A KOHLER COMPANY

CURTIS

Manual
F Series Motor Controller

For Controllers: AC F4-A Model, AC F6-A Model
AC F2-A Model

» Software Device Profile: 4.5.0.0 «

Curtis Instruments, Inc.
200 Kisco Avenue
Mt. Kisco, NY 10549
www.curtisinstruments.com

Read Instructions Carefully!

53240, Rev A April 2022

TABLE OF CONTENTS

CHAPTERS

TABLE OF CONTENTS cont'd

TABLE OF CONTENTS cont'd

TABLE OF CONTENTS cont'd

TABLE OF CONTENTS cont'd

TABLES

TABLE OF CONTENTS cont'd

FIGURES

TABLE OF CONTENTS cont'd

The Curtis Difference
You feel it when you drive it

1 — OVERVIEW

The Curtis F2-A and F4/6-A motor controllers provide accurate, dependable, and highly efficient control of speed and torque of AC induction motors (ACIM) and permanent magnet AC motors (PMAC).

These traction controllers are optimized for both class III pedestrian-operated (F2-A) and rider-powered (F4/6-A) pallet trucks. They provide vehicle designers with the ability to fully define and control the detailed dynamic performance of their vehicle's drivetrain, and also provide comprehensive vehicle management and CAN manager capabilities. These models are also suitable for traction or hydraulic pump control on other types of battery powered vehicles, based upon the power requirements. The AC-F2-A is a 23-pin model. The AC F4/6-A is a 35-pin model offering a higher-count I/O and a second CAN port.

Together with the Curtis model 3140/3141 CAN LCD display and the Curtis Integrated Toolkit™, these Curtis models are the ultimate Class III truck control system.

Easily enhance any application by using Curtis's Vehicle Control Language (VCL). The setup and diagnostic tasks use the new PC-based Curtis Integrated Toolkit™ program (CIT) or the CAN-based Curtis 1313 handheld programmer (1313HHP).

Figure 1
AC F2-A Motor Controller

Figure 2
AC F4-A Motor Controller

Figure 3
AC F6-A Motor Controller

HOW TO USE THIS MANUAL

This manual combines the 23-pin F2-A and the 35-pin F4/6-A controllers.

The F-Series I/O operates the same way, the difference lies in the input/output (I/O) count. The 35-pin controllers offer a second CAN port and more I/O (drivers, analog & switch inputs) as the principle difference.

This manual describes how to:

- Properly mount and wire the controller
- Understand the configurable inputs and outputs
- Apply specific features to match an application
- Access and change parameters
- View and use monitor variables
- Perform the initial setup
- Setup the PDO Mapping
- Diagnose and troubleshoot faults
- Select and use the available programming and diagnostic tools

3141 CAN Gauge

WHAT IS NEW IN FOS 4.5.0.0 ... HIGHLIGHTS

1. Expanded the motor temperature sensor type to include the mapping of a custom sensor. See the parameter **Sensor Type** (0x3688) = 0. When selected, the sensor map menu will open in CIT.
2. IM Motor Types updated to 558. PMAC Motor Types updated to 149.
3. Added the instructions for using Hall switch motor position sensor with PMAC motors in Chapter 6.
4. Various FOS improvements. For example:
   - Continued Parameter/Monitor variables help text updates.
   - Fixes to main welded and motor open checks to correctly report motor open faults instead of overcurrent.
   - Fixed the monitor variable motor_control_active showing the wrong state. This variable can be used to verify the controller is actively controlling torque to the motor and is typically useful for manual control of an EM brake.
   - Added the EM Brake Test menu (Chapter 4).
   - Various faults, fault-actions updated, including for dual-drive applications.

For complete details on the FOS changes, consult the **FOS 4.5 OEM Release Notes** available from the Curtis distributor or the regional Curtis sales-support office.

GETTING THE MOST OUT OF YOUR CURTIS CONTROLLER

Thoroughly read and refer to this manual to apply and configure the controller. Understanding the installation & wiring guidelines, the parameter settings, the VCL functions, the initial setup & commissioning, and using the diagnostic and troubleshooting guide are critical to a successful application.

This manual describes the typical usage of the controller. If the intended application is not illustrated or discussed within this manual, contact the Curtis distributor or the regional Curtis sales-support office for additional technical support. The Curtis sales and support office-contact information is available from the Curtis Instruments website under the International tab. https://www.curtisinstruments.com/international/

2 — INSTALLATION AND WIRING

PHYSICALLY MOUNTING THE CONTROLLER

Use the controller's outline and mounting-hole dimensions to determine the optimum mounting location. Properly installed, the controller meets the IP65 requirements for environmental protection against dust and water. Nevertheless, in order to prevent external corrosion and leakage paths from developing, choose a mounting location to keep the controller as clean and dry as possible. Protect all ends of the low-power (signal) harness from excessive moisture and water (e.g., any toggle switch, sensor, and device connections on the vehicle).

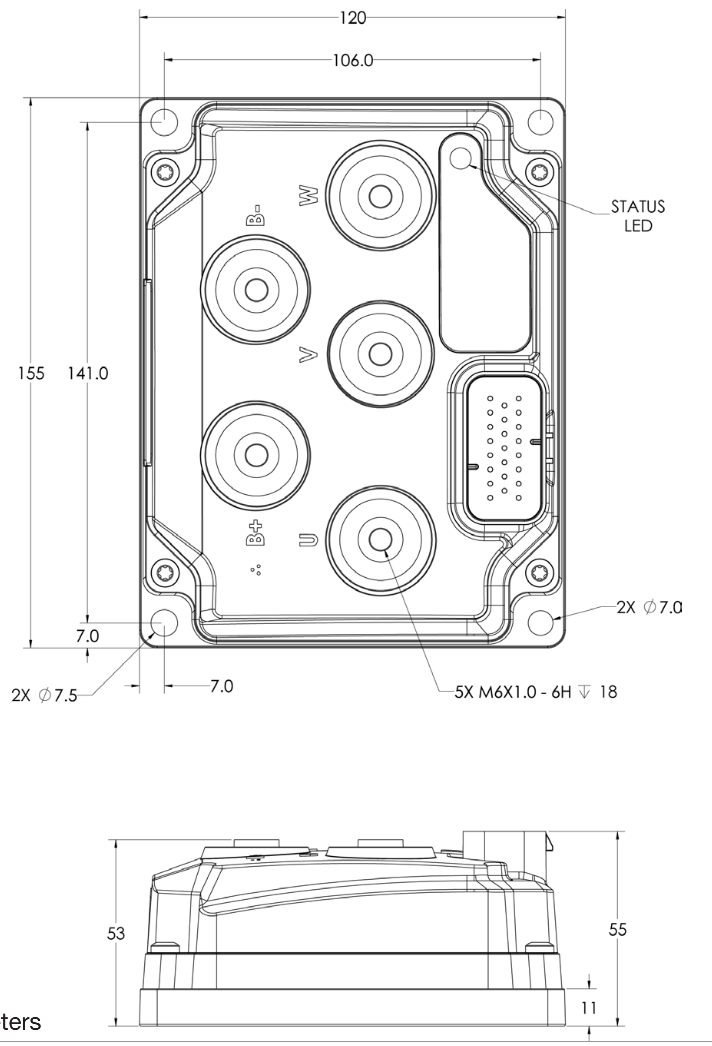
Do not pressure wash the controller or its connectors and terminals.

Mount the controller to a flat surface devoid of protrusions, ridges, or a curvature that can cause damage or distortion to its heatsink (base plate). Secure the controller using evenly torqued bolts to the vehicle's mounting surface. Applying a thin layer of thermal joint compound improves heat conduction from the controller heatsink to the vehicle's mounting surface. Typically, when properly mounted to a larger (smooth) metal surface, an additional heatsink or cooling fan is not necessary to meet the application's peak and continuous current ratings.

Apply a hot surface label to the controller compartment.

Figure 4

AC F2-A dimensions



Dimensions are in millimeters

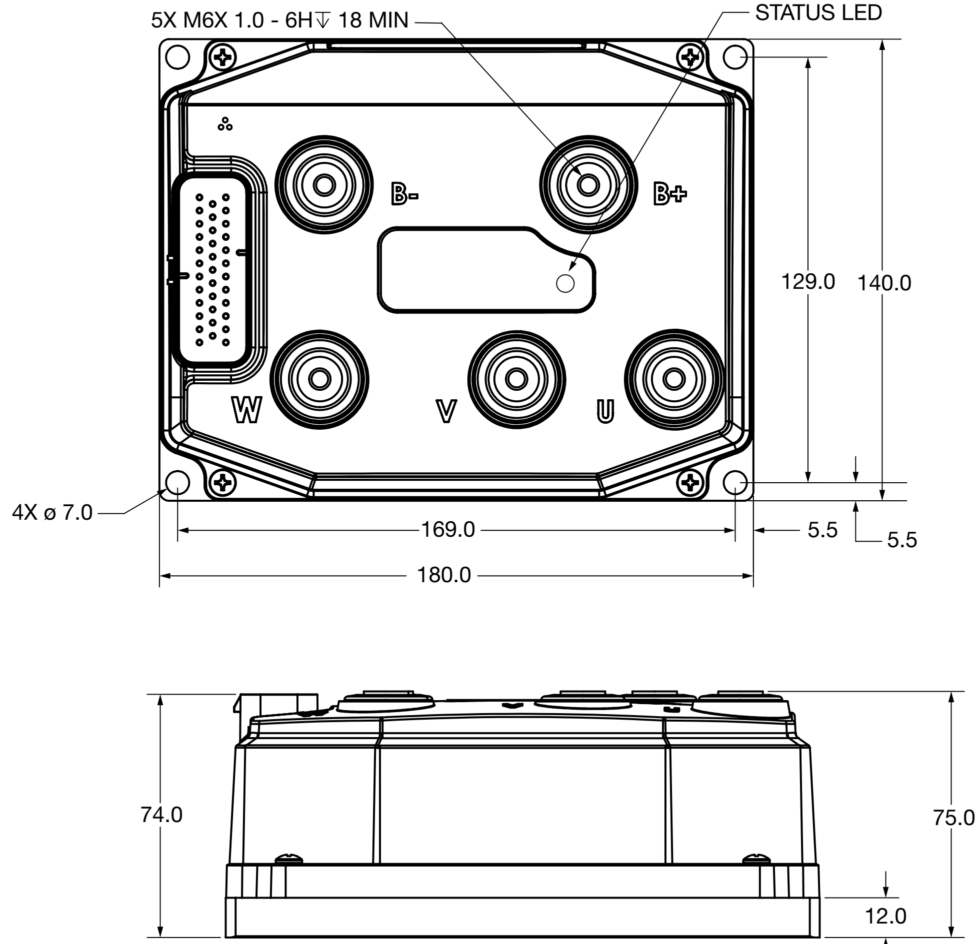


Figure 5
AC F4-A dimensions

⚠ CAUTION

Working on electrical systems is potentially dangerous. Protect personnel and property against uncontrolled operation, high current arcs, outgassing from lead-acid batteries, and voltage shock hazards.

UNCONTROLLED OPERATION — Some conditions could cause the motor to run out of control. Disconnect the motor or jack up the vehicle and get the drive wheels off the ground before attempting any work on the motor control circuitry.

HIGH CURRENT ARCS — Batteries can supply very high power, and arcing can occur if the terminals are short-circuited. Always open the battery circuit before working on the motor control circuit. Wear safety glasses, and use properly insulated tools to prevent shorts.

SHOCK HAZARD — The vehicle design (OEMs) shall prevent user access to the battery/protect controller terminals from a shock hazard, e.g., by mounting them in a compartment or behind a panel that can only be opened with a tool or a key.

LEAD-ACID BATTERIES — Charging or discharging generates hydrogen gas, which can build up in and around the batteries. Follow the battery manufacturer's safety recommendations. Wear safety glasses when servicing, charging and working around the battery.

LITHIUM ION BATTERIES — Follow the battery manufacturer's "safety precautions for the Lithium Ion battery pack." Wear safety glasses when servicing, charging and working around the battery.

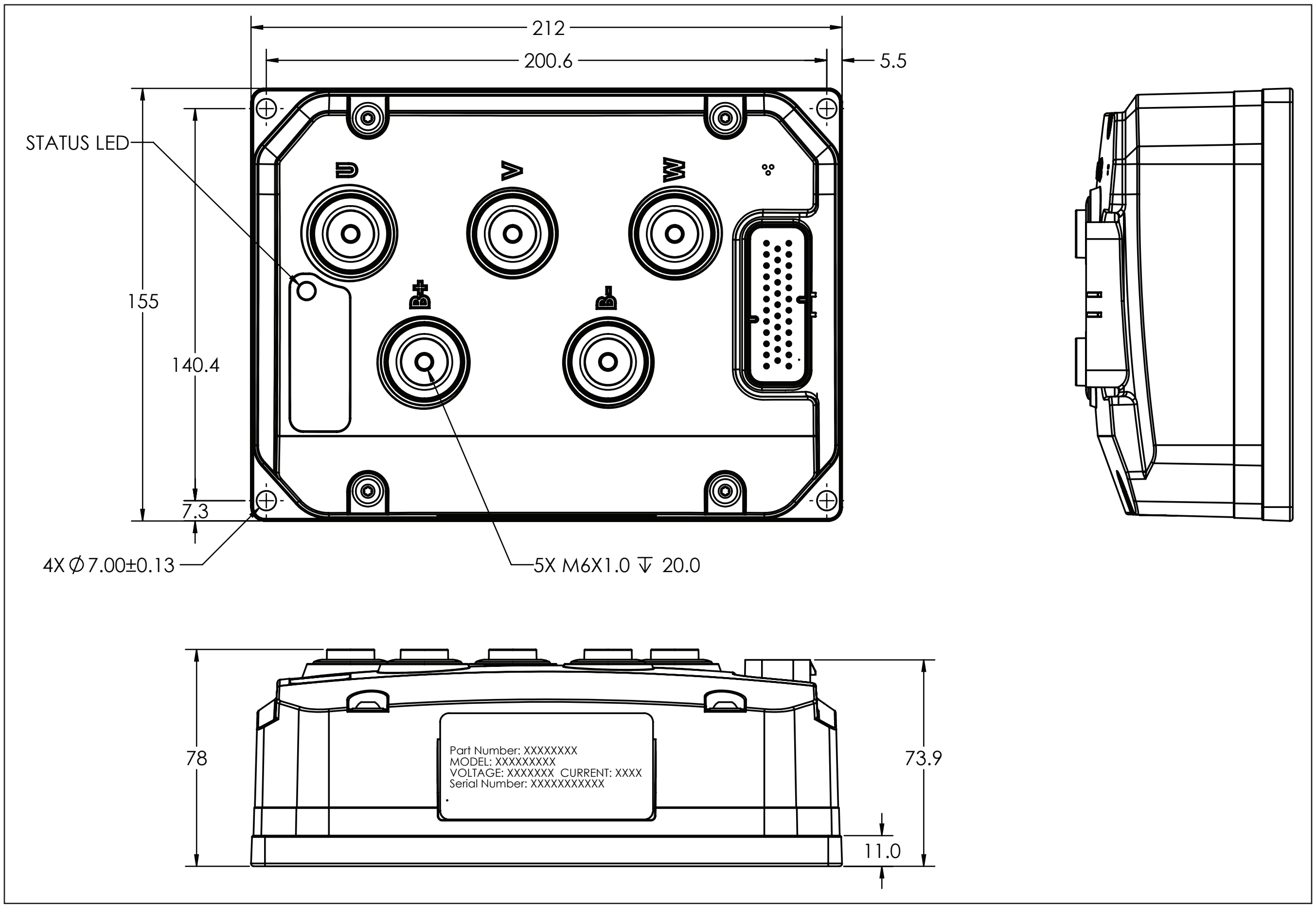


Figure 6
AC F6-A dimensions

PRECAUTIONS

Take the steps during the application's design and development to ensure that the EMC performance complies with applicable regulations; Appendix B presents some design considerations for EMC mitigation.

HIGH CURRENT CONNECTIONS

There are five high-current connections identified on the controller cover as B+, B–, U, V, and W. Install the controller with a (main) contactor between the traction battery positive terminal and the B+ terminal. This wiring circuit shall also be fitted with a fuse (the fuse amperage rating to match the application). Prevent user access to the terminals from a shock hazard, e.g., by mounting it in a compartment or behind a panel that can only be opened with a tool or a key.

Table 1 High Power Connections

Terminal	Function
B+	Positive Battery to Controller
B–	Negative Battery to Controller
U	Motor Phase U
V	Motor Phase V
W	Motor Phase W

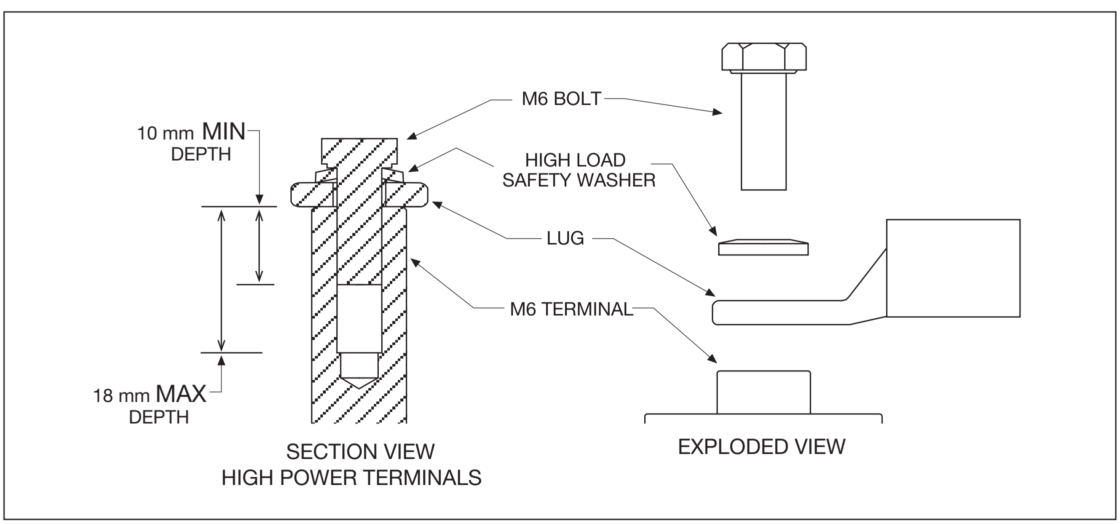
The high power connections are aluminum M6 terminals. Terminate the battery and motor cables with high quality (tin-plated) copper lugs to match the application. Ensure the M6 bolts are the proper length to meet the minimum thread depth engagement. Do not bottom-out the bolts in any terminal. Follow Figure 7 connection guidelines:

- Place the lug on top of the aluminum terminal, followed by a high-load safety washer with its convex side on top. The washer should be a SCHNORR 416320, or equivalent.
- For terminal connections with more than one lug, stack them so the lug carrying the least current is on top.
- Ensure the clamping bolt is within the minimum and maximum depth when assembled.
- Tighten the assembly to 10.2 ±1.1 Nm (90 ±10 in-lbs.).

When routing the battery cables between the battery and controller, run the positive and negative battery cables close to each other, avoiding pinch-points and areas of possible cable abrasion. Typically, the positive cable is red and the negative cable is black. The motor phase cables are often black, yet clearly labeled at both ends. Reference Appendix B, Vehicle Design Considerations for EMC guidelines.

Figure 7

Battery Power and Motor Phase Terminal Connections



LOW CURRENT CONNECTIONS

All logic and low power connections are through a single 23 or 35-pin AMPSEAL connector (mold in the cover) utilizing gold-plated pins. The matching AMPSEAL receptacle's wire silos come sealed by a membrane. Pierce the membrane by inserting the individual terminated wires. To maintain the IP65 rating, use the proper wire gauge and insulation thickness. Seal any non-used wire positions that have their silo-diaphragm pierced with the specific AMPSEAL sealing plug. Do not mix gold and tin pin types.

- Figures 8 and 9 are the AMPSEAL receptacle (plug) housing. Note the molded-in silo numbering.
- Figures 10 and 11 illustrate the connection assignments (as the plug aligns with the controller).
- Tables 2 and 3 list the matching AMPSEAL vehicle-harness component part numbers.

Figure 8

23-Pin AMPSEAL Connector

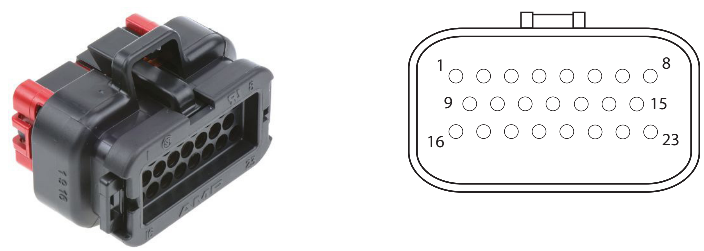
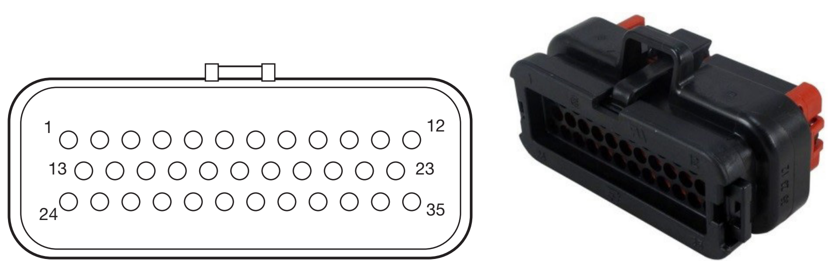


Figure 9

35-Pin AMPSEAL Connector



AMPSEAL 35 PIN Connector Harness-side view

Table 2 The AMPSEAL Connector Components & Part Numbers

Matching AMPSEAL 23-pin Component*	Part Number
AMPSEAL receptacle housing (the black vehicle-harness plug)	770680-1
Plug's gold-plated socket terminals (strip form p/n)	770520-3
Plug's gold-plated socket terminals (loose piece p/n)	770854-3
Silo seal plug (for non-used pin positions with a pierced membrane)	770678-1
Harness wire size (gauge)	0.5 – 1.25 mm^2 (20 – 16 AWG)
Wire diameter (overall) [i.e., uses wire with thin-wall insulation]	1.7 – 2.7 mm
Hand Crimper for the wire-harness socket terminals	58440-1

Note: The silo numbers are molded on the top and sides of the AMPSEAL connector receptacle (plug).

Table 3 The AMPSEAL Connector Components & Part Numbers

Matching AMPSEAL 35-pin Component*	Part Number
AMPSEAL receptacle housing (the black vehicle-harness *plug*)	776164-1
Plug's gold-plated socket terminals (strip form p/n)	770520-3
Plug's gold-plated socket terminals (loose piece p/n)	770854-3
Silo seal plug (for *non-used* pin positions with a *pierced* membrane)	770678-1
Plug's tin-plated socket terminals (strip form p/n)	770520-1
Plug's tin-plated socket terminals (loose piece p/n)	770854-1
Harness wire size (gauge)	0.5 – 1.25 mm^2 (20 – 16 AWG)
Wire diameter (overall) [i.e., uses wire with thin-wall insulation]	1.7 – 2.7 mm
Hand Crimper for the wire-harness socket terminals	58440-1

**AMPSEAL components and tooling are available worldwide from multiple TE Connectivity electrical component distributors.*
Reference the TE Connectivity Document: Application Specification 114-16016.
http://www.te.com/commerce/DocumentDelivery/DDEController
TE Connectivity website: http://www.te.com/usa-en/products/connectors/automotive-connectors/intersection/ampseal-connectors.html

Figure 10

23-Pin AMPSEAL Connection Assignments

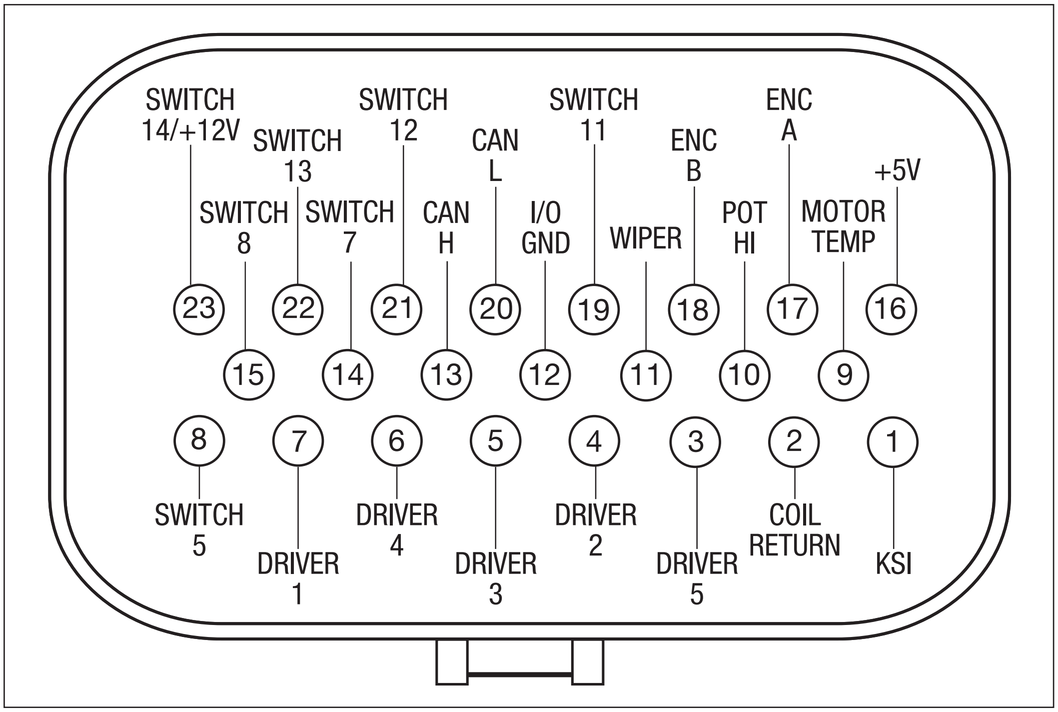
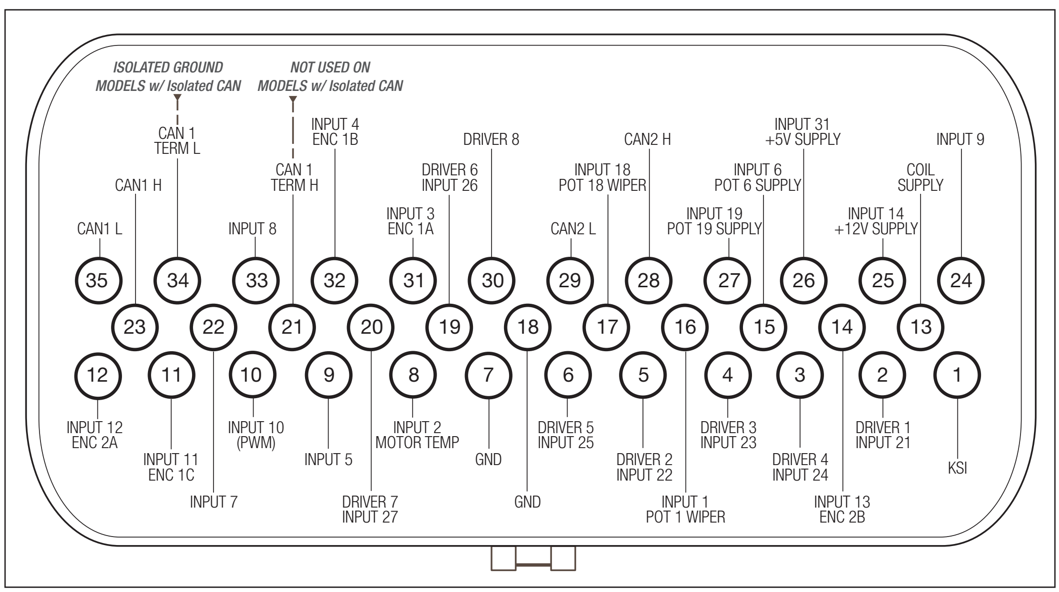


Figure 11

35-Pin AMPSEAL Connection Assignments



The Input and Outputs (I/O) assignments in Figures 10 and 11 follow the basic wiring diagrams. The orientation is looking at the controller, which is also the harness-side of the connector. When designing a vehicle harness and its routing throughout a vehicle, follow these guidelines to avoid common control-signal interference. Protect the wiring from abrasions due to vibrations, pinch, cut and pull-loose damage, which can lead to an inoperative controller. Reference Appendix B, Vehicle Design Considerations for EMC guidelines.

Rotor Position feedback

The rotor-position sensor's wiring (+5V, Feedback A, Feedback B, and Ground) should be bundled together between the motor and controller. These wires are often run with the rest of the low current wiring harness without interference issues, but the encoder wires should not be routed near the motor cables. In applications where this is not possible or there is signal interference, twist the encoder signal wires. In cases using a shielded cable (e.g., shielded 2-wire twisted-pair with drain-wire), only ground the shield-drain wire at the controller ground. In cases where this is not possible, consider a different controller.

CANbus

Use twisted-pair wires for the CANbus connections. Keep the CAN wiring away from the high current cables and cross them at right angles when necessary. In extreme cases, use shielded cable with the shield connected to the controller I/O ground and *only on the controller side for the non-isolated controller models.*

The 35-pin AMPseal controllers allow the implementation of both 11-bit identifiers (CANopen) and 29-bit identifiers (e.g., J1939), and/or different baud rates to support a high-speed CANbus for safety or data logging.

An isolated CAN option uses a separate (isolated) CANbus ground reference, which is useful on vehicles with CANbus systems operating at different battery voltages. It avoids common mode noise issues. The isolated ground is provided for twisted pair shielding, and may not be needed. When using shielded wiring, only connect the shield's drain-wire to isolated ground. On controllers with two CAN ports, the option always includes both CAN1 and CAN2 as isolated CAN.

All other low power wiring

Use standard vehicle-harness routing practices for the remaining connections. When designing the vehicle's wiring and routing, keep the inputs such as the throttle, temperature, and the motor feedback signals separate from controller's output lines such as the coil driver outputs. Avoid routing the low-power wiring parallel to the high power (and high current) battery and motor cables.

Protected voltages

The low-power pins' protected voltage ratings listed in Tables 4 and 5 are absolute and are not for normal operation. To prevent damage to the controller, do not connect (i.e., short circuit) the Coil Supply to battery negative. Further, do not connect (i.e., short circuit) the I/O Ground to battery positive.

Table 4 The AMPseal Connection Protected Voltages – 23 pin controllers

Controller Pin Number	Minimum Reverse Voltage	Maximum Voltage	ESD Air Discharge (non-contact)	ESD Contact (touch)
Inputs 3, 4, 5, 6, 7, 8, 9, 10, 11, 14, 15, 17, 18, 19, 21, 22, 23	Indefinite short to B+ and B–		±15kV	±8kV
CAN 13, 20	Indefinite short to B+ and B–			
Outputs 3, 4, 5, 6, 7, 9, 10, 16, 23	Indefinite short to B+ and B–			
KSI 1	Indefinite short to B+ and B–			
Coil Supply 2	Not Protected against short to B–	Indefinite short to B+		
GND or I/O Ground 12	Not protected			

Table 5 AMPSEAL Connections Protected Voltages – 35 pin controllers

Controller I/O Pin Number	Min (reverse) Voltage	Maximum Voltage	ESD Air Discharge (non-contact)	ESD Contact (touch)
Inputs 8, 9, 10, 11, 12, 14, 15, 16, 17, 22, 24, 25, 26, 27, 31, 32, 33	Indefinite short to B+ and B–		±15kV	±8kV
CAN (non-isolated) 21, 23, 28, 29, 34, 35 CAN (isolated) 23, 28, 29, 34, 35	Indefinite short to B+ and B–			
Outputs 2, 3, 4, 5, 6, 19, 20, 25, 26, 30	Indefinite short to B+ and B–			
KSI 1	Indefinite short to B+ and B–			
Coil Supply 13	Not Protected against short to B–	Indefinite short to B+		
GND or I/O Ground 7, 18	Not protected			

Controller Inputs and Outputs

Tables 6 and 7 list the available Inputs and Outputs (I/O) types by the AMPSEAL pin number.

Table 6 Logic and Low Current Connections-F2A

Pin Number	Pin Name	Wiring Diagram (generic)	Analog Input	Switch Input	PWM Driver
1	**Keyswitch** Input / Switch 20	KSI		Switch 20	
2	**Coil Return**				
3	**Driver5** / Switch 25	Pump		Switch 25	Driver5
4	**Driver2** / Switch 22	EM Brake		Switch 22	Driver2
5	**Driver3** / Switch 23	Main		Switch 23	Driver3
6	Driver4 / Switch 24			Switch 24	Driver4
7	**Driver1** / Switch21	Prop. Valve		Switch 21	Driver1
8	Analog5 / **Switch 5**	Interlock	Analog 5	Switch 5	
9	**Analog2** / Switch 2	Motor Temp.	Analog 2	Switch 2	
10	Analog6/Switch6/**Pot Hi**	Pot High	Analog 6	Switch 6	
11	**Analog1** / Switch 1	Pot Wiper	Analog 1	Switch 1	
12	Ground	GND			
13	CAN High	CANH			
14	Analog7 / **Switch 7**	Forward	Analog 7	Switch 7	
15	Analog8 / **Switch 8**	Reverse	Analog 8	Switch 8	
16	External +5V	+5V Ext			
17	**Analog3** / Switch 3	Enc1-A, Sin	Analog 3	Switch 3	
18	**Analog4** / Switch 4	Enc1-B, Cos	Analog 4	Switch 4	
19	**Switch 11** / Enc1-C	EMR N. C.		Switch 11	
20	CAN Low	CANL			
21	**Switch 12** / Enc2-A	EMR N.O.		Switch 12	
22	**Switch 13** / Enc2-B	Lower		Switch 13	
23	**Switch 14** / Analog14 / +12V	Lift	Analog 14	Switch 14	

Table 7 Logic and Low-Current Connections-F4/6-A

Pin Number	Pin Name	Special I/O (Alternative usage)	Digital Input (type)	Switch Input	Analog Input	PWM Driver
1	KSI	Input 20	Virtual			
2	Driver 1	Input 21	Driver			✔
3	Driver 4	Input 24	Driver			✔
4	Driver 3	Input 23	Driver			✔
5	Driver 2	Input 22	Driver			✔
6	Driver 5	Input 25	Driver			✔
7	GND	–				
8	Motor Temp	Input 2 (Analog 2)	Virtual		✔	
9	Input 5	– (Analog 5)	Virtual	✔	✔	
10	Input 10	(PWM Input)	Generic	✔		
11	Input 11	Enc 1C (Quad Encoder)	Generic			
12	Input 12	Enc 2A (Quad Encoder)	Generic			
13	Coil Supply	Input 30	Virtual			
14	Input 13	Enc 2B (Quad Encoder)	Generic			
15	Pot 6 Supply	Input 6 (Analog 6)	Virtual		✔	
16	Pot 1 Wiper	Input 1 (Analog 1)	Virtual		✔	
17	Pot 18 Wiper	Input 18 (Analog 18)	Virtual		✔	
18	GND	–				
19	Driver 6 (Digital)	Input 26	Driver			
20	Driver 7 (Digital)	Input 27	Driver			
21	CAN1 Termination (H)	–				
22	Input 7	– (Analog 7)	Virtual	✔	✔	
23	CAN1 H	–				
24	Input 9	– (Analog 9)	Virtual	✔	✔	
25	+12V Ext Supply	Input 14 (Analog 14)	Virtual	✔	✔	
26	+5V Ext Supply	Input 31	Virtual	✔	✔	
27	Pot 19 Wiper	Input 19 (Analog 19)	Virtual		✔	
28	CAN2 H	–				
29	CAN2 L	–				
30	Driver 8	(Analog Output)				
31	Enc 1A (Sin/Cos)	Input 3 (Analog 3)	Generic		✔	
32	Enc 1B (Sin/Cos)	Input 4 (Analog 4)	Generic		✔	
33	Input 8	(Analog 8)	Virtual	✔	✔	
34	CAN1 Termination (L)	–				
35	CAN1 L	–				

Controller Wiring Diagram (Examples)

Quick Links:
Appendix E p.262

The basic wiring diagrams for a Class III pallet truck use wired inputs from switches and potentiometers, driving the traction motor. The Interlock, Forward, Reverse, Lift, Lower, and the redundant Emergency Reverse inputs are by external mechanical switches pulled to KSI (B+). The traction motor feedback shall match the motor technology. The throttle input is a 3-wire potentiometer (which meets EEC fault protection). Beyond these assigned I/O usages, the available (non-assigned) switch inputs, drivers, and the analog output, are programmable to suit a diverse range of the controller applications.

The F4/6-A illustrates both the non-isolated and isolated CAN port options. See Appendix E for the controller models and specifications, which support these options. The F2-A does not offer an isolated CAN port option.

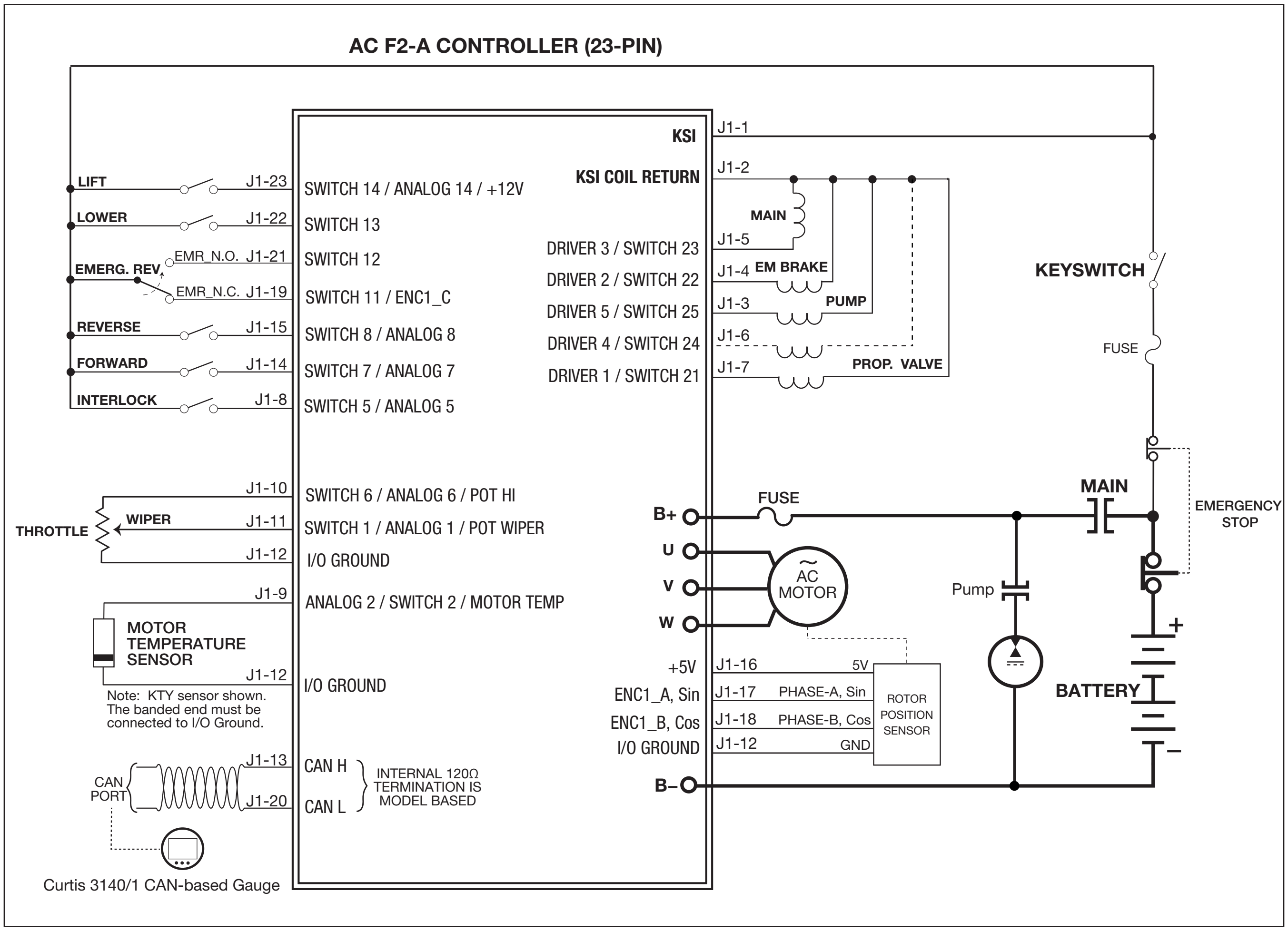
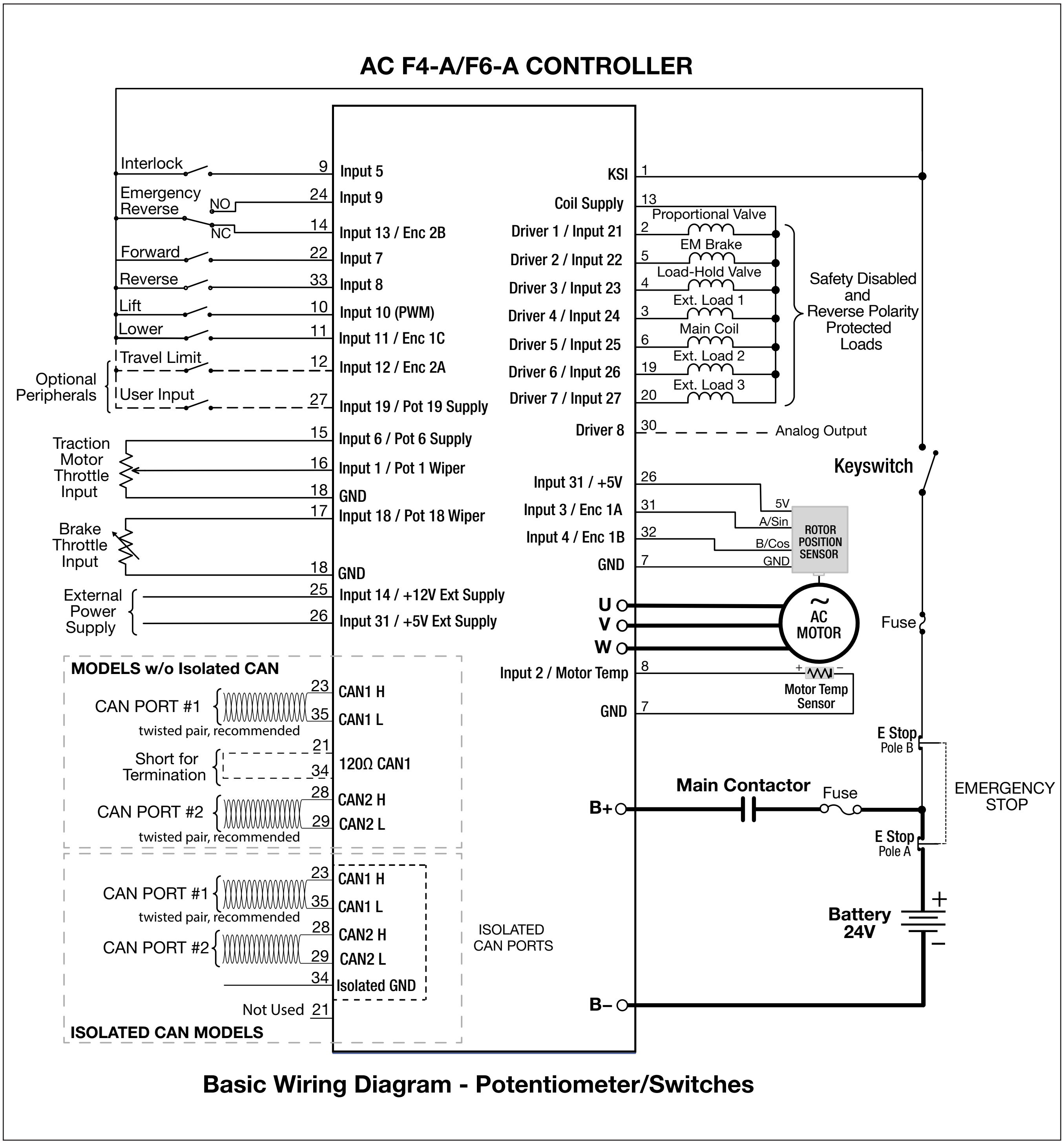


Figure 12

AC F2-A Basic Wiring Diagram



Basic Wiring Diagram - Potentiometer/Switches

Figure 13

AC F4-A, F6-A Basic Wiring Diagram

For CANopen based tiller heads (or similar), see the Figures 14 and 15 wiring examples. In these CAN-based examples, the Interlock, Emergency Reverse, and a Travel Limit are the remaining “wired” switch inputs. The CAN communication between the tiller and controller is always on CAN1.

Figure 15 illustrates the isolated CANbus option.

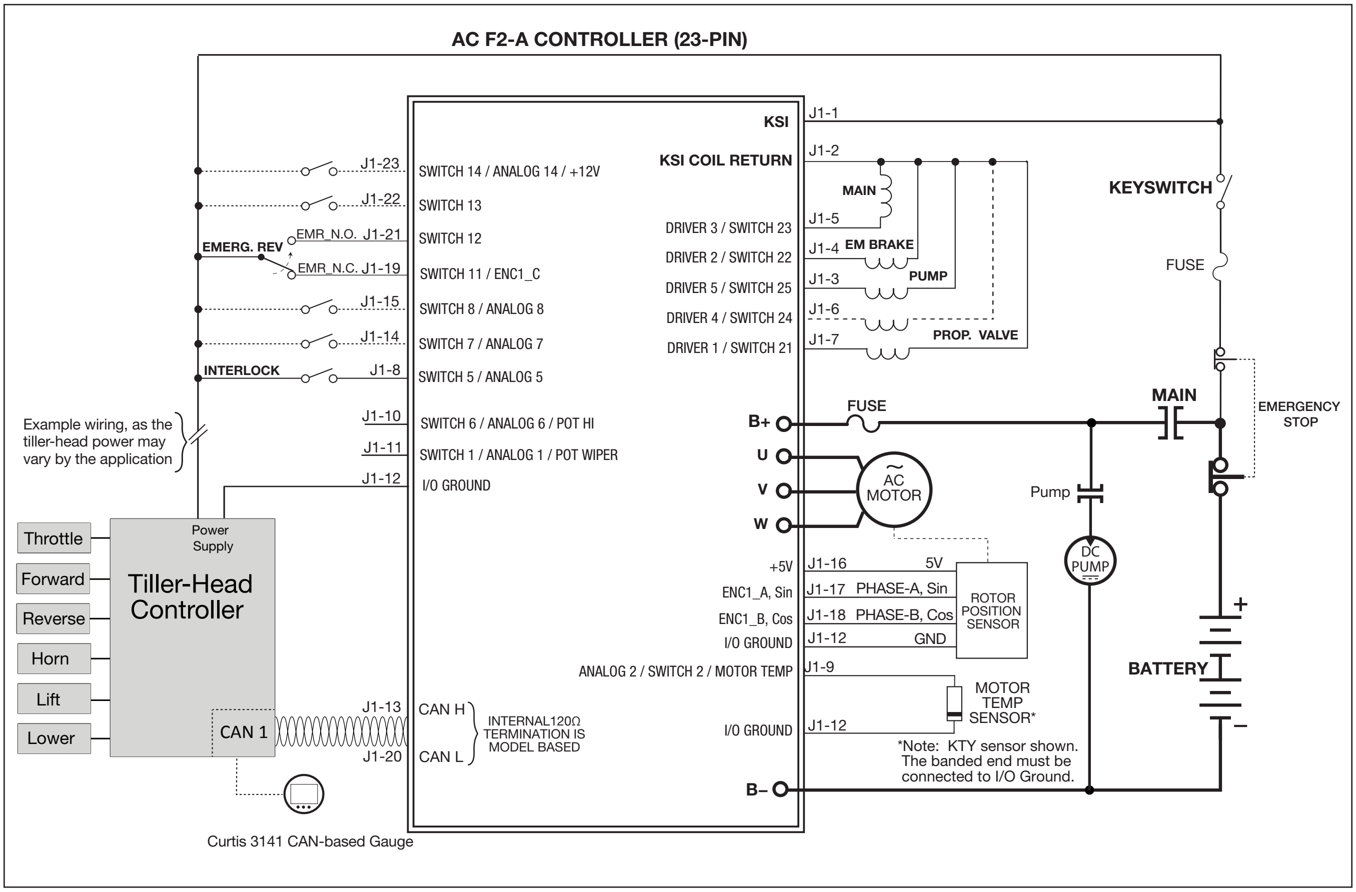


Figure 14

AC F2-A CAN Tiller Head Wiring Diagram

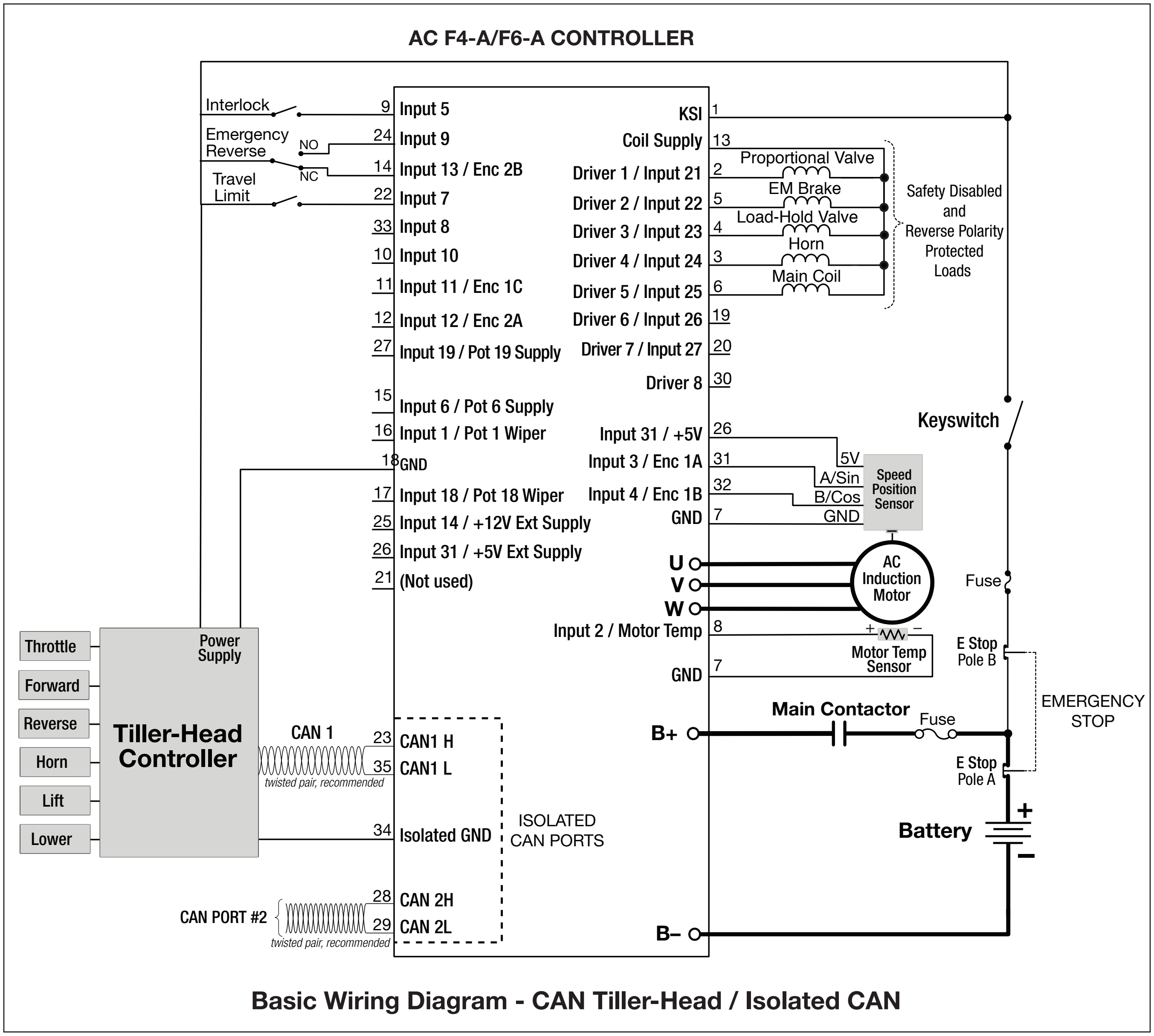


Figure 15

AC F4-A, F6-A Isolated CAN. Tiller Head Wiring Diagram

Low Current Connector Electrical Specifications

Digital (Switch) Inputs

The controllers offer flexibility in configuring the digital inputs. For example, when configured as switch inputs, the available inputs easily interface with user switches/buttons for the following:

- Interlock source.
- Vehicle directional input (forward and reverse).
- Forks lift/lower operations.
- Emergency reverse (NO and NC paired inputs).
- Reach or height limit switches.

The switch inputs in the wiring diagrams illustrate connecting these to B+ (keyswitch) voltage. This asserts the input to the "on" state. When the switch is open (not switched to B+), this is the non-asserted, or "off" state. Table 8 lists the inputs' specifications. Notice that these are principally analog inputs processing the applied voltage.

Virtual inputs

The input state is determined by processing the corresponding voltage from the analog input. These inputs have software configurable High/Low thresholds. By default, these thresholds are set to interface with Transistor-Transistor Logic (TTL) inputs; except for Input 1 & 13 (i.e. KSI and Coil Supply), for which the threshold is set at the brown-out voltage for the controller. These are reported as either On or Off. The virtual KSI monitor variable is *Switch 20* (0x3339 0x00). The virtual Coil Supply monitor variable is *Coil_Supply_State* (0x3C42 0x00).

Driver inputs

These are digital inputs that are located on the same pins as the driver outputs. These have a high threshold of 8V and a low threshold of 1V. These thresholds are as determined by the application's hardware/wiring configuration.

Generic inputs

These digital inputs are not multiplexed with the analog input (such as a pot wiper) or the driver outputs. These have a high threshold of 4V (On) and a low threshold of 1V (Off). These thresholds are as determined by the application's hardware/wiring configuration.

Table 8 Digital (Switch) Inputs Electrical Specifications

Signal Name	23-Pin	35-Pin	Logic Threshold[1]	Input Impedance
Switch 5	8	9	Rising edge = 4V max Falling edge = 1V min (Low / Off (pulled to B–)	24-36V models = 10kOhm ±10% 36-48V models = 18kOhm ±10%
Switch 7	14	22		
Switch 8	15	33		
Switch 9	n/a	24		
Switch 10	n/a	10		
Switch 11	19	11		
Switch 12	21	12		
Switch 13	22	14		
Switch 14	n/a	25		

Table 8 Digital (Switch) Inputs Electrical Specifications, cont'd

Signal Name	23-Pin	35-Pin	Logic Threshold[1]	Input Impedance
AVAILABLE/ALTERNATIVE SWITCH (DIGITAL) INPUTS				
Switch 1	11	16	(Throttle Wiper, typical usage) Rising edge = 4V max Falling edge = 1V min (Low/Off (pulled to B–)	> 20k Ω
Switch 2	9	8	(motor temp sensor, typical usage) Internal pull-up (A connection to I/O Gnd is < 2 mA)	Based upon usage. Contact Curtis.
Switch 3	17	31	(Motor Position Sensor, typical usage) Rising edge = 4V max Falling edge = 1V min (Low/Off (pulled to B-)	As Encoder: 5k Ω As Sin/Cos Sensor: > 50k Ω
Switch 4	18	32		
Switch 6	10	15	(Pot 6 Supply, typical usage)	> 20k Ω
AVAILABLE/ALTERNATIVE SWITCH (DIGITAL) PWM INPUT				
Switch 10	N/A	10	Selectable Low to High Frequency: 500 – 10,000 Hz Selectable Low to High Duty Cycle: 0 – 100% Ability to monitor selection as a percentage of settings.	See Switch 10, above

Frequency Measurement Range	500Hz – 10kHz
Frequency Measurement Accuracy	± 1%
Duty Cycle Measurement Range	10-90%[2]
Duty Cycle Measurement Accuracy	± 1%
Fault Diagnostics	• Input frequency out of configurable min/max limits • Input duty-cycle out of configurable min/max limits

Signal Name	23-Pin	35-Pin	Logic Threshold[2]	Input Impedance
Switch 21	7	2	(Coil Drivers, typical) Rising edge = 8V max Falling edge = 1V min (Low/Off (pulled to B–)	> 30k Ω (800 uA leakage current at nominal voltage)
Switch 22	4	5		
Switch 23	5	4		
Switch 24	6	3		
Switch 25	3	6		
Switch 26	n/a	19		
Switch 27	n/a	20		

SWITCH CLEANING CURRENT
≥ 2.0 mA at nominal battery voltage (24V model) ≥ 1.3 mA at lowest nominal battery voltage (36-48V model)

VCL Functions	VCL Monitor Variables
Enable_Emer_Rev( ) Disable_Emer_Rev( ) Setup_4p_Select( )	Switch_X (X = Switch#) [as assigned in Programmer] Any of the applicable, lift, lower, interlock, EMR input active, etc.

[1] *Logic thresholds are adjustable. See I/O parameter settings in Application and Controller Setup menus.*

[2] *The ability to measure duty cycles near 0% or 100% is a function of the voltage/frequency of the applied signal and the input capacitance that is added for protection against ESD. Lower input frequencies can allow reading closer to 0%/100% duty cycles. Test each application in its actual operating environment and implement with an acceptable margin of error.*

Analog Inputs

The controllers support a variety of analog inputs. The input's allowable voltage range varies depending on the primary purpose of the input. Select the analog input that matches the application.

Table 9 Analog Inputs Electrical Specifications

Input/Signal Name	23-Pin	35-Pin	Measurement Range[1,3]	Input Impedance (± 10%)
Analog 1	11	16	0 – 10 Volts	20k Ω (potentiometer)
Analog 2	9	8	0 – 5 Volts	5k Ω (Enc/Sin/Cos/Temp)
Analog 3	17	31		
Analog 4	18	32		
Analog 5	8	9	0 – 20 Volts	5k Ω
Analog 6	10	15	0 – 10 Volts	20k Ω (potentiometer)
Analog 7	14	22	0 – 20 Volts	5k Ω
Analog 8	16	33		
Analog 9	n/a	24		
Analog 14	23	25		
Analog 18	n/a	17	0 – 10 Volts	20k Ω (potentiometer)
Analog 19	n/a	27		
Analog 31	n/a	26	0 – 20 Volts	5k Ω
VCL Functions				**VCL Monitor Variables[2]**
				Analog_Input_Volts_X *Analog_Input_Percent_X* *Pot_X_Resistance*
				(*X = analog input#*)

[1] *The measurement margin is +4% / –0% margin. This is for analog usage.*

The full-scale accuracy is ± 2% over temperature (referenced to room temperature, 25°C).

The input signal filter is > 1 kHz for standard and pot inputs, > 40 kHz for encoder & sin/cos inputs.

[2] *When using potentiometer inputs, due to the dynamic tests (see text, below), the voltage reading is not constant. Use the input percent variable for the throttle or controls value.*

[3] *Increase voltage normalization range (analog_input_x_high) maximum limit to 30V for analog inputs, such that they can be used as digital (switch) inputs without causing a voltage out-of-range fault.*

Potentiometer Inputs with Dynamic Testing

In this manual, the term ***throttle*** is for the traction motors. If the vehicle uses a dual drive system, the single throttle input controls both traction motors. Figures 12 and 13 illustrate the traction-drive throttle as a traditional 3-wire potentiometer (pot) using Pot 1 Wiper, although Hall-effect voltage throttles are more common today. The brake throttle in Figure 13 is via a 2-wire pot. Using CAN-based throttles, as illustrated in Figures 14 and 15, frees the throttle (or brake) analog inputs for other usages. For example, in a dual-drive system, a CAN-based throttle frees a potentiometer input for use as the steer-angle input.

The three basic wired throttles use these analog input configurations:

Quick Links:
Fig 12 p.16
Fig 13 p.17
Fig 14 p.18
Fig15 p.19
Fig 36 p. 165
Fig 37 p. 166
Fig 39 p. 168
Analog Inputs p. 104

0 = Voltage input (see Table 9 and Figure 39 {voltage throttle in commissioning chapter})

1 = 3-Wire Pot (see Table 10 and Figure 36 {3-wire throttle in commissioning chapter})

2 = 2-Wire Pot (see Table 10 and Figure 37 {2-wire throttle in commissioning chapter})

In Programmer, when selecting Analog 1 Type as a 3-wire pot, Analog 6 becomes unavailable. When selecting Analog 18 as a 3-wire pot, Analog 19 becomes unavailable.

Note: The 3-wire potentiometer throttle provides complete throttle-fault protection that meets the applicable EEC regulations. For voltage throttles, the configured-pins protect against out-of-range input values, but do not detect external wiring faults; it is therefore the responsibility of the OEM to provide full throttle fault protection in vehicles using voltage, current, or CAN-based throttles.

The potentiometer inputs contain internal circuits, which dynamically test the wiper and pot high connections to enable detection of the following faults:

- Pot wiper connected to B+ at any time.
- Pot wiper connected to B– at any time (if this could produce movement).
- Pot high connected to B+ at any time.
- Pot high connected to B– at any time.
- Pot wiper shorted to ground at any time (if this could produce movement).
- Pot wiper shorted to pot high at any time.
- Pot high shorted to ground at any time.
- Other internal tests ensure the potentiometer inputs remains valid.

Therefore, when configuring an input as a potentiometer, always use the percent value for the reading, not the voltage. The voltage value will vary based upon the dynamic test (it is not for control usage).

2-Wire potentiometers use the wiper as the supply voltage while "reading" the voltage at the pin. The circuit is completed by the connection to the I/O ground. One end of the potentiometer is left unconnected. The controllers can configure either or both Analog 1 and Analog 6 as 2-wire potentiometers.

Table 10 Potentiometer Input/Configuration Electrical Specifications

Input Signal Name	23-Pin	35-Pin	Type	Pot Resistance Range Available Current	Input Impedance	Output Voltage	Fault Detection
Pot 6 Supply	10	15	3-Wire	1k – 10k Ω 3 mA supplied, Max.	20k Ω	< 15V	Pot Wiper Open Pot Resistance Low Pot Resistance High Circuit Failure (internal)
Pot 1 Wiper	11	16					
GND	12	18[1]					
Pot 19 Supply	n/a	27	3-Wire				
Pot 18 Wiper	n/a	17					
GND	n/a	18[1]					
Pot 1 Input	11	16	2-Wire	1k – 10k Ω 3 mA supplied, Max.	20k Ω	< 15V	Pot Wiper Open Pot Resistance High Circuit Failure (internal)
GND	12	18[1]					
Pot 6 Input	n/a	15					
GND	n/a	18[1]					
Pot 18 Input	n/a	17					
GND	n/a	18[1]					
Pot 19 Input	n/a	27					
GND	n/a	18[1]					

VCL Functions	VCL Monitor variables
	Throttle_Pot_Percent *Throttle_Command* *Throttle_Multiplier* *Mapped_Throttle* Note: Similar variables exist for the brake, if implemented. *Dual_Steer_Pot_Percent* *Dual_Steer_Angle*

[1] *Can also use pin 7 as a ground (GND).*

PWM and Digital Drivers

Drivers 1 through 5 are low-side pulse-width-modulation (PWM) drivers. These drivers are for inductive loads such as contactor coils and electromagnetic brakes. They can drive a resistive load if the peak current is within the driver's current rating. Use caution if the "load" is a RC-type circuit, however, the high (capacitor) inrush current (currents exceeding 120%) will cause a Type 2 Driver Overcurrent fault.

Each driver has a settable parameter (checks enable) to detect for an open and shorted coil (e.g., vehicle wiring related), and this parameter should be set to Off if the driver is not used. The drivers can withstand shorts to either B+ or B–. Always connect the drivers to the Coil Supply (pin 13) which is the high side of the driver circuit (i.e., these low-side drivers "sink current" from the coil supply, via the load). The Coil Supply provides an internal flyback-diode for the inductive voltage-spike protection.

Drivers 6 and 7 are lower-current digital (On/Off) drivers. Use the drivers for dashboard LEDs, piezo-electric buzzers and other low-current switched loads (i.e., high input impedance devices able to accept full coil supply voltage).

The controllers have two special purpose drivers.

- Driver 1 supports proportional valves, offering a higher frequency and finer current accuracy in addition to the typical dither-related parameters. The proportional driver's minimum duty cycle is 11%, because the current regulation is unavailable below this percentage.
- Driver 2 supports a 3A load for EM Brake usage.

Table 11 summarizes the drivers. Drivers 2–5 also support dither and current control, albeit at an accuracy of 15%. The parameter *Driver_Output_Frequency* collectively sets the PWM frequency for Drivers 2–5, whereas Driver 1 is fixed. To implement additional driver controls, use VCL. As noted in Table 8, these drivers are configurable as switch inputs, and are included in that group as well.

Table 11 Driver Outputs Electrical Specifications

Input Signal Name	23-Pin	35-Pin	Switching Side / PWM (Duty Cycle)	Frequency	Output Current[2]	Current Measurements[3]	Input Impedance
Driver 1	7	2	Low-Side (only) / 0-100% selectable[6]	18 kHz (fixed, ± 500 Hz)	2 Amps	40 mA – 2.6A[4]	> 30k Ω
Driver 2	4	5		200 – 2000 Hz[1] (adjustable)	3 Amps	450 mA – 3.9A[4]	
Driver 3	5	4		200 – 2000 Hz[1] (adjustable)	2 Amps	40 mA – 2.6A[4]	
Driver 4	6	3					
Driver 5	3	6					
Driver 6	n/a	19	Low-Side (only)[5] / 0 **or** 100% (On/Off)	N/A (no PWM)	1 Amp		
Driver 7	n/a	20					

VCL Functions	VCL Monitor Variables
Automate_Driver( ) Put_Driver( ) Battery_Compensate( ) — Driver_Output_Frequency	*Driver_Output_Frequency* *Driver_Pwm* *Driver_Voltage* *Driver_Pwm_Pull_In_And_Hold* *Driver_Voltage_Pull_In_And_Hold* *Driver_Current*

[1] The PWM Frequency parameter collectively sets Drivers 2–5 frequency (± 10%).
[2] The sum of all driver currents shall not exceed Coil Supply (pin 13) current rating.
[3] 2–130% of continuous rating. Minimum duty-cycle of 10% required for current measurement.
[4] Over-current shut down occurs at 120% of current rating in < 8 ms | 200% < 1 ms.
[5] Output Low Voltage: < 0.25V at full current and 100% PWM.
[6] Proportional Driver 1, the minimum current regulation is 11% duty cycle (operate > 11% duty cycle).
Coil impedance affects the lower limit of current control, where I = (11% PWM x Bat Voltage)/Coil Impedance (i.e., simplified steady state conditions, as the basic starting point of the proportional drivers current regulation limit).

Power Supply Outputs

The 5-volt and 12-volt power supplies provide auxiliary power for low power circuits such as electronic throttles, displays/gauges, and motor-position feedback devices. The corresponding ground is the controller I/O ground circuit, GND. Typically, the 5-volt supply is for the motor feedback device (encoder or Sin/Cos sensor) as illustrated in the wiring diagrams. For sensors and devices that can use 12 volts, the available supply current is the same. Use VCL to monitor the output currents for detecting sensor or vehicle-harness wiring faults. Each supply has parameter adjustable min/max current thresholds. By default, the 5-volt supply is On (1) and the 12-volt supply is Off (0).

Table 12 Power-Supply Outputs Electrical Specifications

Signal Name	23-Pin	35-Pin	Output Voltage	Output Current
5 Volt Supply	16	26	5 Volts ± 5%	100 mA[1]
				Measurement Range[2]
				2 – 105 mA
12 Volt Supply	23	25	12 Volts ± 15%	200 mA[1]
				Measurement Range[3]
				0 – 205 mA
VCL Functions				**VCL Monitor Variables**
				External_5V_Supply *External_5V_Current*
				External_12V_Supply *External_12V_Current*

[1] *Total current from both 5V and 12V supplies is 200 mA.*

[2] *Current accuracy.*
1mA (2mA - 50mA)
2mA (50 - 100mA)
Initial tolerance at 25°C. Allow +/- 1mA for temperature and aging.

[3] *Current accuracy = 10mA*
Initial tolerance at 25°C. Allow +/- 5mA for temperature and aging.

Keyswitch and Coil Supply

Connect the Keyswitch (KSI) input to B+ via a keyswitch. The keyswitch input feeds the controller's internal power supplies, the Coil Supply output, and the main-capacitor bank's precharge (before the main contactor closes). The lead-acid Battery Discharge Indicator (BDI) uses the keyswitch voltage.

Always connect the Coil Supply circuits (i.e., the contactors' B+ source) directly to the positive side (+) of the contactors' coil terminals so that the electrical-switching-noise associated with low-side drivers' pulse width modulation (PWM) operation is localized to the contactor wiring only. The controller includes an internal fly-back diode between each Driver and Coil Supply to suppress the coils' inductive voltage spike. Coils with their own inductive-spike suppression diodes are allowed, yet resistive means are discouraged because of leakage currents. Note, the cumulative sum of the Driver 1 through Driver 7 currents shall not exceed the Coil Supply's maximum continuous current rating (see Table 13, below).

It is important to maintain the division between KSI and Coil Supply in order to ensure reverse polarity protection (vehicle wiring correct, battery terminals reversed). Reference the wiring diagrams for the KSI and Coil Supply connections.

Table 13 Keyswitch and Coil Supply Electrical Specifications

Signal	23-Pin	35-Pin	Operating Voltage	Input Current
Keyswitch (KSI)	1	1	Between the under- and overvoltage cutback limits	< 12 Amps[1]
Coil Supply	2	13		12 Amps[1] 24V Models 10 Amps[1] 36-48V Models
AMPSEAL Connector Current Ratings			12 Amps per pin (Maximum, continuous)	
KSI Inrush Current			10 Amps, Max, ≤ 2 milliseconds overall, with initial peak ≤ 20 μs, 25°C	
Precharge Current			2 Amps for < 1 sec (typical)	
VCL Functions			**VCL Monitor Variables**	
			Keyswitch_Voltage	
			Coil_Supply	

[1] Includes current from the Coil Supply (full driver usage). Gold plated terminal basis, default models, peak/momentary. Tin plated terminals will have reduced current rating, i.e., 8 Amps. (On special models, so equipped with Tin pins).

CAN Ports

One or two CAN ports are available. Each CAN port can have a unique Node ID, which may be the same for both ports (if not connected to the same CANbus) and they can operate at different baud rates. On the 35-pin controllers, both isolated CAN and non-isolated CAN port models are available (see Appendix E).

Quick Links:
Fig 13 p.17
Fig 14 p.18
Fig 15 p.19
Appendix D p.252
Appendix E p.262

On the non-isolated 2-port CAN models, both ports share the controller internal power supply and I/O ground (B–) reference. On these models, enable the CAN1 port's internal 120Ω termination by externally connecting pins 21 and 34 together. The CAN2 port does not have internal 120Ω termination.

The models with 35-pins have an isolated CAN option. On these models, the CAN ports (CAN1 H/L and CAN2 H/L) are isolated from B+ and B–, but not from each other. The isolated CAN models provide an isolated CAN ground (pin 34 in Figures 13 and 15). These models do not have internal 120Ω termination. They also do not use pin 21.

The supported protocol is CANopen Physical layer ISO 11898, including the 11-bit and 29-bit identifier protocols. For applications with multiple third party devices, refer to the CANopen Physical Layer for the recommended bit timing settings and bus lengths based upon baud rate.

Use Port 1 (CAN1) for the communication channel with the Curtis Integrated Toolkit™ (CIT) program and the 1313 HHP programmer. See Appendix D for the CAN port hardware configuration to utilize these programming tools. This applies to both non-isolated and isolated CAN controller models.

There are 30 CAN receive mailboxes and 20 CAN transmit mailboxes. These are shared between both CAN ports. Attempting to assign more than this using the Assign_CAN_Mailbox() function will return an error. See the System Information file.

There are four TPDO and four RPDO mailboxes, each for CAN Port 1 and Port 2. See the CAN Interface and CAN Interface 2 menus in Programmer.

Table 14 CAN Ports Electrical Specifications

CAN Port	23-Pin	35-Pin	Baud Rate (bps)	CAN Termination
CAN1 High	13	23	–1 = 100k 0 = 125k 1 = 250K 2 = 500K 3 = 800K 4 = 1M	<u>Non-isolated 35 pin units:</u> Short pins 21 and 34 with external jumper engage 120 Ohm termination. <u>Isolated and 23 pin units:</u> 120 ohm termination is a hardware option.
CAN1 Low	20	35		
CAN2 High	n/a	21		Non-isolated & isolated CAN: No internal termination.
CAN2 Low	n/a	34		

VCL Functions		VCL Monitor Variables
Assign_CAN_Mailbox ( ) Setup_CAN_Transmit_Mailbox ( ) Setup_CAN_Transmit_Data ( ) Get_Transmit_Counter ( ) Get_Transmit_Status ( ) Clear_Transmit_Status ( ) Enable_Transmit_Mailbox ( ) Disable_Transmit_Mailbox ( ) Get_Fault_CAN_Id ( )	Send_Mailbox ( ) Setup_CAN_Receive_ Mailbox ( ) Setup_CAN_Receive_Data ( ) Get_Received_Counter ( ) Get_Received_Status ( ) Clear_Received_Status ( ) Enable_Receive_Mailbox ( ) Disable_Receive_Mailbox ( ) Get_Receive_Timeout ( ) Clear_Receive_Timeout ( ) Get_Receive_ID ( )	*CAN_NMT_State*

Motor Position Sensor Inputs

The controllers running cdev 4.0.0.0 (FOS 4.0 or later) operate AC induction (ACIM) and AC Permanent Magnet Motor (PMAC) motors. To accomplish this, the rotor position inputs accept various types of sensors.

- 2-channel Quadrature encoders.
- Open Collector (OC) encoders.
- Sine/Cosine Position sensors (encoder).
- Hall switch devices.

When configured as a digital Quadrature Encoder, the input is a 2kΩ pull-up resistor to +5V.

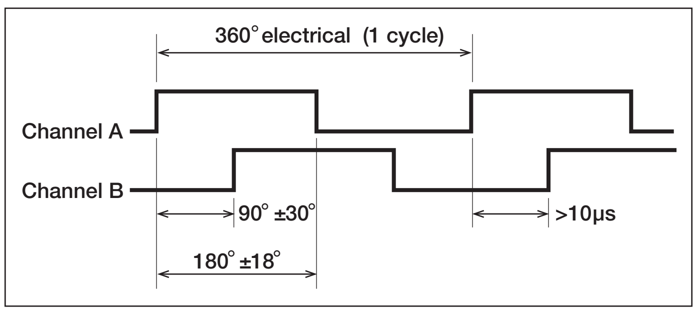
When configured as a Sine/Cosine Analog Encoder, the inputs are setup as analog inputs. These sinusoidal analog signals are mathematically converted to rotor position.

In all cases, connect the sensor to the controller's I/O ground. This is the signals' reference.

Table 15 Digital/Quadrature Encoder Electrical Specifications

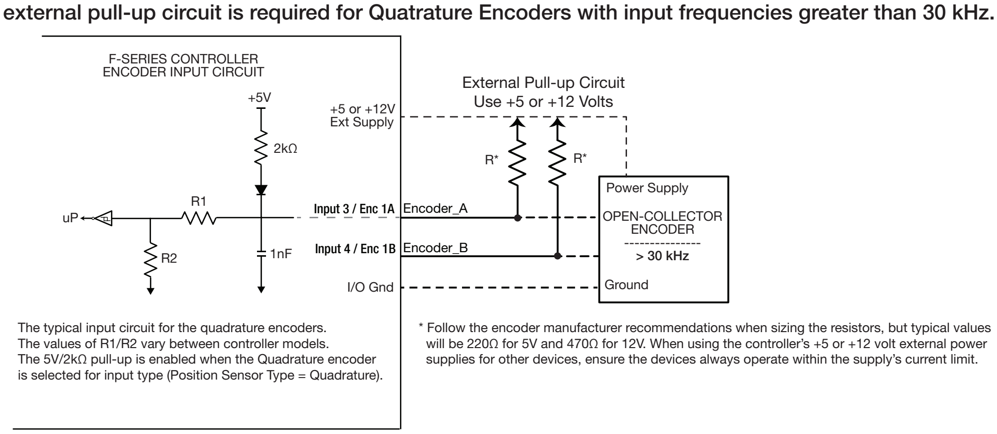
Signal Name	23-Pin	35-Pin	Input Voltage range	High/Low Voltage Threshold	Pull-up Resistance / Input Impedance	Maximum Frequency	A-B Phase Range	Phase Duty Cycle
Enc 1A	17	31	0 – 15V	4V max Rising-edge / 1V min Falling-edge	2k Ω to 5k / 5k Ω	250k Hz[2]	90° ± 30°	50 % ± 10 %
Enc 1B	18	32						
Enc 1C[1]	19	11						
Enc 2A	n/a	12						
Enc 2B	n/a	14						
VCL Functions					**VCL Monitor Variables**			
					Motor_RPM			

The application must maintain these illustrated signal tolerances throughout the application's operating conditions, including voltage, temperature, speed and torque ranges. ACIM applications use the quadrature encoder.



F-Series controllers using FOS versions 4.0 and higher are able to support encoder input frequencies up to 250 kHz. Encoders operating above 30 kHz, must use an external pull-up resistor circuit as illustrated. Use the controller's external supplies (5V or 12V) as a source for the pull-up resistors, while observing the external supply's current limits. Follow the encoder manufacturer recommendations when sizing the resistors, but typical values are 220Ω for 5V and 470Ω for 12V.

The external pull-up circuit is required for Quatrature Encoders with input frequencies greater than 30 kHz.



[1] *Specialty Input (future).*

[2] *Encoders >30kHz require an external pull-up (see circuit).*

Use a Sin/Cos sensor in PMAC applications.

A Sine/Cosine encoder is an absolute position sensor that produces two sinusoidal signals, set 90° out of phase. Measurement and comparison of the two signals at any point can determine the absolute position of the sensor. One 360-degree sin/cos sensor rotation per motor pole pair or 360-degree electrical rotation is also acceptable if the Enable Multiturn Sensor parameter (0x306D) is enabled. Matching sensor frequency to motor electrical frequency is strongly recommended.

The physical waveforms are not bipolar, but center around an offset voltage (Voff), typically around 2.5V. The Peak-Peak voltage (Vpp) swing may be as small as 0.625 volts or up to 4 volts (sensors with less than 2Vpp are not recommended). The larger the Vpp range the better the motor control performance. Both parameters require configuration for the sensor to operate properly.

The sin min, sin max, cos min and cos max parameters are configured during commissioning. These are used to normalize the incoming sin/cos signals.

Sine/Cosine Signal tracking

The values of the inputs are tracked to ensure an accurate position can be obtained if the signal amplitude changes during operation. The amplitude of the signal may alter due to mechanical tolerances, thermal expansion, magnetic field strength changes in the actuator magnet etc.

These signal tolerances must be maintained throughout the application's operating conditions, including voltage, temperature, speed and torque ranges.

The Sin/Cos sensor device must be set up with one waveform cycle per motor electrical cycle or one waveform cycle per mechanical revolution. For a one waveform per electrical cycle sensor, the Feedback_Multiturn parameter (0x306D) must be set to 1. One waveform cycle per electrical cycle is strongly recommended.

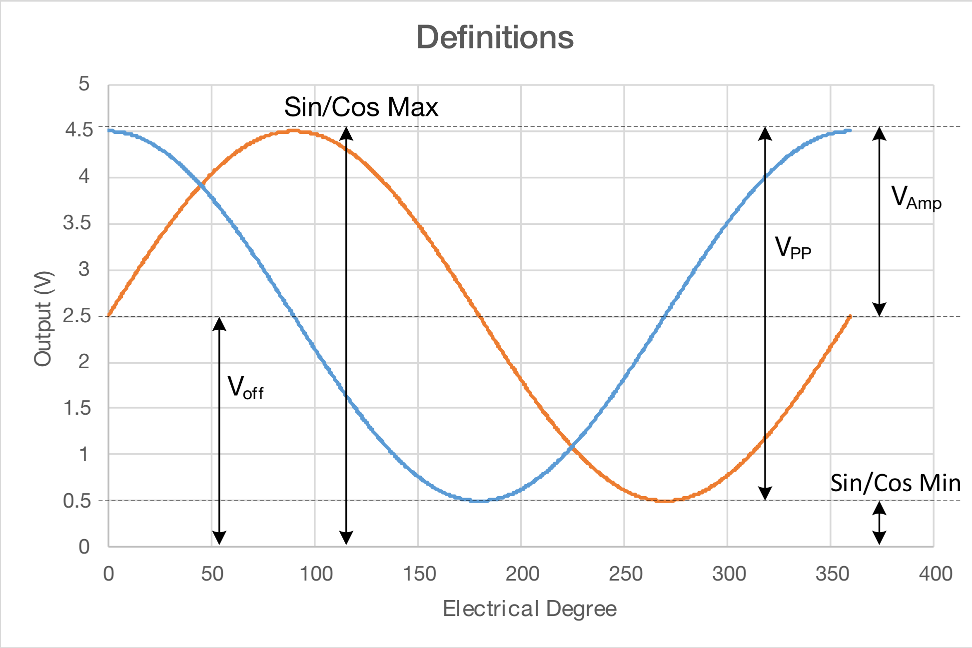
Table 16 Sin/Cos Sensor Device Electrical Specifications

Signal	23-Pin	35-Pin	Operating Voltage / Signal Range[3]	Max Input Frequency[1]	Recommended Propagation Delay[2] Phase Lag	Input Impedance
Sine	17	31	0.5–4.5V (+10% / –0%) Optimum: Multiturn sensor. 4Vpp with the signal centered at 2.5V.	599 Hz Sinewave	≤20° at 599 Hz ±0.5° Max difference between Sine/Cosine inputs	> 50k Ω
Cosine	18	32				
VCL Functions					**VCL Monitor Variables**	
					Motor_RPM *AD_Encoder_Sine* *AD_Encoder_Cosine* *Encoder_Sin_Calibrated* *Encoder_Cos_Calibrated*	

[1] *Max input frequency may not exceed max controller electrical fundamental frequency.*

[2] *Minimum propagation delay is necessary to maintain rotor alignment at high speed. Larger delays are acceptable for lower frequency operation. Consult Curtis with application specifics for more information.*

[3] *The maximum voltage difference between the Sin/Cos waveforms at their maximums and minimums is 78mV. If the voltages are > 78mV during commissioning the Type 84 Motor Characterization fault is triggered. Furthermore, their minimum voltage must be > 625mV. (i.e., Sin_Max-Sin_Min & Cos_Max-Cos_Min must be > 625 mV).*



Motor Temperature Input

The traction motor's temperature sensor input measures the resistance of the connected sensor. The controller supports KTY8x, and PT1000 temperature sensors. A parameter-VCL sensor configuration profile allows the usage of custom solutions, with a maximum source current of less than 2 mA. The controller is unable to use low resistance sensors, such as the PT100.

In all cases, connect the sensor's ground/negative lead to the controller's I/O ground.

Table 17 Motor Temperature Sensor Specifications

Signal	Pin	Temperature • Type[4]	Resistance Measurement Range	Accuracy
Motor Temp	8	–40 – 250°C [1] • KTY83-122 [2] • KTY84-130 [2] • KTY84-150 [2] • PT1000 [3]	250 – 5k Ω	± 20 Ω @ ≤ 2k Ω
VCL Functions			**VCL Monitor Variables**	
			Motor_Temperature *MotorTempCutback* *Analog_Input_Volts_2* *MotorTemp_Sensor1_Resistance* *MotorTemp_Sensor2_Resistance*	

[1] *LOS Mode: 100°C in case of sensor failure.*
80°C in case of disabled sensor.

[2] *±5°C full range temperature accuracy.*

[3] *±15°C for 24V F2-A, ±5°C for other models.*

[4] *When using custom sensors, note that maximum resistance measurement range supported in default configuration is 4k Ohms. Temperature sense accuracy does not apply to custom sensors.*
Custom sensor setup is via VCL. Consult Curtis.

3 — APPLICATION-SPECIFIC FEATURES

Some of the controller features affect more than the electrical connections or the parameter settings. This chapter provides background information on application-specific features, to assist the vehicle designer in the design and vehicle-development process.

MOTOR SPEED CONSTRAINTS

The maximum motor speed (RPM) the controller will allow is constrained by the number of motor poles, the encoder pulses per motor revolution, and the maximum speed constraint imposed by the Max Speed parameter. The *Max_Speed_Controller_Limit* is always the lowest of the *Max_Speed_Encoder_Limit*, *Max_Speed_Frequency_Limit*, and the *Max_Speed_RPM_Limit*.

The overall maximum motor speed is the least of the following three constraints:

1. Electrical Frequency Constraint

The controller's (fundamental) electrical frequency goes to 599 Hz. It accomplishes this by clamping the Max Speed allowed, using the equation:

Max Speed Frequency Limit = 71880/Number of Motor Poles

For example, an 8-pole motor running synchronously at 599 Hz can rotate to a maximum rpm of 71880/8 = 8,985 rpm (max). In this case, the control software will limit the max speed to 8,985 rpm for an 8-pole motor. There is a limited over-speed, should the motor exceed this speed (e.g., going down a hill). The controller will still attempt to produce the correct frequency for maximized torque and proper control; it will not simply clamp to 599 Hz.

2. Encoder Pulses/Revolution Constraint (quadrature encoder)

The maximum Encoder frequency the controller will accept without the external pull-up circuit is 30 kHz.[1]

To determine how fast the encoder steps constraint will allow the motor to spin, use the equation:

Max Speed Encoder Limit = 15000000/Encoder Steps

Due to the high encoder frequency ability, encoders with lower pulses per revolution (ppr) will not be the limiting factor in motor rpm (e.g., a motor with a 256-pulse encoder can run up to 58,593 rpm, which is greater than the firmware limit, yet a 1024-pulse encoder will be limited to 14,648 rpm).

3. Max Speed Parameter Constraint

The maximum motor speed the controller will allow is 30,000 rpm.

Max Speed RPM Limit = 24,000[2]

[1] FOS versions 4.0 and higher are able to support input frequencies up to 250 kHz, provided there is an external pull-up resistor for encoders > 30kHz. The external supplies (5V or 12V) may be used as a source for the pull-up resistor. Follow the Encoder manufacturer recommendations when sizing this resistor, but typical values would be 220Ω for 5V and 470Ω for 12V. See the external circuit in the Encoder specifications, Table 15.

[2] The variable, Max Speed Controller Limit *(0x30AF 0x00) displays this maximum limit. See the Motor Setup parameter menu.*
Note: In the case where the Max Speed parameter is the prevailing constraint, greater RPM is possible.
Contact the Curtis distributor or support engineer to discuss the particular application.

VOLTAGE LIMITS

The F-Series controllers have both hardware and parameter-based voltage limits. During regenerative braking, the system voltage increases as the motor acts like a generator to slow the vehicle. The overvoltage protection cuts back the regenerative braking (regen) to prevent damage to the traction battery and other electrical system components due to the increased voltage once it is above the normal-voltage region. Regen cutback typically occurs when the traction-battery is near full charge, rather than when it is near its discharged voltage. Conversely, as the traction-battery nears depletion, strong acceleration or load demands can lower the battery voltage. To prevent the controller from operating below its full capabilities, the undervoltage protection will reduce the drive current when the voltage is below the normal-voltage region. Both of these protection methods help to prevent vehicles and systems from operating at voltages above and below their design or the application's thresholds. Understanding the voltage limit regions will aid in setting the battery and speed controller related parameters.

The following conditions define the standard F-Series operating voltage ranges.

Condition	Operating Voltages
Severe over-voltage	150% of maximum nominal rating[1]
Over-Voltage	125% of maximum nominal rating[1]
Under-voltage	50% of maximum nominal rating
Severe under-voltage	40% of maximum nominal rating
Brownout	33% of maximum nominal rating

The voltage ranges, based upon model voltage, are in the voltage table below[2].

Nominal Battery Voltage	Brownout	Severe Under-voltage	Under-voltage	Over-voltage	Severe Over-voltage
24V	8V	9.6V	12V	30V	36V
24-36V	8V	9.6V	12V	50V	54V
36-48V	12V	14.4V	18V	63V	72V
48-80V	16V	19.2V	24V	100V	120V
72-96V	24V	28.8V	36V	120V	130V

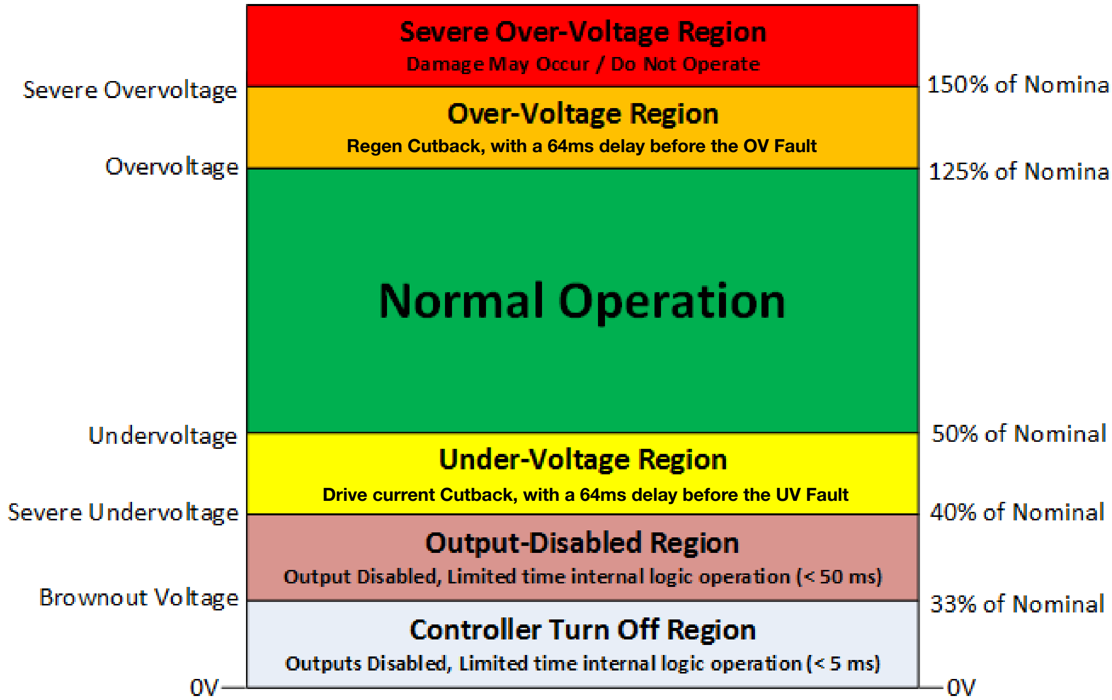
When the supply voltage is within the normal operation zone, the controllers operate with their full current and features. To narrow the normal operating window, for example, to tailor an application to work with a more restrictive battery, change the User Overvoltage and Undervoltage parameters.

Condition	User/Parameter Defined (Application Setup/Battery Setup/…Under-and Overvoltage parameters)
Over-voltage	Either the Maximum Voltage (voltage-range table, above), or User_Overvoltage × Nominal _Voltage, whichever is lower.
Under-voltage	Either the Minimum Voltage (voltage-range table, above) or User_Undervoltage × Nominal_Voltage, whichever is higher.

[1] There are slightly higher over-voltage limits for the 36V and 48V models than in this table's definitions. See the above voltage table.

[2] See Appendix E for the models and specifications. See the F-Series controller datasheets, available on the Curtis website, for new model updates.

Beyond the default or a redefined normal operating window, the controller's actions for over- and under-voltage conditions are both voltage and time-duration based. As illustrated in the graph and following paragraphs, these responses are fixed. No parameters can change or modify the hardware restrictions.



Over-Voltage Region

There is no delay on the Over-Voltage cutback. There is a 64ms delay on asserting the OV fault. The controller will seek to reduce the voltage by progressively reducing regen (motor-braking torque).

The Overvoltage Cutback fault (flash code 2-4) is set.

Severe Over-Voltage Region

Should the B+ voltage rise to this region, it triggers the Severe B+ Over-voltage fault (flash code 1-8).

Normal controller operation will cease.

Under-Voltage Region

There is no delay in the drive current reduction (motor torque) in this region. There is a 64ms delay before the Under-voltage Cutback fault is asserted (flash code 2-3).

Output-Disabled Region

This is a limited operation region. The output may be disabled and other functions may be disabled.

Motor operation is disabled. Displays may turn off and input devices stop working.

- After 5ms controller bridge is disabled, motor current shut off.
- After 64ms, CAN, external supplies and VCL execution is disabled, and the controller will reset when voltage recovers.

Brownout Voltage

The F-Series controller model determines the brownout voltage. It is fixed and not changeable. When the controller's keyswitch voltage falls into the brownout voltage region (33% > brownout ≥ 0V), the controller bridge is switched off (i.e., motor current is stopped). Functions such as the inputs, communications, and the external power supplies may also be disabled or go out-of-range. Once in the brownout region, the controller will react accordingly should the brownout-voltage condition recover:

- After 5ms controller bridge is disabled, motor current shut off.
- After 64ms, CAN, external supplies and VCL execution is disabled, and the controller will reset when voltage recovers.

The 24V F2-A may brown out sooner based upon loads.

KSI and B+ Input

Note that KSI (pin 1) and the B+ terminal (when the main contactor is closed) are at battery voltage. The controller's capacitor-bank pre-charge is via KSI, as the pre-charge function is performed prior to the main contactor's closure. An incorrect battery or parameter settings can trigger the various under- and over-voltage faults and controller responses.

BATTERY DISCHARGE INDICATOR

The lead-acid battery discharge indicator (BDI) algorithm continuously calculates the battery state-of-charge (SOC) from the keyswitch voltage (KSI, pin 1). The result of the BDI algorithm is the variable BDI that is the state-of-charge percentage. The BDI variable is viewable in the Programmer app's *System Monitor » Battery menu*. When the controller is powered-down (KSI off), the present BDI percentage is stored in nonvolatile memory.

For flooded lead-acid batteries and sealed maintenance-free lead-acid batteries, the standard values for volts per cell are as follows.

	Lead-Acid Battery Type	
	Flooded	**Sealed**
Reset Volts Per Cell	2.09	2.09
Full Volts Per Cell	2.04	2.04
Empty Volts Per Cell	1.73	1.90

Use these standard values for the battery's starting point in setting the Reset, Full, and Empty Volts Per Cell parameters (see the Programmer app, *Application Setup » Battery Setup » BDI Setup* menu).

Note: For non-lead-acid batteries, including Lithium-Ion battery packs, use the battery pack's or cell manufacturer's approved Battery Management System (BMS) for determining the SOC.

4 — PROGRAMMABLE PARAMETERS

PROGRAMMABLE PARAMETERS

The controller's programmable parameters enable the user to customize it to the needs of specific applications. VCL adds the option of changing applicable parameters during operation. To program (change the value) of parameters, a CANbus connection to the CAN1 port is required, using either the Curtis Integrated Toolkit™ program or the 1313-CANbus model handheld programmer (see below).

PROGRAMMING MENUS

Quick Links:
Table 18 p.41

Table 18 groups the programmable parameters into nested hierarchical menus, similar to the programming tools. Here, the menu charts contain descriptions of each parameter. Each parameter description includes specific features and/or the parameter's operation. The page-links in Table 18 allow quick access to specific menus, when using the PDF version of the manual. Otherwise, for printed copies, follow the indicated page number. Note that parameters are controller model dependent.

This manual lists all parameters based upon a 35-pin controller and a generic cdev. The available parameters or their range of options is both controller model and device profile limited. 23-pin controllers will populate the CIT/1313 menus appropriate to the model and will not have all the parameters listed in this manual.

PARAMETER CHANGE FAULTS [PCF]

Parameters marked [PCF] in the menu charts will set a Parameter Change Fault (code 4-9) if they are changed while the motor bridge is enabled (Interlock = On). Although the parameter value will change, the fault will prevent motor control functions until the Parameter Change Fault is cleared by cycling the keyswitch. If the motor bridge is disabled (Interlock = Off), changing these parameters will not cause a fault and the changes will take effect immediately. Exceptions are parameters that trigger the fault even with the Interlock Off. In either case, the value of the *Parameter_Change_Fault_Type* [PCF] fault will indicate the CAN Object Index of the parameter that triggered the fault. Note that if a parameter is changed in VCL and it is not obvious, search in CIT by the CAN object to identify the parameter.

NOTICE

Read Chapter 6, Commissioning, Initial Setup, before adjusting the parameters—it is imperative to perform the procedures outlined in Chapter 6, which sets up the basic system characteristics for an application.

MONITOR VARIABLES WITHIN PARAMETER MENUS

Some specific monitor variables are contained in their respective parameter menus, which helps in programming, as the parameter change is visible within the parameter menu. In addition, these monitor variables are in the Monitor menu (Chapter 5). Monitor variables are always Read Only.

PROGRAMMING TOOLS

The controller uses the same programming tools as the other F-Series controllers.

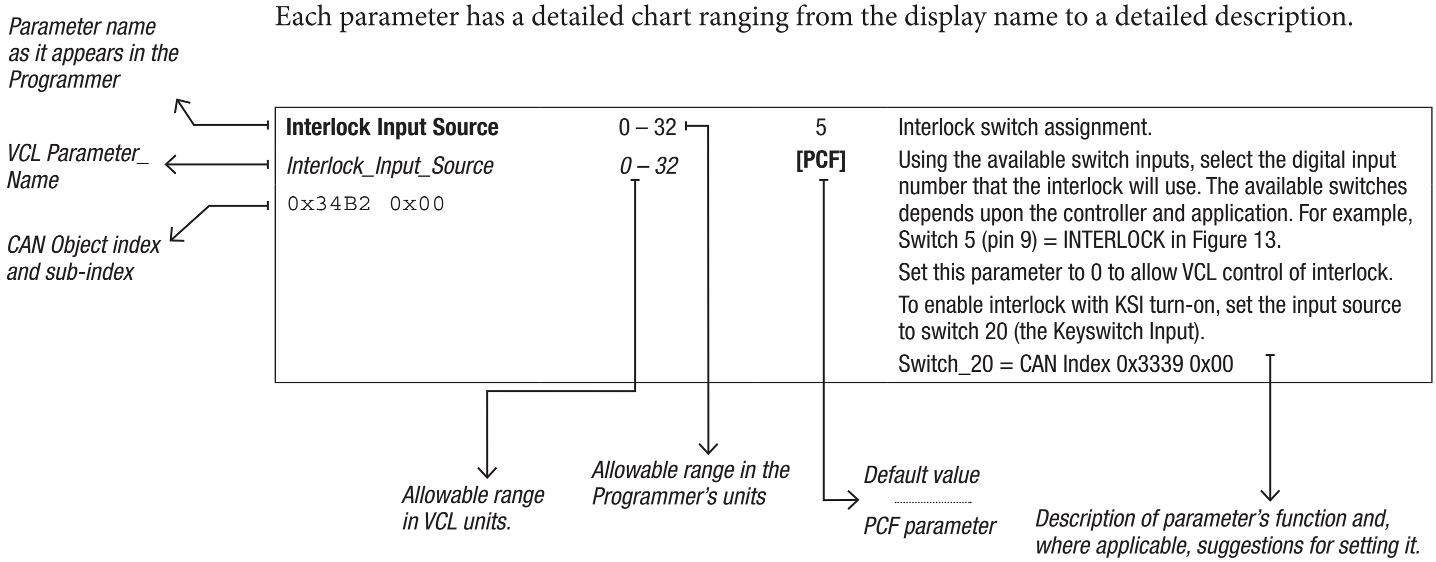
- The PC based Curtis Integrated Toolkit™ (CIT)
  - Requires a supported CAN dongle as listed in the toolkit's documentation.
- The Curtis 1313 handheld *(blue band)* programmer (1313 HHP)
  - Comes complete with the connector cable to mate to a female 9-pin D-Sub connector.

The controller does not have a serial port (it does not support parameter programming via a serial connection).

Contact the Curtis distributor or the regional Curtis sales office to obtain the Curtis Integrated Toolkit™ and the 1313 HHP. The distributor and Curtis offer training with the setup and using these programming tools.

MENU CHART FORMAT

Each parameter has a detailed chart ranging from the display name to a detailed description.



TERMINOLOGY

When setting parameters and commissioning the vehicle, follow these definitions.

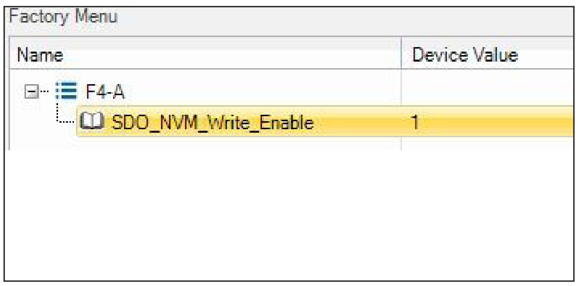
CiA/CANopen	CAN in Automation (CiA) is the international users' and manufacturers' group for the CAN network (Controller Area Network), internationally standardized in the ISO 11898 series. CANopen is a CAN-based communication system. The CANopen design was originally for motion-oriented machine control systems, such as factory handling system. Today, CANopen sees usage in various application fields, such as medical equipment, off-road vehicles, maritime electronics, railway applications, or building automation. References include CiA 301, 303-1, etc. The Curtis controllers are designed to use the CANopen features.
Curtis Integrated Toolkit	The Curtis Integrated Toolkit™ (CIT) is a Curtis Instruments developed software program for configuring and communicating with Curtis Instruments products. Use CIT to program (change and edit parameters, etc.) the Controller. See Appendix D for programming, monitoring, and diagnostic software and toolsets.
I/O	Input/Output. I/O generally refers to the controller AMPSEAL connector's input signals or switches, output signals, power, or low-side drivers. Controllers with different AMPSEAL pin-counts have different I/O assignments.
Object Index	The object dictionary is essentially a table that stores configuration and process data. The CANopen standard defines a 16-bit index and an 8-bit sub-index. The object dictionary is the method by which CANopen devices communicate. Every parameter and monitor variable has its own unique CAN Object Index. The parameter and monitor tables list each item's CAN Object Index.

Forward	Forward movement is a positive (value) traction speed. On a Class III truck with a tiller, "forks trailing" is the forward movement. For a reach truck or counterbalanced truck, "forks leading" is forward movement.
Reverse	Reverse movement is a negative (value) traction speed. On a Class III truck with a tiller, "forks leading" is reverse movement. For a reach truck or a counterbalanced truck, "forks trailing" is reverse movement.
PDO	PDOs (Process Data Objects) pack up to 8 bytes of data into highly efficient messages that are used to transfer run-time data between devices while in the Operational Mode. PDOs can be sent (TPDO) or received (RPDO) by any device. Most devices rely on PDOs for operational data transfer, but it is not mandatory. PDOs use the Producer-Consumer protocol. There can only be one Producer (transmitter) of a specific PDO COB-ID, but many (or no) Consumers (receivers). The PDO mapping structure uses the object index to define the data within any given PDO. See PDO Setup, Appendix A. Curtis uses the Manager for RPDO and the Ancillary for the TPDO.
RPDO	Receive Process Data Object (RPDO). Data received by the Consumer from Producer communication (e.g., the ancillary controller receives data from the manager controller).
RX	Receive. In CANopen, RX (Rx) is from the perspective of the ancillary controller.
TPDO	Transmit Process Data Object (TPDO). Data transmission by the PDO Producer to PDO Consumer (e.g., the ancillary controller(s) transmits data to the manager controller).
SDO	Service Data Object (SDO). A SDO is a low priority message used to transfer multiple data sets from a client to a server and vice versa. Several types of data transfer are available, with the Client (manager controller) taking the initiative for a transfer. Use the SDO process to read or write to an object index of a Server (ancillary controller). A SDO is used for configuring the controller via the CAN network. Defined within the Object Dictionary are the contents of the data set. See Appendix A for an example SDO to configure the PDO map.
SDO Download	Through this service, the client (e.g., manager) of a SDO downloads data to the server (e.g. the owner of the Object Dictionary). A "write" operation.
SDO Upload	Through this service, the client of an SDO uploads (reads) data from the server (owner of the Object Dictionary). A "read" operation.
TX	Transmit. In CANopen, TX (Tx) is from the perspective of the ancillary controller.
VCL	VCL is the unique Curtis Vehicle Control language. VCL provides the application level programmability to customize the usage, or allow Curtis AC motor controllers to perform as 'vehicle managers', eliminating the need for additional system controllers. See the Curtis website: Curtis Vehicle Control Language (VCL).

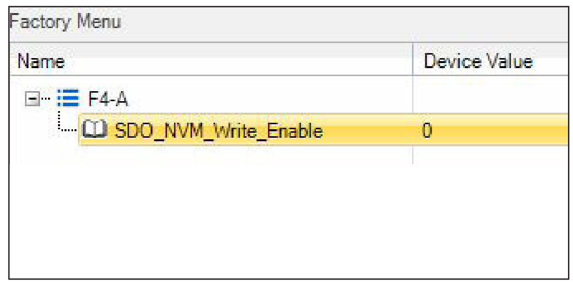
SDO WRITE MESSAGE

To retain parameter values in non-volatile memory (NVM) via CANopen SDO write messages, first write the value of one (01h) to the Least Significant [data] Byte (LSB) of the 32-bit parameter *SDO_NVM_Write_Enable* (Object Index 0x2008, sub-index 0x00). This will cause parameter changes to be written to non-volatile memory. Note that having the *SDO_NVM_Write_Enable* parameter set to zero (0) only saves the parameter changes to ephemeral (RAM) memory. RAM values are not stored over keyswitch cycles. Always return this parameter to the disabled state (= 0), when finished writing parameter values/changes. The state can be checked in CIT using the List View option in Programmer.

Use the CIT List-View image illustrated below (F4 example). In the first image, the controller (Node ID 0x26) is set to write parameter values to NVM (value = 1). In the second image (the default value = 0), the controller will only write parameter values to RAM.



SDO_NVM_Write_Enable = 1 (enabled)



SDO_NVM_Write_Enable = 0 (disabled)

In the example illustrated above, the SDO message to a controller with a Node ID = 26h (38d, 0010 0110b) is constructed by first writing a value of "1" to the *SDO_NVM_Write_Enable* parameter (enables), then a value of "0" (disables) as defined, below. Here, the data bytes are shown using the 1 – 8 nomenclature. It could as easily be labeled data bytes 0 – 7. The control byte uses the SDO-Rx 23h, as the parameter is 32-bits, so all data bytes are "written" even as the values in the data bytes are zero (00h). The message format is Little Endian, therefore data byte #5 is the LSB.

COB-ID

Function Code				Node ID of target Ancillary Controller						
1	1	0	0	0	1	0	0	1	1	0
SDO-Rx				Node 26h						
COB-ID = 626h										

SDO_NVM_Write_Enable (Object Index 0x2008, sub-index 0x00) = '**enable**' (01 = 1 in byte 5, the LSB)

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Control (write)	**LSB Object Index**	**MSB Object Index**	**Sub Index**	**Data**	**Data**	**Data**	**Data**
23h	08	20h	00	**01**	00	00	00

SDO_NVM_Write_Enable (Object Index 0x2008, sub-index 0x00) = '**disable**' (00 = 0 in byte5, the LSB)

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Control (write)	**LSB Object Index**	**MSB Object Index**	**Sub Index**	**Data**	**Data**	**Data**	**Data**
23h	08	20h	00	**00**	00	00	00

PCAN-View SDO Transmit messages, examples.

CAN-ID	Type	Length	Data	Cycle Time	Count	Trigger	Comment
626h		8	23 08 20 00 00 00 00 00	Wait	0		Write a zero (0) value to SDO_NVM_Write_Enable (0x2008.00) to disable.
626h		8	23 08 20 00 01 00 00 00	Wait	0		Write a one (1) value to SDO_NVM_Write_Enable (0x2008.00) to enable.

In this example, a PCAN-View message trace will result in the ancillary controller replying to the write message with the *SDO_NVM_Write_Enable* (Object Index 0x2008, sub-index 0x00), CAN-ID (COB-ID) 5A6h (10110100110b). Note that in the above message, the CANopen CiA 301 terminology of the SDO-Rx is used in the COB-ID Function Code.

CAN-ID	Rx/Tx	Type	Length	Data	
626h	Tx	Data	8	23 08 20 00 01 00 00 00	Enable *SDO_NVM_Write_Enable*
5A6h	Rx	Data	8	60 08 20 00 00 00 00 00	
626h	Tx	Data	8	23 08 20 00 00 00 00 00	Disable *SDO_NVM_Write_Enable*
5A6h	Rx	Data	8	60 08 20 00 00 00 00 00	

PCAN-View trace example.

CONTROL MODE SELECT INDEX

Table 18 Programmable Parameters Menus: Curtis Integrated Toolkit™/1313 HHP

PARAMETER CATEGORY AND MENU INDEX

Application Setup: menu index

Controller Setup: menu index

Motor Setup: menu index

NOTICE

Until the motor type is selected, AND the applicable motor characterization and setup is complete, the default CIT Programmer will show these motor related faults. Follow the parameter programming setup instructions related to each fault to clear and verify commissioning.

Default Faults: prior to motor selection, setup and commissioning.

- Motor Temp Sensor
- * Parameter Change
- * Motor Temp Hot Cutback
- Motor Setup Needed

** Controller and device profile basis.*

Note: The Analog 1 & 6, and 18 & 19 Type parameters are context sensitive based upon the selection as illustrated.

Note: The Analog 2, 3, 4, 5, 7, 8, 9, 14, and 31 are voltage only inputs as illustrated.

INDUCTION MOTOR (ACIM), cont'd	PMAC (PERMANENT MAGNET MOTOR), cont'd
— QUADRATURE ENCODER .... p. 137	— MOTOR SETUP STATUS .... p. 144
— 5V Output Enable	~ *Current Regulator Setup*
— 12V Output Enable	~ *Slip Gain Setup*
— Encoder Steps	~ *Base Speed Setup*
— Phasing Order	~ *Automated Test Run*
— Pullup Override	— SIN/COS ENCODER .... p. 144
— ENCODER FAULT SETUP .... p. 138	— Swap Motor Direction
— Fault Detection Enable	— Enable Multiturn Sensor
— Encoder Pulse Fault Detect Time	— Sin Cos Fault Threshold
— Fault Stall Time	— Sin Cos Fault Threshold High
— SPEED FILTERS .... p. 138	— Sin Cos Fault Time
— Speed filter Frequency	— Sin Cos Startup Time
— TEMPERATURE SENSOR .... p. 145	— Sin Min
— Sensor Enable	— Sin Max
~ *Temperature*	— Cos Min
~ *Motor Temp Cutback*	— Cos Max
— Braking Thermal Cutback Enable	— SPEED FILTERS .... p. 144
— Temperature Hot	— Speed filter Frequency
— Temperature Max	— TEMPERATURE SENSOR .... p. 145
— Motor Temp LOS Max Speed	— Sensor Enable
— Sensor Type	~ *Temperature*
— Sensor Offset	~ *Motor Temp Cutback*
— SENSOR MAP .... p. 145	— Braking Thermal Cutback Enable
— Sensor Resistance 0 - 6	— Temperature Hot
— Temperature 0 - 6	— Temperature Max
— Offset of Windings Due to Sensor Heating	— Motor Temp LOS Max Speed
	— Sensor Type
	— Sensor Offset
	— SENSOR MAP .... p. 145
	— Sensor Resistance 0 - 6
	— Temperature 0 - 6
	— Offset of Windings Due to Sensor Heating

CONTROL MODE SELECT

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Control Mode Select *Control_Mode_Select* `0x3080 0x00`	*Enumerated* *0 – 7*	0	This parameter determines which motor control method will be in effect when controlling the motor: 0 = Speed Mode Express 1 = Speed Mode 2 = Torque Mode Consult Curtis for these Control Modes: 3 = Direct Torque Mode (Nm) [1] 4 = Simplified Torque Mode [2] 5 = Steering Mode [3] 6 = PMDC open loop voltage/speed control 7 = Servo Speed Mode

Use speed and torque modes for either traction or pump motors. The speed modes are typically for material handling vehicle applications. Torque mode is for on-road vehicle applications.

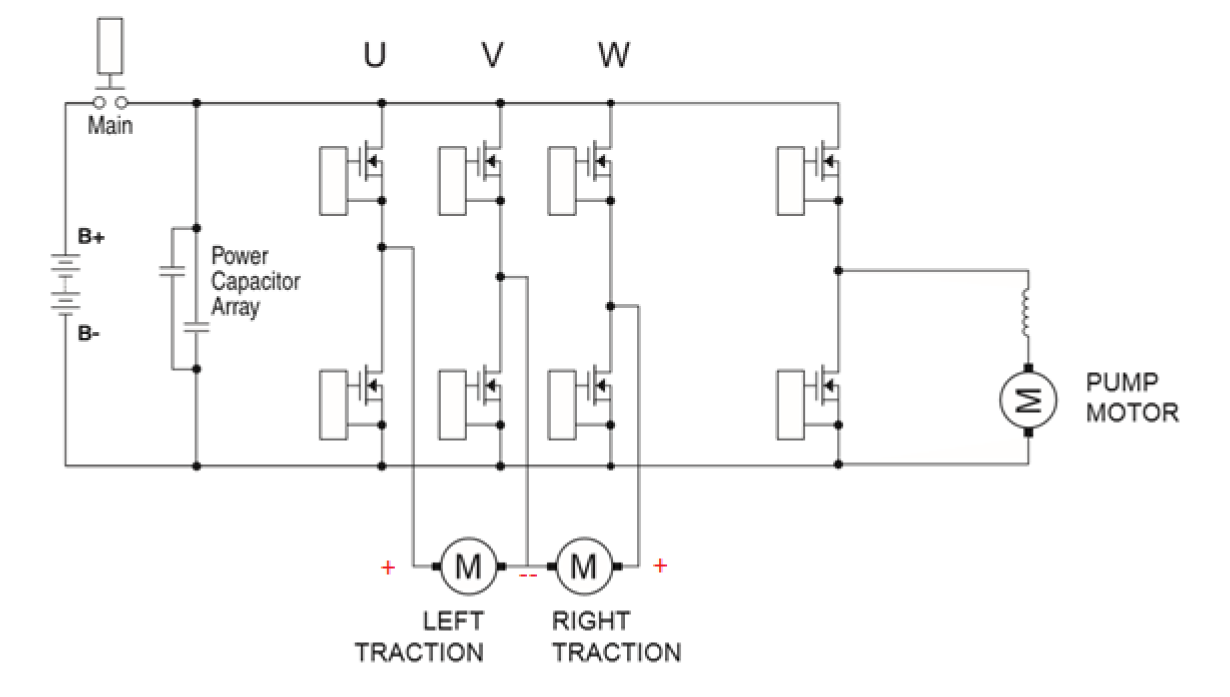
Note: Tune using the specific control mode, speed or torque, but not both. For example, if adjusting a torque control parameter while Speed Mode or Speed Mode Express has been selected as the tuning mode, the (CIT/1313) programmer will show the new setting but it will have no effect.

PMDC open loop voltage/speed control

This is the Motor Technology parameter (0x3534) value 6, PMDC (permanent magnet DC motors).

The diagram illustrates two permanent magnet DC motors connected to the three AC phases UVW.

Consult Curtis and the PMDC supplement manual before proceeding with this control mode.



Servo Speed Mode

A common application of this mode is expected to be Automated Guided Vehicles (AGVs).

Servo Speed Mode is intended for low latency, tight control of motor speed. The software is optimized for high resolution, high bandwidth control. Speed Mode and Speed Mode Express control modes provide an optimized out-of-the-box experience for operator controlled vehicles. While these modes provide enhanced vehicle drive ability, they may get in the way when engineers are seeking pure speed control for automated systems. Servo speed mode aims to be simple to set up and tune, with new tuning features and control structures that are easy to understand. However, a lot of the higher-level functions are not included, and using this control mode will require a greater integration effort.

Consult Curtis and the Servo Mode manual supplement before proceeding with this control mode.

---

[1] *For the Direct Torque mode, no normal vehicle control functionality is included, this mode is intended only if implementing vehicle control externally, either in VCL or using an external VCM.*

This mode must ONLY be used with motors Curtis has characterized in house and verified the torque estimate accuracy of.

[2] *The same as mode 2, but no normal vehicle control functionality is included. This mode is intended only if implementing vehicle control externally, either in VCL or using an external VCM.*

Note: Both modes 3 and 4 offer a unique hill hold function – please contact Curtis for more information.

[3] *The Steering Mode is not applicable to the F2-T/D controller.*

Speed Mode Express Parameter menus

Speed Mode Express is a simplified version of Speed Mode with a reduced set of parameters that is adequate for most speed-controlled applications. Speed Mode Express offers a less complex setup when multiple modes are employed (i.e., different maximum speeds, acceleration rates, or braking rates based upon vehicle operating conditions).

Note: The Curtis Integrated Toolkit™ Programmer App does not lockout the Speed Mode parameters. To prevent conflicts, follow the images here, paying attention to the parameter's VCL name where:

- **Speed Mode Express** parameters end in "_Spd**Mx**".
- **Speed Mode** parameters end in "_Spd**M**".

The six basic Speed Mode Express parameters whose VCL Parameter_Names end with "*_SpdMx*" cannot be substituted with the similar Speed Mode parameters which end in "*_SpdM*" in VCL programs. Always use the VCL parameter name as shown in the selected Curtis Integrated Toolkit™ control mode selection.

Within the Speed Mode menus, parameters devoid of the "_SpdM" apply to the Speed Mode Express setup. Therefore, after completing the Speed Mode Express parameters, proceed through the Speed Mode menus to set those parameter that may apply to the application. Otherwise, leave at the default value.

Quick Link:
RPM Constraints p.32

SPEED MODE EXPRESS — SPEED CONTROLLER MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Max Speed *Max_Speed_SpdMx* `0x3866 0x00`	100 – 24000 *100 – 24000*	4000 rpm	Defines the maximum requested motor rpm at full throttle. Partially applied throttle is scaled proportionately; e.g. 40 % applied throttle corresponds to a request for 40 % of the set Max Speed Value. NOTE: The maximum motor rpm is subject to the Traction-Motor Speed constraints.
Kp *Kp_SpdMx* `0x3865 0x00`	5 – 100 % *50 – 1000*	30 %	The proportional term (Kp) determines how aggressively the speed controller attempts to limit the speed of the motor to Max Speed. Larger values provide tighter control. If Kp is set too high the system may experience oscillations as the controller tries to control speed. Setting Kp too low may result in an overshoot beyond Max Speed.
Ki *Ki_SpdMx* `0x3864 0x00`	5 – 100 % *50 – 1000*	30 %	The integral term (Ki) forces zero steady state error so the motor will run at exactly the commanded speed. Larger values provide tighter control. Oscillations may occur when setting the gain too high, as the controller tries to control the speed. If it is set too low the motor may take a long time to approach the exact commanded speed.
Accel Rate *Accel_Rate_SpdMx* `0x3861 0x00`	0.1 – 30.0 s *100 – 30000*	2.5 sec	Sets the rate (in seconds/Typical Max Speed) at which the speed command increases when throttle is applied. Larger values represent slower response.
Decel Rate *Decel_Rate_SpdMx* `0x3863 0x00`	0.1 – 30.0 s *100 – 30000*	10.0 sec	Sets the rate (in seconds/Typical Max Speed) that is used to slow down the vehicle when throttle is reduced. Larger values represent slower response.
Brake Rate *Brake_Rate_SpdMx* `0x3862 0x00`	0.1 – 30.0 s *100 – 30000*	1.0 sec	Sets the rate (in seconds/Typical Max Speed) at which the vehicle slows down when brake is applied or when throttle is applied in the opposite direction. Larger values represent slower response.
AC Pump Enable *AC_Pump_Enable_SpdM* `0x3807 0x00`	Off/On *0 – 1*	Off	This parameter should be programmed On to operate a pump motor rather than a vehicle drive motor. Speed controller responsiveness, stability, and smoothness are enhanced. This should give a more consistent feel to hydraulic functions regardless of load differences.

Speed Mode Parameter menus

Speed Mode offers the highest number of motor response parameters. Note, pertaining to the optional Speed Mode Express parameters within the Speed Mode menus as noted above, always use the VCL parameter name as shown in the selected Curtis Integrated Toolkit™ control mode selection. Within the parameter menus are read-only monitor variables (~ *italicized* below), which are helpful when setting parameters and their effects.

SPEED MODE — SPEED CONTROLLER MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Max Speed *Max_Speed_SpdM* `0x383A 0x00`	100 – 24000 rpm *100 – 24000*	4000 rpm	Defines the maximum requested motor rpm at full throttle. Partially applied throttle is scaled proportionately; e.g. 40% applied throttle corresponds to a request for 40% of the set Max Speed Value. Note: • The maximum motor rpm is subject to the lower of the three rpm constraints.
Kp *Kp_SpdM* `0x3831 0x00`	0.0 – 200.0 *0 – 16384*	30 %	The proportional term (Kp) determines how aggressively the speed controller attempts to limit the speed of the motor to Max Speed. Larger values provide tighter control. If Kp is set too high the system may experience oscillations as the controller tries to control speed. Setting Kp too low may result in an overshoot beyond Max Speed.
Ki LS *Ki_SpdM* `0x382F 0x00`	5 – 100 % *50 – 1000*	30 %	The Ki LS parameter sets the Ki for low vehicle speeds. The integral term (Ki) forces zero steady state error so the motor will run at exactly the commanded speed. Larger values provide tighter control. High gain values may cause the system to experience oscillations as the controller tries to control speed. If the gain is set too low, the motor may take a long time to approach the exact commanded speed.
Ki HS *Ki_HS_SpdM* `0x382C 0x00`	5 – 100 % *50 – 1000*	30 %	The Ki HS parameter sets the Ki for high vehicle speeds. The integral term (Ki) forces zero steady state error so the motor will run at exactly the commanded speed. Larger values provide tighter control. If the gain is set too high, oscillations may occur as the controller tries to control speed. If the gain is set too low, the motor may take a long time to approach the exact commanded speed.

SPEED MODE/SPEED CONTROLLER — VEL FEEDFORWARD MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Kvff *Kvff_SpdM* `0x3832 0x00`	0 – 500A *0 – 5000*	0 Amp	The design of the velocity feedforward term is to improve throttle responsiveness and speed controller performance, especially at low speeds. This velocity feedforward is a constant value in the direction of the velocity trajectory. For traction systems set it to 50–70% of the current needed to maintain a very low speed unloaded on flat ground. For a pump system set it to the lowest load current (i.e. the current running at the minimum load). Alternatively, the responsiveness of a pump speed-control-loop is typically enhanced by using a VCL program to continuously update this parameter to the appropriate value as each pump load is requested.
Build Rate *Vel_FF_Build_Rate_SpdM* `0x385F 0x00`	0.1 – 5.0 s *100 – 5000*	1.0 sec	Determines how fast the Kvff term builds up. For traction systems if you feel or hear the mechanical lash (backlash) pick up abruptly when you move the throttle from neutral to a very small value, slowing the build rate (i.e. setting it to a higher value) will soften the feel. For a pump system start with this parameter at the minimum setting. Slowing it down (i.e. setting it to a higher value) will reduce speed over-shoot if too much feedforward has been commanded.
Release Rate *Vel_FF_Release_Rate_SpdM* `0x3860 0x00`	0.1 – 5.0 s *100 – 5000*	0.4 sec	Determines how fast the Kvff term releases. If the release seems too abrupt, slowing the release rate (i.e. setting it to a higher value) will soften the feel. It should be set fast enough (i.e. at a low enough value) to prevent the vehicle from running on after throttle release.

SPEED MODE/SPEED CONTROLLER — ACC FEEDFORWARD MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Kaff *Kaff_SpdM* `0x3828 0x00`	0 – 500A *0 – 5000*	0 Amp	The design of the acceleration feedforward term is to improve throttle responsiveness and speed controller performance at all speeds. As a quick start function, it enhances the responsiveness at all speeds. This acceleration feedforward is a constant value applied in the direction of travel during acceleration. Set this parameter while using the application's present Accel and Decel rates by observing the average current while running full throttle at low speeds, accelerating without a load on flat ground. Set Kaff to 50–70% of that value. **Note: If any Accel rate parameters change, this parameter will also need to be changed (updated).**
Kbff *Kbff_SpdM* `0x3829 0x00`	0 – 500A *0 – 5000*	0 Amp	The design of the braking feedforward term is to improve braking responsiveness at all speeds. This braking feedforward is a constant value applied opposite the direction of travel during braking or deceleration. Using the application's present decel rates, observe the average current at full throttle braking and set Kbff to that value.
Build Rate *Acc_FF_Build_Rate_SpdM* `0x3808 0x00`	0.1 – 5.0 s *100 – 5000*	1.0 sec	Determines how fast the Kaff and Kbff terms build up. For traction systems, if the mechanical lash (slop) picks up abruptly when moving the throttle from neutral to a very small value, slowing the build rate (i.e. setting it to a higher value) will soften the feel. For a pump system, start with this parameter at the minimum setting. Slowing it down (i.e. setting it to a higher value) will reduce speed over-shoot if too much feedforward has been commanded.
Release Rate *Acc_FF_Release_Rate_SpdM* `0x3809 0x00`	0.1 – 5.0 s *100 – 5000*	0.4 sec	Determines how fast the Kaff and Kbff terms release. It should be set fast enough (i.e. at a low enough value) to prevent the vehicle from running on after throttle release.

SPEED MODE — RESPONSE MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Full Accel Rate HS *Full_Accel_Rate_HS_SpdM* `0x381D 0x00`	0.1 – 30.0 s *100 – 30000*	5.0 sec	Sets the rate (in seconds/Typical Max Speed) at which the speed command increases when applying full throttle at high vehicle speeds. Larger values represent slower response. See Figure 16 for the relationship between Full Accel Rate HS, Full Accel Rate LS, and Low Accel Rate.
Full Accel Rate LS *Full_Accel_Rate_LS_SpdM* `0x381E 0x00`	0.1 – 30.0 s *100 – 30000*	2.5 sec	Sets the rate (in seconds/Typical Max Speed) at which the speed command increases when applying full throttle at low vehicle speeds. See Figure 16 for the relationship between Full Accel Rate HS, Full Accel Rate LS, and Low Accel Rate.
Low Accel Rate *Low_Accel_Rate_SpdM* `0x3834 0x00`	0.1 – 30.0 s *100 – 30000*	10.0 sec	Sets the rate (in seconds/Typical Max Speed) at which the speed command increases when a small amount of throttle is applied. Adjusting this rate affects the low speed maneuverability.
Neutral Decel Rate HS *Neutral_Decel_Rate_HS_SpdM* `0x383C 0x00`	0.1 – 30.0 s *100 – 30000*	10.0 sec	Sets the rate (in seconds/Typical Max Speed) used to slow down the vehicle when releasing the throttle to neutral at high vehicle speeds.
Neutral Decel Rate LS *Neutral_Decel_Rate_LS_SpdM* `0x383D 0x00`	0.1 – 30.0 s *100 – 30000*	20.0 sec	Sets the rate (in seconds/Typical Max Speed) used to slow down the vehicle when releasing the throttle to neutral at slow vehicle speeds.
Partial Decel Rate *Partial_Decel_Rate_SpdM* `0x3843 0x00`	0.1 – 30.0 s	30.0 sec	Sets the rate (in seconds) to slow down the vehicle when the throttle (command) is reduced without being released to neutral. Larger values represent slower response. See Figures 16 and 17 for the relationship between Full Brake Rate HS, Full Brake Rate LS, and Low Brake Rate.
Full Brake Rate HS *Full_Brake_Rate_HS_SpdM* `0x381F 0x00`	0.1 – 30.0 s *100 – 30000*	1.0 sec	Sets the rate (in seconds/Typical Max Speed) at which the vehicle slows down from high speeds when full brake is applied or when full throttle is applied in the opposite direction. See Figures 16 and 17 for the relationship between Full Brake Rate HS, Full Brake Rate LS, and Low Brake Rate.
Full Brake Rate LS *Full_Brake_Rate_LS_SpdM* `0x3820 0x00`	0.1 – 30.0 s *100 – 30000*	2.0 sec	Sets the rate (in seconds/Typical Max Speed) at which the vehicle slows down from low speeds when full brake is applied, or when applying full throttle in the opposite direction.
Low Brake Rate *Low_Brake_Rate_SpdM* `0x3835 0x00`	0.1 – 30.0 s *100 – 30000*	4.0 sec	Sets the rate (in seconds/Typical Max Speed) at which the vehicle slows down at all speeds when a small amount of brake is applied or when applying a small amount of throttle in the opposite direction.

Quick Links:
Fig. 16 p.55
Fig. 17 p.56

SPEED MODE/RESPONSE — FINE TUNING MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
HS (High Speed) *HS* `0x3107 0x00`	0 – 100 % *0 – 32767*	70 %	Sets the percentage of the Typical Max Speed above which the HS parameters will be used. See Figures 16 and 17 (below).
LS (Low Speed) *LS* `0x3109 0x00`	0 – 100 % *0 – 32767*	30 %	Sets the percentage of the Typical Max Speed below which the LS parameters will be used. See Figures 16 and 17 (below).
Max Speed Accel *Max_Speed_Accel_SpdM* `0x3836 0x00`	0.1 – 30.0 s *100 – 30000*	1.0 sec	In some applications the Max Speed value is changed frequently through VCL or over CAN (PDOs). The Max Speed Accel parameter controls the rate at which the maximum speed setpoint is allowed to change when the value of Max Speed is raised. The rate set by this parameter is the time to ramp from 0 rpm to Typical Max Speed rpm. For example, suppose the Max Speed is raised from 1000 rpm to 4000 rpm. If the Typical Max Speed is 5000 rpm and the rate is 10.0 seconds it will take 10.0 * (4000-1000) / 5000 = 6.0 seconds to ramp from 1000 rpm to 4000 rpm.
Max Speed Decel *Max_Speed_Decel_SpdM* `0x3837 0x00`	0.1 – 30.0 s *100 – 30000*	10.0 sec	This parameter works like the Max Speed Accel parameter except that it controls the rate at which the maximum speed setpoint is allowed to change when the value of Max Speed is lowered. For example, suppose the Max Speed is decreased from 4500 rpm to 2500 rpm. If the Typical Max Speed is 5000 rpm and the rate is 5.0 seconds it will take 5.0 * (4500-2500) / 5000 = 2.0 seconds to ramp from 4500 rpm to 2500 rpm.
Reversal Soften *Reversal_Soften* `0x310B 0x00`	0 – 100 % *0 – 3000*	20 %	Larger values create a softer reversal from regenerative braking (regen) to drive when near zero speed. This helps soften the transition when the regen and drive current limits are set to different values. Note: This parameter is not mode-specific.

Figure 16

Acceleration Response Rate Diagram.

In this example,
HS = 70 %,
LS = 30 %,
Typical Max Speed = 5000 rpm

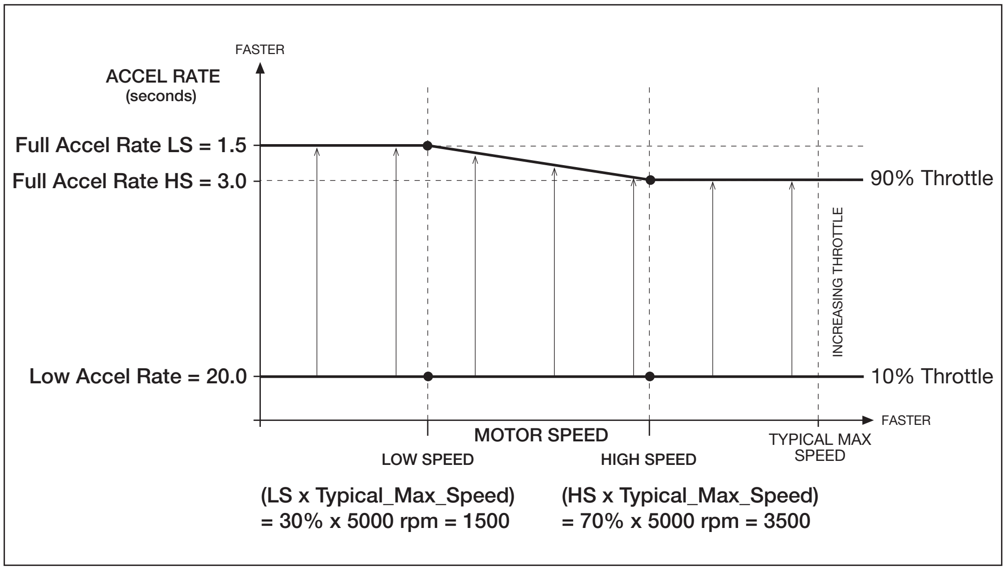
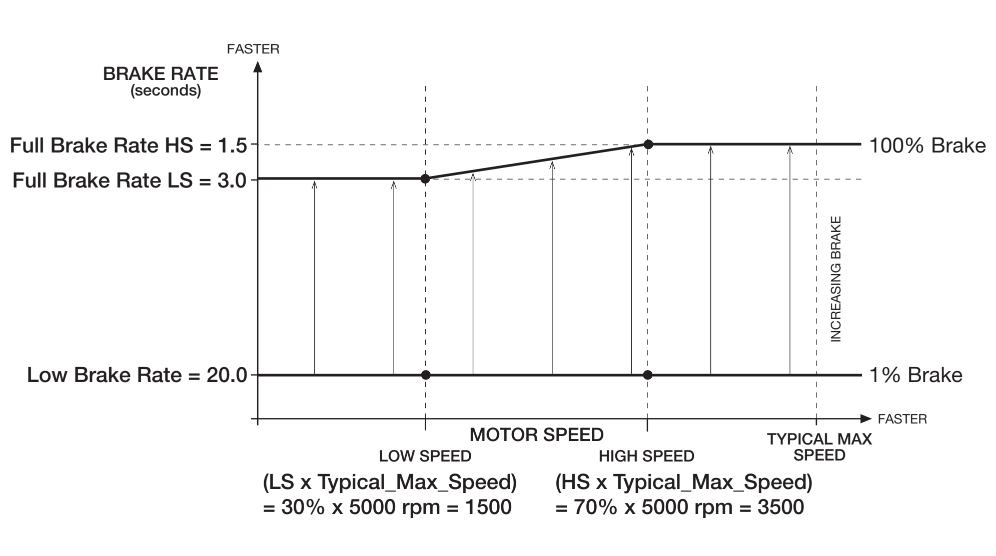


Figure 17

Braking Response Rate Diagram. In this example,
HS = 70 %,
LS = 30 %,
Typical Max Speed = 5000 rpm



SPEED MODE — RESTRAINT MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Restraint Forward *Restraint_Forward_SpdM* `0x3847 0x00`	0 – 100 % *0 – 32767*	50 %	Increases torque when on a steep hill in order to limit roll-forward speed. Setting this parameter too high may cause oscillations in the motor as it attempts to limit the roll-forward speed.
Restraint Back *Restraint_Back_SpdM* `0x3846 0x00`	0 – 100 % *0 – 32767*	50 %	Increases torque when on a steep hill in order to limit rollback speed. Setting this parameter too high may cause oscillations in the motor as it attempts to limit the rollback speed.
Soft Stop Speed *Soft_Stop_Speed* `0x384B 0x00`	0 – 500 rpm *0 – 500*	0 rpm	Defines the speed below which a much slower decel rate is used. A setting of zero disables the function. Note: This parameter works only in Speed Mode and Speed Mode Express. Soft Stop Speed is useful for vehicles that have fast deceleration and vehicles operating on ramps using the Position Hold function. With vehicles that have fast deceleration the driver may find the final speed reduction to zero rpm uncomfortable (disconcerting); the vehicle may even rock back due to tire wind-up. The Soft Stop Speed parameter allows the vehicle to slow at the same fast rate until it reaches the set threshold at which point it changes to a slower (softer) deceleration rate. However if the threshold is set too high the vehicle will feel like it is running on (also disconcerting). When releasing the throttle on a ramp the vehicle may roll back before Position Hold takes control. Use the Soft Stop Speed to reduce the amount of rollback, but do not set it so high that the vehicle drives up the ramp after the throttle is released.

SPEED MODE/RESTRAINT — POSITION HOLD MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Position Hold Enable *Position_Hold_Enable* `0x3793 0x00`	On/Off *On/Off*	Off **[PCF]**	Allows entering the Position Hold mode at zero throttle when the vehicle comes to a stop. Note: EM Brake Type = 2 also enables the Position Hold function.
Position Hold State *Position_Hold_Engaged* `0x3794 0x00`	–32768 – 32767 *–32768 – 32767*	Read Only	Position Hold State 1 = Engaged 0 = Not Engaged
Kp *Kp_Position_Hold* `0x378D 0x00`	2 – 100 % *41 – 2048*	10 %	Determines the stiffness with which position is regulated when in the Position Hold mode. High Kp values will produce less rollback on a ramp but more bouncing. Too much Kp will cause instability.
Kd *Kd_Position_Hold* `0x378C 0x00`	0 – 100 % *0 – 8192*	15 %	Determines the damping in Position Hold mode. Some damping must be present in the control system to keep the vehicle from oscillating slowly (bouncing). High Kd values will improve the dynamic response of the Position Hold controller, but too much Kd will cause instability.
Zero Speed Threshold *Zero_Speed_Threshold* `0x30F2 0x00`	5 – 300 rpm *5 – 300*	30 rpm	Determines the speed below which the EM brake is (commanded) set. Setting this speed too high may cause a jerky stop when the EM brake sets and stops the motor. This parameter appears twice in the menu structure. Changing the value of this parameter affects this (same) parameter in the EM Brake Control menu.
Zero Speed Threshold Time *Zero_Speed_Threshold_Time* `0x30F3 0x00`	0 – 480 ms *0 – 60*	32 ms	Determines how long motor speed must be below *Zero_Speed_Threshold* to declare zero speed. This parameter appears twice in the menu structure. Changing the value of this parameter affects this (same) parameter in the EM Brake Control menu.
Entry Rate *Entry_Rate_Position_Hold* `0x3786 0x00`	5 – 100 % *50 – 1000*	50 %	When the vehicle transitions from forward speed to reverse speed or from reverse speed to forward speed (for example, when coming to a stop while traveling up a steep ramp), Position Hold is automatically entered immediately at zero speed regardless of this parameter. This parameter applies when the vehicle needs to stop without the assistance of gravity (for example, when moving forward down a ramp). This rate determines how quickly zero speed is attained after the ramped speed request reaches zero. Setting this parameter too high will make the stop seem very abrupt and may even cause the vehicle to roll back slightly. When the parameter is set lower the vehicle takes longer to come to a stop and enter Position Hold mode.
Position Hold Settling Time *Position_Hold_Settling_Time* `0x3783 0x00`	0.0 – 5.0 s *0 – 156*	3.0 sec	Determines how long the position hold function is to operate before the EM brake is set. This time should be long enough for the hold to settle. This parameter appears twice in the menu structure. Changing the value of this parameter affects this (same) parameter in the EM Brake Control menu.
Position Hold Timeout Time *Position_Hold_Timeout_Time* `0x3795 0x00`	0.0 – 20.0 s *0 – 625*	0.0 sec	This parameter, plus the Set Speed Settling Time parameter, together set the maximum time the vehicle will stay in Position Hold before releasing the hold and going into Restraint mode. Setting this parameter to zero disables this timeout function, which means the Position Hold will be held. Activating the interlock resets the timer.

SPEED MODE/RESTRAINT — POSITION HOLD MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Exit Rollback Reduction *Exit_Rollback_Reduction* `0x3787 0x00`	0 – 100 % *0 – 2048*	50 %	This function is applicable when the Torque Preload Enable function has been Disabled (Off) or the Torque Preload Cancel Delay timer has expired (see EM Brake menu). Exit Rollback Reduction is only intended for use with EM Brake Type = 2; for EM Brake Type = 1 (or 0) set Exit Rollback Reduction = 0. Exit Rollback Reduction introduces an additional control function for the speed controller to reduce rollback on a ramp after applying a throttle command from a stop. For example, suppose the vehicle is on a ramp facing upwards and after a forward throttle request, the vehicle rolls back slightly before climbing the ramp (again, the assumption is the torque preload function is inactive). As the vehicle rolls back, this additional term is added to the torque request until forward speed is sensed, attempting to minimize rollback. The effect increases with percentage, and is disabled when = 0%. Setting the value too high may introduce temporary oscillations. If the Torque Preload Cancel Delay expires before the throttle is re-engaged, the torque preload [Position Hold] memory will be cleared. In such a case, setting Exit Rollback Reduction to > 0% will reduce unintended roll.

SPEED MODE — HYDRAULIC FEATURES MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Pump Enable *AC_Pump_Enable_SpdM* `0x3807 0x00`	On/Off	Off	Set this parameter to On when operating the controller as a hydraulic pump-motor controller. This enhances the speed controller responsiveness, stability and smoothness. It gives a more consistent feel to hydraulic functions regardless of load differences.
Regen Lower Enable *Regen_Lower_Enable_SpdM* `0x3845 0x00`	On/Off	Off	This parameter works together with Pump Enable as follows: When Pump Enable = On and Regen Lower Enable = On the pump motor can turn in both the forward and reverse directions. In this case, the pump is usable as the Lower function (reverse). This is similar to regenerative braking, yet uses the hydraulic pressure. When Pump Enable = On and Regen Lower Enable = Off the pump motor can turn only in the forward direction. In this case, a hydraulic valve controls the Lower function. When Pump Enable = Off the Regen Lower Enable parameter has no effect on the control system.

TORQUE MODE

TORQUE MODE MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Max Speed *Max_Speed_TrqM* `0x391D 0X00`	500 – 24000 RPM *500 – 24000*	7000 rpm	Defines the maximum allowed motor rpm for torque control mode (independent of throttle position). In torque control mode full throttle requests 100% of the available torque. Partially-applied throttle is scaled proportionately.
Kp *Kp_TrqM* `0x391A 0x00`	0 – 100 % *0 – 8192*	30 %	The proportional term (Kp) determines how aggressively the speed controller attempts to limit the speed of the motor to Max Speed. Larger values provide tighter control. If Kp is set too high the system may experience oscillations as the controller tries to control speed. Setting Kp too low may result in an overshoot beyond Max Speed.
Ki *Ki_TrqM* `0x3919 0x00`	0 – 100 % *0 – 1000*	30 %	The integral gain (Ki) forces zero steady state error so the motor speed will be limited to Max Speed. Larger values provide faster control. If the gain is set too high you may experience oscillations as the controller tries to limit speed. If it is set too low it may take a long time for the motor to approach Max Speed from over speed.
Kd *Kd_TrqM* `0x3918 0x00`	0 – 100 % *0 – 8192*	10 %	Provides damping as the vehicle approaches top speed thereby reducing overshoot. If Kd is set too high the vehicle may take too long to reach top speed. If Kd is set too low the vehicle may overshoot top speed, especially when traveling downhill.
Accel Rate *Accel_Rate_TrqM* `0x3902 0x00`	0.1 – 30.0 sec *100 – 30000*	1.0 sec	Sets the rate (in seconds) at which the motor torque increases to full when full throttle is applied. Larger values represent slower response.
Accel Release Rate *Accel_Release_Rate_TrqM* `0x3904 0x00`	0.1 – 2.0 sec *100 – 2000*	0.4 sec	Determines how quickly deceleration will be initiated when the throttle is released while the vehicle is still accelerating. If the release rate is fast (i.e. set to a low value) the transition is initiated abruptly. The transition is smoother if the release rate is set to a higher value (slower transition); however setting the rate too high can cause the vehicle to feel uncontrollable when the throttle is released as it will continue to drive for a short time.
Brake Rate *Brake_Rate_TrqM* `0x3907 0x00`	0.1 – 5.0 sec *100 – 5000*	1.0 sec	Adjusts the rate (in seconds) at which braking torque builds. Lower values represent faster times and therefore faster braking; gentler braking is achieved by setting the braking rate to a higher value.
Brake Release Rate *Brake_Release_Rate_TrqM* `0x3908 0x00`	0.1 – 2.0 sec *100 – 2000*	0.4 sec	Adjusts the rate (in seconds) at which braking torque releases as the vehicle transitions from braking to drive.
Neutral Braking *Neutral_Braking_TrqM* `0x391F 0x00`	0 – 100 % *0 – 32767*	10 %	Neutral braking occurs progressively when the throttle is reduced toward the neutral position or when no direction is selected. The neutral braking parameter is adjustable from 0 to 100% of the regen current limit.
Neutral Taper Speed *Neutral_Taper_Speed_TrqM* `0x3922 0x00`	20 – 24000 RPM *20 – 24000*	500 rpm	Determines the motor speed below which neutral braking current is adjusted when throttle is reduced. The neutral braking current is linearly reduced from *Neutral Braking* x *Regen Current Limit* at the Neutral Taper Speed to the Creep Torque current at zero rpm motor speed. Note: Setting the taper speed too low may cause oscillations in the motor.

TORQUE MODE MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Forward Full Restraint Speed *Forward_Full_Restraint_Speed_TrqM* `0x3912 0x00`	100 – 32000 RPM *100 – 32000*	800 rpm	Sets the speed point at which the full regen current will be applied to restrain the vehicle from rolling forward. Although this speed is never actually reached it does set the slope of the restraint strength and can be thought of as a gain. Setting this parameter too low can cause oscillations.
Back Full Restraint Speed *Back_Full_Restraint_Speed_TrqM* `0x3905 0x00`	100 – 32000 RPM *100 – 32000*	800 rpm	Sets the speed point at which the full regen current will be applied to restrain the vehicle from rolling in reverse (backward). Although this speed is never actually reached it does set the slope of the restraint strength and can be thought of as a gain. Setting this parameter too low can cause oscillations.

TORQUE MODE MENU — FINE TUNING MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Creep Torque *Creep_Torque_TrqM* `0x3910 0X00`	0 – 100 % *0 – 32767*	0 % ⚠ **WARNING**	Determines the amount of torque applied to the vehicle at a stop with no throttle input to emulate the feel of an automatic transmission automobile. See Figure 18. **When interlock is engaged creep torque allows vehicle propulsion if a direction is selected even though no throttle is applied. Care should be taken when setting up this parameter.** If pedal braking is enabled creep torque is progressively disabled as brake is applied so as to prevent the motor from driving into the brakes and thus wasting energy. Creep Torque and Neutral Taper Speed interact to create the slope of the torque response as the vehicle approaches zero speed. If the vehicle oscillates as it coasts down toward zero speed try lowering Creep Torque or increasing Neutral Taper Speed.
Brake Full Creep Cancel *Brake_Full_Creep_Cancel_TrqM* `0x3906 0X00`	25 – 100 % *8192 – 32767*	50 %	Determines the amount of brake pedal input that will fully cancel the creep torque. The amount of cancellation is proportional to the brake input.
Creep Build Rate *Creep_Build_Rate_TrqM* `0x390B 0X00`	0.1 – 5.0 sec *100 – 5000*	0.1 sec	Determines how fast the programmed creep torque builds when a direction is selected.
Creep Release Rate *Creep_Release_Rate_TrqM* `0x390E 0x00`	0.1 – 5.0 sec *100 – 5000*	3 sec	Determines how fast the programmed creep torque releases when the brake is cancelling the creep torque or when the direction switches are cleared (neutral).
Gear Soften *Gear_Soften_TrqM* `0x3914 0x00`	0 – 100 % *0 – 5000*	20 %	Adjusts the throttle take-up from linear (0% setting) to an S curve. See Figure 19. Larger values create softer throttle take-up in forward and reverse. Softening is progressively reduced at higher speeds.
Brake Taper Speed *Brake_Taper_Speed_TrqM* `0x3909 0x00`	20 – 24000 RPM *20 – 24000*	1000 rpm	Determines the motor speed below which the maximum braking current is linearly reduced from 100% to 0% at zero speed. Setting the taper speed too low for the braking current will cause oscillations in the motor as it attempts to brake the vehicle to a stop on very steep slopes. Taper speed is applicable only in response to brake pedal input; it does not affect direction reversal braking or neutral braking. If the vehicle is in restraint when the brake is pressed the applied braking torque is affected by both Brake Taper Speed and Forward (or Back) Full Restraint Speed. If the vehicle oscillates in this mode it may be necessary to increase one or more of these parameters. See Figure 20.
Reversal Soften *Reversal_Soften* `0x310B 0x00`	0 – 100 % *0 – 3000*	20 %	Larger values create a softer reversal from regen braking to drive when near zero speed. This helps soften the transition when the regen and drive current limits are set to different values. Note: This parameter is not mode-specific.

TORQUE MODE MENU — FINE TUNING MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Max Speed Decel *Max_Speed_Decel_TrqM* `0x391B 0x00`	0.1 – 30.0 sec *100 – 30000*	10.0 sec	In some applications the Max Speed value is changed frequently through VCL or over the CAN bus. The Max Speed Accel parameter controls the rate at which the maximum speed setpoint is allowed to change when the value of Max Speed is lowered. The rate set by this parameter is the time to ramp from Typical Max Speed rpm to 0 rpm. For example suppose you change Max Speed from 3000 rpm to 1000 rpm. If Typical Max Speed is 5000 rpm and the rate is 5.0 seconds it will take 5.0 x (3000 – 1000) / 5000 – 2.0 seconds to ramp from 3000 rpm to 1000 rpm.

Figure 18
Creep Mode Throttle — Torque Mode

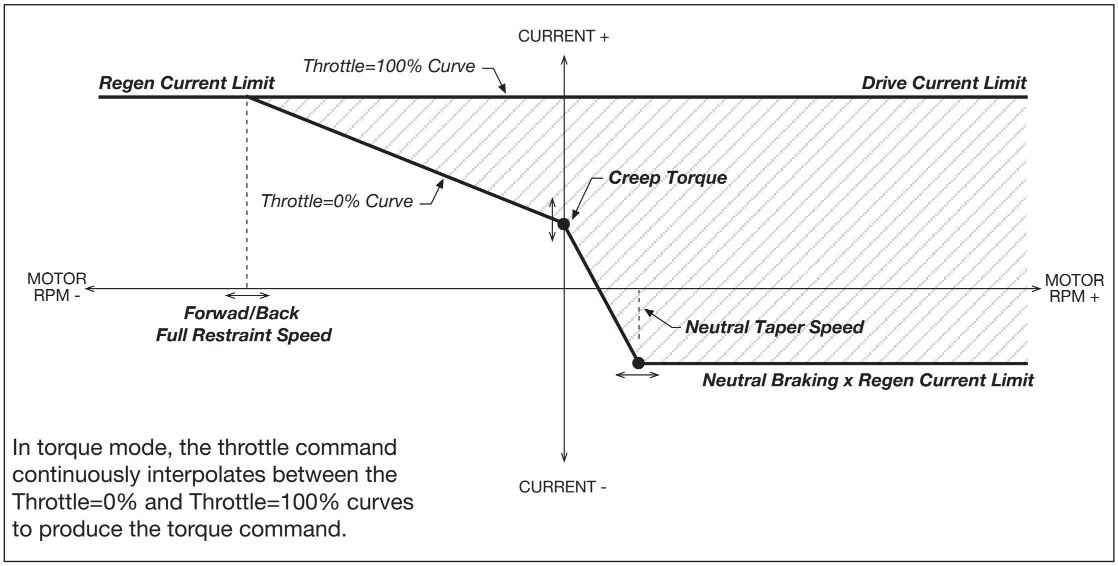


Figure 19
Gear Soften — Torque Mode

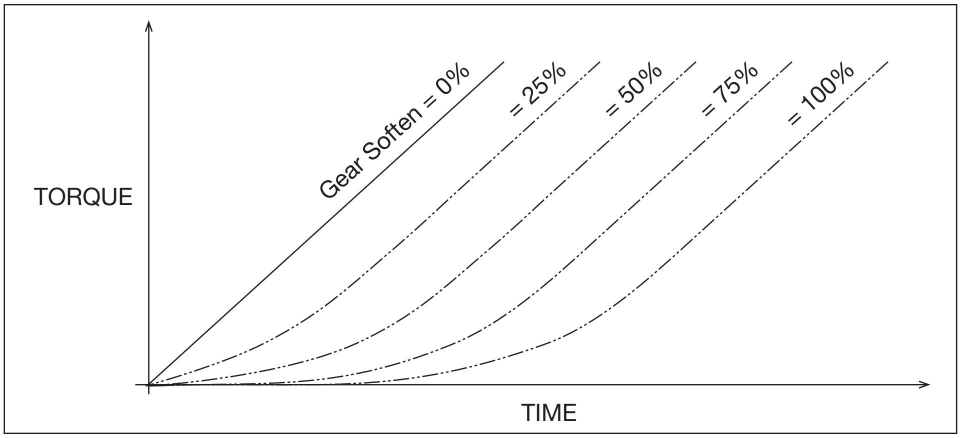
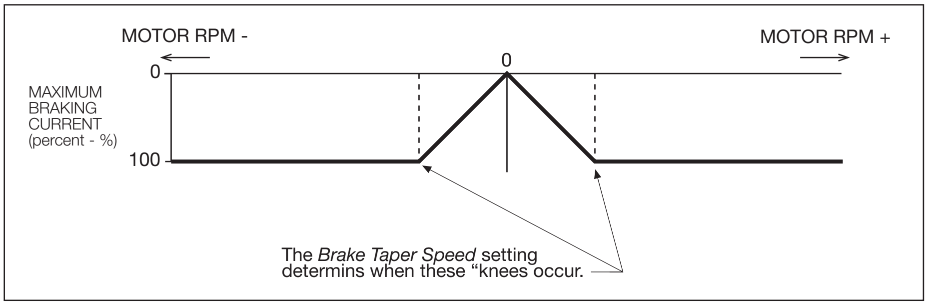


Figure 20
Brake Taper Speed — Torque Mode



APPLICATION SETUP

APPLICATION SETUP — THROTTLE MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Throttle Input *Throttle_Pot_Percent* `0x3360 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	Normalized percentage of the throttle input (pin 16 basis). Note, this monitor variable appears in the *System Monitor » Command Inputs* » menu.
Throttle Command *Throttle_Command* `0x335D 0x00`	–100.0 – 100.0 % *–32767 – 32767*	Read Only	Throttle request to the slew rate block of the motor control. See Figure 22, Throttle Signal Processing. **Note:** *Throttle_Command* = 0 is Neutral, *Throttle_Command* = +1 is zero throttle with fwd direction selected, *Throttle_Command* = –1 is zero throttle with rev direction selected.
Throttle Multiplier *Throttle_Multiplier* `0x335F 0x00`	–200 – 200 % *–256 – 256*	+100 %	Multiplies the throttle signal, which is useful in VCL throttle processing. See Figure 22, Throttle Signal Processing.
Mapped Throttle *Mapped_Throttle* `0x3352 0x00`	–100.0 – 100.0 % *–32767 – 32767*	Read Only	Mapped throttle request. See Figure 22, Throttle Signal Processing.
Direction Source *Direction_Source* `0x3345 0x00`	Fwd/Rev Wig/Wag ——— *0 – Fwd/Rev* *1 – Wig/Wag*	Fwd/Rev **[PCF]**	Configures how throttle direction is determined. 0. Fwd/Rev Switch. Requires a dedicated forward or reverse input to turn the motor in the assigned direction. Asserting both inputs at the same time results in a HPD Sequencing fault. 1. Based on the throttle mapping (Wig Wag). Here, the neutral point must be set up somewhere in the center of the throttle throw, with increasing voltage beyond this point providing increasing forward command and voltages below this point providing increasing reverse command. Wigwag throttles are only applicable to 0–5V voltage and 3-wire potentiometer throttles.
Forward Min Input *Forward_Deadband_Percent* `0x3349 0x00`	0 – 100 % *0 – 1000*	15 %	Defines the 2- or 3-wire wiper or the voltage-throttle percentage at the minimum-throttle threshold for the forward direction (positive motor rpm). Increasing this parameter will increase the neutral range (deadband). This parameter is especially useful with throttle assemblies that do not reliably return to a well-defined neutral point because it allows the “deadband” to be defined wide enough to ensure that the controller goes into neutral when the throttle mechanism is released. See Figure 21, *Throttle Adjustments Diagram*, below.
Forward Max Input *Forward_Max_Percent* `0x334B 0x00`	0 – 100 % *0 – 1000*	85 %	Defines the 2- or 3-wire wiper or voltage-throttle percentage required to produce 100% controller output. Decreasing this parameter value reduces the voltage necessary to produce full-throttle controller output. This parameter accommodates reduced-range throttle assemblies. See Figure 21, *Throttle Adjustments Diagram*, below.
Forward Map Shape *Forward_Map* `0x334A 0x00`	0 – 100 % *0 – 32767*	35 %	Modifies the vehicle’s response to the throttle input. Setting this parameter to 50% provides a linear output response to throttle position. Values below 50% reduce the controller output at low throttle settings, providing enhanced slow speed maneuverability. Values above 50% give the vehicle a faster, more responsive feel at low throttle settings. See Figure 21, *Throttle Adjustments Diagram*, below. The map value is the percentage of controller output at half throttle [(Min Input + Max Input)/2].
Reverse Min Input *Reverse_Deadband_Percent* `0x3358 0x00`	0 – 100 % *0 – 1000*	15 %	The same as the Forward Min Input parameter counterpart and applies when throttle direction is reversed (negative motor rpm). See Figure 21, *Throttle Adjustments Diagram*, below.
Reverse Max Input *Reverse_Max_Percent* `0x335A 0x00`	0 – 100 % *0 – 1000*	85 %	The same as the Forward Max Input parameter counterpart and applies when throttle direction is reversed (negative motor rpm). See Figure 21, *Throttle Adjustments Diagram*, below.

Quick Links:
Fig. 21 p.64
Fig. 22 p.66

APPLICATION SETUP — THROTTLE MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Reverse Map Shape *Reverse_Map* `0x3359 0x00`	0 – 100 % *0 – 32767*	35 %	The same as the Forward Map Shape parameter counterpart and applies when throttle direction is reversed (negative motor rpm). See Figure 21, *Throttle Adjustments Diagram*, below.
Throttle Filter *Throttle_Filter_Frequency* `0x33D9 0x00`	0.1 – 10.0 Hz *10 – 1000*	0.6 Hz	Sets the low-pass filter cutoff frequency for the 2- or 3-wire pot-wiper or voltage-throttle input. Higher values will make the throttle more responsive to quick changes. Lower values will make the throttle less responsive and subject to electrical noise.
HPD SRO Type *HPD_SRO_Type* `0x334D 0x00`	0 – 3 *0 – 3*	1 **[PCF]**	Determines the type of HPD (High Pedal Disable) and SRO (Static Return to Off) protection. One type of check is available for material-handling vehicles and two types for golf-style vehicles. Note: If any of the HPD/SRO checks finds an input-sequencing problem, an HPD/Sequencing Fault (flash code 4-7) is set. 0. HPD/SRO feature is disabled. 1. HPD SRO enabled for material-handling vehicles: ***HPD:*** If throttle input is received before interlock input. ***SRO:*** If direction input is received before interlock input. The HPD SRO check is made when the interlock input changes from Off to On. If the throttle input >25% or a direction input is On, an HPD/Sequencing Fault is set. The HPD/Sequencing Fault is cleared by returning the throttle input to <25% and the direction inputs to Off. 2. Golf-style HPD that allows direction reversal while driving: ***HPD:*** Throttle input received before interlock, or throttle input received before direction input while vehicle is stationary. ***SRO:*** None. The HPD check is made when the interlock input or direction inputs are Off and the vehicle is stationary. If the throttle input >25%, an HPD/Sequencing Fault is set. No SRO check is made with this type so the order of the interlock and direction inputs does not matter. The HPD/Sequencing Fault is cleared by returning the throttle input to <25% and the direction inputs to Off. 3. Golf-style HPD that prevents direction reversal while driving: ***HPD:*** If throttle input is received before interlock or direction input. ***SRO:*** None. HPD check is made when the interlock input or direction inputs are Off. If the throttle input >25%, an HPD/Sequencing Fault is set. The check is done regardless of vehicle speed so reversing direction with throttle input >25% will result in a fault. No SRO check is made with this type so the order of the interlock and direction inputs does not matter. The HPD/Sequencing Fault is cleared by returning the throttle input to <25 % and the direction inputs to Off.
Sequencing Delay *Sequencing_Delay* `0x335B 0x00`	0.0 – 5.0 s *0 – 1250*	0.1 sec	Typically, the sequencing delay feature allows the cycling of the interlock switch within a set time (the defined sequencing delay) thus preventing inadvertent activation of HPD/SRO. This feature is useful in applications where the interlock switch may bounce or be momentarily cycled during operation. This parameter also works to delay the Hydraulics_Inhibit_Type at KSI = On (by this same Sequencing Delay value). See Chapter 6, Commissioning.
VCL Throttle Enable *VCL_Throttle_Enable* `0x3367 0x00`	On/Off *On/Off*	Off	When programmed On, the throttle processing with fault detection will operate normally; however, the throttle command will require VCL to define the connection between the OS_Throttle and VCL_Throttle variables. This allows VCL flexibility and customization of throttle processing while still allowing throttle fault detection.

Figure 21

Throttle Adjustments Diagram

The effect of throttle adjustment parameters. Together these parameters determine the controller's response to throttle demand (in forward or reverse) and to brake demand, if applicable.

In the examples shown in this figure,

Min Input = 10 %

Max Input = 95 %

Map Shape: six examples.

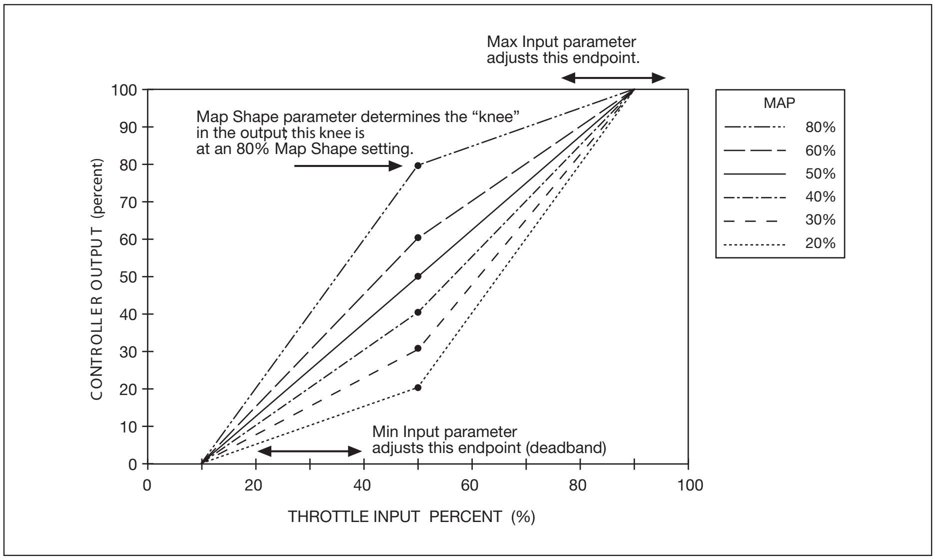


Figure 22 illustrates the throttle's signal chain where the throttle parameters are applied. In addition to these throttle-specific parameters, refer to the Controller Setup/Inputs menu (below) for the selection of the throttle configurations. The default physical-throttle input is via Analog 1 Type (Pot 1 Wiper, pin 16), which is an entry point for the throttle signal in Figure 22 (signal flow is left to right). See Figure 13, the default wiring diagram, and Chapter 6, Commissioning, for systematic throttle-option setup instructions.

Quick Links:
Fig. 13 p.17

Based upon the selection in the Inputs menu (i.e., voltage or potentiometer throttle) the throttle will use two to three of the controller's I/Os. For 5-volt voltage throttles, the +5V supply (pin 26) and I/O Ground (Pin 18 or 7) are used to power the throttle, with the signal connected to Analog 1 (pin 16). Resistive 3-wire potentiometer throttles use the Pot 6 Supply (pin 15) and the I/O Ground (pin 18 or 7) to set up the voltage-supply circuit with the potentiometer's wiper (working as a voltage divider) connected to the Pot 1 Wiper (pin 16). When using a 2-wire potentiometer throttle, connect the potentiometer's wiper to Pot 1 Wiper (pin 16) and one end to I/O Ground (pin 18 or 7). Leave the other end/lead open, as the brake throttle in Figure 13 is a 2-wire input. When selecting these resistive throttles, the monitor item Analog 1 (*analog_input_volts_1*) voltage reading at pin 16 undergoes dynamic cycles verifying the external connections. This coupled to the assigned nominal resistance value, results in the voltage reading having no relevance to the throttle signal. The motor direction (*Direction_Source*) options are via the Forward/Reverse switch inputs or by Wigwag throttle. The Single Switch option assigns forward as the default direction, with reverse its only input (hence, single switch). Wigwag throttles are similar to voltage throttles, except their input voltage also determines the motor direction. In all cases, the first feedback in the throttle's signal chain is the *Throttle_Pot_Percent* variable. Notice that the controller processes these inputs as a percentage, not as a voltage (as noted above).

VCL can interface and modify the throttle signals at several points, from the voltage or potentiometer input at pin 16 to the final motor controller command. Use VCL to create unique throttle commands, adjust parameters to provide Multimode operation, or modify the throttle command based on steering angle, mast height, load, etc.

The throttle signal chains within the controller are sophisticated and flexible. Before applying VCL to modify these chains, it is important to understand the ramifications of implementing changes. With the physical throttle's parameters set, the *Throttle_Pot_Percent* variable passes to the Throttle Mapping block, which re-shapes the throttle signal magnitude and direction based on the various Throttle menu parameters and the directional inputs. When other Analog inputs are used for the throttle, the signal-voltage is also converted to a percentage (Percent, see the Inputs menu) and then processed in VCL. In these cases, the variable *VCL_Throttle_Pot* is used to process throttle signals through the map function (where 0 – 1000 ≡ 0 – 100%), versus using *VCL_Throttle*, which does not pass through the mapping blocks in Figure 22.

Following the throttle mapping is the switch for selecting *VCL_Throttle* as the input, when the parameter *VCL_Throttle_Enable* is set (On). Continuing, the next modifier is the HPD State machine that uses the settings based upon the HPD SRO Type and the *Sequencing_Delay* parameters. At this point in the signal chain, to modify the throttle signal further, use the *Throttle_Multiplier* parameter, which is useful in VCL throttle processing. The throttle signal will then be between ± 100% and accessible using ***Mapped_Throttle*** monitor variable. Final modification is via the *Throttle_Filter_Frequency (0x33D9)*, where a low-pass filter is applied. Higher values will make the throttle more responsive to quick changes. Lower values will make the throttle less responsive and subject to electrical noise. Barring events that will "zero" the throttle (see the list in Figure 22), the throttle signal will pass to the control mode block and onto the motor control algorithm code as the ***Throttle_Command*** monitor variable. Figure 22 shows the branch to dual drive. For dual drive applications, refer to the F-Series Dual Drive manual supplement.

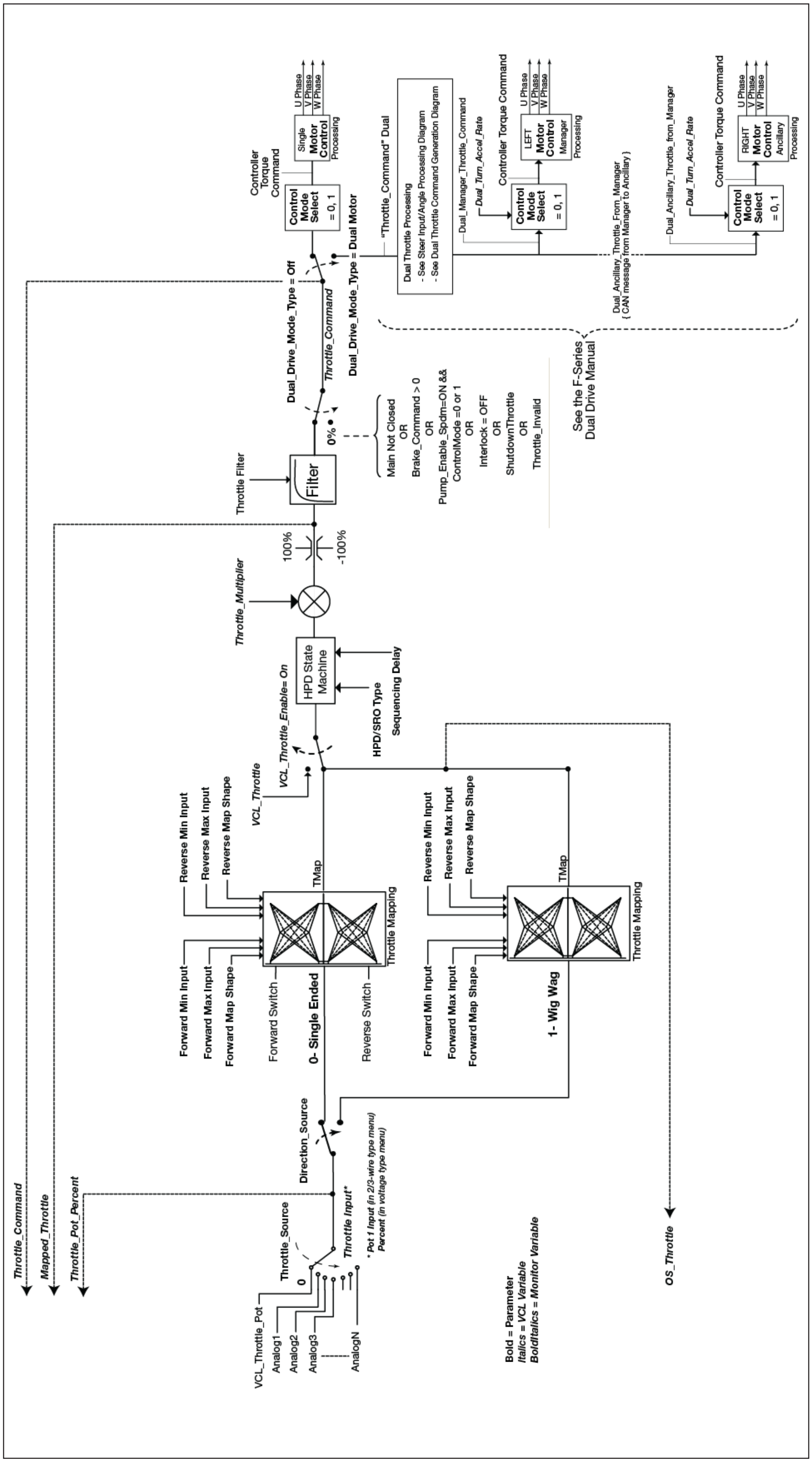


Figure 22
Throttle Signal Processing

APPLICATION SETUP — BRAKE MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Brake Input *Brake_Pot_Percent* `0x33D3 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	Normalized percentage of the brake input. Similarly to the Throttle Input variable, the controller processes the voltage at the assigned Analog Input as a percentage, and not as a voltage (due to dynamic testing), to determine the amount of motor braking (regen). Note, the controller does not offer a specific brake input pin(s) or circuit(s), as a default (*Brake_Source* = 0). Without a physical throttle, the analog input at pin 16 can be a physical brake input. Figure 13 illustrates using the 2-wire option as a brake signal input on the 35-pin controllers.
Mapped Brake *Mapped_Brake* `0x3350 0x00`	0.0 – 100.0 % *0 – 32767*	Read Only	Mapped brake request. See Figure 23.
Brake Command *Brake_Command* `0x33D2 0x00`	0.0 – 100.0 % *0 – 32767*	Read Only	Brake request to slew rate block. Note that the percent value is only positive for the Brake Command. See Figure 23.
Brake Pedal Enable *Brake_Pedal_Enable* `0x33C9 0x00`	On/Off *On/Off*	Off	When this parameter is set to On, the Brake Input (from a brake pedal input) forms the basis of the Brake Command. If using a VCL Brake, set this parameter to Off.
Brake Min Input *Brake_Deadband_Percent* `0x33C2 0x00`	0 – 100 % *0 – 1000*	15 %	Defines the minimum brake input's threshold. Increasing this parameter will increase the neutral range (deadband). This parameter is especially useful with the pedal assemblies that do not reliably return to a well-defined neutral point because it allows the "deadband" to be defined wide enough to ensure that the controller cancels regen (motor braking) when the pedal mechanism is released.
Brake Max Input *Brake_Max_Percent* `0x33C6 0x00`	0 – 100 % *0 – 1000*	85 %	Defines the brake input percentage required to produce 100% controller regen. Decreasing this parameter setting reduces the analog input percentage (voltage at the input) and therefore the full stroke necessary to produce full controller regen. This parameter accommodates reduced-range throttle assemblies.
Brake Map Shape *Brake_Map* `0x33C5 0x00`	0 – 100 % *0 – 32767*	50 %	Modifies the vehicle's response to a brake input. Setting the brake map at 50% provides a linear output response to pedal position (brake input). Values below 50% increase the controller regen at low brake settings, providing enhanced slow speed braking. Values above 50% give the vehicle a slower, less responsive feel at low brake inputs. The map value is the percentage of controller regen at half throttle [(Min Input + Max Input)/2].
Brake Offset *Brake_Offset* `0x33C8 0x00`	0 – 100 % *0 – 32767*	0 %	Defines the initial controller response generated when first moving the brake out of the neutral deadband. For most vehicles, a setting of 0 is appropriate. For heavy vehicles, however, decreasing the offset may improve controllability by reducing the amount of brake-input required to start the vehicle regen braking.
Brake Filter *Brake_Filter_Frequency* `0x33DA 0x00`	0.1 – 10.0 Hz *10 – 1000*	0.6 Hz	Sets the low pass filter's cutoff frequency for the assigned brake (analog) input. Higher values will make the brake more responsive to quick changes. Lower values will make the brake less responsive and subject to electrical noise.
VCL Brake Enable *VCL_Brake_Enable* `0x33D1 0x00`	On/Off *On/Off*	Off	When programmed On, the brake processing with fault detection at the assigned analog input will operate normally; however, the brake command will require VCL to define the connection between the *Brake_Pot_Percent* and *VCL_Brake* variables. This allows VCL flexibility and customization of throttle processing while still allowing brake fault detection.

Quick Link:
Fig. 23 p.69

Figure 23 illustrates the Brake's signal chain where the brake parameters are applied. Brake processing is optional as it can be turned off by setting *Brake_Pedal_Enable = Off*. When turned on, note that the brake processing can be with or without VCL. When the controller is in Speed Mode, any non-zero brake command will then override the throttle signal and the motor controller will brake to a stop as determined by the Brake Current Limit parameter. When using a brake pedal, its input follows a similar design to the throttle input with the major exception that it is always a unidirectional deceleration command (provide regenerative commands only). There is no brake multiplier parameter, although VCL may still inject its own command, replacing the normal connection from the brake pedal inputs.

The brake signal chain illustrates the use of the controller's analog potentiometer inputs, shown in Figure 13 as a 2-wire brake pot. The input can also be a voltage input, using the common F-Series analog inputs configurations and the "source" in the IO Assignments Controls menu. In the diagram, the brake signal flows left-to-right, passing through various gates to the *Brake Command* fed into the motor control-processing block. Included in the diagram is the separation of the dual drive manager and ancillary controller. Point Ⓑ is where the dual drive manager controller generates the brake command for the ancillary controller, for the ancillary's points Ⓒ and Ⓐ. (Reference the F-Series Dual Drive manual, supplement).

Like the throttle, the output of the analog brake input is a percentage, not a voltage (*Brake_Pot_Percent*). The mapping block uses the brake menu parameters (above), eventually feeding into the brake filter, 0x33DA, (typically) and onto the motor control block. Notice that a VCL brake input eliminates the analog inputs and mapping block (i.e., a VCL Brake will free up analog inputs for other purposes). The brake signal passes through a limiter, which limits the brake signal to a range of 0–100% (0–32767). After the limiter, the brake signal is a VCL variable called *Mapped_Brake*, which displays as Mapped Brake in the System Monitor menu. Checking the value of *Mapped_Brake* is a good way to see if the brake menu parameters are set correctly. A VCL program can control the brake by changing the variable *VCL_Brake* (0x33D0 0x00). In Figure 23, notice that the decision gates are in the analog-input to brake-command flow state. If the Brake Pedal Enable parameter is off, the brake command is zero. If a fault action results in "fullbrake" then the brake command becomes 100%. The *Brake_Command* (0x33D2 0x00) is a value of 0-100% (0 – 32767).

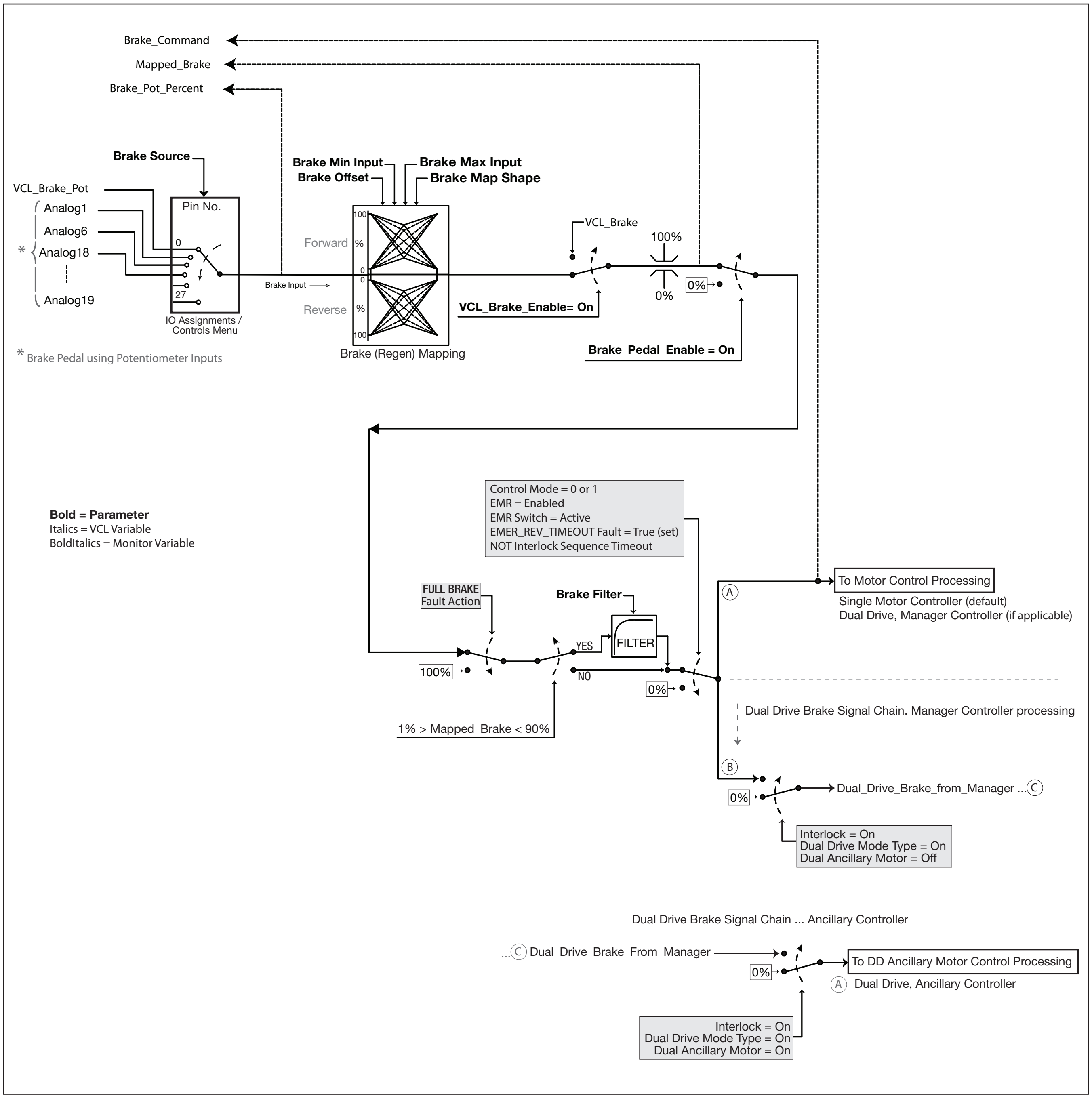


Figure 23

Brake Signal Processing

APPLICATION SETUP — CAN INTERFACE MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
CANopen Interlock *CANopen_Interlock_Enable* `0x34B4 0x00`	On/Off *On/Off*	Off	When programmed On, CAN NMT State must = 5 (operational state) in order for the interlock to be set. This parameter also occurs in the CAN Interface 2 menu. Changing it in one menu also changes it in the other location.
Baud Rate *CAN_Baud_Rate* `0x2001 0x01`	–1 – 4 *–1 – 4*	0	Sets the CAN baud rate for the CANopen Ancillary system: –1 = 100Kbps 0 = 125Kbps 1 = 250Kbps 2 = 500Kbps 3 = 800Kbps 4 = 1000Kbps.
Heartbeat Rate *CANopen_Heart_Beat_Rate* `0x1017 0x00`	16 – 2000 ms *0 – 2000*	100 ms	Sets the transmission rate of the CAN heartbeat messages from the CANopen Ancillary system.
Emergency Message Rate *CAN_Open_Emergency_Inhibit_Time* `0x1015 0x00`	0 – 6554 ms *0 – 65535*	16 ms	Sets the minimum transmission rate of the CAN Emergency Messages from the CANopen Ancillary system. This prevents quickly changing fault states from generating so many emergency messages that they flood the CANbus. The step size is 10 ms.
CAN NMT State *CAN_NMT_State* `0x32A4 0x00` This Read Only variable also occurs in the CAN Interface 2 menu.	0 – 127 *0 – 127*	Read Only	Controller CAN NMT state: 0 = initialization 4 = stopped 5 = operational 127= pre-operational
CAN Node ID *Can_Node_Id* `0x2000 0x01`	0001 – FFFFh *1 – 65535*	0026h (0x26) (38d)	Displays the controller's Node ID, in hexadecimal. For example: 0x26 = 38d Do not assign 0x00 as a device's CAN Node ID. Such an ID will never be detected by CIT or the 1313 HHP, and therefore be inoperable.

CAN ports 1 and 2

If the controller is equipped with two external CAN ports (Ports 1 and 2), each may have a unique baud rate, and CANopen node ID. Either may be used for the secure node connection to the Curtis programming tools (see Appendix D). Note that only one port at a time may be used for Curtis tools connection. Note also that there is a single CANopen NMT state for the controller, which will be indicated via heartbeats unique to each port's node ID, but which can be changed by sending NMT commands to either port. In addition, CANopen Emergency messages will be transmitted to all CAN ports. Each port has a unique CANopen PDO configuration, and may receive and transmit unique data for each port.

Non-isolated CAN models

CAN1 has the internal 120Ω termination when pins 21 and 34 are connected. CAN2 does not have the internal 120Ω termination. For applications requiring CAN2 termination, add it externally from the controller. On these models, the CAN-circuit reference is from the controller's I/O ground (pins 7 and 18).

CAN1

CAN1 L = pin 35
CAN1 H = pin 23
CAN1 120-Ohm termination: externally connect (short) pins 21 and 34.

CAN2

CAN2 L = pin 29
CAN2 H = pin 28

Isolated CAN models

CAN1 and CAN2 do not have internal 120Ω termination. Pin 34 is the CAN-circuit isolated (reference) ground. Do not use the controller's I/O ground (pins 7 and 18) for the isolated CAN connections.

CAN1

CAN1 L = pin 35
CAN1 H = pin 23

CAN2

CAN2 L = pin 29
CAN2 H = pin 28

Isolated GND = pin 34

Pin 21 = Not used (no connection).

APPLICATION SETUP/CAN INTERFACE — PDO SETUPS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
RPDO1			Note: PDO transmissions are from the perspective of the ancillary controller. RPDOs are messages received by the ancillary (i.e., sent from the manager).
Timeout *can_rpdo_1_event_timer* `0x1400 0x05`	0 – 65535 *0 – 65535*	40 ms	Sets the PDO timeout period for the CANopen Ancillary system. After the ancillary controller has sent a TPDO (PDO-TX), it will declare a PDO Timeout Fault if the manager controller has not sent a reply RPDO (PDO-RX) message within the time set by this parameter. Any RPDO1-4 message will reset the timer. Setting this parameter to zero (Timeout = 0) disables the PDO timeout fault check. *See Appendix A for PDO mapping/setup.* *Application Note: Based upon the F-Series cdev revision, this timer may begin as soon as the controller goes operational (NMT = 5), and not upon the first RPDO message received. Check this aspect using a CANbus trace if unexpected PDO Timeout errors occur.*
COB ID *can_rpdo_1_cob_id* `0x1400 0x01`	0 – 0xFFFFFFFF *0 – 42949672295*	80000226h	The application's 11-bit COB-ID (Communication Object ID). Note: Use VCL for a 29-bit COB-ID. See Appendix A.
Length *can_rpdo_1_length* `0x1600 0x00`	0 – 8 *0 – 8*	0	Number of CAN objects in map (not the number of bits or bytes). See Appendix A.
1 *can_rpdo_1_map_1* `0x1600 0x01`	0h – FFFFFFFFh *0 – 4294967295*	0x00050008 *327688*	The Manager to Ancillary communication RPDO message(s). Map 1st object variable and its bit length (8 = 08h, 16 = 10h, 24 = 18h, or 32= 20h). Map the value as the CAN-Index+SubIndex+Length. If = 0x00050008, this is a dummy 8-bit PDO variable (as per CANopen). Note: 50008h = 0x00050008.
2 *can_rpdo_1_map_2* `0x1600 0x02`	0h – FFFFFFFFh *0 – 4294967295*	0x00050008 *327688*	Map 2nd object variable and bit length (8, 16, 24, or 32). See Appendix A.
3 *can_rpdo_1_map_3* `0x1600 0x03`	0h – FFFFFFFFh *0 – 4294967295*	0x00050008 *327688*	Map 3rd object variable and bit length (8, 16, 24, or 32). See Appendix A.
4 *can_rpdo_1_map_4* `0x1600 0x04`	0h – FFFFFFFFh *0 – 4294967295*	0x00050008 *327688*	Map 4th object variable and bit length (8, 16, 24, or 32). See Appendix A.
5 *can_rpdo_1_map_5* `0x1600 0x05`	0h – FFFFFFFFh *0 – 4294967295*	0x00050008 *327688*	Map 5th object variable and bit length (8, 16, 24, or 32). See Appendix A.
6 *can_rpdo_1_map_6* `0x1600 0x06`	0h – FFFFFFFFh *0 – 4294967295*	0x00050008 *327688*	Map 6th object variable and bit length (8, 16, 24, or 32). See Appendix A.
7 *can_rpdo_1_map_7* `0x1600 0x07`	0h – FFFFFFFFh *0 – 4294967295*	0x00050008 *327688*	Map 7th object variable and bit length (8, 16, 24, or 32). See Appendix A.
8 *can_rpdo_1_map_8* `0x1600 0x08`	0h – FFFFFFFFh *0 – 4294967295*	0x00050008 *327688*	Map 8th object variable and bit length (8, 16, 24, or 32). See Appendix A.

Quick Link:
Appendix A p.240

APPLICATION SETUP/CAN INTERFACE — PDO SETUPS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
TPDO1	Note: PDO transmissions are from the perspective of the ancillary controller. TPDOs are messages transmitted from the ancillary (i.e., sent to the manager).		
Event Time *can_tpdo_1_event_timer* `0x1800 0x05`	0 – 65535 *0 – 65535*	0	The Event Time sets the periodical transmission rate of the TPDOs, in milliseconds. This is a local timer, and not synchronized with other devices on the network. See Appendix A for setting up the TPDO mapping.
COB ID *can_tpdo_1_cob_id* `0x1800 0x01`	0h- FFFFFFFFh *0 – 4294967295*	0xC00001A6h	The application's 11-bit COB-ID (Communication Object ID). Note: Use VCL for a 29-bit COB-ID. See Appendix A.
Length *can_tpdo_1_length* `0x1A00 0x00`	0 – 8 *0 – 8*	0	Number of CAN objects in map (not the number of bits or bytes). See Appendix A.
1 *can_tpdo_1_map_1* `0x1A00 0x01`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	The Ancillary to Manager communication TPDO message(s). TPDO1. Map 1st object variable and its bit length (8 = 08h, 16 = 10h, 24 = 18h, or 32= 20h). Map the value as the CAN-Index+SubIndex+Length. If = 0x00050008, this is a dummy 8-bit PDO variable (as per CANopen). Note: 50008h = 0x00050008.
2 *can_tpdo_1_map_2* `0x1A00 0x02`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 2nd object variable and bit length (8,16, 24, or 32). See Appendix A.
3 *can_tpdo_1_map_3* `0x1A00 0x03`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 3rd object variable and bit length (8,16, 24, or 32). See Appendix A.
4 *can_tpdo_1_map_4* `0x1A00 0x04`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 4th object variable and bit length (8,16, 24, or 32). See Appendix A.
5 *can_tpdo_1_map_5* `0x1A00 0x05`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 5th object variable and bit length (8,16, 24, or 32). See Appendix A.
6 *can_tpdo_1_map_6* `0x1A00 0x06`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 6th object variable and bit length (8,16, 24, or 32). See Appendix A.
7 *can_tpdo_1_map_7* `0x1A00 0x07`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 7th object variable and bit length (8,16, 24, or 32). See Appendix A.
8 *can_tpdo_1_map_8* `0x1A00 0x08`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 8th object variable and bit length (8,16, 24, or 32). See Appendix A.

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
RPDO {2, 3, 4} Byte Map			See RPDO 1 (above) and Appendix A.
TPDO {2, 3, 4} Byte Map			See TPDO 1 (above) and Appendix A.

Quick Link:
Appendix A p.240

APPLICATION SETUP — CAN INTERFACE 2 MENU

CAN2: The second CAN port operates with a different parameter value when the matching parameter has a different CAN Object Index. When the CAN Object is the same, set both CAN1 and CAN2 parameters at the same value. Always use the CAN Interface 2 as the secondary port. The CAN Interface (CAN 1) is for the secure node connections to the Curtis programming tools.

APPLICATION SETUP — CAN INTERFACE 2 MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
CANopen Interlock *CANopen_Interlock_Enable* `0x34B4 0x00`	On/Off *On/Off*	Off	When programmed On, CAN NMT State must = 5 (operational state) in order for the interlock to be set. This parameter also occurs in the CAN Interface menu. Changing it in one menu also changes it in the other location.
Baud Rate *CAN_2_Baud_Rate* `0x2001 0x02`	–1 – 4 *–0 – 4*	1	Sets the CAN baud rate for the CANopen Ancillary system: –1 = 100Kbps 0 = 125Kbps 1 = 250Kbps 2 = 500Kbps 3 = 800Kbps 4 = 1000Kbps.
Heartbeat Rate *CANopen_Heart_Beat_Rate* `0x1017 0x00`	0 – 2000 ms *0 – 2000*	100 ms	Sets the transmission rate of the CAN2 heartbeat messages from the CANopen Ancillary system. Notice: If set to 0, restores to 20ms upon KSI cycle.
Emergency Message Rate *CAN_Open_Emergency_Inhibit_Time* `0x1015 0x00`	0 – 6554 ms *160 – 65535*	16 ms	Sets the minimum transmission rate of the CAN2 Emergency Messages from the CANopen Ancillary system. This prevents quickly changing fault states from generating so many emergency messages that they flood the CANbus.
CAN NMT State *CAN_NMT_State* `0x32A4 0x00` This Read Only variable also occurs in the CAN Interface menu.	0 – 127 *0 – 127*	Read Only	Controller CAN NMT state: 0 = initialization 4 = stopped 5 = operational 127= pre-operational
CAN Node ID *Can_2_Node_Id* `0x2000 0x02`	1h – FFFFh *1 – 65535*	27h	Displays the controller's Node ID, in hexadecimal. For example: 27h = 0x27 = 39d Reference Appendix D for setting multiple generic controllers. Generic controllers all have the same default Node ID from the factory. Do not assign 0x00 as a device's CAN Node ID. Such an ID will never be detected by CIT or the 1313 HHP.

APPLICATION SETUP/CAN INTERFACE 2 — PDO SETUPS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
RPDO1 (CAN2)	Note: PDO transmissions are from the perspective of the ancillary controller. RPDOs are messages received by the ancillary (i.e., sent from the manager).		
Timeout *can_rpdo_1_event_timer_p2* `0x1440 0x05`	0 – 65535 *0 – 65535*	40 ms	Sets the PDO timeout period for the CANopen Ancillary system. After the ancillary controller has sent a TPDO (PDO-TX), it will declare a PDO Timeout Fault if the manager controller has not sent a reply RPDO (PDO-RX) message within the time set by this parameter. Any RPDO1-4 message will reset the timer. Setting this parameter to zero (Timeout = 0) disables the PDO timeout fault check. See Appendix A for PDO mapping/setup. *Application Note: Based upon the F-Series cdev revision, this timer may begin as soon as the controller goes operational (NMT = 5), and not upon the first RPDO message received. Check this aspect using a CANbus trace if unexpected PDO Timeout errors occur.*
COB ID *can_rpdo_1_cob_id_p2* `0x1440 0x01`	0h – FFFFFFFFh *0 – 4294967295*	80000227h	The application's 11-bit COB-ID (Communication Object ID). Note: Use VCL for 29-bit COB-ID. See Appendix A. Note: Hexadecimal 80000227h = 0x80000227.
Length *can_rpdo_1_length_p2* `0x1640 0x00`	0 – 8 *0 – 8*	0	Number of CAN objects in map (not the number of bits or bytes). See Appendix A.
1 *can_rpdo_1_map_1_p2* `0x1640 0x01`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 1st object variable and bit length (8, 16, 24, or 32). See Appendix A. The default value, 0x00050008 is a dummy 8-bit PDO variable (as per CANopen). Note: 50008h = 0x00050008.
2 *can_rpdo_1_map_2_p2* `0x1640 0x02`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 2nd object variable and bit length (8, 16, 24, or 32). See Appendix A.
3 *can_rpdo_1_map_3_p2* `0x1640 0x03`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 3rd object variable and bit length (8, 16, 24, or 32). See Appendix A.
4 *can_rpdo_1_map_4_p2* `0x1640 0x04`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 4th object variable and bit length (8, 16, 24, or 32). See Appendix A.
5 *can_rpdo_1_map_5_p2* `0x1640 0x05`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 5th object variable and bit length (8, 16, 24, or 32). See Appendix A.
6 *can_rpdo_1_map_6_p2* `0x1640 0x06`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 6th object variable and bit length (8, 16, 24, or 32). See Appendix A.
7 *can_rpdo_1_map_7_p2* `0x1640 0x07`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 7th object variable and bit length (8, 16, 24, or 32). See Appendix A.
8 *can_rpdo_1_map_8_p2* `0x1640 0x08`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 8th object variable and bit length (8, 16, 24, or 32). See Appendix A.

Quick Link:
Appendix A p.240

APPLICATION SETUP/CAN INTERFACE 2 — PDO SETUPS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
TPDO1 (CAN2)	Note: PDO transmissions are from the perspective of the ancillary controller. TPDOs are messages transmitted from the ancillary (i.e., sent to the manager).		
Timeout *can_tpdo_1_event_timer_p2* `0x1840 0x05`	0 – 65535 *0 – 65535*	40	The Event Time sets the periodical transmission rate of the TPDOs, in milliseconds. This is a local timer, and not synchronized with other devices on the network. See Appendix A for setting up the TPDO mapping.
COB ID (TPDO 1, CAN2) *can_tpdo_1_cob_id_p2* `0x1840 0x01`		C00001A7h	The application's 11-bit COB-ID (Communication Object ID). Note: Use VCL for a 29-bit COB-ID. See Appendix A. Note: Hexadecimal 80000227h = 0x80000227.
Length (TPDO 1, CAN2) *can_tpdo_1_length_p2* `0x1A40 0x00`	0 – 8 *0 – 8*	0	Number of CAN objects in map (not the number of bits or bytes). See Appendix A.
1 *can_tpdo_1_map_1_p2* `0x1A40 0x01`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 1st object variable and bit length (8, 16, 24, or 32). See Appendix A. The default value, 0x00050008 is a dummy 8-bit PDO variable (as per CANopen). Note: 50008h = 0x00050008.
2 *can_tpdo_1_map_2_p2* `0x1A40 0x02`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 2nd object variable and bit length (8, 16, 24, or 32). See Appendix A.
3 *can_tpdo_1_map_3_p2* `0x1A40 0x03`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 3rd object variable and bit length (8,16, 24, or 32). See Appendix A.
4 *can_tpdo_1_map_4_p2* `0x1A40 0x04`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 4th object variable and bit length (8,16, 24, or 32). See Appendix A.
5 *can_tpdo_1_map_5_p2* `0x1A40 0x05`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 5th object variable and bit length (8,16, 24, or 32). See Appendix A.
6 *can_tpdo_1_map_6_p2* `0x1A40 0x06`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 6th object variable and bit length (8,16, 24, or 32). See Appendix A.
7 *can_tpdo_1_map_7_p2* `0x1A40 0x07`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 7th object variable and bit length (8,16, 24, or 32). See Appendix A.
8 *can_tpdo_1_map_8_p2* `0x1A40 0x08`	0h – FFFFFFFFh *0 – 4294967295*	50008h *327688*	Map 8th object variable and bit length (8,16, 24, or 32). See Appendix A.

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
RPDO {2, 3, 4} Byte Map			See RPDO 1 (above) and Appendix A (CAN2).
TPDO {2, 3, 4} Byte Map			See TPDO 1 (above) and Appendix A (CAN2).

Quick Link:
Appendix A p.240

APPLICATION SETUP — BATTERY SETUP MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Keyswitch Voltage *Keyswitch_Voltage* `0x3398 0x00`	0.00 – 105.00V *0 – 10500*	Read Only	Voltage at KSI (pin 1).
Nominal Voltage *Nominal_Volts* `0x33AA 0x00`	24.0 – 60.0V *2400 – 6000* *The default is controller model based.	24 Volt*	This parameter is limited to the controller's model voltage. It must be set to the vehicle's nominal battery pack voltage. Also, use this parameter in determining the overvoltage and undervoltage protection thresholds. See the defined nominal voltage in Table E-1 in Appendix E, and the Voltage Limits.
BDI *BDI_Percentage* `0x33A5 0x00`	0 – 100 % *0 – 100*	% (0-100)	Battery Discharge Indicator (BDI). This is the battery State of Charge (SOC) as a percentage. The percentage is based upon lead-acid battery chemistry and the associated parameters in the *BDI Setup* menu (below).
Battery Current *Battery_Current_Display* `0x338F 0x00`	–3276.8 – 3276.7A *–32768 – 32767*	Read Only	Calculated value in DC Amps. The value is in 0.1A steps.
Battery Power *Battery_Power* `0x3390 0x00`	–3276.8 – 3276.7 kW *–327678 – 32767*	Read Only	Calculated value in Watts (W = V x I). The value is in 0.1 kW steps.
Battery Current Limiter Enable *Battery_Current_Limiter_Enable* `0x3C00 0x00`	Off/On 0/1	Off	Setting this parameter to 1 enables drive and regen battery current limits. This is particularly useful when interfacing to Lithium Batteries with defined current limits.

Quick Links:
Voltage limits p.33
BDI info p.35
BDI menu p.78
Table E-1 p.262

APPLICATION SETUP/BATTERY SETUP — UNDERVOLTAGE CONTROLLER MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
User Undervoltage *User_Undervoltage* `0x33A1 0x00`	0 – 100 % *0 – 256*	70 %	The value of this parameter is a percentage of the Nominal Voltage setting. Utilize the User Undervoltage parameter to adjust the undervoltage threshold, which is the voltage when the controller begins cutting back drive current to prevent damage to the electrical system or an untimely shutdown. Typically, this parameter is applicable in an application at the high end of the controller's voltage range, such as a 24-36V controller in a system with a 36V battery pack. In this case, the undervoltage threshold adjustment enables a usable battery voltage by setting the User Undervoltage to an appropriate value, versus that of the controller's 24V undervoltage specification. Note that the undervoltage threshold can never be a lower value than the controller's power base minimum voltage rating. See the Voltage Limits section in Chapter 3, and the controller specifications in Appendix E.
Kp UV *Batt_Kp_UV* `0x338B 0x00`	0.0 – 100.0 % *0 – 1024*	2.0 %	When the battery voltage goes below the undervoltage threshold, a closed loop PI (Proportional/Integral) algorithm attempts to keep the battery voltage from drooping. It accomplishes this by cutting back the drive current thereby reducing the load on the battery. The Kp term is the proportional gain and is set in units of percent cutback per volt; for example, a setting of 25 would provide full current cutback with 4V of droop. The Ki term is the integral gain. Integral gain will accumulate the voltage droop and attempt to bring the battery droop back to 0V. Higher gains will react more strongly and quickly.
Ki UV *Batt_Ki_UV* `0x3389 0x00`	0 – 100 % *0 – 16384*	50 %	

APPLICATION SETUP/BATTERY SETUP — OVERVOLTAGE CONTROLLER MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
User Overvoltage *User_Overvoltage* `0x33A0 0x00`	100 – 200 % *256 – 512*	125 %	The value of this parameter is a percentage of the Nominal Voltage setting. Utilize the User Overvoltage parameter to adjust the overvoltage threshold, which is the voltage when the controller begins cutting back regen braking to prevent damage to the electrical system or an untimely overvoltage shutdown. Typically, this parameter is applicable in an application at the low end of the controller's voltage range, such as a 48-80V controller with a 48V battery pack. In this case, the overvoltage threshold adjustment enables a usable battery voltage by setting the User Overvoltage to an appropriate value, versus that of the controller's 80V overvoltage specification. The overvoltage threshold can never be raised above the controller's power base maximum voltage rating.

APPLICATION SETUP/BATTERY SETUP — BDI SETUP MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Reset Volts Per Cell *BDI_Reset_Volts_Per_Cell* `0x33A7 0x00`	0.900 – 3.000V *900 – 3000*	2.090V	The reset voltage level is checked only once when the keyswitch is first turned on. The **BDI Reset Percent** parameter also influences the algorithm that determines whether the BDI percentage is reset to 100% at key on. The **Reset Volts Per Cell** parameter should always be set higher than the **Full Volts Per Cell** parameter. This parameter is applicable for lead-acid battery chemistry. Reset Voltage Level = (Reset Volts Per Cell) × (Number of cells in the battery pack).*
Full Volts Per Cell *BDI_Full_Volts_Per_Cell* `0x33A4 0x00`	0.900 – 3.000V *900 – 3000*	2.040V	This parameter sets the *Keyswitch Voltage* variable considered to be 100% state-of-charge, and when a loaded battery drops below this voltage, it begins to lose charge (lowering the BDI percentage). The *Keyswitch Voltage* variable is viewable in the programmer's *Monitor » Battery* menu. Full Voltage Level = (Full Volts Per Cell) × (Number of cells in the battery pack).*
Empty Volts Per Cell *BDI_Empty_Volts_Per_Cell* `0x33A3 0x00`	0.900 – 3.000V *900 – 3000*	1.730V	The empty voltage level sets the Keyswitch_Voltage that is considered to be 0% state-of-charge. Empty Voltage Level = (Empty Volts Per Cell) × (Number of cells in the battery pack).*
Discharge Time *BDI_Discharge_Time* `0x33A2 0x00`	0 – 600 min *0 – 600*	34 min	This parameter sets the minimum time for the BDI algorithm to count down the BDI Percentage from 100% to 0%. The BDI algorithm integrates the time the filtered keyswitch voltage is below the state of charge voltage level. When that cumulative time exceeds the Discharge Time/100, the BDI Percentage is decremented by one percentage point and a new state of charge voltage level is calculated. State of Charge Level = ((Full Voltage Level - Empty Voltage Level) × BDI Percentage / 100) + (Empty Voltage Level).
BDI Reset Percent *BDI_Reset_Percent* `0x33A6 0x00`	0 – 100 % *0 – 100*	75 %	When a battery has a high BDI percentage, its float voltage at KSI On can sometimes cause false resets. The BDI Reset Percent parameter addresses this problem by allowing the user to define a BDI Percentage value above which the BDI Percentage variable will not reset. When KSI is first powered on, the BDI Percentage variable will reset to 100% only if [Keyswitch Voltage > Reset Voltage Level] and [BDI Percentage < BDI Reset Percent].

Quick Links:
BDI Indicator p.35

** To determine the number of cells in the battery pack, divide the Nominal Voltage setting (above) by the battery chemistry's nominal volts-per-cell. Lead-acid: 2.0V/cell, nominal.*

Note: For non-lead-acid batteries, including Lithium-Ion battery packs, use the pack's or cell manufacturer's approved Battery Management System (BMS) for the SOC.

APPLICATION SETUP — MAIN CONTACTOR MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Main Enable *Main_Enable* `0x34C5 0x00`	On/Off *On/Off*	On	When programmed On, the controller's native software controls the main contactor when enabling the interlock. When programmed Off, VCL controls the main contactor. See the assignment parameter in *Programmer/Controller Setup/ IO Assignments/Coil Drivers/Main Contactor Driver*. The default is Driver 3. **Note: With Main Enable programmed Off the controller will not be able to open the main contactor in serious fault conditions and the system will therefore not meet EEC safety requirements.**
Main State *Main_State* `0x34C9 0x00`	0 – 10 *0 – 10*	Read Only	Main contactor state: 0 = open 1 = precharge 2 = weld check 3 = closing delay 4 = missing check 5 = closed (when Main Enable = On) 6 = delay 7 = arc check 8 = open delay 9 = fault 10 = closed (when Main Enable = Off)
Keyswitch Voltage *Keyswitch_Voltage* `0x3398 0x00`	0.00 – 105.00V *0 – 10500*	Read Only	Voltage at KSI (pin 1). The value is in 0.01V steps.
Capacitor Voltage *Capacitor_Volts* CAN = `0x34C1 0x00`	0.00 – 200.00V *0 – 20000*	Read Only	Voltage of the controller's internal capacitor bank at the B+ terminal. The value is in 0.01V steps. Note that the precharge, main weld check and DNC tests affect this reading when those actions are active.
Pull In Voltage *Main_Pull_In_Voltage* `0x34C8 0x00`	0 – 100 % *0 – 32767*	0 %	The main contactor pull-in-voltage parameter allows a high initial voltage when the main contactor driver first turns on to ensure contactor closure. After 1 second, this peak voltage drops to the contactor holding voltage. Typical is 100% to ensure closure. Note: The Battery Voltage Compensated parameter (0x34C4) controls whether the pull-in and holding voltages are battery voltage compensated.
Holding Voltage *Main_Holding_Voltage* `0x34C6 0x00`	0 – 100 % *0 – 32767*	0 %	The main contactor holding-voltage parameter allows a reduced average voltage to be applied to the contactor coil once it has closed. Typical is 75-80% to conserve energy and coil heating. Set this parameter (applied voltage) high enough to hold the contactor closed during all vehicle shock and vibration conditions. If the voltage is too low, the contactor tips may open when subjected to shock and vibration conditions. Note: The Battery Voltage Compensated parameter (0x34C4) controls whether the pull-in and holding voltages are battery voltage compensated.
Battery Voltage Compensated *Main_Driver_Battery_Voltage_ Compensated* `0x34C4 0x00`	On/Off *On/Off*	On	This parameter determines whether the main pull-in and holding voltages are battery voltage compensated. When set On, the pull-in and holding voltages are set relative to the set Nominal Voltage. In other words the output voltage is adjusted to compensate for swings in battery voltage so the percentage is relative to the set Nominal Voltage not to the actual voltage. For example suppose Nominal Voltage is set to 48V and Holding Voltage is set to 75 % (36V) to the output driver. Now suppose the bus voltage dips to 40V. If Battery Voltage Compensated = On, the output will still be 36V (Nominal Voltage × Holding Voltage) to the coil. If the Battery Voltage Compensated = Off, the output will be 30V (Actual Voltage × Holding Voltage) to the coil.

APPLICATION SETUP — MAIN CONTACTOR MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Open Delay *Open_Delay* `0x34CA 0x00`	0.0 – 40.0 s *0 – 10000*	0.1 sec	The delay can be set to allow the contactor to remain closed for a period of time (the delay) after opening the interlock switch. The delay is useful for preventing unnecessary cycling of the contactor and for maintaining battery voltage to auxiliary functions and devices for a short time after the interlock switch has opened. See the basic wiring diagram for the main contactor's location between the battery and controller.
Weld Check Enable *Weld_Check_Enable* `0x34D0 0x00`	On/Off *On/Off*	On	When programmed On, the controller performs a test to make sure the main contactor is open (not welded shut) before it is commanded to close. This is accomplished by passing a small current through a connected motor's phase leads, to bleed-down the controller's capacitor bank a few volts. If the contactor is "welded shut", the connected-battery will prevent any voltage drop. When this parameter is Off, this test is not performed. The main contactor driver, however, is always protected from short circuits if the contactor tips are welded shut. Note, due to the nature of this test, an unloaded motor may rotate (oscillate) slightly. This is expected (the physics of the test).
Main DNC Check Enable *Main_DNC_Check_Enable* `0x34C2 0x00`	On/Off *On/Off*	On	When programmed On, the controller performs a test immediately after commanding the main contactor to close, making sure the contactor has in fact closed. The test is similar to the Weld Check, yet in this case, the battery upholds the capacitor voltage. When this parameter is Off, this test is not performed. The main contactor driver, however, is always protected from short circuits if the contactor tips are welded shut. Note, due to the nature of this test, an unloaded motor may rotate (oscillate) slightly. This is expected (the physics of the test).
Main DNC Runtime Threshold *Main_DNC_Runtime_Volts* `0x34C3 0x00`	0.0 – 200.0V *0 – 20000*	5.0V	Sets the threshold used for the ongoing check that ensures the main contactor remains closed while in operation. The Main DNC Runtime Threshold is the maximum voltage difference between the Keyswitch and Capacitor voltages. When the voltage difference is above this threshold and the battery current is low, a Main Did Not Close fault will be set. Setting this parameter lower will increase the sensitivity of the fault detection. Setting this parameter too low may cause false fault trips due to normal voltage drops between the keyswitch and capacitor voltages. Setting this parameter = 0 volts will disable the Main Did Not Close fault check.
Precharge Enable *Precharge_Enable* `0x34CC 0x00`	On/Off *On/Off*	On	Turns the precharge feature on and off. Precharge provides a limited current, via the keyswitch input, to charge the controller's internal capacitor bank before the main contactor is closed. This decreases the arcing that would otherwise occur when closing the contactor with a discharged capacitor bank. The Precharge must be set to On to enable the Weld Check.

APPLICATION SETUP — EM BRAKE CONTROL MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
EM Brake State *EMBrakeState* `0x347A 0x00`	0 – 4 *0 – 4*	Read Only	EM Brake State: 0 = engaged 1 = releasing 2 = released 3 = engaging 4 = engaged and vehicle stopped
EM Brake Type *EM_Brake_Type* `0x3479 0x00`	0 – 2 *0 – 2*	0 **[PCF]**	The brake type parameter determines how the EM brake responds to the interlock input throttle and vehicle motor speed. **0.** EM brake function disabled. Releases the default EM brake Driver to general I/O use with VCL. **1.** EM brake controlled by interlock. The controller will command the EM brake to release whenever the interlock is closed (Interlock = On). If interlock braking is enabled and the interlock opens when the vehicle is moving at motor speed greater than *Zero_Speed_Threshold* the controller will brake the vehicle to a stop (with interlock braking) and then command the EM brake to set. If the vehicle motor speed is less than this threshold, the EM brake will engage after the *Sequencing_Delay* has expired. If interlock braking is disabled, the EM brake will engage after the *Sequencing_Delay* has expired. **2.** EM brake controlled by interlock and neutral. The controller will command the EM brake to set whenever the throttle command is zero and motor speed is less than *Zero_Speed_Threshold*. Position Hold will be enabled automatically.
Pull In Voltage *EM_Brake_Pull_In_Voltage* `0x3473 0x00`	0 – 100 % *0 – 32767*	0 %	The EM brake pull-in voltage allows a high initial voltage when the EM brake first turns on to ensure brake release. After 1 second, this peak voltage drops to the EM brake holding voltage. Note: The Battery Voltage Compensated parameter (0x3470) controls whether the pull-in and holding voltages are battery voltage compensated.
Holding Voltage *EM_Brake_Holding_Voltage* `0x3472 0x00`	0 – 100 % *0 – 32767*	0 %	The EM brake holding voltage allows a reduced average voltage to be applied to the brake coil once the brake releases. Set this parameter (applied voltage) high enough to hold the brake off during all vehicle shock and vibration conditions. If the voltage is too low, the brake can reset when subjected to shock and vibration conditions. Note: The EM Brake's Battery Voltage Compensated parameter (0x3470) controls whether the pull-in and holding voltages are battery voltage compensated.
Battery Voltage Compensated *EM_Brake_Battery_Voltage_Compensated* `0x3470 0x00`	On/Off *On/Off*	On	This parameter determines whether the EM brake pull-in and holding voltages are battery voltage compensated. When set On, the pull-in and holding voltages are compensated relative to the Nominal Voltage setting. In other words, the output voltage is adjusted to compensate for swings in battery voltage so the percentage is relative to the set Nominal Voltage not to the actual voltage. For example, suppose Nominal Voltage setting is 48V and the Holding Voltage setting is 75% (36V) to the output driver. Now suppose the bus voltage dips to 40V. If Battery Voltage Compensated = On, the output will still be 36V to the coil (Nominal Voltage × Holding Voltage). If Battery Voltage Compensated = Off, the output will be 30V to the coil (Actual Voltage × Holding Voltage).

APPLICATION SETUP — EM BRAKE CONTROL MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Set EM Brake On Fault *EM_Brake_Set_Upon_Fault* 0x3475 0x00	On/Off *On/Off*	On	When programmed On, the controller's operating system will stop the assigned EM Brake's coil driver whenever a fault occurs that has the ShutdownEMBrake fault action. The de-energized electromagnetic brake coil engages the EM brake ("drops the EM Brake").
Zero Speed Threshold *Zero_Speed_Threshold* 0x30F2 0x00	5 – 300 rpm *5 – 300*	30 rpm	Determines the speed below which the EM brake is set (On). Setting this speed too high may cause a jerky stop when the EM brake sets and stops the motor rotation. This parameter appears twice in the menu structure. Changing the value of this parameter affects this (same) parameter in the Position Hold menu.
Zero Speed Threshold Time *Zero_Speed_Threshold_Time* 0x30F3 0x00	0 – 480 ms *0 – 60*	32 ms	Determines how long motor speed must be below Zero_Speed_Threshold to declare zero speed. This parameter appears twice in the menu structure. Changing the value of this parameter affects this (same) parameter in the Position Hold menu.
Position Hold Settling Time *Position_Hold_Settling_Time* 0x3783 0x00	0.0 – 5.0 s *0 – 156*	3.0 sec	Determines how long the position hold function is allowed to operate before the EM brake is set. This time should be set long enough for the position hold to settle. This parameter appears twice in the menu structure. Changing the value of this parameter affects this (same) parameter in the Position Hold menu.
Brake Set Time *EM_Brake_Set_Time* 0x3484 0x00	40 – 2000 ms *5 – 250*	800 ms	The estimated time for the EM brake to physically set after releasing the holding voltage. This determines how long the controller waits, after removing voltage from the EM Brake Driver, before releasing the motor torque. This should be set longer than the actual brake setting time to ensure the vehicle does not move before the brake fully engages.
Torque Release Time *EM_Brake_Torque_Release_Time* 0x3481 0x00	40 – 2000 ms *5 – 250*	200 ms	Sets the time (in milliseconds) to release the torque after EM Brake has set and the motor has stopped (EMBrakeEngagedAndStopped state).
Brake Release Time *EM_Brake_Release_Time* 0x3483 0x00	40 – 2000 ms *5 – 250*	48 ms	Estimated time for the EM brake to physically release after the pull-in voltage is applied. Use this to ensure the position hold's torque buildup is complete before the brake releases. When set too low the vehicle may experience rollback on EM brake release.
Torque Preload Time *EM_Brake_Torque_Preload_Time* 0x3482 0x00	0 – 800 ms *0 – 100*	200 ms	The estimated worst-case time to build up the torque required to hold the vehicle stationary on a hill prior to EM brake release. Use this in conjunction with Release Delay to determine when to release the brake and allow the speed request to slew away from zero.
Torque Preload Enable *EM_Brake_Torque_Preload_Enable* 0x3478 0x00	On/Off *On/Off*	On	When enabled, this function eliminates rollback when the throttle is re-engaged on a ramp by forcing the vehicle to first enter position-hold before setting the EM brake and then remembering the amount of torque that was necessary to hold it on the ramp. When the throttle is re-engaged, this value is loaded in the motor before releasing the EM brake. The torque value is cleared automatically when KSI power is cycled. Off = When a valid throttle input is received, the speed controller will start with no torque preload as soon as the Release Delay expires. This will allow some rollback when the EM brake releases. This rollback can be reduced by raising the Exit Rollback Reduction parameter in the Position Hold menu. On = When a valid throttle input is received, the speed controller will start with a pre-set torque as measured by the position hold when the vehicle came to a stop.

APPLICATION SETUP — EM BRAKE CONTROL MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Save Torque Preload *Save_Torque_Preload* `0x3474 0x00`	On/Off *On/Off*	Off	Enabling this parameter will save the Torque Preload (*HillHoldMemory*, 0x347C) across keyswitch cycles. If *EM_Brake_Torque_Preload_Cancel_Delay* is nonzero and this parameter = On, the timer starts again upon startup (KSI cycle) such that the countdown is from the full value of the *EM_Brake_Torque_Preload_Cancel_Delay* parameter.
Torque Preload Cancel Delay *EM_Brake_Torque_Preload_Cancel_Delay* `0x3476 0x00`	0 – 120 s *0 – 15000*	0 sec	The timer starts after the EM brake is set. If the timer expires before the throttle is re-engaged, the torque preload memory is cleared. Setting this parameter to zero disables the timer (i.e., the preload is never cancelled). The purpose of this delay is to prevent the vehicle from lunging forward if it is unloaded on a hill such that the torque measured by position-hold is no longer valid. Note: This parameter is applicable only when Torque Preload Enable = On (see above).
EM Brake Fault Motor Revs *EM_Brake_Fault_Motor_Revs* `0x3471 0x00`	1.0 – 20.0 *10 – 200*	4.0	Defines the allowable number of motor revolutions after the EM brake is set before triggering the EM BRAKE FAILED TO SET fault (Fault Code 9-2).

APPLICATION SETUP \EM BRAKE CONTROL — EM BRAKE TEST MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Mode *EM_Brake_Test_Mode* `0x4F5E 0x00`	Enumeration* *0 – 1* *VCL only = 0 *At Startup = 1	VCL Only	Controls the test mode for the EM brake.
Direction *EM_Brake_Test_Direction* `0x4F5F 0x00`	Enumeration* *0 – 2* **Forward = 0* **Reverse = 1* **Both = 2*	Forward	Controls which direction torque is applied during the EM brake test.
Current *EM_Brake_Test_Current* `0x4F60 0x00`	0 – 100 % *0 – 32767*	20%	The current as a percentage of base current applied during the EM brake test.
Ramp Time *EM_Brake_Test_Current_Ramp_Time* `0x4F61 0x00`	0 – 1000 ms *0 – 1000*	500 ms	The time taken to ramp the current to the EM_Brake_Test_Current value. The same time is used to ramp it down again.
Test Threshold *EM_Brake_Test_Threshold* `0x4F62 0x00`	1.0 – 20.0 rpm *10 – 200*	1.0 rpm	The number of motor revolutions the motor can move during the EM brake test before a fault is declared.
Hold Time *EM_Brake_Test_Hold_Time* `0x4F63 0x00`	0.0 – 1.0 sec *0 – 125*	0.5 s	The amount of time the torque is held between ramp up and ramp down times.
Max Speed Limit *Max_Speed_EM_Brake_Fault* `0x4F68 0x00`	0 – 24000 *0 – 24000*	800	Set the motor rpm above which the EM Brake fault is triggered.
Hill Compensation Enable *EM_Brake_Test_Hill_Compensation_Enable* `0x4F67 0x00`	0 – 1 *0 – 1*	0	Enables or disables modifying the torque applied during the EM brake test while the vehicle is on a hill. If turned on, HillHoldMemory is added to the request to avoid overstressing the brake in 1 direction and not sufficiently testing it in the other.
EM Brake Test State *EM_Brake_Test_State* `0x4F66 0x00`	0 – 4 *0 – 4*	0	The current state of the EM brake test. Can be Stop (0), Start (1), Running (2), Done (3), or Failed (4).

APPLICATION SETUP — EMERGENCY REVERSE (EMR) MENU

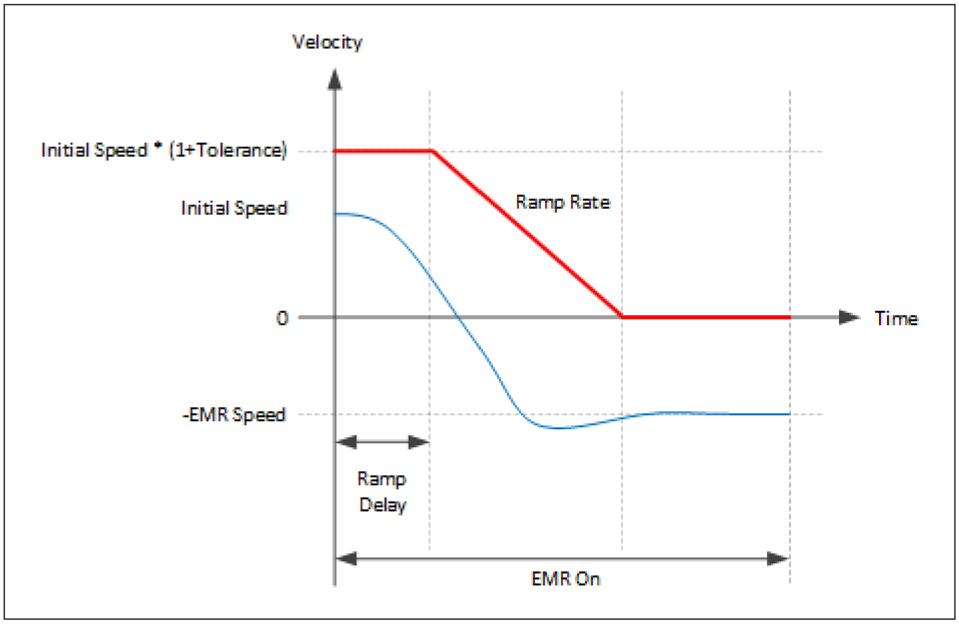
PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
EMR Enable *EMR_Enable* 0x3495 0x00	On/Off *On/Off*	Off	Enables or disables the Emergency Reverse function. On = emergency reverse is enabled. Off = emergency reverse is disabled.
EMR State *EMR_State* 0x3490 0x00	On/Off *On/Off*	Read Only	Emergency reverse input is On or Off, from the selected source (Programmer *IO Assignments » Controls menu*).
EMR Forward Lock *Forward_EMR_Lock* 0x349B 0x00	On/Off *On/Off*	Off	Only allows EMR to operate if the vehicle is moving forward.
EMR Dir Interlock *EMR_Dir_Interlock* 0x3494 0x00	On/Off *On/Off*	Off	Determines whether the cycling of the interlock switch is required following an emergency reverse event before driving the vehicle, again. On = Interlock and throttle and direction must all be cleared (cycled). Off = Only throttle and direction must be cleared (cycled).
EMR Time Limit *EMR_Time_Limit_mSec* 0x3497 0x00	0.0 – 30.0 s *0 – 30000*	3.0 sec	Defines how long emergency reverse is active after the vehicle is moving in the reverse direction. This timer will restart if the vehicle ever goes forward while emergency reverse is still active. The allowable range is 0-30 seconds where 30 seconds is a special case of no time out. When emergency reverse times out the Emer Rev Timeout fault is set. Cycling the emergency reverse input will clear the Emer Rev Timeout fault. To stop the vehicle after an EMR event (not move in reverse direction) set this parameter to 0.
EMR Speed *EMR_Speed* 0x3496 0x00	50 – 24000 rpm *50 – 24000*	1000 rpm	Defines the maximum reverse speed of the motor (in motor rpm) when emergency reverse is active.
EMR Accel Rate *EMR_Accel_Rate* 0x3492 0x00	0.1 – 3.0 s *100 – 3000*	0.1 sec	Sets the rate (in seconds/Typical Max Speed) at which the vehicle accelerates in the opposite direction after being brought to a stop. If the vehicle is already traveling in the reverse direction below the EMR Speed, the EMR Accel Rate will bring the vehicle to the EMR Speed.
EMR Decel Rate *EMR_Decel_Rate* 0x3493 0x00	0.1 – 3.0 s *100 – 3000*	0.1 sec	Sets the rate (in seconds/Typical Max Speed) at which the vehicle brakes to a stop when activating the emergency reverse while the vehicle is moving forward. If the vehicle is already traveling in the reverse direction above the EMR Speed, the EMR Decel Rate will bring the vehicle down to the EMR Speed.

APPLICATION SETUP/EMERGENCY REVERSE (EMR) — EMR SUPERVISION MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Supervision Enable *EMR_Supervision_Enable* `0x34A4 0x00`	On/Off *On/Off*	On	Enables or disables the EMR supervision function. On = EMR supervision is enabled. Off = EMR supervision is disabled. To be compliant with the EN 13849 ratings, enable this parameter.
Tolerance *EMR_Supervision_Tolerance* `0x34A3 0x00`	0 – 24000 rpm *0 – 24000*	500 rpm	The tolerance window defined by the entry speed plus this speed, ramp rate, and zero speed. Use this to mitigate nuisance faults. Detection is due to application of the EMR switch or the VCL function, *Enable_Emer_Rev( )*. See in Figure 24, Initial Speed * (1 + Tolerance).
Ramp Delay *EMR_Supervision_Ramp_Delay* `0x349D 0x00`	0 – 1000 ms *0 – 1000*	500 ms	Determines how long from the time EMR = On before the speed needs to move toward zero for EMR supervision. See in Figure 24, Ramp Delay.
Ramp Rate *EMR_Supervision_Ramp_Rate* `0x349E 0x00`	0.1 – 30.0 s *100 – 30000*	5.0 sec	Determines the slowest ramp rate allowed during EMR direction reversal without an EMR supervision fault. This ramp rate is the number of seconds to transition from typical max speed to zero.

Figure 24

Emergency Reverse Supervision Function



The EMR Supervision Limit curve, as shown by the red line in Figure 24, is framed by the initial speed, the parameter *EMR_Supervision_Tolerance*, the parameter *EMR_Supervision_Ramp_Delay*, and the parameter *EMR_Supervision_Ramp_Rate*. The blue line represents a normal EMR operation. If the motor speed exceeds the red curve for more than 64 ms continuously, an EMR Supervision fault is declared. See the Troubleshoot Chart, fault code 9-11.

APPLICATION SETUP — INTERLOCK BRAKING MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Interlock Brake Enable *Interlock_Brake_Enable* `0x30F6 0x00`	On/Off *On/Off*	On	Determines whether the interlock braking function is active. On = The controller will attempt to bring the vehicle to a stop using regen braking when the interlock signal is removed. Off = The controller will disable the bridge after the Sequencing Delay expires and allow the vehicle to roll freely when the interlock signal is removed. Typically, use this option only when there is an operator/driver controlled mechanical or hydraulic brake system.
Decel Rate HS *Interlock_Brake_Decel_Rate_HS* `0x30F4 0x00`	0.1 – 30.0 s *100 – 30000*	1.5 sec	Sets the rate (in seconds/Typical Max Speed) to slow down the vehicle when the interlock is released at high vehicle speeds. Larger values represent slower response. See the *Speed Mode/Response/Fine Turning/HS* parameter.
Decel Rate LS *Interlock_Brake_Decel_Rate_LS* `0x30F5 0x00`	0.1 – 30.0 s *100 – 30000*	2.0 sec	Sets the rate (in seconds/Typical Max Speed) to slow down the vehicle when the interlock is released at low vehicle speeds. Larger values represent slower response. See the *Speed Mode/Response/Fine Turning/LS* parameter.
Interlock Brake Settling Time *Interlock_Brake_Settling_Time* `0x321D 0x00`	0.0 – 8.0 s *0 – 8000*	0.5 sec	Time after *Zero_Speed_Threshold* is reached during interlock braking that the main contactor remains closed and the bridge remains enabled, allowing position hold to settle.
Interlock Brake Timeout *Interlock_Brake_Timeout* `0x30F7 0x00`	0.0 – 500.0 s *0 – 62500*	5.0 sec	Controls the maximum allowable duration of an interlock braking event. The timer starts as soon as the interlock signal is removed (i.e., the interlock switch is opened). If the time expires before the vehicle has slowed below the *Zero_Speed_Threshold* the EM brake will automatically engage. This timeout allows parallel usage of regen braking and the EM brake to reduce stopping distance. If the Interlock Brake Timeout expires and the motor is still moving, regen braking will continue to retard vehicle motion in conjunction with the EM brake. Note: This parameter is only applicable when the *EM_Brake_Type* = 1 or 2.
Supervision Enable *Interlock_Brake_Supervision_Enable* `0x311F 0x00`	On/Off *On/Off*	On	Enables or Disables the *interlock brake supervision* function. On = Supervision is enabled. Off = Supervision is disabled. Note, to comply with EN 13849, set this parameter to On (enabled).
Interlock Anti Tiedown *Interlock_Antitiedown_Time* `0x32D6 0x00`	0 – 1000 ms *0 – 1000*	0 ms	The *Interlock_Antitiedown_Time* is the time after KSI voltage is applied that interlock has to be held low (Off) to prove it is not shorted. To disable this functionality, set this Interlock Anti Tiedown parameter to 0 ms.

APPLICATION SETUP/INTERLOCK BRAKING — INTERLOCK BRAKING SUPERVISION MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Tolerance *Interlock_Brake_Supervision_Tolerance* `0x311E 0x00`	0 – 24000 rpm *0 – 24000*	500 rpm	The motor speed must exceed this value plus the window defined by the entry speed, ramp rate, and zero speed. Use this parameter to mitigate nuisance faults. See Figures 25 and 26, Initial Speed * (1 + Tolerance).
Ramp Delay *Interlock_Brake_Supervision_Ramp_Delay* `0x3118 0x00`	0 – 1000 ms *0 – 1000*	500 ms	Determines how long from the time Interlock = Off before the speed needs to move toward zero for interlock brake supervision. See Figures 25 and 26, Ramp Delay.
Ramp Rate *Interlock_Brake_Supervision_Ramp_Rate* `0x3119 0x00`	0.1 – 30.0 s *100 – 30000*	5.0 Sec	Determines slowest ramp rate allowed during interlock braking without an Interlock Braking Supervision fault. This ramp rate is the number of seconds to transition from the typical max speed to zero.
Position Settling Limit *Interlock_Brake_Supervision_Position_Settling_Limit* `0x3121 0x00`	0.1 – 20.0 *410 – 81920*	10.0	Determines the farthest distance (in motor revolutions) the vehicle can move after the first zero speed crossing after interlock without an Interlock Braking Supervision fault. This limit should be set based on the worst-case rollback allowed. See Figure 27.

Quick Links:
Fig. 26 p.87
Fig. 27 p.87

Figure 25

Interlock Brake Supervision Speed Limit for Positive Initial Speed

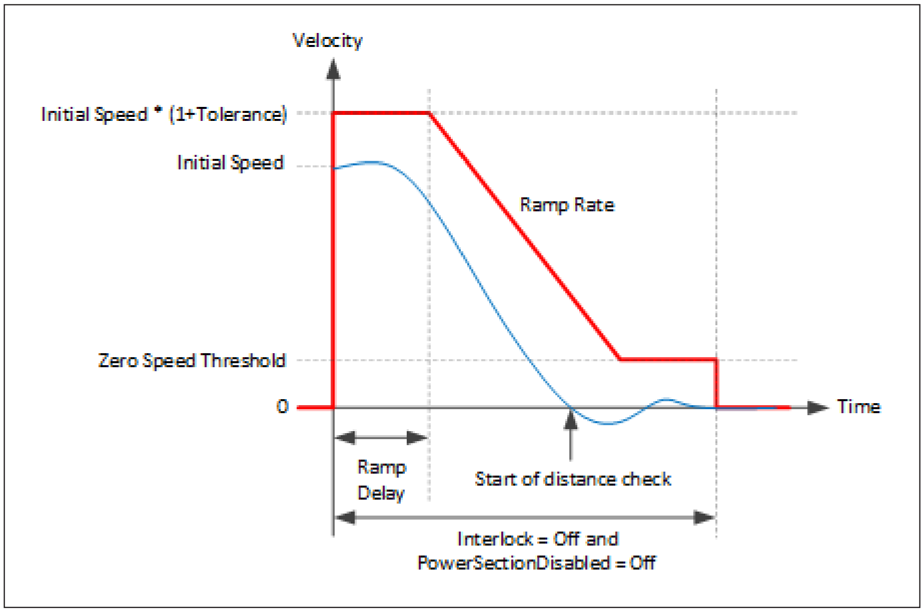


Figure 26
Interlock Brake Supervision Speed Limit for Negative Initial Speed

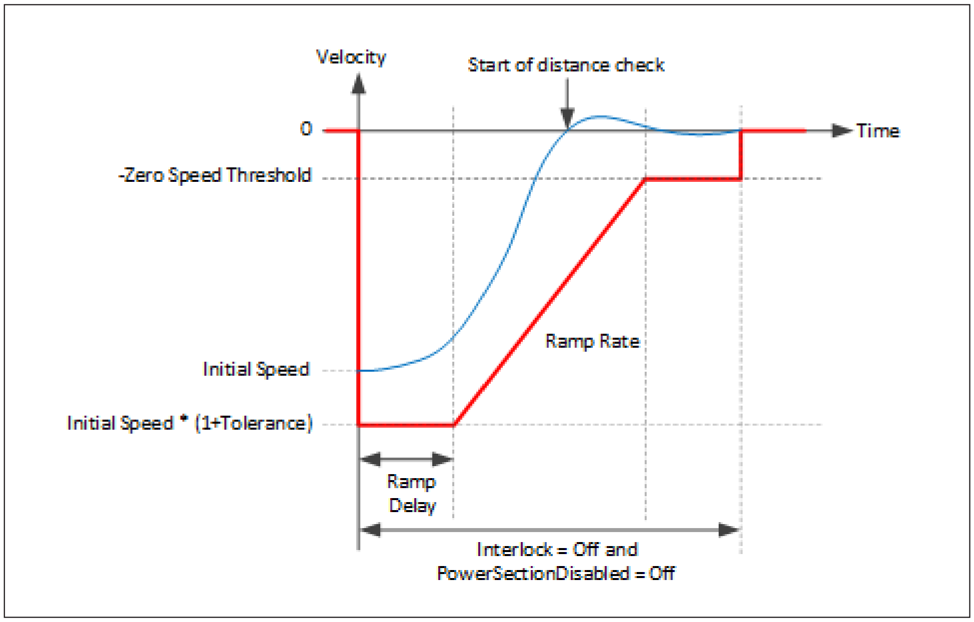
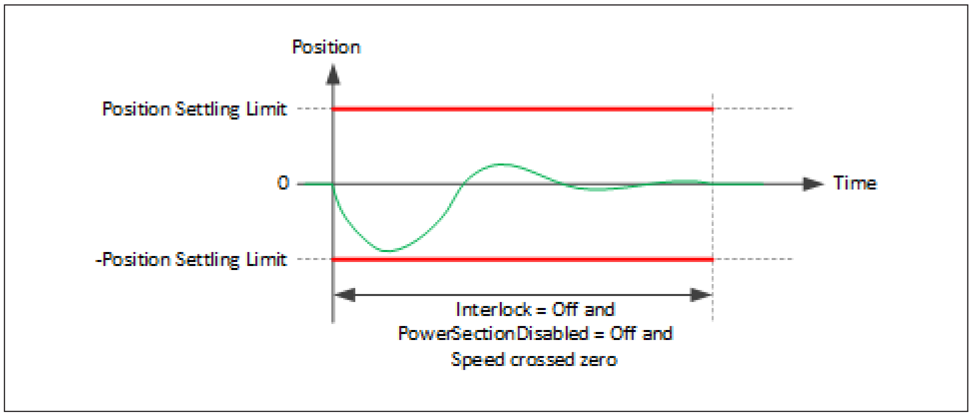


Figure 27
Interlock Brake Supervision Distance Check



APPLICATION SETUP/HYDRAULICS — LIFT SETTINGS MENU

The parameters in this menu, for this non-combi controller, do not offer "throttling" of the contactor-driven DC pump motor. These parameters are applicable to alternating the response to a 0-100% VCL input, an analog voltage input, and less so for a typical switch input.

Hydraulic Pump Motor, Load-Hold and Proportional Valves

See the Controller Setup/IO Assignments/Controls menu.

See a pump-combi controller manual, e.g., 53228_AC F2-C_RevA (or newer).

HYDRAULIC OPERATION

This controller does not control the speed of the DC motor driving the hydraulic pump. It closes a contactor via an assigned driver, which applies full voltage to a DC pump motor in the traditional method. This controller does control a proportional lowering valve, and, if used, a load hold valve. Figures 28 and 29 illustrate two typical hydraulic systems and their components for the Lift and Lower operations. For other hydraulic operations (e.g., reach, tilt, side-shift, rotate), the vehicle manufacturer will provide those hydraulic path(s), whose setup/operation follow similar methods. To control these other hydraulic valves, use VCL programming and the available (spare) coil-drivers, switch and analog inputs and the available outputs. Use the figures and parameters to match the application. If throttling a pump motor is required, select the appropriate combination traction and pump "combi-controller" … for example, the AC F2-C controller.

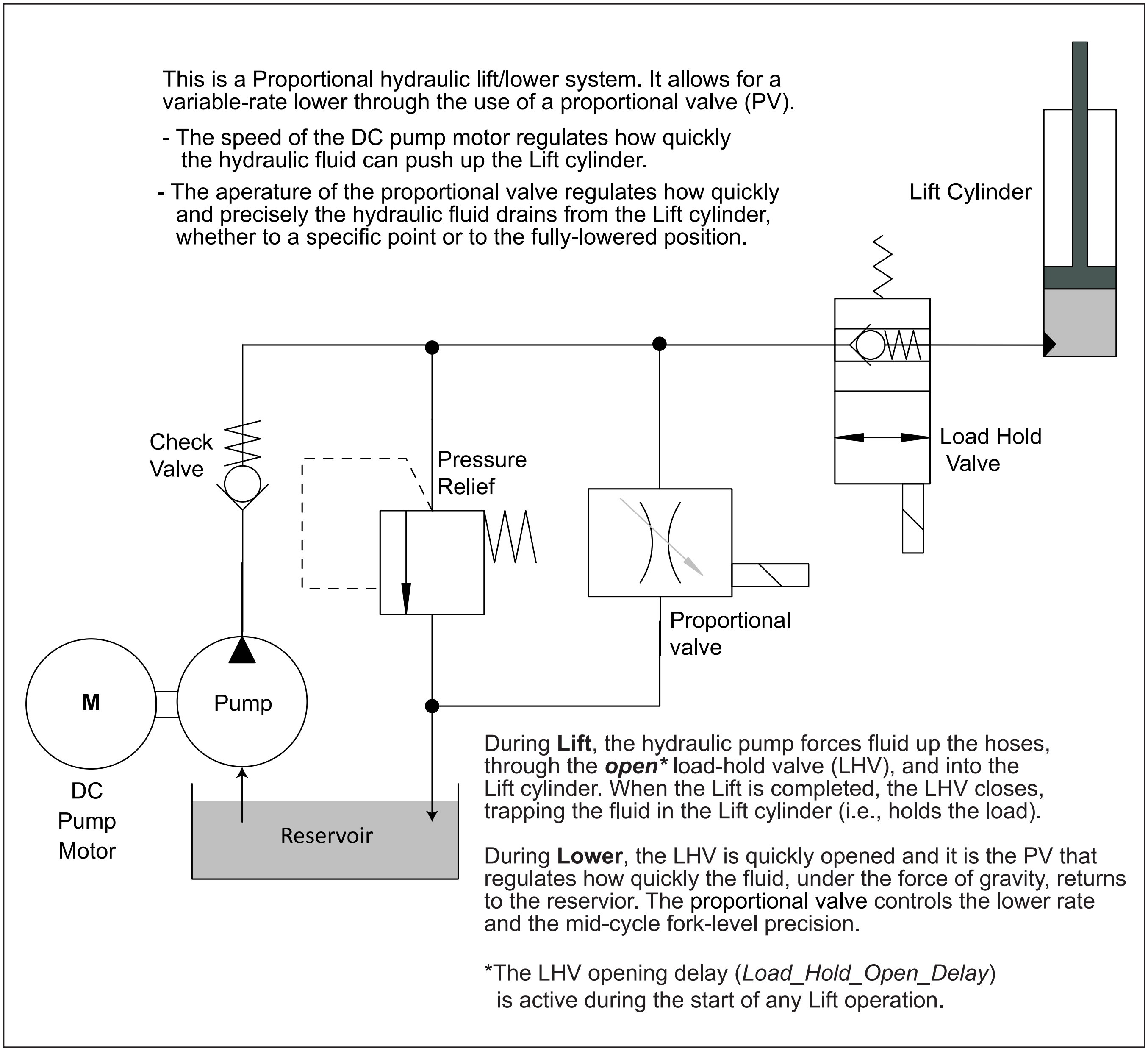


Figure 28

Hydraulic system diagram, with load-hold and proportional lower valves

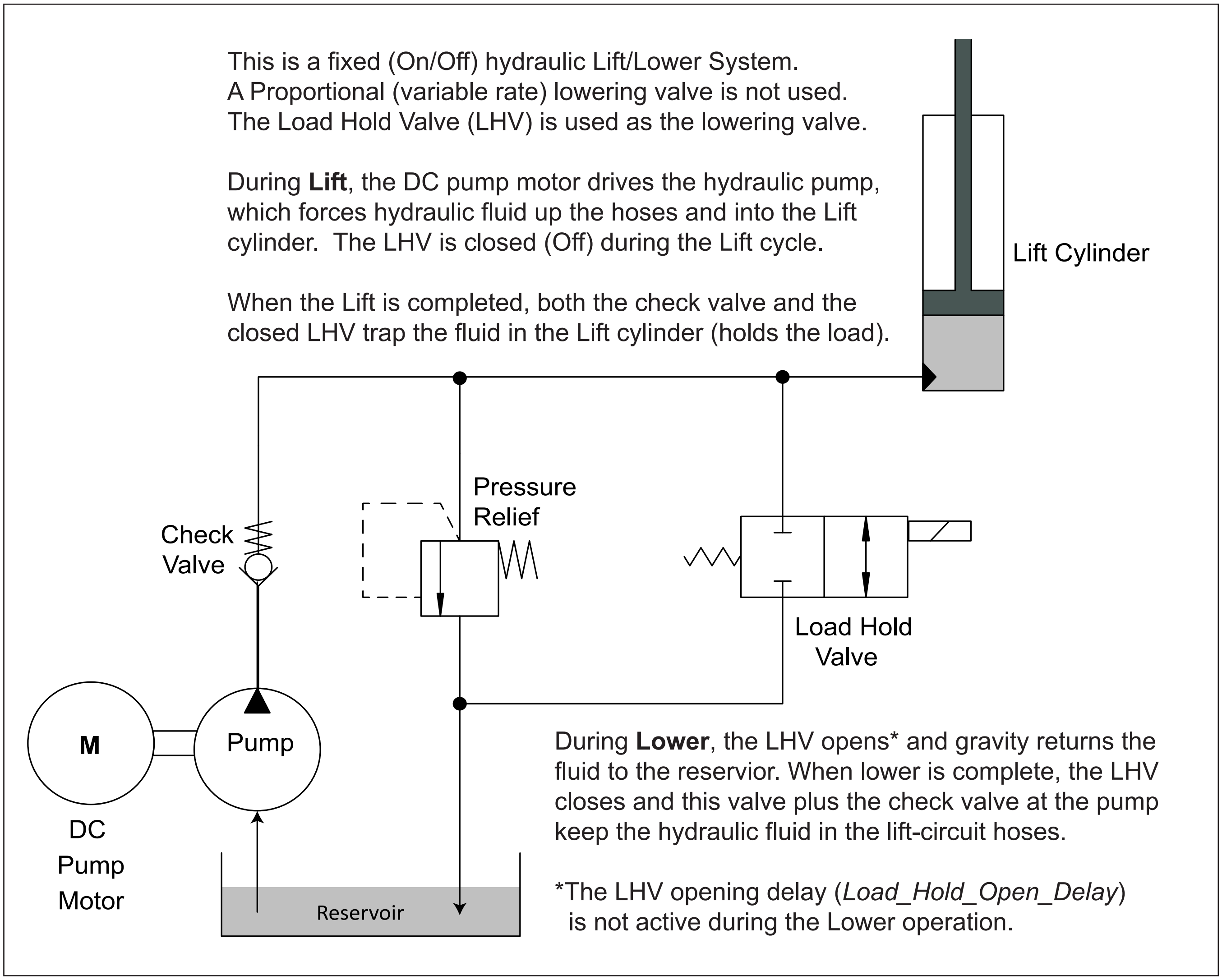


Figure 29
Hydraulic system diagram, with a fixed (On/Off) load-hold valve

Quick Links:
Fig. 14 p.18

The hydraulics parameters adjust the system's operating characteristics — knowing that for a non-combi controller, the lift settings apply to a minimal extent when setting the "trigger" points of the type of lift command input. A "switch to KSI voltage" is Off/On. A VCL command of 0-100% is similar, as is a voltage input into an analog input. Use the parameters to tailor the hydraulic system to a specific application, or to a specific operator's preferences, in conjunction with the parameters in the Controls menu (*Controller Setup » IO Assignments » Controls*).

Figures 30 and 31 show the signal-chain process for the Lift and Lower functions. The inputs applicable to a combi-controller are in light/greyed-out text. These are not directly applicable to the standard motor controllers. If a simple "switch" triggers the pump, notice its 0-100% step-response in the throttle/switch mapping box. The CAN tiller system (Figure 14) will use a processed VCL variable to drive the pump motor, skipping the throttle-mapping block and utilizing the *VCL_Lift_Throttle_Enable* (On/Off) and *VCL_Lift_Throttle* (%) functions. Use these Lift and Lower process figures as a guide to the appropriate hydraulics parameters, VCL functions, and the parameters in the Controls menu (*Controller Setup » IO Assignments » Controls*). Hint: Review how the traction throttle and brake throttle signal chains operate.

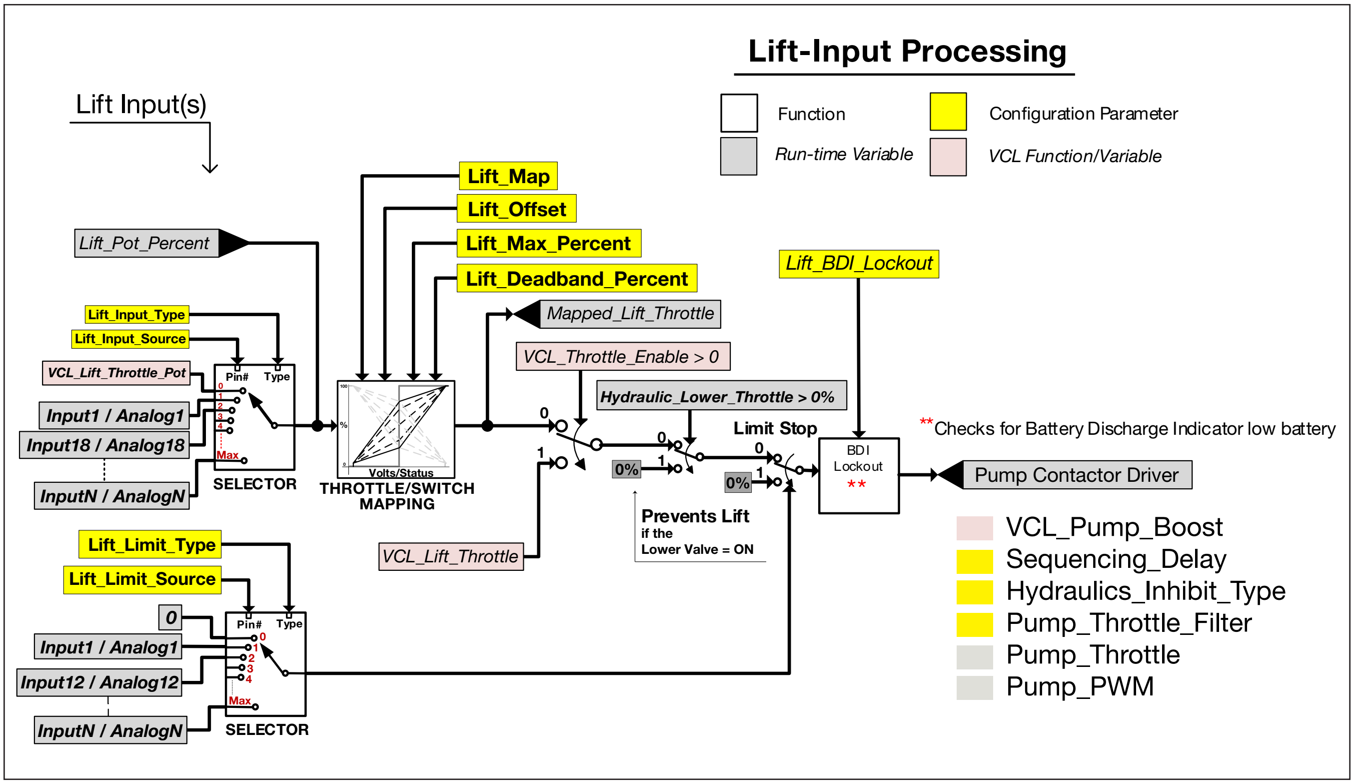


Figure 30

Lift-input signal chain (processing)

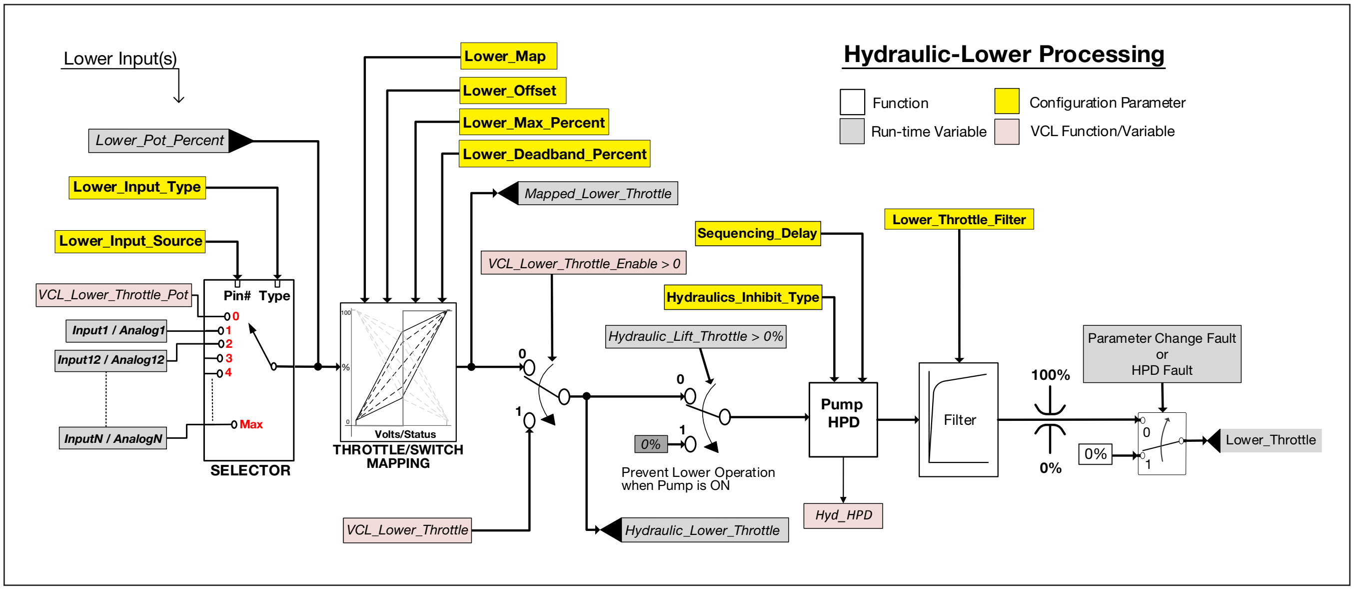


Figure 31

Lift-lowering signal chain (processing)

APPLICATION SETUP/HYDRAULICS — LIFT SETTINGS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Lift Input *Lift_Pot_Percent* `0x3045 0x00`	*0.0 – 100.0 %* *0 – 1000*	Read Only	Lift pot input after source selection and before mapping as a percentage.
Mapped Lift Throttle *Mapped_Lift_Throttle* `0x4FD8 0x00`	0.0 – 100.0 % *0 – 32767*	Read Only	Hydraulics lift throttle after mapping.
Lift Command *Hydraulic_Lift_Throttle* `0x3C25 0x00`	0.0 – 100.0 % *0 – 32767*	Read Only	Pump demand after input processing.
Lift Min Input *Lift_Deadband_Percent* `0x3708 0x00`	0 – 100 % *0 – 1000*	15 %	Input voltage percentage below which throttle is made 0.
Lift Max Input *Lift_Max_Percent* `0x370B 0x00`	0 – 100 % *0 – 1000*	85 %	Input value corresponding to 100% throttle.
Lift Map Shape *Lift_Map* `0x370A 0x00`	0 – 100 % *0 – 32767*	35 %	Defines sensitivity of input. Lower value results in lesser output variation to input. Set at 50% for linear response.
Lift Offset *Lift_Offset* `0x370C 0x00`	0 – 100 % *0 – 32767*	0 %	Minimum lift throttle command after input becomes greater than *Lift_Deadband_Percent parameter.*
Lift Limit Switch Source *Lift_Limit_Source* `0x3C3E 0x00`	0 – 35 *0 – 35*	0	The switch input name (number) to which the limit switch is applied.
Lift Limit Switch Type *Lift_Limit_Type* `0x3C3F 0x00`	0 – 3 *0 – 3*	0	Describes the limit switch type: 0 - Normally Open (NO) Switch 1 - Normally Closed (NC) Switch 2 - Voltage Input (Turns off the pump if input > 50%)
Lift Battery Lockout *Lift_BDI_Lockout* `0x3707 0x00`	0 – 100 % *0 – 100*	0 %	Battery state of charge (BDI) threshold below which pump turn-on is disabled.

APPLICATION SETUP/HYDRAULICS — LOWER SETTINGS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Lower Input *Lower_Pot_Percent* `0x3044 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	Lower pot input after source selection and before mapping as a percentage.
Mapped Lower *Mapped_Lower_Throttle* `0x4FD9 0x00`	0.0 – 100.0 % *0 – 32767*	Read Only	Hydraulics lower throttle after mapping.
Lower Command *Lower_Throttle* `0x3725 0x00`	0.0 – 100.0 % *0 – 32767*	Read Only	Proportional driver current request.
Lower Min Input *Lower_Deadband_Percent* `0x3711 0x00`	0 – 100 % *0 – 1000*	5 %	Input voltage percentage below which throttle is made 0.
Lower Max Input *Lower_Max_Percent* `0x3714 0x00`	0 – 100 % *0 – 1000*	95 %	Input value corresponding to 100% throttle.
Lower Map Shape *Lower_Map* `0x3713 0x00`	0 – 100 % *0 – 32767*	35 %	Defines sensitivity of input. Lower value results in lesser output variation to input. Set at 50% for linear response.
Lower Offset *Lower_Offset* `0x3715 0x00`	0 – 100 % *0 – 32767*	0 %	Minimum lower throttle command after input becomes greater than the *Lower_Deadband_Percent* parameter.
Lower Filter *Lower_Throttle_Filter* `0x3C3D 0x00`	0.1 – 100.0 Hz *1 – 1000*	10.0 Hz	Sets low pass filter cutoff frequency for lowering proportional valve. Higher values make it more responsive, lesser values make it less responsive to input variations.

APPLICATION SETUP/HYDRAULICS — LOAD HOLD VALVE SETTINGS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Load Hold Valve Driver *Load_Hold_Driver* `0x3C41 0x00`	0 – 5 *0 – 5*	0 **[PCF]**	Select the PWM Driver to which the load hold valve is connected. Changes require cycling the keyswitch to take effect. The default value, 0, does <u>not</u> select a driver. *Shown as Driver 3 in Figure 13, Basic Wiring Diagram*
Load Hold Valve Enable On Lift *Load_Hold_Enable_On_Lift* `0x37E0 0x00`	Off/On *0 – 1*	Off	0 - The load hold valve is disabled on lift. Specify 0 if the application does not use a load-hold valve. 1 - The load hold is enabled on lift. Specify 1 if the application uses a load-hold valve. Note: Set *Load_Hold_Open_Delay* to configure the delay time.
Load Hold Opening Delay *Load_Hold_Open_Delay* `0x370E 0x00`	0 – 2000 ms *0 – 2000*	0 ms	This parameter delays opening of the load-hold valve when the hydraulic pump is turned on (i.e., the lift command). 0 ms: The load-hold valve is continually kept closed (i.e., the load-hold driver is off) when the DC pump motor is turned on. 8-2000 ms: The duration of time before opening the load hold valve after the DC pump motor is turned on.
Load Hold Valve Driver Voltage *Load_Hold_Driver_Voltage* `0x3005 0x00`	0 – 100 % *0 – 32767*	0 %	Defines the battery-compensated commanded PWM of Load Hold Valve Driver. Higher values will open the valve faster/firmer (similar to driving a contactor coil close when applying 100%).

DUAL DRIVE

See the Dual Drive Supplement manual: 53231_F-Series Dual Drive (FOS 4.2).

APPLICATION SETUP — VEHICLE MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Metric Units *Metric_Units* `0x37C9 0x00`	Off / On *0 – 1*	Off (0)	When this parameter is On, the distance variables (Vehicle Odometer, Braking Distance Captured, Distance Since Stop, Distance Fine, and the Capture Distance variables) will accumulate and display in metric units (km, meters, or decimeters). When programmed Off, the distance variables will accumulate and display in English units (miles, feet, or inches).
Speed to RPM *Speed_to_RPM* `0x37CE 0x00`	10.0 – 3000.0 *100 – 30000*	253.0	This parameter affects the vehicle speed displayed in the Monitor » Motor menu, and modifies the VCL variable *Vehicle_Speed*; it does not affect actual vehicle performance. The value entered for Speed to RPM is a conversion factor that scales motor speed to vehicle speed. KPH to RPM: (G/d)*5305, where G = gear ratio, d = tire diameter [mm]. MPH to RPM: (G/d)*336.1, where G = gear ratio, d = tire diameter [in]. Hint: Use the vehicle tire's rolling diameter for "d" based upon a rollout measurement. Roll the vehicle a known number of wheel revolutions and measure the distance. Circumference (C) = Distance/wheel-revolutions. Diameter (d) = C/Π. Π (pi)= 3.14159. Then use the **Distance Since Stop** monitor variable to cover a known distance (e.g., 50 – 100 feet) to align the value to the measured distance while adjusting the Speed to RPM parameter. See the Distance Since Stopped variable (below).
Vehicle Speed *Vehicle_Speed* `0x37DD 0x00`	–3276.8 – 3276.7 *–32768 – 32767*	Read Only MPH or KPH	Vehicle speed in units of MPH or KPH depending on the setting of the Metric Units parameter.
Vehicle Odometer *Vehicle_Odometer* `0x37DA 0x00`	0.0 – 10000000.0 *0 – 100000000*	Read Only Mile or Km	Vehicle distance traveled, in units of miles or km, depending on the setting of the Metric Units parameter in the Vehicle parameters menu. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Trip Odometer 1 *Trip_Odometer_1* `0x37D7 0x00`	0.0 – 10000000.0 *0 – 100000000*	Read Only Mile or Km	Vehicle distance traveled, in units of miles or km, depending on the setting of the Metric Units parameter in the Vehicle parameters menu. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Trip Odometer 2 *Trip_Odometer_2* `0x37D8 0x00`	0.0 – 10000000.0 *0 – 100000000*	Read Only Mile or Km	A second vehicle distance traveled variable, and works identically to Trip Odometer 1.
Reset Trip Odometer 1 *Reset_Trip_Odometer_1* `0x37CB 0x00`	0 – 1 *0 – 1*	0	Resets the first trip odometer.
Reset Trip Odometer 2 *Reset_Trip_Odometer_2* `0x37CC 0x00`	0 – 1 *0 – 1*	0	Resets the second trip odometer.

APPLICATION SETUP/VEHICLE — PERFORMANCE METRICS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Capture Speed 1 *Capture_Speed_1* `0x37C4 0x00`	0 – 24000 rpm *0 – 24000*	4500	The controller captures the time it takes the motor to go from 0 rpm to the **programmed** Capture Speed. This timer starts every time the motor accelerates from zero speed. The result is stored as "Time to Speed 1" in the *Programmer System Monitor » Vehicle* menu.
Capture Speed 2 *Capture_Speed_2* `0x37C5 0x00`	0 – 24000 rpm *0 – 24000*	4500	This parameter allows a second capture speed to be defined, and works identically to Capture Speed 1.The result is stored as "Time to Speed 2" in the *Programmer System Monitor » Vehicle* menu.
Capture Distance 1 *Capture_Distance_1* `0x37C1 0x00`	1 – 1320 *1 – 1320*	22	The controller captures the time it takes the vehicle to travel from 0 rpm to the programmed Capture Distance. The result is stored as "Time to Dist 1" in the *Programmer System Monitor » Vehicle* menu. This timer starts every time the vehicle accelerates from zero speed. Note: For accurate distance measuring, the Speed to RPM parameter must be set correctly. With the Metric Units parameter programmed Off, distance is in units of feet. With Metric Units programmed On, distance is in units of meters.
Capture Distance 2 *Capture_Distance_2* `0x37C2 0x00`	1 – 1320 *1 – 1320*	100	This parameter allows a second capture distance to be defined, and works identically to Capture Distance 1. The result is stored as "Time to Dist 2" in the *Programmer System Monitor » Vehicle* menu.
Capture Distance 3 *Capture_Distance_3* `0x37C3 0x00`	1 – 1320 *1 – 1320*	150	This parameter allows a third capture distance to be defined, and works identically to Capture Distance 1. The result is stored as "Time to Dist 3" in the *Programmer System Monitor » Vehicle* menu.
Vehicle Acceleration *Vehicle_Acceleration* `0x37D9 0x00`	0.000 – 10.000 g *0 – 10000*	Read Only	This is a calculated g-force value for the vehicle's acceleration. For an accurate measurement, correctly set the Speed to RPM parameter.
Time to Speed 1 *Time_to_Capture_Speed_1* `0x37D3 0x00`	0.00 – 128.00 s *0 – 32000*	Read Only	Time taken for the vehicle to go from zero rpm to the programmed Capture Speed 1 during its most recent such acceleration.
Time to Speed 2 *Time_to_Capture_Speed_2* `0x37D4 0x00`	0.00 – 128.00 s *0 – 32000*	Read Only	Time taken for the vehicle to go from zero rpm to the programmed Capture Speed 2 during its most recent such acceleration.
Time Between Speeds *Time_Between_Capture_Speeds* `0x37CF 0x00`	0.00 – 128.00 s *0 – 32000*	Read Only	Time taken for the vehicle to go from programmed Capture Speed 1 to programmed Capture Speed 2 during its most recent such acceleration.
Time to Dist 1 *Time_to_Capture_Distance_1* `0x37D0 0x00`	0.00 – 128.00 s *0 – 32000*	Read Only	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 1 during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Time to Dist 2 *Time_to_Capture_Distance_2* `0x37D1 0x00`	0.00 – 128.00 s *0 – 32000*	Read Only	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 2 during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly.

APPLICATION SETUP/VEHICLE — PERFORMANCE METRICS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Time to Dist 3 *Time_to_Capture_Distance_3* `0x37D2 0x00`	0.00 – 128.00 s *0 – 32000*	Read Only	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 3 during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Braking Distance Captured *Braking_Distance_Captured* `0x37C0 0x00`	0.0 – 1000000.0 *0 – 40000000*	Read Only Feet or Meters	Distance traveled by the vehicle starting with vehicle braking (initiated by throttle reversal, VCL_Brake, or interlock braking) and ending when *Motor_RPM* = 0. Units are meters or feet, depending on the setting of the Metric Units parameter. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Distance Since Stop *Distance_Since_Stop* `0x37C8 0x00`	0.0 – 1000000.0 *0 – 40000000*	Read Only Feet or Meters	Distance traveled by the vehicle starting from a stop. In effect, it uses the vehicle as a tape measure. (In other words, if you travel 300 feet forward and then 300 feet in reverse, the distance would be 600.) The distance is continuously updated and will stop (and restart) when *Motor_RPM* = 0. For accurate distance measurements, the Speed to RPM parameter must be set correctly. Units are meters or feet, depending on the setting of the Metric Units parameter. Usage examples: Set the Speed to RPM parameter by using this variable to cover a known distance (e.g., 50 feet). Adjust the Speed to RPM parameter until this value matches the known distance.
Distance Fine *Distance_Fine_Long* `0x37C7 0x00`	0.0 – 1000000.0 *0 – 40000000*	Read Only Inches or Decimeters	Position measurement. Net distance in both the forward and reverse directions. (In other words, if you travel 20 inches forward and then 20 inches in reverse, the distance would be zero.) The distance is continuously updated and will roll over when the variable goes over the limits. Resets to zero on key cycle. Units are decimeters or inches, depending on the setting of the Metric Units parameter. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Reset Distance Fine *Reset_Distance_Fine* `0x37CA 0x00`	0 – 1 *0 – 1*	0	Resets the Distance Fine value.

APPLICATION SETUP — MAX SPEED SUPERVISION MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Present Max Speed Limit *Current_Max_Speed_Limit* `0x37AB 0x00`	–2147483648 – 2147483647 *–2147483648 – 2147483647*	Read Only rpm	A monitor variable, in rpm, that shows the present (real time) Max Speed Limit accounting for the Max Speed Limit percent and Max Speed Limit Slew Rate time parameters (below). This variable is useful for setting up the Max Speed Supervision parameters.
Max Speed Limit Timer *Max_Speed_Limit_Timer* `0x37AC 0x00`	–2147484.00 – 2147484.00 *–2147483648 – 2147483647*	Read Only Seconds	The timer associated with the Max Speed Time Limit parameter (below). e.g., 0.01 – 10.0s This variable is useful for setting up the Max Speed Supervision parameters.
Max Speed Limit *Max_Speed_Limit* `0x37A7 0x00`	0.0 – 500.0 % *0 – 5000*	25.0 %	The Max Speed Supervision parameters detect conditions where the speed limiting function is no longer following the maximum speed limit of the vehicle. This parameter defines the percent over the programmed Max Speed parameter before the Speed Limit Supervision fault is checked and contributes to set the Speed Limit Supervision fault. This set of Max Speed Supervision parameters operates independently from the normal motor control to ensure that motor speed remains at a safe level for typically multi-mode (e.g., elevated load travel) speeds. Note: Motor rpm is the basis of vehicle speed.
Max Speed Limit Slew Rate *Max_Speed_Limit_Slew_Rate* `0x37A8 0x00`	0.1 – 60.0 s *100 – 6000*	20.0 Sec	This parameter defines a deceleration rate for the current Max Speed Limit. The rate definition is the *Typical_Max_Speed (RPM)/ Max_Speed_Limit_Slew_Rate*. In the event that the Max Speed decreases, the Max Speed Limit will slew towards the new target limit at this rate. This parameter should be set significantly longer than the *Max_Speed_Decel_ SpdM* parameter to avoid false trips when adjusting accel and decel rates. Conversely, if the Max Speed increases, the limit will immediately step to the new target to avoid unnecessary complications with *Max_Speed_Accel_SpdM* parameter. Usage example: When the Max Speed Limit is changed in a multi-mode event (e.g., reduced speed with elevated forks or load), slows the vehicle in a defined decel rate.
Max Speed Time Limit *Max_Speed_Time_Limit* `0x37A9 0x00`	0.1 – 10.0 s *100 – 10000*	5.0 Sec	Controls the maximum time of the up/down counter for max speed supervision. If the Max Speed Limit is exceeded, the timer counts up, otherwise it counts down. If the timer reaches this parameter's time (setting), the Speed Limit Supervision fault is set. (See the Max Speed Limit Timer and the Max Speed Limit parameters, above.)

APPLICATION SETUP — MOTOR NOT STOPPED MENU

These parameters affect the **Motor Not Stopped** fault, which is a safety function implemented on a Category 2 architecture per ISO 13849. The purpose of this function is to detect hazardous movement when the AC motor is stopped and expected to stay stopped (i.e., no throttle command). There are three main checks done when the motor is in the stopped state, each of which can be independently enabled and each of which has a unique fault type as described.

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Motor Not Stopped State Time *Motor_Not_Stopped_State_Time* `0x37AD 0x00`	0.0 – 100.0 s *0 – 100000*	10.0 sec	The time when the vehicle is at zero speed (motor rpm = 0) before the controller enters the Stopped State and begins checking for motor movement, or the Stopped State exit commands and conditions. This feature detects the motor moving when it is supposed to be stopped.
Motor Not Stopped Max Frequency *Motor_Not_Stopped_Max_Frequency* `0x37AE 0x00`	1.0 – 40.0 Hz *60 – 2400*	40.0 Hz	The controller's maximum motor-control electrical frequency when in the Stopped State before the Motor Not Stopped (Type 3) fault is triggered. Setting this value to 0 will disable this fault type.
Motor Not Stopped Distance Error *Motor_Not_Stopped_Distance_Error* `0x37B2 0x00`	–4000000.0 – 4000000.0 *–4000000.0 – 4000000.0*	0.0*	The maximum distance the vehicle can move while in the Stopped State before the Motor Not Stopped (Type 1) fault is triggered. Setting this value to 0 will disable this fault type. This is most useful for traction drives. * This distance is either in inches or decimeters ($^1/_{10}$ meter), depending on the *Metric_Units* (0x37C9) parameter setting.
Motor Not Stopped Speed Error *Motor_Not_Stopped_Speed_Error* `0x37B3 0x00`	0 – 10000 rpm *0 – 10000*	0 rpm	The maximum speed of the motor while in the Stopped State before the Motor Not Stopped (Type 2) fault is triggered. Setting this value to 0 will disable this fault type. This is most useful for hydraulic pump drives.
Motor Not Stopped Max Current *Motor_Not_Stopped_Max_Current* `0x37AF 0x00`	0.0 – 100.0A *0 – 1000*	0.0A	The controller's maximum motor-phase current (Irms) when in the Stopped State before the Motor Not Stopped (Type 3) fault is triggered. Setting this value to 0 will disable this fault type.

APPLICATION SETUP — HAZARDOUS MOVEMENT MENU

These parameters affect the **Hazardous Movement** fault, which detects such movements when the motor is requested to rotate.

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Hazardous Direction Response Time *Hazardous_Direction_Response_Time* `0x3E8D 0x00`	0 – 10000 ms *0 – 10000*	0 ms	The (up/down) debounce timer on the Hazardous Movement fault, Type 1. Setting this value to 0 will disable this fault type.
Hazardous Throttle Response Time *Hazardous_Throttle_Response_Time* `0x3E90 0x00`	0 – 10000 ms *0 – 10000*	0 ms	The acceleration threshold for the Hazardous Movement fault, Type 2. Setting this value to 0 will disable this fault type.
Hazardous Speed Error *Hazardous_Speed_Error* `0x3E8F 0x00`	0 – 20000 rpm *0 – 20000*	15000 rpm	The speed threshold for the Hazardous Movement fault, Type 2.
Hazardous Accel *Hazardous_Accel* `0x3E8E 0x00`	0 – 10000 rpm/s *0 – 10000*	5000 rpm/s	The acceleration threshold for the Hazardous Movement fault, Type 2.

APPLICATION SETUP — MOTOR BRAKING SUPERVISION MENU

These parameters affect the **Motor Braking Impaired** fault, which detects when motor braking was impaired beyond a safe threshold.

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Overall Cutback *OverallCutback* `0x32D9 0X00`	0.0 – 100.0 % *0 – 4096*	Read Only	The accumulated controller cutback.
Motor Braking Impaired Threshold *Motor_Braking_Impaired_Threshold* `0x32DA 0x00`	0.0 – 100.0 % *0 – 4096*	0.0 %	The threshold for OverallCutback below which a Motor Braking Impaired fault will be generated after the specified time (duration).
Motor Braking Impaired Timer *Motor_Braking_Impaired_Timer* `0x32D8 0x00`	0 – 32767 ms *0 – 32767*	Read Only	This timer counts down when the OverallCutback is below the parameterized threshold, if it reaches 0 a Motor Braking Impaired fault will be generated.
Motor Braking Impaired Time *Motor_Braking_Impaired_Time* `0x32DB 0x00`	0 – 30000 ms *0 – 30000*	64 ms	The time the OverallCutback must be below the threshold for a Motor Braking Impaired fault to be generated.

APPLICATION SETUP — IMU MENU

The controllers with part number -1xx are available with an Inertial Measurement Unit (IMU) for improved safety, better drivability and other fault detection. The acceleration and rational speeds are reported as VCL variables with the following descriptions.

IMU Parameter	Description
Acceleration	Axes: 3 Range: ±16G Sample Rate: >200Hz Resolution: ≤0.5 mG/count
Rotational Speed	Axes: 3 Range: ±2000°/sec Sample Rate: >200Hz Resolution: ≤70mdeg/sec/count
Temperature Range	–40°C to +105°C

The listed (below) IMU functionality is provided without need for additional VCL:

- Configuration of controller reference frame to that of vehicle's to account for varied installation orientations[1].
- VCL accessible variables for vehicle acceleration, turn-rates(yaw, etc), and angles(roll, pitch).
- Improved grade estimate for hill-hold through estimating better holding torque.
- Improved safety through improved hill-hold, tip-over protection, detection of unintended accel/deceleration and prevention of motor run-aways due to encoder position loss.
- Improved estimation of speed/motor position at low speeds, low-count encoders or to provide a better encoder LOS mode.

---

[1] Allows controller to report/determine vehicle's roll, pitch, acceleration, jerk, heading for any installation orientation.

APPLICATION SETUP — IMU OUTPUT MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
IMU Ready *IMU_Ready* `0x3F82 0x00`	0 – 1 *0 – 1*	0 Read Only	This parameter will transition to 1 when Pitch & Roll values are valid from the IMU. 0 = Not ready. 1 = Ready.
IMU Pitch *IMU_Pitch* `0x3F6C 0x00`	–180 – 180° *–180 – 180*	–180 Deg. Read Only	Estimated pitch.
IMU Roll *IMU_Roll* `0x3F6B 0x00`	–180 – 180° *–180 – 180*	–180 Deg. Read Only	Estimated roll.
Vehicle Gyro X Raw *Raw_Gyro_X_v* `0x3F68 0x00`	–2147.484 – 2147.484 *–2147483647 – 2147483647*	–0.002 Read Only	Raw gyro rotation as measured by the IMU, converted to the vehicle coordinate frame x-axis.
Vehicle Gyro Y Raw *Raw_Gyro_Y_v* `0x3F69 0x00`	–2147.484 – 2147.484 *–2147483647 – 2147483647*	–0.002 Read Only	Raw gyro rotation as measured by the IMU, converted to the vehicle coordinate frame y-axis.
Vehicle Gyro Z Raw *Raw_Gyro_Z_v* `0x3F6A 0x00`	–2147.484 – 2147.484 *–2147483647 – 2147483647*	–0.002 Read Only	Raw gyro rotation as measured by the IMU, converted to the vehicle coordinate frame z-axis.
Vehicle Accel X Raw *Raw_Accel_X_v* `0x3F65 0x00`	–2147.484 – 2147.484 *–2147483647 – 2147483647*	–0.002 Read Only	Raw acceleration as measured by the IMU, converted to the vehicle coordinate frame x-axis.
Vehicle Accel Y Raw *Raw_Accel_Y_v* `0x3F66 0x00`	–2147.484 – 2147.484 *–2147483647 – 2147483647*	–0.002 Read Only	Raw acceleration as measured by the IMU, converted to the vehicle coordinate frame y-axis.
Vehicle Accel Z Raw *Raw_Accel_Z_v* `0x3F67 0x00`	–2147.484 – 2147.484 *–2147483647 – 2147483647*	–0.002 Read Only	Raw acceleration as measured by the IMU, converted to the vehicle coordinate frame z-axis.

APPLICATION SETUP — IMU INITIAL SETUP MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
IMU Select *IMU_Select* `0x3F83 0x00`	0 – 2 *0 – 2*	0	This parameter indicates whether the internal IMU exists and should be enabled, as well as allowing the use of an external IMU. 0: No IMU. 1: Use Internal IMU. 2: Use External IMU.
IMU Pitch offset *IMU_Pitch_Offset* `0x3F6E 0x00`	–180 – 180 degs. *–180 – 180*	0.000 degrees	Offset value in units of degrees subtracted from the 'Pitch' value after the attitude estimation algorithm, before data is sent to the user via VCL.
IMU Roll Offset *IMU_Roll_Offset* `0x3F6D 0x00`	–180 – 180 degs. *–180 – 180*	0.000 degrees	Offset value in units of degrees subtracted from the 'Roll' value after the attitude estimation algorithm, before data is sent to the user via VCL.
IMU Euler X Axis *IMU_Alignment_Euler_X* `0x3F62 0x00`	–180 – 180 degs. *–180 – 180*	0.000 degrees	Controller to vehicle frame Euler angle about X (phi) in units of degrees (Default 0, Range: –180 to 180).
IMU Euler Y Axis *IMU_Alignment_Euler_Y* `0x3F63 0x00`	–180 – 180 degs. *–180 – 180*	0.000 degrees	Controller to vehicle frame Euler angle about Y (omega) in units of degrees (Default 0, Range: –180 to 180).
IMU Euler Z Axis *IMU_Alignment_Euler_Z* `0x3F64 0x00`	–180 – 180 degs. *–180 – 180*	0.000 degrees	Controller to vehicle frame Euler angle about Z (psi) in units of degrees (Default 0, Range: –180 to 180).

APPLICATION SETUP — GYRO CALIBRATION MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Gyro Cal Type *Gyro_Cal_Type* `0x3F73 0x00`	0 – 1 *0 – 1*	0	Set *Gyro_Cal_Type* to 0 for commanded Calibration use with VCL function Start_Gyro_Calibration(), or when setting *Gyro_X_Cal*, *Gyro_Y_Cal*, *Gyro_Z_Cal* values directly. Set *Gyro_Cal_Type* to 1 to attempt automatic calibration when *Stationary_Wait_Time* has been satisfied at bootup.
Gyro Calibration Status *Gyro_Calibration_Status* `0x3F74 0x00`	0 – 2 *0 – 2*	0	The results of the commanded IMU Gyro re-calibration initiated by the VCL function Start_Gyro_Calibration(). 0 = Idle / Calibration Not Completed. 1 = Calibration Attempt In-Progress. 2 = Calibration Completed.
Gyro X Cal *Gyro_X_Cal* `0x3F7B 0x00`	–2147.484 – 2147.484 *–2147483648 – 2147483647*	0.000	*Gyro_X_Cal* is the degrees/second calibration offset.
Gyro Y Cal *Gyro_Y_Cal* `0x3F7C 0x00`	–2147.484 – 2147.484 *–2147483648 – 2147483647*	0.000	*Gyro_Y_Cal* is the degrees/second calibration offset.
Gyro Z Cal *Gyro_Z_Cal* `0x3F7D 0x00`	–2147.484 – 2147.484 *–2147483648 – 2147483647*	0.000	*Gyro_Z_Cal* is the degrees/second calibration offset.

CONTROLLER SETUP

Use the menus within the Controller Setup to configure the controller's inputs and output signals at the low-voltage (35-pin connector) and the current and power at the UVW motor-phase and Battery connections. The Inputs menu is where the analog inputs are setup and assigned. Following the Inputs menu are the IO Assignments, Outputs (coil drivers), External Supplies, and Current Limits menus.

CONTROLLER SETUP — INPUTS MENU

The Inputs parameter menu describes the optional usages for Analog 1, 6, 18, and 19. Based upon these input's setup selection, the CIT and 1313 HHP Programmer app will open and/or hide a different set of menu options. This is the context sensitivity aspect of the F-Series programmer menus.

The Analog Inputs 1, 6, 18, and 19 are selectable for use as Voltage, 3-Wire Potentiometer, 2-Wire Potentiometer, or a Voltage with Supply input. Based upon the selection, the available (visible) setup menus will differ. Review all the descriptions, below, to select the optimum option for the application.

Quick Links:
Fig. 12 p.16
Fig. 13 p.17
Fig. 32 p.113
Fig. 36 p.165

- When configuring either Analog Input 1 or Analog Input 6 as a 3-Wire Potentiometer input, the Programmer app will pair these inputs. The 3-wire input is the *wiper* and the other becomes the 5V supply, labeled *high*. For example, Figure 13 has Input 1 as the 3-wire wiper, while Input 6 is the supply (*high*). If Input 6 is set as the 3-wire (wiper), Input 1 will become the 5V supply (*high*). In all cases, the actual potentiometer wiper connects to ground (pins 7 or 18) as illustrated in Figures 13 and 36.
- The same type of coupling is true with Analog Input 18 (paired with Analog 19) if its 3-Wire Potentiometer is selected.

The other Analog inputs, 2 – 5, 7 – 9, 14, and 31 are voltage inputs. In a typical application, as shown in Figures 12 and 13, the drive motor temperature sensor is Analog 2. The motor encoder signals are Analog 3 and 4.

PWM Input 10 (Switch 10) on the 35-pin controllers is configurable for either an analog frequency or duty cycle input, besides its typical digital switch input. Figure 32 illustrates how this input's optional signal chain operates.

Note that the analog input's usage as a digital (switch) input has a higher normalization range (*analog_input_x_high*) maximum voltage limit of 30 volts. This allows their usage as digital (switch) inputs without causing a voltage out-of-range fault. This does not change the indicated limits for an input's analog measurement (usage) range.

CONTROLLER SETUP — INPUTS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Analog 1 Type *Analog_Input_1_Type* `0x32E6 0x00`	Enumeration *0 – 3*	Voltage	Configure the Analog1 input by throttle or load type. 0 – Voltage (Hall-effect or voltage throttle) 1 – 3-Wire Pot Wiper (3-wire resistive potentiometer throttle) 2 – 2-Wire Pot Wiper (2-wire resistive potentiometer throttle) 3 – Voltage with Supply (a non-throttle load alternative)
Analog 1 menu			
Parameters for **Voltage** selection	Reference the **Voltage Throttle** section, Chapter 6.		
Analog 1 Type *Analog_Input_1_Type* `0x32E6 0x00`	Voltage (selection menu)	–	Selecting the Voltage option opens the menu to its corresponding monitor variables and the low/high parameters.
Voltage *Analog_Input_Volts_1* `0x3B2E 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	The analog voltage at the analog 1 input.
Percent *Analog_Input_Percent_1* `0x3B39 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	The percentage of the voltage at the input pin based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_1) – (analog_input_1_low)] / [(analog_input_1_high) – (analog_input_1_low)] Voltage Throttle usage: Reference the Forward Min/Max & Reverse Min/Max Input parameters located in the *Application Setup » Throttle menu.* Brake input usage: Reference the Brake Min/Max Input parameters located in the *Application Setup » Brake* menu.
Low *Analog_Input_1_Low* `0x32F3 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. When the Analog 1 Type selection is voltage, this set point represents the 0% point for the normalized input. Triggers the Throttle Input fault (0x2210) when the voltage reading is below this input voltage reading, if *Throttle_Source* = 1. Triggers the Brake Input fault (0x2310) when the voltage reading is below this input voltage reading, if *Brake_Source* = 1.
High *Analog_Input_1_High* `0x32F4 0x00`	0.0 – 11.0V *0 – 110*	5.1V	The maximum input voltage before a fault is declared. When the Analog 1 Type selection is voltage, this set point represents the 100% point for the normalized input. Triggers the Throttle Input fault (0x2210) when the voltage reading is above this input voltage reading, if *Throttle_Source* = 1. Triggers the Brake Input fault (0x2310) when the voltage reading is above this input voltage reading, if *Brake_Source* = 1.
Fault Tolerance *Analog_Input_1_Fault_Tolerance* `0x331A 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 1 menu			
Parameters for **3-Wire Pot** selection	Reference the **3-wire potentiometer throttle** section, Chapter 6.		
Potentiometer 1 (wiper) **& 6** (high)			
Analog 1 Type *Analog_Input_1_Type* `0x32E6 0x00`	3 Wire Pot (selection menu)	–	Selecting the 3-Wire Pot option opens the Nominal Resistance parameter. Note, when configuring Analog input 1 as a 3-wire Potentiometer input, it becomes the *wiper* (Input 1) and is paired with Analog 6 (Input 6) as the potentiometer wiper's 5V supply voltage (*high*). The Programmer app will automatically adjust the menu to match this parameter configuration.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 12 p.16
Fig. 13 p.17

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Nominal Resistance *Pot_1_R* `0x3356 0x00`	800 – 15000 Ohms	5000 Ω	Set the potentiometer resistance. Throttle or Brake potentiometers are typically 1k, 5k, or 10k Ohms. Note: If this parameter value is outside the actual resistance, it will trigger a Throttle Input fault (Flash code 0x42) if *Throttle_Source* = 1. Note: If this parameter value is outside the actual resistance, it will trigger a Brake Input fault (Flash code 0x44) if *Brake_Source* = 1.
Analog 1 menu			
Parameters for **2-Wire Pot** selection		Reference the **2-wire potentiometer throttle** section, Chapter 6.	
Potentiometer 1			
Analog 1 Type *Analog_Input_1_Type* `0x32E6 0x00`	2 Wire Pot (selection menu)	–	Selecting the 2-Wire Pot option opens the Nominal Resistance parameter. Note, when configuring Analog input 1 as a 2-wire Potentiometer input, the connection (pin) is both the voltage supply and the wiper input. As a 2-wire pot, one side of the potentiometer is left open (no connection) and the other end is connected to ground (i.e., I/O Gnd). The Programmer app will automatically adjust the menu to match this parameter configuration.
Nominal Resistance *Pot_1_R* `0x3356 0x00`	800 – 15000 Ohms	5000 Ω	Set the nominal resistance of the potentiometer. Throttle or Brake potentiometers are typically 1k, 5k, or 10k Ohms. Note: If this parameter value is outside the actual resistance, it will trigger a Throttle Input fault (Flash code 0x42) if *Throttle_Source* = 1. Note: If this parameter value is outside the actual resistance, it will trigger a Brake Input fault (Flash code 0x44) if *Brake_Source* = 1.
Analog 1 menu			
Parameters for **Voltage with Supply** selection			
Analog 1 Type *Analog_Input_1_Type* `0x32E6 0x00`	Voltage with Supply (selection menu)	–	The Analog 1 Type "Voltage with Supply" selection operates the Analog 1 input as a raw voltage at the pin with an internal pull-up supply voltage. The pin's output voltage is 10V (approx.) with a 3 mA current limit (to the external load/device). Reference Table 10. Use this selection for operating resistive devices such as thermistors and photocells and reading the voltage in VCL using the Monitor variable *analog_input_volts_1* (0x3B2E), then processing the value in the VCL program.
Voltage *Analog_Input_Volts_1* `0x3B2E 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	The analog voltage at analog 1 input (input 1).
Percent *Analog_Input_Percent_1* `0x3B39 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_1) – (analog_input_1_low)] / [(analog_input_1_high) – (analog_input_1_low)]

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17
Table 10 p.24

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Low *Analog_Input_1_Low* `0x32F3 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. When the Analog 1 Type selection is Voltage with Supply, this set point represents the 0% point for the normalized input. Faults: When the voltage falls below this parameter, it will trigger the *Analog_1_OUT_OF_RANGE fault (flash code 0xB1, CAN Index 0x2620)*. If assigning the Throttle to Analog 1, the resulting fault will become the *Throttle Input* fault *(flash code 0x42, CAN Index 0x2210) if Throttle_Source = 1*. If assigning the Brake to Analog 1, the resulting fault will become the *Brake Input* fault (flash code 0x44) if *Brake_Source* = 1.
High *Analog_Input_1_High* `0x32F4 0x00`	0.0 – 11.0V *0 – 110*	5.1V	The maximum input voltage before a fault is declared. When the Analog 1 Type selection is Voltage with Supply, this set point represents the 100% point for the normalized input. Faults: When the voltage goes above this parameter, it will trigger the *Analog_1_OUT_OF_RANGE fault (flash code 0xB1, Can Index 0x2620)*. If assigning the Throttle to Analog 1, the resulting fault will become the *Throttle Input* fault *(flash code 0x42, CAN Index 0x2210) if Throttle_Source = 1*. If assigning the Brake to Analog 1, the resulting fault will become the *Brake Input* fault (flash code 0x44) if *Brake_Source* = 1.
Fault Tolerance *Analog_Input_1_Fault_Tolerance* `0x331A 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 2 menu			
Voltage *Analog_Input_Volts_2* `0x3B2F 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at analog 2 (Input 2) (*as the motor temp sensor in Fig 13*).
Percent *Analog_Input_Percent_2* `0x3B3A 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the analog 2 input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_2) – (analog_input_2_low)] / [(analog_input_2_high) – (analog_input_2_low)]
Low *Analog_Input_2_Low* `0x32F5 0x00`	0.0 – 7.0V *0 – 700*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0% point for the normalized inputs.
High *Analog_Input_2_High* `0x32F6 0x00`	0.0 – 7.0V *0 – 700*	7.0V	The maximum input voltage before a fault is declared. This voltage is the 100% point for the normalized inputs.
Fault Tolerance *Analog_Input_2_Fault_Tolerance* `0x331B 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17
Table 10 p.24

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Analog 3 menu			
Voltage *Analog_Input_Volts_3* 0x3B30 0x00	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at the analog 3 input (*as the motor encoder-A signal in Fig 13*).
Percent *Analog_Input_Percent_3* 0x3B3B 0x00	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_3) – (analog_input_3_low)] / [(analog_input_3_high) – (analog_input_3_low)]
Low *Analog_Input_3_Low* 0x32F7 0x00	0.0 – 7.0V *0 – 700*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0% point for the normalized inputs.
High *Analog_Input_3_High* 0x32F8 0x00	0.0 – 7.0V *0 – 700*	5.0V	The maximum input voltage before a fault is declared. This voltage is the 100% point for the normalized inputs.
Fault Tolerance *Analog_Input_3_Fault_Tolerance* 0x331C 0x00	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 4 menu			
Voltage *Analog_Input_Volts_4* 0x3B31 0x00	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at the analog 4 input (*as the motor encoder-A signal in Figure 13*).
Percent *Analog_Input_Percent_4* 0x3B3C 0x00	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_4) – (analog_input_4_low)] / [(analog_input_4_high) – (analog_input_4_low)]
Low *Analog_Input_4_Low* 0x32F9 0x00	0.0 – 7.0V *0 – 700*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0 % point for the normalized inputs.
High *Analog_Input_4_High* 0x32FA 0x00	0.0 – 7.0V *0 – 700*	5.0V	The maximum input voltage before a fault is declared. This voltage is the 100 % point for the normalized inputs.
Fault Tolerance *Analog_Input_4_Fault_Tolerance* 0x331D 0x00	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 5 menu			
Voltage *Analog_Input_Volts_5* 0x3B32 0x00	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at the analog 5 input (Input 5).

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Percent *Analog_Input_Percent_5* `0x3B3D 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_5) – (analog_input_5_low)] / [(analog_input_5_high) – (analog_input_5_low)]
Low *Analog_Input_5_Low* `0x32FB 0x00`	0.0 – 30.0V *0 – 3000*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0 % point for the normalized inputs.
High *Analog_Input_5_High* `0x32FC 0x00`	0.0 – 30.0V *0 – 3000*	30.0V	The maximum input voltage before a fault is declared. This voltage is the 100 % point for the normalized inputs.
Fault Tolerance *Analog_Input_5_Fault_Tolerance* `0x331E 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 6 menu			
Analog 6 Type *Analog_Input_6_Type* `0x3310 0x00`	Enumeration *0 – 3*	Voltage	Configure the Analog 6 input by throttle or load type. 0 – Voltage (Hall-effect or voltage throttle) 1 – 3-Wire Pot Wiper (3-wire resistive potentiometer throttle) 2 – 2-Wire Pot Wiper (2-wire resistive potentiometer throttle) 3 – Voltage with Supply (a non-throttle load alternative)
Analog 6 menu			
Parameters for **Voltage** selection		Reference the **Voltage Throttle** section, Chapter 6.	
Analog 6 Type *Analog_Input_6_Type* `0x3310 0x00`	Voltage (selection menu)	–	Selecting the Voltage option opens the menu to its corresponding monitor variables and the low/high parameters.
Voltage *Analog_Input_Volts_6* `0x3B33 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	The analog voltage at the analog 6 input (Input 6).
Percent *Analog_Input_Percent_6* `0x3B3E 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_6) – (analog_input_6_low)] / [(analog_input_6_high) – (analog_input_6_low)] Voltage Throttle usage: Reference the Forward Min/Max & Reverse Min/Max Input parameters located in the *Application Setup » Throttle menu.* Brake input usage: Reference the Brake Min/Max Input parameters located in the *Application Setup » Brake* menu.
Low *Analog_Input_6_Low* `0x32FD 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. When the Analog 6 Type selection is voltage, this set point represents the 0% point for the normalized input. Triggers the Throttle Input fault (0x2210) when the voltage reading is below this input voltage reading, if *Throttle_Source* = 6. Triggers the Brake Input fault (0x2310) when the voltage reading is below this input voltage reading, if *Brake_Source* = 6.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
High *Analog_Input_6_High* `0x32FE 0x00`	0.0 – 11.0V *0 – 1100*	11.0V	The maximum input voltage before a fault is declared. When the Analog 6 Type selection is voltage, this set point represents the 100% point for the normalized input. Triggers the Throttle Input fault (0x2210) when the voltage reading is above this input voltage reading, if *Throttle_Source* = 6. Triggers the Brake Input fault (0x2310) when the voltage reading is above this input voltage reading, if *Brake_Source* = 6.
Fault Tolerance *Analog_Input_6_Fault_Tolerance* `0x331F 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 6 menu			
Parameters for **3-Wire Pot** selection			Reference the **3-wire potentiometer throttle** section, Chapter 6.
Potentiometer 6 (wiper) **& 1** (high)			
Analog 6 Type *Analog_Input_6_Type* `0x3310 0x00`	3 Wire Pot (selection menu)	–	Selection of the 3-Wire Pot opens the Nominal Resistance parameter. Note: When configuring Analog Input 6 as a 3-wire Potentiometer input, it becomes the *wiper* (Input 6) and is paired with Analog 1 (Input 1) as the potentiometer wiper's 5V supply voltage (*high*). The Programmer app will automatically adjust the menu to match this parameter configuration.
Nominal Resistance *Pot_6_R* `0x336C 0x00`	800 – 15000 Ohms	5000 Ω	Set the potentiometer resistance. Throttle or Brake potentiometers are typically 1k, 5k, or 10k Ohms. Note: If this parameter value is outside the actual resistance, it will trigger a Throttle Input fault (Flash code 0x42) if *Throttle_Source* = 6. Note: If this parameter value is outside the actual resistance, it will trigger a Brake Input fault (Flash code 0x44) if *Brake_Source* = 6.
Analog 6 menu			
Parameters for **2-Wire Pot** selection			Reference the **2-wire potentiometer throttle** section, Chapter 6.
Potentiometer 6			
Analog 6 Type *Analog_Input_6_Type* `0x3310 0x00`	2 Wire Pot (selection menu)	–	Selection of the 2-Wire Pot opens the Nominal Resistance parameter. Note, when configuring Analog input 6 as a 2-wire Potentiometer input, the connection (pin) is both the voltage supply and the wiper input. As a 2-wire pot, one side of the potentiometer is left open (no connection) and the other end is connected to ground (i.e., I/O Gnd). The Programmer app will automatically adjust the menu to match this parameter configuration.
Nominal Resistance *Pot_6_R* `0x336C 0x00`	800 – 15000 Ohms	5000 Ω	Set the nominal resistance of the potentiometer. Throttle or Brake potentiometers are typically 1k, 5k, or 10k Ohms. Note: If this parameter value is outside the actual resistance, it will trigger a Throttle Input fault (Flash code 0x42), if *Throttle_Source* = 6. Note: If this parameter value is outside the actual resistance, it will trigger a Brake Input fault (Flash code 0x44), if *Brake_Source* = 6.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17
Table 10 p.24

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Analog 6 menu			
Parameters for **Voltage with Supply** selection			
Analog 6 Type *Analog_Input_6_Type* `0x3310 0x00`	Voltage with Supply (selection menu)	–	The Analog 6 Type "Voltage with Supply" selection operates the Analog 6 input as a raw voltage at the pin with an internal pull-up supply voltage. The pin's output voltage is 10V (approx.) with a 3 mA current limit (to the external load/device). Reference Table 10. Use this selection for operating resistive devices such as thermistors and photocells and reading the voltage in VCL using the Monitor variable analog_input_volts_6 (0x3B33), then processing the value in the VCL program.
Voltage *Analog_Input_Volts_6* `0x3B33 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	The analog voltage at the analog 6 input (Input 6).
Percent *Analog_Input_Percent_6* `0x3B3E 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	The percentage of the voltage at the analog 6 input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_6) – (analog_input_6_low)] / [(analog_input_6_high) – (analog_input_6_low)]
Low *Analog_Input_6_Low* `0x32FD 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. When the Analog 6 Type selection is Voltage with Supply, this set point represents the 0% point for the normalized input. Faults: When the voltage falls below this parameter, it will trigger the *Analog_6_OUT_OF_RANGE fault (flash code 0xB6, CAN Index 0x2625).* If assigning the Throttle to Analog 6, the resulting fault will become the *Throttle Input* fault *(flash code 0x42, CAN Index 0x2210) if Throttle_Source = 1.* If assigning the Brake to Analog 6, the resulting fault will become the *Brake Input* fault (Flash code 0x44) if *Brake_Source* = 6.
High *Analog_Input_6_High* `0x32FE 0x00`	0.0 – 11.0V *0 – 110*	11.0V	The maximum input voltage before a fault is declared. When the Analog 6 Type selection is Voltage with Supply, this set point represents the 100% point for the normalized input. Faults: When the voltage rises above this parameter, it will trigger the *Analog_6_OUT_OF_RANGE fault (flash code 0xB6, Can Index 0x2625).* If assigning the Throttle to Analog 6, the resulting fault will become the *Throttle Input fault (flash code 0x42, CAN Index 0x2210) if Throttle_Source = 1.* If assigning the Brake to Analog 6, the resulting fault will become the *Brake Input* fault (Flash code 0x44) if *Brake_Source* = 6.
Fault Tolerance *Analog_Input_6_Fault_Tolerance* `0x331F 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 7 menu			
Voltage *Analog_Input_Volts_7* `0x3B34 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at analog 7 input (Input 7).

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Percent *Analog_Input_Percent_7* 0x3B3F 0x00	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_7) – (analog_input_7_low)] / [(analog_input_7_high) – (analog_input_7_low)]
Low *Analog_Input_7_Low* 0x32FF 0x00	0.0 – 30.0V *0 – 3000*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0% point for the normalized inputs.
High *Analog_Input_7_High* 0x3300 0x00	0.0 – 30.0V *0 – 3000*	30.0V	The maximum input voltage before a fault is declared. This voltage is the 100% point for the normalized inputs.
Fault Tolerance *Analog_Input_7_Fault_Tolerance* 0x334C 0x00	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 8 menu			
Voltage *Analog_Input_Volts_8* 0x3B35 0x00	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at analog 8 input (Input 8).
Percent *Analog_Input_Percent_8* 0x32E4 0x00	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_8) – (analog_input_8_low)] / [(analog_input_8_high) – (analog_input_8_low)]
Low *Analog_Input_8_Low* 0x3307 0x00	0.0 – 30.0V *0 – 3000*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0% point for the normalized inputs.
High *Analog_Input_8_High* 0x3308 0x00	0.0 – 30.0V *0 – 3000*	30.0V	The maximum input voltage before a fault is declared. This voltage is the 100% point for the normalized inputs.
Fault Tolerance *Analog_Input_8_Fault_Tolerance* 0x3377 0x00	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 9 menu			
Voltage *analog_input_volts_9* 0x3B36 0x00	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at the analog 9 input (Input 9).
Percent *Analog_Input_Percent_9* 0x3B40 0x00	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the input pin is based upon the High and Low settings. i.e., the % of: [(analog_input_volts_9) – (analog_input_9_low)] / [(analog_input_9_high) – (analog_input_9_low)]

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Low *analog_input_9_low* `0x3301 0x00`	0.0 – 30.0V *0 – 3000*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0% point for the normalized inputs.
High *analog_input_9_high* `0x3302 0x00`	0.0 – 30.0V *0 – 3000*	30.0V	The maximum input voltage before a fault is declared. This voltage is the 100% point for the normalized inputs.
Fault Tolerance *Analog_Input_9_Fault_Tolerance* `0x3378 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
PWM Input 10			
PWM Input 10 Type *PWM_Input_10_Type* `0x5088 0x00`	Enumerated* *0 – 2* *Enumerated Options: 0 – Disabled 1 – Duty Cycle 2 – Frequency	0	Input 10 is a high frequency input for either duty-cycle or frequency signals. When not using the PWM input options, the input operates as a switch input. See the signal-chain figure, next page. 0 – The PWM Input is not used, and disables fault checking on this PWM input. This option, for example, is when this input is used as a switch input. 1 – The *PWM_Input_10_Percent* is normalized to the duty cycle low/high range. Fault check is active. 2 – The *PWM_Input_10_Percent* is normalized to the frequency low/high range. Fault check is active.
PWM Input 10 Type = (1) duty cycle or (2) frequency These PWM Input 10 monitor and parameter items are only visible when options 1 or 2 are selected These items are visible within the Programmer app, *Controller Setup » Inputs » PWM **Input 10*** menu			
PWM Input 10 menu			
Duty Cycle *PWM_Input_10_Cycle* `0x3B69 0x00`	0.0 – 100.0 % *0 – 1000*	0.0 % Read Only	PWM Input 10 duty cycle measurement.
Frequency *PWM_Input_10_Frequency* `0x3B67 0x00`	0 – 10000 *0 – 10000*	0 Hz Read Only	PWM Input frequency measurement.
Percent *PWM_Input_10_Percent* `0x3192 0x00`	0.0 – 100.0 % *0 – 1000*	0.0 % Read Only	PWM input 10 as a percentage of the configured type (duty cycle, or frequency). 0% maps to min, 100% maps to max.
Duty Cycle Low *PWM_Input_10_Low_Duty_Cycle* `0x5080 0x00`	0 – 100 % *0 – 1000*	0 %	PWM Input low duty cycle threshold.
Duty Cycle High *PWM_Input_10_High_Duty_Cycle* `0x5081 0x00`	0 – 100 % *0 – 1000*	100 %	PWM Input high duty cycle threshold. Note: Do not set the high value lower than the corresponding low value.
Duty Cycle Fault Tolerance *PWM_Input_10_Duty_Cycle_Fault_Tolerance* `0x33AF 0x00`	0 – 100 % *0 – 1000*	0 %	Specifies the PWM input duty cycle threshold above the configured high limit or below the low limit. Exceeding this results in a PWM input out of range fault. For example, if High = 96%, Low = 1% and Tolerance = 0.2%, a PWM input duty cycle between 1% to 96% is mapped to 0-100%. PWM input above 96.2% (96 + 0.2) or below 0.8% (1% – 0.2%) triggers the fault.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Frequency Low *PWM_Input_10_Low_Frequency* 0x5082 0x00	500 – 10000 Hz *500 – 10000*	500 Hz	PWM Input low frequency threshold.
Frequency High *PWM_Input_10_High_Frequency* 0x5083 0x00	0 – 10000 Hz *0 – 10000*	10000 Hz	PWM Input high frequency threshold. Note: Do not set the high value lower than the corresponding low value.
Frequency Fault Tolerance *PWM_Input_10_Frequency_Fault_Tolerance* 0x33B3 0x00	0 – 10000 Hz *0 – 10000*	0 Hz	Specifies the PWM input frequency threshold above the configured high limit or below the low limit. Exceeding this results in a PWM input out of range fault. For example, if High = 9000 Hz, Low = 100 Hz and Tolerance = 20 Hz, a PWM input frequency between 100 Hz to 9000 Hz is mapped to 0-100%. PWM input above 9020 Hz (9000 + 20) or below 80 Hz (100 Hz - 20 Hz) triggers the fault.

Quick Links:

Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17

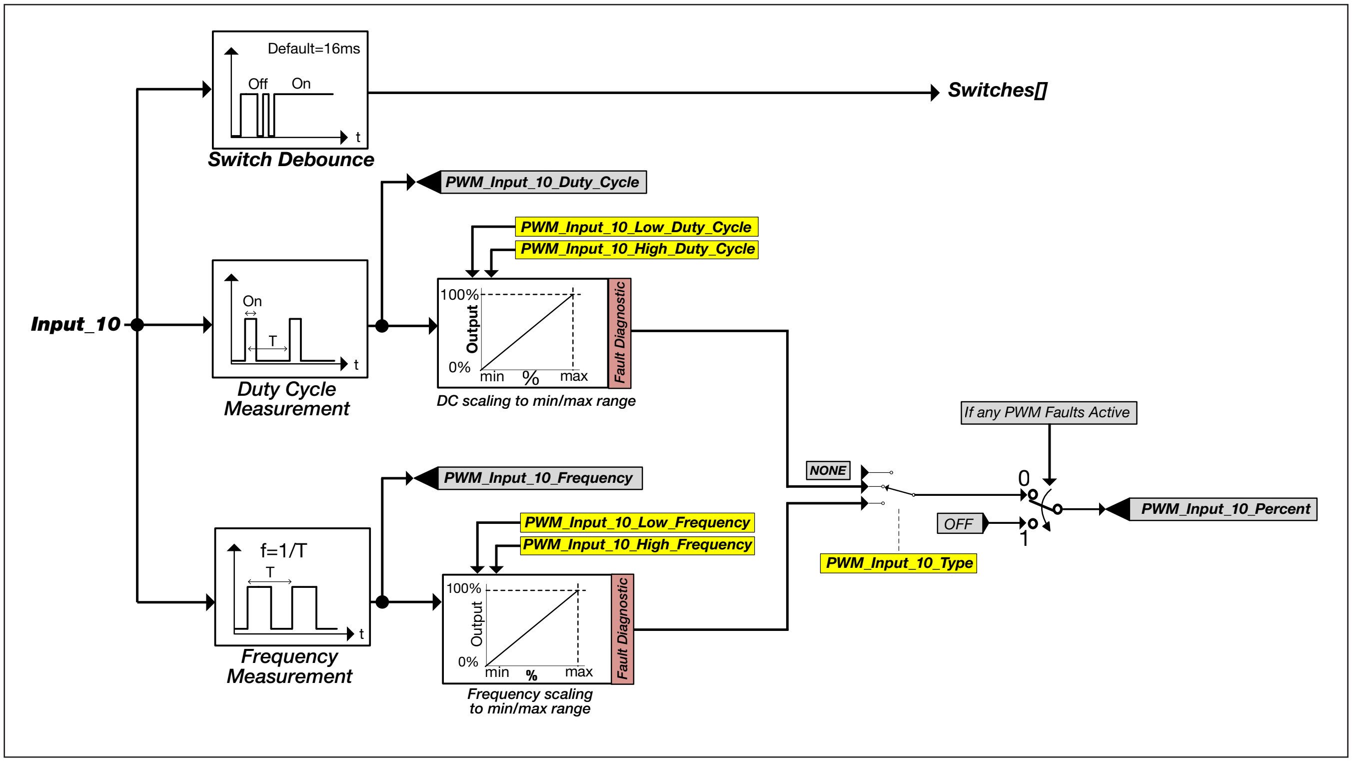


Figure 32

Input 10 Digital and Analog Signal Chain

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Analog 14 menu			
Voltage *Analog_Input_Volts_14* `0x3B38 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at the analog 14 input (Input 14). (*as the +12V External Supply in Fig 13*).
Percent *Analog_Input_Percent_14* `0x3B42 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_14) – (analog_input_14_low)] / [(analog_input_14_high) – (analog_input_14_low)]
Low *Analog_Input_14_Low* `0x3303 0x00`	0.0 – 30.0V *0 – 3000*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0 % point for the normalized inputs.
High *Analog_Input_14_High* `0x3306 0x00`	0.0 – 30.0V *0 – 3000*	30.0V	The maximum input voltage before a fault is declared. This voltage is the 100% point for the normalized inputs.
Fault Tolerance *Analog_Input_14_Fault_ Tolerance* `0x3379 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 18 menu			
Analog 18 Type *Analog_Input_18_Type* `0x330F 0x00`	Enumeration *0 – 3*	Voltage	Configure the Analog 18 input by throttle or load type. 0 – Voltage (Hall-effect or voltage throttle) 1 – 3-Wire Pot Wiper (3-wire resistive potentiometer throttle) 2 – 2-Wire Pot Wiper (2-wire resistive potentiometer throttle) 3 – Voltage with Supply (a non-throttle load alternative)
Parameters for Analog 18 **Voltage** selection		Reference the **Voltage Throttle** section, Chapter 6.	
Analog 18 Type *Analog_Input_18_Type* `0x330F 0x00`	Voltage (selection menu)	–	Selecting the Voltage option opens the menu to its corresponding monitor variables and the low/high parameters.
Voltage *Analog_Input_Volts_18* `0x3B60 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	The analog voltage at the analog 18 input (Input 18).
Percent *Analog_Input_Percent_18* `0x3B62 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_18) – (analog_input_18_low)] / [(analog_input_18_high) – (analog_input_18_low)] Voltage Throttle usage: Reference the Forward Min/Max & Reverse Min/Max Input parameters located in the *Application Setup » Throttle* menu. Brake input usage: Reference the Brake Min/Max Input parameters located in the *Application Setup » Brake* menu.
Low *Analog_Input_18_Low* `0x3309 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. When the Analog 18 Type selection is voltage, this set point represents the 0% point for the normalized input. Triggers the *Throttle Input* fault (0x2210) when the voltage reading is below this input voltage reading, if *Throttle_Source* = 18. Triggers the *Brake Input* fault (0x2310) when the voltage reading is below this input voltage reading, if *Brake_Source* = 18.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
High *Analog_Input_18_High* 0x330A 0x00	0.0 – 11.0V *0 – 110*	5.1V	The maximum input voltage before a fault is declared. When the Analog 18 Type selection is voltage, set point represents the 100% point for the normalized input. Triggers the Throttle Input fault (0x2210) when the voltage reading is above this input voltage reading, if *Throttle_Source* = 18. Triggers the Brake Input fault (0x2310) when the voltage reading is above this input voltage reading, if *Brake_Source* = 18.
Fault Tolerance *Analog_Input_18_Fault_Tolerance* 0x337A 0x00	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 18 menu			
Parameters for **3-Wire Pot** selection		Reference the **3-wire potentiometer throttle** section, Chapter 6.	
Potentiometer 18 (wiper) **& 19** (high)			
Analog 18 Type *Analog_Input_18_Type* 0x330F 0x00	3 Wire Pot (selection menu)	–	Selection of the 3-Wire Pot opens the Nominal Resistance parameter. Note: When configuring Analog Input 18 as a 3-wire Potentiometer input, it becomes the *wiper* (Input 18) and is paired with Analog 19 (Input 19) as the potentiometer wiper's 5V supply voltage (*high*). The Programmer app will automatically adjust the menu to match this parameter configuration.
Nominal Resistance *Pot_18_R* 0x3357 0x00	800 – 15000 Ω *800 – 15000*	5000 Ω	Set the nominal resistance of the potentiometer. Throttle or Brake potentiometers are typically 1k, 5k, or 10k Ohms. Note: If this parameter value is outside the actual resistance, it will trigger a Throttle Input fault (Flash code 0x42) if *Throttle_Source* = 18. Note: If this parameter value is outside the actual resistance, it will trigger a Brake Input fault (Flash code 0x44) if *Brake_Source* = 18.
Analog 18 menu			
Parameters for **2-Wire Pot** selection		Reference the **2-wire potentiometer throttle** section, Chapter 6.	
Potentiometer 18			
Analog 18 Type *Analog_Input_18_Type* 0x330F 0x00	2 Wire Pot (selection menu)	–	Selection of the 2-Wire Pot opens the Nominal Resistance parameter. Note: when configuring Analog input 18 as a 2-wire Potentiometer input, the connection (pin) is both the voltage supply and wiper input. As a 2-wire pot, one side of the potentiometer is left open (no connection) and the other end is connected to ground (i.e., I/O Gnd). The Programmer app will automatically adjust the menu to match this parameter configuration.
Nominal Resistance *Pot_18_R* 0x3357 0x00	800 – 15000 Ω *800 – 15000*	5000 Ω	Set the nominal resistance of the potentiometer. Throttle or Brake potentiometers are typically 1k, 5k, or 10k Ohms. Note: If this parameter value is outside the actual resistance, it will trigger a Throttle Input fault (Flash code 0x42) if *Throttle_Source* = 18. Note: If this parameter value is outside the actual resistance, it will trigger a Brake Input fault (Flash code 0x44) if *Brake_Source* = 18.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Analog 18 menu			
Parameters for **Voltage with Supply** selection			
Analog 18 Type *Analog_Input_18_Type* 0x330F 0x00	Voltage with Supply (selection menu)	–	The Analog 18 Type "Voltage with Supply" selection operates the Analog 18 input as a raw voltage at the pin with an internal pull-up supply voltage. The pin's output voltage is 10V (approx.) with a 3 mA current limit (to the external load/device). Reference Table 10. Use this selection for operating resistive devices such as thermistors and photocells and reading the voltage in VCL using the Monitor variable analog_input_volts_18 (0x3B60), then processing the value in the VCL program.
Voltage *Analog_Input_Volts_18* 0x3B60 0x00	–327.68 – 327.67V *–32768 – 32767*	Read Only	The analog voltage at the analog 18 input (Input 18).
Percent *Analog_Input_Percent_18* 0x3B62 0x00	0.0 – 100.0 % *0 – 1000*	Read Only	The percentage of the voltage at pin 17 based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_18) – (analog_input_18_low)] / [(analog_input_18_high) – (analog_input_18_low)]
Low *Analog_Input_18_Low* 0x3309 0x00	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. When the Analog 18 Type selection is Voltage with Supply, this set point represents the 0% point for the normalized input. Faults: When the voltage falls below this parameter, it will trigger the *Analog_18_OUT_OF_RANGE fault (flash code 0xBD, CAN Index 0x262B)*. If assigning the Throttle to Analog 18, the resulting fault will become the *Throttle Input* fault *(flash code 0x42, CAN Index 0x2210)* if *Throttle_Source* = 18. If assigning the Brake to Analog 18, the resulting fault will become the *Brake Input* fault (Flash code 0x44) if *Brake_Source* = 18.
High *Analog_Input_18_High* 0x330A 0x00	0.0 – 11.0V *0 – 110*	5.1V	The maximum input voltage before a fault is declared. When the Analog 18 Type selection is Voltage with Supply, this set point represents the 100% point for the normalized input. Faults: When the voltage rises above this parameter, it will trigger the *Analog_18_OUT_OF_RANGE fault (flash code 0xBD, CAN Index 0x262B)*. If assigning the Throttle to Analog 18, the resulting fault will become the *Throttle Input* fault *(flash code 0x42, CAN Index 0x2210)* if *Throttle_Source* = 18. If assigning the Brake to Analog 18, the resulting fault will become the *Brake Input* fault (Flash code 0x44) if *Brake_Source* = 18.
Fault Tolerance *Analog_Input_18_Fault_Tolerance* 0x337A 0x00	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17
Table 10 p.24

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Analog 19 menu			
Analog 19 Type *Analog_Input_19_Type* `0x3311 0x00`	Enumeration *0 – 3*	Voltage	Configure the Analog 19 input by throttle or load type. 0 – Voltage (Hall-effect or voltage throttle) 1 – 3-Wire Pot Wiper (3-wire resistive potentiometer throttle) 2 – 2-Wire Pot Wiper (2-wire resistive potentiometer throttle) 3 – Voltage with Supply (a non-throttle load alternative)
Parameters for Analog 19 **Voltage** selection		Reference the **Voltage Throttle** section, Chapter 6.	
Analog 19 Type *Analog_Input_19_Type* `0x3311 0x00`	Voltage (selection menu)	–	Selecting the Voltage option opens the menu to its corresponding monitor variables and the low/high parameters.
Voltage *Analog_Input_Volts_19* `0x3B60 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	The analog voltage at the analog 19 input (Input 19).
Percent *Analog_Input_Percent_19* `0x3B62 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_19) – (analog_input_19_low)] / [(analog_input_19_high) – (analog_input_19_low)] Voltage Throttle usage: Reference the Forward Min/Max & Reverse Min/Max Input parameters located in the *Application Setup » Throttle* menu. Brake input usage: Reference the Brake Min/Max Input parameters located in the *Application Setup » Brake* menu.
Low *Analog_Input_19_Low* `0x330B 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. When the Analog 19 Type selection is voltage, this set point represents the 0% point for the normalized input. Triggers the *Throttle Input fault* (0x2210) when the voltage reading is below this input voltage reading, if *Throttle_Source* = 19. Triggers the *Brake Input* fault (0x2310) when the voltage reading is below this input voltage reading, if *Brake_Source* = 19.
High *Analog_Input_19_High* `0x330C 0x00`	0.0 – 11.0V *0 – 110*	11.0V	The maximum input voltage before a fault is declared. When the Analog 19 Type selection is voltage, this set point represents the 100% point for the normalized input. Triggers the Throttle Input fault (0x2210) when the voltage reading is above this input voltage reading, if *Throttle_Source* = 19. Triggers the Brake Input fault (0x2310) when the voltage reading is above this input voltage reading, if *Brake_Source* = 19.
Fault Tolerance *Analog_Input_19_Fault_Tolerance* `0x337B 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Analog 19 menu			
Parameters for **3-Wire Pot** selection			Reference the **3-wire potentiometer throttle** section, Chapter 6.
Potentiometer 19 (wiper) **& 18** (high)			
Analog 19 Type *Analog_Input_19_Type* `0x3311 0x00`	3 Wire Pot (selection menu)	–	Selection of the 3-Wire Pot opens the Nominal Resistance parameter. Note: When configuring Analog Input 19 as a 3-wire Potentiometer input, it becomes the *wiper* (Input 19) and is paired with Analog 18 (Input 18) as the potentiometer wiper's 5V supply voltage (*high*). The Programmer app will automatically adjust the menu to match this parameter configuration.
Nominal Resistance *Pot_19_R* `0x336D 0x00`	800 – 15000 Ohms	5000 Ω	Set the potentiometer resistance. Throttle or Brake potentiometers are typically 1k, 5k, or 10k Ohms. Note: If this parameter value is outside the actual resistance, it will trigger a Throttle Input fault (Flash code 0x42) if *Throttle_Source* = 19. Note: If this parameter value is outside the actual resistance, it will trigger a Brake Input fault (Flash code 0x44) if *Brake_Source* = 19.
Analog 19 menu			
Parameters for **2-Wire Pot** selection			Reference the **2-wire potentiometer throttle** section, Chapter 6.
Potentiometer 19			
Analog 19 Type *Analog_Input_19_Type* `0x3311 0x00`	2 Wire Pot (selection menu)	–	Selection of the 2-Wire Pot opens the Nominal Resistance parameter. Note: when configuring Analog input 19 as a 2-wire Potentiometer input, the connection (pin) is both the voltage supply and wiper input. As a 2-wire pot, one side of the potentiometer is left open (no connection) and the other end is connected to ground (i.e., I/O Gnd). The Programmer app will automatically adjust the menu to match this parameter configuration.
Nominal Resistance *Pot_19_R* `0x336D 0x00`	800 – 15000 Ohms *800 – 15000*	5000 Ω	Set the nominal resistance of the potentiometer. Throttle or Brake potentiometers are typically 1k, 5k, or 10k Ohms. Note: If this parameter value is outside the actual resistance, it will trigger a Throttle Input fault (Flash code 0x42) if *Throttle_Source* = 19. Note: If this parameter value is outside the actual resistance, it will trigger a Brake Input fault (Flash code 0x44) if *Brake_Source* = 19.
Analog 19 menu			
Parameters for **Voltage with Supply** selection			
Analog 19 Type *Analog_Input_19_Type* `0x3311 0x00`	Voltage with Supply (selection menu)	–	The Analog 19 Type "Voltage with Supply" selection operates the Analog 19 input as a raw voltage at the pin with an internal pull-up supply voltage. The pin's output voltage is 10V (approx.) with a 3 mA current limit (to the external load/device). Reference Table 10. Use this selection for operating resistive devices such as thermistors and photocells and reading the voltage in VCL using the Monitor variable *analog_input_volts_19* (0x3B61), then processing the value in the VCL program.
Voltage *Analog_Input_Volts_19* `0x3B61 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	The analog voltage at the analog 19 input (Input 19).

Quick Links:
Voltage Throttle p.166
3-Wire Throttle p.165
2-Wire Throttle p.165
Fig. 13 p.17
Table 10 p.24

CONTROLLER SETUP — INPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Percent *Analog_Input_Percent_19* `0x3B63 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	The percentage of the voltage at the input pin is based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_19) – (analog_input_19_low)] / [(analog_input_19_high) – (analog_input_19_low)]
Low *Analog_Input_19_Low* `0x330B 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. When the Analog 19 Type selection is Voltage with Supply, this set point represents the 0% point for the normalized input. Faults: When the voltage falls below this parameter, it will trigger the *Analog_19_OUT_OF_RANGE fault (flash code 0xBE, CAN Index 0x262C).* If assigning the Throttle to Analog 19, the resulting fault will become the *Throttle Input* fault *(flash code 0x42, CAN Index 0x2210)* if *Throttle_Source* = 19. If assigning the Brake to Analog 19, the resulting fault will become the *Brake Input* fault (Flash code 0x44) if *Brake_Source* = 19.
High *Analog_Input_19_High* `0x330C 0x00`	0.0 – 11.0V *0 – 110*	11.0V	The maximum input voltage before a fault is declared. When the Analog 19 Type selection is Voltage with Supply, this set point represents the 100% point for the normalized input. Faults: When the voltage rises above this parameter, it will trigger the *Analog_19_OUT_OF_RANGE fault (flash code 0xBE, CAN Index 0x262C).* If assigning the Throttle to Analog 19, the resulting fault will become the *Throttle Input* fault *(flash code 0x42, CAN Index 0x2210)* if *Throttle_Source* = 19. If assigning the Brake to Analog 19, the resulting fault will become the *Brake Input* fault (Flash code 0x44) if *Brake_Source* = 19.
Fault Tolerance *Analog_Input_19_Fault_ Tolerance* `0x337B 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.
Analog 31 menu			
Voltage *Analog_Input_Volts_31* `0x3B56 0x00`	–327.68 – 327.67V *–32768 – 32767*	Read Only	Voltage at pin 26 (*as the +5V External Supply in Fig 13*).
Percent *Analog_Input_Percent_31* `0x3B6B 0x00`	0.0 – 100.0 % *0 – 1000*	Read Only	Voltage on a 0-100 percentage basis. The percentage of the voltage at the input pin based upon the High and Low settings, i.e., the percent of: [(analog_input_volts_31) – (analog_input_31_low)] / [(analog_input_31_high) – (analog_input_31_low)]
Low *Analog_Input_31_Low* `0x3305 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	The minimum input voltage before a fault is declared. This voltage represents the 0% point for the normalized inputs.
High *Analog_Input_31_High* `0x3304 0x00`	0.0 – 11.0V *0 – 1100*	7.0V	The maximum input voltage before a fault is declared. This voltage is the 100% point for the normalized inputs.
Fault Tolerance *Analog_Input_31_Fault_ Tolerance* `0x33B7 0x00`	0.0 – 11.0V *0 – 1100*	0.0V	Specifies the voltage threshold above the configured high limit or below the low limit, exceeding this results in an analog out of range fault. For example, if High = 9.6V, Low = 1V and Tolerance = 0.2V, an input voltage between 1V to 9.6V is mapped to 0–100%. Voltages above 9.8V (9.6 + 0.2) or below 0.8V (1V – 0.2V) trigger the fault.

CONTROLLER SETUP/IO ASSIGNMENTS — CONTROLS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Interlock Input Source *Interlock_Input_Source* `0x34B2 0x00`	0 – 32 *0 – 32*	5 **[PCF]**	Interlock switch assignment. Using the available switch inputs, select the digital input number that the interlock will use. The available switches depend upon the controller and application. For example, Switch 5 (pin 9) = INTERLOCK in Figure 13. Set this parameter to 0 to allow VCL control of interlock. To enable interlock with KSI turn-on, set the input source to switch 20 (the Keyswitch Input). Switch_20 = CAN Index 0x3339 0x00
Forward Input Source *Forward_Input_Source* `0x3342 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Forward switch assignment. Using the available switch inputs, select the digital input number that the forward input signal will use. The available switches depends upon the controller and application.
Reverse Input Source *Reverse_Input_Source* `0x3344 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Reverse switch assignment. Using the available switch inputs, select the digital input number that the reverse input signal will use. The available switches depends upon the controller and application.
Throttle Source *Throttle_Source* `0x3340 0x00`	0 – 32 *0 – 32*	1 **[PCF]**	Throttle assignment. Assigns which analog input is the wired-throttle (potentiometer or voltage). Default uses Analog 1. See (above): *Programmer » Controller Setup » Inputs » Analog 1 Type.* Note: A VCL Throttle is set under the *Application Setup » Throttle » VCL_Throttle_Enable* parameter. This frees up a switch input. See Figure 35 for more on the throttle setup options.
Brake Source *Brake_Source* `0x33D4 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Brake assignment. Assigns which analog input is the wired-brake (potentiometer or voltage). Figure 13 illustrates a 2-wire pot setup. Note: A VCL Brake is set under the *Application Setup » Brake » VCL_Brake_Enable* parameter. This frees-up a switch input.
Dual Steer Source *Dual_Steer_Source* `0x3A48 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	The dual steer assignment. For using a steer input, see the Dual Drive Operation manual supplement (doc #53231_FSeriesDD).
EMR State *EMR_State* `0x3490 0x00`	On-Off *0 – 1*	Read Only *Off*	Indicates if EMR is active.
EMR Switch Source NO *EMR_Input_Source* `0x3729 0x00`	0 – 32 *0 – 32*	0	Sets a normally open EMR (NO, Emergency Reverse) input. When both this and the redundant EMR NC (0x372F) parameter are unmapped, the EMR is settable by VCL. To supervise both inputs to verify they remain in complementary states, map both this NO and the NC (below) inputs parameters.
EMR Switch Source NC *EMR_Input_Source_Redundant* `0x372F 0x00`	0 – 32 *0 – 32*	0	Sets a normally closed EMR (NC, Emergency Reverse) input. When both this and the EMR NO (0x3729) parameter are unmapped, the EMR is settable by VCL. To supervise both inputs to verify they remain in complementary states, map both this NC and the NO (above) inputs parameters.

Quick Links:
Fig. 13 p.17
Fig. 35 p.164

CONTROLLER SETUP/IO ASSIGNMENTS — CONTROLS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Hydraulics Inhibit Type *Hydraulics_Inhibit_Type* `0x3702 0x00`	0 – 3 *0 – 3*	0	Prevents accidental turn-on of peripherals at startup/key-cycle. 0 - No inhibit 1 - Lift Only 2 - Lower only 3 - Both
Lift Input Type *Lift_Input_Type* `0x3C29 0x00`	0 – 3	0 **[PCF]**	Sets the type of lift input connected. 0 - NO Switch 1 - NC Switch 2 - Voltage Input. Note, set the respective min, max, map, offset, offset parameters within the Hydraulics Lift menu.
Lift Input Source *Lift_Input_Source* `0x372A 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Selects which input number to use as lift source. 0: VCL Throttle (range: 0-100) 1-32: Input Number Note, VCL may require further input mapping setup, see the Hydraulics Lift menu.
Lift Limit Switch Type *Lift_Limit_Type* `0x3C3F 0x00`	0 – 3 *0 – 3*	0 **[PCF]**	Sets the limit switch type. 0-Normally Open (NO) Switch 1-Normally Closed (NC) Switch 2-Voltage Input (Turns off Pump if input >50) Note, set the respective min, max, map, offset, offset parameters within the Hydraulics Lift menu.
Lift Limit Switch Source *Lift_Limit_Source* `0x3C3E 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Assign the Lift Input (switch) number.
Lower Input Type *Lower_Input_Type* `0x3C3A 0x00`	0 – 2 *0 – 2*	0 **[PCF]**	Sets the type of Lower input connected. 0-NO Switch 1-NC Switch 2-Voltage Input Note, set the respective min, max, map, offset, offset parameters within the Hydraulics Lower menu.
Lower Input Source *Lower_Input_Source* `0x372B 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Sets the input (switch) number to use as the Lower source. 0: VCL Throttle (range: 0-100) 1-32: Input Number Note, VCL may require further input mapping setup, see the Hydraulics Lift menu.

SWITCH ASSIGNMENT MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Interlock Input Source *Interlock_Input_Source* `0x34B2 0x00`	0 – 32 *0 – 32*	5 **[PCF]**	Interlock Switch Assignment. Select the digital input number that will be used for the interlock. The Interlock Input Source must be set to 0 to allow VCL control of interlock. KSI control can be accomplished by setting the input source to switch 20 (Keyswitch Input).
Interlock Input Source Redundant *Interlock_Input_Source_ Redundant* `0x3E6F 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Selects the redundant Interlock Switch input source.
Forward Input Source *Forward_Input_Source* `0x3342 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Forward switch assignment. Using the available switch inputs, select the digital input number that the forward input signal will use. The available switches depends upon the controller and application.
Reverse Input Source *Reverse_Input_Source* `0x3344 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Reverse switch assignment. Using the available switch inputs, select the digital input number that the reverse input signal will use. The available switches depends upon the controller and application.
EMR Switch Source NO *EMR_Input_Source* `0x3729 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Sets a normally open EMR input. When both this and the redundant EMR NC (0x372F) parameter are unmapped, the EMR is settable by VCL. When both are mapped, inputs are supervised to verify they remain in complementary states.
EMR Switch Source NC *EMR_Input_Source_Redundant* `0x372F 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Sets a normally closed EMR input. When both this and the EMR NO (0x3729) parameters are unmapped, the EMR is settable by VCL.
Lift Input Source *Lift_Input_Source* `0x372A 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Select which input # to use as Lift source. Requires a key cycle to take effect. Requires further input mapping setup. See Hydraulics menu. 0: VCL Throttle (range: 0-100) 1–32: Input Number.
Lower Input Source *Lower_Input_Source* `0x372B 0x00`	0 – 32 *0 – 32*	0 **[PCF]**	Select which input # to use as Lower source. Requires a key cycle to take effect. Requires further input mapping. See Hydraulics menu. 0: VCL Throttle (range: 0-100) 1–32: Input Number.
Tow Input Source *Tow_Input_Source* `0x33B8 0x00`	0 – 32 *0 – 32*	0 **[PCF]** **NOTICE**	Selects which input number to use as the Tow Mode. **Contact Curtis BEFORE using this option.** **Both ACIM and PMAC motors behave differently and require strict parameter settings to prevent controller damage.**

CONTROLLER SETUP/IO ASSIGNMENTS — SWITCH STATUS MENU

The read only switch inputs On/Off status include the inputs on any pins that can process a voltage reading. Use the on/off status in this menu during setup and application development to verify an "input" status. The 35-pin and 23-pin controllers have different pin number assignments. Reference the wiring diagrams and I/O tables for the AMPSEAL pin number aligned to the switches.

The On/Off (1/0) indicates the state of the binary input. The same value is encoded bitwise in the variable *Switches* (variable 0x3321) which contains the state of all 32 Switches as applicable to the controller. For the variable *Switches*, *Switch_1* corresponds to bit position 0 (LSB), and *Switch_32* corresponds to bit position 31 (MSB).

CONTROLLER SETUP/IO ASSIGNMENTS — SWITCH STATUS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION MONITORS THE CONTROLLER SWITCH STATUS
Switch 1 *Switch_1* `0x3324 0x00`	On/Off *On/Off*	Read Only	Pin 16 (35 pin controllers). Pin 11 (23 pin controllers).
Switch 2 *Switch_2* `0x3325 0x00`	On/Off *On/Off*	Read Only	Pin 8 (normally = Motor Temp sensor input) (35 pin controllers). Pin 9 (normally = Motor Temp sensor input) (23 pin controllers).
Switch 3 *Switch_3* `0x3326 0x00`	On/Off *On/Off*	Read Only	Pin 31 (normally = Encoder A input) (35 pin controllers). Pin 17 (normally = Encoder A input) (23 pin controllers).
Switch 4 *Switch_4* `0x3327 0x00`	On/Off *On/Off*	Read Only	Pin 32 (normally = Encoder B input) (35 pin controllers). Pin 18 (normally = Encoder B input) (23 pin controllers).
Switch 5 *Switch_5* `0x3328 0x00`	On/Off *On/Off*	Read Only	Pin 9 (35 pin controllers). Pin 8 (23 pin controllers).
Switch 6 *Switch_6* `0x3329 0x00`	On/Off *On/Off*	Read Only	Pin 15 (35 pin controllers). Pin 10 (23 pin controllers).
Switch 7 *Switch_7* `0x332A 0x00`	On/Off *On/Off*	Read Only	Pin 22 (35 pin controllers). Pin 14 (23 pin controllers).
Switch 8 *Switch_8* `0x332B 0x00`	On/Off *On/Off*	Read Only	Pin 33 (35 pin controllers). Pin 15 (23 pin controllers).
Switch 9 *Switch_9* `0x332C 0x00`	On/Off *On/Off*	Read Only	Pin 24 (35 pin controllers). Not available/defined for 23 pin controllers.
Switch 10 *Switch_10* `0x332D 0x00`	On/Off *On/Off*	Read Only	Pin 10 (35 pin controllers). Not available/defined for 23 pin controllers.
Switch 11 *Switch_11* `0x332E 0x00`	On/Off *On/Off*	Read Only	Pin 11 (35 pin controllers). Pin 19 (23 pin controllers).
Switch 12 *Switch_12* `0x332F 0x00`	On/Off *On/Off*	Read Only	Pin 12 (35 pin controllers). Pin 21 (23 pin controllers).

CONTROLLER SETUP/IO ASSIGNMENTS — SWITCH STATUS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION MONITORS THE CONTROLLER SWITCH STATUS
Switch 13 *Switch_13* `0x3330 0x00`	On/Off *On/Off*	Read Only	Pin 14 (35 pin controllers). Pin 22 (23 pin controllers).
Switch 14 *Switch_14* `0x3331 0x00`	On/Off *On/Off*	Read Only	Pin 25 (35 pin controllers). Pin 23 (23 pin controllers).
Switch 18 *Switch_18* `0x333D 0x00`	On/Off *On/Off*	Read Only	Pin 17 (35 pin controllers). Not available/defined for 23 pin controllers.
Switch 19 *Switch_19* `0x333E 0x00`	On/Off *On/Off*	Read Only	Pin 27 (35 pin controllers). Not available/defined for 23 pin controllers.
Switch 20 *Switch_20* `0x3339 0x00`	On/Off *On/Off*	Read Only	Pin 1 (the keyswitch, always = On when powered). 35 and 23 pin controllers.
Switch 21 *Switch_21* `0x3332 0x00`	On/Off *On/Off*	Read Only	Pin 2 (when the driver is used as a switch input. 35 pin controllers). Pin 7 (when the driver-1 is used as a switch input 23-pin controllers).
Switch 22 *Switch_22* `0x3333 0x00`	On/Off *On/Off*	Read Only	Pin 5 (when the driver is used as a switch input 35 pin). Pin 4 (when the driver-2 is used as a switch input 23-pin controllers).
Switch 23 *Switch_23* `0x3334 0x00`	On/Off *On/Off*	Read Only	Pin 4 (when the driver is used as a switch input 35 pin). Pin 5 (when the driver-3 is used as a switch input 23-pin controllers).
Switch 24 *Switch_24* `0x3335 0x00`	On/Off *On/Off*	Read Only	Pin 3 (when the driver is used as a switch input 35 pin). Pin 6 (when the driver-4 is used as a switch input 23-pin controllers).
Switch 25 *Switch_25* `0x3336 0x00`	On/Off *On/Off*	Read Only	Pin 6 (when the driver is used as a switch input 35 pin). Pin 3 (when the driver-5 is used as a switch input 23-pin controllers).
Switch 26 *Switch_26* `0x3337 0x00`	On/Off *On/Off*	Read Only	Pin 19 (when the driver is used as a switch input. 35 pin controllers). Not available/defined for 23 pin controllers.
Switch 27 *Switch_27* `0x3338 0x00`	On/Off *On/Off*	Read Only	Pin 20 (when the driver is used as a switch input. 35 pin). Not available/defined for 23 pin controllers.

CONTROLLER SETUP/IO ASSIGNMENTS — COIL DRIVERS MENU

These are the standard coil-driver assignments using Drivers 1-5. A selection of "0" indicates no driver is assigned. Note that the EM Brake driver, as shown in Figures 12–15, is a 3-amp driver. The other PWM drivers are 2 amperes. All drivers are connected to the coil return which has the inductive coil suppression. Using the coil drivers for RC or resistive loads can cause an overcurrent fault due to in-rush/peak current over the driver's intended usage.

Drivers 6 and 7 are "digital drivers" available as 1 Amp 100% On low-side drivers.

CONTROLLER SETUP/IO ASSIGNMENTS — COIL DRIVERS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Main Contactor Driver *Main_Contactor_Driver* 0x34D2 0x00	0 – 5 *0 – 5*	3 **[PCF]**	Main Contactor Driver assignment. Changes require a key cycle to take effect. Same default (Driver-3) for both 35 and 23 pin controllers.
Em Brake Driver *EM_Brake_Driver* 0x3480 0x00	0 – 5 *0 – 5*	0 **[PCF]**	EM Brake Driver assignment. Changes require a key cycle to take effect. Use Driver-2 (3 amps) on both the 35 and 23 pin controllers.
Pump Contactor Driver *Pump_Contactor_Driver* 0x372C 0x00	0 – 5 *0 – 5*	0 **[PCF]**	DC Pump (i.e., hydraulic) Driver assignment. Changes require a key cycle to take effect. This driver will close a contactor to fully "turn on" a DC hydraulic pump motor.
Load Hold Driver *Load_Hold_Driver* 0x3C41 0x00	0 – 5 *0 – 5*	0 **[PCF]**	Driver assignment for the load-hold valve. Changes require a key cycle to take effect. Neither the 35 nor 23 pin controllers have this as an assigned (default) driver.
Lower Driver *Lower_Driver* 0x4FCC 0x00	0 – 7 *0 – 7*	0 **[PCF]**	Driver assignment for the lower valve. Use the proportional valve (Driver-1) as the lowering valve on both the 35-pin and 23-pin contollers. Changes require a key cycle to take effect.

CONTROLLER SETUP — OUTPUTS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Driver Output Frequency *Driver_Output_Frequency* 0x3422 0x00	500 – 2000 Hz *500 – 2000*	500 Hz	Controls the frequency of the PWM on drivers 2-5.
Coil Supply Start Up Checks *Coil_Supply_Startup_Check* 0x34D3 0x00	Off/*On* *On/Off*	On	On: The controller performs a verification check of the internal KSI-to-Coil Return circuit at startup if the hardware supports it. This involves switching the coil-supply on/off at startup. If the check fails, it triggers the COIL_SUPPLY_FAULT (flash code 10-9, 0xA9). Off: To prevent (stop) the startup verification check.
Driver 1			
Driver 1 Checks Enable *Driver_1_Checks* 0x341B 0x00	Enumeration 0 – 2 Enumeration 0 = Checks off 1 = Checks On 2 = Safety Designated	Checks Off	The Checks Enable parameter enables the driver and coil fault detection. When set to On, this parameter checks the associated driver wiring and driver load to verify that the driver correctly drives the load both high and low. The checks will occur regardless of the PWM output of the driver. The check detects both open and shorted conditions. When a fault occurs, the controller opens the driver and issues the DRIVER_1_fault. Turn this parameter off when not using a particular driver. If unused (open pin), set this Checks Enable parameter to Off. Note: Short circuit protection is always active at these five drivers regardless of how Checks Enable parameter is set. If a driver is safety designated, it is treated the same way as if it was set to (1 – ON) except it will have an additional check for if the driver is stuck-on which would result in the disabling of the coil supply.

Quick Links:
Table 11 p.25
Fig. 12 p.16
Fig. 13 p.17

CONTROLLER SETUP — OUTPUTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Driver 2–7			
Driver X Checks Enable *Driver_X_Checks* **Driver_2_Checks :** 0x341C 0x00 **Driver_3_Checks :** 0x341D 0x00 **Driver_4_Checks :** 0x341E 0x00 **Driver_5_Checks :** 0x341F 0x00 **Driver_6_Checks :** 0x3420 0x00 **Driver_7_Checks :** 0x3421 0x00	Enumeration 0 – 2	Checks Off	See Driver 1. X = 2–7. Note: Drivers 6 and 7 do not apply to the 23-pin controllers.

CONTROLLER SETUP — EXTERNAL SUPPLIES MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
5V Output Enable *Ext_5V_Output_Enable* 0x36A8 0x00	On/Off *On/Off*	On	Enables the 5V Power Supply Output.
12V Output Enable *Ext_12V_Output_Enable* 0x36A7 0x00	On/Off *On/Off*	Off	Enables the 12V Power Supply Output.
5V Measured *External_5V_Supply* 0x36AA 0x00	0.00 – 100.00V *0 – 10000*	Read Only	Voltage at the +5V output.
5V Supplied Current *Ext_5V_Current* 0x36AE 0x00	0 – 200 mA *0 – 200*	Read Only	Current (mA) at the +5V output. The limit is based upon the controller model.
12V Measured *External_12V_Supply* 0x36AB 0x00	0.00 – 100.00V *0 – 10000*	Read Only	Voltage at the +12V output.
12V Supplied Current *Ext_12V_Current* 0x36AF 0x00	0 – 200 mA *0 – 200*	Read Only	Current (mA) at the +12V output. The limit is based upon the controller model.
Total Supplied Current *Ext_Supply_Current_mA* 0x36A9 0x00	0 – 200 mA *0 – 200*	Read Only	Total current from both the 5V and 12V supplies. Maximum is 200 mA.
Encoder Power Source *Encoder_Power_Source* 0x36A0 0x00	Enumerated *0 - 2* 0 = None 1 = 5V 2 = 12V	5V	Sets the Power Supply Source to the 5 or 12 volts for the Encoder. Use this parameter to determine which supply voltage, if any, is used to issue an encoder fault if the supply itself faults.
5V Supply Min *Ext_5v_Supply_Min* 0x36A4 0x00	0 – 200 mA	0 mA	Sets the lower threshold of the current of the 5V external supply. Current below this threshold will generate a fault (*Ext_5V_Supply_Failure_Active* 0x2531) that can be read by VCL.
5V Supply Max *Ext_5v_Supply_Max* 0x36A3 0x00	0 – 100 mA	100 mA	Sets the upper threshold of the current of the 5V external supply. Current above this threshold will generate a fault (*Ext_5V_Supply_Failure_Active* 0x2531) that can be read by VCL.
12V Supply Min *Ext_12v_Supply_Min* 0x36A6 0x00	0 – 200 mA	0 mA	Sets the lower threshold of the current of the 12V external supply. Current below this threshold will generate a fault (*Ext_12V_Supply_Failure_Active* 0x2532) that can be read by VCL.
12V Supply Max *Ext_12v_Supply_Max* 0x36A5 0x00	0 – 100 mA	100 mA	Sets the upper threshold of the current of the 12V external supply. Current above this threshold will generate a fault (*Ext_12V_Supply_Failure_Active* 0x2532) that can be read by VCL.

CONTROLLER SETUP — CURRENT LIMITS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Drive Current Limit *Drive_Current_Limit* `0x3441 0x00`	0 – 100 % *0 – 32767*	100 %	Sets the maximum RMS current the controller will supply to the motor during drive operation as a percentage of the controller's full rated current. Note: Reducing this value will reduce the maximum drive torque.
Regen Current Limit *Regen_Current_Limit* `0x3445 0x00`	5 – 100 % *0 – 32767*	100 %	Sets the maximum RMS regen current as a percentage of the controller's full rated current. Note: The regen current limit applies during neutral braking, direction reversal braking, and speed limiting when traveling downhill.
Brake Current Limit *Brake_Current_Limit* `0x343C 0x00`	5 – 100 % *1638 – 32767*	100 %	Sets the maximum RMS regen current during braking. The brake command is the percentage of the controller's full rated current (100% is maximum current). Note: Typically, the brake current limit is set equal to the regen current limit. The brake current limit overrides the regen current limit when the brake input is active.
EMR Current Limit *EMR_Current_Limit* `0x3443 0x00`	5 – 100 % *1638 – 33767*	100 %	Sets the maximum RMS current allowed for braking and driving when in emergency reverse. The emergency reverse current limit is a percentage of the controller's full rated current.
Interlock Brake Current Limit *Interlock_Brake_Current_Limit* `0x3444 0x00`	5 – 100 % *1638 – 32767*	100 %	Sets the maximum RMS regen current during interlock braking as a percentage of the controller's full rated current.

*Note: Parameters **Pump Drive Current Limit** (0x343D) and **Pump Regen Current Limit** (0x343E) apply to the F2-C controllers. See manuals 53228_ACF2-C.*

CONTROLLER SETUP/CURRENT LIMITS — POWER LIMITS MENU

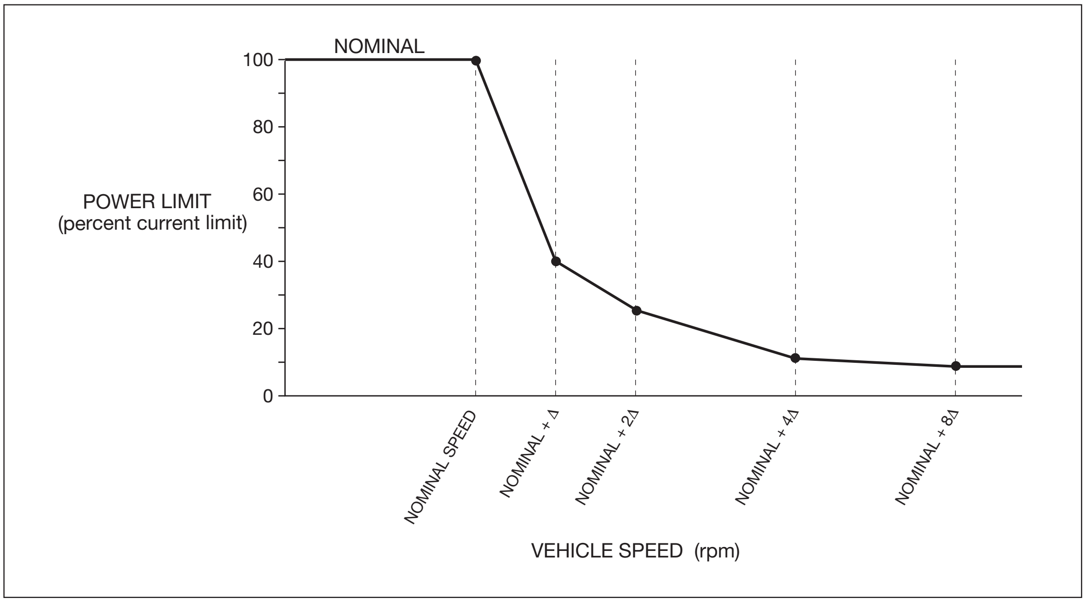
PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Nominal Speed *PL_Nominal_Speed* `0x344E 0x00`	100 – 24000 rpm *100 – 24000*	1500 rpm	Sets the base speed that the drive limiting map and regen limiting map use.
Delta Speed *PL_Delta_Speed* `0x3448 0x00`	10 – 24000 rpm *10 – 24000*	500 rpm	Sets the width of the delta increment that the drive limiting map and regen limiting map use.

CONTROLLER SETUP/CURRENT LIMITS/POWER LIMITS — DRIVE LIMITING MAP MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Nominal *PL_Drive_Nominal* `0x3449 0x00`	0 – 100 % *0 – 32767*	100 %	These parameters define the percentage of the applied drive current limit at the speeds defined by the nominal speed and delta speed parameters. The resulting map allows the controller to reduce the drive current as a function of speed. Reducing the power requirements at certain speeds restricts performance. This can be useful for reducing motor heating. This is useful to keep consistent vehicle power with the changing battery state-of-charge.
Plus Delta *PL_Drive_Nominal_Plus_Delta* `0x344D 0x00`	0 – 100 % *0 – 32767*	100 %	
Plus 2xDelta *PL_Drive_Nominal_ Plus_2xDelta* `0x344A 0x00`	0 – 100 % *0 – 32767*	100 %	
Plus 4xDelta *PL_Drive_Nominal_ Plus_4xDelta* `0x344B 0x00`	0 – 100 % *0 – 32767*	100 %	
Plus 8xDelta *PL_Drive_Nominal_ Plus_8xDelta* `0x344C 0x00`	0 – 100 % *0 – 32767*	100 %	

Figure 33 illustrates a typical Drive Current Limit mapping.

Figure 33
Drive Current Limiting Map

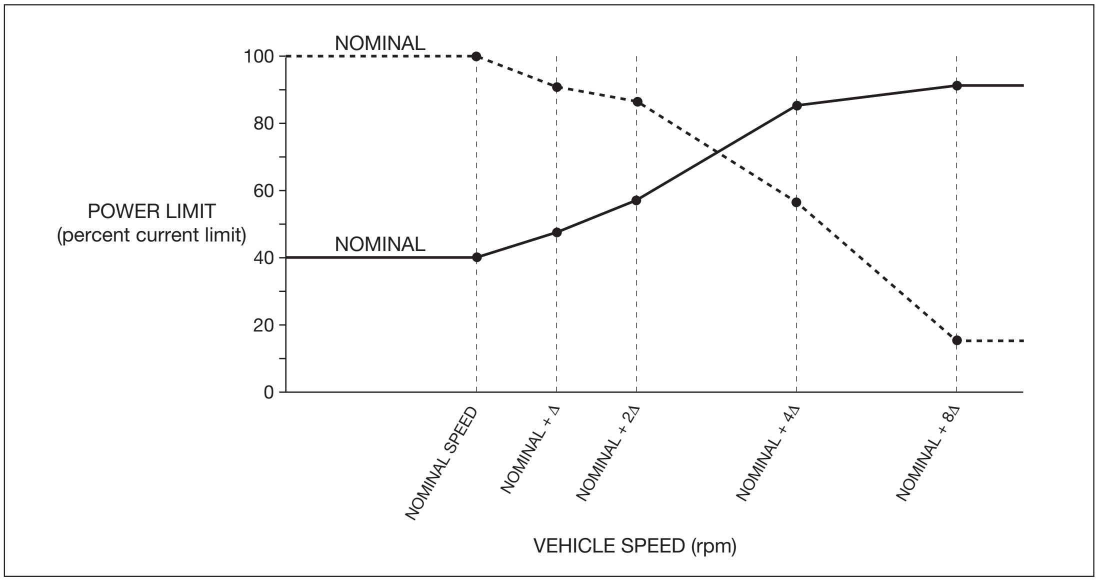


CONTROLLER SETUP/CURRENT LIMITS/POWER LIMITS — REGEN LIMITING MAP MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Nominal *PL_Regen_Nominal* `0x344F 0x00`	0 – 100 % *0 – 32767*	100 %	These parameters define the percentage of the applied regen current limit or braking current limit at the speeds defined by the nominal speed and delta speed parameters. Use these parameters to shape the curve for limiting the available torque at various speeds. One possible use is to compensate for the torque-speed characteristic of the motor.
Plus Delta *PL_Regen_Nominal_Plus_Delta* `0x3453 0x00`	0 – 100 % *0 – 32767*	100 %	
Plus 2xDelta *PL_Regen_Nominal_ Plus_2xDelta* `0x3450 0x00`	0 – 100 % *0 – 32767*	100 %	
Plus 4xDelta *PL_Regen_Nominal_ Plus_4xDelta* `0x3451 0x00`	0 – 100 % *0 – 32767*	100 %	
Plus 8xDelta *PL_Regen_Nominal_ Plus_8xDelta* `0x3452 0x00`	0 – 100 % *0 – 32767*	100 %	

Figure 34 illustrates two Regen Current Limit maps.

Figure 34
Regen Current Limiting Map



MOTOR SETUP

When selecting either the ACIM or PMAC via Motor Technology parameter, the Motor Setup menus will show the selected motor's parameters, hiding the non-selected motor's parameters. ACIM motors use the quadrature encoder. See the Motor Setup Index for a side-by-side comparison of the ACIM and PMAC parameter menus.

Quick Link:
Motor Setup Index p. 47-48

MOTOR SETUP MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Max Speed Controller Limit *Max-Speed_Controller_Limit* `0x30AF 0x00`	0 – 24000 rpm *0 – 24000*	Read Only	Displays the maximum allowed speed that clamps the upper limit of the Max Speed parameters. Most material-handling vehicle applications will not have rpm speeds beyond the typical 8-10,000 rpm, and often much lower rpm as they travel at low warehouse speeds.
Typical Max Speed *Typical_Max_Speed* `0x3542 0x00`	500 – 24000 rpm *500 – 24000*	4000 rpm	Set this parameter to the typical maximum motor speed of the vehicle. This value does not need to be set precisely; an estimate will do. This setting normalizes all the vehicle response rates to this value (Typical Max Speed rpm). For example, suppose the Typical_Max_Speed is fixed at 6000 rpm and the *Full_Accel_Rate_LS_SpdM* = 3.0 seconds, and *Full_Accel_Rate_HS_SpdM* = 3.0 seconds: If *Max_Speed_SpdM* = 6000 rpm it will take 3.0 seconds to accelerate from zero to top speed (6000 rpm). If *Max_Speed_SpdM* = 3000 rpm it will take 1.5 seconds to accelerate from zero to top speed (3000 rpm). If *Max_Speed_SpdM* = 1000 rpm it will take 0.5 seconds to accelerate from zero to top speed (1000 rpm).
Motor Technology *Motor_Technology* `0x3534 0x00`	0 – 1 *0 – 1*	0	Set this parameter to the type of motor in your vehicle: 0 = ACIM (induction motor) 1 = PMAC (permanent magnet synchronous [AC] motors). Note: A Parameter Mismatch Fault will result unless the Feedback Type matches the Motor Technology setting: If Motor Technology = 0 (ACIM) then Feedback Type must = 1 (quadrature encoder) or 2 (sin/cos encoder). If Motor Technology = 1 (PMAC) Feedback Type must = 2 (sin/cos device).
Position Sensor Type *Feedback_Type* `0x3520 0x00`	Enumeration *1 – 2*	quadrature	Set this parameter to the type of position feedback device in your vehicle: 1 = quadrature encoder. 2 = sin/cos encoder. 3 = internal steering (n/a). 4 = Hall switch (*contact Curtis*). 5 = PMDC open-loop/sensorless (*contact Curtis*). Note: A Parameter Mismatch Fault will result unless the Position Sensor Type matches the Motor Technology setting: If Motor Technology = 0 (ACIM) then Position Sensor Type must = 1 (quadrature encoder) or 2 (sin/cos encoder). If Motor Technology = 1 (PMAC) Position Sensor Type must = 2 (sin/cos encoder) or = 4 (Hall switch).

MOTOR SETUP MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Swap Motor Direction *Swap_Motor_Direction* `0x362F 0x00` Note: In previous FOS versions this parameter was called: **Swap Two Phases** *Swap_Two_Phases* `0x362F 0x00`	Off/On *0 – 1*	Off **NOTICE**	Swaps the mechanical direction of motor spinning. If forward throttle produces reverse direction change this parameter. **This parameter is critical for the emergency reverse to work properly.**

MOTOR SETUP — INDUCTION MOTOR (ACIM) MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Motor Type *IM_Motor_Type* `0x3635 0x00`	–1 – 543 *–1 – 543*	0 **[PCF]**	This parameter references a predefined table of motor parameters for many AC motors. Consult your local Curtis customer support engineer for information on how to set this parameter based on your application and motor. If Motor Type = –1, setting Test Enable and Test Throttle will run full autocharacterization.If Motor Type = 0, MotorData values must be manually entered by a Curtis Applications Engineer. Changing this parameter always results in a Parameter Change Fault (fault code 49). Clear it by cycling the keyswitch (KSI cycle). The Parameter Change Fault protects the controller and the operator. For questions, or to verify the application's motor number, consult the local or regional Curtis office/support engineer.

MOTOR SETUP/INDUCTION MOTOR (ACIM) — FIELD WEAKENING MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Base Speed Captured *Base_Speed_Captured* `0x352B 0x00`	0 – 65535 rpm *0 – 65535*	Read Only	This variable is the value of the motor's base speed captured in the most recent acceleration. Use this value for setting the FW Base Speed parameter when using the FW Base Speed procedure or the ACIM motor characterization procedure.
FW Base Speed *FW_Base_Speed* `0x36C8 0x00`	20 – 24000 rpm *20 – 24000*	20 rpm	Sets the speed at which modulation depth has reached 100% (all available voltage is used) and where field weakening begins. The scaling of this parameter is by an internal motor characterization procedure, so the setting of the FW Base Speed should always use the tuning test, which will take into account the scaling factor. Do not enter a speed observed from a torque vs. Speed plot, as this number fails to take into account the internal scaling factor. **Set this parameter during the initial setup. Reset it each time the Motor Type or the low speed current limit is changed.** For example if lowering the *Drive_Current_Limit* or *PL_Drive_Nominal* parameters, always reset this parameter. To determine the correct value (see the variable **Base Speed Captured** in the Characterization tests, below), perform the FW Base Speed ("base speed") test (for traction systems or for hydraulic systems depending on your application) with traction-batteries that have a reasonable charge.
Test Field Current *Test_Field_Current* `0x3092 0x00`	0.0 – 800.0A *0 – 8000*	Read Only	Use this reading during the ACIM motor characterization procedure.

MOTOR SETUP/INDUCTION MOTOR (ACIM) — FIELD WEAKENING MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Field Weakening Drive *Field_Weakening_Drive* `0x36C1 0x00`	0 – 100 % *0 – 1024*	100 %	Sets the amount of field weakening allowed while driving the motor. Field Weakening Drive will affect efficiency and torque only at speeds above the programmed FW Base Speed. Low settings will result in better efficiency but less torque; higher settings will result in more torque and less efficiency. A setting of zero will disable field weakening thus resulting in the highest efficiency and lowest torque above base speed. To keep motor and controller heating to a minimum (high efficiency), Field Weakening Drive should be set just high enough to meet the high-speed performance specifications. (Typical, high-speed performance specifications are maximum drive speed with full load or maximum lift speed with full load.) Use the drive current limiting map to restrict performance at all speeds. The maximum setting of the Field Weakening Drive parameter depends on the type (model) of motor in the characterization test. If the ACIM motor was dyno characterized (i.e. sent to the Curtis factory for characterization on the motor dyno) Field Weakening Drive can be set anywhere in the range of 0% (lowest torque highest efficiency) to 100% (highest torque lowest efficiency). If the ACIM motor was auto characterized, the results of the Field Weakening Test restricts the range. For these motors Field Weakening Drive can be set anywhere in the range of 0% (lowest torque highest efficiency) to Max Field Weakening Drive (highest torque lowest efficiency) result. For auto characterized motors, setting Field Weakening Drive greater than the maximum setting found in the Field Weakening Test will result in poorer efficiency and less torque as the motor will be operating "over the slip curve". Note: The Field Weakening Drive setting will have no effect at motor speeds below FW Base Speed.
Weakening Rate Drive *Field_Weakening_Rate_Drive* `0x36C2 0x00`	0 – 100 % *0 – 500*	5 %	Sets the control loop gains for field weakening. Setting the rate too low may create surging in the vehicle as it accelerates at mid to high speeds. Setting the rate too high may create high frequency oscillations (usually audible) when the vehicle accelerates at mid to high speeds.
Min Field Current *Min_Field_Current* `0x30A3 0x00`	0 – 800A *0 – 8000*	0 Amps	This current pre-fluxes the motor and can improve initial take-off at the expense of some battery consumption. Set this parameter only when there is a need to increase the initial take-off (vehicle launch) without motor torque (e.g., creep torque at idle).

MOTOR SETUP/INDUCTION MOTOR (ACIM) — LIMITED OPERATING STRATEGY (LOS) MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
LOS Upon Encoder Fault *LOS_Upon_Encoder_Fault* `0x3630 0x00`	On/Off *On/Off*	On *(1)*	The Limited Operating Strategy (LOS) is typically to drive the vehicle back to a repair center at very low speeds in the event the motor encoder fails. Following an encoder failure (either Encoder Fault (fault code 36) or Stall Detected (fault code 73)) and after the Interlock is cycled the vehicle enters LOS mode, thus allowing drive without motor encoder feedback. In LOS mode, the ability to achieve maximum torque even for a very short time is considered more important than smoothness. When LOS mode is entered the Encoder LOS fault (fault code 9-3) becomes active and the encoder fault (either Encoder Fault (fault code 3-6) or Stall Detected (fault code 7-3)) is cleared. When this parameter setting is On, the LOS mode is entered in the event of an encoder fault followed by an Interlock cycle. When programmed Off in the event of an encoder fault the encoder fault remains and drive is disabled.
LOS Max Speed *Enc_LOS_Max_Speed* `0x362A 0x00`	100 – 24000 rpm *100 – 24000*	800 rpm	This parameter indirectly defines the maximum speed for LOS mode by setting the maximum frequency that corresponds to LOS Max Speed. In LOS mode the throttle commands a frequency that is interpolated linearly between zero (at Throttle Command = 0%) and the programmed LOS Max Speed (at Throttle Command = 100%).
LOS Max Current *Enc_LOS_Max_Current* `0x3626 0x00`	100 – 650A *1000 – 6500*	400A	In LOS mode, a partial or full throttle command will result in the maximum current set by this parameter. The controller's rated current clamps this parameter's value.
LOS Max Mod Depth *Enc_LOS_Max_Mod_Depth* `0x3628 0x00`	15 – 100 % *177 – 1182*	50 %	In LOS mode, the maximum modulation depth acts to limit the current at higher speeds. Set this parameter such that the modulation depth limit is reached prior to the LOS Max Speed limit so that the motor current will fall off from LOS Max Current at higher speeds. This may allow the vehicle to drive longer in LOS mode as it lessens the chance of the motor or controller overheating.
LOS Accel Rate *Enc_LOS_Accel_Rate* `0x3620 0x00`	2.0 – 15.0 s *2000 – 15000*	7.0 sec	Defines the rate (in seconds/Typical Max Speed) at which the frequency increases when full throttle is applied while operating in LOS mode. This parameter should be set to a slow rate (high parameter value) so the frequency command has a very slow slew rate to ensure that the max torque point is hit for a reasonable period of time; this decreases the probability of going over the slip curve and allows ramps or obstacles to be overcome.
LOS Decel Rate *Enc_LOS_Decel_Rate* `0x3624 0x00`	2.0 – 15.0 s *2000 – 15000*	3.0 sec	Defines the rate (in seconds/Typical Max Speed) at which the frequency decreases when throttle is released while operating in LOS mode.

MOTOR SETUP/INDUCTION MOTOR (ACIM) — CHARACTERIZATION TESTS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Test Enable *IM_AutoChar_Test_Enable* `0x324F 0x00`	Off/On *0 – 1*	Off	Enables ACIM (induction motor) characterization with quadrature encoder to begin. Notice: Only assert (*change to = On*) when the test is ready to begin.
Test Throttle *IM_Test_Throttle* `0x3241 0x00`	–1 – 1 *–1 – 1*	0	Begins characterization if Test Enable is asserted. IMPORTANT: If the motor starts to turn in the positive (forward) vehicle speed direction, set the Test Throttle to 1. If it turns in the negative (reverse) direction, set the Test Throttle to -1.
Motor Poles *IM_Motor_Poles* `0x4F5D 0x00`	2 – 8 *1 – 4*	4 **[PCF]**	The number of motor pole *pairs*. This parameter selects the number of poles before beginning the ACIM motor characterization procedure. See Chapter 6 for the complete description of the ACIM motor characterization procedure, which shows how to use this parameter. Changing this parameter always results in a Parameter Change Fault (fault code 49). Clear it by cycling the keyswitch (KSI cycle). The PCF protects the controller and the operator.
Max Test Speed *IM_AutoChar_Max_Test_Speed* `0x324B 0x00`	500 – 24000 rpm *500 – 24000*	2000 rpm	This parameter sets the maximum motor speed allowed during ACIM motor characterization.
Max Test Current *IM_AutoChar_Max_Test_ Current* `0x3245 0x00`	10 – 100 % *3277 – 32767*	70 %	This parameter sets the maximum motor current allowed during ACIM motor characterization.
SlipGain *IM_MotorData* `0x364A 0x00`	0.10 – 1000.00 *0.0001 – 1*	3.20	Adjust this parameter during ACIM motor characterization to tune the motor so it delivers the maximum torque per amp. See the autocharacterization test description of the ACIM Motor Characterization procedure for information on how to use this parameter.
Current Reg Tuning Test Enable *IM_CR_Tuning_Test_Enable* `0x3259 0x00`	Off/On *0 – 1*	Off	This parameter only performs the current-regulator tuning portion of the autocharacterization test. 0 = Off 1 = On This parameter is typically applicable for previously auto-characterized (Type 0) induction motors. It performs only the current-regulation tuning portion of the motor characterization on existing Type 0 motors. To begin, clear any existing faults, then set Interlock = On and this parameter = 1. Similarly to the auto-characterization routine steps in Chapter 6, then set Test Enable = 1 followed by setting Test Throttle = 1. The controller will run the tuning test and issue a Parameter Change Fault when finished. If a new (or existing Type 0) motor is auto-characterized following the steps in Chapter 6, do not use this parameter, as it is automatically included in the full auto-characterization routine. Consult your Curtis distributor or support engineer for further assistance based on your motor and its application.

NOTICE

Treat these parameters as READ ONLY commissioning RESULTS. The values are generated (automatically) during ACIM commissioning. After successfully commissioning the ACIM motor, an OEM may copy these values to another matching controller/system to duplicate the motor performance in another vehicle. Copy all values exactly as commissioned.

Contact Curtis before making any changes, or if there are any questions.

MOTOR SETUP/INDUCTION MOTOR (ACIM) — CURRENT REGULATOR MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Tuning Voltage *Tuning_Nominal_Pack_Volts* `0x366B 0x00`	*Controller model basis* *Auto generated by the commissioning process*	**RESULTS** **READ ONLY**	The voltage when the current regulator was tuned. Used to scale gains AND base speed. Base speed MUST be tuned at the same voltage as the current regulator.
Kp d Axis Current *Kp_1_Current* `0x3667 0x00`	*Controller model basis* *Auto generated by the commissioning process*	**RESULTS** **READ ONLY**	The current regulator proportional (Kp) gain for one axis.
Ki d Axis Current *Ki_1_Current* `0x3666 0x00`	*Controller model basis* *Auto generated by the commissioning process*	**RESULTS** **READ ONLY**	The current regulator integral (Ki) gain for one axis.
Kp q Axis Current *Kp_2_Current* `0x306A 0x00`	*Controller model basis* *Auto generated by the commissioning process*	**RESULTS** **READ ONLY**	The current regulator proportional (Kp) gain for one axis.
Ki q Axis Current *Ki_2_Current* `0x306B 0x00`	*Controller model basis* *Auto generated by the commissioning process*	**RESULTS** **READ ONLY**	The current regulator integral (Ki) gain for one axis.

MOTOR SETUP — MOTOR SETUP STATUS MENU

Treat these parameters as read only, as outputs to the motor commissioning. To protect the controller, change them only when authorized.

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Current Regulator Setup *Current_Regulator_Setup* `0x326B 0x00`	Off - On *0 – 1*	Off Read Only	A flag indicating that the current regulator test has been run.
Slip Gain Setup *Slip_Gain_Setup* `0x326C 0x00`	Off - On *0 – 1*	Off Read Only	A flag indicating that the slip gain test has been run.
Base Speed Setup *Base_Speed_Setup* `0x326D 0x00`	Off - On *0 – 1*	Off Read Only	A flag indicating that the base speed test has been run.
Automated Test Run *Automated_Test_Setup* `0x326E 0x00`	Off - On *0 – 1*	Off Read Only	A flag indicating that the motor specific automated test has been run.

MOTOR SETUP — QUADRATURE ENCODER MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
5V Output Enable *Ext_5V_Output_Enable* `0x36A8 0x00`	On – Off *0 – 1*	On	Enables the 5V Power Supply Output.
12V Output Enable *Ext_12V_Output_Enable* `0x36A7 0x00`	On – Off *0 – 1*	Off	Enables the 12V Power Supply Output.
Encoder Steps *Encoder_Steps* `0x34E7 0x00`	32 – 4096 *32 – 4096*	64 **[PCF]**	Sets the number of encoder pulses per revolution. This must be set to match the encoder; see the motor nameplate. Note: Do not change this parameter while the controller is powering the motor. Any time this parameter is changed a Parameter Change Fault (fault code 49) is set and must be cleared by cycling power; this protects the controller and the operator. Adjusting this parameter can be hazardous; setting it improperly may cause vehicle malfunction including un-commanded drive.
Phasing Order *Phasing_Order* `0x34EA 0x00`	Off – On *0 – 1*	Off ⚠ **WARNING**	Inverts the direction of feedback relative to control which can result from phasing and encoder wiring swaps. Changing this parameter in PMAC requires commissioning, do not set manually. **Changing this parameter can cause uncommanded motion. Care should be taken when changing this parameter.**
Pullup Override *Pullup_Override* `0x32D5 0x00`	0 – 6 *0 – 6*	0	Set this to override the state of the controller's internal pullups as follows: 0 = Not overridden (use feedback type default). 1 = Encoder primary pullups on. 2 = Encoder primary pullups off. 3 = Encoder index pullup on. 4 = Encoder index pullup off. 5 = Both pullups on. 6 = Both pullups off. When using external pullups (as must be done with high frequency encoder inputs, both pullups must be turned off.

MOTOR SETUP/QUADRATURE ENCODER — ENCODER FAULT SETUP MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Fault Detection Enable *Encoder_Fault_Detection_Enable* 0x34E3 0x00	On/Off *On/Off*	On	Setting this parameter to On enables the encoder fault checking. The three fault conditions checked are: Encoder Fault (fault code 36). Stall Detected (fault code 73). Encoder Pulse Error (fault code 88).
Encoder Pulse Fault Detect Time *Encoder_Pulse_Fault_Detect_Time* 0x34E5 0x00	0.0 – 3.0 s *0 – 94*	0.5 sec	Defines the minimum time it takes for the controller, while the vehicle is in motion, to detect that the encoder and the Encoder Steps parameter do not match. When the Encoder Steps setup is incorrect, the motor controller cannot properly calculate the AC motor field orientation. The loss of field orientation can cause the motor to spin up toward full speed once any throttle is applied. Applying the throttle and then releasing it, while the drive current is significant and the motor is still accelerating, sets this timer. Note that the motor can spin to high rpms for several seconds before conditions allow the controller to detect a fault. The Encoder Pulse Error (fault code 88) is set when this fault occurs. Setting the parameter to zero will disable this fault detection.
Fault Stall Time *Enc_Fault_Stall_Time* 0x34E2 0x00	0 – 10 s *0 – 5000*	5 sec	Sets a timer when no motor encoder movement is detected. If no motor movement (encoder pulse) is detected for the programmed Fault Stall Time with high current applied a Stall Detected fault is issued (fault code 73). For example, when driving up a ramp and the vehicle 'stalls' because it is under-powered or overloaded (driver applying full throttle), or if driven into an immovable object, these types of conditions will set this fault.

MOTOR SETUP/QUADRATURE ENCODER — SPEED FILTER MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Speed Filter Frequency *Speed_Filter_Frequency* 0x3224 0x00	0.1 – 250 Hz *0.1 – 250*	30.0 Hz	Cutoff frequency for speed filtering. Note, this value is now in hertz, when updating from old versions you may need to divide by 2pi (3.14159) to get the exact performance.

MOTOR SETUP/PMAC (PERMANENT MAGNET MOTOR) — MOTOR TYPE MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Motor Type *PMAC_Motor_Type* `0x3669 0x00`	0 – 137 *0 – 137*	0	This parameter references a predefined table of motor parameters for many PMAC motors. Consult the local Curtis customer support engineer for information on how to set this parameter based on the application and motor.
Override Short Circuit Current Protection *PMAC_Override_Short_Circuit_ Current_Protection* `0x31FD 0x00` **NOTICE The controller warranty may be refused if the failure is determined to be a mismatch between motor and controller.**	Off/On *0 – 1*	Off	The purpose of this function is to protect the controller in the event of a severe fault requiring the motor control to be shut down. If a severe fault occurs, and the voltage generated by the motor exceeds the maximum voltage rating of the controller, the controller will short circuit the motor to bring the voltage down to an acceptable level. The controller will check the short circuit current for the selected PMAC motor type against the current rating of the controller at startup. If the short circuit current of the selected motor type exceeds the controller max current rating, then a Type 6 Parameter Mismatch Fault will be declared disallowing operation of the motor. If this parameter is enabled (On), and the short circuit current of the selected motor type is greater than the max current rating of the controller, then a motor short circuit will result in currents that will likely damage the controller. This damage will be logged by the controller for warranty purposes. **IMPROPER SETTING OF THIS PARAMETER MAY RESULT IN CONTROLLER DAMAGE AND A VOIDED WARRANTY.**

MOTOR SETUP/PMAC (PERMANENT MAGNET MOTOR) — COMMISSIONING TESTS MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Typical Max Speed *Typical_Max_Speed* `0x3542 0X00`	500 – 24000 RPM *500 – 24000*	4000 rpm	Set this parameter to the typical maximum motor speed of the vehicle. This value does not need to be set precisely; an estimate will do. All of the vehicle response rates are normalized to Typical Max Speed. For example suppose Typical_Max_Speed is fixed at 6000 rpm, Full_Accel_Rate_LS_SpdM = 3.0 seconds, and Full_Accel_Rate_ HS_SpdM = 3.0 seconds: If Max_Speed_SpdM = 6000 rpm at full throttle it will take 3.0 seconds to accelerate from zero to top speed (6000 rpm). If Max_Speed_SpdM = 3000 rpm at full throttle it will take 1.5 seconds to accelerate from zero to top speed (3000 rpm). If Max_Speed_SpdM = 1000 rpm at full throttle it will take 0.5 seconds to accelerate from zero to top speed (1000 rpm).
Max Test Speed *PMAC_Commissioning_Max_ Test_Speed* `0x3287 0x00`	100 – 24000 RPM *100 – 24000*	4000 rpm	This parameter is used to set the maximum motor speed allowed during PMAC motor commissioning.
Max Speed Limit *Max_Speed_Limit* `0x37A7 0x00`	0.0 – 500.0 % *0 – 5000*	25.0 %	Defines the percent over the programmed max speed before a fault will be checked.
Max Speed Time Limit *Max_Speed_Time_Limit* `0x37A9 0x00`	0.1 – 10.0 sec *100 – 10000*	5.0 sec	Controls the maximum time of the up/down counter for max speed supervision. If speed limit is exceeded, the timer counts up, otherwise counts down. If the timer reaches this time, a fault is declared.
Motor Type *PMAC_Motor_Type* `0x3669 0x00`	0 – 137 *0 – 137*	0	This parameter references a predefined table of motor parameters for many PMAC motors. Consult your local Curtis customer support engineer for information on how to set this parameter based on your application and motor.

MOTOR SETUP/PMAC (PERMANENT MAGNET MOTOR) — COMMISSIONING TESTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Position Sensor Type *Feedback_Type* 0x3520 0x00	Enumeration 1 – 2	Sin/Cos	Set this parameter to the type of position feedback device in your vehicle: 1 = quadrature encoder 2 = sin/cos encoder 3 = internal steering 4 = Hall switch 5 = PMDC open-loop/sensorless (contact Curtis). Note: A Parameter Mismatch Fault will result unless the Position Sensor Type matches the Motor Technology setting as follows: If Motor Technology = 0 (ACIM) then Position Sensor Type must = 1 (quadrature encoder) or 2 (sin/cos encoder). If Motor Technology = 1 (PMAC) Position Sensor Type must = 2 (sin/cos encoder) or = 4 (Hall switch).
Enable Multiturn Sensor *Feedback_Multiturn* 0x306D 0x00	0 – 1 *0 – 1*	0 **[PCF]**	Enable Multiturn Sensor = 0 (Off): One sensor cycle per mechanical revolution. (Not using a multiturn encoder). Enable Multiturn Sensor = 1 (On): One sensor cycle per electrical cycle. (Using a multiturn encoder).
Test Enable *PMAC_Commissioning_Test_Enable* 0x3288 0x00	Off – On *0 – 1*	Off	This parameter is used to start the PMAC motor commissioning procedure.
Test Throttle *PMAC_Test_Throttle* 0x3291 0x00	–1 – 1 *–1 – 1*	0	Begins characterization if a Test Enable is asserted. IMPORTANT: If the motor starts to turn in the positive (forward) vehicle speed direction, set the Test Throttle to 1. If it turns in the negative (reverse) direction, set the Test Throttle to -1.
Max Test Current *PMAC_Commissioning_Max_Test_Current* 0x3286 0x00	2 – 40 % *655 – 13108*	20 %	This parameter is used to set the maximum motor current allowed during PMAC motor commissioning.
CR Tune Max Current *CR_Tune_Max_Current* 0x328B 0x00	10 – 100 % *102 – 1024*	100 %	Sets the maximum current regulator routine's current (percent).
Current Reg Autotune Bypass *PMAC_CR_AutoTuneBypass* 0x328A 0x00	0 – 1 *0 – 1*	0	Bypasses the CR autotune during motor characterization. This should only be done if the autotuning isn't working correctly for a particular motor. In that case the CR gains need to be determined manually.

MOTOR SETUP/PMAC (PERMANENT MAGNET MOTOR) – MOTOR DATA VALUES MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Kemf (LL) Vrms/krpm *Kemf_LL_Vrms_krpm* 0x3665 0X00	0.00 – 100.00 *0 – 10000*	4.50 **[PCF]**	The back EMF of the motor expressed as V_{RMS} line to line at 1000RPM. PCF if motor control active.

MOTOR SETUP/PMAC/PERMANENT MAGNET MOTOR – COMMISSIONING RESULTS MENU

NOTICE

Treat these parameters as READ ONLY commissioning RESULTS. The values are generated (automatically) during PMAC commissioning. After successfully commissioning the motor, an OEM may copy these values to another matching controller/system to duplicate the motor performance in another matching vehicle. Hence these are writable parameters. Copy all values exactly as commissioned. This menu also applies to the support for the Hall-effect position sensor feedback, which may only be used with PMSM (Permanent Magnet Synchronous Motor) motors. Consult the FOS Release Notes for further details.

Contact Curtis before making any changes, or if there are any questions.

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Tuning Voltage *Tuning_Nominal_Pack_Volts* `0x366B 0X00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	The voltage when the current regualtor was tuned. This parameter is used to scale the gains AND base speed. The Base Speed MUST be tuned at the same voltage as the current regulator.
Phasing order *Phasing_Order* `0x34EA 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Inverts direction of feedback relative to control which can result from phasing and encoder wiring swaps. Changing this parameter in PMAC requires commissioning, do not set manually. WARNING: Changing this parameter can cause uncommanded motion. Care should be taken when changing this parameter.
Position Sensor Offset *PMAC_Position_Sensor_Offset* `0x328E 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	The electrical offset to the position sensor for magnet alignment in PMAC applications.
Position Sensor Compensation *PMAC_Position_Sensor_Compensation* `0x328D 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Compensation of position sensor.
Kp 1 Current *Kp_1_Current* `0x3667 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	The current regulator proportional (Kp) gain for one axis.
Ki 1 Current *Ki_1_Current* `0x3666 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	The current regulator integral (Ki) gain for one axis.
Kp 2 Current *Kp_2_Current* `0x306A 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	The current regulator proportional (Kp) gain for one axis.
Ki 2 Current *Ki_2_Current* `0x306B 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	The current regulator integral (Ki) gain for one axis.

MOTOR SETUP/PMAC/PERMANENT MAGNET MOTOR — COMMISSIONING RESULTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Sin Min *Feedback_Sin_Min* `0x350F 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Minimum output by the Sin/Cos sensor on the Sin channel. Input detected at Position Feedback A. The value is set during the PMAC motor commissioning procedure.
Sin Max *Feedback_Sin_Max* `0x350E 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Maximum output by the Sin/Cos sensor on the Sin channel. Input detected at Position Feedback A. The value is set during the PMAC motor commissioning procedure.
Cos Min *Feedback_Cos_Min* `0x350C 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Minimum output by the Sin/Cos sensor on the Cos channel. Input detected at Position Feedback B. The value is set during the PMAC motor commissioning procedure.
Cos Max *Feedback_Cos_Max* `0x350B 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Maximum output by the Sin/Cos sensor on the Cos channel. Input detected at Position Feedback B. The value is set during the PMAC motor commissioning procedure.
Rsys *PMAC_Rsys* `0x306C 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	The calculated system resistive loss term.
Switch Hall Position 0 *Switch_Hall_Calibrated_ Position_0* `0x50A0 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors. This parameter represents one boundary in the angle of these six sectors.
Switch Hall Position 1 *Switch_Hall_Calibrated_ Position_1* `0x50A1 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors. This parameter represents one boundary in the angle of these six sectors.
Switch Hall Position 2 *Switch_Hall_Calibrated_ Position_2* `0x50A2 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors. This parameter represents one boundary in the angle of these six sectors.
Switch Hall Position 3 *Switch_Hall_Calibrated_ Position_3* `0x50A3 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors. This parameter represents one boundary in the angle of these six sectors.
Switch Hall Position 4 *Switch_Hall_Calibrated_ Position_4* `0x50A4 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors. This parameter represents one boundary in the angle of these six sectors.
Switch Hall Position 5 *Switch_Hall_Calibrated_ Position_5* `0x50A5 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors. This parameter represents one boundary in the angle of these six sectors.

MOTOR SETUP/PMAC/PERMANENT MAGNET MOTOR — COMMISSIONING RESULTS MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Switch Hall Pattern 0 *Switch_Hall_Pattern_0* `0x50A6 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors, and each sector has a sector pattern which consists of the outputs of Hall switches. This parameter represents the pattern of one sector.
Switch Hall Pattern 1 *Switch_Hall_Pattern_1* `0x50A7 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors, and each sector has a sector pattern which consists of the outputs of Hall switches. This parameter represents the pattern of one sector.
Switch Hall Pattern 2 *Switch_Hall_Pattern_2* `0x50A8 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors, and each sector has a sector pattern which consists of the outputs of Hall switches. This parameter represents the pattern of one sector.
Switch Hall Pattern 3 *Switch_Hall_Pattern_3* `0x50A9 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors, and each sector has a sector pattern which consists of the outputs of Hall switches. This parameter represents the pattern of one sector.
Switch Hall Pattern 4 *Switch_Hall_Pattern_4* `0x50AA 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors, and each sector has a sector pattern which consists of the outputs of Hall switches. This parameter represents the pattern of one sector.
Switch Hall Pattern 5 *Switch_Hall_Pattern_5* `0x50AB 0x00`	*Controller model basis* *Auto generated by the commissioning process* Used for copying	**N/A** **See the RESULTS post commissioning**	Three Hall switches separate an electrical cycle of motor to six sectors, and each sector has a sector pattern which consists of the outputs of Hall switches. This parameter represents the pattern of one sector.

MOTOR SETUP – MOTOR SETUP STATUS MENU

Reference the Motor Setup Status menu on page 136.

MOTOR SETUP — SIN/COS ENCODER MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Speed Filter Frequency *Speed_Filter_Frequency* `0x3224 0x00`	0.1 – 250.0 Hz *0.1 – 250*	30.0 Hz	Cutoff frequency for speed filtering. Note this value is now in hertz, when updating from old versions you may need to divide by 2pi to get the exact performance.
Swap Motor Direction *Swap_Motor_Direction* `0x362F 0x00`	Off – On *0 – 1*	Off	Swaps the mechanical direction of motor spinning. If forward throttle produces reverse direction change this parameter. This parameter is critical for emergency reverse to work properly.
Enable Multiturn Sensor *Feedback_Multiturn* `0x306D 0x00`	0 – 1 *0 – 1*	0 **[PCF]**	Enable Multiturn Sensor = 0 (Off): One sensor cycle per mechanical revolution. (Not using a multiturn encoder). Enable Multiturn Sensor = 1 (On): One sensor cycle per electrical cycle. (Using a multiturn encoder).
Sin Cos Fault Threshold *Sin_Cos_Fault_Threshold* `0x3500 0x00`	0.0 – 100.0 % *0 – 1000*	10.0 %	Controller compares expected and real angle from sine and cosine inputs and faults if this threshold is exceeded. See Sin_ Difference and Cos_Difference.
Sin Cos fault Threshold High *Sin_Cos_Fault_Threshold_High* `0x3502 0x00`	0.0 – 100.0 % *0 – 1000*	30.0 %	Fault threshold used for a short time at startup to allow adaptation algorithm to work. See Sin_Cos_Fault_Threshold.
Sin Cos Fault Time *Sin_Cos_Fault_Time* `0x3503 0x00`	10 – 1000 ms *10 - 1000*	100 ms	A Sin/Cos Sensor fault (flash code 0x36) will be generated if the sensor is outside the configuration/tolerance by 10% or at/above the supply rails for >100ms.
Sin Cos Startup Time *Sin_Cos_Startup_Time* `0x306E 0x00`	0 – 2000 ms *0 – 2000*	1000 ms	Time at startup for the sin/cos sensor to start producing valid data. No sin/cos faults will be declared during this time but the power section will not enable.
Sin Min *Feedback_Sin_Min* `0x350F 0x00`	0 – 32767 *0 – 32767*	0	Minimum output by the Sin/Cos sensor on the Sin channel. Input detected at Position Feedback A. The value is set during the PMAC motor commissioning procedure.
Sin Max *Feedback_Sin_Max* `0x350E 0x00`	0 – 32767 *0 – 32767*	0	Maximum output by the Sin/Cos sensor on the Sin channel. Input detected at Position Feedback A. The value is set during the PMAC motor commissioning procedure.
Cos Min *Feedback_Cos_Min* `0x350C 0x00`	0 – 32767 *0 – 32767*	0	Minimum output by the Sin/Cos sensor on the Cos channel. Input detected at Position Feedback B. The value is set during the PMAC motor commissioning procedure.
Cos Max *Feedback_Cos_Max* `0x350B 0x00`	0 – 32767 *0 – 32767*	0	Maximum output by the Sin/Cos sensor on the Cos channel. Input detected at Position Feedback B. The value is set during the PMAC motor commissioning procedure.

MOTOR SETUP/ SIN/COS ENCODER — SPEED FILTER MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Speed Filter Frequency *Speed_Filter_Frequency* `0x3224 0x00`	0.1 – 250 Hz *0.1 – 250*	30.0 Hz	Cutoff frequency for speed filtering. Note this value is now in hertz, when updating from old versions you may need to divide by 2pi (3.14159) to get the exact performance.

MOTOR SETUP — TEMPERATURE SENSOR MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Sensor Enable *MotorTemp_Sensor_Enable* 0x3686 0x00	On/Off *On/Off*	Off	When programmed On, this parameter enables both the motor temperature cutback and the motor temperature compensation features. For this parameter to be valid, properly configure the temperature sensor. The motor temperature cutback feature linearly cuts back the current from 100% to 0% between the Temperature Hot and Temperature Max temperatures. The motor temperature compensation feature will adapt the motor control algorithms to varying motor temperatures for improved efficiency and a more consistent performance.
Temperature *Motor_Temperature* 0x3536 0x00	–100.0 – 300.0°C *–1000 – 3000*	Read Only	Temperature sensor readout.
Motor Temp Cutback *MotorTempCutback* 0x3439 0x00	0.00 – 100.00 % *0 – 4096*	Read Only	Displays the available current resulting from the motor temperature cutback function. A value of 100% indicates no cutback in current.
Braking Thermal Cutback Enable *MotorBrakingThermalCutback_Enable* 0x3680 0x00	Off – On *0 - 1*	Off	When programmed On, drive current and all forms of regen braking current will be cut back based on motor temperature. This applies to all forms of regen braking current ***(Regen_Current_Limit, Brake_Current_Limit, EMR_Current_Limit, Interlock_Brake_Current_Limit)*** and includes emergency reverse braking, interlock braking, brake input braking, direction reversal braking, neutral braking and speed limit braking. If the vehicle has mechanical brakes, setting this parameter to On may help reduce motor heating. When programmed Off, only drive current (and not the regen braking currents) will be cut back based on motor temperature. **CAUTION:** Contact Curtis support to make changes. Enabling will affect regen current cutback causing braking distance to increase when motor or controller is hot. Do not enable without independent braking system.
Temperature Hot *MotorTemp_Hot* 0x3683 0x00	0 – 250°C *0 – 2500*	145°C	Defines the temperature at which current cutback begins.
Temperature Max *MotorTemp_Max* 0x3685 0x00	0 – 250°C *0 – 2500*	160°C	Defines the temperature at which current cutback is zero.
Motor Temp LOS Max Speed *MotorTemp_LOS_Max_Speed* 0x3684 0x00	100 – 24000 rpm *100 – 24000*	800 rpm	When a Motor Temp Sensor Fault (fault code 29) is set, it engages the LOS (Limited Operating Strategy) mode. This reduces the maximum speed to the programmed Max Speed. The speed is the lower of the speed mode's settings (*Max_Speed_SpdMx* or *Max_Speed_SpdM*), or the *MotorTemp_LOS_Max_Speed* setting, whichever is lower.
Sensor Type *MotorTemp_Sensor_Type* 0x3688 0x00	0 – 4* *0 – 4* **Enumeration: Custom KTY83 KTY84-130 KTY84-150 PT1000*	4	Sensor types are predefined in the software: Type 0- Custom Type 1 KTY83-122 Type 2 KTY84-130 Type 3 KTY84-150 Type 4 PT1000 Custom sensor types are configurable, if none of the five predefined types are appropriate for the application. Low resistance sensors, such as the PT100, are not configurable. Note: KTY silicon temperature sensors have a polarity band, which must be the end connected to I/O Ground.

MOTOR SETUP — TEMPERATURE SENSOR MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Sensor Offset *MotorTemp_Sensor_Offset* `0x3687 0x00`	–20 – 20°C *–200 – 200*	0°C	Often the motor temperature sensor is placed in the motor at a location with a known offset to the critical temperature; the offset can be corrected with this parameter. The parameter can also be used to correct a known offset in the sensor itself.

TEMPERATURE SENSOR-SENSOR MAP MENU

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Sensor Resistance 0 *Sensor_0* `0x3689 0x00`	0 – 100000 Ω *0 – 10000000*	0 Ω	The user defined motor temperature sensor resistance at the corresponding *Temp_0*. For PTC thermistors, this is the resistance for the first or lowest temperature in the mapped range (i.e., -40°C). For NTC thermistors, this is the resistance for the first or highest temperature in the mapped range (i.e., 200°C). Consult the temperature sensor datasheet for this value.
Sensor Resistance 1 *Sensor_1* `0x368A 0x00`	0 – 100000 Ω *0 – 10000000*	0 Ω	The user defined motor temperature sensor resistance at the corresponding *Temp_1*. This is the second mapped resistance in the low-to-high or high-to-low temperature range. Consult the temperature sensor datasheet for this value.
Sensor Resistance 2 *Sensor_2* `0x368B 0x00`	0 – 100000 Ω *0 – 10000000*	0 Ω	The user defined motor temperature sensor resistance at the corresponding *Temp_2*. This is the third mapped resistance in the low-to-high or high-to-low temperature range. Consult the temperature sensor datasheet for this value.
Sensor Resistance 3 *Sensor_3* `0x368C 0x00`	0 – 100000 Ω *0 – 10000000*	0 Ω	The user defined motor temperature sensor resistance at the corresponding *Temp_3*. This is the fourth mapped resistance in the low-to-high or high-to-low temperature range. Consult the temperature sensor datasheet for this value.
Sensor Resistance 4 *Sensor_4* `0x368D 0x00`	0 – 100000 Ω *0 – 10000000*	0 Ω	The user defined motor temperature sensor resistance at the corresponding *Temp_4*. This is the fifth mapped resistance in the low-to-high or high-to-low temperature range. Consult the temperature sensor datasheet for this value.
Sensor Resistance 5 *Sensor_5* `0x3685 0x00`	0 – 100000 Ω *0 – 10000000*	0 Ω	The user defined motor temperature sensor resistance at the corresponding *Temp_5*. This is the sixth mapped resistance in the low-to-high or high-to-low temperature range. Consult the temperature sensor datasheet for this value.
Sensor Resistance 6 *Sensor_6* `0x3686 0x00`	0 – 100000 Ω *0 – 10000000*	0 Ω	The user defined motor temperature sensor resistance at the corresponding *Temp_6*. For PTC thermistors, this is the resistance for the seventh or highest temperature in the mapped range (i.e., 200°C). For NTC thermistors, this is the resistance for the seventh or lowest temperature in the mapped range (i.e., -40°C). Consult the temperature sensor datasheet for this value.

MOTOR SETUP— TEMPERATURE SENSOR - SENSOR MAP MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DEFAULT	DESCRIPTION
Temperature 0 *Temp_0* `0x368E 0x00`	– 3276 – 3276°C *– 3276 – 3276*	0°C	The user defined motor temperature at the corresponding *Sensor_0* resistance. Consult the temperature sensor datasheet for this value.
Temperature 1 *Temp_1* `0x368F 0x00`	– 3276 – 3276°C *– 3276 – 3276*	0°C	The user defined motor temperature at the corresponding *Sensor_1* resistance. Consult the temperature sensor datasheet for this value.
Temperature 2 *Temp_2* `0x3690 0x00`	– 3276 – 3276°C *– 3276 – 3276*	0°C	The user defined motor temperature at the corresponding *Sensor_2* resistance. Consult the temperature sensor datasheet for this value.
Temperature 3 *Temp_3* `0x3691 0x00`	– 3276 – 3276°C *– 3276 – 3276*	0°C	The user defined motor temperature at the corresponding *Sensor_3* resistance. Consult the temperature sensor datasheet for this value.
Temperature 4 *Temp_4* `0x3692 0x00`	– 3276 – 3276°C *– 3276 – 3276*	0°C	The user defined motor temperature at the corresponding *Sensor_4* resistance. Consult the temperature sensor datasheet for this value.
Temperature 5 *Temp_5* `0x3697 0x00`	– 3276 – 3276°C *– 3276 – 3276*	0°C	The user defined motor temperature at the corresponding *Sensor_5* resistance. Consult the temperature sensor datasheet for this value.
Temperature 6 *Temp_6* `0x3698 0x00`	– 3276 – 3276°C *– 3276 – 3276*	0°C	The user defined motor temperature at the corresponding *Sensor_6* resistance. Consult the temperature sensor datasheet for this value.
Offset of Windings Due to Sensor Heating *MotorTemp_Sensor_Offset* `0x3687 0x00`	–20 – 20°C *– 200 – 200*	0°C	Often the motor temperature sensor is in the motor at a location with a known offset to the critical temperature; correct the offset with this parameter. Use this parameter to correct a known offset in the sensor itself. Note: This parameter (0x3687) is also in the temperature sensor menu.

5 — SYSTEM MONITOR MENU

The numbered I/O is controller model specific. See Tables 6 and 7.

Quick Links:
Table 6 p.14
Table 7 p.15

Table 19 System Monitor Variables

COMMAND INPUTS MENU.......... p.149
- Throttle Input
- Mapped Throttle
- Throttle Multiplier
- Throttle Command
- Brake Input
- Mapped Brake
- Brake Command
- Lift Input
- Mapped Lift Throttle
- Lift Command
- Lower Input
- Mapped Lower Throttle
- Lower Command
- Interlock
- EMR Input Active

HARDWARE INPUTS MENU......... p.150
- Analog 1
- :
- Analog 31
- Pot 1 Resistance
- Pot 18 Resistance
- Pot 6 Resistance
- Pot 19 Resistance

SWITCH STATUS MENU............. p.151
- Switch 1
- :
- Switch 31

OUTPUTS MENU...................... p.153
- Driver 1 PWM
- :
- Driver 7 PWM
- Driver 1 Current
- :
- Driver 7 Current
- External 5V
- External 12V
- External 5V Current
- External 12V Current

CONTROLLER MENU.................... p.154
- Current (RMS)
- Modulation Depth
- Electrical Frequency
- Controller Temperature
- Capacitor Bank Temperature
- Main Contactor State
- Precharge State
- EM Brake State
- EMR State
- NMT State
- Regen
- Internal Timer
- CUTBACKS...................... p. 155
  - Motor Temperature Cutback
  - Controller Temperature Cutback
  - Capacitor Bank Temperature Cutback
  - Undervoltage Cutback
  - Overvoltage Cutback
  - LF Cutback
  - Battery Current Cutback
  - Overall Cutback
  - Traction FET Max Temp
  - Traction FET Max Temp Low
  - Traction FET Max Temp High
- MOTOR TUNING p.155
  - Motor RPM
  - Base Speed Captured
  - Test Field Current

BATTERY MENU............................ p.156
- BDI
- Keyswitch Voltage
- Capacitor Voltage
- Battery Current
- Battery Power

MOTOR MENU.......................... p.156
- Motor RPM
- Temperature
- Motor Torque
- Motor Power
- Feedback A
- Feedback B
- Rotor Position

VEHICLE MENU......................... p.157
- Vehicle Speed
- Vehicle Odometer
- Trip Odometer 1
- Trip Odometer 2
- Vehicle Acceleration
- Time to Speed 1
- Time to Speed 2
- Time Between Speeds
- Time to Dist 1
- Time to Dist 2
- Time to Dist 3
- Braking Distance Captured
- Distance Since Stop
- Distance Fine

FAULT HISTORY....................... p. 158

SYSTEM MONITOR MENU: COMMAND INPUTS

Quick Links:
Fig. 22 p.66
Fig. 23 p.69

VARIABLE	DISPLAY RANGE	DESCRIPTION
Throttle Input *Throttle_Pot_Percent* `0x3360 0x00`	0.0 – 100.0 % *0 – 1000*	The percent of maximum voltage at the pot wiper (Figures 12 and 13).
Mapped Throttle *Mapped_Throttle* `0x3352 0x00`	–100.0 – 100.0 % *–32767 – 32767*	Mapped throttle request. See Figure 22, Throttle Signal Processing.
Throttle Multiplier *Throttle_Multiplier* `0x335F 0x00 :00`	–200 – 200 % *–256 – 256*	Multiplies the throttle signal, which is useful in VCL throttle processing. See Figure 22, Throttle Signal Processing.
Throttle Command *Throttle_Command* `0x335D 0x00`	–100.0 – 100.0 % *–32767 – 32767*	Throttle request to slew rate block. See Figure 22, Throttle Signal Processing.
Brake Input *Brake_Pot_Percent* `0x33D3 0x00`	0.0 – 100.0 % *0 – 1000*	Normalized percentage of the brake input. Similarly to the Throttle Input variable, the controller processes the voltage at the assigned Analog Input as a percentage, and not as a voltage (due to dynamic testing), to determine the amount of motor braking (regen). Note, the controller does not offer a specific brake input pin(s) or circuit(s) as a default (*Brake_Source* = 0). Figure 13 illustrates using the 2-wire option as a brake signal input. Without a physical throttle, the analog input at pin 16 can be a physical brake input.
Mapped Brake *Mapped_Brake* `0x3350 0x00`	0.0 – 100.0 % *–32767 – 32767*	Mapped brake request. See Figure 23, Brake Signal Processing.
Brake Command *Brake_Command* `0x33D2 0x00`	0.0 – 100.0 % *0 – 32767*	Brake request to slew rate block. See Figure 23, Brake Signal Processing.
Lift Input *Lift_Pot_Percent* `0x3045 0x00`	0.0 – 100.0 % *0 – 1000*	Lift pot input after source selection and before mapping as a percentage.
Mapped Lift Throttle *Mapped_Lift_Throttle* `0x4FD8 0x00`	0.0 – 100.0 % *0 – 32766*	Hydraulics lift throttle after mapping.
Lift Command *Hydraulic_Lift_Throttle* `0x3C25 0x00`	0.0 – 100.0 % *0 – 32766*	Pump demand after input processing.
Lower Input *Lower_Pot_Percent* `0x3044 0x00`	0.0 – 100.0 % *0 – 1000*	Lower pot input after source selection and before mapping as a percentage.
Mapped Lower Throttle *Mapped_Lower_Throttle* `0x4FD9 0x00`	0.0 – 100.0 % *0 – 32766*	Hydraulics lower throttle after mapping.
Lower Command *Lower_Throttle* `0x3725 0x00`	0.00 – 100.00 % *0 – 32767*	Proportional driver current request.
Steer Input *Steer_Pot* `0x3A44 0x00`	0.0 – 100.0 % *0 – 1000*	**See the Dual Drive Operation manual (#53231_FSeriesDD).** Similar to the Throttle or Brake Input variables, when the controller's input is setup to operate as a steer input.
Steer Angle *Steer_Angle* `0x335C 0x00`	–135 – 135 degrees *–135 – 135*	**See the Dual Drive Operation manual (#53231_FSeriesDD).** Steer angle degrees calculated in Dual Drive traction manager controller.
Interlock *Interlock* `0x3012 0x00`	On/Off *On (1)/Off (0)*	Interlock state, based upon the assigned input.
EMR Input Active *EMR_Switch_Active* `0x349C 0x00`	On/Off *On (1)/Off (0)*	Indicates the EMR status. This is the combination of the NO/NC/VCL inputs.

SYSTEM MONITOR / INPUTS MENU: HARDWARE INPUTS

VARIABLE	DISPLAY RANGE	DESCRIPTION[1]
Analog 1 *analog_input_volts_1* `0x3B2E 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 1 (pin 16).
Analog 2 *analog_input_volts_2* `0x3B2F 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 2 (pin 8) (e.g., the Motor Temp Sensor).
Analog 3 *analog_input_volts_3* `0x3B30 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 3 (pin 31). Note: Typically used as the Encoder A input; therefore will track the encoder pulses.
Analog 4 *analog_input_volts_4* `0x3B31 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 4 (pin 32). Note: Typically used as the Encoder B input; therefore will track the encoder pulses.
Analog 5 *analog_input_volts_5* `0x3B32 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 5 (Input 5: Switch 5, pin 9).
Analog 6 *analog_input_volts_6* `0x3B33 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 6 (Input 6: Pot 6 Supply, pin 15).
Analog 7 *analog_input_volts_7* `0x3B34 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 7 (Input 7: Switch 7, pin 22).
Analog 8 *analog_input_volts_8* `0x3B35 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 8 (Input 8: Switch 8, pin 33).
Analog 14 *analog_input_volts_14* `0x3B38 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 14 (Input 14: +12V Ext Supply, pin 25).
Analog 18 *analog_input_volts_18* `0x3B60 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 18 (Input 18: Pot 18 Wiper, pin 17).
Analog 19 *analog_input_volts_19* `0x3B61 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 19 (Input 19: Pot 19 Supply, pin 27).
Analog 31 *analog_input_volts_31* `0x3B56 0x00`	–327.68 – 327.67V *–32768 – 32767*	Voltage at analog 31 (Input 31: +5V Ext Supply, pin 26).
Pot 1 Resistance *Pot_1_resistance* `0x32E2 0x00`	0 – 10k Ω *–32768 – 32767*	The dynamically calculated resistance of a connected potentiometer.
Pot 18 Resistance *Pot_18_resistance* `0x337E 0x00`	0 – 10k Ω *–32768 – 32767*	The dynamically calculated resistance of a connected potentiometer.
Pot 6 Resistance *Pot_6_resistance* `0x3314 0x00`	0 – 10k Ω *–32768 – 32767*	The dynamically calculated resistance of a connected potentiometer.
Pot 19 Resistance *Pot_19_resistance* `0x3318 0x00`	0 – 10k Ω *–32768 – 32767*	The dynamically calculated resistance of a connected potentiometer.

[1] *Pin number is on a 35-pin controller basis. For 23-pin controllers, use the variable name (see Fig. 12).*

SYSTEM MONITOR / INPUTS MENU: SWITCH STATUS

The On/Off (1/0) indicates the state of the binary input. The same value is encoded bitwise in the variable *Switches* (variable 0x3321) which contains the state of all 32 Switches as applicable to the controller. For the variable *Switches*, *Switch_1* corresponds to bit position 0 (LSB), and *Switch_32* corresponds to bit position 31 (MSB).

VARIABLE	DISPLAY RANGE	DESCRIPTION[1]
Switch 1 *Switch_1* `0x3324 0x00`	On/Off *On/Off*	Switch 1 On or Off (pin 16). Note: Typically used as the 3-wire throttle Pot Wiper input.
Switch 2 *Switch_2* `0x3325 0x00`	On/Off *On/Off*	Switch 2 On or Off (pin 8). As a digital (switch) input, internally pull up. On is pulled to I/O Ground, Off is an open. Note: Typically used as the Motor Temp input.
Switch 3 *Switch_3* `0x3326 0x00`	On/Off *On/Off*	Switch 3 On or Off (pin 31). Note: Typically used as the Encoder A input; therefore will cycle Off/On.
Switch 4 *Switch_4* `0x3327 0x00`	On/Off *On/Off*	Switch 4 On or Off (pin 32). Note: Typically used as the Encoder B input; therefore will cycle Off/On.
Switch 5 *Switch_5* `0x3328 0x00`	On/Off *On/Off*	Switch 5 On or Off (pin 9). Note: Typically used as the Interlock input.
Switch 6 *Switch_6* `0x3329 0x00`	On/Off *On/Off*	Switch 6 On or Off (pin 15). Note: Typically used as the 3-wire throttle Pot Hi input.
Switch 7 *Switch_7* `0x332A 0x00`	On/Off *On/Off*	Switch 7 On or Off (pin 22). Note: Typically used as the Forward input.
Switch 8 *Switch_8* `0x332B 0x00`	On/Off *On/Off*	Switch 8 On or Off (pin 33). Note: Typically used as the Reverse input.
Switch 9 *Switch_9* `0x332C 0x00`	On/Off *On/Off*	Switch 9 On or Off (pin 24). Note: Typically used as the N.O. Emergency Reverse input.
Switch 10 *Switch_10* `0x332D 0x00`	On/Off *On/Off*	Switch 10 On or Off (pin 10). Note: Typically used as the Lift input.
Switch 11 *Switch_11* `0x332E 0x00`	On/Off *On/Off*	Switch 11 On or Off (pin 11). Note: Typically used as the Lower input.
Switch 12 *Switch_12* `0x332F 0x00`	On/Off *On/Off*	Switch 12 On or Off (pin 12). Note: Typically used as the optional fork's Travel Limit input.
Switch 13 *Switch_13* `0x3330 0x00`	On/Off *On/Off*	Switch 13 On or Off (pin 14). Note: Typically used as the N.C. Emergency Reverse input.
Switch 14 *Switch_14* `0x3331 0x00`	On/Off *On/Off*	Switch 14 On or Off (pin 25). Note: The +12V supply (pin 25), if not enabled, can be configured as a switch input.
Switch 18 *Switch_18* `0x333D 0x00`	On/Off *On/Off*	Switch 18 On or Off (pin 17). Note: Typically used as the 2 or 3-wire Pot Wiper input.

[1]Pin number is on a 35-pin controller basis. For 23-pin controllers, use the variable name (see Fig. 12).

SYSTEM MONITOR / INPUTS MENU: SWITCH STATUS, cont'd

VARIABLE	DISPLAY RANGE	DESCRIPTION[1]
Switch 19 *Switch_19* `0x333E 0x00`	On/Off *On/Off*	Switch 19 On or Off (pin 27). Note: This is for the 3-wire Pot Hi input, coupled with Input 18.
Switch 20 *Switch_20* `0x3339 0x00`	On/Off *On/Off*	Switch 20 On or Off (pin 1). Note: This is the KSI input.
Switch 21 *Switch_21* `0x3332 0x00`	On/Off *On/Off*	Switch 21 On or Off (pin 2). Note: Typically not configured as a switch, but as Driver1.
Switch 22 *Switch_22* `0x3333 0x00`	On/Off *On/Off*	Switch 22 On or Off (pin 5). Note: Typically not configured as a switch, but as Driver2.
Switch 23 *Switch_23* `0x3334 0x00`	On/Off *On/Off*	Switch 23 On or Off (pin 4). Note: Typically not configured as a switch, but as Driver3.
Switch 24 *Switch_24* `0x3335 0x00`	On/Off *On/Off*	Switch 24 On or Off (pin 3). Note: Typically not configured as a switch, but as Driver4.
Switch 25 *Switch_25* `0x3336 0x00`	On/Off *On/Off*	Switch 25 On or Off (pin 6). Note: Typically not configured as a switch, but as Driver5.
Switch 26 *Switch_26* `0x3337 0x00`	On/Off *On/Off*	Switch 26 On or Off (pin 19). Note: Typically not configured as a switch, but as Driver6.
Switch 27 *Switch_27* `0x3337 0x00`	On/Off *On/Off*	Switch 27 On or Off (pin 20). Note: Typically not configured as a switch, but as Driver7.
Switch 31 *Switch_31* `0x333F 0x00`	On/Off *On/Off*	Switch 31 On or Off (pin 26). Note: The +5V supply (pin 26), if not enabled, can be configured as a switch input.

[1]Pin number is on a 35-pin controller basis. For 23-pin controllers, use the variable name (see Fig. 12).

SYSTEM MONITOR MENU: OUTPUTS

VARIABLE	DISPLAY RANGE	DESCRIPTION[1]
Driver 1 PWM *Driver_1_PWM* `0x3402 0x00`	0.000 – 100.000 % *0 – 32767*	Driver 1 PWM output percentage (pin 2).
Driver 2 PWM *Driver_2_PWM* `0x3405 0x00`	0.000 – 100.000 % *0 – 32767*	Driver 2 PWM output percentage (pin 5).
Driver 3 PWM *Driver_3_PWM* `0x3408 0x00`	0.000 – 100.000 % *0 – 32767*	Driver 3 PWM output percentage (pin 4).
Driver 4 PWM *Driver_4_PWM* `0x340B 0x00`	0.000 – 100.000 % *0 – 32767*	Driver 4 PWM output percentage (pin 3).
Driver 5 PWM *Driver_5_PWM* `0x340E 0x00`	0.000 – 100.000 % *0 – 32767*	Driver 5 PWM output percentage (pin 6).
Driver 1 Current *driver_1_current* `0x3400 0x00`	–327.68 – 327.67 Amps *–32768 – 32767*	Calculated current through Driver 1. See Driver current rating.
Driver 2 Current *driver_2_current* `0x3403 0x00`	–327.68 – 327.67 Amps *–32768 – 32767*	Calculated current through Driver 2. See Driver current rating.
Driver 3 Current *driver_3_current* `0x3406 0x00`	–327.68 – 327.67 Amps *–32768 – 32767*	Calculated current through Driver 3. See Driver current rating.
Driver 4 Current *driver_4_current* `0x3409 0x00`	–327.68 – 327.67 Amps *–32768 – 32767*	Calculated current through Driver 4. See Driver current rating.
Driver 5 Current *driver_5_current* `0x340C 0x00`	–327.68 – 327.67 Amps *–32768 – 32767*	Calculated current through Driver 5. See Driver current rating.
External_5V *External_5V_Supply* `0x36AA 0x00`	0.00 – 100.00V *0 – 10000*	Voltage at +5V output (pin 26).
External_12V *External_12V_Supply* `0x36AB 0x00`	0.00 – 100.00V *0 – 10000*	Voltage at +12V output (pin 25).
External 5V Current *Ext_5V_Current* `0x36AE 0x00`	0 – 200 mA *0 – 200*	Current (mA) of the external +5V voltage supply (pin 26).
External 12V Current *Ext_12V_Current* `0x36AF 0x00`	0 – 200 mA *0 – 200*	Current (mA) of the external +12V voltage supply (pin 25).

[1] Pin number is on a 35-pin controller basis. For 23-pin controllers, use the variable name (see Fig. 12).

SYSTEM MONITOR MENU: CONTROLLER

VARIABLE	DISPLAY RANGE	DESCRIPTION
Current (RMS) *Current_RMS* `0x3082 0x00`	0.0 – 1000.0A *0 – 10000*	The RMS current of the controller, taking all three phases into account.
Modulation Depth *Modulation_Depth* `0x30B0 0x00`	0.0 – 100.0 % *0 – 1182*	The usage percentage of the available B+ voltage.
Electrical Frequency *Electrical_Frequency_Display* `0x33E6 0x00`	–35791394.0 – 35791394.0 Hz *–2147483648 – 2147483647*	The controller's (filtered) electrical frequency.
Controller Temperature *Controller_Temperature* `0x3000 0x00`	–100.0 – 300.0°C *–1000 – 3000*	The controller's internal temperature (on the power base).
Capacitor Bank Temperature *CapBankTemperature* `0x3214 0x00`	–100.0 – 300.0°C *–1000 – 3000*	Temperature of capacitor bank.
Main Contactor State *Main_State* `0x34C9 0x00`	0 – 10 *0 – 10*	Main contactor state: 0 = open. 1 = precharge. 2 = weld check. 3 = closing delay. 4 = missing check. 5 = closed (when Main Enable = On). 6 = delay. 7 = arc check. 8 = open delay. 9 = fault. 10 = closed (when Main Enable = Off).
Precharge_State *Precharge_State* `0x34C0 0x00`	–32768 – 32767 *–32768 – 32767*	The Precharge state: 0 = Precharge not run. 1 = Precharge in progress. 2 = Precharge completed. 3 = Precharge aborted. 4 = Failed energy limit. 5 = Failed time limit.
EM Brake State *EMBrakeState* `0x347A 0x00`	0 – 4 *0 – 4*	EM brake state: 0 = engaged. 1 = releasing. 2 = released. 3 = engaging. 4 = engaged and vehicle stopped.
EMR State *EMR_State* `0x3490 0x00`	0 – 1 *0 – 1*	0 = Off. 1 = On.
NMT State *CAN_NMT_State* `0x32A4 0x00`	0 – 127 *0 – 127*	Controller CAN NMT state: 0 = initialization. 4 = stopped. 5 = operational. 127 = pre-operational.
Regen *Regen* `0x30BA 0x00`	0 – 1 *0 – 1*	1 = Regenerative Braking (mode). 0 = Drive (mode).
Internal Timer *Internal_timer* `0x4E14 0x00`	0.0 – 429496729.5 s *0 – 4294967295*	The manager timer is a timer of the total controller hours (keyswitch "On" hours). This manager timer is part of the operating system (OS) and is not resettable.

SYSTEM MONITOR / CONTROLLER MENU: CUTBACKS

VARIABLE	DISPLAY RANGE	DESCRIPTION
Motor Temperature Cutback *MotorTempCutback* `0x3439 0x00`	0 – 100 % *0 – 4096*	Displays the current available as a result of the motor temperature cutback function. A value of 100% indicates no cutback in current.
Controller Temperature Cutback *ControllerTempCutback* `0x3437 0x00`	0 – 100 % *0 – 4096*	Displays the current available as a result of the controller temperature cutback function. A value of 100% indicates no cutback in current.
Undervoltage Cutback *UnderVoltageCutback* `0x343B 0x00`	0 – 100 % *0 – 4096*	Displays the current available as a result of the undervoltage cutback function. A value of 100% indicates no cutback in current. Does only limit drive current, but does not affect regen current limit.
Overvoltage Cutback *OverVoltageCutback* `0x343A 0x00`	0 – 100 % *0 – 4096*	Displays the current available as a result of the overvoltage cutback function. A value of 100% indicates no cutback in current. Does only limit regen current, but does not affect drive current limit.
LF Cutback *LF_Cutback* `0x4F5C 0x00`	0 – 100 % *0 – 4096*	Displays the current available as a result of the low motor speed (electrical-frequency) cutback function. A value of 100% indicates no cutback in current.
Battery Current Cutback *BatteryCurrentCutback* `0x3C05 0x00`	0.0 – 100.0 % *0 – 4096*	Battery current cutback percent. 100% is no cutback, 0% is full cutback.
Overall Cutback *OverallCutback* `0x32D9 0x00`	-1000 - 1000°C *-1000 - 1000*	The accumulated controller cutback.
Traction FET Max Temp *Max_Traction_FET_Temp* `0x33DF 0x00`	-1000 - 1000 °C *-1000 - 1000*	Estimated maximum FET temperature from thermal model.
Traction FET Max Temp Low *FET_Max_Temp_Low* `0x4F8D`	–50 – 200°C *– 50 – 200*	Traction FET Junction Temperature at which the controller begins cutting back current.
Traction FET Max Temp High *FET_Max_Temp_High* `0x4F8E`	–50 – 200°C *– 50 – 200*	Traction FET Junction Temperature at which the controller allows zero current.

**Note that the displayed Pump FET Temp variables do not apply to the non-pump controllers.*

SYSTEM MONITOR / CONTROLLER MENU: MOTOR TUNING

VARIABLE	DISPLAY RANGE	DESCRIPTION
Motor RPM *Motor_RPM* `0x352F 0x00`	–30000 – 30000 rpm *–30000 – 30000*	Motor speed in revolutions per minute (rpm).
Base Speed Captured *Base_Speed_Captured* `0x352B 0x00`	0 – 65535 rpm *0 – 65535*	Displays the value of the motor's base speed captured in the most recent acceleration. Use this value to set the FW Base Speed parameter using the FW Base Speed set procedure or the ACIM motor characterization procedure.
Test Field Current *Test_Field_Current* `0x3092 0x00`	0.0 – 800.0 Amps *0 – 8000*	The Field Test Current reading during the ACIM motor autocharacterization procedure. See Chapter 6, Commissioning – Automated ACIM (motor) Characterization Procedure.

SYSTEM MONITOR MENU: BATTERY

VARIABLE	DISPLAY RANGE	DESCRIPTION
BDI *BDI_Percentage* `0x33A5 0x00`	0 – 100 % *0 – 100*	Battery state-of-charge (SOC).
Keyswitch Voltage *Keyswitch_Volts* `0x3398 0x00`	0.00 – 105.5V *0 – 10500*	Voltage at the keyswitch (KSI, pin 1). This will be at/near the battery voltage during operation.
Capacitor Voltage *Capacitor_Volts* `0x34C1 0x00`	0.00 – 200.00V *0 – 20000*	Voltage of the controller's internal capacitor bank at B+ terminal. This will be at/near the battery voltage during operation after precharge.
Battery Current *Battery_Current_Display* `0x338F 0x00`	–3276.8 – 3276.7A *–32768 – 32767*	Controller battery current. A calculated value in DC Amps.
Battery Power *Battery_Power* `0x3390 0x00`	–3276.8 – 3276.7 kW *–32768 – 32767*	Controller battery power. A calculated value in kW.

SYSTEM MONITOR MENU: MOTOR MENU

VARIABLE	DISPLAY RANGE	DESCRIPTION
Motor RPM *Motor_RPM* `0x352F 0x00`	–30000 – 30000 rpm *–30000 – 30000*	Motor speed in revolutions per minute (rpm).
Temperature *Motor_Temperature* `0x3536 0x00`	–100.0 – 300.0°C *–1000 – 3000*	Motor temperature sensor readout in °C or °F. See P*rogrammer » Application setup » Vehicle » Metric Units*.
Motor Torque *Motor_Torque* `0x3538 0x00`	–500 – 500 Nm *–500 – 500*	Motor torque. A calculated value in Nm.
Motor Power *Motor_Power* `0x352C 0x00`	–100.0 – 100.0 kW *–10000 – 10000*	Active motor power. A calculated value in kilowatts (kW).
Feedback A *AD_Encoder_Sine* `0x3B24 0x00`	–32768 – 32767 *–32768 – 32767*	Encoder or Sine sensor input (pulse count).
Feedback B *AD_Encoder_Cosine* `0x3B23 0x00`	–32768 – 32767 *–32768 – 32767*	Encoder or Cosine sensor input (pulse count).
Rotor Position *Position_Rotor* `0x3523 0x00`	–524288.000 – 524288.000 revs *–2147483648 – 2147483647*	Rotor revolutions since the last KSI= On. Rotor revolutions in the forward direction are positive (count-up), while revolutions in the reverse direction are negative (countdown); the sum of which returns the number of resultant rotor "positions" (revolutions).

SYSTEM MONITOR MENU: VEHICLE MENU

VARIABLE	DISPLAY RANGE	DESCRIPTION
Vehicle Speed *Vehicle_Speed* `0x37DD 0x00`	–3276.8 – 3276.7 *–32768 – 32767*	Vehicle speed in units of MPH or KPH depending on the setting of the Metric Units parameter.
Vehicle Odometer *Vehicle_Odometer* `0x37DA 0x00`	0.0 – 10000000.0 *0 – 100000000*	Vehicle distance traveled, in units of miles or km, depending on the setting of the Metric Units parameter in the Vehicle parameters menu. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Trip Odometer 1 *Trip_Odometer_1* `0x37D7 0x00`	0.0 – 10000000.0 *0 – 100000000*	Vehicle distance traveled, in units of miles or km, depending on the setting of the Metric Units parameter in the Vehicle parameters menu. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Trip Odometer 2 *Trip_Odometer_2* `0x37D8 0x00`	0.0 – 10000000.0 *0 – 100000000*	A second vehicle distance traveled variable, and works identically to Trip Odometer 1.
Vehicle Acceleration *Vehicle_Acceleration* `0x37D9 0x00`	0.000 – 10.000 g *0 – 10000*	Vehicle acceleration. This is a calculated value. Set the Speed to RPM parameter correctly, for an accurate measurement.
Time to Speed 1 *Time_to_Capture_Speed_1* `0x37D3 0x00`	0.00 – 128.00 s *0 – 32000*	Time taken for the vehicle to go from zero rpm to the programmed Capture Speed 1 during its most recent such acceleration.
Time to Speed 2 *Time_to_Capture_Speed_2* `0x37D4 0x00`	0.00 – 128.00 s *0 – 32000*	Time taken for the vehicle to go from zero rpm to the programmed Capture Speed 2 during its most recent such acceleration.
Time Between Speeds *Time_Between_Capture_Speeds* `0x37CF 0x00`	0.00 – 128.00 s *0 – 32000*	Time taken for the vehicle to go from programmed Capture Speed 1 to programmed Capture Speed 2 during its most recent such acceleration.
Time to Dist 1 *Time_to_Capture_Distance_1* `0x37D0 0x00`	0.00 – 128.00 s *0 – 32000*	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 1 during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Time to Dist 2 *Time_to_Capture_Distance_2* `0x37D1 0x00`	0.00 – 128.00 s *0 – 32000*	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 2 during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Time to Dist 3 *Time_to_Capture_Distance_3* `0x37D2 0x00`	0.00 – 128.00 s *0 – 32000*	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 3 during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly.
Braking Distance Captured *Braking_Distance_Captured* `0x37C0 0x00`	0.0 – 1000000.0 *0 – 40000000*	Distance traveled by the vehicle starting with vehicle braking (initiated by throttle reversal, *VCL_Brake*, or interlock braking) and ending when *Motor_RPM* = 0. Units are meters or feet, depending on the setting of the Metric Units parameter. For accurate distance measurements, the Speed to RPM parameter must be set correctly.

SYSTEM MONITOR MENU: VEHICLE, cont'd

VARIABLE	DISPLAY RANGE	DESCRIPTION
Distance Since Stop *Distance_Since_Stop* `0x37C8 0x00`	0.0 – 1000000.0 *0 – 40000000*	Distance traveled by the vehicle starting from a stop. In effect, this parameter uses the vehicle as a tape measure. For example, if the vehicle travels 300 feet forward and then 300 feet in reverse, the distance would be 600. The distance is continuously updated and will stop (and restart) when *Motor_RPM* = 0. For accurate distance measurements, the Speed to RPM parameter must be set correctly. Units are meters or feet, depending on the setting of the Metric Units parameter.
Distance Fine *Distance_Fine_Long* `0x37C7 0x00`	0.0 – 4000000.0 *0 – 40000000*	Position measurement. The net distance from the forward and reverse directions. (In other words, if the travel is 20 inches forward and then 20 inches in reverse, the distance would be zero.). Continuously updated, this distance will roll over when the variable goes over the limits. Resets to zero on key cycle. Units are decimeters or inches, depending on the setting of the Metric Units parameter. For accurate distance measurements, the Speed to RPM parameter must be set correctly.

SYSTEM MONITOR MENU: FAULT HISTORY

VARIABLE	DISPLAY RANGE	DESCRIPTION
Clear History *Fault_History_Clear_Command* `0x20F0 0x01`	0 – 1 (increment to clear) *0 – 1*	Changing this variable from 0 to 1 will clear the controller's Fault History. Faults in the menu will clear, and the parameter will automatically revert to a value of 0 when complete. See Chapter 7, Diagnostic and Troubleshooting.

6 — COMMISSIONING

INITIAL SETUP

The F-Series controllers are suitable for a variety of vehicles that differ widely in characteristics. Consequently, not all parameters or control modes may be appropriate to a given application, nor are the default parameter values. Therefore, before driving the vehicle, it is imperative to follow these initial setup steps to ensure that the controller parameter settings are suited to the application. To gain a better insight into the parameters and their settings, completely read this chapter before conducting the individual commissioning steps. Review and set up the methods to program the application's controller(s) as summarized in Appendix D.

BEFORE BEGINNING

⚠ WARNING

Before beginning these setup steps, jack the vehicle's drive wheels up off the ground so that they spin freely and the vehicle is stable— especially when the drive wheels accelerate, decelerate, or spin at higher speeds. Reference the vehicle-wiring diagram when assigning the controller's inputs, outputs and control functions. Double-check all wiring to ensure it is consistent with the wiring guidelines described in Chapter 2. Finally, check the torque of all the electrical and mechanical fasteners before proceeding with these setup steps. Ensure the vehicle harness and power cables are secure, and that there are no pinch or rub points that can cause a short circuit. Never spin pneumatic or solid tires beyond the manufacturers' maximum rpm specifications.

When commissioning and tests require the vehicle to be on the ground, able to operate/drive; ensure there are no tools and objects on/near the traction motors/drivetrain, the moving steering components/system, the wheels, and moving parts/accessories. Always operate the vehicle in a *safe test area*, where protection for personnel and property from uncontrolled operation is verified. Ensure the vehicle brakes operate, including the EM Brake and Emergency Reverse, if so equipped.

TRACTION MOTOR SELECTION

All applications must complete a revised motor commissioning process for new and existing applications changing from an E-series to F-Series controllers.

- **The existing PMAC parameters cannot be copied across from EOS to FOS.**
- **For ACIM, the parameters are transferable from EOS to FOS. However, the current regulator gains are not transferable (now removed from the ACIM motor model number) and are now part of the FOS motor commissioning routine.**
- **For applications entering motor data by hand, this requires a Curtis Engineer (applies to ACIM Motor Type = 0).**
- **The ACIM motor types are user (OEM Factory) selectable, yet require running the current regulator and base speed routines. Field weakening is optional, yet recommended for efficiency.**

The commissioning (controller/application setup) includes setting the non-motor parameters followed by the applicable motor setup. For planning purposes, outlined here is what the ACIM and PMAC options entail, with the actual motor commissioning routines following later in this chapter.

In all cases, to ensure the correct motor commission, the *Motor Setup Needed* fault will be present in the CIT/programmer. In addition, the *Motor Temp Sensor* fault is set. Both of these are "normal" default settings and designed to ensure the final application will operate correctly.

ACIM

For AC Induction Motor (ACIM) motors, use one of these three options to match the motor in the vehicle:

1. Contact the Curtis distributor or support engineer with the motor manufacturer's part number. Curtis has a database of induction motors (cdev 4.0) for which the motor data has already been determined. Each induction motor in the Curtis database is assigned a number, which is the number used for the Motor Type parameter (*IM_Motor_Type*, 0x3635).
2. Send the ACIM motor to Curtis for testing on the Curtis motor dynamometer (dyno) located in California. Once characterized, the motor's data goes into the Curtis database. This requires new software (device profile revision) with the appropriate parameter values set to match the motor. Contact the Curtis distributor or support engineer before shipping the motor. Shipping and characterizing a motor requires extra time. This process can take weeks to months, plus shipping.
3. Use the in-vehicle Induction Motor Characterization Procedure (later in this chapter), in which the controller "learns" the motor data as installed in the vehicle. Once set up, the procedure takes about half an hour with the vehicle drive wheels off the ground. Completing the characterization involves operating the vehicle in a test area (wheels on the ground tests). When choosing this option, complete steps 1 – 21 of these Initial Setup steps before conducting the ACIM characterization procedure.

ACIM motors use the quadrature Encoder. Contact Curtis if an ACIM application will use a Sin/Cos sensor (encoder).

Correct values for the motor parameters (Motor Type, FW Base Speed, and Field Weakening) must be determined individually for each motor.

PMAC

For permanent magnet AC (PMAC) motors, use one of these two options to match the motor in the vehicle:

1. Contact the Curtis distributor or support engineer with the motor manufacturer's part number. Curtis has a database of PMAC motors for which the motor data has already been determined. Each motor in the Curtis database is assigned a number, which is the number used for the Motor Type parameter (*PMAC_Motor_Type*, 0x3669).
2. Send the PMAC motor to Curtis for testing on the Curtis motor dynamometer (dyno) located in California. Once characterized, the motor's data goes into the Curtis database. This requires new software (device profile revision) with the appropriate parameter values set to match the motor. Contact the Curtis distributor or support engineer before shipping the motor. Shipping and characterizing a motor requires extra time. This process can take weeks to months, plus shipping.

PMAC motors use the Sin/Cos sensor type Encoder. Contact Curtis if a PMAC application will use a different type of rotor-position sensor. PMAC always requires additional in-vehicle motor commissioning.

Quick Links:
Fig. 12 p.16
Fig. 13 p.17
Appendix D p.252

To Begin

Verify the controller and motor wiring as per the application. Figures 12 and 13 are the basic wiring diagrams based upon the default assignment settings for the application. Each application can be different from these figures. Turn on the controller by closing the keyswitch circuit. Using a PC, connect the **Curtis Integrated Toolkit™** (CIT) to the controller via a CANbus port as described in Appendix D. Within the toolkit's Launchpad window, *highlight* the controller that is being-setup, then click to open the **Programmer** application tool. Ensure the controller and CIT project have matching device profiles. A 1313 HHP can also set/change parameters, yet if using VCL, the project/application must use the Curtis Integrated Toolkit™.

Contact the Curtis distributor or regional support engineer for help or training with the setup, use, and licensing of the Curtis Integrated Toolkit™. See Appendix D.

Note: Do not assert (close the switch or enable) the controller's Interlock at this time. This will prevent motor operation, which will come later when operating and tuning the motor.

Control Mode Selection

As described in Chapter 4, the programmable parameters and motor response characteristics are tunable through two speed control modes, depending on the application. Use the **Control Mode Select** parameter to select the tuning mode:

- Speed Mode Express
- Speed Mode
- Torque Mode

For other modes, contact Curtis.

Speed Mode Express is a simplified version of *Speed Mode* with a reduced set of parameters that is adequate for most speed-controlled applications. Speed Mode Express and Speed Mode are for applications where throttle input corresponds to motor speed output. The available parameters are context sensitive based upon the control-mode selected.

Use *Torque Mode* for applications where throttle input corresponds to motor torque output.

Note: Tune using torque control or speed control, but not both. For example, if adjusting a torque control parameter while Speed Mode or Speed Mode Express has been selected as the tuning mode, the programmer will show the new setting but it will have no effect.

Parameter Settings – Method Overview

After setting the Control Mode, the application's hardware aspects should be set first, including their range limits. Typically, do this by working from the "bottom" of the Programmer's parameter menus, setting the hardware–based parameters and assignments first, and finishing by setting the software-based parameters. For example;

Motor Setup menu (hardware) parameters,

Controller Setup menu parameters,

Application Setup menu parameters,

Speed Mode Express /Speed Mode /Torque Mode parameters.

Use the following steps as a guide to the minimum parameter selections. These steps follow the generic/default wiring diagram (Figure 12 or 13) for selectable induction motor types, unless noted otherwise (PMAC is similar). Be aware that some parameters may appear in multiple menus, so setting such a parameter in a given menu will change it in the other menu. Notice that within the parameter menus, relevant monitor items are visible— which are helpful when making/checking a selection by observing the effect of a parameter change.

These steps do not cover all variations or application specific settings. An application may change many parameters from their default values, while other parameters remain at their default values. VCL may not be required in all applications. For assistance with the application, parameter settings, and VCL, contact the local Curtis distributor or regional Curtis office.

Steps 1–4 Will setup most of the motor settings, leaving some parameters for the final motor or vehicle response tuning.

Steps 5–8 Will setup the controller to match the application. Here is where the Analog Inputs, I/O Assignments, Outputs (drivers), External Supplies, and Current Limits are set.

Steps 9–21 Covers the application's functional options. These steps generally cover how the operating system or VCL will process, control, or tune the Throttle, Brake, CAN Interface, Main Contactor, EM Brake Control, Emergency Reverse, Interlock Braking, Max Speed Supervision, Motor Not Stopped, Hazardous Movement, Motor Braking Supervision, and the IMU.

These initial setup steps do not work through all the Speed Mode/Speed Mode Express parameter-by-parameter. See Chapter 4, Programmable Parameters, for a full description of each parameter, its usage, and its available range. Some parameters cannot be fully set until the traction motor and vehicle tuning.

Finally, should the application differ from Figures 12–15, or use different hardware/control devices, then use these steps as a setup-process guideline.

Traction Motor Setup Guide

Step 1: Motor Type. Set the Motor Type

See *Programmer » AC Motor Setup » Induction Motor »* ***Motor Type***

The commissioning/characterization will come later.

Step 2: Motor Temperature Sensor

Quick Links:
Fig. 12 p.16
Fig. 13 p.17
Temp Sensor p.144

2.1 Enable the Motor Temperature Sensor by setting the Sensor Enable parameter = On.

See *Programmer » AC Motor Setup » Temperature Sensor »* ***MotorTemp_Sensor_Enable***

2.2 Set the Sensor Type parameter to the predefined type that corresponds to the motor temperature sensor. The motor temperature (thermistor) sensor connects between the Motor Temp and I/O Ground as shown in Controller Wiring Diagram. See *Programmer* » Motor Setup » *Temperature Sensor* » ***menu***.

Check whether the *Sensor Type* and *Sensor Offset* parameter settings and the motor-thermistor connections yield the correct motor temperature by reading the Temperature parameter, which is displayed in the Temperature Sensor menu, or in the CIT toolkit's Programmer app, » *System Monitor* » *Motor menu* » *Temperature*.

If the motor temperature is not correct, double-check the motor temperature control parameters Sensor Enable, Sensor Type, and Sensor Offset. Ensure that the thermistor wiring orientation has its negative (ground) side connected to ground input e.g., pin 18 on 35-pin controllers, pin 12 of 23-pin controllers.

To set up a custom sensor type, set Type = Custom (Type 0 = Custom). Using the temperature sensor datasheet, set the six sensor resistance parameters to match the six corresponding sensor temperature parameters.

To proceed if the correct motor temperature is not displayed or if a motor temperature sensor isn't used, or while awaiting a custom sensor setup/VCL, the initial setup procedure can continue, but only if the Sensor Type (*MotorTemp_Sensor_Enable*) parameter is set to Off.

If the Programmer displays the correct motor temperature, complete this step by setting the remaining motor temperature control parameters: Sensor Offset (if needed or not already completed, above), Braking Thermal Cutback Enable, Temperature Hot, Temperature Max, and MotorTemp LOS Max Speed. See Chapter 4 for these parameters' descriptions and ranges.

Quick Links:
Quadrature Encoder p.136
Encoder Steps
Swap Encoder Direction
Encoder Fault Setup (menu)

Step 3: Motor Technology & Position Sensor Type

3.1 Set the motor technology type based upon the motor.

3.2 Set the motor's position sensor type based upon the motor. Select quadrature or sin/cos for ACIM and sin/cos or Hall switch for PMAC motors.

See *Programmer » Motor Setup »* ***Motor Technology Position Sensor Type***

3.3 Complete the following parameters based upon the application-wiring and encoder kind.

See *Programmer » Motor Setup » Quadrature Encoder » menu*

5V Output Enable = On, if the 5V external supply will power the encoder. This is the default setup, as per Figures 12–15, where the 5 volts powers the encoder.

12V Output Enable = On, if the 12V external supply will power the encoder. Note: the default encoder supply is the 5 volts. Set to Off if not used, thereby freeing pin for the indicated input.

Note: Both the 5V and 12V Output Enable parameters are mirrored in the *Programmer » Controller Setup » External Supplies menu*. See this menu (Step 6) for setting the minimum and maximum supply currents, if applicable to the encoder and the application. Typically, the default supply currents are not changed.

Quadrature Encoder Steps. Set the number of encoder pulses per revolution. Note: If doing the auto-characterization, only encoder steps of 32, 48, 64, and 80 are auto-determined. If the steps are different, enter the steps now.

⚠ WARNING **Setting the Encoder Steps parameter improperly may cause vehicle malfunction, including uncommanded drive.**

Changing the Phasing Order parameter can cause uncommanded motion. Care should be taken when changing this parameter.

Phasing Order. This parameter is set such that when the motor is turning in the vehicle's forward direction the controller reports a positive motor speed (rpm). This parameter will be adjusted, as needed, when the motor is commissioned or the vehicle driven as part of the tuning process.

Encoder Fault Setup. See *Programmer » Motor Setup » Quadrature Encoder » Encoder Fault Setup »* menu.

The parameters' default values within this menu are typically sufficient for operation.

The Fault Detection Enable should be On.

Unless known now, the default Encoder Pulse Fault Detect Time of 0.5 seconds and Fault Stall Time of 5 seconds adjustments are re-examined during the application's vehicle tuning.

3.4 ACIM and PMAC motors using the Sin/Cos position sensor.

Set the parameter Enable Multiturn Sensor to match the device.

0 = OFF. 1 encoder revolution per mechanical cycle.

1 = ON. 1 encoder revolution per electrical cycle.

Set the other parameters, if known. Else, these will be set during the commissioning.

3.5 Motors using the Hall-effect switch position sensor.

Support for Hall effect position sensor feedback has been added on all controller models. This option may only be used with PMSM motors.

To use this feedback, select type 4 (Hall Switch) for *Feedback_Type* (0x3520). Commissioning is required to use this new sensor type as the commutation sequence, positions, and hysteresis need to be measured.

The values of the following parameters will be established during commissioning and need to be set as the project defaults for volume production application software releases:

Switch_Hall_Calibrated_Position_0 (0x50A0)

Switch_Hall_Calibrated_Position_1 (0x50A1)

Switch_Hall_Calibrated_Position_2 (0x50A2)

Switch_Hall_Calibrated_Position_3 (0x50A3)

Switch_Hall_Calibrated_Position_4 (0x50A4)

Switch_Hall_Calibrated_Position_5 (0x50A5)

Switch_Hall_Pattern_0 (0x50A6)

Switch_Hall_Pattern_1 (0x50A7)

Switch_Hall_Pattern_2 (0x50A8)

Switch_Hall_Pattern_3 (0x50A9)

Switch_Hall_Pattern_4 (0x50AA)

Switch_Hall_Pattern_5 (0x50AB)

Switch_Hall_Hysteresis (0x32D1)

The following variables can be used to monitor/diagnose the status of the Hall sensors: *Switch_Hall_Pattern* (0x32A2)

It is not recommended to use Hall switches for systems that need significant field weakening (operation past base speed).

Unfortunately, it is difficult to determine how much field weakening is problematic, as it's highly dependent on the motor and system design.

The Hall switch control uses the same Field Orientation as other position sensor feedback types and, therefore, produces sinusoidal control. This is in contrast to a trapezoidal drive (e.g. 1226BL).

When using Hall switches, it is expected that very low speed control gains will be necessary at low speeds.

Quick Links:
Typical Max Speed p.130
Swap Two Phases p.131
LOS Upon Encoder Fault p.133

Step 4: Remaining Motor parameters

4.1 ***Typical Max Speed***. Set the motor's typical maximum speed as per the application's requirements. This value does not need to be set precisely, an estimate will do. All the vehicle response rates are normalized to the Typical Max Speed parameter.

See *Programmer » Motor Setup »* ***Typical Max Speed***

4.2 ***Swap Motor Direction.*** This parameter will be adjusted, as needed, when the motor is driven. It swaps the mechanical direction of the motor rotation. If forward throttle produces reverse direction, change this parameter. This parameter is critical for the emergency reverse to work properly. Note that the encoder also has the phasing order parameter.

See *Programmer » Motor Setup »* ***Swap Motor Direction***

4.3 **Limited Operating Strategy.** Limited Operating Strategy (LOS) is for driving the vehicle back to a repair center at very low speeds in the event the motor encoder fails. When programmed Off, in the event of an encoder fault, the encoder fault remains and drive is disabled (i.e., the vehicle cannot be driven). If the application's LOS is known, set these parameters now, else leave the default values (LOS is enabled) and adjust these parameters as needed during motor or vehicle tuning.

See *Programmer » Motor Setup » Induction Motor » Limited Operating Strategy »* menu

4.4 **Field Weakening.** This applies only to ACIM. Regardless of whether an existing motor-type or the auto-characterization method is used, <u>all applications will need to run the Field Weakening Base Speed test</u>. Adjust these parameters, in the tuning section, or the auto-characterization procedure as applicable.

See *Programmer » Motor Setup » Induction Motor » Field Weakening »* menu

4.5 **Characterization Tests.** The parameters and monitor items in this menu are only applicable when conducting the auto-characterization procedure. See the traction motor procedures, below.

Controller Setup Guide

Step 5: Current Limits

The Drive, Regen, Brake, EMR, and Interlock Current Limit parameters are a percentage of the controller's full rated current. The controller's full rated current is on the label of the controller. Set the current limit parameters to your application's objectives, or wait until tuning the motor and/or vehicle.

See *Programmer » Controller Setup » Current Limits »* menu

Step 6: External Supplies

If the encoder's supply was not set in Step 3, set it up in the External Supplies menu. If there are minimum and/or maximum supply-current trip points, set these now. Otherwise, leave at their default values (recommended).

Quick Links:
Fig. 13 p.17
Driver checks p.125
Coil Drivers p.125
Switch Asgmt. p.120–121
Switch Status p.123

Step 7: Outputs

Here is where the output frequency for drivers 2–5 is set, and the "driver checks" are individually enabled. Using Figure 13 as the example, enable the driver checks for Driver 1 (Proportional "lower" valve), Driver 2 (EM Brake), Driver 3 (Load-hold valve), and Driver 5 (Main). Since Driver 4 is not used, its *Driver 4 Checks Enable* parameter remains at 0 (Off). Within this menu, select whether or not to use the **Coil Supply Start Up Checks** feature (On/Off).

See *Programmer » Controller Setup » Outputs » Coil Supply Start Up Checks*

Step 8: IO Assignments

As noted in Chapter 2, the controller does not have pre-defined (or fixed) coil drivers or switch assignments. The analog inputs are configurable as digital inputs and vice versa. The coil drivers are assignable as digital (switch) inputs. In this step, set the application's principle coil drivers, switch inputs, and the common analog control sources. The order of setting these parameters is not important.

See *Programmer » Controller Setup » IO Assignments*

8.1 **Coil Drivers.** Assign the driver to these common coils as applicable.

8.1.1 Set the **Main Contactor Driver** as per the application's main-contactor wiring. Figure 13 uses Driver 5.

8.1.2 Set the **EM Brake** to the application's EM Brake wiring. Note: Driver 2 can supply up to 3A, hence it will typically be the EM Brake driver as illustrated in Figures 12–15.

Note: Driver 1 is a high frequency driver, intended for proportional valves (e.g., a lowering valve).

Driver 4 is a standard driver.

The 5 drivers only have the driver's checks as an associated parameter. All other options are via VCL.

Contact the Curtis distributor or regional Curtis office for help with the coil drivers or their application.

See *Programmer » Controller Setup » IO Assignments » Coil Driver*

8.2 **Switch Assignments.** Based upon the application's wiring, assign each of the menu's parameters to a specific input (switches in these examples).

Refer to wiring diagrams (Figures 12–15) as a guide*:

Interlock Input Source = 5
Forward Input Source = 7
Reverse Input Source = 8
EMR Switch Source NO = 9
EMR Switch Source NC = 13
Lift Input Source = 10
Lower Input Source = 11

See *Programmer » Controller Setup » IO Assignments » Controls*

** Set according to the application which may not match the default settings.*

Note: The status of any switch, On or Off, is visible in the associated sub-menu.

See *Programmer » Controller Setup » IO Assignments » Switch Status*

Quick Links:
Throttle Source p.120
Inputs p.103
Throttle Setup
Fig. 35 p.164

8.3 **Controls.** This menu sets which input source is used for the listed items. Based upon the application's wiring, assign each of the menu's parameters to a specific switch. The Throttle, Brake, and Steer sources are analog inputs. Set the input type (i.e., either a switch or an "analog" voltage) for the Lift, Lift Limit Switch, and the Lower Input that matches the application. Refer to Figures 12 & 13 as a guide.

Throttle Source = 1 *

Brake Source = 0 (not used in Fig. 12, 14, & 15) *

See *Programmer » Controller Setup »IO Assignments » Controls*

Step 9. Inputs

Configure the Inputs menu. Analog1 default setting is as a 3-wire potentiometer. The other analog inputs are available for general usage, although as illustrated in the wiring diagrams, Analog 2 is the motor temperature sensor, and Analog 3 and 4 are the encoder signal inputs. As such, Analog 2, 3, and 4 do not need to be configured in this menu because their assignments are the default when the motor temperature sensor is enabled and the motor Position Sensor Type is assigned.

If a voltage-based throttle is used, set it now. If a potentiometer-based throttle is used, set it now as either a 3-wire or 2-wire. If a VCL throttle is employed, the Analog 1 Type parameter is ignored as a throttle, freeing this input to function as assigned. For example, the input can be assigned as a brake input, which would necessitate the Brake Source in Step 10 being set = 1 (for Analog 1). Figure 13 shows Analog 18 as a 2-wire potentiometer for the brake throttle. Yet, it could as easily be a voltage-type input, or via the CANbus and processed in VCL, as shown in Figure 15, the CANbus Tiller-Head example.

See *Programmer » Controller Setup » Inputs*

Traction Throttle Setup

The controller accepts Class III tiller handles with flipper and/or twist-grip throttles whether with forward and reverse switches or those of the wigwag type (symmetrical throttle response in both the forward and reverse directions, yet without directional switches). From Curtis, the ET-126 throttle assembly offers a symmetrical 0–5V Hall-effect throttle featuring a wigwag type throttle-return-to-neutral, and solid-state forward and reverse switches. Alternatively, when implementing a traditional foot throttle, Curtis offers both 3-wire *potentiometer* and Hall-effect voltage throttles—as complete pedal-assemblies, which easily integrate into vehicles.

The throttle setup depends upon the type used. Possible types include voltage, 3-wire or 2-wire potentiometers, and those received over the CANbus and processed using VCL. The controller also accepts PWM (signal output) position sensor type throttles. See each of the representative setup examples below.

In all cases, the throttle command is as a percentage (%) and not voltage. The voltage at an input will change due to the dynamic checks at the input which verify the integrity of the input signal. This is new to the FOS-series controllers and differs from the E/SE-series controllers (that display a "steady" voltage).

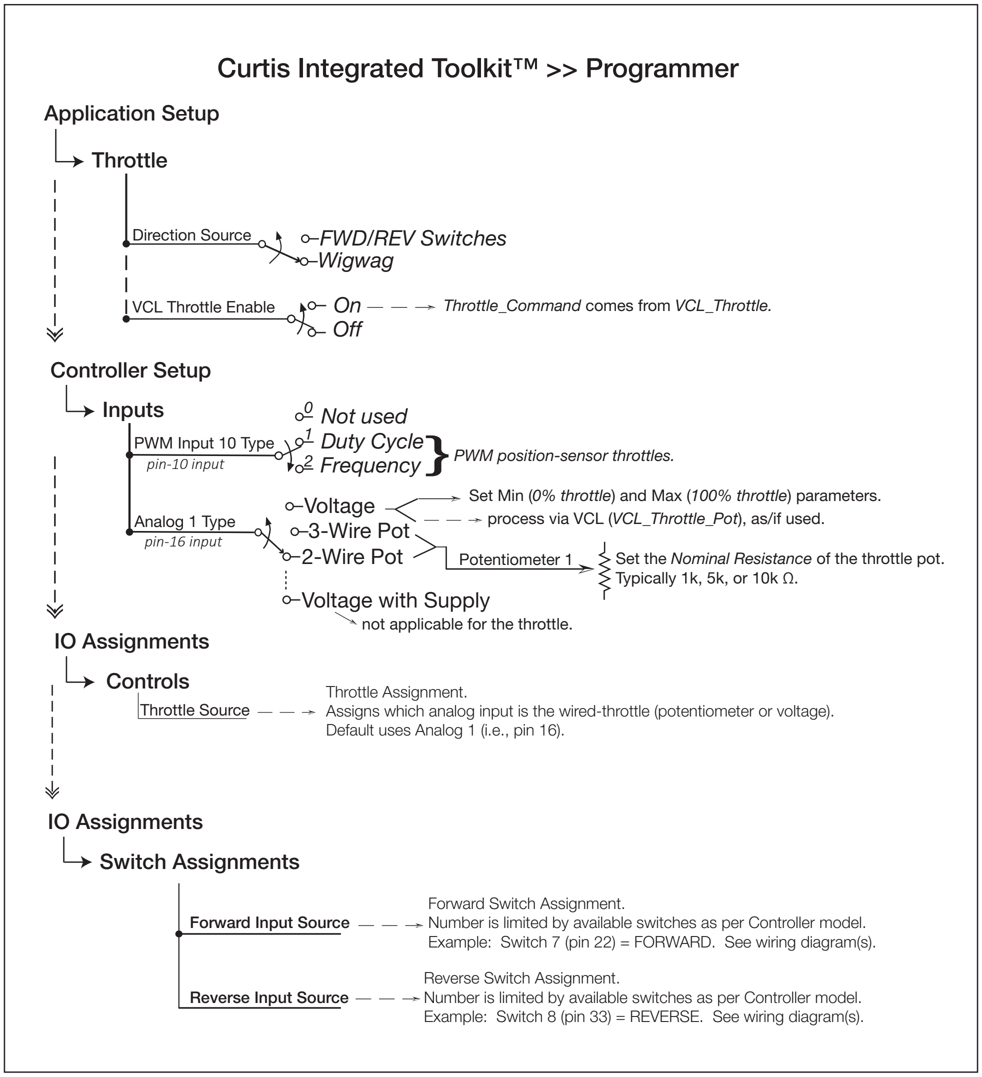
All throttle setups use the Curtis Integrated Toolkit™ to assign/set the throttle parameters. Figure 35 illustrates the Programmer's throttle-related parameters.

Before setting up the throttle parameters, turn the Interlock Off. This will prevent accidental motor operation. The Interlock was "assigned" to a switch (digital) input in the wiring diagrams. Read the Interlock Input Source parameter (value displayed) and then the indicated switch for its On/Off status in the *Programmer » Controller Setup » IO Assignments » Switch Assignment* menu and then the *Switch Status* menu (or use the *Programmer » System Monitor » Inputs » Switch Status* menu). If the Programmer indicates the interlock is On, review how the Interlock Type parameter is set and turn the interlock Off.

** If the analog input is set to zero, the input is by VCL and the associated analog input is ignored.*

Figure 35

Throttle Related Parameters – Setup Options



Quick Links:
Fig. 13 p.17
Fwd/Rev Source p.120
Switch Status p.123
Throttle Source p.120
VCL Throttle Enable p.63
Inputs p.103

9.1 Set the Forward and Reverse input sources. See *Programmer » Controller Setup » IO Assignments » Switch Assignment »* ***Forward Input Source*** *and* ***Reverse Input Source***. These are numbered switch assignments, viewable in the Switch Status menu when they are cycled On/Off.

See Figure 13 and the *Programmer » Controller Setup » Switch Status menu.*

Note, the switch status is also viewable in the *Programmer » System Monitor »* Switch Status menu.

9.2 Set the Throttle Source, which for voltage, 3-wire, or 2-wire throttles will be the Analog 1 input—set by inputting 1 for the Throttle Source parameter. See *Programmer » Controller Setup » IO Assignments »* Controls » ***Throttle Source.*** If using a VCL Throttle source, change the Throttle Source from the default 1 to 0, even though it ignores this throttle source when enabling the VCL Throttle parameter (see Step 9.5, below).

9.3 When the Throttle Source will use Analog 1 input, next set the type of input—Voltage, 3-wire, or 2-wire from the pull-down menu. See *Programmer » Controller Setup » Inputs »* ***Analog 1 Type***. Based upon the throttle type selected, that type's parameters and sub-menu will become visible, allowing targeted setup. See the voltage and potentiometer throttle setups (below).

Within the Inputs menu, if using a PWM output signal position sensor throttle, select the PWM Input 10 Type parameter, and then the sensor's appropriate duty cycle or frequency parameters in the PWM Input 10 menu (in the Analog list).

9.4 The direction source is set using the **Direction Source** parameter. See *Programmer » Application Setup » Throttle » **Direction Source***. Here, the source will be the **Fwd/Rev** selection if the switches are selected as noted in step 9.1. Alternatively, if voltage determines the direction, select Wig/Wag (wigwag throttle).

9.5 If the throttle will be via VCL, enable VCL Throttle. *Programmer » Application Setup »* Throttle » ***VCL Throttle Enable***.

The VCL Throttle provides a different way of sending the throttle command to the controller. This throttle type uses VCL to define the throttle signal that will be "input" into the throttle signal chain as *VCL_Throttle*.

See *Programmer » Controller Setup » IO Assignments » Controls » Throttle Source = 0*

The VCL program/coding will determine the source of the throttle signal, making this a very flexible throttle input method. VCL can be written to use any of the controller's analog inputs or CAN communication messages (e.g., from a vehicle manager controller) as the source of the throttle signal.

Setting the throttle to the parameter **VCL Throttle Enable** (On) re-assigns (frees) the throttle pot input for uses other than throttle input.

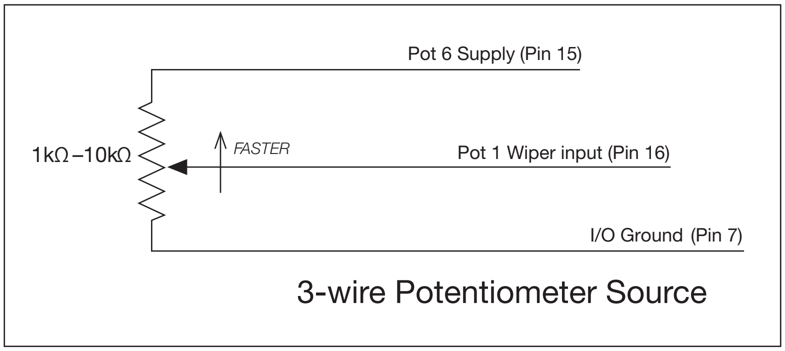
For questions regarding this throttle type, contact the Curtis distributor or support engineer.

3-wire potentiometer throttle

When a **3-wire potentiometer** is used, the controller provides full fault protection in accordance with EN 1175-1 requirements. In its voltage divider mode, the controller provides the voltage source and the potentiometer return/ground. The *Pot Supply* circuit provides a current limited to 5 volts to the *3-wire potentiometer*, and pin 18 or 7, *I/O Ground*, provides the return path. This is the 3-wire potentiometer throttle shown in the basic *wiring diagram* (*Figure 13, Chapter 2*) and Figure 36, below. If any of the three connections open, it triggers a Throttle Input fault (flash code 4-2).

The Curtis throttle FP-10 model is a 0–5 kΩ 3-wire potentiometer throttle. It offers, besides the controller's 3-wire fault detection, two throttle spring (position) detection switches, two neutral and two full throttle position switches. This throttle is configurable as a 0–5k Ω or 5k–0 Ω potentiometer.

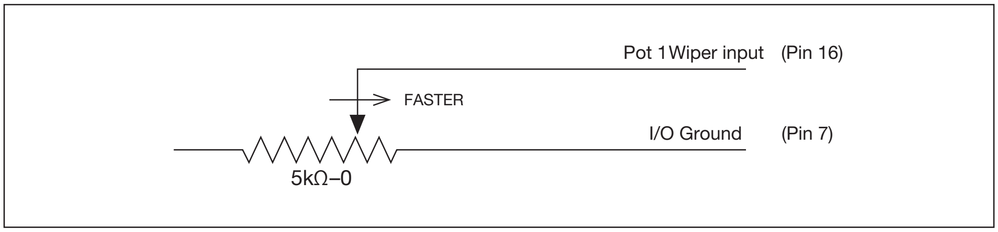
Figure 36
Wiring for 3-Wire Potentiometer throttles



2-wire potentiometer throttle

For 2-wire resistive potentiometers, shown in Figure 37, full throttle request corresponds to 0 Ω measured between the pot wiper and the I/O Ground. A 2-wire throttle requires the Forward & Reverse Min Input parameters to be set towards a higher percentage (e.g., 95%) and the Forward & Reverse Max Input parameters set to a lower percentage (e.g., 5%). Note that this is the opposite of these parameters' default settings. The broken wire protection is by the controller sensing the current flow from the pot wiper input through the potentiometer and into I/O Ground. If either connection opens, it triggers the Throttle Input fault (flash code 4-2).

Figure 37
Wiring for 2-Wire Potentiometer throttles



3-Wire or 2-Wire throttle parameter setup

Wire the throttle corresponding to type as illustrated in Figures 12, 13, 36 or 37. For potentiometer throttles, the potentiometer percentage (**Throttle Input**) variable represents the throttle position as a percentage of full throttle (100%). For throttle assemblies with validation switches or similar signals, wire and program the validation signal(s) as per their type using the controller's available switch or analog inputs. Reference Figure 13 and Tables 8 and 9 for these available switch and analog inputs. Be sure to include these additional signals in a VCL program as the means to integrate such throttle validation signals into the controller application.

Note: When selecting a resistive throttle, the monitor item **Analog 1** (*analog_input_volts_1*) reading at pin 16 relates to the assigned potentiometer value and resistive validation, and as such, the indicated voltage has no relevance to the throttle's setup or diagnostics. The analog voltage monitor value will cycle with the internal validation. Do not use the analog voltage for control purposes.

Voltage Throttle

When using a **voltage source** as a throttle, it is the responsibility of the OEM to provide appropriate throttle fault detection. For ground-referenced 0–5 volt throttles, the controller will detect open breaks (i.e., wire disconnect) in the Analog1 input (*analog_input_volts_1*), but it cannot provide full throttle fault (valid throttle signal) protection. For *tiller-handles providing a voltage signal, and for tiller-handles providing a CANbus throttle command (i.e., VCL_Throttle), throttle validation is the responsibility of the vehicle OEM. It is recommended that throttle validation be handled within the tiller-handle itself (i.e., as throttle assemblies offer).*

To use a **current source** as a voltage throttle, add a resistor in parallel to the circuit to convert the current source value to a voltage. Size the resistor to provide a 0–5V or 0–10V signal variation over the full current range. It is the responsibility of the OEM to provide appropriate *throttle fault detection* in these throttles as well. Reference the diagram in Figure 39, below.

Hall-effect voltage throttles

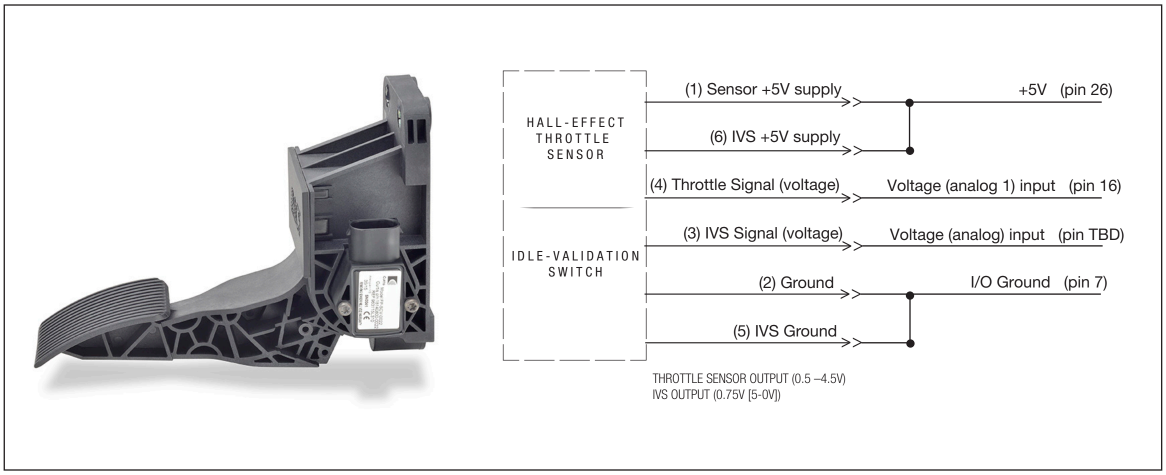
The Curtis FP Series of electronic throttles offers multiple pedal angles and mounting configurations (floor, suspended, flush) with 0–5 Volt operation and an *Idle Validation Switch (IVS)*. The IVS will connect to an assigned analog input (e.g., pin 16). See Figure 38 (35-pin controller basis).

The Curtis ET-XXX (e.g., ET-126) electronic throttle is typically a drive *throttle* (as illustrated in Figure 39). When used in Class III tiller handles or twist-throttle grips, it offers symmetrical throttle response in both the forward and reverse directions of the twist-grip/flippers (i.e., CW and CCW rotation). Based upon rotation it switches the forward or reverse switches to KSI and thus provides the directional switches within the throttle assembly. It is a wigwag type throttle, which when released, returns the throttle to neutral for convenience and safety. Other, similar third party wigwag throttles follow the same setup process.

The ET-XXX voltage throttles contain no built-in fault detection, or a throttle validation signal. It is the responsibility of the OEM to provide appropriate throttle fault detection.

Figure 38

Curtis FP-SCV-0022 Hall-effect Throttle



Voltage Throttle setup

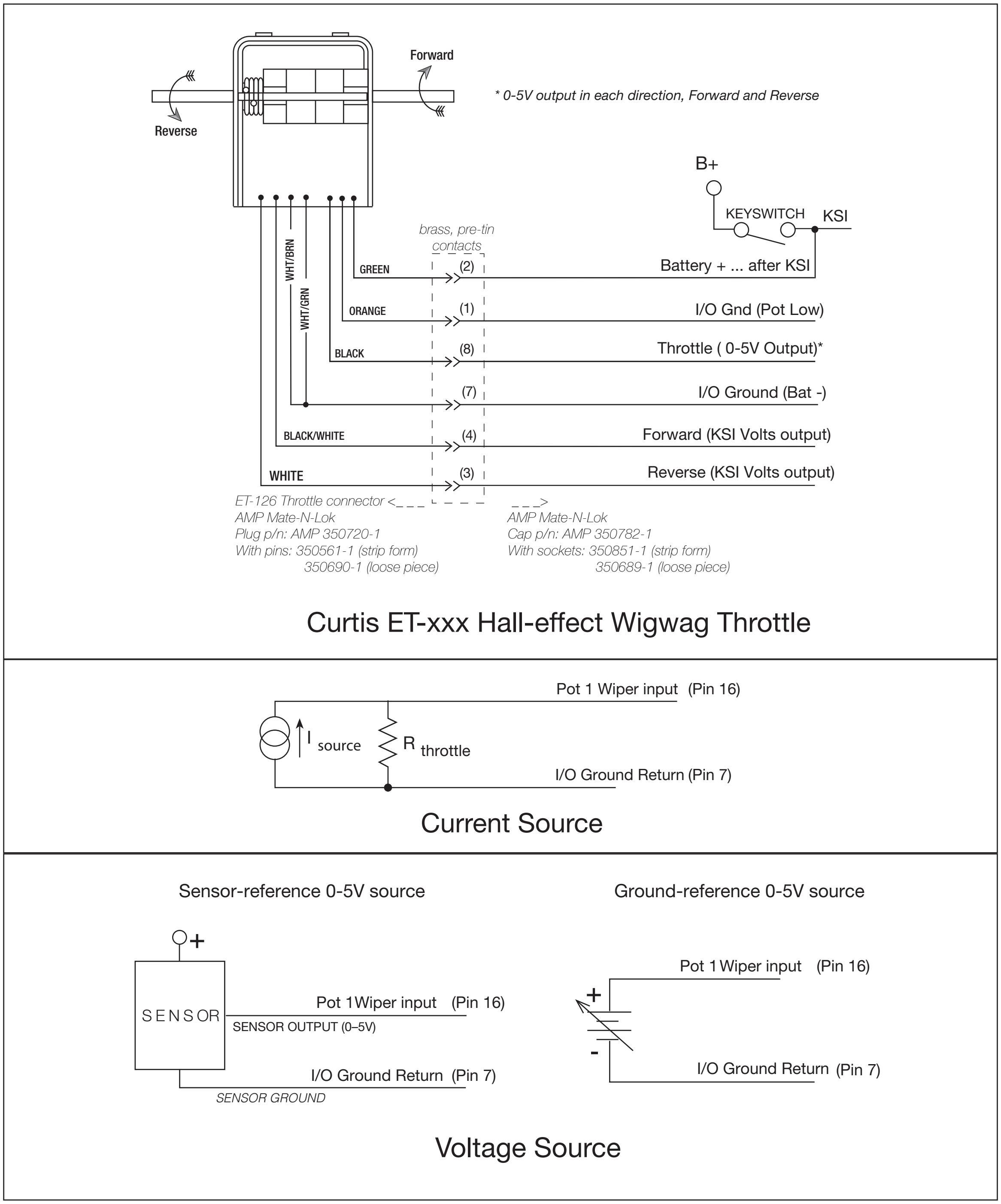
Wire the voltage throttle as illustrated in the options shown in Figures 38 and 39. With the throttle connected and powered, set the **Low** and **High** parameters based upon the observed input Voltage (***analog_input_volts_1***) while moving the throttle throughout its physical range (stroke). These parameters set the range for 0% (**Low**) and 100% (**High**) throttle. Inputs below and above these set points will declare a fault. The **Percent** (*Analog_Input_Percent_1*) monitor variable is repeated as **Throttle Input** (*Throttle_Pot_Percent*) in the *Application Setup » Throttle* menu. Use either of these variables as feedback when adjusting the potentiometer throttle's **Forward & Reverse Min/Max Input** parameters (described below).

Quick Links:
Table 8 (Switch Inputs) p.20
Table 9 (Analog Inputs) p.22

For voltage throttles with an idle validation switch or additional voltage outputs, program these signals as per their type using the controller's available switch or analog inputs. Reference the wiring diagrams and Tables 8 and 9 for these available switch and analog inputs. Be sure to include these additional signals in a VCL program as the means to integrate such throttle validation signals into the controller application.

Note: The pin numbers in Figures 36–39 are on the 35-pin controller basis.

Figure 39
Wiring for Voltage type throttles



Other throttle parameters

Quick Link:
Throttle menu p.62

Complete the throttle setup by adjusting the remaining parameters to match the range of the throttle, else tune these parameters when operating the motor following this initial setup. See the *Programmer » Application Setup » Throttle* menu.

Forward Min Input,

Forward Max Input,

Forward Map Shape,

Reverse Min Input,

Reverse Max Input,

Reverse Map Shape,

Throttle Filter,

HPD SRO Type,

Sequencing Delay.

Hint: For immediate feedback while adjusting these throttle parameters, reference the *Throttle Input* (%) monitor variable within the Throttle menu. Note that this variable is duplicated in the System Monitor.

When adjusting the forward and reverse parameters, read the displayed *Throttle Input* percentage at the point when the throttle moves out of neutral and at the point just before full throttle and enter these values for the *min* and *max* threshold settings for that direction **(Forward & Reverse Min/ Max Input)**. Set up the remaining parameters in the Throttle Menu as required by the application. (See the Throttle menu for further details on these parameters).

Verify that the throttle settings are correct by checking the *Mapped Throttle* and *Throttle Command* values over the entire range of throttle movement in the parameter or monitor menus (*Application Setup » Throttle* or *System Monitor » Inputs*). The value displayed should be positive in the forward direction and negative when in reverse. Verify that it is 0% throughout the range of throttle's neutral motion. The displayed percentage should be 100% throughout the range considered the maximum (± based upon whether forward or reverse).

Contact the Curtis distributor or support engineer to resolve any issues about the throttle setup before continuing with the initial setup procedures.

Step 10: Brake

Figure 13 illustrates a 2-wire "brake" input. If the application does not use a wired brake input, set the **Brake Pedal Enable** parameter = Off. If a brake input will be used, set Brake Pedal Enable = On, while keeping the Interlock off.

See *Programmer » Application Setup » Brake »* ***Brake Pedal Enable***

Quick Link:
Brake menu p.67

The Brake parameter setup follows that of the throttle, keeping in mind that only when using a CANbus VCL Throttle, will the Analog1 input (pin 16) be free to operate as the brake input. Otherwise, assign an available analog input and process as a voltage input in VCL. For CAN based inputs, where a physical input is not employed, set the VCL Brake Enable parameter = On. Note that the Pot 18 Wiper can be setup as a brake input, as either a voltage or 2-wire.

See *Programmer » Controller Setup » IO Assignments » Controls » Throttle Source = 0*

See *Programmer » Application Setup » Brake »* ***VCL Brake Enable***

Verify the brake settings using the Brake Command and Mapped Brake variables within the Brake menu, or those in the System Monitor Inputs menu. The value displayed for Mapped Brake should be = 0% through the range of brake pot motion that is considered neutral. The displayed Mapped Brake should be = 100% through the range of motion that is considered maximum brake. Note: The Throttle Command is zeroed if the Brake Command is > 0 for Control Modes 0 and 1 (Speed modes).

Contact the Curtis distributor or support engineer to resolve any issues about the brake setup before continuing with the setup procedure.

Step 11: CAN Interface

Set up the CAN parameters to match the application. If using PDO maps, follow the description for the PDO parameters (Chapter 4) and Appendix A, PDO Map Setup.

Quick Links:
CAN Interface menu p.70
Battery menu p.77
Main Contactor p.79
EM Brake Control p.81
Emergency Reverse p.83
Interlock Braking p.85
Vehicle menu p.94
Max Speed Supv. p.97

Step 12: Battery Setup

Set the Nominal Voltage parameter to match the nominal battery pack voltage of your system. See *Programmer » Applications Setup » Battery Setup* menu. When using a lead-acid battery, set up the Battery Discharge Indicator (BDI) parameters. Use the standard values for your type of batteries as the starting point in setting the reset, full, and empty volts-per-cell parameters. Note: For non-lead-acid batteries, including Lithium-Ion battery packs, use the pack or cell manufacturer's approved Battery Management System (BMS) for determining BDI. Process the BMS in the VCL program.

Step 13: Main Contactor

Set the main's parameters in the *Programmer » Application Setup » Main Contactor* menu to match the application. Note, the default **Pull In** and **Holding** voltages can be 0% and 0% (not set) or set to 100% and 80% respectively. The default driver assignment for the main contactor is *Driver5*. See *Programmer » Controller Setup » IO Assignments » Coil Drivers* menu. The driver's **Checks Enable** parameter is located in *Programmer » Controller Setup » Outputs » Driver 5 » Driver 5 Checks Enable*. For consistent operation across the battery voltage range, set the **Battery Voltage Compensation** parameter to On. Set the remaining main contactor parameters to match the application.

Step 14: EM Brake Control

Set up the EM Brake parameters in the *Programmer » Controller Setup » EM Brake Control* menu. Assign the EM Brake to Driver 2 as shown in Figures 12-15 — because it is rated 3 Amps. See the *Controller Setup » IO Assignments » Coil Drivers* menu. The driver's *Checks Enable* parameter is located in *Controller Setup » Outputs » Driver 2 » Driver2 Checks Enable* menu.

Step 15: Emergency Reverse (EMR)

Set up the parameters in the Emergency Reverse Menu. See *Programmer » Application Setup » Emergency Reverse* menu and » *EMR Supervision* sub-menu.

Step 16: Interlock Braking

Set up the parameters in the Interlock Braking menu, and Interlock Braking Supervision sub-menu. See *Programmer » Application Setup » Interlock Braking* menu and » *Interlock Braking Supervision* sub-menu.

Step 17: Vehicle

Set the units the application will use, metric or English. The other parameters and monitor variables within this menu will be used/set during the Tuning Guide section once the vehicle is on the ground and operational. See *Programmer » Application Setup » Vehicle* menu and » *Speed/Dist/Accel* sub-menu.

Step 18: Max Speed Supervision

Set up the parameters in the Max Speed Supervision Menu. See *Programmer » Application Setup » Max Speed Supervision* menu.

Step 19: Motor Not Stopped, Hazardous Movement, and Motor Braking Supervision

Set up the parameters in the Motor Not Stopped, Hazardous Movement, and Motor Braking Supervision menus.

See *Programmer»Application Setup menus; » Motor Not Stopped.*
» Hazardous Movement.
» Motor Braking Supervision.

Step 20: IMU

Set up the IMU as applicable to the controller and application.

See *Programmer » Application Setup » IMU*

Step 21: Clear Faults

Cycle the keyswitch to clear any parameter change faults. Use the programmer to check for faults. Clear and resolve all faults, including those in the Fault History (*Programmer » System Monitor » Fault History » Clear History*) before continuing with the initial setup.

Quick Link:
Clear History (faults) p.155

Use Chapter 7 for help in troubleshooting. Contact your Curtis customer support engineer to resolve any fault issues.

Step 22: Setting Motor Feedback Direction

ACIM:

This step "hand turns the motor" to verify the encoder direction. With the vehicle drive wheels still off the ground (vehicle jacked up on stable stands), no faults present, the interlock **Off**, and both the throttle and brake in neutral (Mapped Throttle = 0% and Mapped Brake = 0%) as verified in the *Programmer » System Monitor » Inputs* menu, the encoder direction can be checked.

While viewing the Motor RPM in *Programmer » System Monitor » AC Motor » Motor RPM*, turn the motor by hand and observe the sign of Motor RPM. Positive values are forward and negative values are reverse. If you get a positive Motor RPM when you rotate the motor in the forward direction, and a negative Motor RPM when you rotate the motor in the reverse direction, the *Swap Motor Direction* parameter is correct. If getting a negative Motor RPM when rotating the motor forward, change the *Swap Motor Direction* parameter. See *Programmer » Motor Setup » **Swap Motor Direction***. Cycle the keyswitch and repeat the procedure until you are satisfied that the *Swap Motor Direction* parameter setting is correct.

⚠ CAUTION

If the vehicle will use the emergency reverse feature, it is imperative the reverse direction (negative Motor RPM) is set correctly so that when the Emergency Reverse input is active the motor will rotate in the reverse direction, away from the operator in a Class III application.

Hint: If using the 1313 handheld programmer, add the frequent "tuning" parameters and monitor variables to the Favorites menu for quick changes and checks. The equivalent in CIT is creating a dashboard.

Contact your Curtis distributor or support engineer to resolve any issues about encoder direction or emergency reverse before continuing with the setup procedure.

PMAC:

Step 22 for checking the PMAC motor direction will be covered in its commission routine.

⚠ WARNING

Note: Do not take the vehicle down off the blocks if re-running the Motor Characterization procedure.

Quick Link:
Tuning Guide p.183

Step 23: Motor Characterization

Tuning Guide

After completing these initial setup steps, and the auto-characterization procedure if applicable, perform further tuning by following the Tuning Guide.

Hint: If using the 1313 handheld programmer, add the "tuning" parameters and monitor variables to the Favorites menu for quick changes and checks. The equivalent in CIT is creating a dashboard.

⚠ WARNING

The motor will rotate during this procedure. Do not take the vehicle down off the blocks.

AUTOMATED ACIM (MOTOR) CHARACTERIZATION PROCEDURE

This procedure is only for applications using an ACIM traction motor and quadrature encoder.

Assumptions:

- The initial setup steps 1 through 21 are complete.
- The motor is approximately at room temperature (20–25° C); do not characterize a hot motor without a temperature sensor.

When characterizing a traction system, ensure the vehicle drive wheels are clearly off the ground. Safely block it from accidental movement. The drive wheels should be freely spinning — any dragging brake or excessive friction may invalidate this test, or cause it to fail.

AC MOTOR SYSTEMS

The ACIM motor parameters can be copied from EOS to FOS, however, current regulator gains cannot and are now removed from the ACIM motor model.

Commission ACIM motors using a Quadrature Encoder. If using a Sin/Cos sensor, follow the Sin/Cos sensor setup. Consult Curtis if there are any questions.

When the FOS (cdev) defaults are loaded, the (new to FOS 4.0) *Motor Setup Needed* fault will be present. This fault is cleared automatically by following the commissioning procedure. The *Motor Setup Needed* fault *Type* (value) will indicate the tests that need to be run, as shown in the table. Convert the hexadecimal number to binary for the bit.

Bit Positions	Description
1	Current regulator tuning.
2	Slip gain setting.
3	Base speed setting.
4	Full automated test run.

ACIM Commissioning Routine:

Be sure that the motor is free to spin without load and that the motor/machine is secure.

1. Set the Motor Type and the Motor Pole Pairs.

1.1 **Motor Type = > 0** indicates a Curtis characterized motor from the motor table.

1.2 **Motor Type = 0** is for manually entering motor data values.

1.3 **Motor Type** = **–1** is the default and will run the full autocharacterization routine.

*Programmer: Motor Setup » Motor Technology: **0 = ACIM**.*

*Programmer: Motor Setup » Induction Motor (ACIM): **Motor Type**.*

2. Set the Feedback Type (Quadrature encoder or sin/cos encoder).

   *Programmer: Motor Setup » **Position Sensor Type**: = **quadrature encoder**.*

3. For Motor Type = –1, set the induction motor poles.

   *Programmer: Motor Setup » Induction Motor (ACIM) » Characterization Tests » **Motor Poles**.*

   3.1 When setting this parameter in Programmer, set this to the **number of motor poles**.

   3.2 When setting this parameter via CAN or VCL, set this to the **number of motor pole pairs**.

Steps 4 – 6 are not relevant if using a **Motor Type >=0** with a quadrature encoder as the motor does not move and it does not use the current and speed parameters.

4. Enter the desired Max Test Speed for the characterization test. Set this to the maximum motor speed in the application. Note, this speed might not be achieved, depending on system characteristics; this is normal.

   *Programmer: Motor Setup » Induction Motor (ACIM) » Characterization Tests » **Max Test Speed**.*

5. Enter the desired Max Test Current for the characterization test. A typical setting is 70% (70% of the maximum motor or controller rating, whichever is higher). Generally, this is only reduced if motor heating during the test is a problem, or resonance in the motor occurs at high currents.

   *Programmer: Motor Setup » Induction Motor (ACIM) » Characterization Tests » **Max Test Current**.*

6. Using the Programmer, clear the Fault History.

   *Programmer: System Monitor » Faults History » **Clear History**.*

⚠ WARNING

These next steps may cause the motor to move (see Table 20). Do not take the vehicle down off the blocks.

7. Engage Interlock.

   *Programmer: System Monitor » Inputs » **Interlock** (= On).*

8. Set Test Enable = **On**.

   *Programmer: Motor Setup » Induction Motor (ACIM) » Characterization Tests » **Test Enable**.*

9. Set Test Throttle = **+1** (positive).

   *Programmer: Motor Setup » Induction Motor (ACIM) » Characterization Tests » **Test Throttle**.*

   After approximately 30 or more seconds, the motor will begin to rotate. It is critical to verify the motor is turning in the FORWARD vehicle direction. If it is not, set Test Throttle = 0, wait for the motor to come to a stop, then set Test Throttle = –**1** (negative).

Note: (1) This step runs faster if TACT is not running.

(2) If the motor starts to accelerate then slows down again, you may need to increase Max Test Current (see Step 5, above):

*Programmer: Motor Setup » Induction Motor (ACIM) » Characterization Tests » **Max Test Current**.*

This will then run the tests as per Table 20 (below).

10. The automated test may take several minutes. When it is complete, the controller will have a Parameter Change Fault. This is normal. Check whether other faults are present.

    If there are any Characterization Error faults, read the number at System Monitor » Fault History » Motor_Characterization_Fault_Type (i.e., Motor Characterization Error, flash code 8-7) and reference the **Motor Characterization Errors**, Table 21. Table 21 includes the Parameter Mismatch fault list, for immediate reference for clearing faults that prevent a successful motor commission.

    All errors except "81" indicate the motor characterization data is invalid. For error "81," the data is valid but Encoder Steps must be set manually. Contact your Curtis distributor or support engineer if the Motor Characterization Errors indicated cannot be resolved.

    *Programmer: System Monitor » Fault History » **Motor_Characterization_Fault_Type**.*

11. **If Motor Type = –1**

    11.1 Complete the **Slip Gain Tests** (See below).

    11.2 Complete the **Base Speed Tests** (See below).

12. **If the Motor Type >= 0**

    12.1 Complete the **Base Speed Tests** (see below).

Table 20 ACIM Test Table

Motor Technology	Motor Type	Feedback Type	Test Run
ACIM	>=0 (0, 1 – 531)	Quadrature Encoder	Current Regulator (CR) test only. The motor will not spin.
ACIM	>=0 (0, 1 – 531)	Sin/Cos*	CR and sensor commissioning, will spin to max test speed parameter in one direction.
ACIM	–1	Any (i.e., quadrature)	The full auto-characterization routine. The motor will spin full speed in one direction.

** Consult Curtis to use a sin/cos sensor for induction motors.*

The motor setup warning will clear when slip gain and base speed have both been manually set.

Table 21 Motor Characterization and Parameter Mismatch Error Types

VARIABLE	RANGE	DESCRIPTION
Motor Characterization Error *Motor_Characterization_Fault_Type* `0x2850 0x06` Reference Fault 8-7 (0x87) in Table 24, the Fault Code Troubleshooting Chart, in Chapter 7.	0 – 506 *0 – 506*	**Fault Types** 0 = No motor characterization errors. 71 = Encoder signal seen but step size not auto-detected; setup Encoder Steps manually. 72 = Motor temp sensor fault. 73 = Motor hot. Motor temp hot cutback fault. 74 = Controller overtemp cutback fault. 76 = Undervoltage cutback fault. 77 = Overvoltage cutback fault. 78 = Encoder not reading properly. The Encoder signal not seen, or one or both channels are missing. 79 = Current Regulator Tuning out of range. 80 = Current Regulator Tuning out of range. 81 = Encoder signal seen but step size not auto-detected; set up Encoder Steps manually. 82 = Aborted commissioning. Autochar disabled while running. Must cycle KSI. 83 = Sin/Cos signal too noisy for characterization. 84 = Motor not rotating, Sin/Cos sensor voltages out of spec, or Multiturn Sensor setting incorrect. 85 = Sin/Cos signal too noisy for characterization. 86 = Sin/Cos sensor missing or sensor voltage out of range. 87 = PMAC Motor Type must be set before commissioning. 88 = PMAC motor fell to zero speed, check the system for excessive friction or loading, retry with a higher test speed, or consult Curtis. 91 = PMAC motor not rotating or motor type incorrect. 92 = PMAC motor not accelerating. Low acceleration. 93 = Started characterization procedure while motor rotating. 94-98 = PMAC lag compensation out of range. 99 = PMAC motor not accelerating. Low acceleration. 102 = PMAC motor temp sensor fault. 103 = PMAC motor temp hot cutback fault. 104 = PMAC controller temp cutback fault. 106 = PMAC undervoltage cutback fault. 107 = PMAC overvoltage cutback fault. 108 = Commissioning stopped by user. 500 = The Hall patterns do not match the pattern table during refining process. 501 = Hall patterns and angles are not consistent during rebuilding sectors in reverse direction. 502 = The rebuilt angle in reverse direction does not align to the calibrated angle. 503 = Hall patterns and angles are not consistent during rebuilding sectors in forward direction. 504 = The rebuilt angle in forward direction does not align to the calibrated angle. 504 = The controller does not get enough Hall switch pulses. 505 = The Hall switch patterns are not consistent. 506 = Invalid patterns are detected.
Parameter Mismatch *Parameter_Mismatch_Fault_Type* `0x2812 0x06`	0 – 100 *0 – 100*	**Fault Types** 0 = Dual Drive is setup incorrectly. Speed Mode or Speed Mode Express must be used, EM Brake Type must be 2, Dual_Drive_Mode_Type must be 1. 4 = EM Brake Control Mode is invalid.

Table 21 Motor Characterization and Parameter Mismatch Error Types, cont'd

VARIABLE	RANGE	DESCRIPTION
		Fault Types 5 = Interlock Brake Control Mode is invalid. 6 = PMAC_Short_Circuit_Current set above Base_Current_Limit in a non-test mode. 7 = In a differential steer system, fault actions are misconfigured. 8 = Dual motor type must be Differential. 9 = PMAC EMF Restriction - In a PMAC application configured for restricted mode operation, the back EMF per speed value is not configured. 10 = PMAC Release - A restricted and test mode for PMAC is being used in released software. 11 = Torque preload is configured to be saved across key-cycles, but EM Brake preload torque is not set. 12 = Invalid Torque Estimate - Configured torque estimation type is incompatible with the selected ""Direct Torque"" control mode. 13 = Command Map Stop - [STEERING] CommandMapLeftStop or CommandMapRightStop equals zero. 14 = [STEERING] Improper sequence of the redundant Command Analog map points. 15 = Analog Feedback Maps - [STEERING] Primary or Secondary Analog Feedback maps do not have continuous slope. 16 = Sawtooth Command - [STEERING] For Sin/Cos or Sawtooth Command device selection, the primary and secondary types do not match. 17 = Sawtooth Feedback - [STEERING] For Sin/Cos or Sawtooth Feedback device selection, the primary and secondary types do not match. 18 = Feedback Type - [STEERING] Autocenter is declared as Never and the feedback device type is a relative position device type. 19 = Interlock braking supervision must be enabled in PMAC if interlock braking is on. 20 = The selected motor type has not been fully characterized and should only be used for development. 21 = R_sys did not commission properly. Contact Curtis. 100 = Parameter integrity problem.

These next tests require the vehicle to be on the ground, able to operate/drive. Ensure there are no tools and objects on/near the traction, moving steering components/system, and any other vehicle wheels and tyres. Always operate the vehicle in a safe test area, where protection for personnel and property from un-controlled operation is verified. Ensure the vehicle brakes operate, including the EM Brake and Emergency Reverse, if so equipped.

13. Cycle (keyswitch) power. The motor control should now be operational, though likely poorly optimized. The SlipGain test is very important. Perform the SlipGain test and get the result. The following steps will complete the optimization process.

14. Run the **SlipGain** test (<u>do not skip this critical step</u>). This test requires the vehicle to be on the ground, able to apply traction torque to the road.

    Adjust the SlipGain parameter to provide maximum torque at stall (0 rpm motor speed). Note that the present SlipGain value in Programmer was determined by the auto-characterization procedure, and is not the default value.

    Programmer: Motor Setup » Induction Motor (ACIM) » Characterization Tests » ***SlipGain****.*

To determine the SlipGain setting, use either of the following two methods. The load cell/draw bar is the most accurate.

14.1 Configure the vehicle to measure stall torque, by using a load cell/draw bar test. Note: Perform this test quickly to avoid excessive motor heating and provide the most accurate results.

- First, turn off encoder fault detection.

Programmer: Motor Setup » Quadrature Encoder » Encoder Fault Setup » Fault Detection Enable (= Off).

Test: While applying and holding full throttle, adjust SlipGain until the peak torque is measured.

When the SlipGain test is completed, re-enable encoder fault detection:

Fault Detection Enable = On.

14.2 Alternatively, use the vehicle as the "drawbar test" load by accelerating the motor to a predetermined speed (rpm)—with the quickest time corresponding to the optimum SlipGain. This method uses the controller's time-to-speed function. The test-acceleration will be at full controller current.

Begin by setting the corresponding Accel Rates to very fast values, based upon the application's Control Mode.

*Speed Mode Express: **Accel Rate**.*
*Speed Mode: **Full Accel Rate LS, Full Accel Rate HS, Low Accel Rate**.*
*Torque Mode: **Accel rate**.*

Next, set the Capture Speed 1 parameter to an rpm encompassing the motor's desired maximum torque range. The monitor item, Time to Speed 1, will record the time taken for the vehicle (motor) to go from zero rpm to the programmed Capture Speed 1 during its most recent acceleration.

*Programmer: Application Setup » Vehicle » Speed/Dist/Accel » **Performance Metrics » Capture Speed 1**.*

*Programmer: System Monitor » Vehicle » Speed/Dist/Accel » **Performance Metrics » Time to Speed 1**.*

Test: From a stop, accelerate the vehicle through the predetermined speed. The timer starts counting when full throttle is applied, and stops counting when the speed is reached. A built-in trigger will allow the test to begin again once the vehicle comes to a stop and the throttle is reengaged, so be sure to note the value of the test first before accelerating away in subsequent runs. Each time, adjust the SlipGain value until the Time to Speed 1 is minimized.

Programmer: Motor Setup » Induction Motor (ACIM) » Characterization Tests » SlipGain.

Run this test repeatedly over the same stretch of flat or uphill ground. Loading the vehicle will improve results, the idea being to have the motor produce maximum torque as in the drawbar method. Work quickly, to avoid excessive motor heating and to obtain the most accurate results.

15. Run the **FW Base Speed** test (do not skip this critical step).

Set the FW Base Speed to 6000 rpm. Run the test with batteries that have a reasonable charge.

Programmer: Motor Setup » Induction Motor (ACIM) » Field Weakening » FW Base Speed.

Set the Accel Rates to be very fast, so that the vehicle will be accelerating at full current during the test. From a stop, quickly apply full throttle to accelerate to a high speed. After the motor settles at a speed for a moment, release the throttle to stop the test run. Note the value of the Base Speed Captured and enter this value for the application's FW Base Speed setting.

Programmer: Motor Setup » Induction Motor » Field Weakening » Base Speed Captured.

Note that the base_speed_capture test restarts each time the motor comes to a stop and the throttle is released, so be sure to note the value before re-accelerating the motor.

If the Motor Type or the low speed current limit are changed after performing the initial setup procedures, the FW Base Speed parameter also needs to be reset.

For example, if lowering the *Drive_Current_Limit* or *PL_Drive_Nominal*, retest and reset this parameter.

16. Run the Field Weakening test.

This test will enable the choice between a more efficient (less torque) or more torque (less efficient) operation above the base speed, by adjusting the Field Weakening Drive parameter.

Programmer: Motor Setup » Induction Motor (ACIM) » Field Weakening » Field Weakening Drive.

Note: The present Field Weakening Drive value (0%) was determined by the above autocharacterization routine, and is not the default value (100%) in the native OS intended for the Curtis dynamometer-characterized motors when a motor type (number) is utilized. The differences are:

Auto-Characterized: Field Weakening Drive is adjusted UP from 0% to the max Field Weakening Drive determined in this test.

Dyno-Characterized: Field Weakening Drive is adjusted DOWN from 100% (can be set from 0–100%).

Increasing this parameter toward 100% will progressively allow more torque (along with more current) at high motor speeds, but increasing it too much will actually reduce torque while still drawing a great deal of current. The auto-characterization value of 0% will give the most efficient motor operation, but will not give the highest torque at high speeds.

If more torque is required at high speeds, run this Field Weakening Drive test. This test will use these items:

*Programmer: Application Setup » Vehicle » Speed/Dist/Accel » **Performance Metrics » Capture Speed 1, 2**.*
*Programmer: System Monitor » Vehicle » Speed/Dist/Accel » **Performance Metrics » Time Between Speeds**.*

Set ***Capture Speed 1 and 2*** to values that are close to the Max Speed setting. The *Capture Speed 1* and *2* values MUST be set higher than the measured *FW Base Speed* in step 15 (because Field Weakening only affects performance at speeds above the "base speed"). Ideally, the *Capture Speed 1, 2* values should be above the second corner point of the motor torque – speed curve.

For example, if the Max Speed is set to 4000 rpm, set *Capture Speed 1 = 3500 rpm* and *Capture Speed 2 = 3800 rpm*:

*Programmer: Speed Mode Express » **Max Speed**.*
*Programmer: Speed Mode » Speed Controller » **Max Speed**.*
*Programmer: Torque Mode » Speed Controller » **Max Speed**.*

From a stop, apply full throttle to accelerate to a speed greater than Capture Speed 2. Note (write down) the value displayed for Time Between Speeds (Time_Between_Capture_Speeds). The test progression is to minimize this time.

Now increase the Field Weakening Drive setting and repeat the acceleration in the same direction, with the same load, and with the same full throttle and again note the Time Between Speeds value.

Keep adjusting the Field Weakening Drive value and repeating the test until you find the Field Weakening Drive value that results in the smallest Time Between Speeds. Use this Field Weakening Drive value for applications that require the most torque at high speeds.

Some Field Weakening Drive settings may result in a test acceleration run where the Capture Speed 2 cannot be obtained. These Field Weakening Drive settings cannot provide enough torque to get the vehicle past Capture Speed 2, and thus can be ignored.

Use the Field Weakening Drive setting that best matches the application. It will be somewhere between 0% (most efficient, least motor heating) and the setting found in this test (highest torque at high speeds).

Note: The Field Weakening Drive setting will have no effect at motor speeds below FW Base Speed.

AUTOMATED PMAC (MOTOR) COMMISSIONING PROCEDURE

⚠ WARNING The motor will rotate during this procedure. Do not take the vehicle down off the blocks.

When commissioning PMAC motors, the parameters cannot be copied across from EOS to FOS for PMAC. Use these instructions for the present commissioning process for PMAC motors. The default cdev (FOS 4.0) will show the ***Motor Setup Needed*** fault will be present. This fault must be cleared to allow motor operation by following the below PMAC commission procedures.

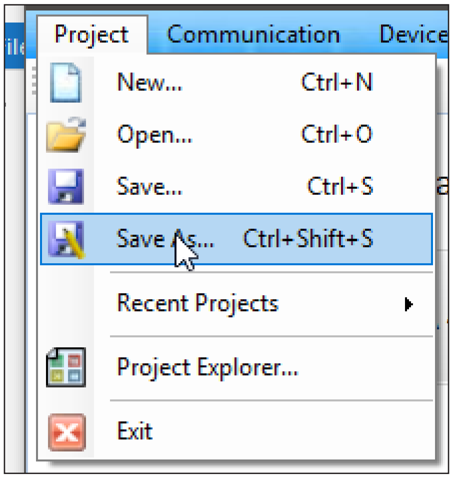
The *Motor Setup Needed* fault **Type** will indicate the tests that need to be run. Bit positions are:

Bit Positions	Description
1	Current regulator tuning.
4	Full automated test run.

Prior to performing the commissioning routine, set up and save this controller/motor as a CIT project.

Save the project: CIT» Launchpad » Project tab/ *Save As*… option.

Be sure to navigate to a defined file folder location to save this ".cprj" file.

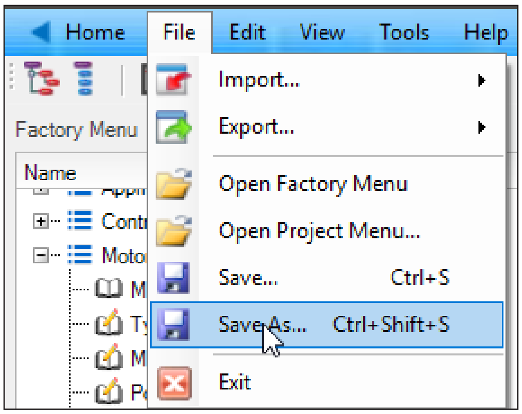


After saving the CIT project, now save the parameter changes (to date) as the Project menu.

In Programmer, under the File tab, save the Project Menu using the ***Save As***… option.

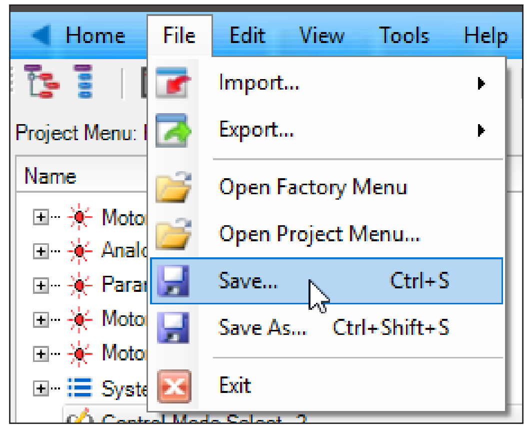
This transitions the initial Factory menu to the option of loading and saving the Project menu.

Then load and work within the PROJECT MENU for the PMAC commissioning.



Saving the *Factory menu* as a *Project menu*

Then use the Save... option (Ctrl-S) as changes occur in the project



Saving the Project menu (and project) is required, so setting the project and project menu beforehand will ensure the data is properly saved as part of the PMAC commissioning. The project menu (.cmnu) and a TACT trace (.ctrc) files are sent to the Curtis support engineer to complete the PMAC commissioning.

Note: Familiarization with CIT Launchpad, Programmer, and TACT are required for PMAC commissioning.

PMAC Commissioning Routine:

1. Within the CIT Programmer app, navigate to the Motor Setup menu and adjust these parameters.

    Motor technology (1 = PMAC),

    Position Sensor Type (2 = Sin/Cos encoder),

    Check the Sin/Cos Encoder parameter *Enable Multiturn Sensor* to the value that matches the application,

        1 = ON, 1 encoder revolution per electrical cycle.

        0 = OFF, 1 encoder revolution per mechanical cycle.

    Temperature Sensor (KTY84-130 = Type 2, PT1000 = Type 4).

    *Programmer: Motor Setup » Motor Technology: **1 = PMAC**.*
    *Programmer: Motor Setup » Position Sensor Type: = **Sin/Cos encoder**.*
    *Programmer: Motor Setup » Sin/Cos Encoder » **Enable Multiturn Sensor** = 0 or 1.*

Be sure that the motor is free to spin without a load and that the motor/machine is secure. During commissioning, the motor will pulse and spin up to speed several times in both directions. This will take up to 10 minutes, and there can be some long pauses with the motor stalled. When complete, the Main Contactor will open and there will be an active fault.

2. Set **Typical Max Speed** to the expected Max Speed the motor will achieve in the application.

    *Programmer: Motor Setup » PMAC (Permanent Magnet Motor) » Commissioning Tests » **Typical Max Speed**.*

3. Set **Max Test Speed** as the Max Speed to run during the commissioning test (usually this is the same as Typical Max Speed), higher speeds = better results.

    *Programmer: Motor Setup » PMAC (Permanent Magnet Motor) » Commissioning Tests » **Max Speed Test**.*

4. Set **Max Test Current** for the commissioning.
   (This should not exceed the rated current of the motor, typically 10 – 20% of the maximum).
   Note – this is a percentage of the Drive Current Limit (*Controller Setup » Current Limits menu*).

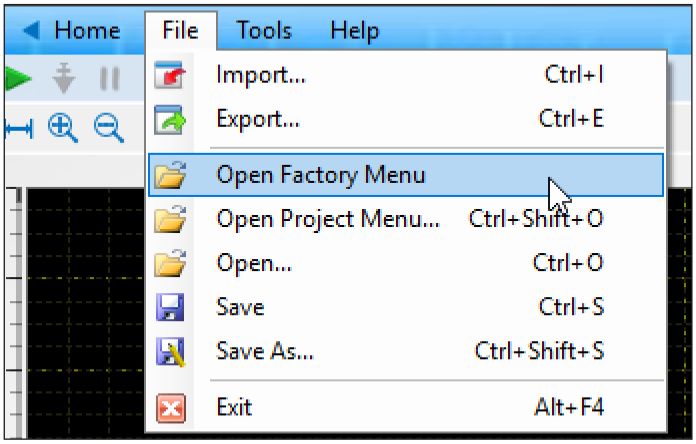
   Programmer: Motor Setup » PMAC (Permanent Magnet Motor) » Commissioning Tests » ***Max Test Current****.*

5. Enter the PMAC Motor Type.

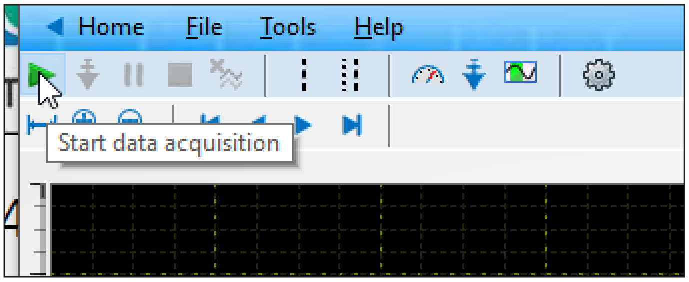
   Programmer: Motor Setup » PMAC (Permanent Magnet Motor) » ***Motor Type****.*

6. Open TACT (the CIT app) and import the PMAC commissioning template file "CTRC File". Obtain this PMAC *.ctrc* template file from the Curtis support engineer.

   6.1 Be sure to open the Factory menu (*TACT: File » Open Factory menu*). See image.



7. Engage Interlock. In TACT, start the data acquisition. See image.



⚠ WARNING

These next steps will cause the motor to move.

Do not take the vehicle down off the blocks.

8. In CIT Programmer, set **Test Enable** to ON, and then set **Test Throttle** to 1.

   Programmer: Motor Setup » PMAC (Permanent Magnet Motor) » Commissioning Tests » ***Test Enable*** *= On*
   Programmer: Motor Setup » PMAC (Permanent Magnet Motor) » Commissioning Tests » ***Test Throttle*** *= 1*

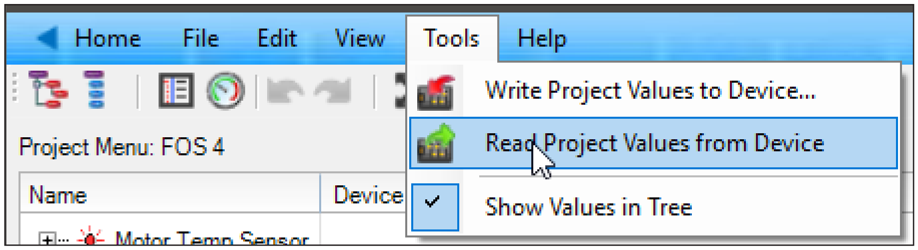
   The motor will spin during the commissioning.

   Do not interrupt the process or TACT during the commissioning.

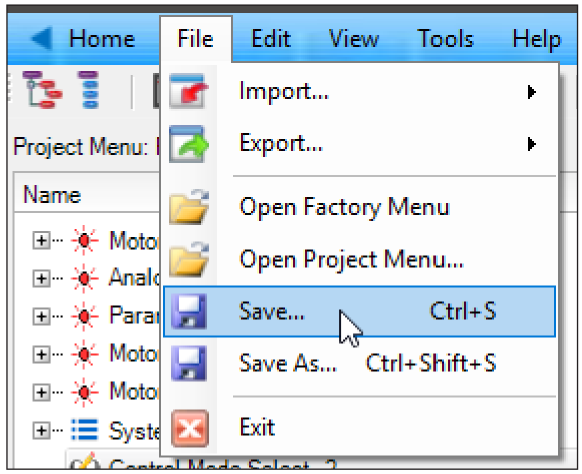
9. When the commissioning is COMPLETE a Parameter Change Fault will appear (in CIT Programmer), and the Main Contactor will open.

10. Stop the TACT data Acquisition and SAVE the CTRC file.
    The file is exportable to a PC file folder location (see the TACT File-tab image, above).

11. In Programmer, under the Tools tab, select the *Read Project Values from Device* option.



12. In Programmer, under the File tab, save the Project Menu.



13. Email the CIT Project Menu file (.cmnu) and the TACT trace file (.ctrc) to the Curtis support engineer.

When the test is complete, cycle the keyswitch (KSI) to clear the Parameter Change fault, and then attempt to drive the motor in the forward direction while monitoring the Motor Speed (the vehicle is still on the blocks). If the motor speed shows a negative value, invert the **Swap Direction** parameter in the Commissioning Tests menu (note, there is no swap two phases parameter like the induction motor control – swap direction takes care of both).

The test data files (emailed to Curtis) will determine the final PMAC commissioning.

AC MOTOR TUNING GUIDE

Once a vehicle/motor/controller combination is tuned, use the parameter values as the standard for the system or vehicle model. Any changes in the motor, the vehicle drive system, or the controller will require that the system be re-tuned to provide optimum performance. These steps are for tuning the AC traction motor, not an AC pump motor.

Hint: If using the 1313 handheld programmer, add the "tuning" parameters and monitor variables to the Favorites menu for quick access to these items. The equivalent in CIT is creating a dashboard.

Selecting the Control Mode

Before starting to tune your vehicle's performance, you must select which control mode to use. Set the Control Mode Select parameter = 0 (Speed Mode Express for traction motors), or = 1 (Speed Mode, for traction motors). Cycle KSI Off and then On (to clear any Parameter Change Faults) and use the CIT/1313 Programmer app to check for faults in the controller. Clear any faults, and then proceed to the tuning steps for the control mode selected.

Conduct the steps in the sequence given, because successive steps build upon the preceding steps. Understanding these programmable parameters is necessary to take full advantage of the controller's powerful features. See the descriptions of the applicable parameters in Chapter 4.

0 – Speed Mode Express Tuning

Quick Link:
Speed Mode Express p.51

Speed Mode Express is the same as Speed Mode with the exception that it has fewer parameters and is therefore simpler to use. Most vehicle applications will find success with Speed Mode Express; however, for some applications vehicle performance cannot be satisfactorily fine-tuned in Speed Mode Express. In this case, change the control mode to Speed Mode (i.e., set **Control Mode Select =1**).

0.1 Set the Pump Enable and Regen Lower Enable parameters to Off (do not have the pump turn on during the traction motor tuning).

0.2 Adjust Max Speed to the maximum speed the traction motor should turn in the vehicle application; this speed setting corresponds to an input of full (100%) throttle.

0.3 Adjust Typical Max Speed to the approximate maximum speed that the motor will spin. This is usually the same value as the setting for Max Speed, but some applications have a Max_Speed_SpdMx that changes (in the VCL software). If the Max_Speed_SpdMx changes, set Typical Max Speed to the highest speed the motor is expected to reach. This value does not need to be set precisely since it will not change motor performance. Typical Max Speed sets a reference point for the "rate" parameters (accel, decel, brake rates), so that applications that have a changing Max_Speed_SpdMx will not experience changes in the rates (because the rates are referenced to the unchanging Typical Max Speed value). Once the Typical Max Speed parameter is set, do not readjust it without adjusting all the rate parameters as well.

0.4 Kp and Ki typically do not need to be changed as the default values will work well in most applications. If you want to adjust Kp (for looser or tighter following of the speed trajectory set by the accel, decel, and brake rates), follow the procedure in step "1.4" in the Speed Mode tuning section.

0.5 Adjust the Accel Rate and Decel Rate as necessary while moving the throttle to different positions (i.e., neutral to full throttle, half throttle to full throttle, full throttle to half throttle, full throttle to neutral, neutral to low throttle, etc.).

0.6 Adjust the Brake Rate as necessary while reversing the throttle input (i.e., full throttle forward to low throttle reverse, full throttle forward to full throttle reverse, full throttle reverse to low throttle forward, etc.). If a brake input is present in the application (Brake Pedal Enable = On) continue adjusting Brake Rate by applying different amounts of brake throttle (i.e., full throttle forward, then apply full brake or full throttle forward, then apply low brake, etc.).

1 – Speed Mode Tuning

Quick Links:
Speed Mode p.52

1.1 Set the Pump Enable and Regen Lower Enable parameters to Off (do not have the pump turn on during the traction motor tuning).

1.2 Adjust Max Speed to the maximum speed the motor should turn in the vehicle application; this speed setting corresponds to an input of full (100%) throttle.

1.3 Adjust the Typical Max Speed to the approximate maximum speed that the motor will spin. This is usually the same value as the setting for Max Speed, but some applications have a Max_Speed_SpdM that changes (in the VCL software). If the Max_Speed_SpdM changes, set the Typical Max Speed to the highest speed the motor is expected to reach. This value does not need to be set precisely since it will not change motor performance. Typical Max Speed sets a reference point for the "rate" parameters (accel, decel, brake rates), so that applications that have a changing Max_Speed_SpdM will not experience changes in the rates (because the rates are referenced to the unchanging Typical Max Speed value). Once the Typical Max Speed parameter is set, do not adjust it further without adjusting all the rate parameters as well.

1.4 Kp typically does not need to be changed as the default value will work well in most applications. This parameter controls how tightly the actual motor speed will track the requested speed trajectory (speed trajectory is set by the accel, decel, and brake rates).

If seeking to adjust the Kp (for looser or tighter following of the speed trajectory), follow these guidelines:

Set the following parameters. Before setting them, make a note of their present (or default) settings so you can return them to these original values at the end of this procedure.

- In the Speed Mode » Speed Controller menu, set the Max Speed to a low value (≈1000 rpm), as high-speed operation is not needed to observe system response.
- In the Speed Mode » Response menu, set all the accel and decel rates to their fastest values (0.1 seconds); this allows better observation of the system response.
- Set the Soft Stop Speed parameter to 0 rpm to disable the soft stop speed function (see Restraint menu).
- In Speed Mode » Restraint » Position Hold menu, set Position Hold Enable = Off such that the position hold function will not interfere with the speed control gain setup procedure.
- If the vehicle has an EM Brake, set the EM Brake Type to 1. This setting releases the EM Brake as soon as the interlock is closed (Interlock = On). Note: After completing the fine tuning, reset the EM Brake Type to the final type (see the *Application Setup » EM Brake Control* menu).

Cycle KSI to clear any faults.

- Using very quick, pulsing throttle movements, increase the throttle and then release it to 0%. The intent is to give the speed controller torque impulses.
- Increase Kp and repeat the throttle tests. Increase Kp until you start to notice marginal stability (normally motor bouncing, or continuous oscillation in the gears, is heard).

  Note: It is possible that very heavy vehicles will not experience marginal stability even at the highest setting of Kp.

- Once the Kp setting for marginal stability is found, reduce the Kp value by about one-third (i.e., final Kp = marginal stability Kp * 2/3).
- If using Speed Mode Express, enter this Kp value for the Kp parameter in the Speed Mode Express menu.
- Set the Max Speed, Accel/Decel, Soft Stop Speed, Position Hold, and Brake Type parameters back to their original values.

1.5 In the Speed Mode » Response menu, adjust the Accel and Decel Rate parameters as necessary while moving the throttle to different positions (i.e., neutral to full throttle, half throttle to full throttle, full throttle to half throttle, full throttle to neutral, neutral to low throttle, etc.).

1.6 In the Speed Mode » Response menu, adjust the remaining Brake Rate parameters as necessary while reversing the throttle input (i.e., full throttle forward to low throttle reverse, full throttle forward to full throttle reverse, full throttle reverse to low throttle forward, etc.). If a brake input is present in the application (Brake Pedal Enable = On) continue adjusting these Brake Rates by applying different amounts of brake throttle (i.e., full throttle forward, then apply full brake or full throttle forward, then apply low brake, etc.).

1.7 The parameters in the Speed Mode » Response » Fine Tuning menu typically do not need to be changed as the default values work well in most applications.

2 – Torque Mode Tuning

2.1 Set the Torque Mode's **Max Speed** to the application's maximum motor speed (rpm).
Programmer: *Torque Mode » Max Speed (0x391D)*

2.2 Set the **Typical Max Speed** to the expected maximum motor speed (rpm).
Programmer: *Motor Setup » Typical Max Speed* (0x3543)

Quick Links:
Torque Mode p.59
Fine Tuning p.60
Figures 12–14 p.16-18
Current Limits menu p.127
Typical Max Speed p.130

2.3 The Torque Mode's **Kp**, **Ki**, and **Kd** parameters typically do not need to be changed as the default values will work well in most applications. These parameters control how tightly the controller limits the speed of the motor to the programmed Max Speed.
Programmer: *Torque Mode:*
Kp
Ki
Kd

2.4 Adjust the parameters shown in Figure 12 to set up the throttle mapping:
Programmer: *Controller Setup » Current Limits:*
Drive Current Limits
Regen Current Limits
Programmer: *Torque Mode:*
Neutral Braking
Neutral Taper Speed (Reference Fig. 13)
Forward Full Restraint Speed
Back Full Restraint Speed
Programmer: *Torque Mode » Fine Tuning:*
Creep Torque

2.5 In the Torque Mode menu, adjust the four Accel, Brake, and Release Rate parameters as necessary while moving the throttle to different positions (i.e., neutral to full throttle, half throttle to full throttle, full throttle to half throttle, full throttle to neutral, neutral to low throttle, etc.).

Programmer: *Torque Mode:*

Accel Rate (0x3902)
Accel Release Rate (0x3904)
Brake Rate (0x3907)
Brake Release Rate (0x3908)

2.6 Adjust the Gear Soften and Reversal Soften parameters to match vehicle driveline tolerances to the application. Reference Fig. 13.

Programmer: *Torque Mode » Fine Tuning:*

Gear Soften
Reversal Soften

2.7 Adjust the remaining parameters in the Torque menu to fine tune the Torque Mode to the application. Read the parameter descriptions and adjust as necessary.

Programmer: *Torque Mode » Fine Tuning:*

Brake Full Creep Cancel
Creep Build Rate
Creep Release Rate
Max Speed Decel

Other Parameter Tuning

Quick Link:
Vehicle menu p.94

Set the Speedometer/Odometer. Refer to the *Application Setup » Vehicle* menu.

Helpful hints:

(1) Set the **Speed to RPM** parameter by using the **Distance Since Stopped** variable to cover a known distance (e.g., 50 feet). Adjust the Speed to RPM parameter until the Distance Since Stopped value matches the known distance.

(2) Use the **Distance Since Stopped** variable to verify the vehicle tire's rolling diameter based upon a rollout measurement for the tire/wheel combination and the Speed to RPM parameter.

- Tire Circumference (C) = Distance/wheel-revolutions.
- Diameter (d) = C/Π, where Π (pi) = 3.14159.

SETTING UP THE HYDRAULIC SYSTEM

Before beginning the setup procedures for the hydraulics (contactor driven DC pump motor):

- Check that the hydraulic system wiring is consistent with the wiring guidelines presented in Chapter 2.
- Review the Hydraulics and Controls parameters in Chapter 4, including Figures 28-31

  Note: Due to changing device profiles, some parameters/variables may be in the CIT *List View* ( ).

Quick Links:
Fig. 28 p.89
Fig. 29 p.90
Fig. 30 p.91
Fig. 31 p.91

- Confirm that the hydraulic system is consistent with the system diagram shown in either Figure 28 or 29. If it is not, then the following commission guide may not be fully applicable.
- Make sure all electrical and hydraulic connections are tight, and the hydraulic fluid filled to the appropriate level.
- Check that any forks are free to rise and lower. The same applies to any optional hydraulic actuators.
- Turn off the traction interlock while setting up the hydraulic system. If the vehicle is on blocks, ensure that if the forks are lifted and lowered, the vehicle remains stable.

HYDRAULIC LIFT AND LOWER COMMAND INPUTS

Several methods are available to control the hydraulic pump motor and lowering valve(s) comprising the hydraulic system.

- Lift and Lower input switches.
  - Lift and Lower commands are single inputs for an On/Off operation. See steps 1–8.
  - Lift is an On/Off operation, yet the Lower is a variable throttle for a proportional valve. See steps 9–20.
- Individual Lift and Lower throttle voltage inputs.
  - Voltage input determines the lift or lower operations.
- CAN and/or VCL.

When setting up the hydraulic system, reference Figure 40 (below) for the proportional driver signal chain. The diagram lists the configurable parameters, VCL function/variables, and the run-time variables (monitor variables) throughout proportional driver signal chain.

For configuring the load hold valve, reference Figure 41 (below). The diagram lists the signal chain inputs throughout, the configurable parameters, VCL function/variables, and the run-time variables (monitor variables).

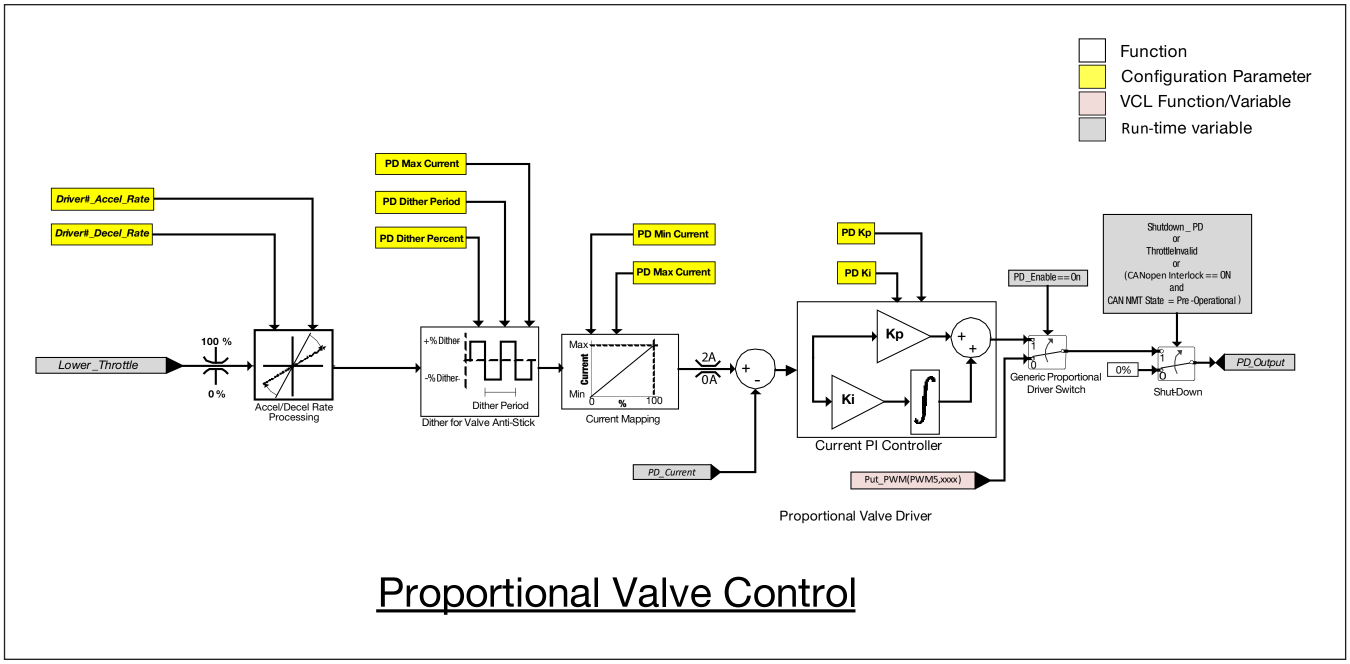


Figure 40

The Proportional Valve Signal Chain

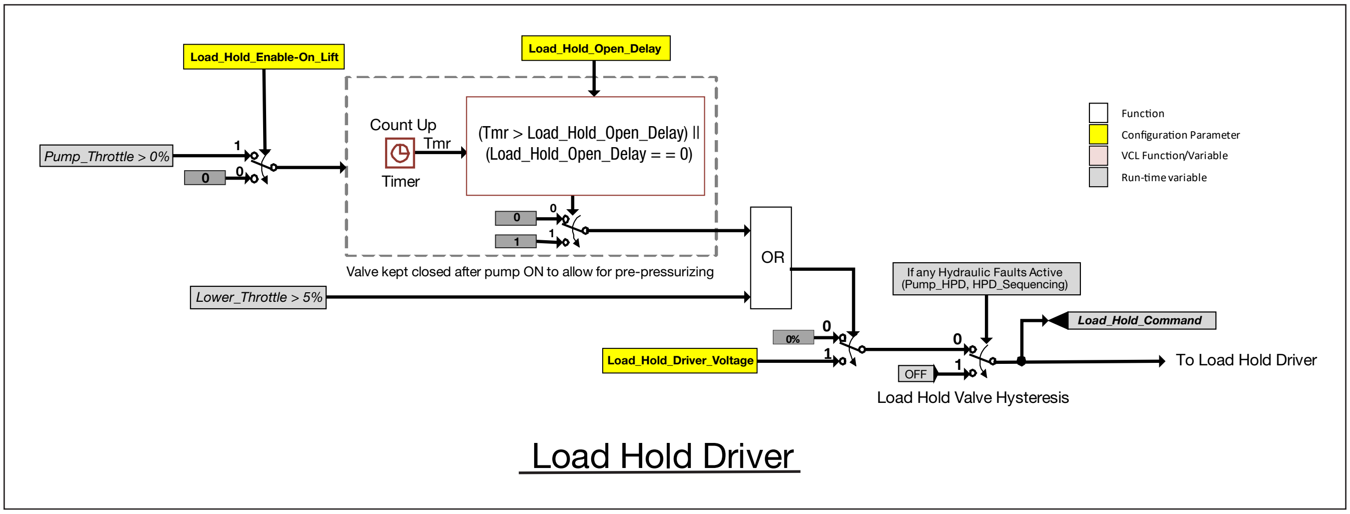


Figure 41

The Load Hold Valve Signal Chain

Lift & Lower Switch Inputs

This method uses the On/Off switch inputs to command the hydraulic throttle (Lift). Another On/Off switch commands the Lower operation, where a designated coil driver is either fully off (load hold) or fully on (lowering). In this method, the hydraulic throttle and lower are non-variable inputs. This is the illustrated "step function" (_⌐) in the figures 30 and 31 Throttle/Switch Mapping. In this example, reference Figures 13 and 29-31. The Lift switch is Input 10 (pin 10). The Lower switch is Input 11 (pin 11). A load hold valve is used.

Quick Links:
Fig. 13 p.17
Fig. 28 p.89
Fig. 29 p.90
Fig. 30 p.91
Fig. 31 p.91

Lift Command Switch

1. Set the *Lift Input Type* to either Normally Open (NO) or Normally Closed (NC).
2. Assign the switch (input) number for the *Lift Input Source* parameter.

Controller Setup » IO Assignments » Controls » Lift Input Type & Lift Input Source

In Programmer, notice that with the lift-switch asserted, the Switch 10 status is On and the Lift Input, Mapped Lift Throttle, and Lift Command monitor variables are 100%. The hydraulic pump motor will be fully on.

Application Setup » Hydraulics » Lift Settings

Lower Command Switch

3. Set the *Lower Input Type* to either Normally Open (NO) or Normally Closed (NC).
4. Assign the switch (input) number for the *Lower Input Source* parameter.

For the example in Figure 13, the Lower is on Input 11 (pin11) as an NO switch input.

Controller Setup » IO Assignments » Controls » Lower Input Type & Lower Input Source.

5. Using the CIT Programmer app's List View ( ), assign the *Lower_Driver* parameter (CAN Index 0x4FCC 0x00) to the driver operating the hydraulic lowering valve (for example, Driver 3). This will be the driver for the lowering valve.

   In Programmer, notice that with the lower-switch asserted, the switch status is On and the Lower Input, Mapped Lower Throttle, and Lower Command monitor variables are 100%. The lower-valve (Driver 3) will be fully On (note that if the Lift Command was previously On [100%], it is now Off [0%]). The lower command takes precedence over the lift command.

Application Setup » Hydraulics » Lower Settings

To prevent an accidental turn-on of the hydraulic system and peripherals at startup/key-cycle, adjust the parameters ***Pump Interlock Source***, *Hydraulics_Inhibit_Type*, and ***Sequencing Delay*** to match the application.

6. To specify that the Lift and Lower commands are only by switches, ensure that parameter

   Pump Interlock Source = 0

   Note the default value is for switch inputs; hence, this parameter was not set in the above steps.

   Pump Interlock Source (0x4FDA 0x00).

7. Set the *Hydraulics_Inhibit_Type* (0x3702 0x00) to match the application. This parameter is in the CIT Programmer "flat list". The options are,

   0 = Diagnostics are disabled.
   1 = Lift Only. The lift throttle is stuck high (> 25%) at startup.
   2 = Lower Only. The lower throttle is stuck high (> 25%) at startup.
   3 = Both. The Lift or the Lower throttles are stuck high (> 25%) at start up.

The hydraulics inhibit fault diagnostics can detect a stuck-high hydraulic throttle condition (>25%) at startup. For the switch inputs, this occurs when the switch is asserted (i.e., 100%). The diagnostics are configurable (as above) to detect either the lift throttle chain, lower throttle chain or both. The diagnostics begin execution after a configurable delay set by the *Sequencing Delay* after a KSI turn-on (cycle). The diagnostics stop execution after a period of 64msec. This startup delay is for allowing momentary KSI disruptions or quick KSI cycling. Prior to the inhibit fault diagnostic, the controller processes the *Sequencing Delay* parameter.

8. *Sequencing Delay* works to prevent inadvertent activation of HPD/SRO, which are traction throttle based. To account for the *Hydraulics_Inhibit_Type*, add 64ms to compute the total delay before the hydraulic operation will commence.

Application Setup » Throttle » Sequencing Delay

Lift switch input, with a proportional lowering valve

Lift Command Input

9. Set the *Lift Input Type* to either Normally Open (NO) or Normally Closed (NC).
10. Assign the switch (input) number for the *Lift Input Source* parameter.

Controller Setup » IO Assignments » Controls » Lift Input Type & Lift Input Source.

Proportional Lower Command

Quick Link:
Fig. 31 p.91

For a proportional lower operation, setup the hydraulic throttle for the proportional valve. For the F-Series controllers, this means using an analog input assigned to the Lower operation (see Figure 31). This example will use Analog 18, Pot 18 Wiper, as a 2-wire lower hydraulic throttle. A voltage throttle is similar, using the Analog 18 voltage setting versus 2-wire.

11. Set the Analog 18 Type to a 2-wire pot.

Analog 18 Type = 2-wire

Controller Setup » Inputs » Analog 18 Type

11.1 Set the Potentiometer 18 to the nominal resistance of the 2-wire potentiometer.

Nominal Resistance = 5000 Ohm (i.e., a typical pot resistance)

Controller Setup » Inputs » Potentiometer 18

12. Set the Lower Input Type and Lower Input Source.

Lower Input Type = 2 (Voltage Input)
Lower Input Source = 18 (corresponds to the analog 18 input, pin 17)

Controller Setup » IO Assignments » Controls » Lower Input Type & Lower Input Source.

13. Set the Lower Settings to match the 2-wire voltage responses for the application.

Lower Min Input
Lower Max Input
Lower Map Shape
Lower Offset
Lower Filter

Application Setup » Hydraulics » Lower Settings (menu)

14. Assign the *Lower_Driver* parameter (CAN Index 0x4FCC 0x00) to the driver operating the hydraulic lowering valve, Driver 1. This is the proportional driver. Set the other driver parameters to match the application following the valve manufacturer's ratings. The hydraulic system configuration will be as illustrated in Figure 28.

    Lower_Driver = 1
    Lower_Deadband_Percent = 15% (default)
    Lower_Decel_Rate = 0.1 sec (default)
    Lower_Offset = 0% (default)
    Driver_1_Dither_Percent
    Driver_1_Dither_Period

15. If the application uses a hydraulic load hold valve, set the Load Hold Valve Enable On Lift parameter On; otherwise set it Off.

    Application Setup » Hydraulics » Load Hold Valve Settings » Load Hold Valve Enable On Lift

16. Set the load hold driver to 3.

    Load Hold Driver = 3

    Controller Setup » IO Assignments » Coil Drivers » Load Hold Driver

    To prevent an accidental turn-on of the hydraulic system and peripherals at startup/key-cycle, adjust the parameters Pump Interlock Source, *Hydraulics_Inhibit_Type*, and the *Sequencing Delay* to match the application.

17. To enable the Lift via switches and Lower via Analog 18, set the pump interlock parameter.

    Pump Interlock Source = 0

    CIT List View » Pump Interlock Source

18. Set the *Hydraulics_Inhibit_Type* to match the application. This parameter is in the CIT Programmer "flat list".

    0 = Diagnostics are disabled.
    1 = Lift Only. The lift throttle is stuck high (> 25%) at startup.
    2 = Lower Only. The lower throttle is stuck high (> 25%) at startup.
    3 = Both. The Lift or the Lower throttles are stuck high (> 25%) at start up.

    The hydraulics inhibit fault diagnostics can detect a stuck-high hydraulic throttle condition (>25%) at startup. For the switch inputs, this occurs when the switch is asserted (i.e., 100%). The diagnostics are configurable (as above) to detect either the lift throttle chain, lower throttle chain or both. The diagnostics begin execution after a configurable delay set by the *Sequencing Delay* after a KSI turn-on (cycle). The diagnostics stop execution after a period of 64msec. This startup delay is for allowing momentary KSI disruptions or quick KSI cycling. Prior to the inhibit fault diagnostic, the controller processes the *Sequencing Delay* parameter.

19. The *Sequencing Delay* works to prevent inadvertent activation of HPD/SRO, which are traction throttle based. To account for the *Hydraulics_Inhibit_Type*, add 64ms to compute the total delay before the hydraulic operation will commence.

    Application Setup » Throttle » Sequencing Delay

Wigwag Throttle

Set the Lift and Lower throttles to match the wigwag voltages. For a 2-wire throttle with the neutral at the center voltage, typical assignments are in the ensuing table. A voltage throttle is similar, using the Analog 18 voltage setting versus 2-wire. Notice that the throttle parameters are by percentage, not voltage.

Throttle/ Region	Percentage	0 – 5 Volts	0 – 10 Volts
Lower deadband	< 10 %	0.0 – .49V	0 0 – 1.0V
Lower Throttle Command	10 – 45 %	0.50 – 2.25V	1.0 – 4.5V
Middle deadband	45 – 55 %	2.25 – 2.75V	4.5 – 5.5V
Lift Throttle Command	55 – 90 %	2.75 – 4.5V	5.5 – 9.0V
Lift deadband	> 90 %	4.5 – 5.0V	9.0 – 10.0V

20. Lift Throttle settings (from the neutral throttle voltage to the maximum).

    Lift Min Input = 55%
    Lift Max Input = 90%
    Lift Map Shape = 35% (default, adjust for lift-throttle response, see Figure 30)
    Lift_Deadband_Percent = 55%
    Lift_Offset = 0%

21. Lower Throttle settings (from the neutral throttle voltage to the maximum).

    Lower Min Input = 45%
    Lower Max Input = 10%
    Lower Map Shape = 35% (default, adjust for lower-throttle response, see Figure 31)
    Lower_Deadband_Percent = 10%

Lower Proportional Driver

22. The lower driver parameter is setup as in the steps above. Assign the Lower_Driver parameter (CAN Index 0x4FCC 0x00) to the driver operating the hydraulic lower valve, Driver 1. This is the proportional driver. Set the other driver parameters to match the application following the valve manufacturer's ratings. Refer to Figure 28.

23. Complete the wigwag throttle by adjusting the remaining hydraulic throttle parameters as stated in steps 15-18 (above).

Wigwag with Lift and Lower Switches

If the application uses a wigwag throttle similar to the Curtis ET-126 throttle, use the output switches to command the lift and lower throttles. Since this is a single voltage throttle (each direction outputs 0 – 5 volts), connect the switch outputs as the lift and lower switch inputs in Figure 17, then process the switch status inputs in VCL.

Lift = Switch 10:	*Switch_10 = On or Off*
Lower = Switch 11:	*Switch_11 = On or Off*

Process the throttle via Analog 18, similar to a voltage throttle.

Lift Input Type = 2 (Voltage input)
Lift Input Source = 18 (unless the value of *analog_input_18* is processed in VCL).
Lower Input Type = 2 (Voltage input)
Lower Input Source = 18 (unless the value of *analog_input_18* is processed in VCL).

Unless the value of *analog_input_18* is processed in VCL, complete the Pump, Lift, Lower, and Load Hold Valve parameters as matches the application.

Application Setup » Hydraulics » Pump Settings
Application Setup » Hydraulics » Lift Settings
Application Setup » Hydraulics » Lower Settings
Application Setup » Hydraulics » Load Hold Valve Settings

Hydraulic System Tuning

To further tune the Lift response, adjust the Pump Accel Rate and Pump Decel Rate.

Application Setup » Hydraulics » Pump Settings

To further tune the Lower response, adjust the Lower Accel Rate and Lower Decel Rate.

Application Setup » Hydraulics » Lower Settings

If a bump is felt at the end of Lift or Lower operations, increase the Load Hold Opening Delay value to allow the hydraulic fluid to stop flowing before the load hold valve closes (Figure 28).

Application Setup » Hydraulics » Load Hold Valve Settings

Set the Lift Battery Lockout and *Hydraulics_Inhibit_Type* parameters as required by the application.

Application Setup » Hydraulics » Lift Settings

7 — DIAGNOSTICS AND TROUBLESHOOTING

The Troubleshooting Chart, Table 24, describes the fault flash codes. The faults list is in the numerical flash code order. The fault's name, VCL name and CAN Object Index are listed, including the fault types. Possible causes, the set and clear conditions and the fault actions are listed. Fault actions are "what the controller will do" when the fault is active. Table 23 lists the fault actions and their associated 'bits' when using the *System_Action* variable (0x4E00 0x00). Table 22 is a quick index to the faults by flash codes.

Faults, after the set condition(s) have been resolved, are usually cleared by cycling the keyswitch (On-Off-On), yet can be handled using the applicable VCL Reset function, or the CAN NMT Reset function. These are collectively referred to as *Reset Controller*.

THE DIAGNOSTICS PROCESS

Obtain diagnostics information in either of three ways: (1) by observing the fault codes flashed by the controller's status indicator, (2) by reading the indicated fault (●) in the *Curtis Integrated Toolkit™* Programmer tool, or (3) the CAN Emergency Messages.

The status indicator is a translucent window on the cover, which blinks red and yellow LEDs. Its illumination indicates the following information:

Off: Controller is not powered on, or is severely damaged.

Slow yellow blinking: Controller is operating normally.

Solid yellow or orange: Controller is in flash program mode, or corrupted software is preventing the unit from completing the startup sequence (boot process).

Red/yellow flashing pattern: Fault code, review the fault table.

Solid red: Internal hardware fault detected by the supervisor or primary microprocessor, or the controller has no software loaded.

Fast flashing red: Non-production/experimental/custom device profile software.

The Troubleshooting Chart Index, Table 22, indicates both the controller's LED flash sequences and the CANbus Emergency Message hexadecimal Error Code values. The fault flash pattern is decimal. For example, the **Driver 1 Fault** is assigned the flash code 10-1, which illuminates in a sequence of 10 red LED flashes (**x10**) followed by 1 yellow LED flash (**x1**) before repeating. The hexadecimal code is 0xA1, which will be the Error Code value in the CAN Emergency message, explained below.

When using the *Curtis Integrated Toolkit™* Programmer tool to diagnose faults, the active faults are at the top of the menu panel. If a fault has more than one possible cause, it indicates the "type" number. Within the System Monitor menu, the Fault History menu provides additional information; *Count* indicates the number of occurrences since the *Clear History* was performed. The menu item *Time* indicates the KSI hours, provided in seconds (#sec/3600 = hours) when the present fault occurred, while the menu's *First Time* item keeps track of the fault's first occurrence, should multiple faults follow before the history is cleared. The *Type*, as listed at the top of the menu panel, is repeated and is the current "fault type" for faults having more than one type. Figure 42 is an example of the CIT Programmer's fault-diagnosis usage. The illustrated sequence, panels left to right, are for the two types of the **Main Contactor Did Not Close** fault, which are also illustrated in the CAN Emergency Messages – Active Faults section, as examples (2a) and (2b) below.

❶ Panel 1: An Active Fault with Type 1. First occurrence (*Count* = 1) with Time and *First Time* are equal. The active fault *Type* matches.

❷ Panel 2: Second occurrence of the same fault *Type* as in panel 1. Note the *Time* differences, with *First Time* matching between panels 1 and 2.

❸ Panel 3: Same fault, yet with a different *Type* (Type = 2). Third occurrence (Count = 3) of the fault, with Time progression and present Type matching.

❹ Panel 4: No active fault, yet data of the last occurrence is preserved in the Fault History menu.

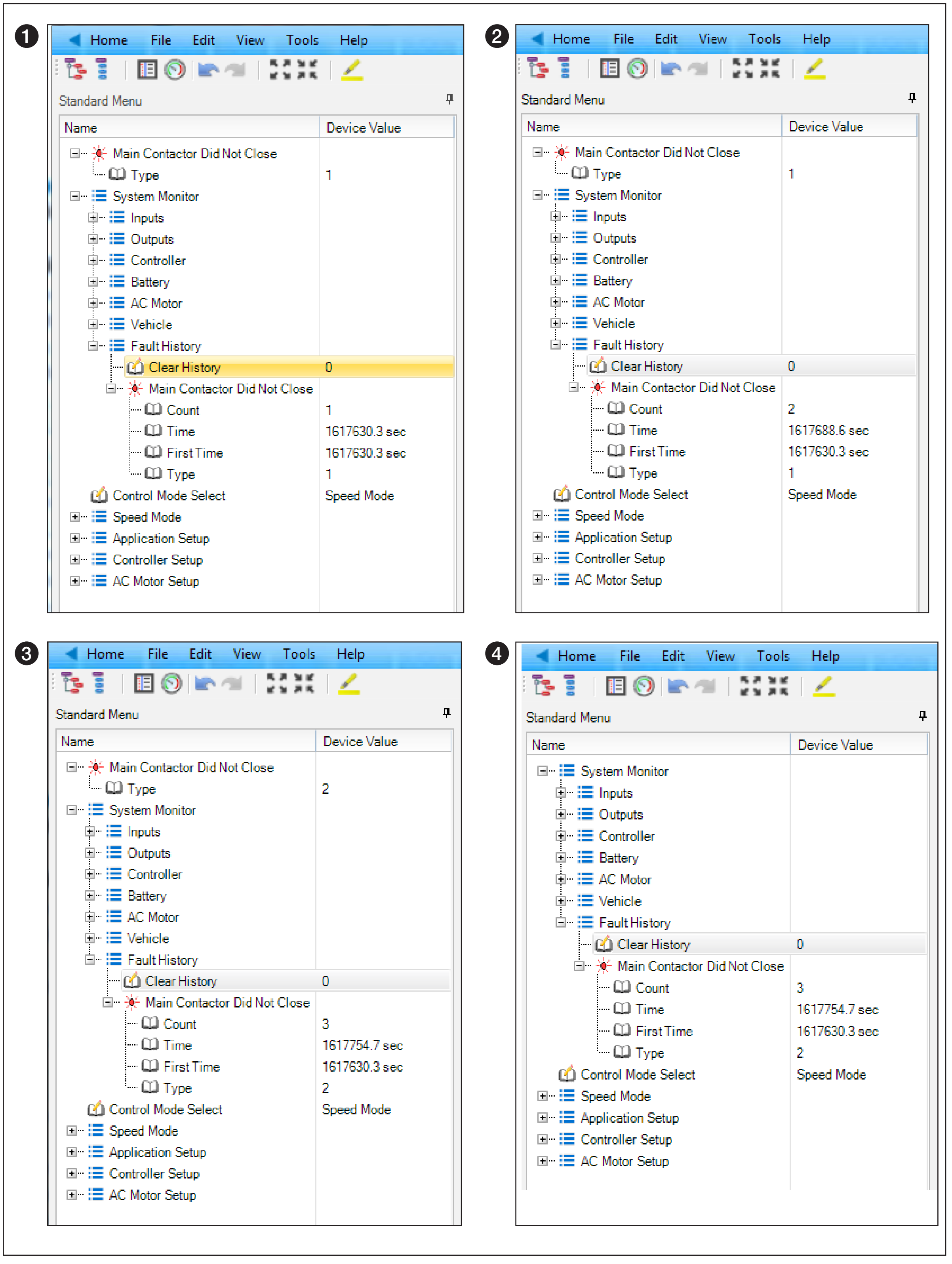


Figure 42

Curtis Integrated Toolkit™ Programmer Present and Fault History Sequence Example

Faults detected by VCL code (i.e., **User 1-32 Fault**) cannot be defined in Table 24 as they will vary from application to application. OEM/Factory user faults are as defined by the user and documented in the application's VCL program. Each of the 32 User Faults can be assigned the controller's defined single or multiple fault actions, or take actions as defined by the User (VCL Program). Refer to the appropriate OEM documentation for information on these faults.

All faults, VCL references (i.e., *Fault_Name_Active*) are assigned a value in VCL: 0, 1, or 2. Each *value* has a specific meaning:

Main Contactor Did Not Close *Main_Contactor_Did_Not_Close_Active*	0 = Never set. 1 = Fault has occurred in the Fault History menu. 2 = Fault is active, and is in the Fault History.
User_1_Fault_Active	

CAN Emergency Messages — Active Faults

Emergency messages are high priority CANopen objects that indicate the controller (the transmitting device) has detected or cleared a fault.

Emergency messages are sent at the occurrence of each new fault (they are not cyclic), but no faster than the Emergency Inhibit Time parameter setting allows*.

If several faults occur within the Emergency Inhibit Time or faster than the controller can transmit them, then they are stored in a queue and transmitted when possible.

When all active** faults are cleared, a final Emergency message is sent indicating the "all clear" status. Note that the "all clear" message is only sent if there were previous faults during this operational period (key on). It is not sent upon startup.

The Emergency Message Identifier

The Emergency message identifier consists of the Standard Function code, EMCY, in the top four bits. The bottom 7-bits of the message identifier contain the device's Node-ID.

Sync-Error Function code				Node ID						
11	10	9	8	7	6	5	4	3	2	1
0001b				0000001b – 1111110b						

Emergency Message Data Bytes

Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
Error Code		Error Register	Fault Record Object Index		Fault Descriptor		

Data Byte 0 & 1 – Error Code

The Error Code is a two byte construction where the MSB indicates the error category and the LSB may contain a device specific code. The controller will report faults in these data bytes as:

0000h = All faults are cleared
FFXXh = The specific Fault Flash Code
62XXh = User Faults (1 – 32). As specified and implemented in the user/OEM VCL and documentation.

** See Programmer: Application Setup » CAN Interface » Emergency Message Rate CAN_Open_Emergency_Inhibit_Time, 0x1015 0x00. The default is 16 ms.*

*** Not referring to the error log/fault history, but only faults that are currently active on the device.*

The Flash Code listed in the first column of Table 24 (Troubleshooting Chart index) includes the hexadecimal representation of the code. For example:

(1) Flash code 1-2 for the CONTROLLER OVERCURRENT fault equates to 0x12, and is reported in Data Bytes 0 & 1 using the Little-Endian format as FF12h (0xFF12). This is represented on the controller's LEDs with the first digit ("1") the red LED and the second digit ("2") the yellow LED. The controller will "flash" the red LED 1-time followed by yellow LED 2-times (LEDs **x1** **x2**) in the decimal equivalent.

(2) Flash code 9-A for the INTERLOCK BRAKING SUPERVISION fault equates to 0x9A, and will be reported in Data Bytes 0 & 1 as FF9Ah (0xFF9A). This is represented on the controller's LEDs with the first digit ("9") the red LED and the second digit ("A") the yellow LED. The controller will "flash" the red LED 9-times followed by yellow LED 10-times (LEDs **x9** **x10**) in the decimal equivalent (9h = 9d, Ah = 10d).

Data Byte 2 – Error Register

This byte mirrors the state of the LSB of the predefined Error Register object 1001h. This byte is 00h if there are no faults and 01h if there are any faults in the device.

Data Bytes 3 & 4 – Fault Record Object Index

The Fault Record Object is the fault's CAN Object Index, thus distinguishing faults sharing the same LED flash codes. For the controller, these objects are stored in indexes 2100h through 27FFh. When no error is present (fault cleared) these bytes will be all 0000h.

Data Byte 5 through 7 – Descriptor

Data Byte 5 indicates the fault's "*type*" as indicated in the Fault Code Troubleshooting Chart, Table 24.

Several examples of the CANbus error messages illustrate the CAN Emergency messages. Example (1) is a single fault type, whereas example (2a-b) has two error types, which help to define the root cause of the fault. Example (3a-b) shares the same flash code, where the Fault Record Object provides the actual fault detected.

1. CANbus Emergency Message: *Motor_Temp_Sensor_Active* | `0x2150 0x01`
   Motor Temp Sensor fault. Flash code 2-9 (LEDs: **x2** **x9**)

Identifier	DLC	Data
0A6	8	***00 00*** 00 ***50 21*** *01* 00 00

In this example, Device ID = 26h (38d). Note the "little-endian" byte-ordering for the ***error code*** (flash code) and ***fault record object index***. Data Byte 5 further defines the fault as per its type. In this case, there is only a single type, resulting in 01. The 11-bit Identifier assembles as: A6h = 00 01 | 0 1 0 0 1 1 0, where the top 4 bits are the Sync-Error Function code, and the bottom 7 bits are the Node ID, as explained above.

If this fault were to clear, while the controller is operational, the CANbus message will clear:

Identifier	DLC	Data
0A6	8	***00 00*** 00 ***00 00*** *01* 00 00

2a. CANbus Emergency Message: *Main_Contactor_Did_Not_Close_Active* | `0x2221 0x01`
Main Contactor Did Not Close fault. Flash code 3-9 (LEDs: **x3** **x9**)

Identifier	DLC	Data
0A6	8	***39 FF*** 01 ***21 22*** *01* 00 00

In this example, Device ID = 26h (38d). Note the "little-endian" byte-ordering for the ***error code*** (flash code) and ***fault record object*** index. The Data Byte 5, *01*, indicates the first fault type for the Main Contactor Did Not Close fault. In this case, the contactor was "open" during initial closure operation (e.g., either the Driver 3 (pin 5) or the KSI Coil Return (pin 2) were disconnected, or the contactor is defective). The 11-bit identifier assembles as: A6h = 00 01 | 0 1 0 0 1 1 0, where the top 4 bits are the Sync-Error Function code, and the bottom 7 bits are the Node ID, as explained above.

2b. CANbus Emergency Message: *Main_Contactor_Did_Not_Close_Active* | `0x2221 0x01`
Main Contactor Did Not Close fault. Flash code 3-9 (LEDs: **x3** **x9**)

Identifier	DLC	Data
0A6	8	***39 FF*** 01 ***21 22*** *02* 00 00

In this example, Device ID = 26h (38d). Note the "little-endian" byte-ordering for the ***error code*** (flash code) and ***fault record object*** index. The Data Byte 5, *02*, indicates the second fault type for the Main Contactor Did Not Close fault. In this case, the contactor "opened" during operation (e.g., either the Driver 3 (pin 5) or the KSI Coil Return (pin 2) were disconnected while operational). The 11-bit identifier assembles as: A6h = 00 01 | 0 1 0 0 1 1 0, where the top 4 bits are the Sync-Error Function code, and the bottom 7 bits are the Node ID, as explained above.

3a. CANbus Emergency Message: *Severe_B_Plus_Undervoltage_Active* | `0x2120 0x01`
Severe B+ Undervoltage. Flash code 1-7 (LEDs: **x1** **x7**)

Identifier	DLC	Data
0A6	8	***17*** *FF* 01 ***20 21*** *01* 00 00

3b. CANbus Emergency Message: *Severe_KSI_Undervoltage_Active* | `0x2122 0x01`
Severe KSI Undervoltage. Flash code 1-7 (LEDs: **x1** **x7**)

Identifier	DLC	Data
0A6	8	***17*** *FF* 01 ***22 21*** *01* 00 00

In examples 3a and 3b, both faults share the same LED flash code. They are distinguished in Data Bytes 3 & 4 – Fault Record Object Index.

The OEM-defined user faults are also accessible in a CANbus emergency message. Error category = 0x62XX (for *User Fault 1-32* are OEM/Factory defined and implemented in VCL). For example:

The User_1_fault is OEM definable and is implemented the same way as the above OS faults.

CANbus Emergency Message: *User_1_Fault_Active* `0x2710 0x01 0x00`
Flash code 5-1 (LEDs **x5** **x1**)

Identifier	DLC	Data
0A6	8	***51 62*** 01 ***10 27*** *01* 00 00

TROUBLESHOOTING CHART INDEX

Table 22 Troubleshooting Chart Index

Flash Code		Fault Name	CAN Index	Page
1-2	0x12	Controller Overcurrent	0x2510	205
1-3	0x13	Current Sensor	0x2832	
1-4	0x14	Precharge Failed	0x2223	
1-5	0x15	Controller Severe Undertemperature	0x2141	206
1-6	0x16	Controller Severe Overtemperature	0x2142	
1-7	0x17	Severe B+ Undervoltage	0x2120	
1-7	0x17	Severe KSI Undervoltage	0x2122	207
1-8	0x18	Severe B+ Overvoltage	0x2130	
1-8	0x18	Severe KSI Overvoltage	0x2132	
1-9	0x19	Speed Limit Supervision	0x2133	208
1-10	0x1A	Motor Not Stopped	0x2134	
1-11	0x1B	Critical OS General	0x2109	
1-12	0x1C	OS General 2	0x210A	209
1-14	0x1E	Motor Short	0x210E	
1-13	0x1D	Reset Rejected	0x2110	
2-2	0x22	Controller Overtemperature Cutback	0x2140	
2-3	0x23	Undervoltage Cutback	0x2121	210
2-4	0x24	Overvoltage Cutback	0x2131	
2-5	0x25	Ext 5V Supply Failure	0x2531	
2-6	0x26	Ext 12V Supply Failure	0x2532	
2-8	0x28	Motor Temp Hot Cutback	0x2151	211
2-9	0x29	Motor Temp Sensor	0x2150	
3-1	0x31	Main Driver	0x2222	
3-2	0x32	EM Brake Driver	0x2320	212
3-3	0x33	Pump Driver	0x2420	
3-4	0x34	Load Hold Driver	0x2430	213
3-5	0x35	Lower Driver	0x2440	214
3-6	0x36	IM Motor Feedback	0x2230	
3-6	0x36	Sin Cos Motor Feedback	0x2232	
3-7	0x37	Motor Open	0x2240	215
3-8	0x38	Main Contactor Welded	0x2220	

Table 22 Troubleshooting Chart Index, cont'd

Flash Code		Fault Name	CAN Index	Page
3-9	0x39	Main Contactor Did Not Close	0x2221	215
3-10	0x3A	Motor Setup Needed	0x2103	216
4-2	0x42	Throttle Input	0x2210	
4-4	0x44	Brake Input	0x2310	
4-6	0x46	NV Memory Failure	0x2830	217
4-7	0x47	HPD Sequencing	0x2211	
4-7	0x47	Emer Rev HPD	0x2331	
4-9	0x49	Parameter Change	0x2813	
4-10	0x4A	EMR Switch Redundancy	0x2817	218
5-1	0x51	User 1 Fault thru User 32 Fault	0x2710	
5-2	0x52	User 2 Fault	0x2711	
5-3	0x53	User 3 Fault	0x2712	
5-4	0x54	User 4 Fault	0x2713	
5-5	0x55	User 5 Fault	0x2720	
5-6	0x56	User 6 Fault	0x2721	
5-7	0x57	User 7 Fault	0x2722	
5-8	0x58	User 8 Fault	0x2723	
5-9	0x59	User 9 Fault	0x2730	
6-1	0x61	User 10 Fault	0x2731	
6-2	0x62	User 11 Fault	0x2732	
6-3	0x63	User 12 Fault	0x2733	
6-4	0x64	User 13 Fault	0x2740	
6-5	0x65	User 14 Fault	0x2741	
6-6	0x66	User 15 Fault	0x2742	
6-7	0x67	User 16 Fault	0x2743	
5-10	0x5A	User 17 Fault	0x2750	
5-11	0x5B	User 18 Fault	0x2751	
5-12	0x5C	User 19 Fault	0x2752	219
5-13	0x5D	User 20 Fault	0x2753	
5-14	0x5E	User 21 Fault	0x2760	
5-15	0x5F	User 22 Fault	0x2761	
6-10	0x6A	User 23 Fault	0x2762	

Table 22 Troubleshooting Chart Index, cont'd

Flash Code		Fault Name	CAN Index	Page
6-11	0x6B	User 24 Fault	0x2763	219
6-12	0x6C	User 25 Fault	0x2770	
6-13	0x6D	User 26 Fault	0x2771	
6-14	0x6E	User 27 Fault	0x2772	
6-15	0x6F	User 28 Fault	0x2773	
7-10	0x7A	User 29 Fault	0x2780	
7-11	0x7B	User 30 Fault	0x2781	
7-12	0x7C	User 31 Fault	0x2782	
7-13	0x7D	User 32 Fault	0x2783	
6-8	0x68	VCL Run Time Error	0x2820	
7-1	0x71	OS General	0x2831	220
7-2	0x72	PDO Timeout	0x2541	
7-3	0x73	Stall Detected	0x2231	
7-7	0x77	Supervision	0x2840	221
7-9	0x79	Supervision Input Check	0x2841	
8-2	0x82	PDO Mapping Error	0x2542	
8-3	0x83	Internal Hardware	0x2835	222
8-4	0x84	Motor Braking Impaired	0x211A	
8-7	0x87	Motor Characterization	0x2850	
8-8	0x88	Encoder Pulse Error	0x2234	224
8-9	0x89	Parameter Out of Range	0x2811	
9-1	0x91	Bad Firmware	0x2815	
9-2	0x92	EM Brake Failed to Set	0x2321	225
9-3	0x93	Encoder LOS	0x2233	
9-4	0x94	Emer Rev Timeout	0x2330	
9-6	0x96	Pump BDI	0x2450	
9-9	0x99	Parameter Mismatch	0x2812	226
9-10	0x9A	Interlock Braking Supervision	0x2332	227
9-11	0x9B	EMR Supervision	0x2333	
10-1	0xA1	Driver 1 Fault	0x2160	
10-2	0xA2	Driver 2 Fault	0x2161	228
10-3	0xA3	Driver 3 Fault	0x2162	
10-4	0xA4	Driver 4 Fault	0x2163	229

Table 22 Troubleshooting Chart Index, cont'd

Flash Code		Fault Name	CAN Index	Page
10-5	0xA5	Driver 5 Fault	0x2164	230
10-6	0xA6	Driver 6 Fault	0x2165	231
10-7	0xA7	Driver 7 Fault	0x2166	232
10-8	0xA8	Driver Assignment	0x2632	
10-9	0xA9	Coil Supply	0x2169	233
11-1	0xB1	Analog 1 Out Of Range	0x2620	
11-2	0xB2	Analog 2 Out Of Range	0x2621	
11-3	0xB3	Analog 3 Out Of Range	0x2622	
11-4	0xB4	Analog 4 Out Of Range	0x2623	
11-5	0xB5	Analog 5 Out Of Range	0x2624	
11-6	0xB6	Analog 6 Out Of Range	0x2625	
11-7	0xB7	Analog 7 Out Of Range	0x2626	
11-8	0xB8	Analog 8 Out Of Range	0x2627	
11-9	0xB9	Analog 9 Out of Range	0x2628	
11-11	0xBB	Analog 14 Out Of Range	0x262A	234
11-13	0xBD	Analog 18 Out of range	0x262B	
11-14	0xBE	Analog 19 Out of range	0x262C	
11-12	0xBC	Analog Assignment	0x2631	
12-1	0xC1	Branding Error	0x2860	
12-2	0xC2	BMS Cutback	0x2861	
12-5	0xC5	PWM Input 10 Out of Range	0x2629	
12-7	0xC7	Analog 31 Out of Range	0x2106	
12-8	0xC8	Invalid CAN Port	0x2107	
12-9	0xC9	VCL Watchdog	0x2108	
12-11	0xCB	Primary State Error	0x2113	235
12-12	0xCC	PWM Input 29 Out of Range	0x210D	
12-13	0xCD	PWM Input 28 Out of Range	0x210C	236
13-1	0xD1	Lift Input	0x2104	
13-2	0xD2	Phase PWM Mismatch	0x2101	
13-3	0xD3	Hardware Compatibility	0x2870	237
13-4	0xD4	Lower Input	0x2105	
13-6	0xD6	Hazardous Movement	0x211C	
13-13	0xDD	IMU Failure	0x2114	238

FAULT ACTIONS

The fault actions (effect of fault) in Table 23 use the action code bits individually or when combined as listed in the fault action column in Table 24. The variable *System_Actio*n (0x4E00), which is available in the CIT Programmer *List View* ( ), or TACT, returns the decimal number corresponding to the active fault action bit(s). The example illustrates how to determine and plan for faults in both single and dual motor applications.

Table 23 Fault Actions

Fault Action[1]	Action Code		Description
	Hex	Bit(s)	
NO_ACTION	0x00000000		
ShutdownMotor	0x00000001	1	Disables the motor.
ShutdownMainContactor	0x00000002	2	Shut down the Main contactor. Opens the main by de-energizing the main contactor driver, only if Main Enable = On.
ShutdownEMBrake	0x00000004	3	Shut down the EM Brake. Sets the EM Brake by de-energizing the driver, only if the parameter **Set EM Brake On Fault** = On.
ShutdownThrottle	0x00000008	4	Set the *Throttle_Command* = 0%.
ShutdownInterlock	0x00000010	5	Set the *Interlock_State* = Off.
ShutdownDriver1	0x00000020	6	Shut Down Driver1 (e.g., turn off the driver's PWM).
ShutdownDriver2	0x00000040	7	Shut Down Driver2 (e.g., turn off the driver's PWM).
ShutdownDriver3	0x00000080	8	Shut Down Driver3 (e.g., turn off the driver's PWM).
ShutdownDriver4	0x00000100	9	Shut Down Driver4 (e.g., turn off the driver's PWM).
ShutdownPD	0x00000200	10	Shut down Proportional Driver (*legacy name*).
FullBrake	0x00000400	11	Set the *Brake_Command* = 100%.
ShutdownPump	0x00000800	12	Disables the pump.
TrimDisable	0x00001000	13	Disable Dual Drive trim calculation.
SevereDual	0x00002000	14	For a Dual Drive system, one controller has a severe fault but the main contactor must stay closed so the other controller can continue to operate.
ShutdownSteer	0x00004000	15	Steer angle = 0° (Dual Drive applicable).
LOSDual	0x00008000	16	For a Dual Drive system, it uses the max speed set by the *Dual_LOS_Max_Speed* parameter.
ShutdownDriver5	0x00010000	17	Shut Down Driver 5 (e.g., turn off the driver's PWM).
ShutdownDriver6	0x00020000	18	Shut Down Driver 6 (as applicable to the controller; e.g., turn off the driver's PWM).
ShutdownDriver7	0x00040000	19	Shut Down Driver 7 (as applicable to the controller; e.g., turn off the driver's PWM).
ShutdownCoilSupply	0x00080000	20	Shuts down (turns off) the Coil Supply (return) voltage. (Note, based upon the controller model, this will also shut down the low-side drivers).
ShutdownVehicle	0x01000C0F	25, 12, 11, 4, 3, 2, 1	Evokes all Fault Actions as per the indicated bits.

Table 23 Fault Actions, cont'd

Fault Action[1]	Action Code		Description
	Hex	Bit(s)	
ShutdownDualSteer	0x00100000	21	Sets the Steer Angle to zero.
RequestTractionStop	0x00200000	22	Regen braking.
ShutdownLower	0x00400000	23	When operating the single motor as a pump; see the Hydraulic » *Lower Settings* menu.
ShutdownLift	0x00800000	24	When operating the single motor as a pump. See the Hydraulic » *Lift Settings* menu.
ShutdownSteeringSafetyOutput	0x01000000	25	Not applicable to traction motor controllers.
ShutdownAll	0x7FFFFFFF	sets all bits (1-31) = 2	Applies all the fault actions.

Fault Action Examples, where the *System_Action* variable applies to the specific motor.[2]

Stall_Detected, Flash Code 0x73. [7-3]

Fault Action — Single Motor = ShutdownEMBrake + ShutdownThrottle[3] + ShutdownMotor.

System_Action — *Single Motor = 0x4h + 0x8h + 0x1h = 0xDh = 13d = bits: 0000 1101.*

Fault Action — Dual Drive (**this motor**) = SevereDual + ShutdownThrottle[3] + ShutdownMotor.

Dual Drive (**other motor**) = SevereDual + LOSDual + TrimDisable.

System_Action — Dual Drive (**this motor**) = 8009h = 32,777d
= bits: 1000 0000 0000 1001.

Dual Drive (**other motor**) = B000h = 45,056d
= bits: 1011 0000 0000 0000.

[1] *Not all fault actions are applicable to all controllers.*

[2] *Reference the dual drive supplement manual for further dual drive details and its application.*

[3] *Note that for the Stall_Detected, the ShutdownThrottle was removed since FOS 4.4.*

TROUBLESHOOTING CHART

Table 24 Fault Code Troubleshooting Chart

The text of Table 24 is available in the System Information (Sys Info) file.

Note that the indicated CAN Index is useful to search within Sys Info and the List View in CIT Programmer.
- The fault's "active" status (the icon in CIT Programmer) is sub-index of 01.
- The fault's "type" has the sub-index of 06.

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
1-2 0x12	**Controller Overcurrent** *Controller_Overcurrent* 0x2510 Fault Type(s): 1 = Controller Overcurrent Phase U 2 = Controller Overcurrent Phase W 3 = Controller Overcurrent Phase V 4 = Irms > 120 % Current Limit	1. External short of phase U, V, or W motor connections. 2. Speed encoder noise problems. 3. Motor parameters are mistuned. 4. Controller defective.	*Set:* Phase current exceeded the current measurement limit. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownVehicle* Other Motor: *ShutdownVehicle*
1-3 0x13	**Current Sensor** *Current_Sensor* 0x2832 Fault Type(s): 1	1. Leakage to vehicle frame from phase U, V, or W (short in motor stator). 2. Controller defective.	*Set:* Controller current sensors have invalid offset reading. *Clear: Reset Controller.*	ShutdownVehicle (except pump): *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* Dual Drive This Motor: Same Other Motor: Same
1-4 0x14	**Precharge Failed** *Precharge_Failed* 0x2223 Fault Type(s): 1 = Abort 2 = Energy Limit Exceeded 3 = Time Limit Exceeded	1. An external load on the capacitor bank (B+ connection terminal) that prevents the capacitor bank from charging. 2. See Programmer » System Monitor menu » Controller » Capacitor Voltage.	*Set:* The precharge failed to charge the capacitor bank. *Clear:* Cycle Interlock or *Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownVehicle* Other Motor: *ShutdownVehicle*

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
1-5 0x15	**Controller Severe Undertemp** *Controller_Severe_Undertemp* 0x2141 Fault Type(s): 1	1. Controller is operating in an extreme environment. 2. See Programmer » System Monitor menu » Controller » Controller Temperature.	*Set:* Heatsink temperature below –40°C (–40°F). *Clear:* Bring the heatsink temperature above –40°C (–40°F) and then *Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownMotor* *SevereDual* Other Motor: *SevereDual* *LOSDual* *TrimDisable*
1-6 0x16	**Controller Severe Overtemp** *Controller_Severe_Overtemp* 0x2142 Fault Type(s): 1	1. Controller is operating in an extreme environment. 2. Excessive load on vehicle. 3. Improper mounting of controller. 4. See Programmer » System Monitor menu » Controller » Controller Temperature.	*Set:* Heatsink temperature above +95°C. *Clear:* Bring heatsink temperature below +95°C, and then *Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownVehicle* Other Motor: *ShutdownVehicle*
1-7 0x17	**Severe B+ Undervoltage** *Severe_B_Plus_Undervoltage* 0x2120 Fault Type(s): 1 = Undervoltage cutback (0x343B = 0%) or capacitor voltage below safe limits for 64ms. 2 = Commanded voltage could not be achieved due to low capacitor voltage, see also phase PWM mismatch.	1. Non-controller system drain on battery. 2. Battery resistance too high. 3. Battery disconnected while driving. 4. Blown B+ fuse or main contactor did not close. 5. Battery parameters are misadjusted. 6. See Programmer » Monitor menu » Controller » Capacitor Voltage. 7. See the Voltage Limits in Chapter 3.	*Set:* When Main is closed and the FET Bridge is enabled, either the undervoltage drive current cut back = 0% for 64 ms or the Brownout Voltage is reached. *Clear:* Undervoltage drive current cut back > 0% for 100 ms and capacitor voltage > brownout voltage.	No drive torque. Fault Action: *ShutdownPump* *ShutdownMotor* Dual Drive This Motor: *TrimDisable* *ShutdownPump* *ShutdownMotor* Other Motor: *TrimDisable*

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
1-7 0x17	**Severe KSI Undervoltage** *Severe_KSI_Undervoltage* 0x2122 Fault Type(s): 1 = Brownout is disabled due to invalid product data configuration. 2 = Keyswitch_Voltage below brownout threshold at bootup. 3 = Keyswitch_Voltage below low brownout threshold for 5 ms. 4 = Keyswitch_Voltage below high brownout threshold for 64 ms.	1. Non-controller system drain on battery/keyswitch circuit wiring. 2. Resistance in low power (KSI) circuit is too high. 3. KSI disconnected while driving. 4. Blown fuse. 5. See Programmer » System Monitor menu » Battery » Keyswitch Voltage. 6. See the Voltage Limits in Chapter 3.	*Set:* The KSI voltage dropped into the Brownout Voltage regions. *Clear:* Bring KSI voltage above Brownout Voltage.	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors
1-8 0x18	**Severe B+ Overvoltage** *Severe_B_Plus_Overvoltage* 0x2130 Fault Type(s): 1	1. Battery parameters are misadjusted. 2. Battery resistance too high for given regen current. 3. Battery disconnected while regen braking. 4. See Programmer » System Monitor menu » Controller » Capacitor Voltage. 5. See the Voltage Limits in Chapter 3.	*Set:* Capacitor bank voltage exceeded the Severe Overvoltage limit with the FET bridge enabled. *Clear:* Bring capacitor voltage below Severe Overvoltage limit, and then *Reset Controller*.	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownMotor* *SevereDual* Other Motor: *SevereDual* *LOSDual* *TrimDisable*
1-8 0x18	**Severe KSI Overvoltage** *Severe_KSI_Overvoltage* 0x2132 Fault Type(s): 1	1. Battery-voltage applied to KSI (pin 1) exceeds the Severe Overvoltage limit. 2. See Programmer » Monitor menu » Battery » Keyswitch Voltage. 3. See the Voltage Limits in Chapter 3.	*Set:* KSI voltage exceeded the Severe Overvoltage limit. *Clear:* Bring KSI voltage below the Severe Overvoltage limit, and then *Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownMotor* *SevereDual* Other Motor: *SevereDual* *LOSDual* *TrimDisable*

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
1-9 0x19	**Speed Limit Supervision** *Speed_Limit_Supervision* 0x2133 Fault Type(s): 1	1. Motor speed detected that exceeds the limit set by the Max Speed Supervision parameter. 2. Misadjusted Max Speed Supervision parameters. 3. See: Programmer » Application Setup » Max Speed Supervision menu.	*Set:* Motor rpm has exceeded the Max Speed Limit setting for the Max Speed Time Limit setting's duration. *Clear: Reset Controller.*	*ShutdownInterlock* *ShutdownEMBrake* Dual Drive This Motor: *ShutdownInterlock* *ShutdownEMBrake* Other Motor: *ShutdownInterlock* *ShutdownEMBrake*
1-10 0x1A	**Motor Not Stopped** *Motor_Not_Stopped* 0x2134 Fault Type(s): 1 = The motor moved more revolutions than the parameter, *Motor_Not_Stopped_Position_Error* setting. 2 = The motor moved faster than the parameter, *Motor_Not_Stopped_Speed_Error* (RPM) for 160 ms. 3 = The three-phase drive has applied an electrical frequency greater than the *Motor_Not_Stopped_Max_Frequency* parameter, and applied an RMS current greater than the *Motor_Not_Stopped_Max_Current* parameter for 64 ms.	1. Misadjusted Motor Not Stopped parameters. 2. See: Programmer » Application Setup » Motor Not Stopped menu. 3. Internal Controller fault or conflict allowing the motor to rotate when in the stopped state.	*Set:* Motor Not Stopped is a safety function implemented in the Primary microprocessor on a category 2 architecture per ISO 13849. The purpose of this function is to detect hazardous movement when the AC motor is stopped and expected to stay stopped (i.e., no throttle command). There are three main checks done when the motor is in the stopped state, each of which can be independently enabled and each of which has a unique fault type. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownVehicle* Other Motor: *ShutdownVehicle*
1-11 0x1B	**Critical OS General** *Critical_OS_General* 0x2109 Fault Type(s): (<100) Internal Fault. Contact Curtis support. (>100) An ill-formed or corrupted application package was loaded into controller.	1. (<100) Internal Fault. 2. (>100) CIT version is too old to fully support the FOS version.	*Set:* Program execution within the controller encountered a serious problem and could not recover. *Clear:* (<100) Internal Fault. (>100) Update CIT version, re-package the project, and re-flash the application package.	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
1-12 0x1C	**OS General 2** *OS_General_2* 0x210A Fault Type(s): (<100) Internal Fault. Contact Curtis support. (>100) An ill-formed or corrupted application package was loaded into controller.	1. (<100) Internal Fault. 2. (>100) CIT version is too old to fully support the FOS version.	*Set:* Program execution within the controller encountered a serious problem and could not recover. *Clear:* (<100) Internal Fault. (>100) Update CIT version, re-package the project, and re-flash the application package.	NO ACTION (controller is not operable) <u>Dual Drive</u> Same, both motors
1-13 0x1D	**Reset Rejected** *Reset_Rejected* 0x2110 Fault Type(s): 1	This occurs if a controller is commanded to reset while controlling a PMAC motor that is not stationary. Examples of resets include sending an NMT reset or calling *reset_controller()* in VCL. Note, the controller will NOT reset when safe unless the NMT is resent or *reset_controller()* is called again. If legacy brownout is set to off, the user may see this fault if the controller is turned off and on again at an unsafe time, but in this instance the controller will reset as soon as it is safe to do so. Consult Curtis Support for further assistance using non legacy brownout.	*Set:* A reset was called at a time unsafe for the controller. *Clear:* Cycle KSI.	*ShutdownInterlock* *ShutdownThrottle* <u>Dual Drive</u> Same, both motors
1-14 0x1E	**Motor Short** *Motor_Short* 0x210E Fault Type(s): 1	Check Motor Type and Parameters. See the PMAC considerations. Indicates whether the fault is presently active or not.	*Set:* The motor was shorted to avoid dangerous voltage levels. *Clear: Reset controller.*	NO ACTION (controller is not operable) <u>Dual Drive</u> Same, both motors
2-2 0x22	**Controller Overtemp Cutback** *Controler_Overtemp_Cutback* 0x2140 Fault Type(s): 1 = Controller heatsink high temperature (affecting AC phases) 2 = Controller heatsink high temperature (affecting pump phase) 3 = Capacitor bank high temperature 4 = AC phase FET high temperature 5 = Pump phase FET high temperature 6 = Low Frequency single phase high temperature.	1. Controller is operating in an extreme environment. 2. Excessive load on vehicle. 3. Improper mounting of controller which is preventing controller cooling. 4. Controller is performance-limited at this temperature. 5. See Programmer » System Monitor menu » Controller » Temperature.	*Set:* Controller's heatsink temperature exceeded 85°C. *Clear:* Bring heatsink temperature below 85°C.	Reduced drive torque. Reduced regen-braking torque. <u>Fault Action:</u> None, unless a fault action is programmed in VCL. <u>Dual Drive</u> This Motor: *TrimDisable* Other Motor: *TrimDisable*

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
2-3 0x23	**Undervoltage Cutback** *Undervoltage_Cutback* 0x2121 Fault Type(s): 1	1. Batteries need recharging. Controller is performance limited at this voltage. 2. Battery parameters are misadjusted. 3. Non-controller system-drain on battery. 4. Battery resistance too high. 5. Battery disconnected while driving. 6. Blown B+ fuse or main contactor did not close. 7. See Programmer » System Monitor menu » Controller » *Cutbacks* » *Undervoltage Cutback*. 8. See Programmer » System Monitor menu » Controller » Capacitor Voltage.	*Set:* Capacitor bank voltage dropped below the *Undervoltage Cutback* limit with the FET bridge enabled. *Clear:* Bring the capacitor voltage above the controller's *Undervoltage Cutback* limit.	Reduced drive torque. Reduced regen braking torque. Fault Action: None, unless a fault action is programmed in VCL. Dual Drive This Motor: *TrimDisable* Other Motor: *TrimDisable*
2-4 0x24	**Overvoltage Cutback** *Overvoltage_Cutback* 0x2131 Fault Type(s): 1	1. Normal operation. Fault shows that regen braking currents elevated the battery voltage during regen braking. Controller is performance limited at this voltage. 2. Battery parameters are misadjusted. 3. Battery resistance too high for given regen current. 4. Battery disconnected while regen braking. 5. See Programmer » System Monitor menu » Controller » *Cutbacks* » *Overvoltage Cutback*. 6. See Programmer » System Monitor menu » Controller » Capacitor Voltage.	*Set:* The controller's capacitor bank voltage exceeded the *Overvoltage Cutback* limit with the FET bridge enabled. *Clear:* Bring controller's capacitor voltage below the *Overvoltage Cutback* limit.	Reduced brake torque. Note: This fault is declared only when the controller is running in regen. Fault Action: None, unless a fault action is programmed in VCL. Dual Drive This Motor: *TrimDisable* Other Motor: *TrimDisable*
2-5 0x25	**Ext 5V Supply Failure** *Ext_5V_Supply_Failure* 0x2531 Fault Type(s): 1 = The 5V supply is outside 5V ± 10%. 2 = The current is outside the limits defined by: - Ext_5V_Supply_Min. - Ext_5V_Supply_Max.	1. External load impedance on the +5V supply is too low (i.e., a short circuit). 2. See the *System Monitor » Outputs* menu: *External_5V_Supply,* *Ext_5V_Current.* .	*Set:* Triggered by the Fault Type indicated. *Clear: Reset Controller*, or Reset using the VCL variable *Ext_5V_Output_Enable.*	Disables the 5V Supply. Fault Action: None, unless a fault action is programmed in VCL.
2-6 0x26	**Ext 12V Supply Failure** *Ext_12V_Supply_Failure* 0x2532 Fault Type(s): 1 = The 12V supply is outside 12V ± 15%. 2 = The current is outside the limits defined by: - Ext_12V_Supply_Min. - Ext_12V_Supply_Max.	1. External load impedance on the +12V supply is too low (i.e., a short circuit). 2. See Programmer » System Monitor menu » Outputs: *External_12V_Supply,* *Ext_12V_Current.*	*Set:* Triggered by the Fault Type indicated. *Clear: Reset Controller.* Or Reset using the VCL variable *Ext_12V_Output_Enable.*	Disables the 12V Supply. Fault Action: None, unless a fault action is programmed in VCL.

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
2-8 0x28	**Motor Temp Hot Cutback** *Motor_Temp_Hot_Cutback* 0x2151 Fault Type(s): 1	1. Motor temperature is at or above the programmed Temperature Hot setting—resulting in a reduction of controller drive current. 2. The motor temperature and sensor control parameters are misadjusted. 3. See Programmer » AC Motor Setup » Temperature Sensor.	*Set:* Motor temperature is at or above the Temperature Hot parameter setting. *Clear:* Bring the motor temperature within range.	Reduced Drive Torque. If MotorBrakingThermal CutBack_Enable = On, then Regen Braking Torque is reduced. Fault Action: None, unless a fault action is programmed in VCL. Dual Drive This Motor: *TrimDisable* Other Motor: *TrimDisable*
2-9 0x29	**Motor Temp Sensor** *Motor_Temp_Sensor* 0x2150 Fault Type(s): 1	1. Motor thermistor is not connected properly. 2. Sensor polarity (between Pin 8 and Pin 18) is incorrect. 3. The motor temperature and sensor parameters are misadjusted. 4. See Programmer » System Monitor menu » AC Motor » Temperature.	*Set:* Motor thermistor input (pin 9) is at the voltage rail. *Clear:* Bring the motor thermistor input voltage within range.	Motor temperature cutback disabled. Fault Action: None, unless a fault action is programmed in VCL. Dual Drive This Motor: *LOSDual* Other Motor: *LOSDual*
3-1 0x31	**MAIN DRIVER** *Main_Driver_Fault* 0x2222 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver x Overcurrent parameter. 5. See Programmer » Controller Setup » Outputs » Driver x » Driver x Overcurrent.	*Set:* Main Contactor driver is either open or shorted. This fault can be set only when Main Enable = On. *Clear:* Restore/repair any external wiring or device-coil to their correct state, then *Reset Controller*.	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownVehicle* Other Motor: *ShutdownVehicle*

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
3-2 0x32	**EM Brake Driver Fault** *EM_Brake_Driver_fault* 0x2320 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver x Overcurrent parameter. 5. See Programmer » Controller Setup » Outputs » Driver x » Driver x Overcurrent.	*Set:* Electromagnetic brake driver (pin 4) is either open or shorted. This fault can be set only when EM Brake Type >0. *Clear:* Restore/repair any external wiring or device-coil to their correct state, then *Reset Controller.*	*ShutdownEMBrake* *ShutdownThrottle* *FullBrake* <u>Dual Drive</u> This Motor: Same Other Motor: Same
3-3 0x33	**Pump Driver Fault** *Pump_Driver_fault* 0x2420 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver x Overcurrent parameter. 5. See Programmer » *Controller Setup » Outputs » Driver x » Driver x Overcurrent*.	*Set:* The assigned pump-contactor driver is either open or shorted, or exceeded its overcurrent setting. Note: This fault is typically associated with non-pump controllers operating a DC pump via a Driver, yet can apply to controllers with the pump (e.g., the F2-T/F2-C) is also so configured. *Clear:* Correct the open or short, and then *Reset Controller*.	*Shutdownpump* <u>Dual Drive</u> This Motor: *Shutdownpump* Other Motor: *Shutdownpump*

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
3-4 0x34	**Load Hold Driver Fault** *Load_Hold_Driver_Fault* 0x2430 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver x Overcurrent parameter. 5. See Programmer » *Controller Setup » Outputs » Driver x » Driver x Overcurrent.*	*Set:* The assigned load hold driver is either open or shorted, or exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller.*	NO ACTION Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
3-5 0x35	**Lower Driver Fault** *Lower_Driver_Fault* 0x2440 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver x Overcurrent parameter. 5. See Programmer » Controller Setup » Outputs » Driver x » Driver x Overcurrent. Note: See Driver 1* Fault * *Driver 1 is the PD Driver, therefore the Lower Driver fault cascades to the Driver 1 fault (see flash code 10-1).*	*Set:* The assigned lower driver is either open or shorted, or exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller.*	*ShutdownPD* Dual Drive This Motor: Same Other Motor: Same
3-6 0x36	**IM MOTOR FEEDBACK** *IM_Motor_Feedback* 0x2230 Fault Type(s): 1. Controller saw a fast transition to zero speed. 2. Encoder supply failed. 3. Sine or Cosine input differs from expected. 4. Controller saw sensor failure at speed. 5. Unrealistic motor acceleration seen. 6. Resolver loss of signal. 7. Resolver degradation of signal. 8. Resolver loss of tracking.	1. Motor encoder failure. 2. Bad crimps or faulty wiring. 3. See Programmer » System Monitor menu » AC Motor: Motor RPM. 4. See Programmer » AC Motor Setup » Quadrature Encoder » Encoder Fault Setup.	*Set:* Motor position/speed sensor fault. *Clear: Either Reset Controller, or if parameter LOS Upon Encoder Fault = On and Interlock has been cycled, then the Encoder Fault is cleared and Encoder LOS fault (flash code 9-3) is set, allowing limited motor control.*	*ShutdownEMBrake* *ShutdownMotor* Dual Drive This Motor: *SevereDual* *ShutdownMotor* Other Motor: *SevereDual* *LOSDual* *TrimDisable*

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
3-6 0x36	**PM Motor Feedback** *PM_Motor_Feedback* 0x2232 Fault Type(s): 1. Controller saw a fast transition to zero speed. 2. Encoder supply failed. 3. Sine or Cosine input differs from expected. 4. Controller saw sensor failure at speed. 5. Unrealistic motor acceleration seen. 6. Resolver loss of signal. 7. Resolver degradation of signal. 8. Resolver loss of tracking.	1. Motor encoder failure. 2. Bad crimps or faulty wiring. 3. See Programmer » System Monitor menu » Hardware Inputs: Analog 3 and 4.	*Set:* Motor position/speed Sin/Cos sensor fault. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors
0x36	**Motor Open** *Motor_Open* 0x2240 Fault Type(s): 1	1. Motor phase is open. 2. Bad crimps or faulty wiring.	*Set:* Motor phase U, V, or W detected open. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive This Motor: *ShutdownMotor* *SevereDual* Other Motor: *SevereDual* *LOSDual* *TrimDisable*
3-8 0x38	**Main Contactor Welded** *Main_Contactor_Welded* 0x2220 Fault Type(s): 1	1. Main contactor tips are welded closed. 2. Motor phase U or V is disconnected or open. 3. An alternate voltage path (such as an external circuit to B+) is providing a current to the capacitor bank (B+ connection terminal).	*Set:* Just prior to the main contactor closing, the capacitor bank voltage (B+ connection terminal) was loaded (via the motor) for a short time and the voltage did not discharge, indicating a direct-contact to the battery (i.e., Main tips are welded closed). *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
3-9 0x39	**Main Contactor Did Not Close** *Main_Contactor_Did_Not_Close* 0x2221 Fault Type(s): 1 = Main did not close when commanded. 2 = Main disconnected during operation. 3 = Battery disconnected with main enable off.	Type 1: 1. Main contactor did not close. 2. Main contactor tips are oxidized, burned, or not making good contact. 3. An external load on the capacitor bank (B+ connection terminal) is preventing the capacitor bank from charging. 4. Blown B+ fuse. 5. Main Contactor parameters mistuned; • Main Pull-in Voltage, • Main Holding Voltage. Type 2: 1. Main contactor opened during operation (while commanded closed). 2. Driver wiring to contactor's coil (e.g., pin wiring) removed during operation. 3. Contactor/coil defective. Type 3: Battery not connected to B+ when main enable is off and interlock applied.	*Set:* With the main contactor commanded closed, the capacitor bank voltage (B+ connection terminal) did not charge to B+. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors
3-10 0x3A	**Motor Setup Needed** *Motor_Setup_Needed* 0x2103 Fault Type(s): Hex # Bit1 – 4 Binary 0000 => bits 4 3 2 1	Motor setup is required. Please refer to fault type. Bit1: The current regulator needs to be configured. Bit2: The slip gain test needs to be run. Bit3: The base speed test needs to be run. Bit4: The automated test needs to be run (full motor commissioning).	*Set:* Default for both ACIM and PMAC motors. *Clear:* Run the appropriate motor commissioning sequences.	Fault Action: None. Yet, the motor will not operate until the motor configuration and/or commissioning tests are complete.
4-2 0x42	**Throttle Input** *Throttle_Input* 0x2210 Fault Type(s): * 1 = Outside the Low or High parameter. Throttle voltage exceeded the Analog Low or Analog High parameters for the analog input defined for the throttle input. 2 = Input 1 fault diagnostics may be either out of range if it is configured as a voltage input, or may include potentiometer faults if configured as a 2/3-wire pot. * based upon the associated Analog Input faults.	1. Throttle voltage exceeded the Analog Low or Analog High parameters for the analog input defined for the throttle input. 2. See Programmer » Controller Setup » Inputs » Analog 1 Type. 3. See Programmer » Controller Setup » Inputs » Configure.	*Set:* Throttle voltage exceeded the Analog Low or Analog High parameters for the analog input defined for the throttle input. *Clear:* Bring throttle input voltage within the Min and Max thresholds. *Reset Controller.*	*ShutdownThrottle* Dual Drive This Motor: *ShutdownThrottle* Other Motor: No Action

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
4-4 0x44	**Brake Input** *Brake_Input* 0x2310 Fault Type(s): 1*	*Triggered by the respective fault diagnostic associated with the brake input source (assigned analogX input).	*Set:* See Throttle Input. Note: An Input 1 fault diagnostics may be out of range if it is configured as a voltage input or may include potentiometer faults if configured as a 2/3-wire pot. *Clear:* Bring Brake Input voltage within the Min and Max thresholds. *Reset Controller.*	*FullBrake* Dual Drive Same, both motors. Any additional fault action that is programmed in VCL (see AnalogX).
4-6 0x46	**NV Memory Failure** *NV_Memory_Failure* 0x2830 Fault Type(s): 1 = Invalid checksum. 2 = NV write failed. 3 = NV read failed. 4 = NV write did not complete during power down.	1. Failure to read or write to nonvolatile (NV) memory. 2. Internal controller fault.	*Set:* Controller operating system tried to read or write to EEPROM memory and failed. *Clear:* Download the correct software and matching parameter default settings into the controller and *Reset Controller.*	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors
4-7 0x47	**HPD Sequencing** *Hpd_Sequencing* 0x2211 Fault Type(s): Type 1 through 9 - HPD depends on HPD_SRO_Type Type 10 - Interlock Anti-Tiedown	1. Incorrect sequence in application of Keyswitch, Interlock, Direction, or Throttle. 2. Faulty wiring, crimps, or switches at KSI, Interlock, Direction, or Throttle. 3. Moisture in above-noted digital input switches causing invalid (real) On/Off state. 4. Verify input switch status. See Programmer » System Monitor menu » Hardware Inputs » Switch Status. 5. Verify Throttle. See Programmer » System Monitor menu » Hardware Inputs » Throttle Command.	*Set:* HPD (High Pedal Disable) or SRO (Static Return to Off) sequencing fault caused by incorrect sequence of KSI, interlock, direction, and throttle inputs. *Clear:* Reapply inputs in correct sequence.	*ShutdownThrottle* Dual Drive Same, both motors
4-7 0x47	**EMER Rev HPD** *Emer_Rev_Hpd* 0x2331 Fault Type(s): 1	Emergency Reverse operation has concluded, but the throttle, forward and reverse, and interlock inputs have not been returned to neutral.	*Set:* At the conclusion of Emergency Reverse, the fault was set because various inputs were not returned to neutral. *Clear:* If EMR_Interlock = On, clear the interlock, throttle, and direction inputs. If EMR_Interlock = Off, clear the throttle and direction inputs.	*ShutdownThrottle* *ShutdownEMBrake* Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
4-9 0x49	**Parameter Change** *Parameter_Change* 0x2813 Fault Type(s): Reports the CAN Object ID of parameter.	While the Interlock was On, a safety-based parameter was changed. Parameters with this property are marked with a [PCF] (Parameter Change Fault) in the Parameter menu listings.	*Set:* Adjustment of a parameter setting that requires cycling of KSI. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors
4-10 0x4A	**EMR Switch Redundancy** *Emr_Switch_Redundancy* 0x2817 Fault Type(s): 1	1. Either or both Emergency Reverse input switches are inoperative, resulting in an invalid state. NO / NC / State On / Off / valid Off / On / valid On / On / invalid Off / Off / invalid 2. Ingress of dirt or moisture in switch(es).	*Set:* The Emergency Reverse Switch NO input does not agree with the Emergency Reverse Switch NC input. They are opposites: NO and NC. *Clear:* Correct the two switch states. *Reset Controller.*	*ShutdownInterlock* *ShutdownEMBrake* Dual Drive Same, both motors
5-1 0x51	**USER 1 FAULT** *User_{1, 2 … 32}_Fault* 0x2710 Fault Type(s): OEM Definable.	1. These faults (and fault actions) can be defined by the User/OEM and are implemented in the application-specific VCL software. 2. See User/OEM documentation.	*Set:* See User/OEM documentation. *Clear:* See User/OEM documentation.	See User/OEM documentation.
5-2 0x52	**USER 2 FAULT** 0x2711	See User 1 fault (above)	*Set:* See User/OEM documentation. *Clear:* See User/OEM documentation.	See User/OEM documentation.
5-3 0x53	**USER 3 FAULT** 0x2712			
5-4 0x54	**USER 4 FAULT** 0x2713			
5-5 0x55	**USER 5 FAULT** 0x2720			
5-6 0x56	**USER 6 FAULT** 0x2721			
5-7 0x57	**USER 7 FAULT** 0x2722			
5-8 0x58	**USER 8 FAULT** 0x2723			
5-9 0x59	**USER 9 FAULT** 0x2730			
6-1 0x61	**USER 10 FAULT** 0x2731			
6-2 0x62	**USER 11 FAULT** 0x2732			
6-3 0x63	**USER 12 FAULT** 0x2733			
6-4 0x64	**USER 13 FAULT** 0x2740			

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
6-5 0x65	**USER 14 FAULT** 0X2741	See User 1 fault (above)	*Set:* See User/OEM documentation. *Clear:* See User/OEM documentation.	See User/OEM documentation.
6-6 0x66	**USER 15 FAULT** 0X2742			
6-7 0x67	**USER 16 FAULT** 0X2743			
5-10 0x5A	**USER 17 FAULT** 0X2750			
5-11 0x5B	**USER 18 FAULT** 0X2751			
5-12 0x5C	**USER 19 FAULT** 0x2752			
5-13 0x5D	**USER 20 FAULT** 0x2753			
5-14 0x5E	**USER 21 FAULT** 0x2760			
5-15 0x5F	**USER 22 FAULT** 0x2761			
6-10 0x6A	**USER 23 FAULT** 0x2762			
6-11 0x6B	**USER 24 FAULT** 0x2763			
6-12 0x6C	**USER 25 FAULT** 0x2770			
6-13 0x6D	**USER 26 FAULT** 0x2771			
6-14 0x6E	**USER 27 FAULT** 0x2772			
6-15 0x6F	**USER 28 FAULT** 0x2773			
7-10 0x7A	**USER 29 FAULT** 0x2780			
7-11 0x7B	**USER 30 FAULT** 0x2781			
7-12 0x7C	**USER 31 FAULT** 0x2782			
7-13 0x7D	**USER 32 FAULT** 0x2783			

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
6-8 0x68	**VCL Run Time Error** *VCL_Run_Time_Error* 0x2820 Fault Type(s): 1	1. Runtime errors are defined using the VCL Error Module and VCL Error. See the System Information file: • Curtis Integrated Toolkit™ » VCL Studio » Help » System Information. 2. Using driver control commands in VCL can lead to VCL runtime errors if the VCL command and the driver assignment do not match.	*Set:* VCL Run Time Error detected. *Clear:* Edit the VCL application software to fix this error condition; flash the new compiled software and matching parameter settings; *Reset Controller.*	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors
7-1 0x71	**OS General** *OS_General* 0x2831 Fault Type(s): 1	Physical damage from external sources/events.	*Set:* Internal controller fault detected. *Clear: Reset Controller.*	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors
7-2 0x72	**PDO Timeout** *PDO_Timeout* 0x2541 Fault Type(s): 1	1. The time between CAN PDO messages received exceeded the PDO Timeout Period as defined by the Event Timer parameter. 2. Adjust PDO Settings. See Programmer » Application Setup » CAN Interface » PDO Setups.	*Set:* Time between CAN PDO messages received exceeded the PDO Timeout Period. *Clear:* Receive CAN NMT message, or *Reset Controller.*	*ShutdownInterlock* Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
7-3 0x73	**Stall Detected** *Stall_Detected* 0x2231 Fault Type(s): 1	1. Stalled motor. 2. Motor encoder failure. 3. Bad crimps or faulty wiring. 4. Problems with power supply for the motor encoder. 5. See Programmer »System Monitor menu » AC Motor » Motor RPM.	*Set:* No motor encoder movement detected. *Clear:* Either *Reset Controller,* or if parameter LOS Upon Encoder Fault = On and Interlock has been cycled, then the Stall Detected fault is cleared and the Encoder LOS fault (flash code 9-3) is set, allowing limited motor control.	*ShutdownEMBrake* *ShutdownMotor* Control Mode changed to LOS (Limited Operating Strategy). Dual Drive This Motor: *SevereDual* *ShutdownMotor* Other Motor: *SevereDual* *LOSDual* *TrimDisable*
7-7 0x77	**Supervision** *Supervision* 0x2840 Fault Type(s): 1-4 = Primary Init Error 10 = Primary Task Queue Check 12 = Primary ALU Check 13 = Primary Message Watchdog 100-103 = Supervisor Init Error 104-108 = Supervisor Write Error 109 = Supervisor Task Queue Check 110 = Supervisor ALU Check 111 = Supervisor Message Watchdog 113-118 = Supervisor Firmware Update Failure 119 = Supervisor CRC Check	Internal controller fault.	*Set:* Internal controller failure. *Clear: Reset Controller.*	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors
7-9 0x79	**Supervision Input Check** *Supervision_Input_Check* 0x2841 Fault Type(s): 1	Internal controller fault.	*Set:* Damaged Controller. *Clear: Reset Controller.*	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
8-2 0x82	**PDO Mapping Error** *PDO_Mapping_Error* 0x2542 Fault Type(s): 1	1. The PDO Map has too many data bytes assigned or has objects mapped that are not compatible. 2. Adjust PDO Settings. See Programmer » Application Setup » CAN Interface » PDO Setups.	*Set:* Incorrect PDO map detected. *Clear: Reset Controller.*	PDO message disabled. Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors
8-3 0x83	**Internal Hardware** *Internal_Hardware* 0x2835 Fault Type(s): Curtis hardware code.	Internal controller fault detected.	*Set:* Internal controller fault detected. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors
8-4 0x84	**Motor Braking Impaired** *Motor_Braking_Impaired* 0x211A Fault Type(s): 1	1. Battery overcharged, excessive motor or controller heating, or misadjusted parameters. 2. Motor braking was impaired beyond a safe threshold.	*Set:* OverallCutback 0x32D9 fell below *Motor_Braking_Impaired_Threshold* for *Motor_Braking_Impaired_Time* during regen (braking). *Clear:* Reset interlock.	ShutdownInterlock: Dual Drive This Motor: *ShutdownInterlock* Other Motor: *ShutdownInterlock*
8-7 0x87	**Motor Characterization Error** *Motor_Characterization* 0x2850 Fault Type(s): 1 Failure to determine encoder pulses. Must be set manually. 72 Temp sensor fault. 73 Motor hot. 74 Controller temperature cutback. 76 Undervoltage cutback. 77 Overvoltage cutback. 78 Encoder not reading properly. 79 Current Regulator Tuning out of range. 80 Current Regulator Tuning out of range. 81 Encoder signal seen but step size not auto-detected, it must be set manually. 82 Aborted commissioning. 83 Sensor signal too noisy for characterization. 84 Motor not rotating, Sin/Cos sensor voltages out of spec, or Multiturn Sensor setting incorrect. 85 Sensor signal too noisy for characterization.	Motor characterization failed during characterization process. The fault type indicates the cause. Type 84: During commissioning, if the Type 84 fault occurs, check that the Sin/Cos signal voltages at their maximums and minimums have differences less than 78mV. (i.e., that $\lvert Sinmax\ \Delta\ Cosmax \rvert < 78mV$ and the $\lvert Sinmin\ \Delta\ Cosmin \rvert < 78mV$). If the differences are greater than 78mV (e.g., 100mV), it will trigger the Type 84 fault and abort the motor characterization routine. Types are also listed in Chapter 6, Table 21.	*Set:* Motor characterization failed during the motor characterization process. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
8-7 0x87	86 Sin/Cos sensor missing or sensor voltage out of range. 87 PMAC Motor Type must be set before commissioning. 88 PMAC motor fell to zero speed, check your system for excessive friction or loading, retry with a higher test speed, or consult CCA. — 91 PMAC motor not rotating or motor type incorrect. 92 PMAC Motor not accelerating. Low acceleration. 93 Started motor characterization while motor was spinning. 94-98 PMAC lag compensation out of range. 99 PMAC Motor not accelerating. Low acceleration. 102 PMAC motor temp sensor. 103 PMAC motor temp hot cutback. 104 PMAC controller temp cutback. 106 PMAC undervoltage cutback. 107 PMAC overvoltage cutback. 108 Commissioning stopped by user. 500 The Hall patterns do not match the pattern table during refining process. 501 Hall patterns and angles are not consistent during rebuilding of sectors in reverse direction. 502 The rebuilt angle in reverse direction does not align to the calibrated angle. 503 Hall patterns and angles are not consistent during rebuilding sectors in forward direction. 504 The rebuilt angle in forward direction does not align to the calibrated angle. 504 The controller does not get enough Hall switch pulses. 505 The Hall switch patterns are not consistent. 506 Invalid patterns are detected.			

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
8-8 0x88	**Encoder Pulse Error** *Encoder_Pulse_Error* 0x2234 Fault Type(s): 1	1. Encoder Steps parameter does not match the actual motor encoder. 2. Verify parameter settings: AC Motor Setup » Quadrature Encoder » Encoder Steps.	*Set:* Detected wrong setting of the Encoder Steps parameter. *Clear:* Ensure the Encoder Steps parameter matches the actual encoder. *Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors
8-9 0x89	**Parameter Out of Range** *Parameter_Out_Of_Range* 0x2811 Fault Type(s): Reports the CAN Object ID of parameter.	1. Parameter value detected outside of the limits. 2. Use CIT or the 1313HHP to view the parameter's range and adjust the parameter's value.	*Set:* Parameter detected outside of limits. *Clear:* Bring parameter within its limits.	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors
9-1 0x91	**Bad Firmware** *Bad_Firmware* 0x2815 Fault Type(s): 1	The firmware in the controller is incorrect. 1. The CRC of the application or OS does not match. 2. The application was built with an incompatible OS version.	*Set:* The loaded software is not compatible with the controller hardware. *Clear:* Load the matching software. Verify that the controller model matches the cdev version for the project and the VCL Studio application.	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors
9-2 0x92	**EM Brake Failed To Set** *EM_Brake_Failed_to_Set* 0x2321 Fault Type(s): 1	1. Vehicle movement sensed after the EM Brake has been commanded to set. 2. EM Brake will not hold the motor from rotating.	*Set:* After the EM Brake was commanded to set and time has elapsed to allow the brake to fully engage, vehicle movement has been sensed. *Clear:* 1. Activate the Throttle (EM Brake type 2). 2. Activate the Interlock (EM Brake type 1).	Position Hold is engaged when Interlock = On. Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
9-3 0x93	**Encoder LOS** *Encoder_LOS* 0x2233 Fault Type(s): 1	1. Limited Operating Strategy (LOS) control mode has been activated as a result of either an Encoder Fault (flash code 3-6) or a Stall Detected fault (flash code 7-3). 2. Motor encoder failure. 3. Bad crimps or faulty wiring. 4. Vehicle has stalled.	*Set:* Either the Encoder Fault (flash code 3-6) or Stall Detected (flash code 7-3) was detected. If the parameter LOS Upon Encoder Fault = On and the Interlock has been cycled, then the Encoder LOS (flash code 9-3) control mode is activated, allowing limited motor control (limp home mode). *Clear:* Cycle KSI or, if LOS Mode was activated by the Stall Detected fault, clear by ensuring the encoder senses the proper operation, Motor RPM = 0, and Throttle Command = 0.	LOS Mode Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors
9-4 0x94	**Emer Rev Timeout** *Emer_Rev_Timeout* 0x2330 Fault Type(s): 1	1. Emergency Reverse was activated and concluded because the EMR Timeout timer expired. 2. The emergency reverse input is stuck On.	*Set:* Emergency Reverse was activated and ran until the EMR Timeout timer expired. *Clear:* Turn the emergency reverse input (switch) to Off.	*ShutdownThrottle* *ShutdownEMBrake* Dual Drive Same, both motors
9-6 0x96	**Pump BDI** *Pump_BDI* 0x2450 Fault Type(s): 1	1. The BDI is below the *Lift_BDI_Lockout* setting. 2. BDI parameters are mistuned.	*Set:* Pump deactivated when BDI Percentage below Lift lockout setting. *Clear:* Charge Battery; Cycle KSI.	No Fault Action. Yet, the pump is deactivated. Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

<table>
<tr><th>FLASH CODE</th><th>FAULT NAME
(Curtis Integrated Toolkit™)</th><th>POSSIBLE CAUSES</th><th>SET/CLEAR CONDITIONS</th><th>FAULT ACTIONS</th></tr>
<tr><td>9-9
0x99</td><td>**Parameter Mismatch**
Parameter_Mismatch
0x2812</td><td>1. A parameter with the [PCF] label was changed.
2. Incorrect position feedback type chosen for motor technology in use.
3. Dual drive is enabled in torque mode.
4. Dual drive enabled on only one controller.</td><td>*Set:*
Two or more parameter settings conflict and cannot both be honored.
Clear: Adjust parameters to appropriate values and then *Reset Controller*. Cycle KSI.</td><td rowspan="2">ShutdownAll:

ShutdownMotor
ShutdownMainContactor
ShutdownEMBrake
ShutdownThrottle
ShutdownInterlock
ShutdownDriver1
ShutdownDriver2
ShutdownDriver3
ShutdownDriver4
ShutdownDriver5
ShutdownDriver6
ShutdownDriver7
ShutdownPD
FullBrake
ShutdownPump
ShutdownCoilSupply
ShutdownVehicle
ShutdownLower
ShutdownLift

Dual Drive
Same, both motors</td></tr>
<tr><td colspan="4">Fault Type(s):
0) Dual Drive is setup incorrectly. Speed Mode or Speed Mode Express must be used, EM Brake Type must be 2, *Dual_Drive_Mode_Type* must be 1. Dual DC Motor Technology does not support AC dual drive.
4) EM Brake Control Mode is invalid.
5) Interlock Brake Control Mode is invalid.
6) PMAC Short Circuit Current if *PMAC_Short_Circuit_Current* set above *Base_Current_Limit* in a non-test mode.
7) In a differential steer system, fault actions are misconfigured.
8) Dual motor type must be Differential.
9) PMAC EMF Restriction - In a PMAC application configured for restricted mode operation, the back EMF per speed value is not configured.
10) PMAC Release - A restricted and test mode for PMAC is being used in released software.
11) Torque preload is configured to be saved across key-cycles, but EM Brake preload torque is not set.
12) Invalid Torque Estimate - Configured torque estimation type is incompatible with the selected "Direct Torque" control mode.
13) Command Map Stop - [STEERING] *CommandMapLeftStop* or *CommandMapRightStop* equals zero.
14) [STEERING] Improper sequence of the redundant Command Analog map points.
15) Analog Feedback Maps - [STEERING] Primary or Secondary Analog Feedback maps do not have continuous slope.
16) Sawtooth Command - [STEERING] For Sin/Cos or Sawtooth Command device selection, the primary and secondary types do not match.
17) Sawtooth Feedback - [STEERING] For Sin/Cos or Sawtooth Feedback device selection, the primary and secondary types do not match.
18) Feedback Type - [STEERING] Autocenter is declared as Never and the feedback device type is a relative position device type.
19) Interlock braking supervision must be enabled in PMAC if interlock braking is on.
20) The selected motor type has not been fully characterized and should only be used for development.
21) *R_sys* did not commission properly, contact CCA support.
22) Product Data Code not found in database.
23) Incompatible Motor Technology and Control Mode.
24) Invalid Feedback Type for selected Motor Technology.
25) External supply max current set above hardware limit.
26) Metric off in servo mode.
27) EM brake enabled in servo mode.
28) Travel control supervision not supported in selected control mode.
29 HPD enabled in servo mode.
30) Dual drive enabled in servo mode.
31) Emergency reverse enabled in servo mode.
32) Selected control mode is incompatible with controller model.
33) Selected Feedback Type is incompatible with controller model.
34) Emergency Reverse not supported for Motor Technology.
35) Selected encoder power source is not enabled.
36) Interlock Braking selected in torque mode.
100) Parameter Integrity.</td></tr>
</table>

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
9-10 0x9A	**Interlock Braking Supervision** *Interlock_Braking_Supervision* 0x2332 Fault Type(s): 1 = MotorSpeed did not ramp down fast enough to meet configuration (set by *Interlock_Brake_Supervision_Ramp_Delay* and *Interlock_Brake_Superivsion_Ramp_Rate*). 2 = Vehicle brought to stop, but then EM brake (if configured) failed to set. 3 = Vehicle brought to stop, but then traversed a distance beyond that set by *Interlock_Brake_Supervision_Position_Settling_Limit.*	1. For 1, ramp rate/time set too conservatively (needs to be set for worst case braking (full load) to prevent false trip). 2. The vehicle could have a full battery due to which regen is limited and cannot decelerate fast enough. 3. For 2, check EM Brake< for failures/wear.	*Set:* The interlock brake supervision function, when enabled, monitors the vehicle speed during interlock braking to ensure the vehicle is decelerating and stops within the stopping distance. *Clear: Reset Controller.*	*ShutdownMotor* *ShutdownEMBrake* *ShutdownMainContactor* <u>Dual Drive</u> Same, both motors
9-11 0x9B	**EMR Supervision** *Emr_Supervision* 0x2333 Fault Type(s): 1	1. During an EMR event, the motor speed exceeded the limit set by the Emergency Reverse Supervision parameters. 2. See Programmer » Application Setup » Emergency Reverse » Emergency Reverse Supervision.	*Set:* During an EMR event, the motor speed exceeded the limit set by the Emergency Reverse Supervision parameters. *Clear: Reset Controller.*	*ShutdownMotor* *ShutdownEMBrake* *ShutdownMainContactor* <u>Dual Drive</u> Same, both motors
10-1 0xA1	**Driver 1 Fault** *Driver_1_Fault* 0x2160 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver 1 Overcurrent parameter. 5. See Programmer » Controller Setup » Outputs » Driver 1 » Driver 1 Overcurrent.	*Set:* Driver 1 is either open or shorted, or Driver 1 exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller.*	*ShutdownDriver1* <u>Dual Drive</u> This Motor: *ShutdownDriver1* Other Motor: No Action

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
10-2 0xA2	**Driver 2 Fault** *Driver_2_Fault* 0x2161 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver 2 Overcurrent parameter. 5. See Programmer » Controller Setup » Outputs » Driver 2 » Driver 2 Overcurrent.	*Set:* Driver 2 is either open or shorted, or Driver 2 exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller.*	*ShutdownDriver2* Dual Drive This Motor: *ShutdownDriver2* Other Motor: No Action
10-3 0xA3	**Driver 3 Fault** *Driver_3_Fault* 0x2162 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver 3 Overcurrent parameter. 5. See Programmer » Controller Setup » Outputs » Driver 3 » Driver 3 Overcurrent.	*Set:* Driver 3 is either open or shorted, or Driver 3 exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller.*	*ShutdownDriver3* Dual Drive This Motor: *ShutdownDriver3* Other Motor: No Action

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
10-4 0xA4	**Driver 4 Fault** *Driver_4_Fault* 0x2163 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver 4 Overcurrent parameter. 5. See Programmer » Controller Setup » Outputs » Driver 4 » Driver 4 Overcurrent.	*Set:* Driver 4 is either open or shorted, or Driver 4 exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller.*	*ShutdownDriver4* Dual Drive This Motor: *ShutdownDriver4* Other Motor: No Action

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
10-5 0xA5	**Driver 5 Fault** *Driver_5_Fault* 0x2164 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver 5 Overcurrent parameter. 5. See Programmer » Controller Setup » Outputs » Driver 5 » Driver 5 Overcurrent.	*Set:* Driver 5 is either open or shorted, or Driver 5 exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller.*	*ShutdownDriver5* Dual Drive This Motor: *ShutdownDriver5* Other Motor: No Action

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
10-6 0xA6	**Driver 6 Fault** *Driver_6_Fault* 0x2165 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver 6 Overcurrent parameter. Note: Driver 6 is a digital (On/Off) 1 Amp output.	*Set:* Driver 6 is either open or shorted, or Driver 6 exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller.*	*ShutdownDriver6* Dual Drive This Motor: *ShutdownDriver6* Other Motor: No Action

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
10-7 0xA7	**Driver 7 Fault** *Driver_7_Fault* 0x2166 Fault Type(s): 1 = Driver current exceeded hardware limits. 2 = Driver current exceeded configured over-current limits. 3 = Driver commanded PWM active, using diagnostic pulses. Voltage measured high, should be low. Typically caused by driver failure, or driver pin short to high. 4 = Driver commanded PWM active, using diagnostic pulses. Voltage measured low, should be high. Either open circuit, or driver pin short to ground. 5 = Driver commanded PWM is 0, and voltage measured low (should be high). Either open circuit, or driver pin short to ground. 7 = Driver undercurrent - Monitored current is below undercurrent threshold. Fault types 1-2 are always checked. Fault types 3-5 are only checked if driver checks are enabled.	1. Open or short on driver load. 2. Dirty connector pins at controller or contactor coil. 3. Bad connector crimps or faulty wiring. 4. Driver overcurrent, as set by the Driver 7 Overcurrent parameter. Note: Driver 7 is a digital (On/Off) 1 Amp output.	*Set:* Driver 7 is either open or shorted, or Driver 7 exceeded its overcurrent setting. *Clear:* Correct the open or short, and then *Reset Controller*.	*ShutdownDriver7* <u>Dual Drive</u> This Motor: *ShutdownDriver7* Other Motor: No Action
10-8 0xA8	**Driver Assignment** *Driver_Assignment* 0x2632 Fault Type(s): 5 {X} = Driver number that caused the fault.	1. A Driver Output is used for two or more functions. 2. See Programmer » Controller Setup » IO Assignments » Coil Drivers: Main Contactor Driver, EM Brake Driver, Hydraulic Contactor Driver.	*Set:* Driver assignment conflict (i.e., duplicate items assigned to the same driver). *Clear:* Resolve the conflicted driver assignment, then *Reset Controller.*	<u>Fault Action:</u> None, unless a fault action is programmed in VCL. <u>Dual Drive</u> Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
10-9 0xA9	**Coil Supply** *Coil_Supply_Fault* 0x2169 Fault Type(s): 1 = Short to B- or hardware fault. 2 = One or more drivers that have the drivers checks configured as "Safety Designated" did not shut down when commanded to do so. 3 = Coil Supply startup enable check failed. 4 = Coil Supply startup disable check failed.	1. Short on driver loads. 2. Dirty connector pins at controller or device. 3. Bad connector crimps or faulty wiring. 4. Controller is defective.	*Set:* Short detected after the startup check has passed. A low side driver short is detected and the respective fault action fails to cut-off driver current. Coil supply startup test fails. *Clear: Reset Controller.*	ShutdownAll: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *ShutdownInterlock* *ShutdownDriver1* *ShutdownDriver2* *ShutdownDriver3* *ShutdownDriver4* *ShutdownDriver5* *ShutdownDriver6* *ShutdownDriver7* *ShutdownPD* *FullBrake* *ShutdownPump* *ShutdownCoilSupply* *ShutdownVehicle* *ShutdownLower* *ShutdownLift* Dual Drive Same, both motors
11-1 0xB1	**ANALOG 1 OUT OF RANGE** *Analog_1_Out_Of_Range* 0x2620 Analog_X_Out_of_Range Fault Type(s): 1 = Above High limit. 2 = Below Low limit.	1. Analog 1 input voltage is above the parameter setting of Analog 1 High. 2. Analog 1 input voltage is below the parameter setting of Analog 1 Low. 3. See Programmer » Controller Setup » Inputs » Analog 1. 4. See Programmer » Controller Setup » Inputs » Configure » Analog 1 Low / Analog 1 High.	*Set:* (1) Input voltage (on pin) is above the parameter's set-point threshold. (2) Input voltage (on pin) is below the parameter's set-point threshold. *Clear:* Return the voltage to within the allowed range, then *Reset Controller.*	Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors
11-2 0xB2	**ANALOG 2 OUT OF RANGE** *Analog_2_Out_Of_Range* 0x2621	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-3 0xB3	**ANALOG 3 OUT OF RANGE** *Analog_3_Out_Of_Range* 0x2622	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-4 0xB4	**ANALOG 4 OUT OF RANGE** *Analog_4_Out_Of_Range* 0x2623	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-5 0xB5	**ANALOG 5 OUT OF RANGE** *Analog_5_Out_Of_Range* 0x2624	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-6 0xB6	**ANALOG 6 OUT OF RANGE** *Analog_6_Out_Of_Range* 0x2625	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-7 0xB7	**ANALOG 7 OUT OF RANGE** *Analog_7_Out_Of_Range* 0x2626	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-8 0xB8	**ANALOG 8 OUT OF RANGE** *Analog_8_Out_Of_Range* 0x2627	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-9 0xB9	**ANALOG 9 OUT OF RANGE** *Analog_9_Out_Of_Range* 0x2628	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
11-11 0xBB	**ANALOG 14 OUT OF RANGE** *Analog_14_Out_Of_Range* 0x262A	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-13 0xBD	**Analog 18 Out of Range** *Analog_18_Out_Of_Range* 0x262B	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-14 0xBE	**Analog 19 Out of Range** *Analog_19_Out_Of_Range* 0x262C	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range.
11-12 0xBC	**Analog Assignment** *Analog_Assignment* 0x2631 Fault Type(s): 13 {X = 1-9, 14, 18-19, 31} X = Analog Input number that caused the fault.	1. An Analog input is used for two or more functions. 2. An Analog input is outside the range of analog inputs. 3. See Programmer » Controller Setup » IO Assignments » Controls.	*Set:* An Analog input is used for two or more functions or is outside the range of analog inputs. *Clear:* Resolve assignment conflict, and then *Reset Controller.*	Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors
12-1 0xC1	**Branding Error** *Branding_Error* 0x2860 Fault Type(s): 1	1. Software and hardware branding mismatch. 2. For technical support on this fault, contact the Curtis distributor where you obtained your controller or the Curtis sales-support office in your region.	*Set:* Software/hardware incompatibility. *Clear:* As applicable: Load Branded software, or use Branded controller with the correct device profile and the correct Curtis Software Suite toolkit key.	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors
12-2 0xC2	**BMS Cutback** *BMS_Cutback* 0x2861 Fault Type(s): 1 = Battery Current Cutback. 2 = Low Cell Cutback. 3 = High Cell Cutback.	A cutback based on cell loading has occurred.	*Set:* See Fault Type. *Clear:* Resolve battery or battery cell issue.	Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
12-5 0xC5	**PWM Input 10 Out of Range** *PWM_Input_10_Out_Of_Range* 0x2629 Fault Type(s): 1 = The input is disconnected. 2 = The measured input frequency is below the (PWM_Input_10_Low_ Frequency) – (PWM_ Input_10_Frequency_ Fault_Tolerance). 3 = The measured input frequency is above the (PWM_Input_10_High_ Frequency) + (PWM_ Input_10_Frequency_Fault_ Tolerance). 4 = The measured duty cycle is below set limits, (PWM_Input_10_Low_ Duty_Cycle) – (PWM_ Input_10_Duty_Cycle_ Fault_Tolerance). 5 = The measured duty cycle is above set limits, (PWM_Input_10_High_ Duty_Cycle) + (PWM_ Input_10_Duty_Cycle_ Fault_Tolerance).	1. This fault diagnostic execution cycles every 4msec. The input is considered disconnected if no PWM signal occurs for 16msec or the measurements are not updated every 16msec. 2. Mistuned parameters. 3. Faulty wiring.	*Set:* The input frequency and/or duty-cycle on Input 10 exceeds the configured limits set by PWM_Input_10 _x_ Duty_Cycle and PWM_Input_10_x_ Frequency , where x = {Low, High}. *Clear:* Reset controller.	Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors
12-7 0xC7	**Analog 31 Out of Range** *Analog_31_Out_Of_Range* 0x2106	See Analog 1 Out of Range.	See Analog 1 Out of Range.	See Analog 1 Out of Range. Dual Drive Same, both motors
12-8 0xC8	**Invalid_CAN_Port** *Invalid_CAN_Port* 0x2107 Fault Type(s): The condition is checked at start-up, (i.e. during the dual drive initialization).	1. Mistuned Dual Drive CAN parameters. 2. Conflicting CAN Node IDs for Dual Drive.	*Set:* This fault is triggered when the Dual CAN Port (*DualMotorCanPort*) parameter is set to a CAN port that does not exist on a controller setup for Dual Drive. *Clear: Reset controller.*	NO ACTION Dual Drive Same, both motors
12-9 0xC9	**VCL Watchdog** *VCL_Watchdog* 0x2108	See the associated VCL Functions, • *Set_Watchdog_Timeout( ).* • *Set_Watchdog_Fault_Action( ).* • *Kick_Watchdog( ).* The fault actions can be defined by the User/OEM in the application-specific VCL software.	*Set:* The time interval of the VCL watchdog (*WD#*) exceeded the timeout value. *Clear: Kick_Watchdog( ).* Start and reset the specified watchdog timer.	NO ACTION Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
12-11 0xCB	**Primary State Error** *Primary_State_Error* 0x2113 Fault Type(s): These are internal issues either occurring during startup, parameter initialization, secondary micro update or other runtime issues. 1 = PRIMARY_DEVICE_STARTUP = 0, 2 = PRIMARY_WAIT_KSI_STABLE, 3 = PRIMARY_DEVICE_STARTUP_VALID, 4 = PRIMARY_INITIALIZE_PARAMETERS, 5 = PRIMARY_WAIT_FOR_FIRST_SIGNALS, 6 = PRIMARY_WAIT_FOR_SUPERVISOR, 7 = PRIMARY_RESTORE_PARAMETER_FAIL, 8 = PRIMARY_SUPERVISOR_FIRST_SIGNALS_ERROR, 9 = PRIMARY_SUPERVISOR_STARTUP_ERROR, 10 = PRIMARY_STARTUP_TIMER_FAILURE, 11 = PRIMARY_WAIT_CAN_HANDSHAKING_DONE, 12 = PRIMARY_RUNNING.	If the fault persists, Contact Curtis.	*Set:* Internal error with the controller. Kindly reset controller. *Clear: Reset controller.*	NO_ACTION (controller is not operable) Dual Drive Same, both motors
12-12 0xCC	**PWM Input 29 Out of Range** *PWM_Input_29_Out_of_Range* 0x210D Fault Type(s): 1 = The input is disconnected. 2 = The measured input frequency is below the (PWM_Input_29_Low_Frequency) – (PWM_Input_29_Frequency_Fault_Tolerance). 3 = The measured input frequency is above the (PWM_Input_29_High_Frequency) + (PWM_Input_29_Frequency_Fault_Tolerance). 4 = The measured duty cycle is below set limits, (PWM_Input_29_Low_Duty_Cycle) – (PWM_Input_29_Duty_Cycle_Fault_Tolerance). 5 = The measured duty cycle is above set limits, (PWM_Input_29_High_Duty_Cycle) + (PWM_Input_29_Duty_Cycle_Fault_Tolerance).	1. This fault diagnostic execution cycles every 4msec. The input is considered disconnected if no PWM signal occurs for 16msec or the measurements are not updated every 16msec. 2. Mistuned parameters. 3. Faulty wiring.	*Set:* The input frequency and/or duty-cycle on Input 29 exceeds the configured limits set by *PWM_Input_29_x_Duty_Cycle* and *PWM_Input_29_x_Frequency, where x = {Low, High}.* *Clear: Reset Controller.*	Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
12-13 0xCD	**PWM Input 28 Out of Range** *PWM_Input_28_Out_of_Range* 0x210C Fault Type(s): 1 = The input is disconnected. 2 = The measured input frequency is below the (PWM_Input_28_Low_Frequency) – (PWM_Input_28_Frequency_Fault_Tolerance). 3 = The measured input frequency is above the (PWM_Input_28_High_Frequency) + (PWM_Input_28_Frequency_Fault_Tolerance). 4 = The measured duty cycle is below set limits, (PWM_Input_28_Low_Duty_Cycle) – (PWM_Input_28_Duty_Cycle_Fault_Tolerance). 5 = The measured duty cycle is above set limits, (PWM_Input_28_High_Duty_Cycle) + (PWM_Input_28_Duty_Cycle_Fault_Tolerance).	1. This fault diagnostic execution cycles every 4msec. The input is considered disconnected if no PWM signal occurs for 16msec or the measurements are not updated every 16msec. 2. Mistuned parameters. 3. Faulty wiring.	*Set:* The input frequency and/or duty-cycle on Input 28 exceeds the configured limits set by *PWM_Input_28_x_Duty_Cycle* and *PWM_Input_28_x_Frequency, where x = {Low, High}.* *Clear: Reset Controller.*	Fault Action: None, unless a fault action is programmed in VCL. Dual Drive Same, both motors
13-1 0xD1	**Lift Input Fault** *Lift_Input* 0x2104 Fault Type(s): 1	The associated fault diagnostic with the assigned lift-input source triggers this fault. For example: If the *Lift_Input_Source* is an analog input, then any faults detected by the respective Input fault diagnostics are cascaded and reported within this fault code. Note: An analog input fault diagnostics may be out of range when set as a voltage input or may include potentiometer faults if configured as a 2/3-wire pot.	*Set:* Faults from the respective/ assigned "Lift_Input_Source" are cascaded and reported. *Clear:* Resolve any input assignment conflict, or out of-range faults, then *Reset Controller.*	*ShutdownLift* Dual Drive Same, both motors
13-2 0xD2	**Phase PWM Mismatch** *Phase_PWM_Mismatch* 0x2101 Fault Type(s): 0 = U phase. 1 = V phase. 2 = W phase.	Internal to Controller Motor Phase PWM.	*Set:* The difference between the commanded phase PWM duty cycle and the measured is greater than allowed. *Clear: Reset Controller.*	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
13-3 0xD3	**Hardware Compatibility** *Hardware_Compatibility* 0x2870 Fault Type(s): 1	The OS (device profile, .cdev file) is incompatible with the controller. The loaded software (.cdev) is not compatible with the controller hardware.	*Set:* Incorrect OS (device profile). *Clear:* Load the matching OS (device profile).	ShutdownVehicle: *ShutdownMotor* *ShutdownMainContactor* *ShutdownEMBrake* *ShutdownThrottle* *FullBrake* *ShutdownPump* Dual Drive Same, both motors
13-4 0xD4	**Lower Input Fault** *Lower_Input* 0x2105 Fault Type(s): 1	The associated fault diagnostic with the assigned lower-input source triggers this fault. For example: If the *Lower_Input_Source* is an analog input, then any faults detected by the respective Input fault diagnostics are cascaded and reported within this fault code. Note: An analog input fault diagnostics may be out of range when set as a voltage input or may include potentiometer faults if configured as a 2/3-wire pot.	*Set:* Faults from the respective/ assigned "*Lower_Input_Source*" are cascaded and reported. *Clear:* Resolve any input assignment conflict, or out-of-range faults, *then Reset Controller.*	*ShutdownLower* Dual Drive Same, both motors
13-6 0xD6	**Hazardous Movement** *Hazardous_Movement* 0x211C Fault Type(s): 1 = The motor speed is in the opposite direction of the speed request and the motor fails to accelerate in the correct direction for a programmed time. In the event of a change to neutral, this hazard will be detected if the motor fails to accelerate toward zero speed for a programmed time. 2 = The acceleration is in the opposite direction of the difference between the operator speed request and the motor speed. The speed in the commanded direction is greater than the commanded speed by more than a parameter (*Hazardous_Speed*) for a programmed time (*Hazardous_Throttle_Response_Time*).	(1) Mistuned *Hazardous_Direction_Response_Time* parameter. (2) Mistuned *Hazardous_Accel* parameter. (2) Mistuned *Hazardous_Speed_Error* parameter. (2) Mistuned *Hazardous_Throttle_Response_Time* parameter.	*Set:* This fault detects hazardous movement when the motor is requested to be moving. The first hazard is a motor that is not able to slow down if the throttle goes to zero or the direction switch is not in the direction of travel. The second hazard is a motor that accelerates the wrong way or goes too fast. Note: This fault only occurs when the Control Mode Select is in *Speed_Mode, Speed_Mode_Express*, or *Servo_Mode*. *Clear: Reset Controller.* Setting *Hazardous_Direction_Response_Time* = 0 will disable these checks.	*ShutdownInterlock* Dual Drive Same, both motors

Table 24 Fault Code Troubleshooting Chart, cont'd

FLASH CODE	FAULT NAME (Curtis Integrated Toolkit™)	POSSIBLE CAUSES	SET/CLEAR CONDITIONS	FAULT ACTIONS
13-13 0xDD	**IMU Failure** *IMU_Failure* 0x2114 Fault Type(s): 1 = SPI Communication Failure. 2 = Curtis Factory Self Test Failure. 3 = Run Time Check Failure, bad data received from the IMU. 4 = Gyro Cal out of range, maximum calibration offset exceeded.	Check if configured correctly or the vehicle is moving when calibrating.	*Set:* Internally set as per fault type. *Clear: Cycle KSI.*	NO_ACTION Dual Drive Same, both motors

Note: For faults not in the above tables, consult the System Information file (from the VCL Studio app). Faults will vary based upon the device profile version or controller model.

8 — MAINTENANCE

There are no user serviceable parts in Curtis controllers. Do not open, repair, or otherwise modify the controller. Doing so may damage the controller and will void the warranty.

Keep the controller and connections clean and dry. Periodically check and clear the controller's fault history.

CLEANING

Periodically cleaning the controller exterior will help protect it against corrosion and possible electrical control problems created by dirt, grime, and chemicals that are part of the operating environment of material handling, off-road, and construction equipment.

⚠ CAUTION

When working around any battery-powered system, ensure proper safety precautions. These include, but are not limited to, proper training, wearing eye protection, and avoiding loose clothing and jewelry.

Facility safety equipment, including an eyewash station, should be close to the work area.

Use the following cleaning procedure for routine maintenance. Never use a high-pressure washer to clean the controller.

1. Remove power by disconnecting the battery. Disconnect B+ first (at the battery, not the controller).
2. Discharge the capacitors in the controller by connecting a load (such as a contactor coil) across the controller's B+ and B– terminals.
3. Remove any dirt or corrosion from the power and signal connector areas. Wipe the controller clean using a dry-to-damp rag. Only connect/re-connect the battery cables to a dry controller. Connect B+ last.
4. Make sure the connections are tight. Refer to Chapter 2 for the maximum tightening torque specifications for the battery and motor connections.

FAULT HISTORY

Use the Curtis Integrated Toolkit™ Programmer (application tool) to access the controller's fault history file. The Programmer can clear the fault history. The 1313 HHP can also access and clear the fault history.

- Faults such as contactor faults may be the result of loose wires or cables.
- Faults such as over-temperature and stalls may be due to operator habits or overloading.

After diagnosing and correcting a problem, it is a good idea to clear the fault history file. This allows the controller to accumulate a new file of faults. Checking the fault history file later will indicate whether the problem was fixed.

APPENDIX A

CAN PDO MAP SETUP

This appendix provides guidance on how to set up PDO maps on F-Series controllers. The maps are set up using the *Curtis Integrated Toolkit*™. See *Programmer » Applications Setup » CAN Interface » PDO Setups*. All parameters with OEM Factory access-level (or below) may be accessed by a CANopen SDO or PDO.

Nomenclature note: A hexadecimal value written as 80000226h = 0x80000226.

A CAN Object Index value written as XXXX.XX = 0xXXXX 0xXX, where the text ".XX" and 0xXX are the CAN Object's sub index value.

For the F-Series controllers, RPDO1-4 are definable CAN messages where the controller receives data from another device (e.g., the manager), such as a Throttle Command, which will be processed by VCL. The TPDO1-4 are definable CAN messages where the controller transmits data to another device (e.g., the manager), such as the keyswitch voltage or motor RPM. An example of how to "map" these messages is provided in this appendix. An example of how to map PDOs using SDO (download) messages is also provided.

CANopen PDO Mapping Object description

CANopen CiA 301 specifies that 5 steps must be taken to re-map a PDO. This must occur while the NMT state is pre-operational.

1. Disable the PDO.
2. Set mapping by setting the map length to 0.
3. Modify mapping.
4. Enable mapping by setting the map length to the correct value.
5. Enable PDO.

Structure of RPDO communication parameter object (4 bytes), the RPDO COB IDs.

Index	Sub-Index	Bit 31	Bit 30	Bit 29	Bits 11–28	Bits 7–10	Bits 0–6
		Disabled	Reserved	Frame	Reserved	Standard Message Type	Node-ID
CAN1: RPDO1: 1400h RPDO2: 1401h RPDO3: 1402h RPDO4: 1403h **CAN2:** RPDO1: 1440h RPDO2: 1441h RPDO3: 1442h RPDO4: 1443h	01h	Enabled = 0 Disabled = 1	0	0 (11 bit CAN base frame)	00000h (Used for 29-bit extended frame)	RPDO1: 0200h + Node ID RPDO2: 0300h + Node ID RPDO3: 0400h + Node ID RPDO4: 0500h + Node ID	01h – 7Eh

Structure of TPDO communication parameter object (4 bytes), the TPDO COB IDs.

Index	Sub-Index	Bit 31	Bit 30	Bit 29	Bits 11–28	Bits 7–10	Bits 0–6
		Disabled	Reserved	Frame	Reserved	Standard Message Type	Node-ID
CAN1: TPDO1: 1800h TPDO2: 1801h TPDO3: 1802h TPDO4: 1803h **CAN2:** TPDO1: 1840h TPDO2: 1841h TPDO3: 1842h TPDO4: 1843h	01h	Enabled = 0 Disabled = 1	Must be 1	0 (11 bit CAN base frame)	00000h (Used for 29-bit extended frame)	TPDO1: 0180h + Node ID TPDO2: 0280h + Node ID TPDO3: 0380h + Node ID TPDO4: 0480h + Node ID	01h – 7Eh

Structure of PDO event timer parameter object (2 bytes).

Index	Sub-Index	Bits 0–15
		Event Timer
CAN1: RPDO1: 1400h RPDO2: 1401h RPDO3: 1402h RPDO4: 1403h TPDO1: 1800h TPDO2: 1801h TPDO3: 1802h TPDO4: 1803h **CAN2:** RPDO1: 1440h RPDO2: 1441h RPDO3: 1442h RPDO4: 1443h TPDO1: 1840h TPDO2: 1841h TPDO3: 1842h TPDO4: 1843h	05h	RPDO: PDO Time-out period. 0 if timeout check is disabled. TPDO: PDO Time-out period. 0 if timeout check is disabled.

Structure of PDO length object (1 byte).

Index	Sub-Index	Bits 0–7 Bit Length
CAN1: RPDO1: 1600h RPDO2: 1601h RPDO3: 1602h RPDO4: 1603h TPDO1: 1A00h TPDO2: 1A01h TPDO3: 1A02h TPDO4: 1A03h **CAN2:** RPDO1: 1640h RPDO2: 1641h RPDO3: 1642h RPDO4: 1643h TPDO1: 1A40h TPDO2: 1A41h TPDO3: 1A42h TPDO4: 1A43h	05h	Number of objects in the map (not the number of bits or bytes)

Structure of PDO mapping object (4 bytes).

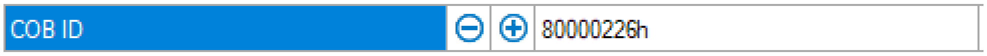
Index	Sub-Index	Bits 31–16	Bits 8–15	Bits 0–7
		PDO Mapping Index	Sub Index	Bit Length
CAN1: RPDO1: 1600h RPDO2: 1601h RPDO3: 1602h RPDO4: 1603h TPDO1: 1A00h TPDO2: 1A01h TPDO3: 1A02h TPDO4: 1A03h **CAN2:** RPDO1: 1640h RPDO2: 1641h RPDO3: 1642h RPDO4: 1643h TPDO1: 1A40h TPDO2: 1A41h TPDO3: 1A42h TPDO4: 1A43h	01h . . . 08h	PDO Mapping Index	Sub-Index	8, 16, or 32. Curtis does not allow mapping of individual bits. 8d = 8h 16d = 10h 24d = 18h 32d = 20h

Example for RPDO Mapping with the CIT Programmer

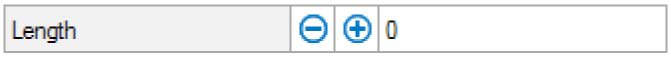
For this example, it will be setting up RPDO1 for a device with Node ID 0x26, and map ***VCL_Throttle*** and ***User1***.

Make sure that the node is pre-operational node. If not, send an NMT message using PCAN-view (or a similar CAN dongle and software).

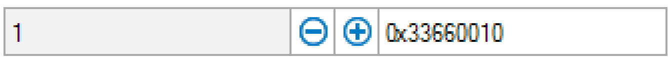
The first step is to disable RPDO1. Do this by setting the most significant bit of *can_rpdo_1_ cob_id* to true (true = 1). Navigate in the CIT Programmer to the following location: ***Application Setup » CAN Interface » PDO Setups***. Set *can_rpdo_1_cob_id* to 80000226h. (Note, the most significant byte is 1000b = 8h, as setting the 31st bit = 1 disables the RPDO. See RPDO COB ID, previous page).



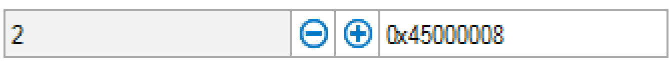
Next, disable the mapping of RPDO1 by setting `can_rpdo_1_length` to 0.



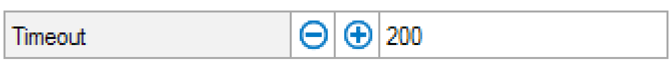
Map the 16-bit *VCL_Throttle* variable with CAN-object 0x3366.00, by setting *can_rpdo_1_ map_1* to a value of 0x33660010. Note that when setting up a PDO that writes to an Operating System variable, the complete word must be written at once (32-bit write to 32-bit variable, 16-bit write to 16-bit variable). Input all values in hex. In this example, the 16-bit VCL_Throttle variable's length is 10h (i.e., the last 2 bytes).



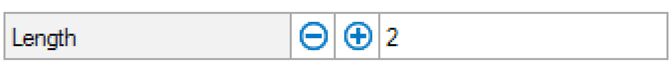
Map 8 bits of the 32-bit User1 variable with CAN-object 0x4500.00, by setting *can_rpdo_1_ map_2* to a value of 0x45000008. In this example, the 8-bits of User1 variable's length is 8h (i.e., the last byte).



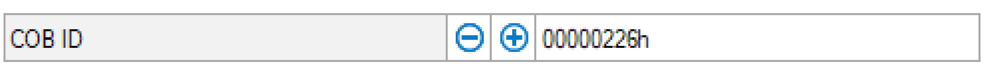
Set *can_rpdo_1_event_timer* to a value in milliseconds, if a timeout check is required on Receive PDO messages.



Set *can_rpdo_1_length* to the number of variables (not bytes) that are mapped. That is 2 in this example.



Now, the PDO can be re-enabled by setting *can_rpdo_1_cob_id* to the value 0x000000226 (i.e., the 31st bit is changed from 1 to 0 for Enabled, see the RPDO COB ID table, above).



The RPDO will become active when changing the NMT State to Operational.

Follow this format for mapping RPDO2 – 4, matching the message type (3rd byte) number, while retaining the same Node ID. For example, for Node ID = 0x26:

RPDO1 ... 80000226 = disabled, 00000226 = enabled

RPDO2 ... 80000326 = disabled, 00000326 = enabled

RPDO3 ... 80000426 = disabled, 00000426 = enabled

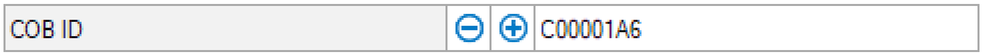
RPDO4 ... 80000526 = disabled, 00000526 = enabled

Example for TPDO Mapping with the Programmer

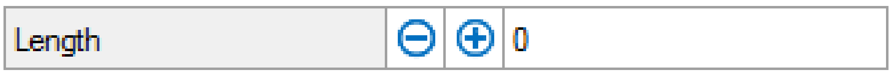
For this example, we will be setting up TPDO1 for the Node ID 0x26, and map ***Keyswitch_Voltage*** and ***User2***.

Make sure that the node is pre-operational. If not, send an NMT message using PCAN-view (or a similar CAN dongle and software).

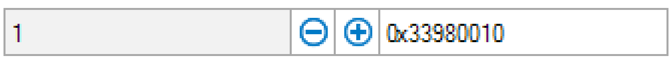
The first step is to disable TPDO1. Do this by setting the most significant bit of *can_ tpdo_1_cob_id* to true (true = 1). Navigate in the *CIT Programmer* to the following location: ***Application Setup*** » ***CAN Interface*** » ***PDO Setups***. Set *can_tpdo_1_cob_id* to 0xC00001A6. (Note, for the TPDO, the MSB is 1100 = Ch. See TPDO COB ID table, above).



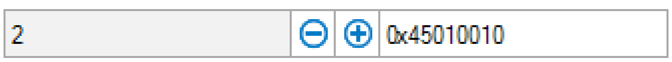
Next, disable the mapping of TPDO1 by setting `can_tpdo_1_length` to 0.



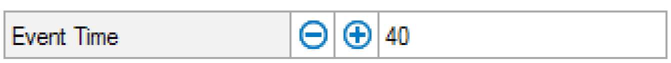
Map the 16-bit Keyswitch_Voltage variable with CAN-object 0x3398.00, by setting *can_tpdo_1_ map_1* to a value of 0x33980010. Note that when setting up a PDO that writes to an Operating System variable, the complete word must be written at once (32-bit write to 32-bit variable, 16- bit write to 16-bit variable). Input all values in hex. In this example, the 16-bit *Keyswitch_Voltage* variable's length is 10h.



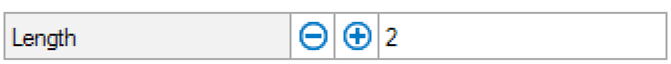
Map 16 bits of the 32-bit User2 variable with CAN-object 0x4501.00, by setting *can_tpdo_1_ map_1* to a value of 0x45010010. In this example, the 16-bits of *User2* variable's length is 10h.



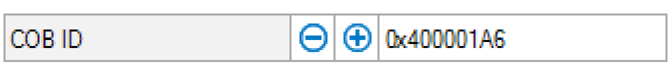
Set `can_tpdo_1_event_timer` to a value in milliseconds, to set the transmit period.



Set `can_tpdo_1_length` to the number of variables (not bytes) that are mapped. That is 2 in this example.



Now, the PDO can be re-enabled by setting *can_tpdo_1_cob_id* to value 0x4000001A6 (i.e., the 31st bit is changed from 1 to 0 for Enabled, setting the MSB to 0100b = 4h. See TPDO COB ID table, above).



The TPDO will become active when changing the NMT State to Operational.

Follow this format for mapping TPDO2 - 4, matching the message type (3rd byte) number, while retaining the same Node ID. For example, for Node ID = 0x26:

TPDO1 ... C00001A6 = disabled, 400001A6 = enabled

TPDO2 ... C00002A6 = disabled, 400002A6 = enabled

TPDO3 ... C00003A6 = disabled, 400003A6 = enabled

TPDO4 ... C00004A6 = disabled, 400004A6 = enabled

EXAMPLE FOR RPDO MAPPING WITH SDO WRITES

For this example, to set up RPDO1, and map *VCL_Throttle (0x3366 0x00)* and *User1 (0x4500 0x00)*:

Send CAN messages in the following table's order to set up RPDO1 mapping on a device with **Node ID 0x28**.

An example using PCAN-View messages is shown, steps 1 – 9 in the PCAN-View Comment column, below.

Note that the data fields are in Little Endian format (e.g., Object Index 0x**3366**.00 is input as **00 66 33**), and this example uses Node ID = 0x28

Header	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7	Description
000	80	28							Send NMT Pre-Operational node 0x28
626	23	00	14	01	28	02	00	80	4-Byte SDO Write: Disable RPDO1
626	2F	00	16	00	00				1-byte SDO Write: Disable Map/Set Length to 0
626	23	00	16	01	10	00	66	33	4-byte SDO Write: Map 1st Object as *VCL_Throttle*
626	23	00	16	02	08	00	00	45	4-Byte SDO Write: Map 2nd Object as *User1*
626	2F	00	16	00	02				1-Byte SDO Write: Enable Map/Set Length to 2
626	2B	00	14	05	C8	00			2-byte SDO Write: Set PDO Timeout to 200 ms
626	23	00	14	01	28	02	00	00	4-Byte SDO Write: Enable PDO
000	01	28							Send NMT Operation node 0x28

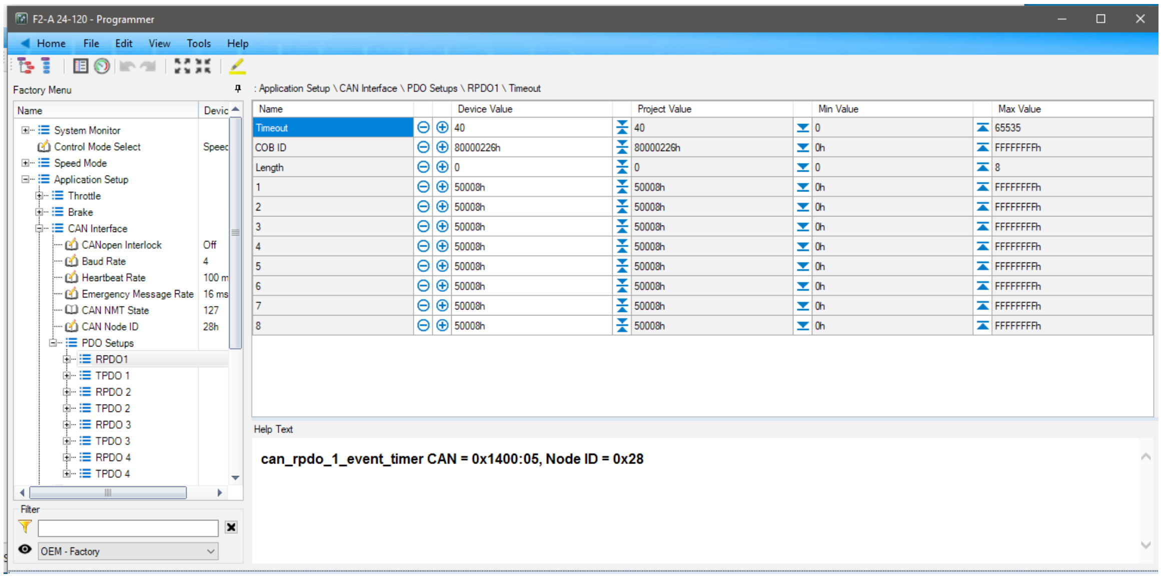
Transmit

Message	DLC	Data	Cycle Time	Count	Trigger	Comment
000h	8	80 28 00 00 00 00 00 00	Wait	1	Manual	1. Send NMT Pre-Operational node 0x28
000h	8	01 28 00 00 00 00 00 00	Wait	1	Manual	9. Send NMT Operation Command to Node 0x28
628h	8	23 00 14 01 28 02 00 80	Wait	1	Manual	2. Send 4 Byte SDO Write: Disable RPDO1
628h	8	2F 00 16 00 00 00 00 00	Wait	1	Manual	3. Send 1 Byte SDO Write: Disable Map/Set Length to 0
628h	8	23 00 16 01 10 00 66 33	Wait	1	Manual	4. Send 4 Byte SDO Write: Map 1st Object VCL_Throttle 0x3366.00
628h	8	23 00 16 02 08 00 00 45	Wait	1	Manual	5. Send 4 Byte SDO Write: Map 2nd Object as User1 0x4500.00
628h	8	2F 00 16 00 02 00 00 00	Wait	1	Manual	6. Send 1 Byte SDO Write: Enable Map/Set length to 2
628h	8	2B 00 14 05 C8 00 00 00	Wait	1	Manual	7. Send 2 byte SDO Write: Set PDO Timeout to 200ms
628h	8	23 00 14 01 28 02 00 00	Wait	1	Manual	8. Send 4 Byte SDO Write: Enable PDO

The resulting PDO Mapping record:

PDO Record	Mapped Object Data	Description
1400.01	00 00 02 28h	RPDO1 communication object enabled for node 0x28
1400.05	00 C8h	RPDO1 timeout set to 200 ms
1600.00	02h	RPDO1 map length of 2
1600.01	33 66 00 10h	RPDO1 first mapped object 3366.00 (*VCL_Throttle*) with 16 bit length
1600.02	45 00 00 08h	RPDO1 second mapped object 4500.00 (User1) with 8 bit length in RPDO1

This resulting PDO mapping record is viewable in the *Programmer » Applications Setup » CAN Interface » RPDO Setups* menu.



APPENDIX B

VEHICLE DESIGN CONSIDERATIONS

ELECTROMAGNETIC COMPATIBILITY (EMC)

Electromagnetic Compatibility (EMC) encompasses two areas: emissions and immunity. Emissions are radio frequency (RF) energy generated by a product. This energy has the potential to interfere with communications systems such as radio, television, cellular phones, dispatching, aircraft, etc. Immunity is the ability of a product to operate as intended in the presence of RF energy generated by other sources as well as itself. EN12895 is the relevant EMC standard for the CE marking of industrial trucks intended for sale in Europe and some other countries.

EMC Compliance is ultimately a system requirement. Part of the EMC performance is designed into or inherent in each component of a system; another part is designed into or inherent in end product/system characteristics such as shielding, wire routing, individual component layout and a portion is a function of the interactions between all these parts. The techniques presented below can help reduce the risk of EMC problems in products that incorporate Curtis motor controllers.

Emissions

High frequency signals can produce RF emissions that are measurable during Radiated Emissions testing. Long cable and wire harness runs essentially become antennas for the emissions to travel. Therefore, emission reduction techniques include making the battery and motor cables as short as possible. Minimize the lengths of the AMPseal connector wire harness runs and the formation of wire loops. Further emission decreases may include using shielded cables or ferrites on the control wires and twisting the motor and battery cables. Route the battery and AC motor cables separate from the control wires. When separating control wires and the battery/motor cable routing is not possible, cross them at right angles.

RF Immunity

Radiated immunity problems may occur when the controller is located close to other devices generating high RF energy. Possible ways to help prevent other devices from interfering with a Curtis controller include:

- Placing the controller as far as possible from such noise sources.
- Shield the controller from the noise.
- Enclose the controller in a metal box and add proper ferrites to all cabling entering and leaving it.
- Other possible solutions include the use of ferrite beads at the RF noise source(s) to prevent the noise from traveling along the wiring harness and cross conducting onto sensitive wires and common connections.

ELECTROSTATIC DISCHARGE (ESD) IMMUNITY

Quick Link:
Tables 4 and 5 p.13

Curtis motor controllers contain ESD-sensitive components. It is therefore necessary to protect them from ESD damage. See Tables 4 and 5 for the controller ESD ratings.

ESD immunity is improved by either providing sufficient distance or isolation between conductors and the ESD source so that a discharge will not occur.

DECOMMISSIONING AND RECYCLING THE CONTROLLER

The controller is for installation into an Original Equipment Manufacturer (OEM) vehicle. As a component, it has no function unless installed as part of a vehicle's electrical or electro-hydraulic control system.

For controller decommissioning and recycling:

1. Follow the OEM's vehicle decommissioning instructions.
2. Follow all applicable landfill directives or regulations for Electrical and Electronic Equipment (EEE) waste.

APPENDIX C

EN 13849 COMPLIANCE

EN 13849 COMPLIANCE

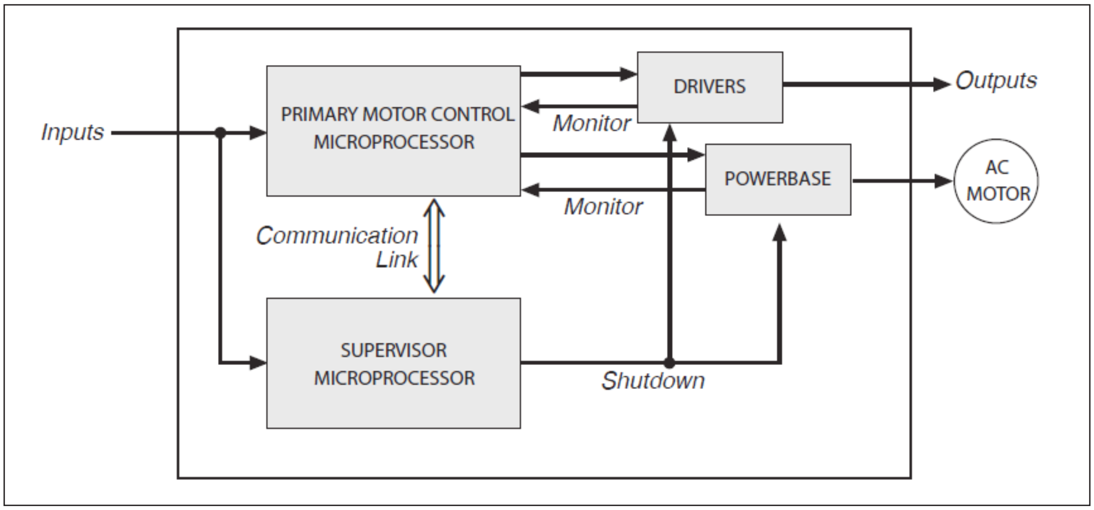
Since January 1, 2012, conformance to the European Machinery Directive has required that the Safety Related Parts of the Control System (SRPCS) be designed and verified upon the general principles outlined in EN13849. EN13849 supersedes the EN954 standard and expands upon it by requiring the determination of the safety Performance Level (PL) as a function of Designated Architecture plus Mean Time To Dangerous Failure (MTTFd), Common Cause Faults (CCF), and Diagnostic Coverage (DC). These figures are used by the OEM to calculate the overall PL for each of the safety functions of their vehicle or machine.

The OEM must determine the hazards that are applicable to their vehicle design, operation, and environment. Standards such as EN13849-1 provide guidelines that must be followed in order to achieve compliance. Some industries have developed further standards (called type-C standards) that refer to EN13849 and specifically outline the path to regulatory compliance. EN1175-1 is a type-C standard for battery-powered industrial trucks. Following a type-C standard provides a presumption of conformity to the Machinery Directive.

Curtis Enhanced AC Motor Controllers comply with these directives using advanced active supervisory techniques. The basic "watchdog" test circuits have been replaced with a Supervisor microcontroller that continuously tests the safety related parts of the control system; see the simplified block diagram in Figure C-1.

Figure C-1

Supervisory system in the Curtis AC motor controller



The Supervisor and Primary motor control processors run diagnostic checks at startup and continuously during operation. At startup, the integrity of the code and NV Memory are ensured through CRC checksum calculations. RAM is pattern checked for proper read, write, and addressing. During operation, the arithmetic and logic processing unit of each micro is cyclically tested through dynamic stimulus and response. The operating system timing and task sequencing are continuously verified. Redundant input measurements are crosschecked over 30 times per second, and operational status information is passed between microprocessors to keep the system synchronized. Any faults in these startup tests, communication timing, crosschecks, or responses will command a safe shutdown of the controller, disabling the driver outputs and motor drive within 200 ms.

To mitigate the hazards typically found in machine operations, EN13849 requires that safety functions be defined; these must include all the input, logic, outputs, and power circuits that are involved in any potentially hazardous operation. Two safety functions are defined for Curtis Enhanced AC Motor Controllers: Uncommanded Powered Motion and Motor Braking Torque.

The Travel Control (Uncommanded Powered Motion) safety function provides detection and safe shutdown in the following circumstances: faulted throttle; improper sequence of forward/reverse switches, throttle, and interlock; incorrect direction of travel; loss of speed control or limiting; uncommanded movement; or movement at start-up. The Braking Torque safety function provides detection and safe shutdown in the event of the loss of braking torque, position/hill hold, or emergency reverse.

Curtis has analyzed each safety function and calculated its Mean Time To Dangerous Failure (MTTFd) and Diagnostic Coverage (DC), and designed them against Common Cause Faults (CCF). The safety-related performance of the F-Series controller are summarized as follows:

AC F2-A

Safety Function	Category	DC_{avg}	$MTTF_d$	PL	CCF
Travel Control	2	94.2 %	>293 yrs.	D	Pass
Prevent of Travel (Interlock Braking)	2	91.5 %	>163 yrs.	D	Pass
Emergency Reverse	2	91.6 %	> 168 yrs	D	Pass
Speed Reduction (Limitation)	2	90.5 %	> 185 yrs	D	Pass
Load Handling Control	2	94.8 %	> 301 yrs	D	Pass

AC F4-A

Safety Function	Category	DC_{avg}	$MTTF_d$	PL	CCF
Travel Control	2	74.4 %	>366 yrs.	D	Pass
Prevent of Travel (Interlock Braking)	2	66.6 %	>215 yrs.	D	Pass
Emergency Reverse	2	69.8 %	> 185 yrs	D	Pass
Speed Reduction (Limitation)	2	69.9 %	> 186 yrs	D	Pass
Load Handling Control	2	66.3 %	> 227 yrs	D	Pass

AC F6-A

Safety Function	Category	DC_{avg}	$MTTF_d$	PL	CCF
Travel Control	2	84.3 %	>453 yrs.	D	Pass
Prevent of Travel (Interlock Braking)	2	77.0 %	>287 yrs.	D	Pass
Emergency Reverse	2	79.0 %	> 254 yrs	D	Pass
Speed Reduction (Limitation)	2	79.0 %	> 254 yrs	D	Pass
Load Handling Control	2	77.4 %	> 313 yrs	D	Pass

EN1175-1:1998+A1:2010 specifies that traction and hydraulic electronic control systems must use **Designated Architecture 1** or greater. This design employs input, logic, and output circuits that are monitored and tested by independent circuits and software to ensure a high level of safety performance (up to PL=D).

Mean Time To Dangerous Failure (MTTFd) is related to the expected reliability of the safety related parts used in the controller. Only failures that can result in a dangerous situation are included in the calculation.

Diagnostic Coverage (DC) is a measure of the effectiveness of the control system's self-test and monitoring measures to detect failures and provide a safe shutdown.

Common Cause Faults (CCF) are so named because some faults within a controller can affect several systems. EN13849 provides a checklist of design techniques that should be followed to achieve sufficient mitigation of CCFs. The CCF value is a pass/fail criterion.

Performance Level (PL) categorizes the quality or effectiveness of a safety channel to reduce the potential risk caused by dangerous faults within the system with "A" being the lowest and "E" being the highest achievable performance.

APPENDIX D

PROGRAMMING, MONITORING, and DIAGNOSTIC SOFTWARE

The AC F-Series controller's parameter values are changed ("programmed") using the Curtis Integrated Toolkit™ (CIT) or the Curtis 1313 handheld programmer (1313 HHP). CIT is a PC based program, while the 1313 HHP is a self-contained hand-held programmer. Both tools communicate/ interface with the F-Series controller via the CANbus (controller's CAN1). The choice to use either tool depends upon the extent the user will set up or modify a controller, the application complexity, or the interaction with the other CAN enabled devices on a vehicle or system CANbus. This appendix summarizes these programming tools. Illustrations are of an F4 using CIT version 1.5.0, which is common to the entire F-Series controllers. Not all aspects and options with the CIT program are covered here. For complete details, consult the respective datasheets and user manuals for specifications and instructions*.

CURTIS INTEGRATED TOOLKIT™

The Curtis Integrated Toolkit™ is a software program for configuring and communicating with Curtis Instruments products. It handles all communications to the controllers, gauges, and modules on a CAN network. It does require a 3rd-party interface from the computer (PC) to the CANbus — and is compatible with many USB-to-CAN interface dongles from Peak, Kvaser, iFAC, Sondheim, etc. The CIT program comprises the following applications (apps) that run in a shared environment:

Launchpad

- Launchpad is the CIT starting point — the opening and main window.
- Create, save, and manage the controller application/vehicle "projects" using Launchpad.
- Launchpad controls the access to the other apps described below.

Programmer

Programmer is similar in function to the serial-based Curtis 1313/1314 programmers' *Parameters*, *Monitor*, and *Diagnostic* features. Use Programmer to change/adjust/set the parameters (i.e., "*program the controller*"). The Programmer app is where the Monitor variables and faults are visible (accessed) in the CIT program.

VCL Studio

The VCL Studio app is a full-featured code editor and compiler for Curtis Instruments' Vehicle Control Language (VCL). Use it to create a new VCL program, import or edit an existing VCL program (source/text file), or export the project's VCL source code program as a text file. VCL Studio is a primary reason to select CIT over the 1313 HHP, and is required for F-Series applications that will use VCL.

Note: If the controller application follows the controller's default wiring example and only the parameters settings will be used to setup and tune the application (no VCL), then the 1313HHP is fully adequate to "program" and diagnose the F-Series controller.

**Contact the Curtis distributor or the regional Curtis sales office to obtain the Curtis Integrated Toolkit™ software and the 1313 HHP.*

Consult with the Curtis distributor's support engineer or the regional Curtis sales office for further help or training with the setup and use of these programming and diagnostic tools.

Menu Editor

The Menu Editor app is for modifying, grouping, or adding new parameters to the parameter or monitor menus. This is the app OEM engineers will use to control (customize) what parameters and monitor variables are changeable/available to other CIT users, by use of access levels. The CIT license or 1313 HHP model determines what access levels are available to a user.

Package and Flash

The Package and Flash app creates the application packages from the CIT project configuration. Application packages include the VCL program, parameter settings, and any menu changes saved as a project menu. Package and Flash is also the software downloader tool to flash application and device profile (.cdev file) software into the controller over the CANbus (the F-Series controllers do not have a serial communication port).

Note: Use VCL Studio to write and compile VCL. Then use Package and Flash to "package" the application and flash it into a controller. Once this is completed, a 1313 HHP can use the generated "packaged" .c13 file to flash the complete-application package into other/multiple controllers. See "extracting the .c13 file" and the 1313 HHP firmware update procedures to perform this process.

TACT

The TACT app is a more powerful CANbus version of the Curtis serial-based TACT, allowing an oscilloscope-style signal trace of a controller's real-time operations. Signals are device profile variables that include parameters, monitor variables and faults. TACT has the ability to acquire and display up to 16 signal traces. Note: Signals displayed in TACT include the Controller-to-CIT processing time for the variable. This includes the VCL program loop timing, Primary and Supervisor processing time, and the update rate of the CIT-to-Controller communications. Delays in signals (e.g., driver outputs) observed in TACT include these timing factors. Oscilloscope traces will produce slightly different results. More complex VCL programs and slower CIT baud rates may widen the differences.

Projects

Projects in CIT are like having a file cabinet of information containing all the Programmer menus, parameter values, VCL files, TACT traces, and the list of electronic devices that are part of the controller's application. Projects may consist of multiple CAN compatible controllers, gauges, expansion or contactor modules, and displays. A project must either be created in or imported into Launchpad to start working with the Curtis Integrated Toolkit™ program, even for a single controller application.

Device Profile

Controllers need more than just the OS (operating system) software to work with CIT. They need a Device Profile. Device profiles contain the information CIT needs to understand how to communicate and work with a specific device. Every compatible controller, gauge or module will have a device profile. Device profiles include data such as the controller or instrument model, factory menus, parameter defaults, the device firmware, and the operating system (OS) software. Furthermore, the device profile (a *.cdev* file) contains the information each of the application tools use to interact with that device. For CIT to work, obtain the proper device profile (*.cdev* file) for each of the devices contained within the application project. *The device profile, similar to the OS in the E/SE controllers using the WinVCL program, is required to create a project. Contact the Curtis distributor or the regional Curtis sales office to obtain the device profile (as a .cdev file) that is applicable to the controller's project/ application.*

The Curtis Integrated Toolkit™ apps all startup in Launchpad from either the individual device (icon) level and/or at the project's system level (icon). Apps that start (launch) from a highlighted device icon work with the selected device. Some apps, like VCL Studio, operate in this manner because the

VCL program is specific to that device. Selecting the project's system level icon in Launchpad brings up the list of apps that can work with the whole system simultaneously. Launching from the system level allows data collection from all the CANbus connected devices. Some apps, such as Package and Flash, can only be run from the system level.

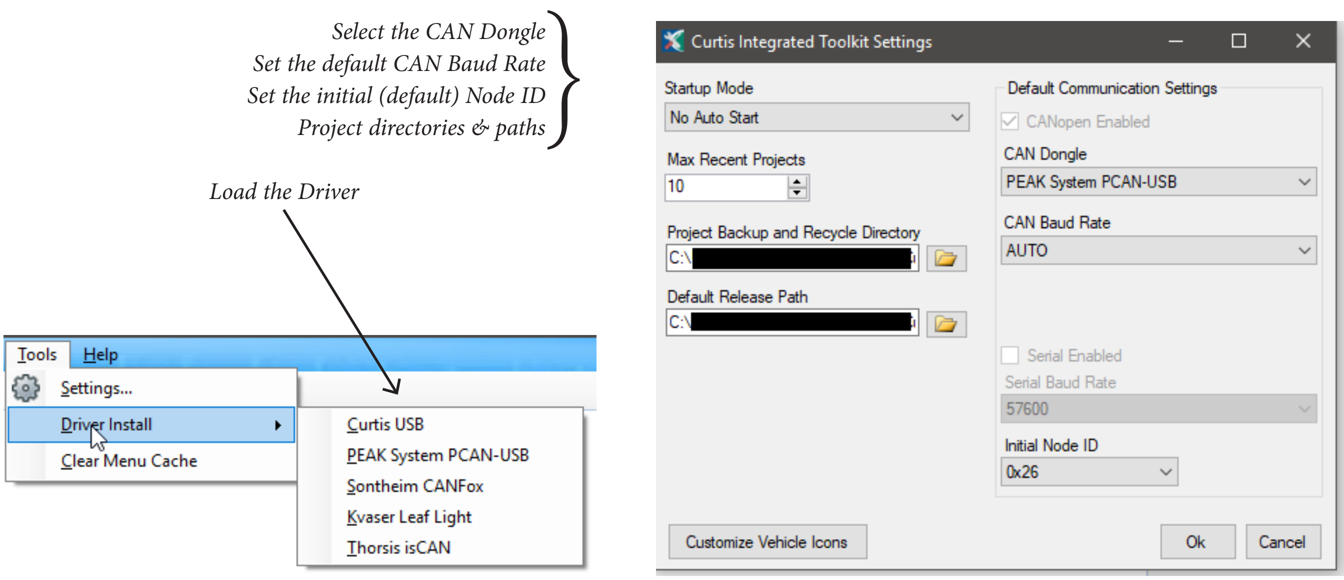
The CIT registration key controls what data is visible or editable, and what apps may be available to the user. There are five user levels.

> The *OEM-Factory* is the most advanced level. The OEM manufacturer's engineering staff will use this level to customize the application and set/restrict the lower-level users' access to critical settings.
>
> The *OEM-Dealer* access level is for the OEM Dealership's technical staff. Based upon the vehicle, access to parameters to customize the vehicle further is common, as well as complete diagnostic access.
>
> The *Field-Advanced* and *Field-Intermediate* levels are for site technicians and fleet maintenance staff or users. Typically, these users will not edit parameters that change the vehicle's characteristics, but may report diagnostic feedback.
>
> The *Field-Basic* level is well suited for the end-user or a location's staff, who will not be changing parameter values or conducting detailed vehicle or device diagnostics.

This manual uses the *OEM-Factory* access level. Therefore, not all items described in this manual may be applicable (or visible) to all CIT users.

Choose the CAN Interface Device Driver

CIT requires a CAN dongle to connect to the controller CAN port and the PC. If not already completed, install the appropriate driver. From the Launchpad, do this under the Tools menu. Then use the settings option to complete the CIT settings.



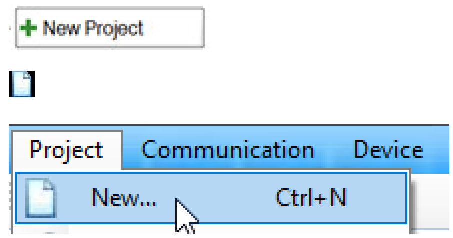
Creating a Project with a new F-Series controller

Always review the instructions and terminology available within the version of CIT in use. (For example, these images are from the Curtis Integrated Toolkit™ program, version 1.5.0).

When starting a new F-Series application, begin by installing/wiring the controller as per this manual. Then, open Launchpad, but do not select "Scan for Devices" or, if previous CIT projects are listed, do not select those either. Rather, use any of the four methods shown below to open the Project Explorer dialog box. New projects will be created (set up) using this Project Explorer window. The first step in creating a new project is to create the project itself. Devices are then *added* to the project (see Step 2c). Devices added to a project (i.e., within a saved project) can always be edited or deleted from the project. The steps to create a project with an F-Series controller are as follows:

Step 1: Open the Project Explorer dialog box (window) by using one of these New Project methods.

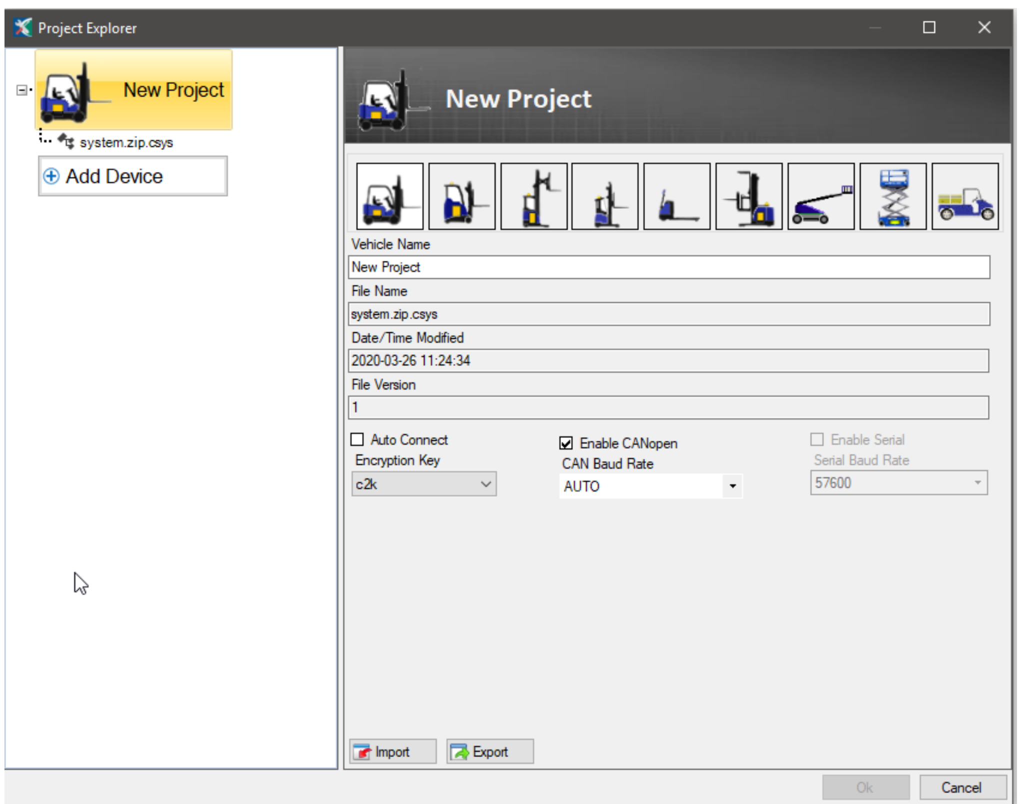
1



Ctrl + N

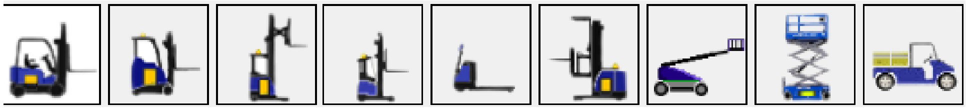
Step 2 – 3: Within Project Explorer, work from the top-down completing each entry. If an entry's value is unknown, use the default. Entries are always editable by re-opening the saved project's Project Explorer window and following these same steps.

2



2 Step 2 sets up the overall project (the vehicle in essence). The controllers will be added in step 3.

Select the Project icon that best represents the project/application vehicle.



Alternatively, create and select a custom image as described in the CIT instructions (***Customize Vehicle Icons***).

Vehicle Name: This name will appear next to the project vehicle image, replacing the "Default" as shown in the above image. Use a meaningful name for the vehicle, project, or application. Often, the name used here is the project name and thus the project file (.cprj) name.

The greyed-out items are not accessible and/or updated with each project 'save' operation:

File Name

Date/Time Modified

File Version

Auto Connect: To start the device communication (i.e., connect CIT to the CANbus) when opening the project, check this box. Note that this is not the same as the "Auto-Load Last Project" in the Settings/Startup Mode option.

Enable CANopen: Enable CANopen is active by default. Unchecking this box will disable CAN (*Do not do it!*). Leave this box checked.

Notice: This is a *future feature* designed to enable connection over the serial bus for devices so configured. Items greyed out are not applicable at this time.

CAN Baud Rate: If the controller's (device) baud rate is known, select it from the pull-down menu. Else, select AUTO.

3 This step is where the controller (device) becomes part of the project. In order to add a device to the project, complete these three selections, then click the OK button.

Add Device: Select the Add Device to open the Select Device dialog box, pictured above.

Device Name: Enter a name that represents the device within the project. For example, when setting up a dual-drive system, naming the traction controllers "left" and "right" will be helpful when editing the parameters. If this is an AC pump motor controller, name it as such. The device name is editable, later, using Project Explorer.

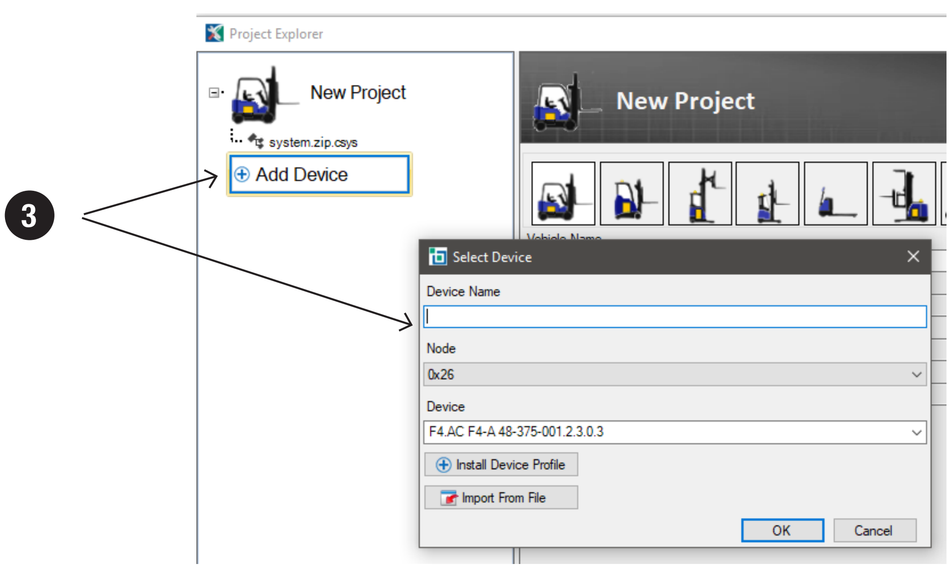
Node: Input the device's Node ID that the project-application or vehicle system will use. The Node ID is a hexadecimal number. Each device must have its own Node ID. Once a Node ID is allocated, that Node ID will not be available for other devices (allocated Node IDs are greyed-out in the pull-down menu).

Note: Adding more than one generic device to a project will cause an initial problem because they come from the factory with the same (generic) node ID. When adding generic devices with the same 'factory installed' node ID, <u>add one device at a time</u>. See "Setting up Multiple Generic Devices" (below) to add multiple generic devices (or devices with the same initial node ID).

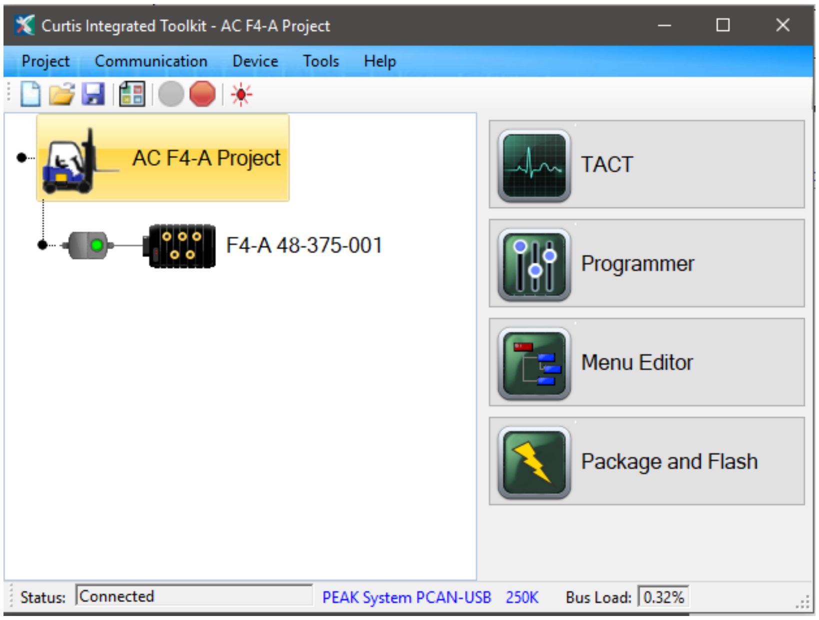
Device: Select the device profile (.cdev) matching the device (the F-Series controller in this case). If the device profile was added to CIT using the device pull-down in the Launchpad menu bar, select it from the drop list. Else, use the Install Device Profile option to navigate to the file location of the device profile (.cdev). Select OK when complete.

Note: Contact the Curtis distributor or the regional Curtis sales office to <u>obtain the individual cdev file (.cdev) that is applicable to the project's controller/device.</u> This is typically the same cdev in the controller as delivered from the factory. To set up a project and use VCL, the controller's individual cdev file has to be "<u>added</u>" into the CIT project in this step. (This .cdev file is similar to the individual OS file when using WinVCL for the serial-bus based controllers).

Use the Import From File option to add a device that has already been configured and saved from another project. This option will open the computer's MS Windows Explorer. Navigate to the location on the computer/network where the device's cnode (.cnode) file is located and select the file.



When the new project is complete, be sure to save the project! When connected, it will look similar to the image below.



Managing Devices within a Project

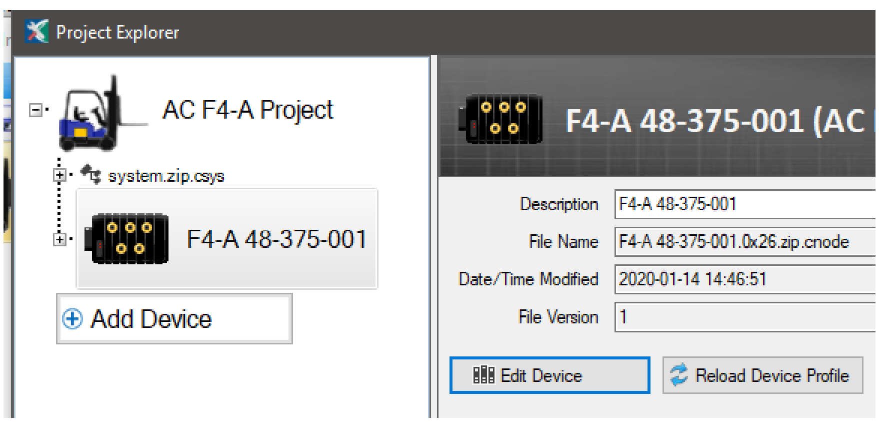
<u>… How to update a device's device profile (cdev)</u>

Always review the instructions and terminology available within the version of CIT in use. (For example, these images are from the Curtis Integrated Toolkit™ program, versions 1.5.0).

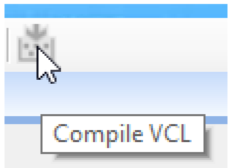
To manage an existing device within a project, <u>disconnect CIT from the CANbus</u> ( ).

Connected *Disconnected*

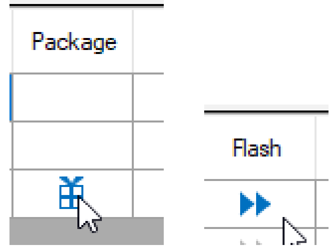
Next, open the Project Explorer window ( ). With Project Explorer open, select (highlight) the device to be edited and the device window will open as illustrated. Select the  button, which will open a similar dialog box as in Step 3. Note, within this dialog box, the device name, as it appears in the project, is editable. Be sure to save the project after making changes.



This completes the update or change to the device profile within the project. To <u>load this device profile *into* the controller,</u> complete the steps in the Package and Flash app, including re-compiling the VCL program (if one) within the VCL Studio application.



VCL Studio: Compile VCL



"Package" *"Flash"*

Package and Flash app

The c13 file is the project's "packaged" version. The file contains all customized programmer menu(s), the parameter settings (values), the compiled VCL program, and the OS. Use it with the Curtis 1313 HHP to transfer a project's device profile (in essence) to new/different, but matching devices (i.e., F4-A to F4-A controllers). Use the c13 file to complete the most detailed 'update' to vehicles in the field using just the 1313 HHP.

Extracting the .c13 file

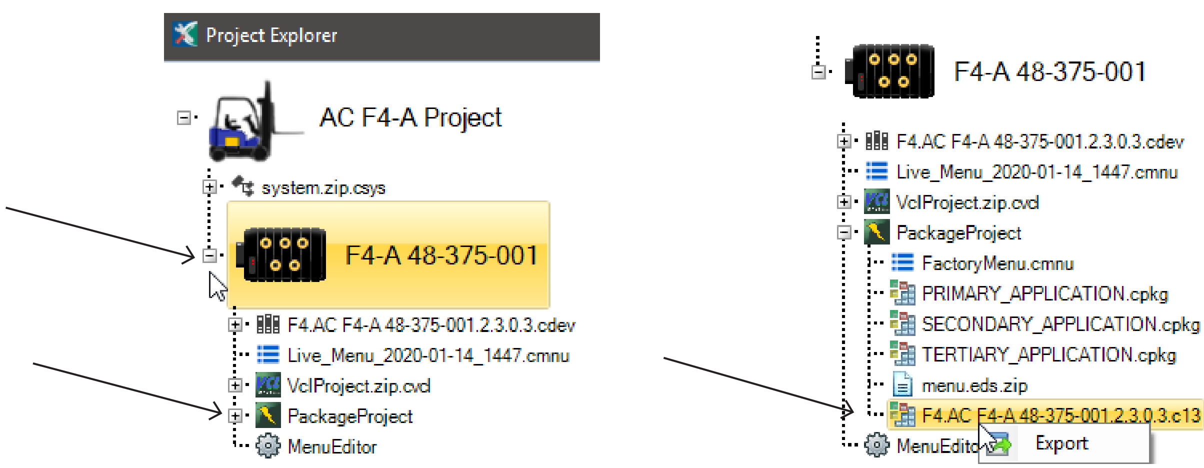
… For use by the 1313 HHP to flash a controller's configuration into matching controllers.

Always review the instructions and terminology available within the version of CIT in use. (For example, these images are from the Curtis Integrated Toolkit™ program, version 1.5.0 and the above project.)

The 1313 HHP offers the means to download a CIT Project's controller configuration into other compatible controllers. Referred to as "flashing", it is similar to the process used to update the specific project's (modified) OS in the E/SE series of controllers. The F-Series controllers use a CIT generated .c13 file. Follow this summary to obtain the "packaged" .c13 file. The process of using this .c13 file is in the 1313 HHP user's manual, Chapter 8 – Flash, *Firmware Update*. The manual, *CANbus 1313 HHP, 53225 Rev A 3/18*, is downloadable from the Curtis website.

https://curtisinstruments.com/products/programming/

With CIT connected to the device, select and open Project Explorer, highlight the controller of interest, and then expand the device node (click to expand) to view the files within. Expand the PackageProject node to access the project's .c13 file.



Highlight the .c13 file, then right-click and select the Export option.

Note the device's project name and device profile version (.cdev = 2.3.0.3 in this example). The reader's actual cdev version may be different from this (dated) example. The name aspect will match the device name in the CIT project.

This .c13 file is the project's "packaged" version. The file contains all customized programmer menu(s), the parameter settings (values), the compiled VCL program, and the OS. Use it with the Curtis 1313HHP to transfer a project's device profile (in essence) to new/different, but matching devices (i.e., F4-A to F4-A controllers). Use the .c13 file to complete the most detailed 'update' to vehicles in the field using just the 1313 HHP.

CURTIS 1313 HHP

The Curtis 1313 Handheld Programmer (1313 HHP) performs programming and troubleshooting tasks for all Curtis programmable motor controllers, gauges, and control systems. The 1313 HHP connects to Curtis devices in one of two ways specific to the device: Either directly via the device's RS232 serial port (applicable to the E/SE and serial-bus controllers), or through a CANbus connection (applicable to the F-Series controllers and CAN devices). The 1313 HHP also connects to a PC using its mini-USB port for transferring files and firmware updates. The cables specific to the connection types are included in the 1313 HHP soft case. Specific models match the Curtis Integrated Toolkit™ program's access levels.

Note: The F-Series controllers do not support the previous grey-band serial-based 1313 HHP.

The generic CANbus 1313 HHP is model 1313-xx31. The user's manual, *CANbus 1313 HHP, 53225*, is downloadable from the Curtis website: https://curtisinstruments.com/products/programming/

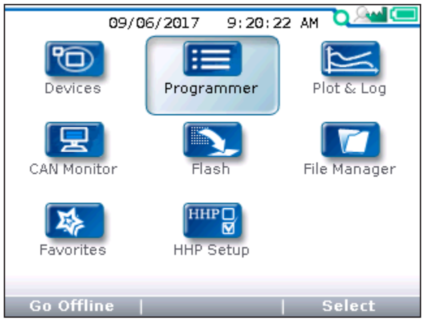
As described in the 1313 HHP manual (Flash app), the 1313 can download a project configuration into the F-Series controllers using a c13 file. This is the recommended method for updating multiple vehicles at customer locations. A c13 file clones compatible controllers. How to obtain the c13 file is described above, in the CIT section. Power the 1313 according to the vehicle CAN port wiring specification (next page) to ensure it remains powered during a c13 flash. Do not rely on the controller +12V power supply, as the controller flash/OS-upate will turn it off. The 1313 batteries are not sufficient for flashing a controller. Set the 1313 power-off setting to a greater duration than the c13 flash, which is greater than 5 minutes.

Furthermore, if only a project's parameter settings (*values*) need to be loaded (updated) into other controllers, then use the 1313HHP to save the parameter settings from a given controller and load them into identical controllers. This procedure is similar to the "cloning" task in the E/SE series controllers using .cpf files. In the F-Series controllers, the file that the 1313 HHP extracts is a .cdf (*clone data file*). The .cdf file is an excellent way to share the parameter settings with colleagues and support engineers. For more detail on this topic, see Chapter 8, Flashing, in the 1313 HHP manual: Save *.cdf file* and Restore *.cdf File*. Additionally, within the CIT Programmer app, in the File tab, Export option, select the "Defaults as 1313 Clone data" to obtain a .cdf (clone data file) for use in the 1313.

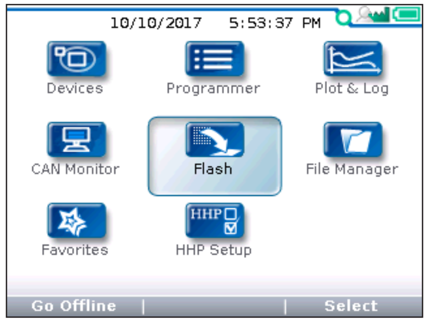
For parameter programming and diagnostic purposes, the 1313 HHP's Programmer app matches the look and function of the Programmer app in the Curtis Integrated Toolkit™ program. A Plot & Log app offers a similar, but limited, function to TACT. The 1313 HHP CAN Monitor app captures CANbus traffic over time and saves the trace as a .csv file. Open and manage the data in the .csv file using a PC spreadsheet program (e.g., Excel). Transfer the CAN Monitor .csv file to a PC using the SD card or the USB connection.

The 1313 HHP is an excellent choice for the field technician performing vehicle service, maintenance, and software update tasks. In addition to its onboard internal memory, SD cards provide an additional 64 MB of memory and can be used for transferring files between the 1313 HHP and either a PC or another 1313 HHP.

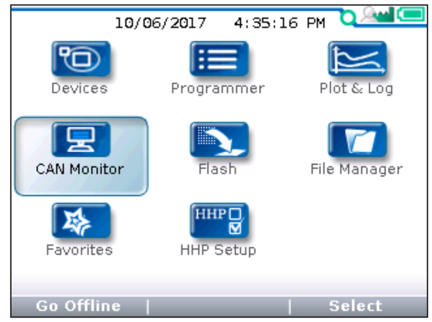
1313HHP_ Model 1313-xx31



1313 HHP Programmer App



1313 HHP Flash App



1313 HHP CAN Monitor App

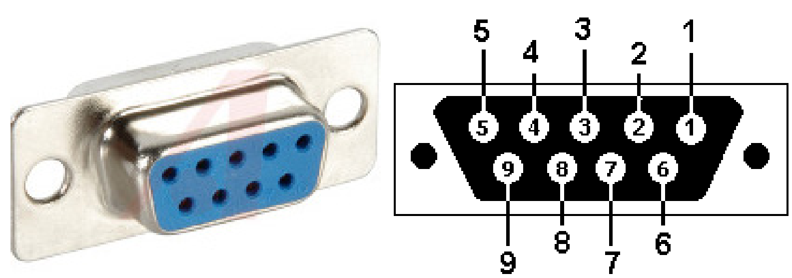
VEHICLE CAN PORT WIRING

Install the vehicle CAN port as specified in CiA 303-1, Section 5.4, including the recommended output voltage at the optional power supply of +18 VDC < V+ < +30 VDC in order to enable the use of standard power supplies (24 VDC).

The Curtis 1313-xx31 Handheld Programmer can use batteries for its power. The preferred method is to use the vehicle's battery voltage (i.e., 24V) to power the handset. To use the vehicle battery, install on the vehicle a 9-pin D-Sub female connector as illustrated, below.

Both the PC based Curtis Integrated Toolkit™ program that uses a USB/CAN dongle and the Curtis 1313-xx31 Handheld Programmer use male (pins) D-Sub connectors. Use D-Sub gender changers should a conflict occur.

Typically, the USB/CAN dongle uses the USB power, so the power pins are not used. For dongles not conforming to CiA 303-1, to prevent damage, check the documentation of the CAN dongle before connecting to a powered D-sub port.



1313-xx31 Wiring	
D-Sub Pin	**Function**
7	CAN_H
2	CAN_L
9	B+ (8 – 36V)
6	B–

Vehicle CAN Port Wiring for 1313-xx31 Programmer

APPENDIX E

SPECIFICATIONS: CONTROLLER

Table E-1 Specifications: AC F6-A, F4-A and F2-A Controllers

MODEL CHART

Model Number	Nominal Battery Voltage	Current Rating[1]	1-Hr Current Rating[2]	Controller Lifetime Rating[3]	Internal 120Ω CAN Termination
AC F2-A 24-120-001	24V	120A^{rms}	120A^{rms}	40A^{rms}	Yes
AC F2-A 24-120-051	24V	120A^{rms}	120A^{rms}	40A^{rms}	No
AC F2-A 24-200-001	24V	200A^{rms}	150A^{rms}	50A^{rms}	Yes
AC F2-A 24-200-051	24V	200A^{rms}	150A^{rms}	50A^{rms}	No
AC F2-A 24-240-001	24V	240A^{rms}	150A^{rms}	60A^{rms}	Yes
AC F2-A 24-240-051	24V	240A^{rms}	150A^{rms}	60A^{rms}	No
AC F2-A 24-280-001	24V	280A^{rms}	130A^{rms}	70A^{rms}	Yes
AC F2-A 24-280-051	24V	280A^{rms}	130A^{rms}	70A^{rms}	No
AC F2-A 48-150-001	36-48V	150A^{rms}	130A^{rms}	38A^{rms}	Yes
AC F2-A 48-150-051	36-48V	150A^{rms}	130A^{rms}	38A^{rms}	No
AC F2-A 48-240-001	36-48V	240A^{rms}	140A^{rms}	60A^{rms}	Yes
AC F2-A 48-240-051	36-48V	240A^{rms}	140A^{rms}	60A^{rms}	No
Model Number	**Nominal Battery Voltage**	**Current Rating[1]**	**1-Hr Current Rating[2]**	**Controller Lifetime Rating[3]**	**CAN Isolation IMU[5]**
AC F4-A 24-375-001	24V	375A^{rms}	270A^{rms}	160A^{rms}	No
AC F4-A 24-375-101	24V	375A^{rms}	270A^{rms}	160A^{rms}	Yes
AC F4-A 36-500-001	24-36V	500A^{rms}	210A^{rms}	160A^{rms}*	No
AC F4-A 36-500-101	24-36V	500A^{rms}	210A^{rms}	160A^{rms}*	Yes
AC F4-A 48-375-001	36-48V	375A^{rms}	185A^{rms}	160A^{rms}	No
AC F4-A 48-375-101	36-48V	375A^{rms}	185A^{rms}	160A^{rms}	Yes
AC F4-A 48-450-001	36-48V	450A^{rms}	185A^{rms}	160A^{rms}*	No
AC F4-A 48-450-101	36-48V	280A^{rms}	185A^{rms}	160A^{rms}*	Yes
AC F4-A 80-300-101	48–80V	300A^{rms}	125A^{rms}*	125A^{rms}*	Yes

Model Number	Nominal Battery Voltage	Current Rating[1]	1-Hr Current Rating[2]	Controller Lifetime Rating[3]	CAN Isolation IMU[5]
AC F6-A 36-650-001	24–36V	650A^{rms}	265A^{rms}	150A^{rms}*	No
AC F6-A 36-650-101	24–36V	650A^{rms}*	265A^{rms}	150A^{rms}*	Yes
AC F6-A 48-650-001	36–48V	650A^{rms}	240A^{rms}	135A^{rms}*	No
AC F6-A 48-650-101	36–48V	650A^{rms}*	240A^{rms}	135A^{rms}*	Yes
AC F6-A 80-450-001	48–80V	450A^{rms}*	175A^{rms}	105A^{rms}*	No
AC F6-A 80-450-101	48–80V	450A^{rms}*	175A^{rms}	105A^{rms}*	Yes

** Subject to change, please contact your Curtis sales representative for more information.*
[1] Traction max current defined by a two minute S2-2 min test with 100% current loading applied.
[2] Traction 1-hr rating is by a S2-60 min rating.
[3] Controller Lifetime Rating calculated @ 60% traction modulation depth.
[4] The isolated CAN option applies to both CAN ports. Each isolated port is without internal 120-Ω termination.
[5] The -101 models include CAN isolation and the Inertia Measurement Unit (IMU).

PWM operating frequency: 10 kHz.

Maximum encoder frequency: 30 kHz. (see Chapter 2 for higher encoder frequency values)

Maximum controller output frequency: 599 Hz.

Electrical isolation to heatsink: 500 Vac (minimum).

Power Terminals: 5 (B+, B–, U, V, W).

Logic Connector Pin Count: 23 or 35 (controller basis).

Logic Connector Current Rating: 8A per pin (Maximum).

KSI Inrush Current: ≤ 10A, Max ≤ 2ms overall, with an initial peak duration ≤ 20 µs, at room temperature.

Precharge Current: 2A for < 1 sec (typical).

Storage ambient temperature range: –40°C to 95°C (–40°F to 203°F).

Operating ambient temperature range: –40°C to 50°C (–40°F to 122°F).

Internal heatsink operating temp. range: –40°C to 95°C (–40°F to 203°F).

Maximum Operating Altitude: 3,000 meters (9842.5 feet)

Package environmental rating: IP65 per IEC60529.
Note: Compliance requires AMPSEAL 23 or 35-pin receptacle [plug] connector with all wire silos sealed.

Thermal cutback: Controller linearly reduces maximum current limit with an internal heatsink temperature from 85°C (185°F) to 95°C (203°F); complete cutoff occurs above 95°C (203°F) and below –40°C (–40°F).

Design life: F2-A 24V = 8,000 hours
F2-A 48V 150A = 8,000 hours
F2-A 48V 240A = 2,000 hours
F4-A and F6-A = 20,000 hours

Operating duration at max. current: All current ratings are rms values per motor phase. S2–60 minute current is the typical current achievable before thermal cutback occurs, with an ambient temperature of 25°C and the controller mounted to a 6 mm thick vertical steel plate with 6 km/h (1.7 m/s) airflow perpendicular to the plate.

	F4/6-A (35 pin)	F2-A (23 pin)
Weight:	1.9 kg (4.4 lbs.).	1kg (2.2 lbs.)
Dimensions, WxLxH:	180 x 140 x 75 mm (7.08 x 5.51 x 2.95 inches).	120 x 155 x 55 mm (4.73 x 6.10 x 2.16 inches)

Baseplate material:	Aluminum.	
Baseplate: (bottom surface)	Roughness grade of N8 (ISO 1302), with a flatness tolerance of 0.3 mm.	
Mounting holes:	4x Ø7 mm	2x Ø7mm and 2x Ø7.5 mm
I/O Connection:	35 Pin AMPseal. Mating receptacle housing: AMP p/n 776164-1.	23 Pin AMPseal. Mating receptacle housing: AMP p/n 770680-1.
	Gold-plated sockets: (default): AMP p/n 770520-3 (strip form), 7708554-3 (loose piece). Tin-plated sockets: (do not mate with gold pins) AMP p/n 770520-1 (strip form), 7708554-1 (loose piece). Silo plug: AMP p/n 770678-1. Hand crimper for wire harness terminals: AMP p/n 58440-1.	
Power Connections:	5x M6x1.0 — 6H ↓ 18 mm.	
EMC:	Designed to the requirements of EN 12895:2015.	
Safety:	Designed to the requirements of: EN 1175-1:1998+A1:2010. EN ISO 13849-1:2015 Category 2.	
UL:	UL recognized component per UL583 (pending). Note: Regulatory compliance of the complete vehicle system with the controller installed is the responsibility of the vehicle OEM.	