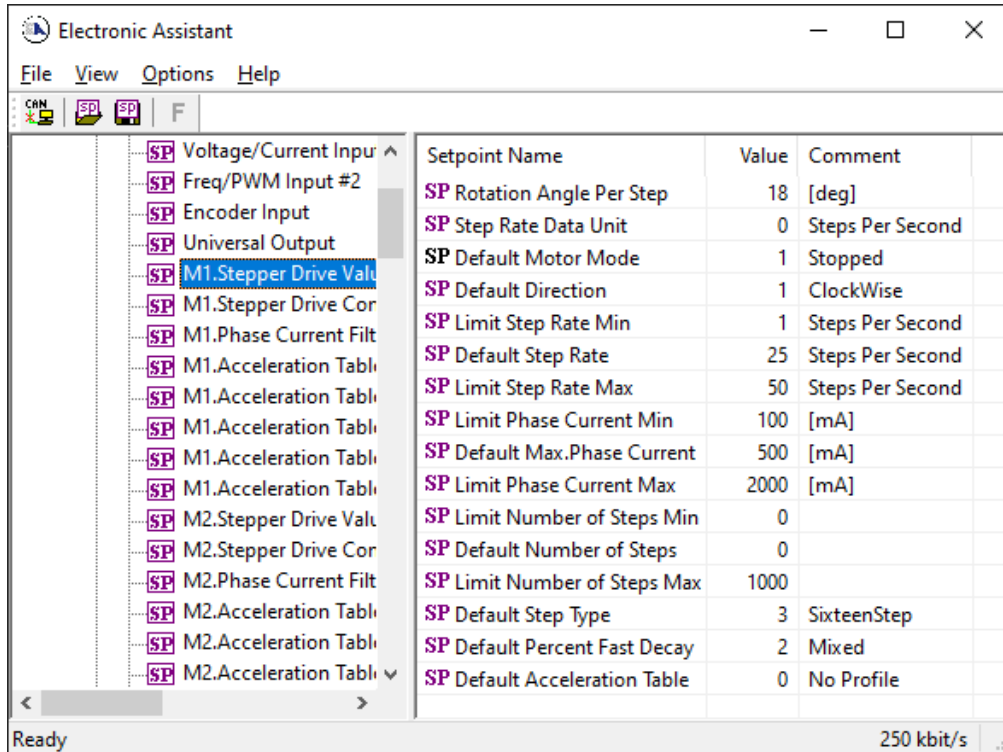
At the bottom of the window, the status bar shows 'Ready' on the left and '250 kbit/s' on the right.

Name	Range	Default	Notes
Output Type	Drop List	Disabled	See Table 21
Output At Minimum Command	0...Limit	300mA	
Output At Maximum Command	0...Limit	2500mA	
Output At Override Command	0...Limit	750mA	
Dither Frequency	50...400Hz	200Hz	
Dither Amplitude	0...500 mA	0	
Ramp Up (Min to Max)	0...10 000ms	1000ms	
Ramp Down (Max to Min)	0...10 000ms	1000ms	
PWM Output Frequency	1...25 000Hz	500Hz	
High/Low Switch	Drop List	1, High	
Hold Current	0... 1500mA	500mA	
Hotshot Current	0...1500mA	1000mA	
Hotshot Time	500...10 000 ms	1000ms	
Digital Response	Drop List	Normal On/Off	See Table 22
Digital Override State	Drop List	ON	ON / OFF
Digital Blink Rate	100...5000 ms	1000ms	
Digital Out Delay	0 to 60000ms	0ms	0ms → No delay
Delay Polarity	Drop List	Rising Edge	See Table 24
Control Source	Drop List	CAN Input Message	See Table 39
Control Number	Depends on control source	13	See Table 39
Enable Source	Drop List	Control not used	See Table 39
Enable Number	Depends on enable source	1	See Table 39
Enable Response	Drop List	Enable When On, else Shutoff	See Table 25
Override Source	Drop List	Control not used	See Table 39
Override Number	Depends on enable source	1	See Table 39
Override Response	Drop List	Override When On	See Table 26
Fault Detection is Enabled	Drop List	True	
Output Fault Response	Drop List	Shutoff Output	See Table 27
Output in Fault Mode	Depends on Output type	500mA	
Hysteresis to Clear Fault	0.01...250.00 mA	100.00mA	
Event Generates a DTC by DM1	Drop List	False	
Event Cleared only by DM11	Drop List	False	
Lamp Set by Event in DM1	Drop List	Amber Warning	See Table 39
SPN for Event used in DTC	0...65535	0	
FMI for Event used in DTC	Drop List	5, Open circuit	See Table 39
Delay Before Sending DM1	0...60000 ms	1000	

4.5 Stepper Drive Values Setpoints

This function block defines the default values used by the driver after a power-up or reset cycle, as well as allowing the user to set the limits for the linear controls.

The Motor Drive function blocks are defined in Section 0. Please refer there for detailed information about how all these setpoints are used.



Name	Range	Default	Notes
Rotation Angle Per Step	0.1 to 360.0°	18.0°	Per Stepper Motor Datasheet
Step Rate Data Units	Drop List	0, Steps Per Second	0 = Steps Per Second 1 = Rotations Per Minute
Default Motor Mode	N/A	1, Stopped	Read-Only Parameter See Table 2
Default Direction	Drop List	1, ClockWise	See Table 3
Limit Step Rate Min	1 to Default	1 [Unit]	Unit = Step Rate Data Units
Default Step Rate	Min to Max	25 [Step/s]	Unit = Step Rate Data Units
Limit Step Rate Max	Default to Limit*	50 [Step/s]	Unit = Step Rate Data Units
Limit Phase Current Min	100mA to Default	100 mA	
Default Max. Phase Current	Min to Max	500 mA	See Figure 5, Figure 6
Limit Phase Current Max	Default to 2000mA	2000 mA	
Limit Number of Steps Min	N/A	0	Read-Only Parameter
Default Number of Steps	0 to Max	0	
Limit Number of Steps Max	Default to 10,000,000	1000	
Default Step Type	Drop List	3, SixteenStep	See Table 4
Default Percent Fast Decay	Drop List	2, Mixed	See Table 5
Default Acceleration Table	Drop List	0, No Profile	See Table 6

*Step Rate maximum limit is dependent on Step Type selected. See table below

Step Type	Full Step	Half Step	Quarter Step	Sixteenth Step
Maximum [Step/s]	12,500	6,250	3,125	780
Maximum [RPM]*	37,500	18,750	9,375	2,340

*Where the RPM limit is calculated by

$$\text{Max} = (\text{Max [Step/s]}) \times (\text{"Rotation Angle Per Step" } [^\circ/\text{Step}]) \times (60[\text{s/min}]) / (360[^\circ/\text{Rotation}])$$

The driver chip A3979 will micro step once on a low to high transition on the STEP input. Since this control signal is driven by a 10us timer that will toggle the drive signal twice per micro step (low to high, and high to low), resolution will be lost at higher speeds. For high step rates, a cruder **"Step Type"** should be selected to get better resolution. At lower speeds, the 'SixteenStep' type should be used for optimum performance from the motor.

An example of how resolution is lost at higher speeds is shown in the table on the following page.

Timer Reload	MicroStep/Sec	1/16 Step Rate	1/4 Step Rate	Half Step Rate	Full Step Rate
1	50000	781.25	3125	6250	12500
2	25000	390.63	1562.5	3125	6250
3	16666.70	260.42	1041.67	2083.34	4166.68
4	12500.00	195.31	781.25	1562.50	3125.00
5	10000.00	156.25	625.00	1250.00	2500.00
6	8333.30	130.21	520.83	1041.66	2083.33
7	7142.90	111.61	446.43	892.86	1785.73
8	6250.00	97.66	390.63	781.25	1562.50
9	5555.60	86.81	347.23	694.45	1388.90
10	5000.00	78.13	312.50	625.00	1250.00
11	4545.50	71.02	284.09	568.19	1136.38
12	4166.70	65.10	260.42	520.84	1041.68
13	3846.20	60.10	240.39	480.78	961.55
14	3571.40	55.80	223.21	446.43	892.85
15	3333.30	52.08	208.33	416.66	833.33
16	3125.00	48.83	195.31	390.63	781.25
17	2941.20	45.96	183.83	367.65	735.30
18	2777.80	43.40	173.61	347.23	694.45
19	2631.60	41.12	164.48	328.95	657.90
20	2500.00	39.06	156.25	312.50	625.00
21	2381.00	37.20	148.81	297.63	595.25
22	2272.70	35.51	142.04	284.09	568.18
23	2173.90	33.97	135.87	271.74	543.48
24	2083.30	32.55	130.21	260.41	520.83
25	2000.00	31.25	125.00	250.00	500.00
26	1923.10	30.05	120.19	240.39	480.78
27	1851.90	28.94	115.74	231.49	462.98
28	1785.70	27.90	111.61	223.21	446.43
29	1724.10	26.94	107.76	215.51	431.03
30	1666.70	26.04	104.17	208.34	416.68
31	1612.90	25.20	100.81	201.61	403.23
32	1562.50	24.41	97.66	195.31	390.63
33	1515.20	23.68	94.70	189.40	378.80
34	1470.60	22.98	91.91	183.83	367.65
35	1428.60	22.32	89.29	178.58	357.15
36	1388.90	21.70	86.81	173.61	347.23
37	1351.40	21.12	84.46	168.93	337.85
38	1315.80	20.56	82.24	164.48	328.95
39	1282.10	20.03	80.13	160.26	320.53
40	1250.00	19.53	78.13	156.25	312.50
41	1219.50	19.05	76.22	152.44	304.88
42	1190.50	18.60	74.41	148.81	297.63
43	1162.80	18.17	72.68	145.35	290.70
44	1136.40	17.76	71.03	142.05	284.10
45	1111.10	17.36	69.44	138.89	277.78
46	1087.00	16.98	67.94	135.88	271.75
47	1063.80	16.62	66.49	132.98	265.95
48	1041.70	16.28	65.11	130.21	260.43
49	1020.40	15.94	63.78	127.55	255.10
50	1000.00	15.63	62.50	125.00	250.00

4.6 Stepper Drive Controls Setpoints

This function block defines the control sources and numbers for all the command and setup parameters associated with the stepper motor drive.

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

The screenshot shows the 'Electronic Assistant' software window. On the left, a tree view lists various setpoints, with 'M1.Stepper Drive Controls' selected and highlighted in blue. The main area displays a table of these setpoints, organized into three columns: 'Setpoint Name', 'Value', and 'Comment'. The table lists 24 setpoints, each with a corresponding value and a comment indicating the control source (e.g., 'Received CAN J1939 Message' or 'CAN Input Signal #1').

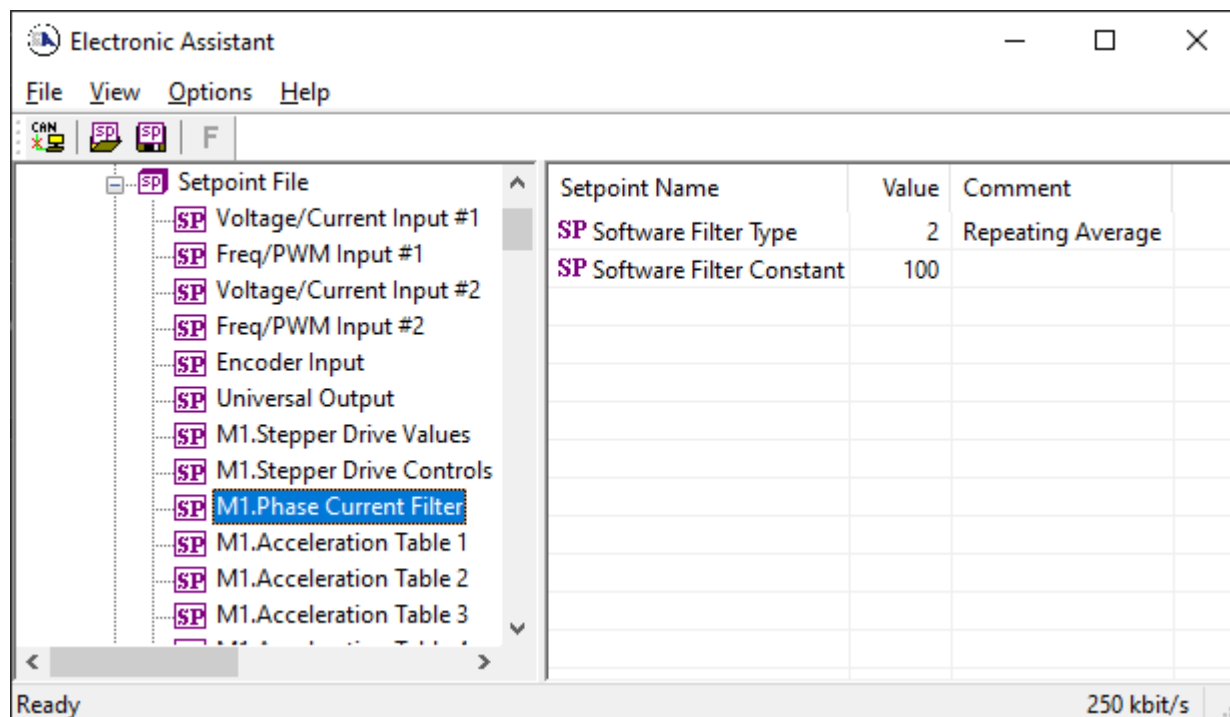
Setpoint Name	Value	Comment
SP Control Source, Motor Mode	1	Received CAN J1939 Message
SP Control Number, Motor Mode	0	CAN Input Signal #1
SP Control Source, Direction	1	Received CAN J1939 Message
SP Control Number, Direction	1	CAN Input Signal #2
SP Control Source, Step Rate	1	Received CAN J1939 Message
SP Control Number, Step Rate	2	CAN Input Signal #3
SP Control Source, Max.Phase Current	1	Received CAN J1939 Message
SP Control Number, Max.Phase Current	3	CAN Input Signal #4
SP Control Source, Number of Steps	1	Received CAN J1939 Message
SP Control Number, Number of Steps	4	CAN Input Signal #5
SP Control Source, Step Type	1	Received CAN J1939 Message
SP Control Number, Step Type	5	CAN Input Signal #6
SP Control Source, Percent Fast Decay	1	Received CAN J1939 Message
SP Control Number, Percent Fast Decay	6	CAN Input Signal #7
SP Control Source, Acceleration Table	1	Received CAN J1939 Message
SP Control Number, Acceleration Table	7	CAN Input Signal #8
SP Control Source, Put Drive to Sleep	1	Received CAN J1939 Message
SP Control Number, Put Drive to Sleep	8	CAN Input Signal #9
SP Control Source, Reset Motor Drive	1	Received CAN J1939 Message
SP Control Number, Reset Motor Drive	9	CAN Input Signal #10
SP Control Source, E - Stop Motor Drive	1	Received CAN J1939 Message
SP Control Number, E - Stop Motor Drive	10	CAN Input Signal #11
SP Control Source, Save Data to Defaults	1	Received CAN J1939 Message
SP Control Number, Save Data to Defaults	11	CAN Input Signal #12

The status bar at the bottom of the window shows 'Ready' on the left and '250 kbit/s' on the right.

By default, all control sources are selected to be a '*Received CAN J1939 Message*', and the number is mapped to the appropriate default. See Section 1.13 for more information about the CAN messages, and Table 39 for a complete list of the available control sources supported by the controller.

4.7 Phase Current Filter Setpoints

The Phase Current Filter function block is defined in Section 1.5. Please refer there for detailed information about how all these setpoints are used. The same type of filter is applied to both phases.



Name	Range	Default	Notes
Software Filter Type	Drop List	2, Repeating Average	See Table 20
Software Filter Constant	1 to 1,000	100	Used with Moving or Repeating Average Filters

4.8 Acceleration Table Setpoints

See Section 1.2.10 for information on how to use the Acceleration Tables.

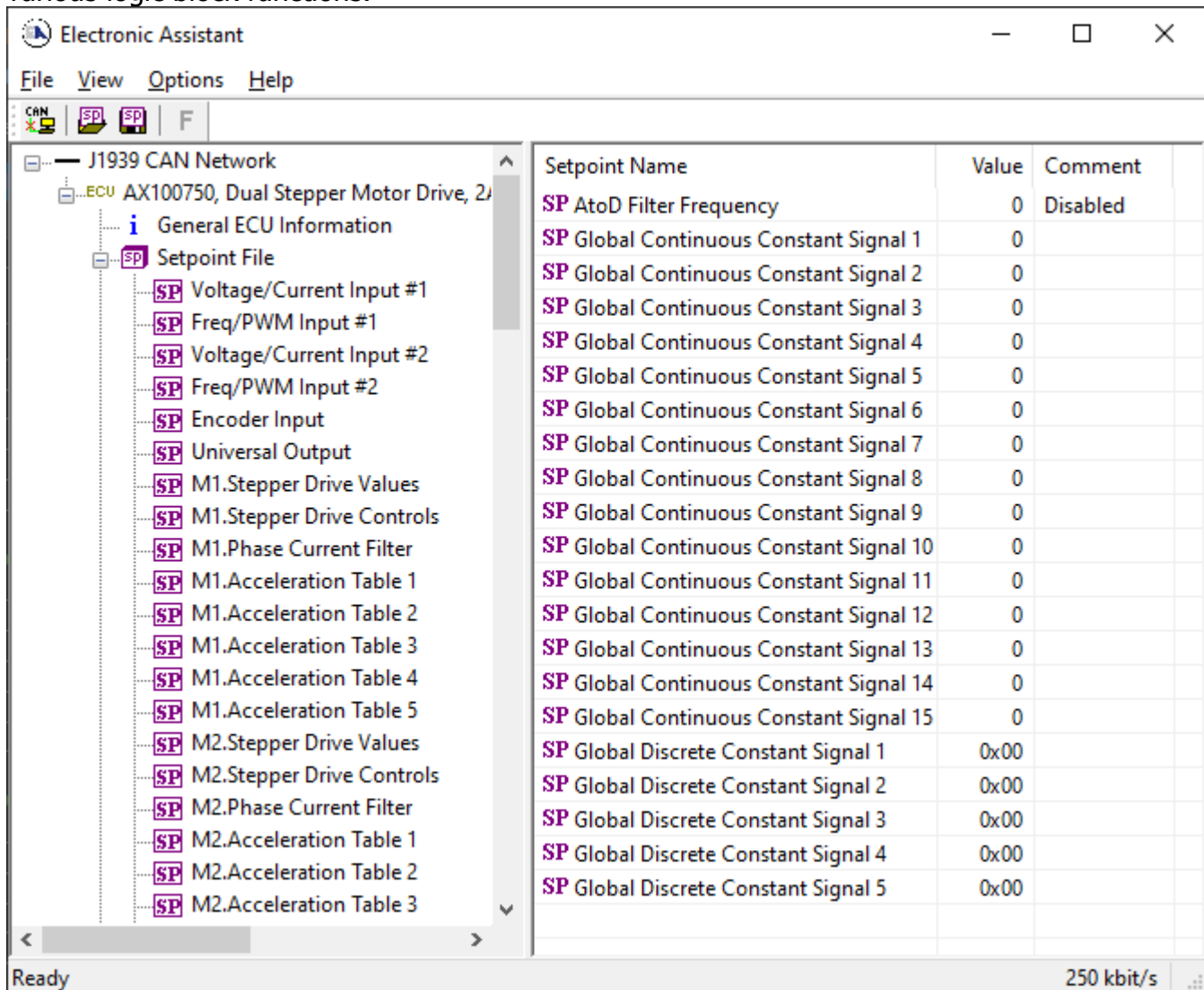
The screenshot shows the 'Electronic Assistant' software window. The left pane displays a tree view of the 'J1939 CAN Network' with the 'Setpoint File' expanded. The 'M1.Acceleration Table 1' is selected. The right pane shows a table of setpoints.

Setpoint Name	Value	Comment
SP Point 1 Step Rate	15	Steps Per Second
SP Point 2 Step Rate	25	Steps Per Second
SP Point 3 Step Rate	33	Steps Per Second
SP Point 4 Step Rate	40	Steps Per Second
SP Point 5 Step Rate	46	Steps Per Second
SP Point 6 Step Rate	51	Steps Per Second
SP Point 7 Step Rate	55	Steps Per Second
SP Point 8 Step Rate	58	Steps Per Second
SP Point 9 Step Rate	60	Steps Per Second
SP Point 10 Step Rate	62	Steps Per Second
SP Point 11 Step Rate	63	Steps Per Second
SP Point 12 Step Rate	64	Steps Per Second
SP Point 13 Step Rate	65	Steps Per Second
SP Point 14 Step Rate	65	Steps Per Second
SP Point 15 Step Rate	65	Steps Per Second
SP Point 16 Step Rate	65	Steps Per Second
SP Point 17 Step Rate	65	Steps Per Second
SP Point 18 Step Rate	65	Steps Per Second
SP Point 19 Step Rate	65	Steps Per Second
SP Point 20 Step Rate	65	Steps Per Second
SP Point 21 Step Rate	65	Steps Per Second
SP Point 22 Step Rate	65	Steps Per Second
SP Point 23 Step Rate	65	Steps Per Second
SP Point 24 Step Rate	65	Steps Per Second
SP Point 25 Step Rate	65	Steps Per Second
SP Point 1 Time	0.01	[ms]
SP Point 2 Time	0.02	[ms]
SP Point 3 Time	0.03	[ms]
SP Point 4 Time	0.04	[ms]

Ready 250 kbit/s

4.9 Global Parameters Setpoints

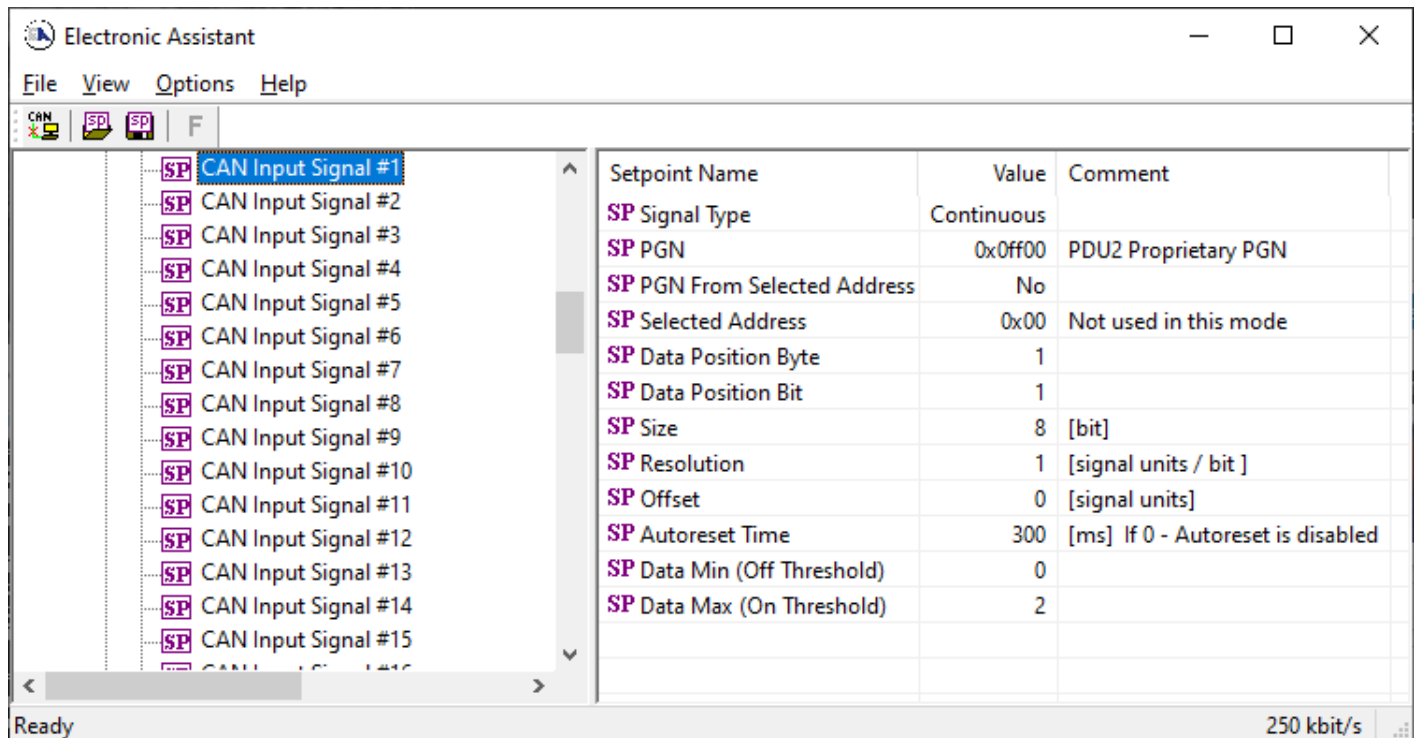
The Global Parameters Function Block is provided to allow the user to select values as desired for various logic block functions.



Name	Range	Default	Notes
Discrete Constant Data	0...4294967295	0	
AtoD Filter Frequency	Drop List	0, None	0 = None, 1 = 50Hz, 2 = 60Hz, 3 = Both
Continuous Constant Data	-3.4e38...3.4e38	0	
Discrete Constant Data	0...4294967295	0	

4.10 CAN Receive Setpoints

The CAN Receive function block is defined in Section 1.8, with additional information in Section 3.3. Please refer there for detailed information about how all these setpoints are used.



Name	Range	Default	Notes
Signal Type	Drop List	2, Continuous	
PGN	0 to 65535	Different for each	See Section 3.3 for defaults
PGN From Selected Address	Drop List	False	
Selected Address	0 to 255	254 (0xFE, Null Addr)	
Data Position Byte	1-8	1	
Data Position Bit	1-8	1	
Size	Drop List	1-Byte	
Resolution	-100000.0 to 100000	1.0	
Offset	-10000 to 10000	0.0	
Auto-Reset Time	0 to 60 000 ms	0 ms	0ms disables the Lost Comm Fault
Data Minimum	-1000000 to Max	0.0	
Data Maximum	-100000 to 100000	100.0	

4.11 CAN Transmit Setpoints

The CAN Transmit function block is defined in Section 1.14, with addition information in Section 0. Please refer there for detailed information about how all these setpoints are used.

The screenshot shows the 'Electronic Assistant' window with the 'CAN' tab selected. On the left, a list of setpoints is shown, with 'CAN Output Message #2' highlighted. The main area displays the configuration for this setpoint in a table format.

Setpoint Name	Value	Comment
SP PGN	0x0ff55	PDU2 Proprietary PGN
SP Transmission Enable	Yes	
SP Transmission Rate	1000	[ms] On request only, if 0
SP Destination Address	0xff	Global Address
SP Length	8	[byte]
SP Priority	6	
SP Signal #1 Type	Continuous	
SP Signal #1 Source	2	Input Measured Value
SP Signal #1 Number	0	Voltage/Current Input #1
SP Signal #1 Byte Position	1	
SP Signal #1 Bit Position	1	
SP Signal #1 Size	16	[bit]
SP Signal #1 Resolution	0.001	[signal units / bit]
SP Signal #1 Offset	0	[signal units]
SP Signal #2 Type	Continuous	
SP Signal #2 Source	2	Input Measured Value
SP Signal #2 Number	1	Freq/PWM Input #1
SP Signal #2 Byte Position	3	

The status bar at the bottom indicates 'Ready' and '250 kbit/s'.

Name	Range	Default	Notes
PGN	0xff00 ... 0xffff	Different for each	See Section 1.14.1
Transmission Enable	Yes/No	No	"No" disables transmit
Transmission Rate	0 ... 65000 ms	0ms	0ms disables transmit
Destination Address	0 to 255	0xFF	
Length	1..8	8	Not used by default
Priority	0...7	6	Proprietary B Priority
Signal # Control Source	Drop List	Different for each	See Table 39 for options
Signal # Control Number	Drop List	Different for each	See 1.14.2
Signal # Transmit Data Size	Drop List	2 bytes	
Signal # Transmit Data Index in Array	0-7	2	
Signal # Transmit Bit Index In Byte	0-7	0	
Signal # Transmit Data Resolution	-100000.0 to 100000	0.001	

4.12 Lookup Table Setpoints

The Lookup Table Function Block is defined in Section 1.81.8 Please refer there for detailed information about how all these setpoints are used. “**X-Axis Source**” is set to ‘*Control Not Used*’ by default. To enable a Lookup Table select appropriate “**X-Axis Source**”.

Electronic Assistant

File View Options Help

CAN SP SP F

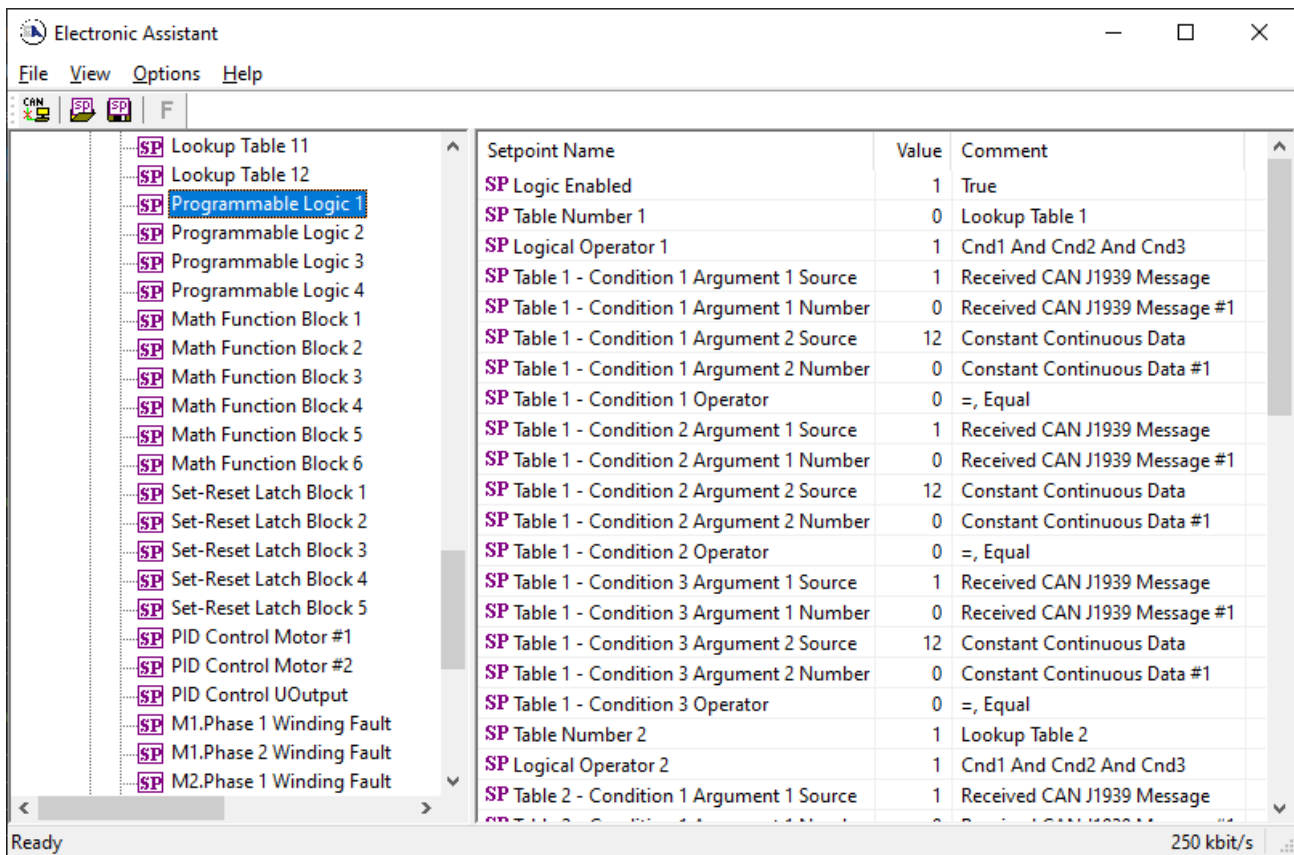
Setpoint Name	Value	Comment
SP X-Axis Source	2	Input Measured Value
SP X-Axis Number	0	Input Measured Value #1
SP X-Axis Type	0	Data Response
SP Auto Repeat	0	False
SP X Decimal Digits	0	
SP Y Decimal Digits	0	
SP Response 1	1	Ramp To
SP Response 2	1	Ramp To
SP Response 3	1	Ramp To
SP Response 4	1	Ramp To
SP Response 5	1	Ramp To
SP Response 6	1	Ramp To
SP Response 7	1	Ramp To
SP Response 8	1	Ramp To
SP Response 9	1	Ramp To
SP Response 10	1	Ramp To
SP Point X1	0.500	
SP Point X2	1.000	
SP Point X3	1.500	
SP Point X4	2.000	
SP Point X5	2.500	
SP Point X6	3.000	
SP Point X7	3.500	
SP Point X8	4.000	
SP Point X9	4.500	
SP Point X10	5.000	
SP Point Y1	10.000	
SP Point Y2	20.000	
SP Point Y3	30.000	
SP Point Y4	40.000	
SP Point Y5	50.000	
SP Point Y6	60.000	
SP Point Y7	70.000	
SP Point Y8	80.000	
SP Point Y9	90.000	
SP Point Y10	100.000	

Ready 250 kbit/s

Name	Range	Default	Notes
X-Axis Source	Drop List	Control Not Used	See Table 39
X-Axis Number	Depends on control source	1	See Table 39
X-Axis Type	Drop List	Data Response	See Table 29
Auto Repeat	Drop List	False	
X Decimal Digits	0...3	0	Resolution is 10^x , affects X points
Y Decimal Digits	0...3	0	Resolution is 10^x , affects Y points
Response 1	Drop List	Ramp To	See Table 30
Response 2	Drop List	Ramp To	See Table 30
Response 3	Drop List	Ramp To	See Table 30
Response 4	Drop List	Ramp To	See Table 30
Response 5	Drop List	Ramp To	See Table 30
Response 6	Drop List	Ramp To	See Table 30
Response 7	Drop List	Ramp To	See Table 30
Response 8	Drop List	Ramp To	See Table 30
Response 9	Drop List	Ramp To	See Table 30
Response 10	Drop List	Ramp To	See Table 30
Point X1	From X-Axis source minimum to Point 1 - X Value	X-Axis source minimum Depends on the Table number	See Section 1.8
Point X2	From Point 0 - X Value to Point 2 - X Value	Depends on the Table number	See Section 1.8
Point X3	From Point 1 - X Value to Point 3 - X Value	Depends on the Table number	See Section 1.8
Point X4	From Point 2 - X Value to Point 4 - X Value	Depends on the Table number	See Section 1.8
Point X5	From Point 3 - X Value to Point 5 - X Value source	Depends on the Table number	See Section 1.8
Point X6	From Point 4 - X Value to Point 6 - X Value	Depends on the Table number	See Section 1.8
Point X7	From Point 5 - X Value to Point 7 - X Value	Depends on the Table number	See Section 1.8
Point X8	From Point 6 - X Value to Point 8 - X Value	Depends on the Table number	See Section 1.8
Point X9	From Point 7 - X Value to Point 9 - X Value	Depends on the Table number	See Section 1.8
Point X10	From Point 8 - X Value to Point 10 - X Value	Depends on the Table number	See Section 1.8
Point Y1	-10^6 to 10^6	Depends on the Table number	
Point Y2	-10^6 to 10^6	Depends on the Table number	
Point Y3	-10^6 to 10^6	Depends on the Table number	
Point Y4	-10^6 to 10^6	Depends on the Table number	
Point Y5	-10^6 to 10^6	Depends on the Table number	
Point Y6	-10^6 to 10^6	Depends on the Table number	
Point Y7	-10^6 to 10^6	Depends on the Table number	
Point Y8	-10^6 to 10^6	Depends on the Table number	
Point Y9	-10^6 to 10^6	Depends on the Table number	
Point Y10	-10^6 to 10^6	Depends on the Table number	

4.13 Programmable Logic Setpoints

The Programmable Logic function block is defined in Section 1.9. Please refer there for detailed information about how all these setpoints are used. “**Logic Enabled**” is ‘*False*’ by default. To enable Logic set “**Logic Enabled**” to ‘*True*’ and select appropriate “**Argument Source**”.

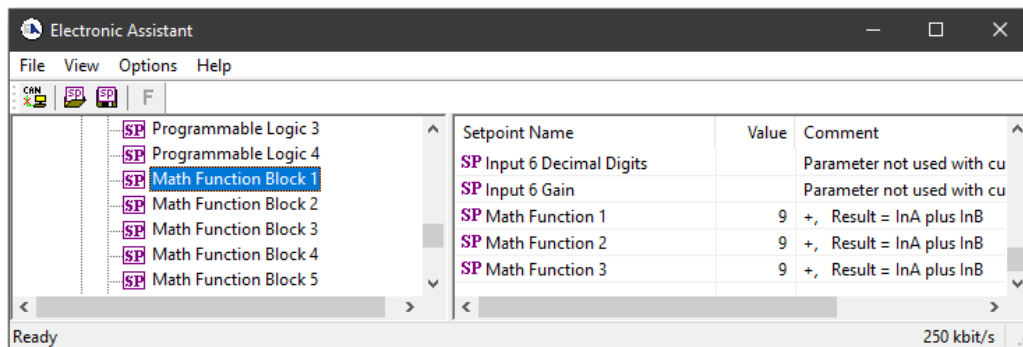
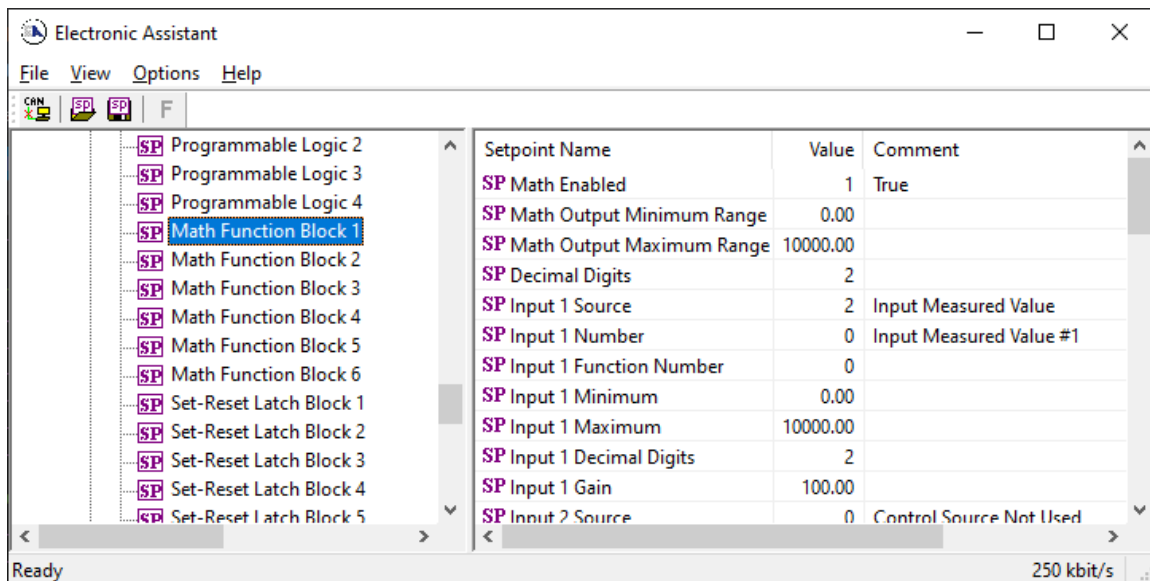


Setpoint ranges and default values for Programmable Logic Blocks are listed below. Only “**Table 1**” setpoint are listed, because other “**Table X**” setpoints are similar, except for the default value of the “**Table Number**” setpoint, which is X for “**Table X**”.

Name	Range	Default	Notes
Logic Enabled	Drop List	False	
Table Number 1	1 to 12	Lookup Table 1	
Logical Operator 1	Drop List	Default Table	See Table 32
Table 1 - Condition 1 Argument 1 Source	Drop List	Control Not Used	See Table 39
Table 1 - Condition 1 Argument 1 Number	Depends on control source	1	See Table 39
Table 1 - Condition 1 Argument 2 Source	Drop List	Control Not Used	See Table 39
Table 1 - Condition 1 Argument 2 Number	Depends on control source	1	See Table 39
Table 1 - Condition 1 Operator	Drop List	=, Equal	See Table 31
Table 1 - Condition 2 Argument 1 Source	Drop List	Control Not Used	See Table 39
Table 1 - Condition 2 Argument 1 Number	Depends on control source	1	See Table 39
Table 1 - Condition 2 Argument 2 Source	Drop List	Control Not Used	See Table 39
Table 1 - Condition 2 Argument 2 Number	Depends on control source	1	See Table 39
Table 1 - Condition 2 Operator	Drop List	=, Equal	See Table 31
Table 1 - Condition 3 Argument 1 Source	Drop List	Control Not Used	See Table 39
Table 1 - Condition 3 Argument 1 Number	Depends on control source	1	See Table 39
Table 1 - Condition 3 Argument 2 Source	Drop List	Control Not Used	See Table 39
Table 1 - Condition 3 Argument 2 Number	Depends on control source	1	See Table 39
Table 1 - Condition 3 Operator	Drop List	=, Equal	See Table 31

4.14 Math Function Block

The Math Function Block is defined in Section 0. Please refer there for detailed information about how all these setpoints are used. **“Math Function Enabled”** is ‘False’ by default. To enable a Math Function Block, set **“Math Function Enabled”** to ‘True’ and select appropriate **“Input Source”**.

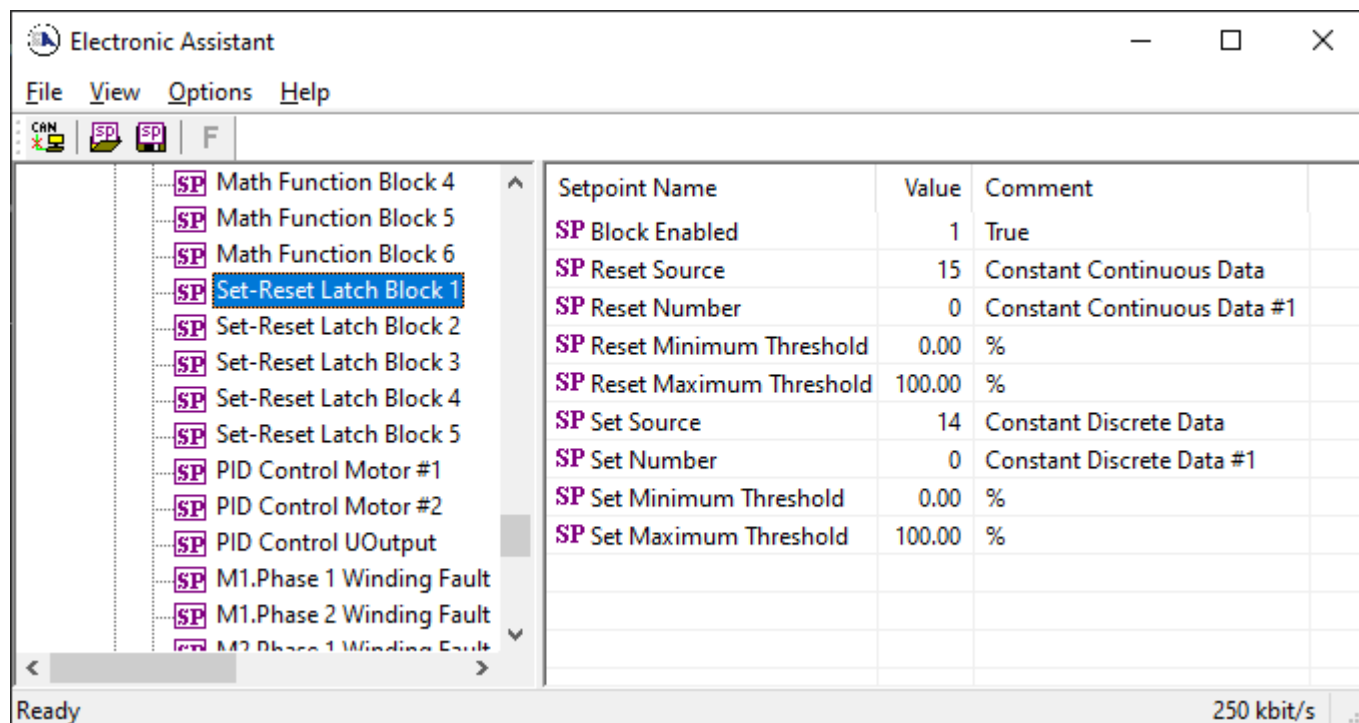


Setpoint ranges and default values for Math Blocks are listed below. **Error! Reference source not found..** Only **“Input 1”** setpoint are listed, because other **“Input X”** setpoints are similar.

Name	Range	Default	Notes
Math Enabled	Drop List	False	
Math Output Minimum Range	-32768...32767	0	
Math Output Maximum Range	-32768...32767	10000.0	
Decimal Digits	0...3	2	Resolution is 10 ^x , affects Output Min/Max Ranges
Input 1 Source	Drop List	Control not used	See Table 39
Input 1 Number	Depends on control source	1	See Table 39
Input 1 Function Number	1...3	1	
Input 1 Minimum	-10 ⁶ ...10 ⁶	0.00	
Input 1 Maximum	-10 ⁶ ...10 ⁶	10000.00	
Input 1 Decimal Digits	0..3	2	Resolution is 10 ^x , affects Input 1 Min/Max Ranges
Input 1 Gain	-100...100	100	
Math Function 1	Drop List	+, Result = InA plus InB	See Table 33
Math Function 2	Drop List	+, Result = InA plus InB	See Table 33
Math Function 3	Drop List	+, Result = InA plus InB	See Table 33

4.15 Set-Reset Latch Block

The Set-Reset Latch Block setpoints are defined in Section 1.11. Refer to that section for detailed information on how these setpoints are used. The screen capture displays the available setpoints for each of the Set-Reset Latch Blocks. The table below the screen capture highlights the allowable ranges for each setpoint.

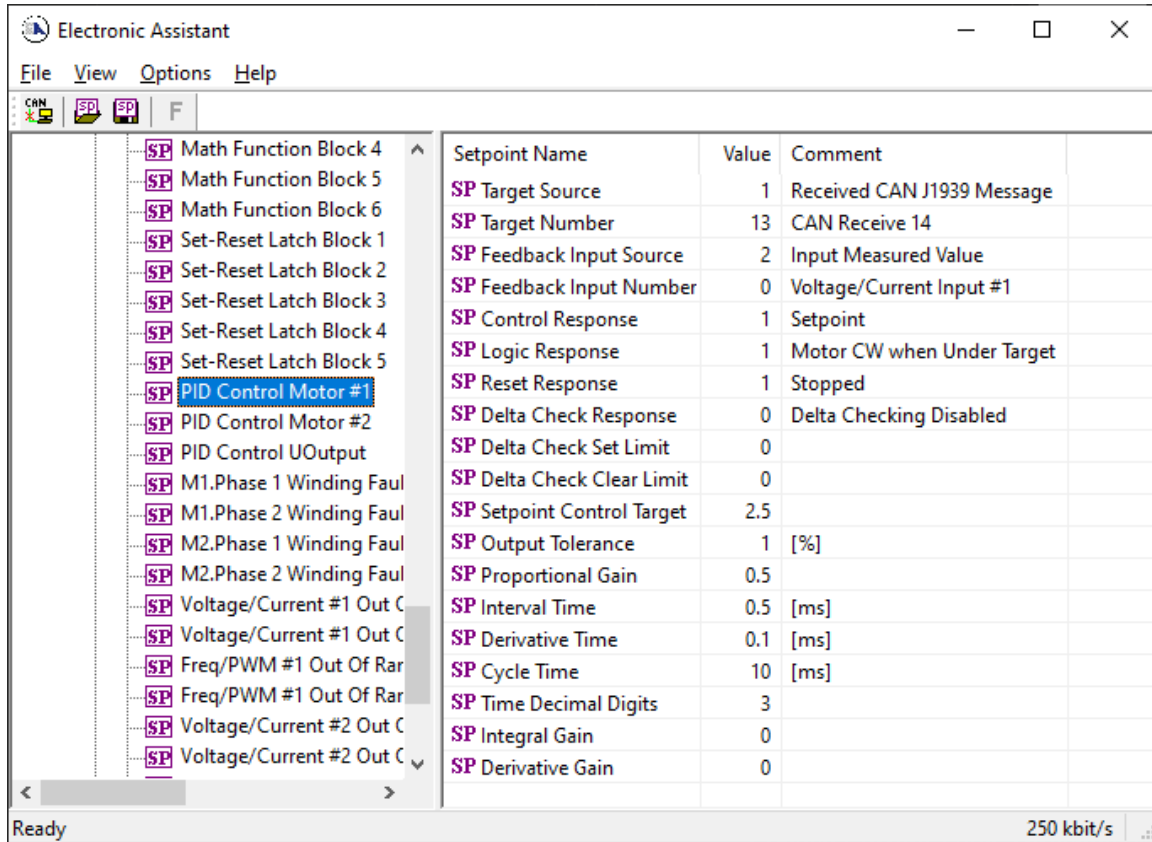


Name	Range	Default	Notes
Block Enabled	Drop List	False	
Reset Source	Drop List	Control Not Used	
Reset Number	Depends on Source Selected	1	
Reset Minimum Threshold	Drop List	0%	Refer to Section 0
Reset Maximum Threshold	Depends on Source Selected	100%	Refer to Section 0
Set Source	Drop List	Control Not Used	
Set Number	Drop List	1	
Set Minimum Threshold	Depends on Source Selected	0%	Refer to Section 0
Set Maximum Threshold	Drop List	100%	Refer to Section 0

4.16 PID Control Function Setpoints

4.16.1.1 PID Control for Stepper Motors:

This function block is only enabled if the **"Control Source, Motor Mode"** has been set to *'PID Function Block Control.'* Otherwise, all setpoints will read "Not Applicable with this type of Control." Please refer to Section 1.2.15 for detailed information about how to use these setpoints.

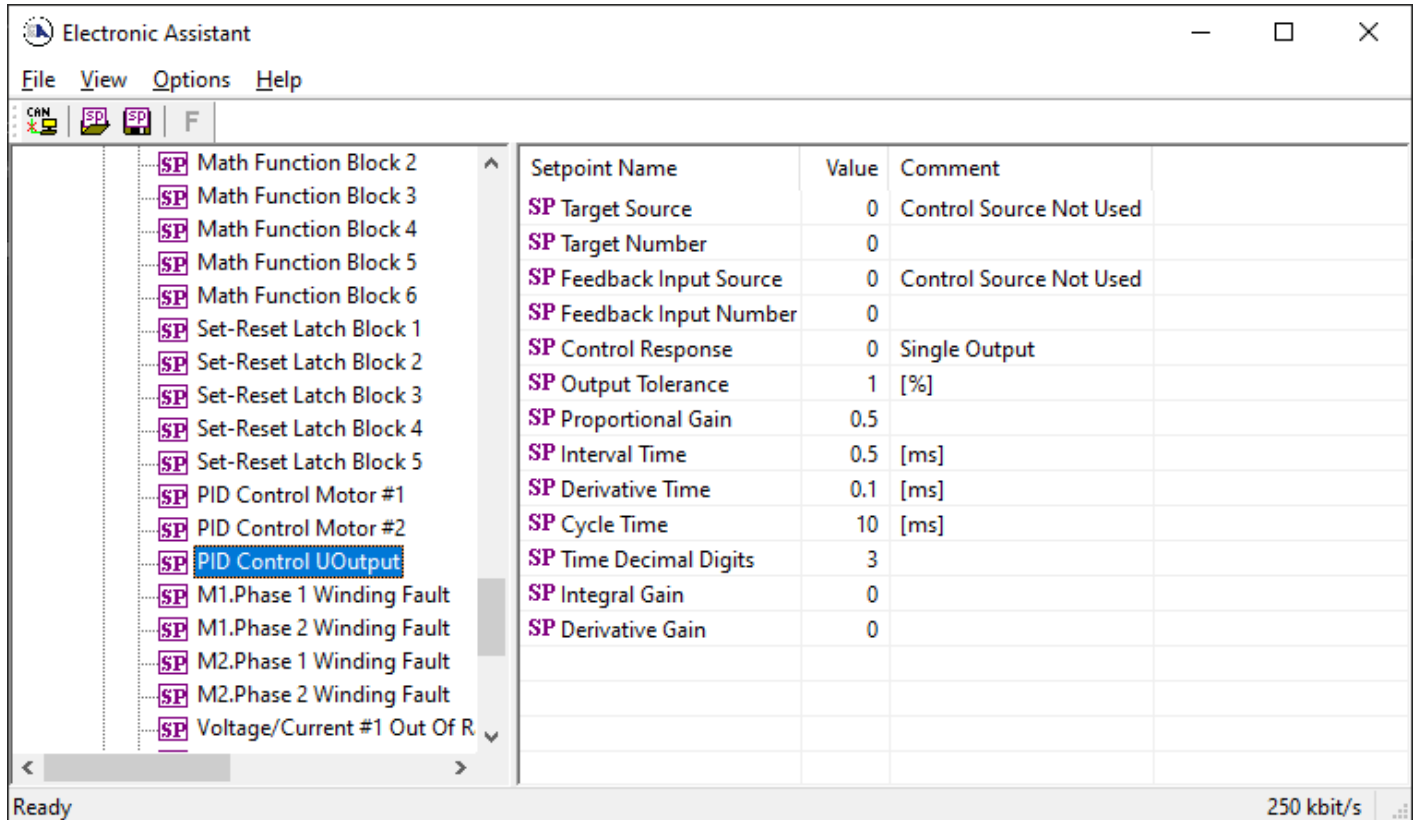


Name	Range	Default	Notes
PID Target Command Source	Drop List	1, Received CAN J1939 Message	See Table 39
PID Target Command Number	Per Source	14, Received Message 14	See Table 39
PID Feedback Input Source	Drop List	2, Input Measured Value	See Table 39
PID Feedback Input Number	Per Source	1, Voltage/Current Input 1	See Table 39
PID Control Algorithm	Drop List	1, Setpoint Control	See Table 7
PID Logic Response	Drop List	1, Motor CW when Under Target	See Table 8
PID Setpoint Control Target	-1,000,000 to 1,000,000	2.50 [V]	In Feedback Input unit
PID Delta Check Response	Drop List	0, Disabled	
PID Delta Check Set Limit	-1,000,000 to 1,000,000		
PID Delta Check Clear Limit	-1,000,000 to 1,000,000		
PID Loop Update Rate	1 to 60,000 ms	10 ms	1 ms resolution
PID Gain Coefficient, G	0.1 to 10	1.5	For proportional response
PID Integral Time Coeff, Ti	0.001 to 10 Sec	0.005 Sec	0.001 Sec (1ms) resolution
PID Derivative Time Coeff, Td	0.001 to 10 Sec	0.001 Sec	0.001 Sec (1ms) resolution
PID Integral Coefficient, Ki	0 to 10	0.0	0 disables integral, PD ctrl
PID Derivative Coefficient, Kd	0 to 10	0.0	0 disables derivative, PI ctrl

4.16.1.2 PID Control for Universal Output:

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

Command Source is set to 'Control Not Used' by default. To enable a PID Control, select appropriate **"Target Source"** and **"Feedback Source"**.

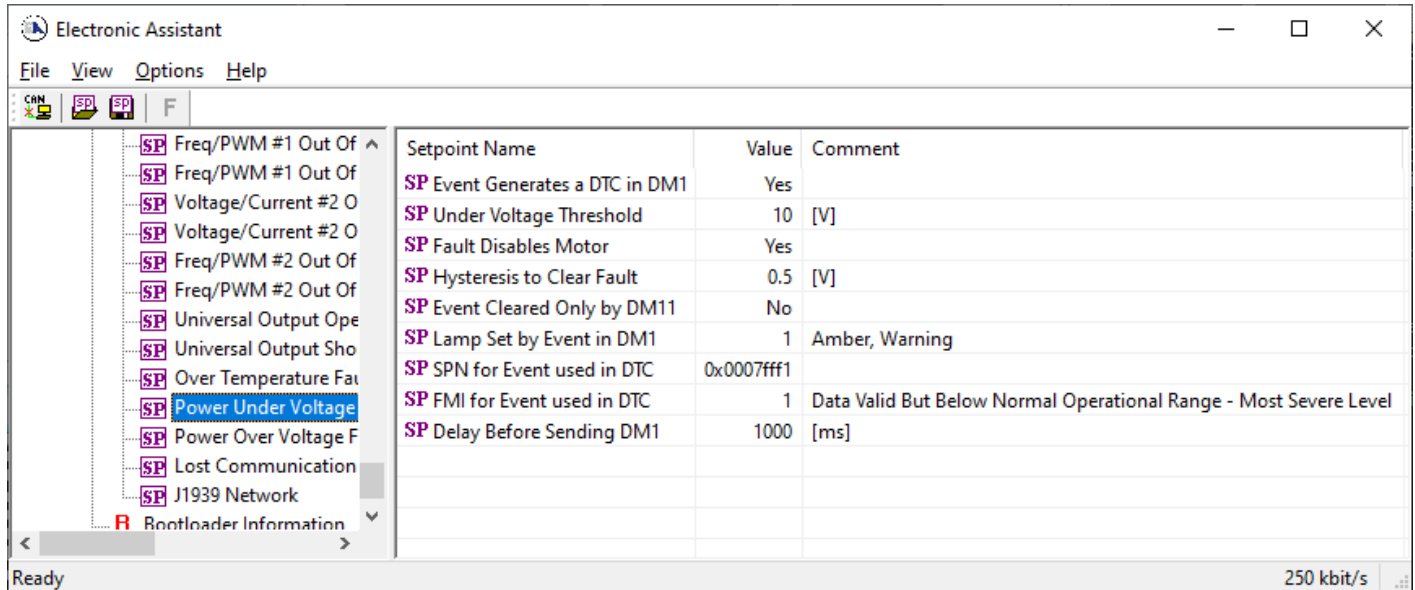


Name	Range	Default	Notes
Proportional Band, G	0...10	5.00	See Equation 2
Integral Time, Ti	0.001...1000.0	5.00	
Derivative Time, Td	0.001...1000.0	1.00	
Cycle Time	1...1000 ms	10ms	
Time Decimal Digits	0...3	3	Resolution is 10 ^x , affects Ti and Td
Output Tolerance	0...100 %	1.00 %	
Integral Gain, Ki	0.0...10.0	10.00	0 disables integral, PD ctrl
Derivative Gain, Kd	0.0...10.0	10.00	0 disables derivative, PI ctrl
Target Source	Drop List	Control Not Used	See Table 39
Target Number	Depends on control source	1	See Table 39
Feedback Source	Drop List	Control Not Used	See Table 39
Feedback Number	Depends on control source	1	See Table 39
Control Response	Drop List	Single Output	See Table 28

4.17 Diagnostic Setpoints

Here, the diagnostics are Phase 1 Winding, Phase 2 Winding, Voltage/current Out of Range, Frequency/PWM Out of Range, Universal Output Open/Short Circuit, Over Temperature Fault, Power Supply and Lost Communication

The Diagnostic function block is defined in Section 1.12. Please refer there for detailed information about how all these setpoints are used.



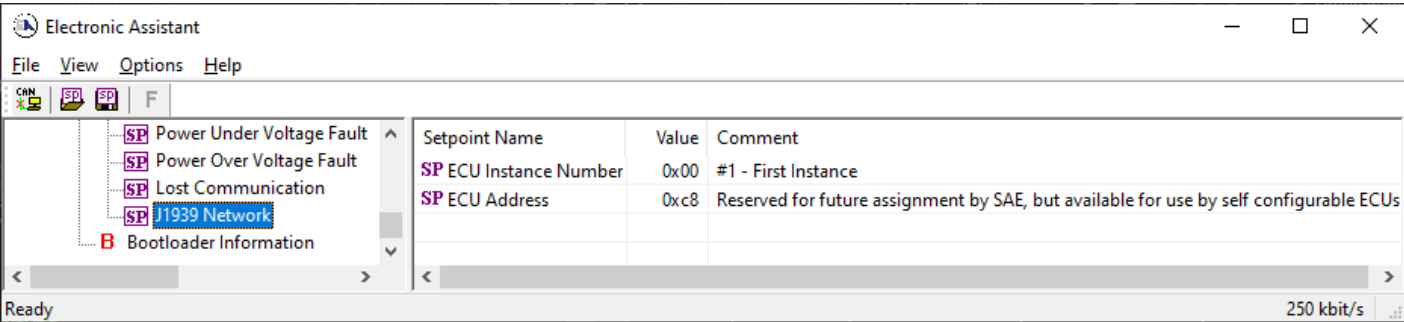
Name	Range	Default	Notes
Power Undervoltage Threshold *	9 to Over V	10.00 V	See * below
Power Overvoltage Threshold *	Under to 32V	30.00 V	See * below
Phase Current Fault Threshold **	5 to 500mA	50.00 mA	See ** below
Hysteresis to Clear Fault	0.01 to 50.0	10.000 mA 0.500 V N/A	Current Power Comm See Section 1.12
Fault Disables Motor *	False or True	True	See * below
Event Generates a DTC in DM1	False or True	True (False for Lost Communication)	See Section 1.12
Event Cleared Only by DM11	False or True	False	See Section 1.12
Lamp Set by Event in DM1	Drop List	1, Amber, Warning 2, Red, Stop 1, Amber, Warning	Current Power Comm See Section 1.12
SPN for Event used in DTC	1 to 524287		See Section 1.12
FMI for Event used in DTC	Drop List		See Section 1.12
Delay Before Sending DM1	0 to 60,000 ms	1000 ms	See Section 1.12

* Only used with Power Supply, Temperature, Lost Communication Diag groups

** Only used with Phase X Winding Diag group, where X = 1 or 2

4.18 CAN Network Setpoints

The CAN Network setpoints are described in detail in Section 3.2. Also, refer to the notes for more information about each setpoint.

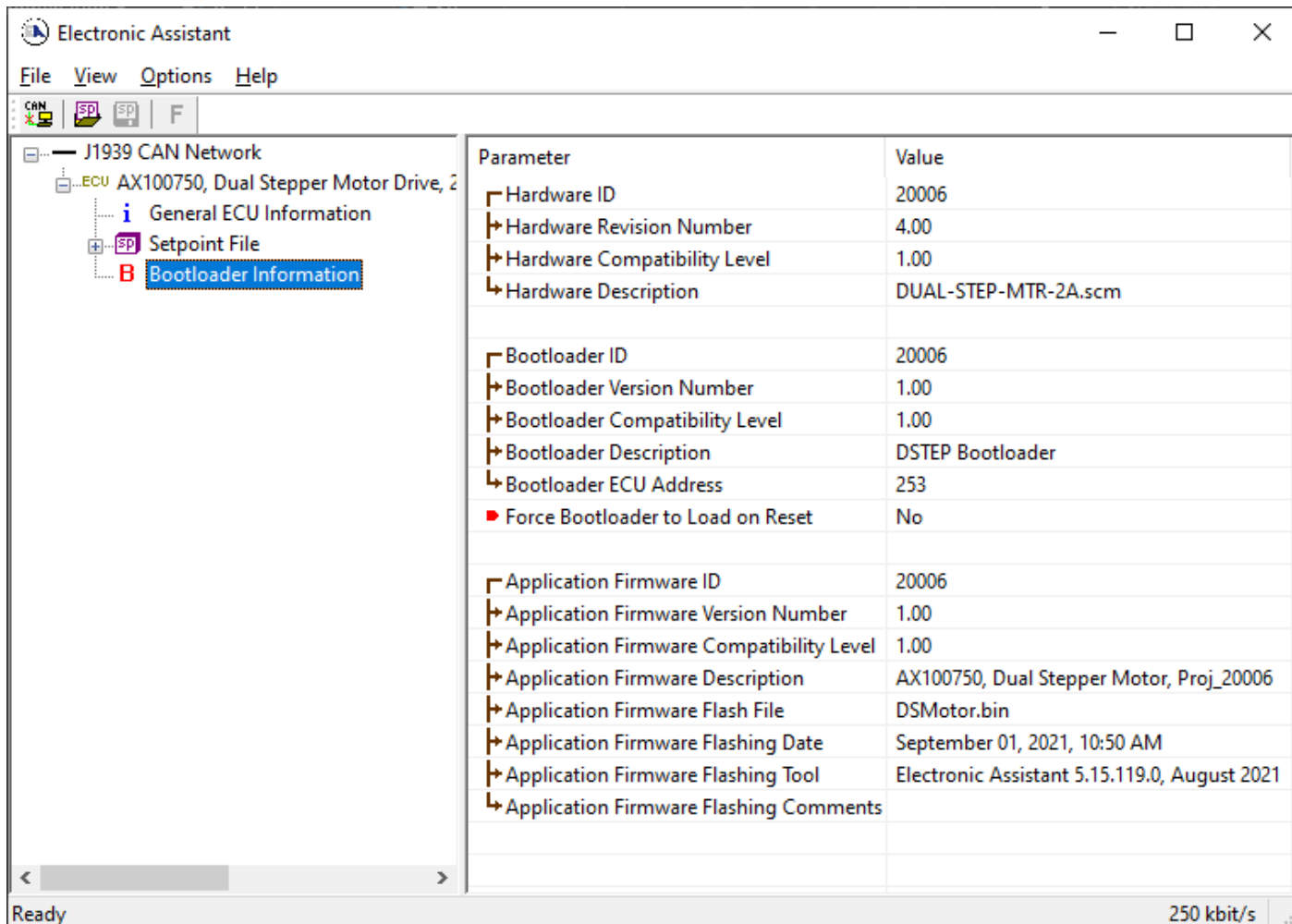


Name	Range	Default	Notes
ECU Instance Number	Drop List	0, #1 – First Instance	Per J1939-81
ECU Address	0 to 253	200	Preferred address for a motor control per J1939

5 REFLASHING OVER CAN WITH THE AXIOMATIC EA BOOTLOADER

The AX100750 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:

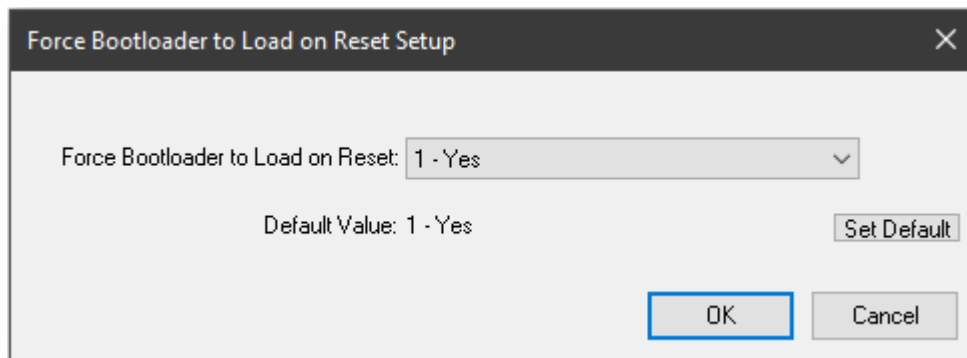


The screenshot shows the 'Electronic Assistant' window. On the left, a tree view under 'J1939 CAN Network' shows 'ECU AX100750, Dual Stepper Motor Drive, 2' expanded, with 'Bootloader Information' selected. The main area displays a table of parameters and their values.

Parameter	Value
Hardware ID	20006
Hardware Revision Number	4.00
Hardware Compatibility Level	1.00
Hardware Description	DUAL-STEP-MTR-2A.scm
Bootloader ID	20006
Bootloader Version Number	1.00
Bootloader Compatibility Level	1.00
Bootloader Description	DSTEP Bootloader
Bootloader ECU Address	253
Force Bootloader to Load on Reset	No
Application Firmware ID	20006
Application Firmware Version Number	1.00
Application Firmware Compatibility Level	1.00
Application Firmware Description	AX100750, Dual Stepper Motor, Proj_20006
Application Firmware Flash File	DSMotor.bin
Application Firmware Flashing Date	September 01, 2021, 10:50 AM
Application Firmware Flashing Tool	Electronic Assistant 5.15.119.0, August 2021
Application Firmware Flashing Comments	

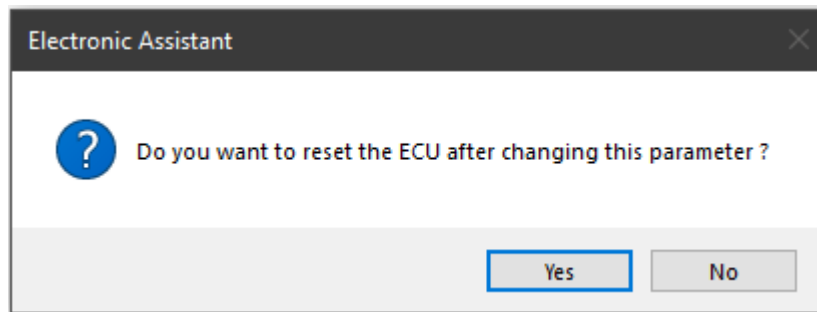
The status bar at the bottom shows 'Ready' and '250 kbit/s'.

2. To use the bootloader for upgrading the firmware running on the ECU, change the variable “**Force Bootloader to Load on Reset**” to Yes.

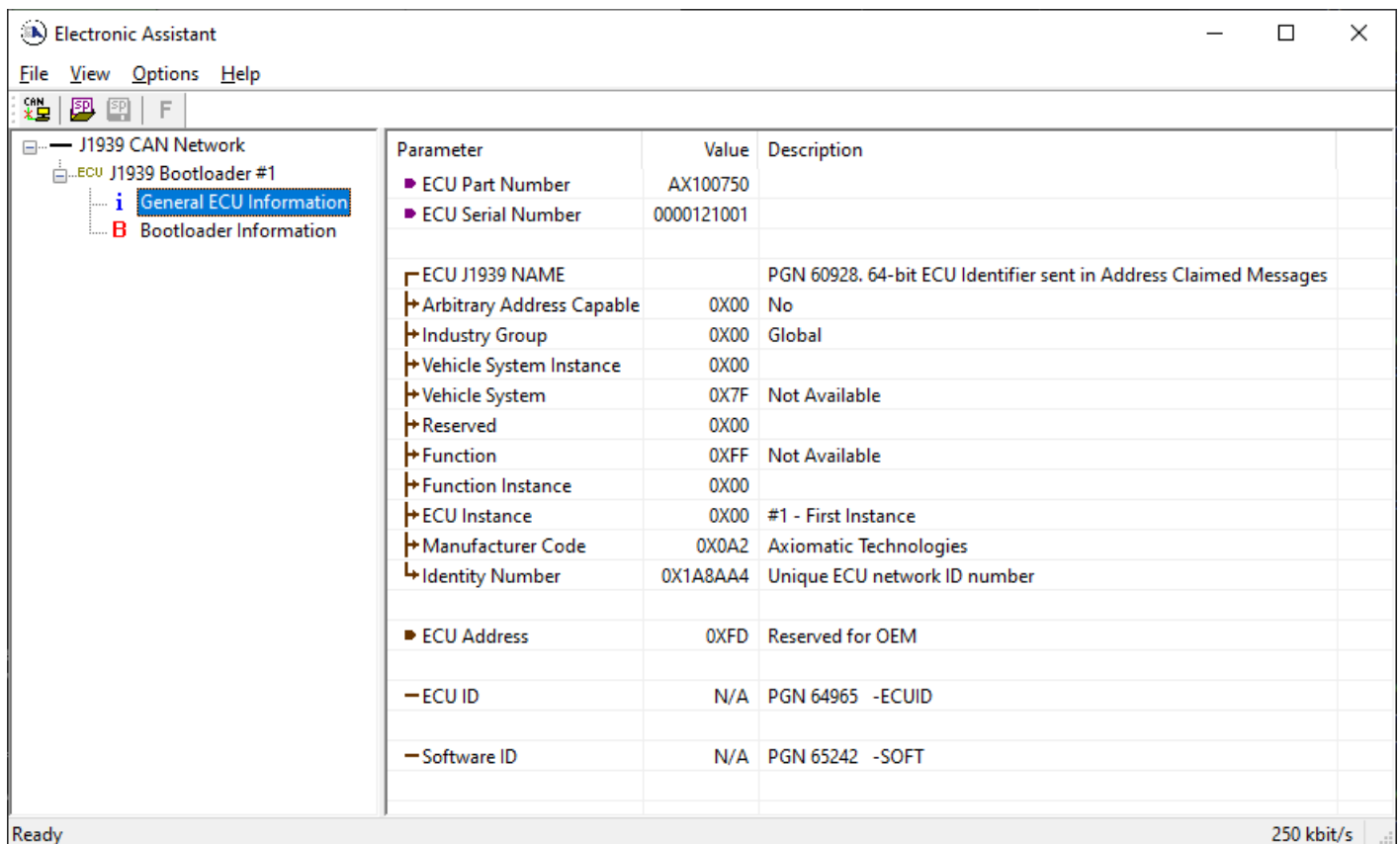
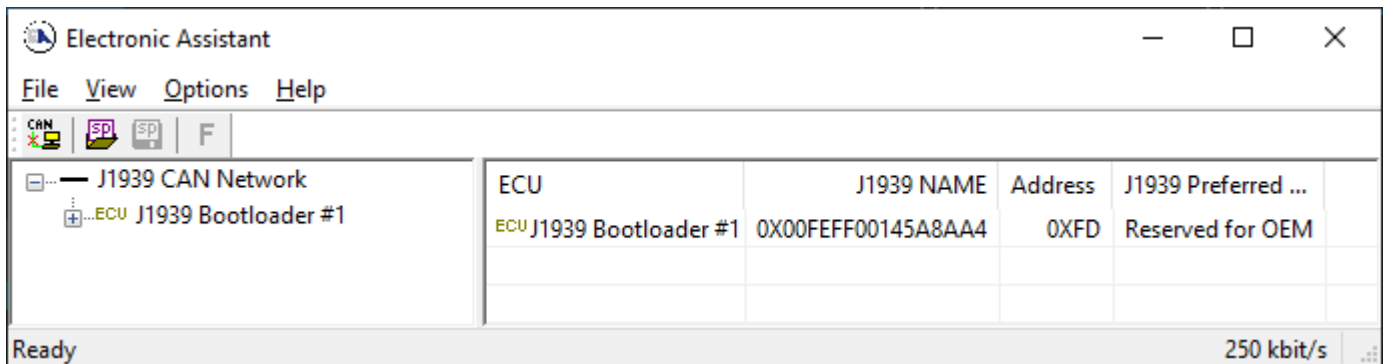


The dialog box is titled 'Force Bootloader to Load on Reset Setup'. It contains a dropdown menu for 'Force Bootloader to Load on Reset:' with '1 - Yes' selected. Below the dropdown, it shows 'Default Value: 1 - Yes' and a 'Set Default' button. At the bottom are 'OK' and 'Cancel' buttons.

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



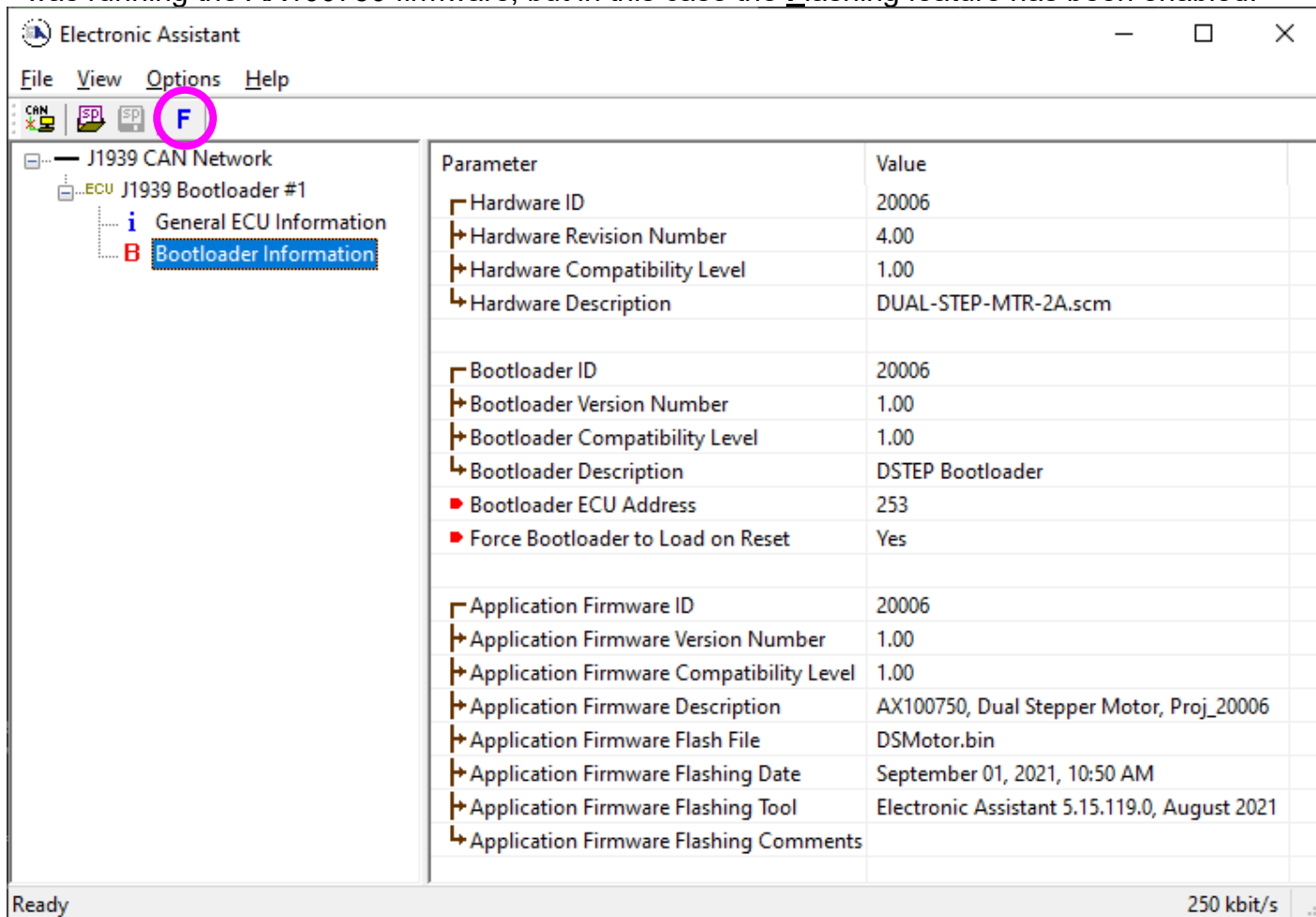
4. Upon reset, the ECU will no longer show up on the J1939 network as an AX100750 but rather as **J1939 Bootloader #1**.



Note that the bootloader is NOT Arbitrary Address Capable. This means that if you want to have multiple bootloaders running simultaneously (not recommended) you would have to manually change the address for each one before activating the next, or there will be address conflicts. And only one

ECU would show up as the bootloader. Once the 'active' bootloader returns to regular functionality, the other ECU(s) would have to be power cycled to re-activate the bootloader feature.

- When the **Bootloader Information** section is selected, the same information is shown as when it was running the AX100750 firmware, but in this case the **Flashing** feature has been enabled.



- Select the **Flashing** button and navigate to where you had saved the **AF-20006-X.XX.bin** file sent from Axiomatic. (Note: only binary (.bin) files can be flashed using the Axiomatic EA tool.)
- 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/timestamp the file, as this is done automatically by the Axiomatic EA tool when you upload the new firmware.

Flash Application Firmware [X]

Flash File Name:

Flashing Comments:
Press CTRL +ENTER to add a new string

☐ Erase All ECU Flash Memory

Flashing Status
Idle



WARNING: Do not check the “Erase All ECU Flash Memory” box unless instructed to do so by your Axiomatic contact. Selecting this will erase ALL data stored in non-volatile flash including the calibration from Axiomatic 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.

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.

Flash Application Firmware [X]

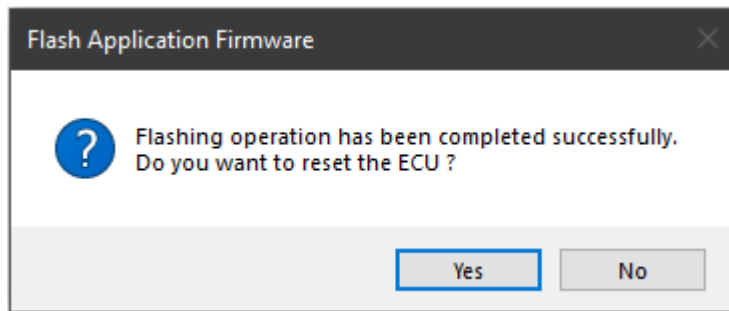
Flash File Name:

Flashing Comments:
Press CTRL +ENTER to add a new string

☐ Erase All ECU Flash Memory

Flashing Status
Flashing Memory...

Once the firmware has finished uploading, a message will pop up indicating the successful operation. If you select to reset the ECU, the new version of the AX100750 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 AX100750 application will run rather than the bootloader function.



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

6 TECHNICAL SPECIFICATIONS:

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

6.1 Power Supply Specifications

Power Supply Input	12 Vdc or 24 Vdc nominal 8...36 Vdc power supply range Maximum supply current is 15A @ 24Vdc at any time.
Protection	Reverse polarity protection is provided up to -80Vdc. Surge and transient protection are provided. Overvoltage shutdown of the output load is provided. Undervoltage protection (hardware and software shutdown) is provided.

6.2 Input Specifications

Universal Inputs	2 Universal Inputs Selectable as: Voltage, Current, PWM or Digital types Protected against shorts to GND or +Vcc	
	Analog Input	Analog input, user configurable <ul style="list-style-type: none">Voltage: 0-5V; 0-10V (Impedance 204 KOhm)Current: 4-20 mA; 0-20 mACurrent sense resistor 124Ω Inputs are sampled every 1 msec. Accuracy: +/- 0.5% over range Resolution: 1mV or 1uA, depending on type
	Digital Input	Digital input, user configurable for inductive or active sensors <ul style="list-style-type: none">DigitalFrequencyPWM Can be configured as active high or low digital input Digital Input: Active High to Supply or Active Low to GND Amplitude: 3.3V to +Supply User configurable for sampling range: 1 – 10,000 Hz PWM Duty Cycle: 0 to 100% Inputs are sampled every 1 msec. Accuracy: +/- 1% over range Resolution: 0.01 Hz or 0.1 Hz or 1Hz, depending on measuring range. 0.01% for PWM input Frequency/PWM @ 10 KOhm pullup or pulldown
Encoder Input	1 2-phase, phase A and B, incremental encoder input 1MΩ Impedance or Active High with 10K Pullup or Active Low with 10K Pulldown resistor to GND Range: 0.5 to 150 kHz Amplitude: up to +Supply	
Magnetic Pick-up Inputs	2 Pulse or PWM inputs, for inductive or active sensors Frequency Range: 0.5 Hz to 10 kHz Can be configured as active high or low digital input Protected against shorts to GND or +Vcc	
Signal Grounds	2 Analog GND (Input and +5Vref) 1 Magnetic Pick-up GND	


6.3 Output Specifications

Outputs to Motors	Two (2) independent outputs for stepper motors Allegro DMOS Microstepping Driver, A3979 Maximum 2A per phase for a Stepper Motor Overcurrent protection is provided by the driver up to 2.5A. User has full access to all the driver features via CAN messages.
Motor Disable	Motor can be disabled in a variety of ways, including "Motor Mode", and "E-Stop" input, or when an error is detected.
Motor Direction	Motor Direction can be user selected for Clockwise or Counter-Clockwise Drive. Refer to the User Manual for details.
Thermal Protection	Thermal protection is built-in to driver chip.
Output	One output selectable as: Proportional Current; Hotshot Digital; PWM Duty Cycle; Proportional Voltage; or On/Off Digital Half-bridge output, current sensing, High-side or Low-side load High side sourcing or low side sinking up to 2.5A Overcurrent protection Short circuit protection in hardware Current Output: Accuracy +/- 1% error Voltage Output: Accuracy +/- 5% error High frequency drive PWM Output: 0.1% resolution, accuracy +/- 0.1% error Digital On/Off: Sourcing from power supply or sinking to ground Load at supply voltage must not draw more than 2.5A. Note: Output 1 has dedicated E-Stop Brake hardware protection function.
Reference Voltage	+5V, 50 mA is available to power a sensor or potentiometer and is referenced to Analog GND

6.4 General Specifications

Microprocessor	STM32F407VGT7
Isolation	1 Isolated CAN port; 200Vrms isolation from power supply
CAN Port	1 SAE J1939 port, auto-baud-rate detection
Motor Drive Logic	Standard embedded software is provided.
Diagnostics	Each input and output channel can be configured to send diagnostic messages to the CAN network if the I/O goes out of range. Diagnostic data is stored in a non-volatile log.
Additional Fault Feedback	There are several types of faults that the controller will detect and provide a response: phase current winding problems; power supply undervoltage and overvoltage; and lost communication. They can be sent to the CAN bus.
Quiescent Current Draw	70 mA @ 12Vdc; 50mA @ 24Vdc typical
Weight	0.55 lbs. (0.25 kg) preliminary
Operating Conditions	-40 to 85 °C (-40 to 185 °F)
Storage Temperature	-55 to 125 °C (-67 to 257°F)
Protection	IP67, Unit is conformal coated in the housing.
Enclosure and Dimensions	High Temperature Nylon PCB Enclosure – (equivalent TE Deutsch P/N: EEC-325X4B) 4.62 x 5.24 x 1.43 inches 117.42 x 133.09 x 36.36 mm (W x L x H excluding mating plugs)
User Interface	Via the Axiomatic Electronic Assistant, P/Ns: AX070502 or AX070506K

VERSION HISTORY

Version	Date	Author	Modification
1.0.0	November 1, 2021	Viktor Rogachov	Initial Draft
-	November 3, 2021	A Wilkins	Marketing Review Updated footer address block Removed ®  from Electronic Assistant Formatted Technical Spec to align left Updated dimensional drawing
1.0.1	September 1, 2023	Kiril Mojsov	Performed Legacy Updates

OUR PRODUCTS

AC/DC Power Supplies
Actuator Controls/Interfaces
Automotive Ethernet Interfaces
Battery Chargers
CAN Controls, Routers, Repeaters
CAN/WiFi, CAN/Bluetooth, Routers
Current/Voltage/PWM Converters
DC/DC Power Converters
Engine Temperature Scanners
Ethernet/CAN Converters,
Gateways, Switches
Fan Drive Controllers
Gateways, CAN/Modbus, RS-232
Gyroscopes, Inclometers
Hydraulic Valve Controllers
Inclometers, 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