MANUAL

MODEL **1298**

INTEGRATED TRACTION & HYDRAULIC SYSTEM MOTOR CONTROLLER

OS 11 with VCL

1298 Manual, p/n 38272
5 August 2011
» Software version OS 11.0 «

CURTIS INSTRUMENTS, INC.
200 Kisco Avenue
Mt. Kisco, New York 10549 USA
Tel. 914.666.2971
Fax 914.666.2188
www.curtisinstruments.com

CONTENTS

FIGURES

TABLES

1 OVERVIEW

The Curtis 1298 motor controller combines AC traction motor control with DC pump motor control. Advanced motor drive software provides smooth control over full speed and torque, including regenerative braking and zero speed control.

The 1298 is designed primarily for Class III material handling vehicles needing solid state control of hydraulic pump motors. Generic software is included for Walkie and Walkie-Stacker applications. Other applications can be easily supported with alternative VCL programming.

Fig. 1 *Curtis 1298 integrated traction and hydraulic system motor controller.*

Like all Curtis controllers, the 1298 offers superior operator control of motor drive performance. **Features include:**

✓ High efficiency, field-oriented motor control algorithms

✓ Advanced Pulse Width Modulation technology for efficient use of battery voltage, low motor harmonics, low torque ripple, and minimized switching losses

✓ Extremely wide torque/speed range including full regeneration capability

✓ Smooth low speed control, including zero speed

✓ Special features and I/O for DC pump motor and hydraulic valve control, allowing economical control of the entire vehicle system

✓ Software selectable options for variable-speed lift and lower or single-speed lift and lower

More Features ☞

✓ Adaptation of control algorithm to motor temperature variation so optimal performance is maintained under widely varying conditions

✓ Real-time battery current, motor torque, and power estimates available

✓ Power limiting maps allow performance customization for reduced motor heating and consistent performance over varying battery state-of-charge

✓ Powerful operating system allows parallel processing of vehicle control tasks, motor control tasks, and user configurable programmable logic

✓ A wide range of I/O can be applied wherever needed, for maximum distributed system control

✓ Internal battery-state-of-charge, hourmeter, and maintenance timers

✓ Easily programmable through the Curtis 1313 handheld programmer and 1314 PC Programming Station

✓ CAN bus connection allows communication with other CAN bus enabled system components such as the Curtis TH-1 tiller head; protocol meets CANopen standards; other 11-bit identifier field CAN protocols can be custom configured through VCL

✓ Field-programmable, with flash downloadable main operating code

✓ Thermal cutback, warning, and automatic shutdown provide protection to traction motor and controller

✓ Rugged sealed housing and connectors meet IP65 environmental sealing standards for use in harsh environments

✓ Insulated metal substrate power base provides superior heat transfer for increased reliability

✓ Built-in Dual Drive software allows easy setup and control of typical dual-drive vehicles, without VCL.

> Note: If you have a dual-drive application, see the Dual Drive Addendum to the 1298 manual, part number 38272-DD.

Familiarity with your Curtis controller will help you install and operate it properly. We encourage you to read this manual carefully. If you have questions, please contact the Curtis office nearest you.

☞ Using the 1313 handheld programmer, you can set up the controller to perform all the basic operations. In this manual, we first show you how to wire your system and adjust its performance characteristics without the use of VCL. Then, in Section 7, we show you how to adjust the system using VCL (Vehicle Control Language), an innovative software programming language developed by Curtis. VCL interacts with a second, independent software realm resident in a powerful logic controller embedded within the 1298 controller.

2

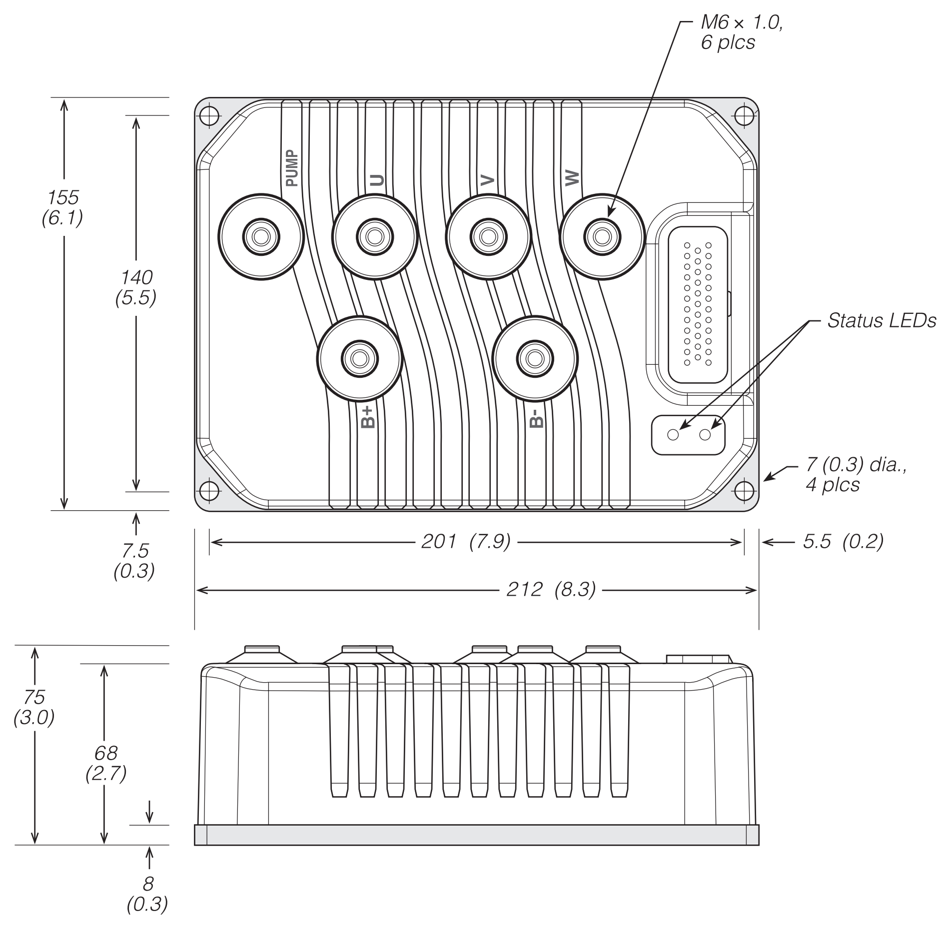
INSTALLATION AND WIRING

MOUNTING THE CONTROLLER

The outline and mounting hole dimensions for the 1298 controller are shown in Figure 2. When an Ampseal plug housing is mated with the 35-pin logic receptacle, this controller meets the IP65 requirements for environmental protection against dust and water. Nevertheless, in order to prevent external corrosion and leakage paths from developing, **the mounting location should be carefully chosen to keep the controller as clean and dry as possible**.

It is recommended that the controller be fastened to a clean, flat metal surface with four 6mm (1/4") diameter bolts, using the holes provided. A thermal joint compound can be used to improve heat conduction from the controller heatsink to the mounting surface. Additional heatsinking or fan cooling may be necessary to meet the desired continuous ratings.

Fig. 2 *Mounting dimensions, Curtis 1298 motor controller.*



Dimensions in millimeters (and inches)

You will need to take steps during the design and development of your end product to ensure that its EMC performance complies with applicable regulations; suggestions are presented in Appendix B.

The1234/36/38 controllers contain **ESD-sensitive components**. Use appropriate precautions in connecting, disconnecting, and handling the controller. See installation suggestions in Appendix B for protecting the controller from ESD damage.

CAUTION

Working on electrical systems is potentially dangerous. You should protect yourself against uncontrolled operation, high current arcs, and outgassing from lead acid batteries:

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

UNCONTROLLED HYDRAULIC OPERATION — Some conditions could cause the hydraulic system to run out of control. Disconnect the pump motor or make sure the hydraulic system has enough room to operate before attempting any work on the pump motor control circuitry.

HIGH CURRENT ARCS — Batteries can supply very high power, and arcing can occur if they 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.

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.

HIGH CURRENT CONNECTIONS

There are six high-current terminals, identified on the controller housing as **B+**, **B-**, **PUMP**, **U**, **V**, and **W**.

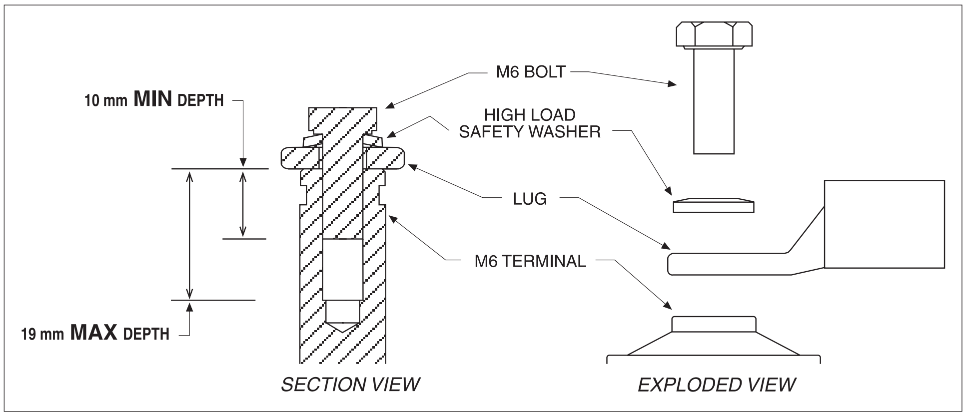
Table 1 High Current Connections

TERMINAL	FUNCTION
B+	Positive battery to controller.
B-	Negative battery to controller.
PUMP	Pump motor.
U	Motor phase U.
V	Motor phase V.
W	Motor phase W.

Lug assembly

Six M6 terminals are provided. Lugs should be installed as follows, using M6 bolts sized to provide proper engagement (see diagram):

- Place the lug on top of the terminal, followed by a high-load safety washer with its convex side on top. The washer should be a SCHNORR 416320, or equivalent.
- If two lugs are used on the same terminal, stack them so the lug carrying the least current is on top.
- Tighten the assembly to 10.2 ±1.1 N·m (90 ±10 in-lbs).



High current wiring recommendations

Battery cables (**B+**, **B-**)

These two cables should be run close to each other between the controller and the battery. Use high quality copper lugs and observe the recommended torque ratings. For best noise immunity the cables should not run across the center section of the controller. With multiple high current controllers, use a star ground from the battery **B-** terminal.

Pump wiring (**PUMP**)

Cable lengths should be kept as short as possible. Use high quality copper lugs and observe the recommended torque ratings. For best noise immunity the motor cables should not run across the center section of the controller. Low current signal wires should not be run parallel to the motor cables. When necessary they should cross the motor cables at a right angle to minimize noise coupling.

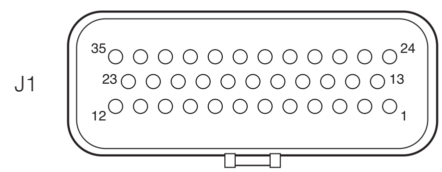
Motor wiring (**U**, **V**, **W**)

The three phase wires should be close to the same length and bundled together as they run between the controller and the motor. The cable lengths should be kept as short as possible. Use high quality copper lugs and observe the recommended torque ratings. For best noise immunity the motor cables should not run across the center section of the controller. In applications that seek the lowest possible emissions, a shield can be placed around the bundled motor cables and connected to the **B-** terminal at the controller. Typical installations will readily pass the emissions standards without a shield. Low current signal wires should not be run parallel to the motor cables. When necessary they should cross the motor cables at a right angle to minimize noise coupling.

LOW CURRENT CONNECTIONS

All low power connections are made through a single 35-pin AMPSEAL connector. The mating plug housing is AMP p/n 776164-1 and the contact pins are AMP p/n 770520-3. The connector will accept 20 to 16 AWG wire with a 1.7 to 2.7mm diameter thin-wall insulation.

The 35 individual pins are characterized in Table 2.



Low current wiring recommendations

Motor encoder (Pins 31, 32)

All four encoder wires should be bundled together as they run between the motor and controller logic connector. These can often be run with the rest of the low current wiring harness. The encoder cables should not be run near the motor cables. In applications where this is necessary, shielded cable should be used with the ground shield connected to the I/O ground (pin 7) at only the controller side. In extreme applications, common mode filters (e.g. ferrite beads) could be used.

CAN bus (Pins 21, 23, 34, 35)

It is recommended that the CAN wires be run as a twisted pair. However, many successful applications at 125 kBaud are run without twisting, simply using two lines bundled in with the rest of the low current wiring. CAN wiring should be kept away from the high current cables and cross them at right angles when necessary.

All other low current wiring

The remaining low current wiring should be run according to standard practices. Running low current wiring next to high current wiring should always be avoided.

Table 2 Low Current Connections

PIN	NAME	DESCRIPTION	RELATED VCL* FUNCTIONS	RELATED VCL* REFERENCES
1	KSI	Keyswitch input. Provides logic power for the controller and power for the coil drivers.	Setup_BDI	Keyswitch_Voltage
2	Prop. Driver	Proportional driver. This is a coil driver with current control capability typically used for a proportional valve on a hydraulic manifold. Can also be used as a digital input.	Automate_PWM Put_PWM	Sw_13 PWM5 PD_Current PD_Output PD_Throttle VCL_PD_Throttle
3	Driver 4	Generic driver #4. Typically used for the load hold valve. Can also be used as a digital input. Has low frequency PWM capabilities.	Automate_PWM Put_PWM	Sw_12 PWM4 PWM4_Output
4	Driver 3	Generic driver #3; can also be used as a digital input. Has low frequency PWM capabilities. Typically used for pump contactor.	Automate_PWM Put_PWM	Sw_11 PWM3 PWM3_Output
5	Driver 2	Generic driver #2; can also be used as a digital input. Has low frequency PWM capabilities and a slightly higher current rating.Typically used for electromagnetic brake.	Automate_PWM Put_PWM	Sw_10 PWM2 PWM2_Output
6	Driver 1	Generic driver #1; can also be used as a digital input. Has low frequency PWM capabilities. Typically used for main contactor.	Automate_PWM Put_PWM Set_Interlock Clear_Interlock	Sw_9 PWM1 PWM1_Output Interlock_State Main_State
7	I/O Ground	Input and output ground reference.		
8	Switch 2 Analog 2	Can be used as generic switch input #2 or as generic analog input #2. Typically used as the motor temperature analog input.		Sw_2 Analog2_Input Motor_Temperature
9	Switch 3	Generic switch input #3. Typically used as the interlock switch.		Sw_3
10	Switch 4	Generic switch input #4.		Sw_4

* The related VCL columns are vital when writing VCL code (see Section 7). VCL "functions" are used to access the various I/Os; VCL "references" are predefined names for specific pins.

Table 2 Low Current Connections, cont'd

PIN	NAME	DESCRIPTION	RELATED VCL	
			FUNCTIONS	REFERENCES
11	Switch 5	Generic switch input #5. Typically used for the Lift switch.		Sw_5
12	Switch 6	Generic switch input #6. Typically used for the Lower switch.		Sw_6
13	Coil Return	This is the coil return pin (at B+ potential) for all the contactor coils.		
14	Switch 16	Generic switch input #16.		Sw_16
15	Throttle Pot High	Pot high connection for a 3-wire throttle pot.		
16	Throttle Pot Wiper	Pot wiper connection for the throttle pot.	Setup_Pot Setup_Pot_Faults	Throttle_Pot Throttle_Pot_Output
17	Pot2 Wiper	Pot wiper connection for the hydraulic throttle pot.	Setup_Pot Setup_Pot_Faults	Brake_Pot Brake_Pot_Output
18	Pot Low	Common pot low connection for the throttle and brake pots.		Pot_Low_Output
19	Digital Out 6	An open collector digital output. Can also be used as a digital input.	Set_DigOut Clear_DigOut	Sw_14 DigOut6 Dig6_Output
20	Digital Out 7	An open collector digital output. Can also be used as a digital input.	Set_DigOut Clear_DigOut	Sw_15 DigOut7 Dig7_Output
21	CAN Term H	High connection for the CAN termination jumper.		
22	Switch 7	Generic switch input #7. Typically used as the Forward switch.		Sw_7
23	CANH	CAN bus high.	Setup_CAN Setup_Mailbox Send_Mailbox etc.	
24	Switch 1 Analog 1	Can be used as generic switch input #1 or as generic analog input #1. Typically used for emergency reverse switch (if applicable).		Sw_1 Analog1_Input
25	+12V Out	Unregulated low power +12V output.		Ext_Supply_Current
26	+5V Out	Regulated low power +5V output.		5_Volts_Output Ext_Supply_Current

Table 2 Low Current Connections, cont'd

PIN	NAME	DESCRIPTION	RELATED VCL	
			FUNCTIONS	REFERENCES
27	Pot2 High	Pot high connection for a 3-wire hydraulic throttle pot.		
28	Serial TX	Serial transmit line for display or flash update.	Setup_Serial	
29	Serial RX	Serial receive line for flash update.	Setup_Serial	
30	Analog Output	Low power, low frequency 0–10V analog output.	Automate_PWM Put_PWM	PWM6 Analog_Output
31	Encoder A	Quadrature encoder input phase A.		Motor_RPM MotorspeedA
32	Encoder B	Quadrature encoder input phase B.		Motor_RPM MotorspeedB
33	Switch 8	Generic switch input #8. Typically used as the Reverse switch.		Sw_8
34	CAN Term L	Low connection for the CAN bus termination jumper.		
35	CANL	CAN bus low.	Setup_CAN Setup_Mailbox Send_Mailbox etc.	

CONTROLLER WIRING: BASIC CONFIGURATION

A basic wiring diagram is shown in Figure 3. The throttles are shown in the diagram as 3-wire potentiometers; other types of throttle inputs are easily accommodated, and are discussed in the following throttle wiring section.

The main contactor coil must be wired directly to the controller as shown in Figure 3 to meet EEC safety requirements. The controller can be programmed to check for welded or missing contactor faults and uses the main contactor

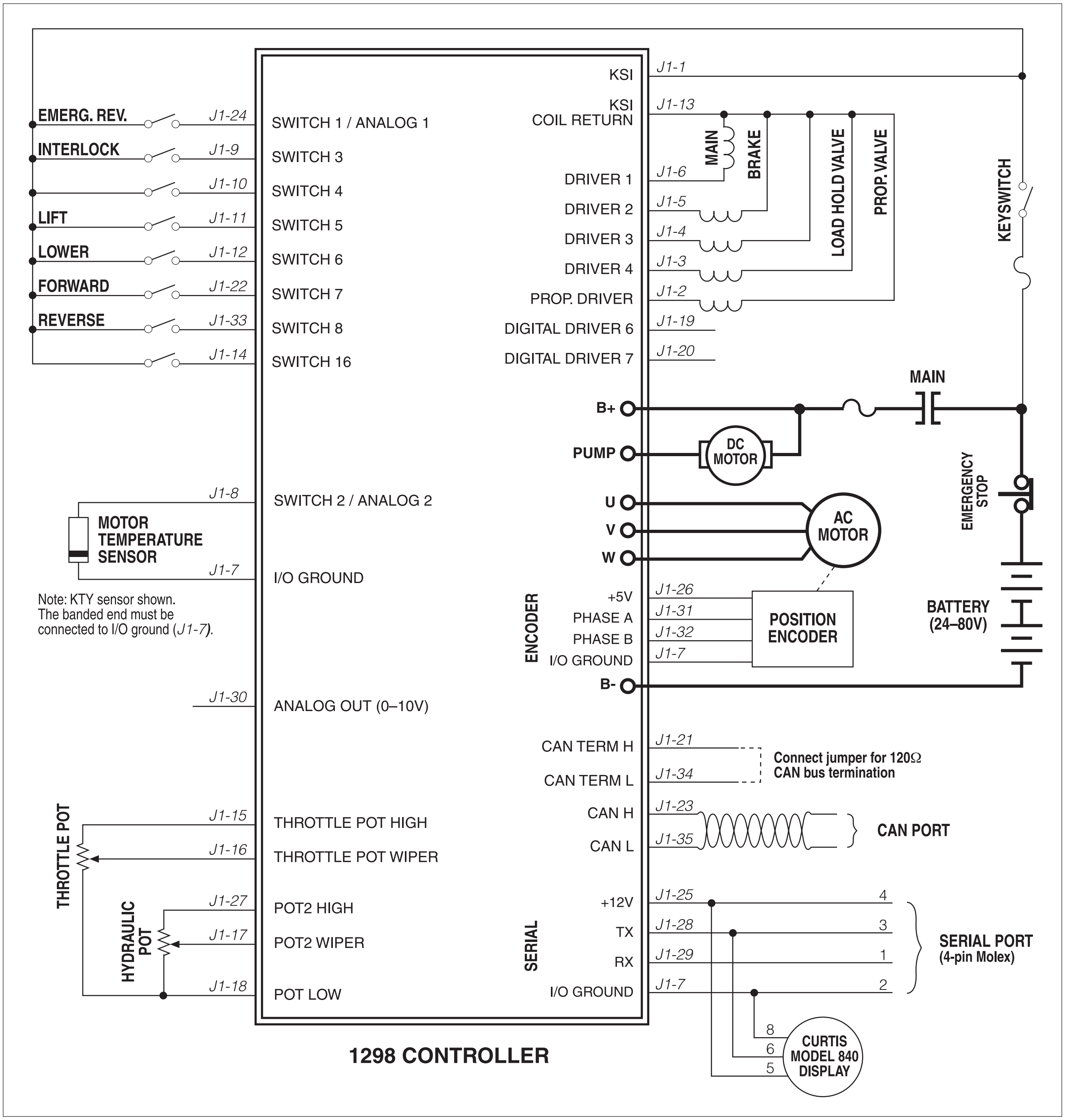


Fig. 3 *Basic wiring diagram, Curtis 1298 motor controller.*

coil driver output to remove power from the controller and motor in the event of various other faults. **If the main contactor coil is not wired to Pin 6 of the 35-pin connector as shown, 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.**

Note that the basic wiring diagram is designed for generic applications and may not fully meet the requirements of your system. These controllers have very flexible I/O and wiring configurations; you may wish to contact your local Curtis representative to discuss your particular application.

SWITCH INPUT WIRING

The following inputs are dedicated to specific functions when the parameter settings are as shown:

Switch 1:	Emergency Reverse input if the EMR Enable = On and EMR Type = 0 (see page 59).
Switch 3:	Interlock input if Interlock Type = 0 (see page 44).
Switch 5:	Lift input if Lift Switch Only Enable = On *or* Off (see page 51).
Switch 6:	Lower input if Lower Switch Only Enable = On *or* Off (see page 51).
Switch 7:	Forward input if Throttle Type = 1–3 (see page 39).
Switch 8:	Reverse input if Throttle Type = 1–3 (see page 39).

THROTTLE WIRING

Various throttles can be used with the 1298 controller. They are characterized as one of five types in the programming menu of the 1313 programmer.

Type 1:	2-wire 5kΩ–0 potentiometers
Type 2:	single-ended 0–5V throttles, current source throttles, 3-wire potentiometers, and electronic throttles
Type 3:	2-wire 0–5kΩ potentiometers
Type 4:	wigwag 0–5V throttles and 3-wire potentiometers
Type 5:	VCL input (VCL_Throttle or VCL_Hyd_Throttle)

The two throttle inputs (drive throttle and hydraulic throttle) are programmed independently.

For potentiometers, the controller provides complete throttle fault protection that meets all applicable EEC regulations. For voltage throttles, the controller protects against out-of-range wiper values, but does not detect wiring faults; it is therefore the responsibility of the OEM to provide full throttle fault protection in vehicles using voltage throttles.

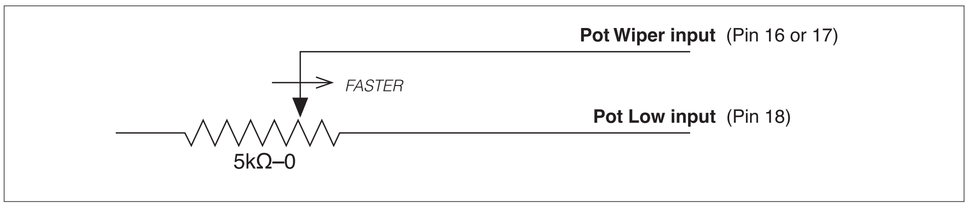
Throttle types 1–3 use the forward and reverse inputs (switches 7 and 8) in addition to the throttle pot input to define the throttle command (see Figure 17, page 93). Throttle types 4 and 5 do not use the forward and reverse inputs (or the Lift and Lower inputs, switches 5 and 6).

Wiring for the most common throttles is described in the following text and shown in the accompanying illustrations. If a throttle you are planning to use is not covered, contact the Curtis office nearest you.

Throttle Type 1

For these 2-wire resistive potentiometers, shown in Figure 4, full throttle request corresponds to 0 Ω measured between the pot wiper pin and the Pot Low pin.

Fig. 4 *Wiring for Type 1 throttles.*



Broken wire protection is provided by the controller sensing the current flow from the pot wiper input (pin 16 or 17) through the potentiometer and into Pot Low (pin 18). If the Pot Low input current falls below 0.65 mA, a throttle fault is generated and the throttle request is zeroed. Note: Pot Low (pin 18) must not be tied to ground (B-).

Throttle Type 2

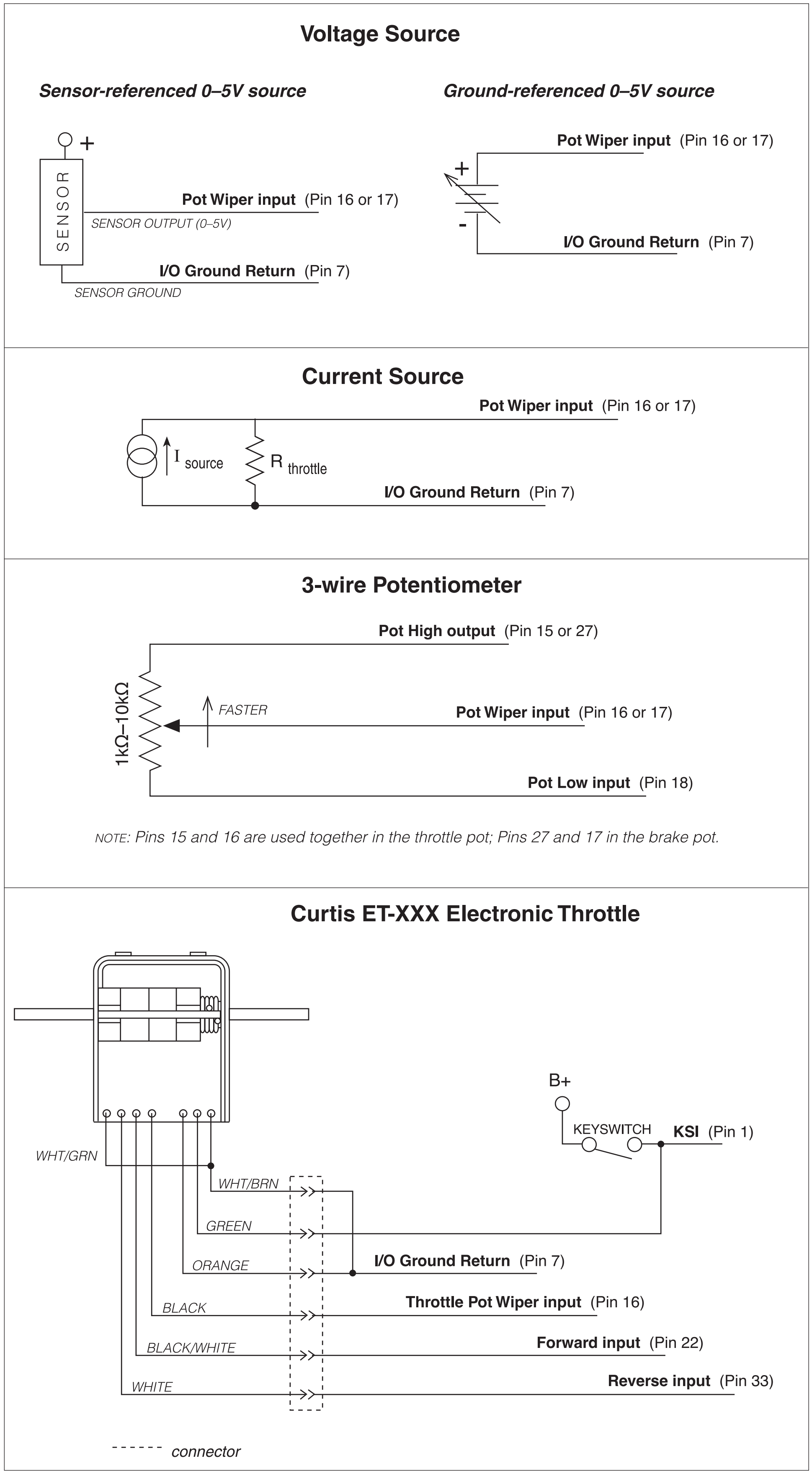
With these throttles, the controller looks for a voltage signal at the wiper input. Zero throttle request corresponds to 0 V and full throttle request to 5 V. A variety of devices can be used with this throttle input type, including voltage sources, current sources, 3-wire pots, and electronic throttles. The wiring for each is slightly different, as shown in Figure 5, and they have varying levels of throttle fault protection.

When a voltage source is used as a throttle, it is the responsibility of the OEM to provide appropriate throttle fault detection. For ground-referenced 0–5V throttles, the controller will detect open breaks in the wiper input but cannot provide full throttle fault protection.

To use a current source as a throttle, a resistor must be added to the circuit to convert the current source value to a voltage; the resistor should be sized to provide a 0–5V signal variation over the full current range. It is the responsibility of the OEM to provide appropriate throttle fault detection.

When a 3-wire potentiometer is used, the controller provides full fault protection in accordance with EEC requirements. The pot is used in its voltage divider mode, with the controller providing the voltage source and return. Pot High provides a current limited 5V source to the pot, and Pot Low provides

Fig. 5 *Wiring for Type 2 throttles.*



the return path. This is the throttle shown in the basic wiring diagram (Figure 3) for the drive throttle and for the hydraulic throttle.

The ET-XXX electronic throttle is typically used only as a drive throttle. The ET-XXX contains no built-in fault detection, and the controller will detect only open wiper faults. It is the responsibility of the OEM to provide any additional throttle fault detection necessary.

Throttle Type 3

For these 2-wire resistive potentiometers, shown in Figure 6, full throttle request corresponds to 5 kΩ measured between the pot wiper pin and the Pot Low pin.

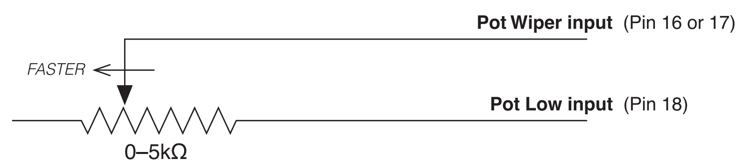


Fig. 6 *Wiring for Type 3 throttles.*

Broken wire protection is provided by the controller sensing the current flow from the wiper input (pin 16 or 17) through the potentiometer and into Pot Low (pin 18). If the Pot Low input current falls below 0.65 mA, a throttle fault is generated and the throttle request is zeroed. Note: Pot Low (pin 18) must not be tied to ground (B-).

Throttle Type 4

Type 4 throttles operate in wigwag style. No signals to the controller's forward and reverse inputs (or Lift and Lower inputs) are required; the direction is determined by the wiper input value. Only 0–5V voltage sources and 3-wire potentiometers can be used as Type 4 throttles. The controller interface for Type 4 throttles is the same as for Type 2 throttles; see Figure 5. The neutral point will be with the wiper at 2.5 V, measured between pot wiper input (pin 16) and I/O ground return (pin 7). The controller will provide increasing forward (Lift) speed as the wiper input value is increased, and increasing reverse (Lower) speed as the wiper input value is decreased.

When a 3-wire pot is used, the controller provides full fault protection. When a voltage throttle is used, the controller will detect open breaks in the wiper input but cannot provide full throttle fault protection.

Throttle Type 5

Throttle Type 5 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; see Figures 16 and 18. This throttle type can be used for either the drive throttle or the hydraulic throttle by using the VCL variables VCL_Throttle and VCL_Hyd_Throttle. How the VCL program is written will determine the source of the throttle signal, making this a very flexible throttle input method. VCL can be written to use the throttle pot inputs, switch inputs, or CAN communication messages as the source of the throttle signals. If you have questions regarding this throttle type, contact the Curtis office nearest you.

Setting the Throttle Type to Type 5 also allows the throttle inputs to be redefined by a VCL program for uses other than as throttle inputs. The variable names that VCL can use to interface with these two inputs are Throttle_Pot_Output (see page 94) for the drive throttle, and Brake_Pot_Output (see page 99) for the hydraulic throttle.

INPUT/OUTPUT SIGNAL SPECIFICATIONS

The input/output signals wired to the 35-pin connector can be grouped by type as follows; their electrical characteristics are discussed below.

— digital inputs
— high power outputs
— analog inputs
— analog output
— power supply outputs
— KSI and coil return inputs
— throttle and brake inputs
— communications port inputs/outputs
— encoder inputs.

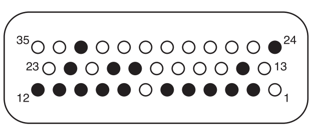
Digital inputs

These control lines can be used as digital (on/off) inputs. Normal "on" connection is direct to B+; "off" is direct to B-. Input will pull low (off) if no connection is made. All digital inputs are protected against shorts to B+ or B-.

Nine of these lines (Switches 1–8, 16) are designed to pull current to keep switch contacts clean and prevent leakage paths from causing false signals.

The remaining lines are digital inputs associated with driver outputs; note that they have much higher input impedances. The two digital output lines can also be read as inputs, and are therefore included in this group.

The lines at pins 24 and 8 can also be used as analog inputs, and are included in that group as well.



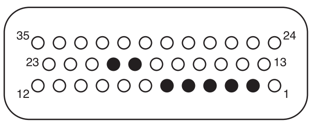
DIGITAL INPUT SPECIFICATIONS					
SIGNAL NAME	PIN	LOGIC THRESHOLDS	INPUT IMPEDANCE	VOLTAGE RANGE*	ESD TOLERANCE
Switch 1	24	Rising edge= 4.4 V max Falling edge= 1.5 V min	*24V models:* about 7.1 kΩ	-10 V to (MaxV + 10 V)	±8 kV (air discharge)
Switch 2	8				
Switch 3	9				
Switch 4	10				
Switch 5	11				
Switch 6	12				
Switch 7	22				
Switch 8	33				
Switch 16	14				
Digital Out 6	19	Rising edge= 4.4 V max Falling edge= 1.5 V min	Below 10 V= 300 kΩ Above 10 V= 150 kΩ	-0.5 V to (MaxV + 10 V)	
Digital Out 7	20				
Driver 1	6				
Driver 2	5				
Driver 3	4				
Driver 4	3				
Prop Driver	2				

* *"MaxV" in this and the following tables is the controller's maximum voltage: 30 V for 24V models, and 45 V for 24–36V models.*

High power outputs

Seven control lines can be used as high power output drivers. One of these, the proportional driver, can be operated in a current control mode for driving a proportional valve or similar load. Each output can be independently turned on continuously (low level) or pulse width modulated to set the average output voltage. These outputs are intended to drive inductive loads such as contactors and electromagnetic brakes but could also be used to drive resistive loads if peak current ratings are not exceeded. All these outputs are protected against shorts to B+ or B-. All high power inductive loads should be connected to the coil return (pin 13), which provides flyback diode protection.

These lines can also be used as digital inputs, and are included in that group as well.



HIGH POWER OUTPUT SPECIFICATIONS

SIGNAL NAME	PIN	PWM	PV CURRENT	FREQ-UENCY	OUTPUT CURRENT*	PROTECTED VOLTAGE	ESD TOLERANCE
Driver 1	6	0 to 100% duty cycle	n/a	200 Hz	2 A max	-0.5 V to keyswitch voltage	±8 kV (air discharge)
Driver 2	5				3 A max		
Driver 3	4				2 A max		
Driver 4	3						
Prop Driver	2		0 to 2 A in 607 nominal steps	18 kHz			
Digital Out 6	19	on/off	n/a	n/a	2 A max		
Digital Out 7	20						

* The combined current supplied by all seven output drivers should not exceed 10 A.

To protect the driver hardware from overcurrent, the software puts a limitation on driver output PWM. When in voltage control (typically using parameters to control the main contactor or EM brake, or using the VCL function Put_PWM) a PWM output of 1–59% is not allowed and the software will "round up" the PWM to 60%. For example, the statement Put_PWM(PWM3, 16383) would result in an output PWM of 60% (even though the argument 16383 would normally be expected to result in an output PWM of 50%).

This limitation on the output PWM also holds true for the main contactor driver PWM output that is set when the parameter Main Contactor Enable = On. With Main Contactor Enable = On, the Pull In Voltage and Holding Voltage parameters set the PWM output percent, but values of 1–59% will be rounded up to 60%.

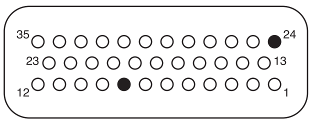
When the parameter PD Enable = On, the PD driver is current controlled and thus is not affected by the overcurrent limitations which means a full range of 0–100% PWM is allowed on this PD driver output.

Analog inputs

Two control lines can be used as analog inputs. Both inputs are protected against shorts to B+ or B-.

Typically Analog 2 is used as the input for the motor temperature sensor. This input provides a constant current appropriate for a thermistor sensor. Some standard predefined motor temperature sensors are supported in software (see Sensor Type parameter, page 49). Note: The industry standard KTY temperature sensors are silicon temperature sensors with a polarity band; **the polarity band of a KTY sensor must be the end connected to I/O Ground** (pin 7).

These lines can also be used as digital inputs, and are included in that group as well.

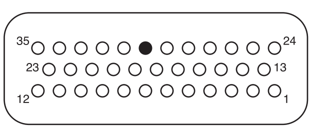


ANALOG INPUT SPECIFICATIONS

SIGNAL NAME	PIN	OPERATING VOLTAGE	INPUT IMPEDANCE	PROTECTED VOLTAGE	ESD TOLERANCE
Analog 1	24	0 to 10 V in 1024 steps	*24-36V models:* about 7.1 kΩ *36-48V models:* about 11.0 kΩ *48-80V models:* about 26.0 kΩ	-10 V to (MaxV + 10 V)	±8 kV (air discharge)
Analog 2	8				

Analog output

A single line is available as a low power analog output and is intended to drive instrumentation such as a battery discharge indicator. This output is generated from a filtered PWM signal and has about 1% ripple. The 2% settling time is <25ms for a 0–5V step and <30 ms for a 0–10V step. This output line is protected against shorts to B+ or B-.

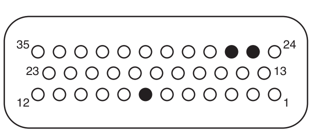


ANALOG OUTPUT SPECIFICATIONS

SIGNAL NAME	PIN	OUTPUT VOLTAGE	OUTPUT IMPEDANCE	PROTECTED VOLTAGE	ESD TOLERANCE
Analog Out	30	0 to 10 V	Source: 100 Ω Sink: 66 kΩ	-1 V to (MaxV + 10 V)	±8 kV (air discharge)

Power supply outputs

Two lines provide auxiliary output power for low power circuits such as electronic throttles, LED indicators, displays, position encoder, and remote I/O boards. I/O Ground (at pin 7) is the return line for these low power devices. Both power supply outputs are protected against shorts to B+ or B-.



POWER SUPPLY OUTPUT SPECIFICATIONS

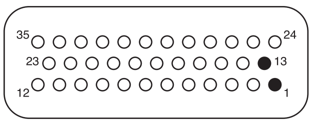
SIGNAL NAME	PIN	OUTPUT VOLTAGE	OUTPUT CURRENT	PROTECTED VOLTAGE	ESD TOLERANCE
+12V Out	25	11.5 to 14.5 V	200 mA max (combined total)	-1 V to (MaxV + 10 V)	±8 kV (air discharge)
+5V Out	26	5 V ±5%			
I/O Ground	7	n/a	500 mA max	not protected	

KSI and coil return

KSI input provides power for all low power control circuits, power capacitor precharge (before main contactor turn on), power supply outputs, and high power output drivers. Battery voltage is sensed on the input for the VCL battery discharge function.

Coil Return should be wired to the positive battery side of the contactors being driven so that switching noise associated with PWM operation of the contactors is localized to the contactor wiring only.

It is important to maintain the division between KSI and coil return in order to ensure reverse polarity protection (vehicle wiring correct, battery terminals reversed).



KSI AND COIL RETURN INPUT SPECIFICATIONS					
SIGNAL NAME	PIN	OPERATING VOLTAGE	INPUT CURRENT	PROTECTED VOLTAGE	ESD TOLERANCE
KSI	1	Between under- and overvoltage cutbacks	1.0 A max * continuous	±(MaxV+10 V)	±8 kV (air discharge)
Coil Return	13		12 A max **	(KSI-0.3 V) to (MaxV+10 V)	

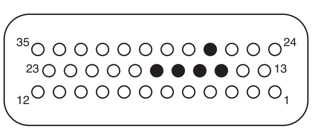
* Additionally must carry the current supplied to the driver loads by the coil return (pin 13).

**The combined current supplied by all seven output drivers should not exceed 10 A.

Throttle inputs

The two pot inputs are independently programmable to allow use of a voltage throttle or a 2-wire or 3-wire resistance throttle. Voltage throttles require only the Pot Wiper input (with I/O Ground for the return line). Resistance throttles require Pot Wiper and Pot Low (2-wire) or Pot High, Pot Wiper, and Pot Low (3-wire). All throttle I/O is protected against shorts to B+ or B-.

Alternatively, these two inputs can be used for analog signals other than the throttle pot inputs. Configuring the inputs for use with other signals requires VCL programming; see Section 7.



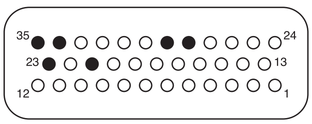
THROTTLE INPUT SPECIFICATIONS						
SIGNAL NAME	PIN	OPERATING VOLTAGE	INPUT IMPEDANCE	S/SINK CURRENT	PROTECTED VOLTAGE	ESD TOLERANCE
Throttle Pot High	15	0 V (shorted to Pot Low) 5 V (open circuit)	n/a	7 mA nominal (source)	-50 V to (MaxV+10 V)	±8 kV (air discharge)
Pot2 High	27					
Throttle Pot Wiper	16	0 to 6.25 V	290 kΩ (voltage and 3-wire)	0.76 mA nominal (source, 2-wire)		
Pot2 Wiper	17					
Pot Low	18	0 to 10 V	20 Ω nom.	Faults if above 11 mA (sink)	-1 V to (MaxV+10 V)	

Communications ports

Separate CAN and serial ports provide complete communications and programming capability for all user available controller information.

The Curtis 1313 handheld programmer plugs into a connector wired to pins 28 and 29, along with ground (pin 7) and the +12V power supply (pin 25); see wiring diagram, Figure 3. The Curtis Model 840 display can plug into the same 4-pin connector.

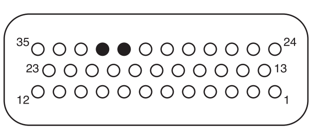
Wiring the CAN Term H and CAN Term L pins together provides a local CAN termination of 120 Ω, 0.5 W; keep the length of these wires short. CAN Term H and CAN Term L should never be connected to any external wiring.



COMMUNICATIONS PORT SPECIFICATIONS					
SIGNAL NAME	PIN	SUPPORTED PROTOCOL/DEVICES	DATA RATE	PROTECTED VOLTAGE	ESD TOLERANCE
CANH	23	CANopen, NODES 2.0, other 11-bit identifier field CAN protocols	up to 500 kbps	-5 V to (MaxV + 10 V) with <30 V differentially	±8 kV (air discharge)
CANL	35				
CAN Term H	21			(no connection to external wiring)	±8 kV (air discharge)
CAN Term L	34				
Serial TX	28	Curtis 840 Display, 1313 Handheld Programmer, 1314 PC Programming Station	as required, 9.6 to 56 kbps	-0.3 to 12 V	±8 kV (air discharge)
Serial RX	29				

Encoder inputs

Two control lines are internally configured to read a quadrature type position encoder. The encoder is typically powered from the 5V supply (pin 26) or 12V supply (pin 25), but can be powered from any external supply (from 5 V up to B+) as long as the logic threshold and differential voltage requirements are met.



ENCODER INPUT SPECIFICATIONS						
SIGNAL NAME	PIN	LOGIC THRESHOLDS	INPUT IMPEDANCE	MAX FREQ.	PROTECTED VOLTAGE	ESD TOLERANCE
Encoder A	31	Rising edge= 2.8 V max Falling edge= 2.2 V min	720 Ω (internal pull-up to +4V)	10 kHz	-5 V to (MaxV + 10 V) with <15 V differentially	±8 kV (air discharge)
Encoder B	32					

3

PROGRAMMABLE PARAMETERS

These controllers have a number of parameters that can be programmed using a Curtis 1313 handheld programmer or 1314 Programming Station. The programmable parameters allow the vehicle's performance to be customized to fit the needs of specific applications.

PROGRAMMING MENUS

The programmable parameters are grouped into nested hierarchical menus, as shown in Table 3.

Motor response tuning

Motor response characteristics can be tuned through Speed Mode or through Speed Mode Express, which is a simplified version of Speed Mode with a reduced set of parameters that is adequate for most applications. Use the Control Mode Select parameter (page 25) to select which tuning mode you will use:

- Speed Mode Express
- Speed Mode.

A third mode, Torque Mode, is also available, but it is unlikely that you will want to use that with the 1298 controller.

We strongly urge you to read Section 5, Initial Setup, before adjusting any of the parameters.

CAUTION

Even if you opt to leave most of the parameters at their default settings, **it is imperative that you perform the procedures outlined in Section 5, which set up the basic system characteristics for your application**.

Table 3 Programmable Parameter Menus: 1313 Programmer

CONTROL MODE SELECT .... p. 25

0 - SPEED MODE EXPRESS .... p. 26
- Max Speed
- Kp
- Ki
- Accel Rate
- Decel Rate
- Brake Rate

1 - SPEED MODE MENU
- **Speed Controller** .... p. 27
  - Max Speed
  - Kp
  - Ki
  - **Vel Feedforward** .... p. 27
    - Kvff
    - Build Rate
    - Release Rate
  - **Acc Feedforward** .... p. 28
    - Kaff
    - Kbff
    - Build Rate
    - Release Rate
- **Response** .... p. 29
  - Full Accel Rate HS
  - Full Accel Rate LS
  - Low Accel Rate
  - Neutral Decel Rate HS
  - Neutral Decel Rate LS
  - Full Brake Rate HS
  - Full Brake Rate LS
  - Low Brake Rate
  - **Fine Tuning** .... p. 30
    - Partial Decel Rate
    - HS (High Speed)
    - LS (Low Speed)
    - Reversal Soften
    - Max Speed Accel
    - Max Speed Decel

2 - TORQUE MODE MENU
- **Speed Limiter** .... p. 31
  - Max Speed
  - Kp
  - Ki
  - Kd
- **Response** .... p. 32
  - Accel Rate
  - Accel Release Rate
  - Brake Rate
  - Brake Release Rate
  - Neutral Braking
  - Neutral Taper Speed
  - **Fine Tuning** .... p. 33
    - Creep Torque
    - Brake Full Creep Cancel
    - Creep Build Rate
    - Creep Release Rate
    - Gear Soften
    - Brake Taper Speed
    - Reversal Soften
    - Max Speed Decel

RESTRAINT MENU .... p. 35
- Restraint Forward
- Restraint Back
- Soft Stop Speed
- **Position Hold** .... p. 35
  - Position Hold Enable
  - Kp
  - Kp Deadband (Motor Deg)
  - Kd
  - Set Speed Settling Time
  - Set Speed Threshold
  - Entry Rate
  - Exit Rollback Reduction

CURRENT LIMITS MENU .... p. 36
- Drive Current Limit
- Regen Current Limit
- Brake Current Limit
- EMR Current Limit
- Interlock Brake Current Limit
- DC Pump Current Limit
- **Power Limiting Map** .... p. 37
  - PL Nominal Speed
  - Delta Speed
  - **Drive Limiting Map** ..... p. 37
    - Nominal
    - Plus Delta
    - Plus 2xDelta
    - Plus 4xDelta
    - Plus 8xDelta
  - **Regen Limiting Map** .... p. 38
    - Nominal
    - Plus Delta
    - Plus 2xDelta
    - Plus 4xDelta
    - Plus 8xDelta

THROTTLE MENU .... p. 39
- Throttle Type
- Forward Deadband
- Forward Map
- Forward Max
- Forward Offset
- Reverse Deadband
- Reverse Map
- Reverse Max
- Reverse Offset
- HPD/SRO Type
- Sequencing Delay
- VCL Throttle Enable

BRAKE MENU *

* *The Brake menu is not typically applicable to 1298 controllers. If you will be using these parameters in your application, refer to the 1234/36/38 manual.*

Table 3 Programmable Parameter Menus: 1313 Programmer, cont'd

Individual parameters are presented as follows in the menu charts:

Parameter name as it appears in the programmer display ⇓	Allowable range in the programmer's units ⇓	Description of the parameter's function and, where applicable, suggestions for setting it ⇓
Max Speed *Max_Speed_SpdM*	100–8000 rpm *100–8000*	Defines the maximum allowed motor rpm at full throttle.
⇑ *Parameter name in VCL*	⇑ *Allowable range in VCL units*	

Note: All bit variables have two VCL parameter names. The first is the name of the bit, and the second is the name of the byte containing the bit. The bit position within the byte is indicated in brackets after the byte name.

Examples:

BIT NAME:	*Metric_Units*
BYTE NAME:	*OptionBits3* [Bit 5]
BIT NAME:	*EMR_Dir_Interlock*
BYTE NAME:	*EMR_Dir_Interlock_Bit0* [Bit 0]

In the second example, "*_Bit0*" is part of the byte name, and does not indicate the bit position; this byte, like all bytes, has 8 available bits.

Within the menu charts, each pair of bit variable names is shown as a grouped set, with the bit name appearing first and then the byte name:

Metric Units	On/Off
Metric_Units	*On/Off*
OptionBits3 [Bit 5]	

CONTROL MODE SELECT		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Control Mode Select *Control_Mode_Select*	0–2 *0–2*	This parameter determines which control method will be in effect when programming AC traction motor response: 0 = SPEED MODE EXPRESS 1 = SPEED MODE 2 = TORQUE MODE. Contact Curtis if you are interested in a custom control method. 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.

Note: Motor Speed Constraints

The maximum motor speed of the AC traction motor is a programmable parameter in each control mode. Regardless of which control mode is used, the maximum motor speed the controller will allow is constrained by the number of motor poles, the number of encoder pulses per motor revolution, and the maximum speed constraint imposed by the firmware.

Electrical frequency constraint

The maximum electrical frequency the controller will output is 300 Hz.
To determine how fast this constraint will allow your motor to spin, use the equation

Max Motor RPM = 36000 / Number of Motor Poles

(e.g., a 6-pole motor can run up to 6000 rpm).

Encoder pulses/revolution constraint

The maximum encoder frequency the controller will accept is 10 kHz.
To determine how fast this constraint will allow your motor to spin, use the equation

Max Motor RPM = 600000 / Encoder Size

(e.g., a motor with a 128-pulse encoder can run up to 4687 rpm).

Firmware max speed constraint

The maximum motor speed the controller will allow is 8000 rpm.

☞ **The overall maximum motor speed allowed is the least of these three constraints.**

0 – SPEED MODE EXPRESS **SPEED MODE EXPRESS MENU**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Max Speed *Max_Speed_SpdMx*	100–8000 rpm *100–8000*	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 constraints on page 25.
Kp *Kp_SpdMx*	0–100 % *0–8192*	Determines how aggressively the speed controller attempts to match the speed of the motor to the commanded speed. Larger values provide tighter control. If the gain is set too high, you may experience oscillations as the controller tries to control speed. If it is set too low, the motor may behave sluggishly and be difficult to control.
Ki *Ki_SpdMx*	5–100 % *50–1000*	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, you may experience oscillations as the controller tries to control 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*	0.1–30.0 sec. *100–30000*	Sets the rate (in seconds) at which the speed command increases when throttle is applied. Larger values represent slower response.
Decel Rate *Decel_Rate_SpdMx*	0.1–30.0 sec. *100–30000*	Sets the rate (in seconds) that is used to slow down the vehicle when the throttle is reduced. Larger values represent slower response.
Brake Rate *Brake_Rate_SpdMx*	0.1–30.0 sec. *100–30000*	Sets the rate (in seconds) at which the vehicle slows down when throttle is applied in the opposite direction. Larger values represent slower response.
Pump Enable *AC_Pump_Enable_SpdM* *AC_Pump_Enable_SpdM_Bit0* [Bit 0]	On/Off *On/Off*	This parameter should be programmed On to operate a pump motor rather than a vehicle drive motor. Speed controller responsiveness and stability are enhanced, and the motor is allowed to turn only in the forward direction.

1 – SPEED MODE **SPEED CONTROLLER MENU**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Max Speed *Max_Speed_SpdM*	100–8000 rpm *100–8000*	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. If Max_Speed_SpdM is set <100 rpm (through VCL or CAN), the throttle request is zeroed. Note: The maximum motor rpm is subject to the constraints on page 25.
Kp *Kp_SpdM*	0–100 % *0–8192*	Determines how aggressively the speed controller attempts to match the speed of the motor to the commanded speed. Larger values provide tighter control. If the gain is set too high, you may experience oscillations as the controller tries to control speed. If it is set too low, the motor may behave sluggishly and be difficult to control.
Ki *Ki_SpdM*	5–100 % *50–1000*	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, you may experience oscillations as the controller tries to control speed. If it is set too low, the motor may take a long time to approach the exact commanded speed.

1 – SPEED MODE **VELOCITY FEEDFORWARD MENU** *[OPTIONAL]*

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Kvff *Kvff_SpdM*	0–500 A *0–5000*	This velocity feedforward term is designed to improve throttle responsiveness and speed controller performance, especially at low speeds. 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 can be significantly 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*	0.1–5.0 sec *100–5000*	Determines how fast the Kvff term builds up. For traction systems, if you feel or hear the mechanical slop 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 overshoot if too much feedforward has been commanded.
Release Rate *Vel_FF_Release_Rate_SpdM*	0.1–2.0 sec *100–2000*	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.

1 – SPEED MODE **ACCELERATION FEEDFORWARD MENU** *[OPTIONAL]*

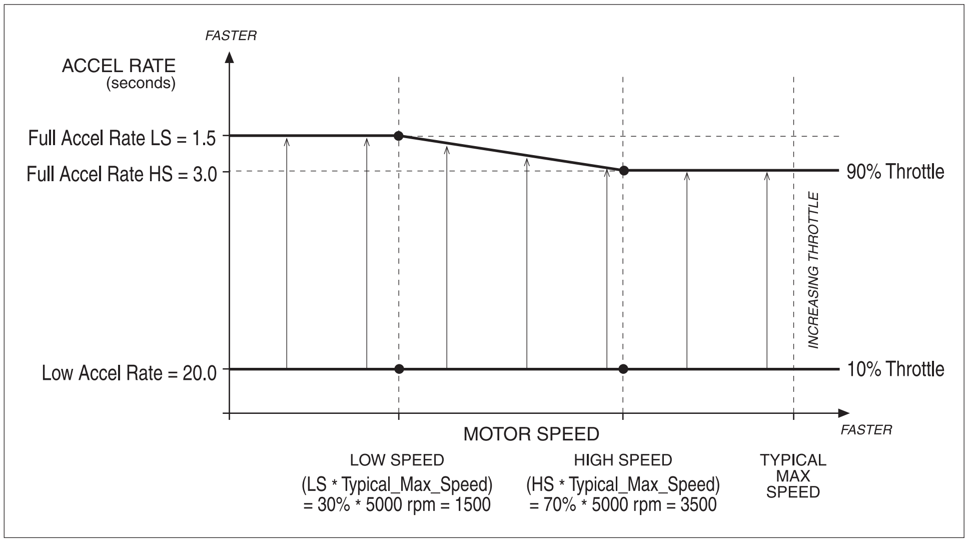
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Kaff *Kaff_SpdM*	0–500 A *0–5000*	This acceleration feedforward term is designed to improve throttle responsiveness and speed controller performance at all speeds. It can be thought of as a “quick start” function which can enhance responsiveness at all speeds. Using your present accel and decel rates, observe the average current you are running at full throttle at low speeds while accelerating without load on flat ground, and set Kaff to 50–70% of that value. Note: If any accel rate parameters get changed, this parameter will need to be changed also.
Kbff *Kbff_SpdM*	0–500 A *0–5000*	This braking feedforward term is designed to improve braking responsiveness at all speeds. Using your present decel rates, observe the average current you are running at full throttle braking, and set Kbff to that value.
Build Rate *Acc_FF_Build_Rate_SpdM*	0.1–5.0 sec *100–5000*	Determines how fast the Kaff and Kbff terms build up. For traction systems, if you feel or hear the mechanical slop 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 overshoot if too much feedforward has been commanded.
Release Rate *Acc_FF_Release_Rate_SpdM*	0.1–2.0 sec *100–2000*	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.

1 – SPEED MODE **RESPONSE MENU**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Full Accel Rate HS *Full_Accel_Rate_HS_SpdM*	0.1–30.0 sec *100–30000*	Sets the rate (in seconds) at which the speed command increases when full throttle is applied at high vehicle speeds. Larger values represent slower response. See Figure 7 for relationship between Full Accel Rate HS, Full Accel Rate LS, and Low Accel Rate.
Full Accel Rate LS *Full_Accel_Rate_LS_SpdM*	0.1–30.0 sec *100–30000*	Sets the rate (in seconds) at which the speed command increases when full throttle is applied at low vehicle speeds.
Low Accel Rate *Low_Accel_Rate_SpdM*	0.1–30.0 sec *100–30000*	Sets the rate (in seconds) at which the speed command increases when a small amount of throttle is applied. This rate is typically adjusted to affect low speed maneuverability.
Neutral Decel Rate HS *Neutral_Decel_Rate_HS_SpdM*	0.1–30.0 sec *100–30000*	Sets the rate (in seconds) that is used to slow down the vehicle when the throttle is released to neutral at high vehicle speeds.
Neutral Decel Rate LS *Neutral_Decel_Rate_LS_SpdM*	0.1–30.0 sec *100–30000*	Sets the rate (in seconds) that is used to slow down the vehicle when the throttle is released to neutral at slow vehicle speeds.
Full Brake Rate HS *Full_Brake_Rate_HS_SpdM*	0.1–30.0 sec *100–30000*	Sets the rate (in seconds) 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 Figure 8 for relationship between Full Brake Rate HS, Full Brake Rate LS, and Low Brake Rate.
Full Brake Rate LS *Full_Brake_Rate_LS_SpdM*	0.1–30.0 sec *100–30000*	Sets the rate (in seconds) at which the vehicle slows down from low speeds when full brake is applied or when full throttle is applied in the opposite direction.
Low Brake Rate *Low_Brake_Rate_SpdM*	0.1–30.0 sec *100–30000*	Sets the rate (in seconds) at which the vehicle slows down at all speeds when a small amount of brake is applied or when a small amount of throttle is applied in the opposite direction.

Fig. 7 *Acceleration response rate diagram.*

In this example,
HS = 70%,
LS = 30%,
Typ Max Spd = 5000 rpm.

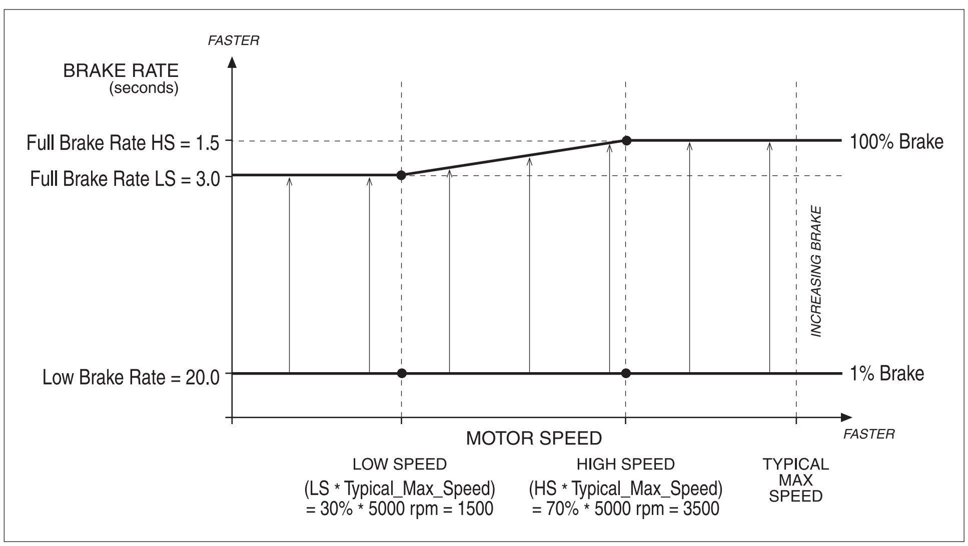


1 – SPEED MODE **FINE TUNING MENU**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Partial Decel Rate *Partial_Decel_Rate_SpdM*	0.1–30.0 sec. *100–30000*	Sets the rate (in seconds) that is used to slow down the vehicle when the throttle is reduced without being released to neutral. Larger values represent slower response.
HS (High Speed) *HS*	0–100 % *0–32767*	Sets the percentage of the Typical Max Speed (page 48) above which the "HS" parameters will be used.
LS (Low Speed) *LS*	0–100 % *0–32767*	Sets the percentage of the Typical Max Speed (page 48) below which the "LS" parameters will be used.
Reversal Soften *Reversal Soften_SpdM*	0–100 % *0–3000*	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.
Max Speed Accel *Max_Speed_Accel_SpdM*	0.1–30.0 sec *100–30000*	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 raised. The rate set by this parameter is the time to ramp from 0 rpm to Typical Max Speed rpm. For example, suppose Max Speed is raised from 1000 rpm to 4000 rpm. If 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*	0.1–30.0 sec *100–30000*	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 you change Max Speed from 4500 rpm to 2500 rpm. If 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.

Fig. 8 *Braking response rate diagram.*

In this example,
HS = 70%,
LS = 30%,
Typ Max Spd = 5000 rpm.



2 – TORQUE MODE **SPEED LIMITER MENU**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Max Speed *Max_Speed_TrqM*	500–8000 rpm *500–8000*	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; e.g., 40% applied throttle corresponds to a request for 40% of the available torque. Note: The maximum motor rpm is subject to the constraints on page 25.
Kp *Kp_TrqM*	0–100 % *0–8192*	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, you may experience oscillations as the controller tries to control speed. Setting Kp too low may result in a top speed much higher than Max Speed.
Ki *Ki_TrqM*	5–100 % *50–1000*	The integral term (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 overspeed.
Kd *Kd_TrqM*	0–100 % *0–8192*	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.

2 – TORQUE MODE **RESPONSE MENU**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Accel Rate *Accel_Rate_TrqM*	0.1–30.0 sec. *100–30000*	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*	0.1–2.0 sec. *100–2000*	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*	0.1–5.0 sec. *100–5000*	Adjusts the rate (in seconds) at which braking torque builds as the vehicle transitions from drive to braking when direction is reversed, the brake pedal is applied, or neutral braking begins. 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*	0.1–2.0 sec. *100–2000*	Adjusts the rate (in seconds) at which braking torque releases as as the vehicle transitions from braking to drive.
Neutral Braking *Neutral_Braking_TrqM*	0–100 % *0–32767*	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 (see Current Limits menu, page 36).
Neutral Taper Speed *Neutral_Taper_Speed_TrqM*	200–6000 rpm *200–6000*	Determines the motor speed below which braking current is adjusted in both the positive and negative directions when throttle is reduced; see Figure 9. In the positive direction, the neutral braking current is linearly reduced from 100% at the Neutral Taper Speed to the Creep Torque current at zero rpm motor speed. In the negative direction, the restraint current is linearly increased from the Creep Torque current at zero rpm motor speed to the restraint current at the Neutral Taper Speed. Note: Setting the taper speed too low may cause oscillations in the motor.

2 – TORQUE MODE **FINE TUNING MENU**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Creep Torque *Creep_Torque_TrqM*	0–100 % *0–32767*	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 9. ***WARNING!*** 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 (see Brake menu in 1234/36/38 manual), creep torque is progressively disabled as brake is applied so as to prevent the motor from driving into the brakes and thus wasting energy.
Brake Full Creep Cancel *Brake_Full_Creep_Cancel_TrqM*	25–100 % *8192–32767*	Determines the amount of brake pedal input that will fully cancel the . creep torque. Amount of cancellation is proportional to the brake input.
Creep Build Rate *Creep_Build_Rate_TrqM*	0.1–5.0 sec *100–5000*	Determines how fast the programmed creep torque builds when a direction is selected.
Creep Release Rate *Creep_Release_Rate_TrqM*	0.1–5.0 sec *100–5000*	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*	0–100 % *0–5000*	Adjusts the throttle take-up from linear (0% setting) to an S curve. Larger values create softer throttle take-up, in forward and reverse. Softening is progressively reduced at higher speeds; see Figure 10.
Brake Taper Speed *Brake_Taper_Speed_TrqM*	200–6000 rpm *200–6000*	Determines the motor speed below which the maximum braking current is linearly reduced from 100% to 0% at zero speed; see Figure 11. 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.
Reversal Soften *Reversal_Soften*	0–100 % *0–3000*	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.
Max Speed Decel *Max_Speed_Accel_TrqM*	0.1–30.0 sec *100–30000*	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 * (3000–1000) / 5000 = 2.0 seconds to ramp from 3000 rpm to 1000 rpm.

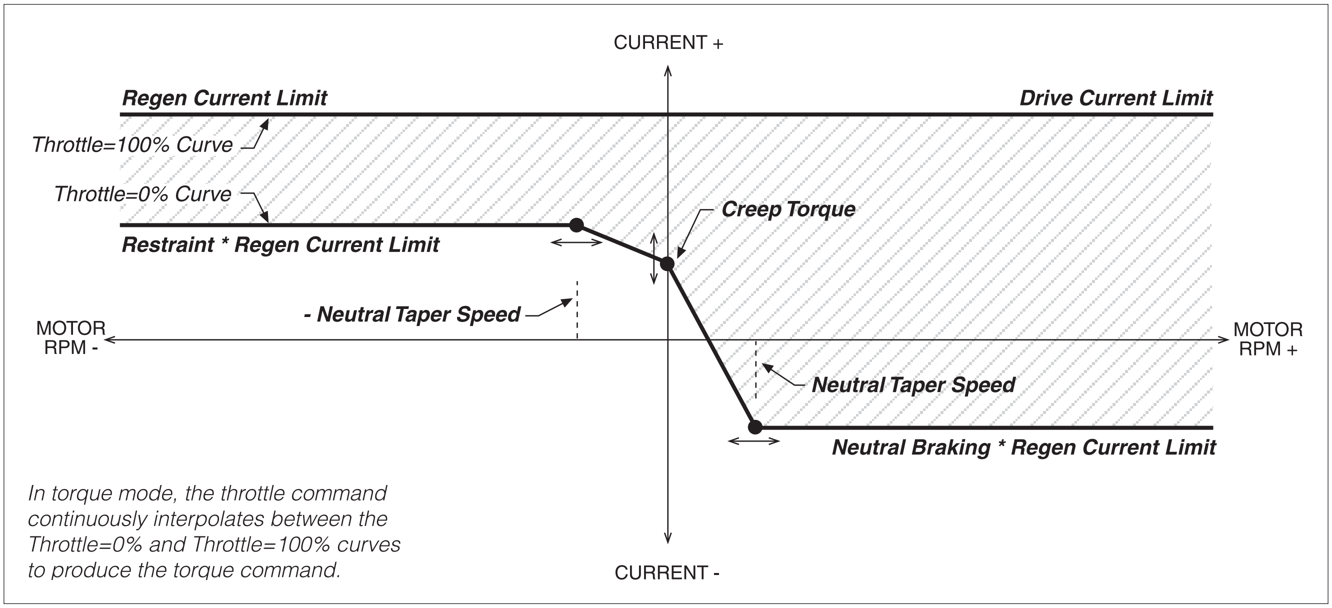


Fig. 9 *Throttle mapping (torque control mode).*

Fig. 10 *Effect of Gear Soften parameter (torque control mode).*

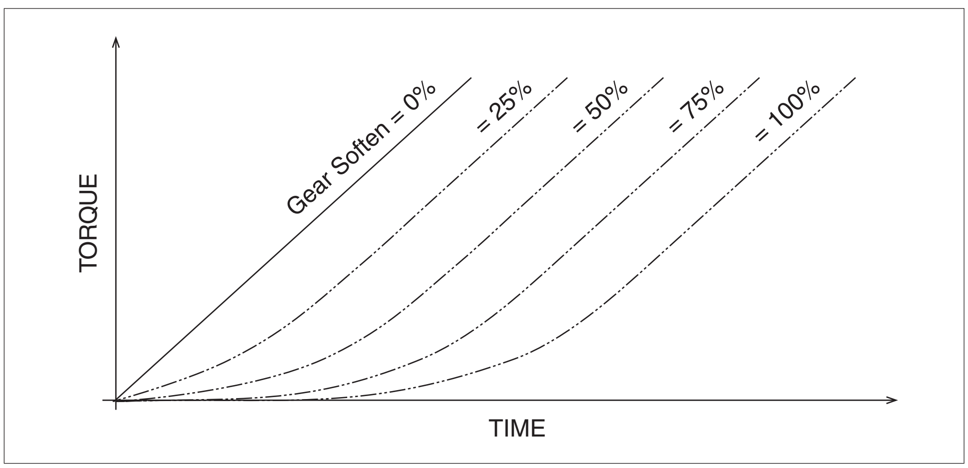
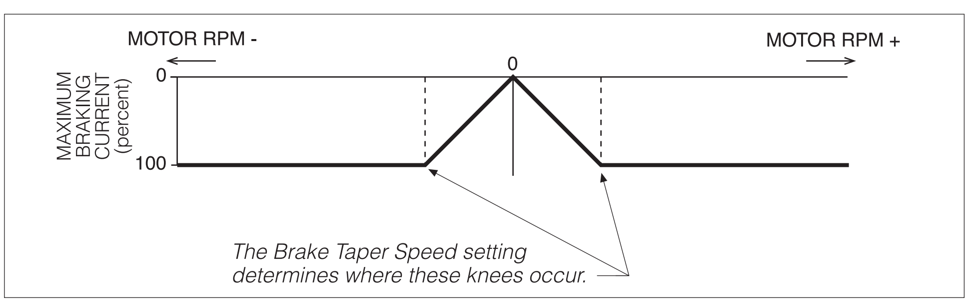


Fig. 11 *Effect of Brake Taper Speed parameter (torque control mode).*



RESTRAINT MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Restraint Forward *Restraint_Forward*	0–100 % *0–32767*	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*	0–100 % *0–32767*	Increases torque when on a steep hill in order to limit roll-back speed. Setting this parameter too high may cause oscillations in the motor as it attempts to limit the roll-back speed.
Soft Stop Speed *Soft_Stop_Speed*	0–500 rpm *0–500*	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; the vehicle may even rock back as a result of tire wind-up. Soft Stop Speed 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.” When throttle is released on a ramp, the vehicle may roll back before Position Hold (see below) takes control. Soft Speed Stop can be used to reduce the amount of rollback, but shouldn’t be set so high the vehicle drives up the ramp after the throttle is released.

POSITION HOLD MENU **[SPEED MODE & SPEED MODE EXPRESS only]**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Position Hold Enable *Position_Hold_Enable* *Position_Hold_Enable_Bit0* [Bit 0]	On/Off *On/Off*	Allows the Position Hold mode to be entered at zero throttle when the vehicle comes to a stop. Note: EM Brake Type = 2 also enables the Position Hold function.
Kp *Kp_Position_Hold*	2–100 % *82–2048*	Determines the stiffness with which position is regulated when in Position Hold mode. High Kp will produce less rollback on a ramp, but more bouncing; see Kd below. Too much Kp will cause instability.
Kp Deadband (motor degrees) *Kp_Deadband_Position_Hold*	0–720 motor degrees *0–8192*	Allows a position feedback deadband around the setpoint, to help avoid instability caused by gear slop.
Kd *Kd_Position_Hold*	0–100 % *0–8192*	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 will improve the dynamic response of the Position Hold controller, but too much Kd will cause fast instability.
Set Speed Settling Time *Set_Speed_Settling_Time*	0–5000 msec *0–156*	This parameter appears twice in the menu structure. For description, see EM Brake Control menu, page 43.
Set Speed Threshold *Set_Speed_Threshold*	5–100 rpm *5–100*	This parameter appears twice in the menu structure. For description, see EM Brake Control menu, page 43.

POSITION HOLD MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Entry Rate *Entry_Rate_Position_Hold*	5–100 % *50–1000*	When the vehicle transitions from forward speed to reverse speed or from reverse speed to forward speed (for example, when coming to a stop going 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 be brought to a 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 take longer to come to a stop and enter Position Hold mode.
Exit Rollback Reduction *Exit_Rollback_Reduction*	0–100 % *0–2048*	This function is applicable only when the Torque Preload function has been disabled (see EM Brake menu), or its timer has expired. It introduces a proportional feedforward term into the speed controller based on the position signal. For example, suppose the vehicle is on a ramp and a forward throttle request is given such that the vehicle rolls back slightly before climbing the ramp (again, assuming the torque preload function is inactive). As the vehicle rolls back a feedforward torque term proportional to the rollback position will be added to the torque request until forward speed is sensed.

CURRENT LIMITS MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Drive Current Limit *Drive_Current_Limit*	5–100 % *1638–32767*	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.* Reducing this value will reduce the maximum drive torque.
Regen Current Limit *Regen_Current_Limit*	5–100 % *1638–32767*	Sets the maximum RMS regen current, as a percentage of the controller's full rated current.* The regen current limit applies during neutral braking, direction reversal braking, and speed limiting when traveling downhill.
Brake Current Limit *Brake_Current_Limit*	5–100 % *1638–32767*	Sets the maximum RMS regen current during braking when a brake command is given, as a percentage of the controller's full rated current.* 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*	5–100 % *1638–32767*	Sets the maximum RMS current allowed for braking and drive 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*	5–100 % *1638–32767*	Sets the maximum RMS regen current during interlock braking, as a percentage of the controller's full rated current.*
DC Pump Current Limit *DC_Pump_Current_Limit*	5–100 % *1638–32767*	Sets the maximum current the controller will supply to the pump motor during lift operation, as a percentage of the controller's full rated current.*

* The full rated current depends on the controller model; see specifications in Table D-1 for the rated current of your model.

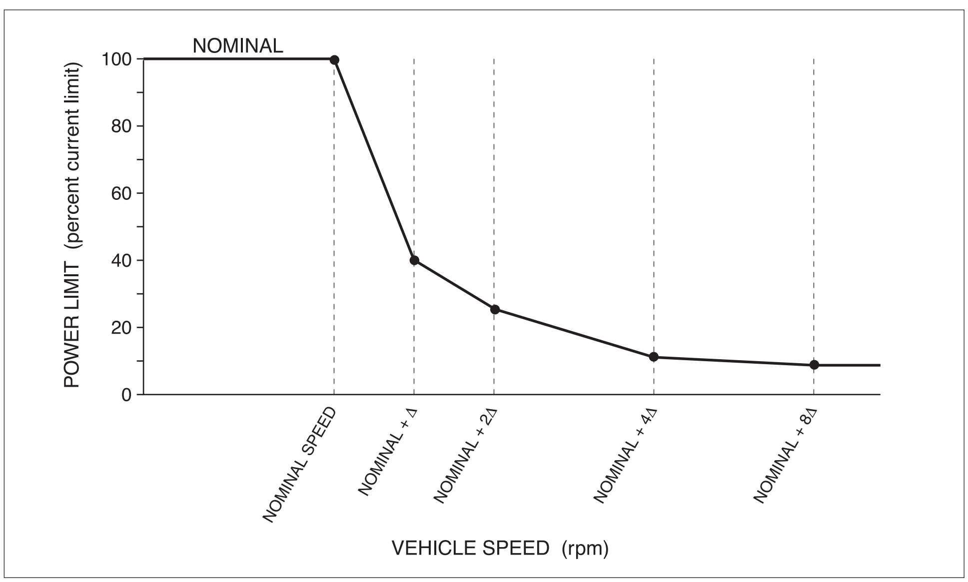
POWER LIMITING MAP MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Nominal Speed *PL_Nominal_Speed*	100–4000 rpm *100–4000*	Sets the base speed that will be used in the drive limiting map and regen limiting map.
Delta Speed *PL_Delta_Speed*	50–1000 rpm *50–1000*	Sets the width of the delta increment that will be used in the drive limiting map and regen limiting map.

DRIVE LIMITING MAP MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Nominal *PL_Drive_Nominal*	0–100 % *0–32767*	These parameters define the percentage of drive current limit that will be applied 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. It can also be used to keep consistent vehicle power with changing battery state-of-charge.
Plus Delta *PL_Drive_Nominal_Plus_Delta*	0–100 % *0–32767*	
Plus 2xDelta *PL_Drive_Nominal_Plus_2xDelta*	0–100 % *0–32767*	
Plus 4xDelta *PL_Drive_Nominal_Plus_4xDelta*	0–100 % *0–32767*	
Plus 8xDelta *PL_Drive_Nominal_Plus_8xDelta*	0–100 % *0–32767*	

Fig. 12 *Drive current limiting map (typical example).*



REGEN LIMITING MAP MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Nominal *PL_Regen_Nominal*	0–100 % *0–32767*	These parameters define the percentage of regen current limit or braking current limit that will be applied at the speeds defined by the nominal speed and delta speed parameters. The curve can be shaped to limit 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*	0–100 % *0–32767*	
Plus 2xDelta *PL_Regen_Nominal_Plus_2xDelta*	0–100 % *0–32767*	
Plus 4xDelta *PL_Regen_Nominal_Plus_4xDelta*	0–100 % *0–32767*	
Plus 8xDelta *PL_Regen_Nominal_Plus_8xDelta*	0–100 % *0–32767*	

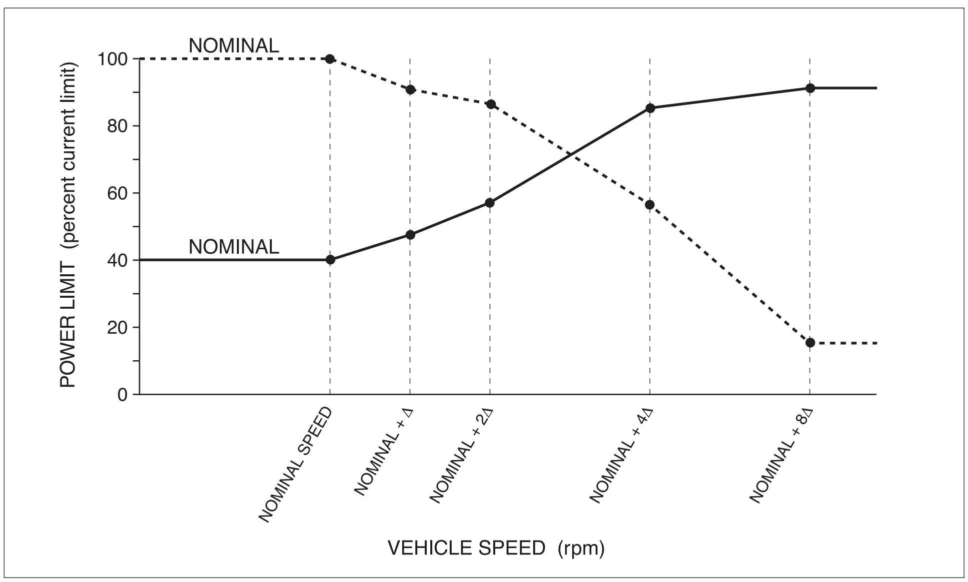


Fig. 13 *Regen current limiting map (two examples).*

THROTTLE MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Throttle Type *Throttle_Type*	1–5 *1–5*	The 1298 controller accepts a variety of throttle inputs. The throttle type parameter can be programmed as follows: 1 2-wire rheostat, 5kΩ–0 input 2 single-ended 3-wire 1kΩ–10kΩ potentiometer, or 0–5V voltage source 3 2-wire rheostat, 0–5kΩ input 4 wigwag 3-wire 1kΩ–10kΩ potentiometer, or 0–5V voltage source 5 VCL input (*VCL_Throttle*) 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.
Forward Deadband *Forward_Deadband*	0–5.00 V *0–32767*	Defines the wiper voltage at the throttle deadband threshold. Increasing the throttle deadband setting will increase the neutral range. 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.
Forward Map *Forward_Map*	0–100 % *0–32767*	Modifies the vehicle's response to the throttle input. Setting the throttle map at 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. The map value is the percentage of controller output at half throttle ((deadband + max)/2).
Forward Max *Forward_Max*	0–5.00 V *0–32767*	Defines the wiper voltage required to produce 100% controller output. Decreasing the throttle max setting reduces the wiper voltage and therefore the full stroke necessary to produce full controller output. This parameter allows reduced-range throttle assemblies to be accommodated.
Forward Offset *Forward_Offset*	0–100 % *0–32767*	Defines the initial controller output generated when the throttle is first rotated out of the neutral deadband. For most vehicles, a setting of 0 is appropriate. For heavy vehicles, however, increasing the offset may improve controllability by reducing the amount of throttle required to start the vehicle moving.

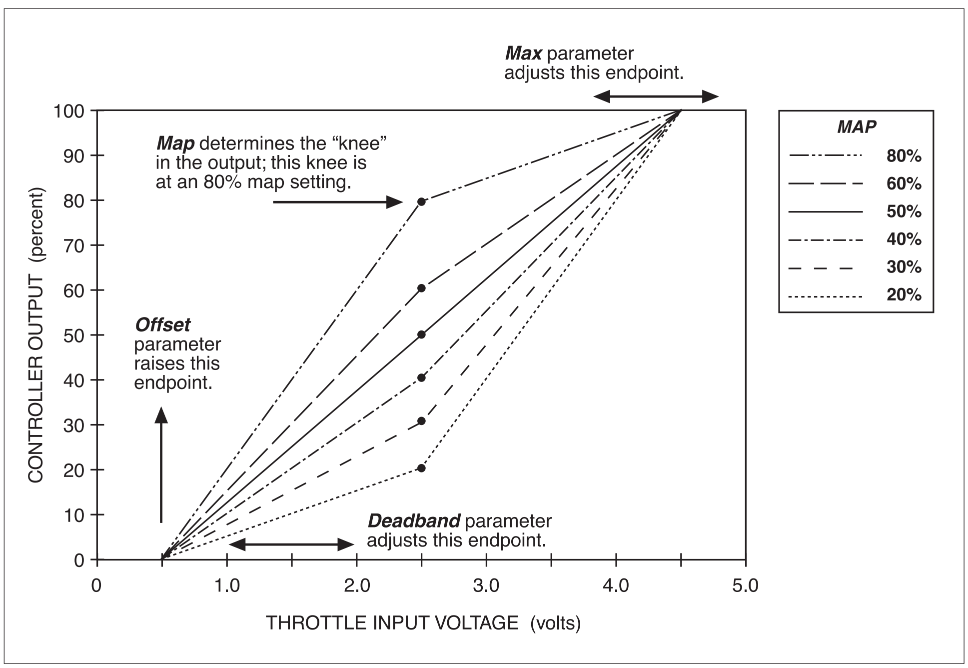
☞ Note: All four throttle adjustment parameters — Deadband, Map, Max, Offset — condition the raw throttle voltage into a single % throttle command, as shown in Figure 14.

THROTTLE MENU, cont'd		
	ALLOWABLE	
Reverse Deadband *Reverse_Deadband*	0–5.00 V *0–32767*	The four Throttle Reverse parameters are the same as their Throttle Forward counterparts, and apply when the throttle direction is reversed.
Reverse Map *Reverse_Map*	0–100 % *0–32767*	
Reverse Max *Reverse_Max*	0–5.00 V *0–32767*	
Reverse Offset *Reverse_Offset*	0–100 % *0–32767*	

Fig. 14 *Effect of throttle adjustment parameters. Together these four generic parameters determine the controller's response to throttle demand (in forward or reverse) and to brake demand.*

In the examples shown in this figure,

Deadband = 0.5V
Max = 4.5V
Offset = 0.



THROTTLE MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
HPD/SRO Type *HPD_SRO_Type* *OptionBits1* [Bit 4]	0–3 *0–3*	Determines the type of HPD/SRO protection. One type of checks is available for material-handling vehicles, and two types for golf-style vehicles. If any of the HPD/SRO checks finds an input sequencing problem, an HPD/Sequencing Fault (flash code 47) 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:* If throttle input is received before interlock or 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*	0.0–5.0 sec. *0–312*	Typically the sequencing delay feature allows the interlock switch to be cycled within a set time (the defined sequencing delay), thus preventing inadvertent activation of HPD/SRO. This feature is especially useful in applications where the interlock switch may bounce or be momentarily cycled during operation.
VCL Throttle Enable *VCL_Throttle_Enable* *VCL_Throttle_Enable_Bit0* [Bit 0]	On/Off *On/Off*	When programmed On, the throttle processing with fault detection will operate normally; however, the throttle command (see Figure 17, page 93) 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_Type 1–4 with throttle fault detection.

EM BRAKE CONTROL MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Brake Type *EM_Brake_Type*	0–2 *0–2*	The brake type parameter determines how the EM brake responds to the interlock input, throttle, and vehicle motor speed. 0 EM brake function disabled. The EM brake driver (PWM2) is released 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 EM_Brake_Set_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 EM_Brake_Set_Speed_Threshold. Position Hold will be enabled automatically.
Pull In Voltage *EM_Brake_Pull_In_Voltage*	0–100 % *0–32767*	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. To protect the driver hardware from overcurrent, the software puts a limitation on driver output PWM. A PWM output of 1–59% is not allowed and the software will “round up” the PWM to 60%. Setting this parameter to a value <60% will therefore result in an output PWM of at least 60%. Note: The Battery Voltage Compensated parameter controls whether the pull-in and holding voltages are battery voltage compensated.
Holding Voltage *EM_Brake_Holding_Voltage*	0–100 % *0–32767*	The EM brake holding voltage allows a reduced average voltage to be applied to the brake coil once the brake has been released. This parameter must be set high enough to hold the brake released under all shock and vibration conditions the vehicle will be subjected to. To protect the driver hardware from overcurrent, the software puts a limitation on driver output PWM. A PWM output of 1–59% is not allowed and the software will “round up” the PWM to 60%. Setting this parameter to a value <60% will therefore result in an output PWM of at least 60%. Note: The Battery Voltage Compensated parameter controls whether the pull-in and holding voltages are battery voltage compensated.
Battery Voltage Compensated *EM_Brake_Battery_Voltage_Compensated* *EM_Brake_Battery_Voltage_Compensated_Bit0* [Bit 0]	On/Off *On/Off*	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 set Nominal Voltage (see Battery menu, page 55). 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 Battery Voltage Compensated = Off, the output will be 30V (Actual Voltage × Holding Voltage) to the coil.

EM BRAKE CONTROL MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Set EM Brake On Fault *EM_Brake_Set_Upon_Fault* *EM_Brake_Set_Upon_ Fault_Bit0* [Bit 0]	On/Off *On/Off*	When programmed On, the controller's operating system will drop the electromagnetic brake when a fault occurs that has a fault action of ShutdownEMBrake. See Section 8 for a list of all the faults that have a fault action of ShutdownEMBrake.
Set Speed Threshold *Set_Speed_Threshold*	5–100 rpm *5–100*	Determines the speed below which the EM brake will be commanded to set. Setting this speed too high may cause a jerky stop when the EM brake sets and stops the motor.
Release Delay *EM_Brake_Release_Delay*	40–2000 msec *5–250*	Estimated time for the EM brake to physically release after the pull-in voltage is applied. This is used to ensure the position hold torque buildup is complete before the brake releases. When set too low, the vehicle may experience rollback on EM brake release.
Set Speed Settling Time *Set_Speed_Settling_Time*	0–5000 msec *0–156*	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. *Note: This parameter is applicable only when Speed Mode or Speed Mode Express is selected* ***and*** *either Position Hold Enable = On or EM Brake Type = 2.*
Torque Preload Delay *EM_Brake_Torque_Preload_ Delay*	0–800 msec *0–100*	Estimated worst-case time to build up the torque required to hold the vehicle stationary on a hill prior to EM brake release. This is used in conjunction with Release Delay to determine when to release the brake and allow the speed request to slew away from zero. *Note: This parameter is applicable only when Speed Mode or Speed Mode Express is selected* ***and*** *either Position Hold Enable = On or EM Brake Type = 2.*
Torque Preload Enable *EM_Brake_Torque_Preload_Enable* *EM_Brake_Torque_Preload_ Enable_Bit0* [Bit 0]	On/Off *On/Off*	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 throttle is re-engaged, this value is loaded in the motor before the EM brake is released. 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. On = When a valid throttle input is received, the speed controller will start with a pre-set torque as measured by position-hold when the vehicle came to a stop. *Note: This parameter is applicable only when Speed Mode or Speed Mode Express is selected* ***and*** *either Position Hold Enable = On or EM Brake Type = 2.*
Torque Preload Cancel Delay *EM_Brake_Torque_Preload_ Cancel_Delay*	0–120 sec *0–15000*	The timer starts after the EM brake is set. If the timer expires before the throttle is re-engaged, the torque preload memory will be 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 conditions above).*

MAIN CONTACTOR MENU		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Main Enable *Main_Enable* *OptionBits1* [Bit 0]	On/Off *On/Off*	When programmed On, the controller's native software controls the main contactor when the interlock is enabled or when Pump_Throttle >0. When programmed Off, the contactor is controlled by VCL. 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.
Pull In Voltage *Main_Pull_In_Voltage*	0–100 % *0–32767*	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. To protect the driver hardware from overcurrent, the software puts a limitation on driver output PWM. A PWM output of 1–59% is not allowed and the software will "round up" the PWM to 60%. Setting this parameter to a value <60% will therefore result in an output PWM of at least 60%. Note: The Battery Voltage Compensated parameter (below) controls whether the pull-in and holding voltages are battery voltage compensated.
Holding Voltage *Main_Holding_Voltage*	0–100 % *0–32767*	The main contactor holding voltage parameter allows a reduced average voltage to be applied to the contactor coil once it has closed. This parameter must be set high enough to hold the contactor closed under all shock and vibration conditions the vehicle will be subjected to. To protect the driver hardware from overcurrent, the software puts a limitation on driver output PWM. A PWM output of 1–59% is not allowed and the software will "round up" the PWM to 60%. Setting this parameter to a value <60% will therefore result in an output PWM of at least 60%. Note: The Battery Voltage Compensated parameter (below) controls whether the pull-in and holding voltages are battery voltage compensated.
Battery Voltage Compensated *Main_Driver_Battery_Voltage_Compensated* *Main_Driver_Battery_Voltage_Compensated_Bit0* [Bit 0]	On/Off *On/Off*	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 (see Battery menu, page 55). 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 Battery Voltage Compensated = Off, the output will be 30V (Actual Voltage × Holding Voltage) to the coil.
Interlock Type *Interlock_Type*	0–2 *0–2*	Three interlock options are available: 0 = interlock turns on with switch 3. 1 = interlock controlled by VCL functions. 2 = interlock turns on with KSI.

MAIN CONTACTOR MENU, cont'd		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Open Delay *Open_Delay*	0–40 sec. *0–2500*	Applicable only when Interlock Type = 0 or 1. The delay can be set to allow the contactor to remain closed for a period of time (the delay) after the interlock switch is opened. The delay is useful for preventing unnecessary cycling of the contactor and for maintaining power to auxiliary functions that may be used for a short time after the interlock switch has opened.
Checks Enable *Checks_Enable* *OptionBits1* [Bit 2]	On/Off *On/Off*	When programmed On, the controller performs ongoing checks to ensure that the main contactor has closed properly each time it is commanded to do so, and that it has not welded closed. These checks (Main Contactor Welded and Main Contactor Did Not Close) are not performed if this parameter is Off. The main contactor driver, however, is always protected from short circuits.
Main DNC Threshold *Main_DNC_Threshold*	0–84.0 V *0–5376*	When Checks Enable = On, this parameter is used as the threshold for detecting a Main Did Not Close fault. The Main DNC 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 detect. 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 V will disable the Main Did Not Close fault check.
Precharge Enable *Precharge_Enable* *OptionBits2* [Bit 6]	On/Off *On/Off*	Turns the precharge feature on and off. Precharge provides a limited current charge of the controller's internal capacitor bank before the main contactor is closed. This decreases the arcing that would otherwise occur when the contactor is closed with the capacitor bank discharged.

PROPORTIONAL DRIVER MENU		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
PD Enable *PD_Enable* *OptionBits1* [Bit 6]	On/Off *On/Off*	Determines how the PWM of the proportional driver is controlled. When programmed On, it is controlled by the controller's PD current control software. When programmed Off, it is controlled by the VCL function *Put_PWM* (PWM5, value); see Figure 20, page 100.
Hyd Lower Enable *Hyd_Lower_Enable* *OptionBits1* [Bit 7]	On/Off *On/Off*	When programmed On, lowering is controlled by throttle position. When programmed Off, lowering is controlled by the VCL variable *VCL_PD_Throttle*; see Figure 20, page 100.
PD Max Current *PD_Max_Current*	0.0–2.0 A *0–607*	* The Lower speed is determined by the aperture of the proportional valve. This parameter sets the maximum allowed current through the valve, which in turn defines its aperture.
PD Min Current *PD_Min_Current*	0.0–2.0 A *0–607*	* Sets the minimum allowed current through the proportional valve. Most proportional valves need a non-zero closed current in order to start opening immediately when Lower is requested.
PD Dither % *PD_Dither_Percent*	0–100 % *0–32767*	* Dither provides a constantly changing current in the coil to produce a rapid back-and-forth motion of the valve; this keeps the valve lubricated and allows low-friction, precise movement. The PD Dither % parameter specifies the amount of dither as a percentage of the PD max current, and is applied in a continuous cycle of add%-subtract%.
PD Dither Period *PD_Dither_Period*	16–112 msec *1–7*	* Sets the period for proportional valve dither.
PD Kp *PD_Kp*	1–100 % *82–8192*	* Sets the proportional gain of the current feedback controller. Higher gains force the control loop to respond quickly but may cause oscillations.
PD Ki *PD_Ki*	1–100 % *327–32767*	* Sets the integral gain of the current feedback controller. Integral gain tries to force the error to zero. Higher gains force the control loop to respond quickly but may cause oscillations.

* These parameter descriptions assume the proportional driver is being used to drive a proportional valve, and that the PD current control software is active (PD_Enable = On).

FAULT CHECKING MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Driver1 Checks Enable *Driver1_Checks_Enable* *OptionBits2* [Bit 1]	On/Off *On/Off*	The five Checks Enable parameters are used to enable driver and coil fault detection at the five individual drivers (at Pins J1-6, J1-5, J1-4, J1-3, and J1-2). When a Checks parameter is enabled, the associated driver, driver wiring, and driver load are checked 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 checks will detect both open and shorted conditions. When a fault is detected, the controller opens the driver and issues a fault code. If nothing is connected to a driver, its Checks Enable parameter should be set Off. Note: Short circuit protection is always active at these five drivers, regardless of how Checks Enable is set.
Driver2 Checks Enable *Driver2_Checks_Enable* *OptionBits2* [Bit 2]	On/Off *On/Off*	
Driver3 Checks Enable *Driver3_Checks_Enable* *OptionBits2* [Bit 3]	On/Off *On/Off*	
Driver4 Checks Enable *Driver4_Checks_Enable* *OptionBits2* [Bit 4]	On/Off *On/Off*	
PD Checks Enable *PD_Checks_Enable* *OptionBits2* [Bit 5]	On/Off *On/Off*	
External Supply Max *External_Supply_Max*	5–200 mA *52–800*	Sets the upper threshold of the combined current of the 5V and 12V external supplies. At or above this threshold a fault will be created that can be read by VCL.
External Supply Min *External_Supply_Min*	5–200 mA *52–800*	Sets the lower threshold of the combined current of the 5V and 12V external supplies. At or below this threshold a fault will be created that can be read by VCL.

MOTOR MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Typical Max Speed *Typical_Max_Speed*	500–8000 rpm *500–8000*	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, and Full_Accel_Rate_LS_SpdM = 3.0 seconds: If Max_Speed_SpdM = 6000 rpm, it will take 3.0 sec to accelerate from zero to top speed (6000 rpm). If Max_Speed_SpdM = 3000 rpm, it will take 1.5 sec to accelerate from zero to top speed (3000 rpm). If Max_Speed_SpdM = 1000 rpm, it will take 0.5 sec to accelerate from zero to top speed (1000 rpm).
Swap Encoder Direction *Swap_Encoder_Direction* *OptionBits3* [Bit 0]	On/Off *On/Off*	Changes the motor encoder's effective direction of rotation. The encoder provides data used to calculate motor position and speed. This parameter must be set such that when the motor is turning forward, the controller reports back a positive motor speed. **Positive motor speed must be in the forward direction in order for the emergency reverse feature to operate properly.** 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. For instructions, see Section 5, Step ⑨ (page 77).***
Swap Two Phases *Swap_Two_Phases* *OptionBits3* [Bit 3]	On/Off *On/Off*	If, after Swap Encoder Direction has been set correctly, the vehicle drives in the wrong direction (i.e., drives forward when in reverse, and vice versa), try changing the setting of the Swap Two Phases parameter. This parameter has the same effect as physically swapping the cables on any two of the three motor phase connections. **Positive motor speed must be in the forward direction in order for the emergency reverse feature to operate properly.** 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. For instructions, see Section 5, Step ⑨ (page 77).***
Encoder Steps *Encoder_Steps*	32–256 *32–256*	Sets the number of encoder pulses per revolution. This must be set to match the encoder; see 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 uncommanded drive. For instructions, see Section 5, Step ① (page 75).***

MOTOR TEMPERATURE CONTROL MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Sensor Enable *MotorTemp_Sensor_Enable* *OptionBits3* [Bit 1]	On/Off *On/Off*	When programmed On, the motor temperature cutback and the motor temperature compensation features are enabled. This parameter can be used only if a temperature sensor has been properly configured. The motor temperature cutback feature will linearly cutback the drive 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 more consistant performance.
Sensor Type *MotorTemp_Sensor_Type*	1–5 *1–5*	Five sensor types are predefined in the software:. Type 1 KTY83–122 Type 2 2× Type 1, in series Type 3 KTY84–130 or KTY84–150 Type 4 2× Type 3, in series Type 5 PT1000. Custom sensor types can be set up easily, if none of the five predefined types is appropriate for your application. Please contact your Curtis customer support engineer. Note: The industry standard KTY temperature sensors are silicon temperature sensors with a polarity band; **the polarity band of a KTY sensor must be the end connected to I/O Ground** (pin 7).
Sensor Temp Offset *MotorTemp_Sensor_Offset*	-20 – 20 °C *-200–200*	Often the 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 Hot *MotorTemp_Hot*	0–250 °C *0–2500*	Defines the temperature at which drive current cutback begins.
Temperature Max *MotorTemp_Max*	0–250 °C *0–2500*	Defines the temperature at which drive current is cut back to zero.
MotorTemp LOS Max Speed *MotorTemp_LOS_Max_Speed*	100–3000 rpm *100–3000*	When a Motor Temp Sensor Fault (fault code 29) is set, a LOS (Limited Operating Strategy) mode is engaged. The maximum speed is reduced to the programmed Max Speed in the operating mode (Max_Speed_SpdMx, Max_Speed_SpdM, Max_Speed_TrqM) ***or*** to the programmed MotorTemp_LOS_Max_Speed, whichever is lower.

HYDRAULIC OPERATION

The 1298 controls the speed of the pump motor, and also the valves on the Lift cylinder's hydraulic line. By so doing, it controls the hydraulic path for Lift and Lower operations. The hydraulic path for any other hydraulic operations (e.g., reach, tilt, sideshift, rotate) is provided by the vehicle manufacturer, with the 1298 controlling the pump motor speed. VCL programming and spare 1298 inputs and outputs could be used to control the other hydraulic valves.

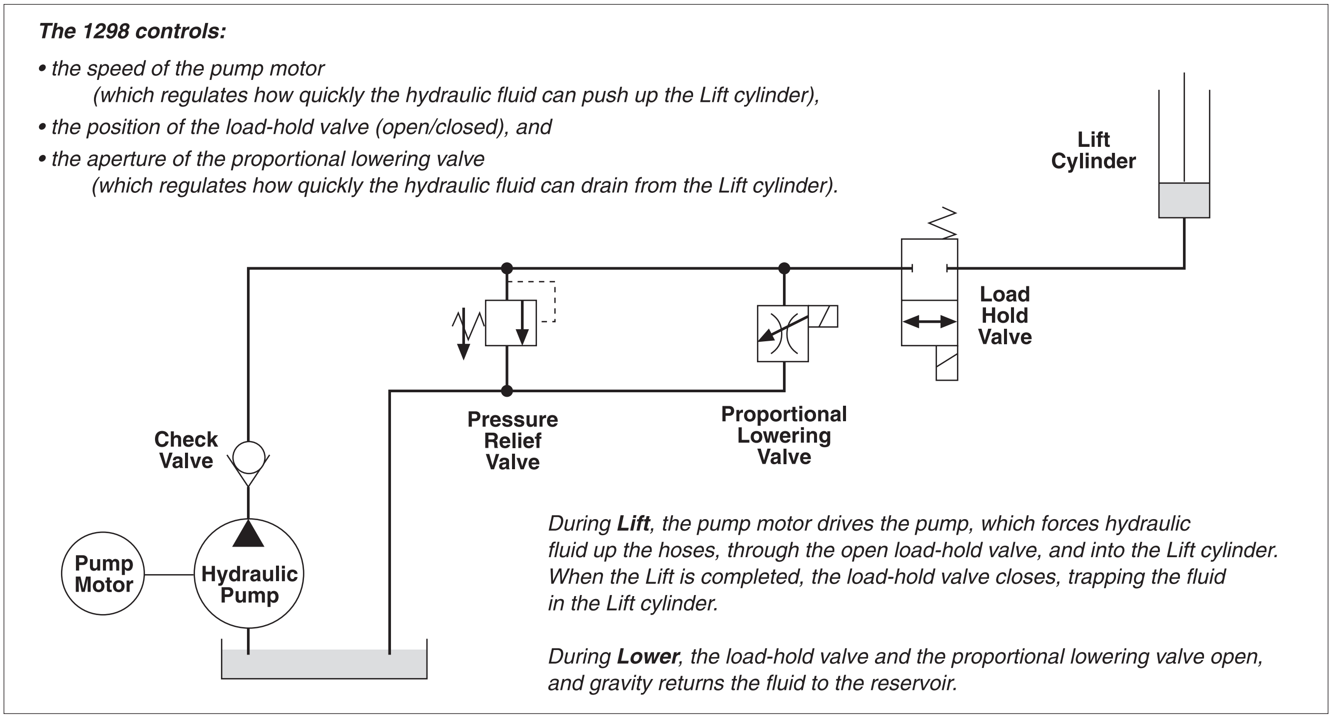


Fig. 15 *Hydraulic system diagram.*

The standard configuration is shown in Figure 15. In some alternative systems a simple open/closed lowering valve is used, as shown in Figure 16.

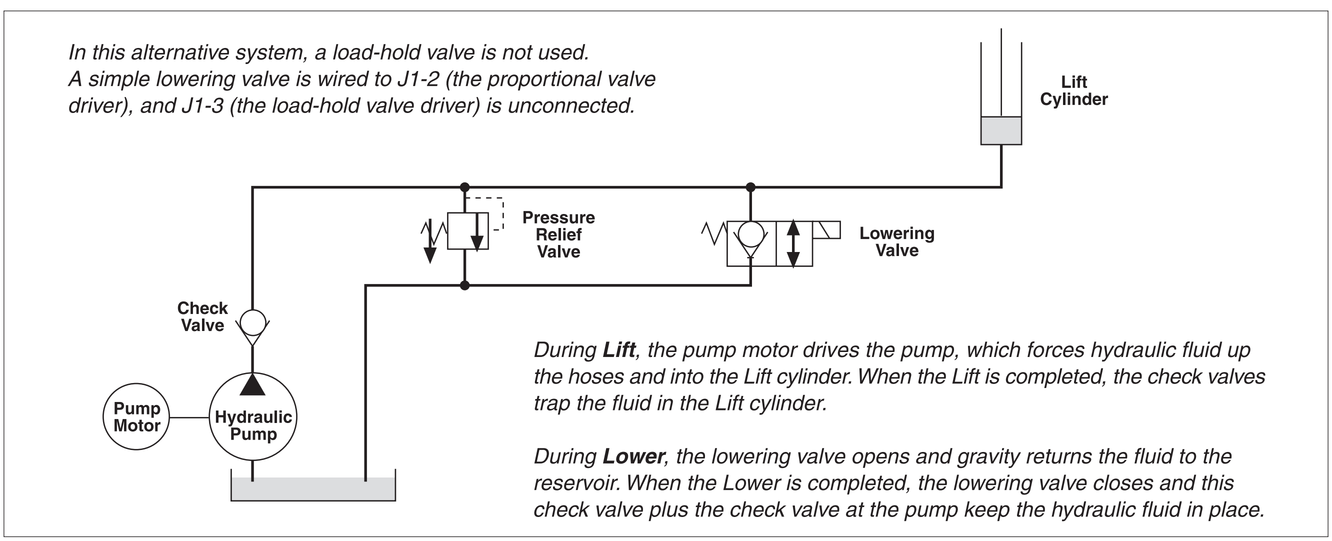


Fig. 16 *Alternative hydraulic system, without proportional lowering.*

The various hydraulics parameters are used to adjust the system's operating characteristics—its acceleration, speed, and responsiveness. These parameters allow the hydraulic system to be tailored to a specific application, or to a specific operator's preferences.

HYDRAULICS MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Lift Switch Only Enable *Lift_Switch_Only_Enable* *OptionBits4* [Bit 4]	On/Off *On/Off*	When programmed On, closing the Lift switch turns on the DC pump, at its maximum speed. When programmed Off, DC pump speed varies according to the position of the hydraulic throttle. Note: 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.
Lower Switch Only Enable *Lower_Switch_Only_Enable* *OptionBits4* [Bit 5]	On/Off *On/Off*	When programmed On, closing the Lower switch opens the proportional valve completely. When programmed Off, the aperture of the proportional valve varies according to the position of the hydraulic throttle. Note: 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.
Pump Max PWM *Pump_Max_PWM*	10–100% *3276–32767*	Defines the maximum allowed armature PWM output during pump operation. Setting the max PWM sets the maximum speed of the pump motor.
Pump Accel Rate *Pump_Accel_Rate*	0.1–2.0 sec *100–2000*	Sets the acceleration rate of the throttle request to the DC pump. Higher values mean slower acceleration.
Pump Decel Rate *Pump_Decel_Rate*	0.1–2.0 sec *100–2000*	Sets the deceleration rate of the throttle request to the DC pump. Higher values mean slower deceleration.
Lower Accel Rate *Lower_Accel_Rate*	0.1–2.0 sec *100–2000*	Sets the acceleration rate of the current request to the proportional valve that controls the lowering function. Higher values mean slower acceleration.
Lower Decel Rate *Lower_Decel_Rate*	0.1–2.0 sec *100–2000*	Sets the deceleration rate of the current request to the proportional valve that controls the lowering function. Higher values mean slower deceleration.
Load Hold Enable *Load_Hold_Enable* *Load_Hold_Enable_Bit0* [Bit 0]	On/Off *On/Off*	When programmed On, the load hold valve is controlled by Driver 4. If your application does not include a load hold valve, set this parameter Off.
Load Hold Delay *Load_Hold_Delay*	0–2048 msec *0–256*	Defines how long the load hold valve is kept open at the end of a lift or lower action (i.e., after pump speed has reached zero at completion of a lift action, or after the proportional valve has closed at completion of a lowering action). The load hold valve is either open or shut, which means it closes abruptly. To prevent jitter it is important that the delay time be set long enough for the hydraulic fluid to stop flowing before the load hold valve snaps shut.

HYDRAULICS MENU, cont'd		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Lift BDI Lockout *Lift_BDI_Lockout* *OptionBits4* [Bit 0]	On/Off *On/Off*	When programmed On, this parameter enables a check of the BDI_Percentage variable. When BDI_Percentage falls to 0%, the Pump BDI fault is set and pump operation is locked out starting with the next pump operation.
Hyd Inhibit Type (HPD) *Hyd_Inhibit_Type*	0–3 *0–3*	The hydraulic inhibit function prohibits lift or lowering operation if the hydraulic throttle request is greater than 25% within 250 msec of KSI being turned on. The following types of hydraulic inhibit can be set: 0 = Hydraulic inhibit disabled. 1 = Hydraulic inhibit enabled for lift (pump) operation and disabled for lowering operation. 2 = Hydraulic inhibit enabled for lowering operation and disabled for lift (pump) operation. 3 = Hydraulic inhibit enabled for both lift (pump) and lowering operation.

For variable speed control, a throttle is required. Without a throttle, when the Lift switch is closed the pump accelerates to the set maximum pump speed in the set Pump Accel time; when the Lower switch is closed, the lowering valve current ramps from 0% to 100% in the set Lower Accel time.

HYDRAULIC THROTTLE MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Hyd Throttle Type *Hyd_Throttle_Type*	1–5 *1–5*	The 1298 controller accepts a variety of throttle inputs. The hydraulic throttle type parameter can be programmed as follows: 1 2-wire rheostat, 5kΩ–0 input 2 single-ended 3-wire 1kΩ–10kΩ potentiometer, or 0–5V voltage source 3 2-wire rheostat, 0–5kΩ input 4 wigwag 3-wire 1kΩ–10kΩ potentiometer, or 0–5V voltage source 5 VCL input (*VCL_Hyd_Throttle*) Note: 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.
Lift Deadband *Lift_Deadband*	0–5.00 V *0–32767*	Defines the wiper voltage at the hydraulic throttle deadband threshold. Increasing the hydraulic throttle deadband setting will increase the neutral range. 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.
Lift Map *Lift_Map*	10–100% *3276–32767*	Modifies the pump's response to the hydraulic throttle input. Setting the lift map at 50% provides linear output response to hydraulic throttle position. Values below 50% reduce controller output at low hydraulic throttle settings, thus providing enhanced low-speed control of the pump. Values above 50% give the pump a faster, more responsive feel at low throttle positions. The map value is the percentage of controller output at half throttle ((deadband + max) / 2).
Lift Max *Lift_Max*	0–5.00 V *0–32767*	Defines the wiper voltage required to produce 100% pump output. Decreasing the lift max setting reduces the wiper voltage and therefore the full stroke necessary to produce full pump output. This parameter allows reduced-range throttle assemblies to be accommodated.
Lift Offset *Lift_Offset*	0–100% *0–32767*	Defines the initial pump output generated when the hydraulic throttle is first rotated out of the neutral deadband. For most pump systems, a setting of zero is appropriate. For some pump systems, however, increasing the offset may improve controllability by reducing the amount of throttle required to start the pump load moving.

HYDRAULIC THROTTLE MENU, cont'd		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Lower Deadband *Lower_Deadband*	0–5.00 V *0–32767*	Defines the wiper voltage at the hydraulic throttle deadband threshold. Increasing the hydraulic throttle deadband setting will increase the neutral range. 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.
Lower Map *Lower_Map*	10–100% *3276–32767*	Modifies the proportional valve driver's response to the hydraulic throttle input. Setting the lower map at 50% provides linear output response to hydraulic throttle position. Values below 50% reduce the driver's output at low hydraulic throttle settings, thus providing enhanced low-speed control of the proportional valve. Values above 50% give the proportional valve a faster, more responsive feel at low throttle positions. The map value is the percentage of controller output at half throttle ((deadband + max) / 2).
Lower Max *Lower_Max*	0–5.00 V *0–32767*	Defines the wiper voltage required to produce 100% proportional valve driver current. Decreasing the lower max setting reduces the wiper voltage and therefore the full stroke necessary to produce full driver current. This parameter allows reduced-range throttle assemblies to be accommodated.
Lower Offset *Lower_Offset*	0–100% *0–32767*	Defines the initial proportional valve driver current generated when the hydraulic throttle is first rotated out of the neutral deadband. For most hydraulic systems, a setting of zero is appropriate. For some hydraulic systems, however, increasing the offset may improve controllability by reducing the amount of throttle required to open the proportional valve.
VCL Hyd Throttle Enable *VCL_Hyd_Throttle_Enable* *VCL_Hyd_Throttle_Enable_Bit0* [Bit 0]	On/Off *On/Off*	When programmed On, this parameter provides a VCL alternative to Throttle Type 5, allowing you instead to select Hyd Throttle Types 1–4, which have automatic throttle fault protection. As shown in the hydraulic command chain (Figure 19, page 97), enabling this parameter breaks the chain—with the first part providing normal throttle pot processing (with throttle fault protection), and the second part under the control of VCL. However, in order to "combine the best of both worlds," you must write VCL code to define the connection between the OS_Hyd_Throttle and VCL_Hyd_Throttle variables. This combination allows VCL flexibility and customization of hydraulic throttle processing, while still allowing hydraulic types 1–4 with their automatic throttle fault detection.

BATTERY MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Nominal Voltage *Nominal_Voltage*	24–84 V *1536–5376*	Must be set to the vehicle's nominal battery pack voltage. This parameter is used in determining the overvoltage and undervoltage protection thresholds for the electronic system. Overvoltage protection cuts back regen braking to prevent damage to batteries and other electrical system components due to overvoltage. Undervoltage protection prevents systems from operating at voltages below their design thresholds. The four threshold points are calculated from the Nominal Voltage, Undervoltage Cutback Range, User Overvoltage, and User Undervoltage parameter settings and the controller's minimum voltage and maximum voltage ratings: VOLTAGE RATINGS CONTROLLER 24V; BROWNOUT VOLTAGE * 15V; MIN VOLTAGE 16.8V; MAX VOLTAGE 30V Overvoltage = Either Max Voltage (see voltage ratings table) *or* User Overvoltage × Nominal Voltage, whichever is lower. Severe Overvoltage = Overvoltage (see previous item) + 10V. Undervoltage = Either Min Voltage (see voltage ratings table) *or* User Undervoltage × Nominal Voltage, whichever is higher. Severe Undervoltage = Undervoltage point – Undervoltage Cutback Range. * The Brownout Voltage is determined by the controller base type and cannot be changed. When the capacitor voltage falls below the Brownout voltage the bridge is switched off (i.e., motor current is switched off). If the capacitor voltage stays below the Brownout voltage for > 64 msec the controller will reset (equivalent to cycling the keyswitch). If the capacitor voltage rises above the Brownout voltage before 64 msec have passed the bridge will be reenabled. The Severe Undervoltage point can be set lower than the Brownout voltage.
Undervoltage Cutback Range *Undervoltage_Cutback_Range*	2.0–14.0 V *0–4096*	This parameter sets the voltage range between the Undervoltage and Severe Undervoltage points (see Nominal Voltage description). A Severe Undervoltage fault will be set if the capacitor voltage falls below either the Severe Undervoltage point (drive current limit set to 0) or the Brownout voltage (bridge disabled, motor current set to 0).
User Overvoltage *User_Overvoltage*	115–200 % *293–512*	The value of this parameter is a percentage of the Nominal Voltage setting. The User Overvoltage parameter can be used to adjust the overvoltage threshold, which is the voltage at which the controller will cut back regen braking to prevent damage to the electrical system. Typically this parameter is changed only when the controller is being used in an application at the low end of the controller's range: such as a 48–80V controller being used in a system with a 48V battery pack. In this case, the overvoltage threshold can be raised by setting the User Overvoltage to a higher value. The overvoltage threshold can never be raised above the controller's power base maximum voltage rating.

BATTERY MENU, cont'd

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
User Undervoltage *User_Undervoltage*	50–80 % *128–204*	The value of this parameter is a percentage of the Nominal Voltage setting. The User Undervoltage parameter can be used to adjust the undervoltage threshold, which is the voltage at which the controller will cut back drive current to prevent damage to the electrical system. Typically this parameter is changed only when the controller is being used in an application at the high end of the controller's range: such as a 24–36V controller being used in a system with a 36V battery pack. In this case, the undervoltage threshold can be lowered by setting the User Undervoltage to a lower value. The undervoltage threshold can never be lowered below the controller's power base minimum voltage rating.

BDI Algorithm

The BDI (battery discharge indicator) algorithm continuously calculates the battery state-of-charge whenever KSI is on. The result of the BDI algorithm is the variable BDI Percentage, which is viewable in the 1313 menu Monitor » Battery. When KSI is turned off, the present BDI Percentage is stored in nonvolatile memory.

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

	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 the standard values for your type of batteries as the starting point in setting the reset, full, and empty volts-per-cell parameters.

BATTERY MENU, cont'd		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Reset Volts Per Cell *BDI_Reset_Volts_Per_Cell*	0.90–3.00 V *900–3000*	The reset voltage level is checked only once, when KSI is first turned on. Note that the BDI Reset Percent parameter also influences the algorithm that determines whether BDI Percentage is reset to 100%. Reset Volts Per Cell should always be set higher than Full Volts Per Cell. Reset Voltage Level = Reset Volts Per Cell × number of cells in the battery pack.*
Full Volts Per Cell *BDI_Full_Volts_Per_Cell*	0.90–3.00 V *900–3000*	The full voltage level sets the Keyswitch Voltage that is considered to be 100% state-of-charge; when a loaded battery drops below this voltage, it begins to lose charge. Keyswitch Voltage is viewable in the 1313 menu Monitor » Battery. Full Voltage Level = Full Volts Per Cell × number of cells in the battery pack.*
Empty Volts Per Cell *BDI_Empty_Volts_Per_Cell*	0.90–3.00 V *900–3000*	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*	0–600 min. *0–600*	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*	0–100 % *0–100*	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)).

* To determine the number of cells in your battery pack, divide your Nominal Voltage setting (page 55) by 2.

DUAL DRIVE MENU
FOR DUAL DRIVE PARAMETERS, SEE THE DUAL DRIVE ADDENDUM, P/N 38272-DD.

VEHICLE MENU		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Metric Units *Metric_Units* *OptionBits3* [Bit 5]	On/Off *On/Off*	When this parameter is programmed 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). Distance variables are displayed in the Monitor » Vehicle menu, page 71.
Speed to RPM *Speed_to_RPM*	10.0–3000.0 *100–30000*	This parameter affects the vehicle speed displayed in the Monitor » Motor menu (see page 68), and also modifies the VCL variable *Vehicle_Speed*; it does <u>not</u> 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].
Capture Speed 1 *Capture_Speed_1*	0–8000 rpm *0–8000*	The controller captures the time it takes the motor to go from 0 rpm to the programmed Capture Speed. The result is stored as “Time to Speed 1” in the Monitor » Vehicle menu (page 71). This timer starts every time the motor accelerates from zero speed.
Capture Speed 2 *Capture_Speed_2*	0–8000 rpm *0–8000*	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 Monitor » Vehicle menu.
Capture Distance 1 *Capture_Distance_1*	1–1320 *1–1320*	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 Monitor » Vehicle menu (page 71). 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*	1–1320 *1–1320*	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 Monitor » Vehicle menu.
Capture Distance 3 *Capture_Distance_3*	1–1320 *1–1320*	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 Monitor » Vehicle menu.

EMERGENCY REVERSE MENU **[SPEED MODE & SPEED MODE EXPRESS only]**

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
EMR Enable *EMR_Enable* *OptionBits1* [Bit 1]	On/Off *On/Off*	Determines whether the emergency reverse function is active. On = emergency reverse is enabled. Off = emergency reverse is disabled.
EMR Type *EMR_Type*	0–1 *0–1*	Determines where the input comes from for emergency reverse. 0 = emergency reverse activated by switch 1 (pin 24). 1 = emergency reverse is activated by VCL functions Enable_Emer_Rev() and Disable_Emer_Rev().
EMR Dir Interlock *EMR_Dir_Interlock* *EMR_Dir_Interlock_Bit0* [Bit 0]	On/Off *On/Off*	Determines whether the interlock switch must be turned off after emergency reverse before the vehicle can be driven again. On = Interlock and throttle and direction must all be cleared. Off = Only throttle and direction must be cleared.
EMR Time Limit *EMR_Time_Limit*	0–30 sec *0–3750*	Defines how long emergency reverse is allowed to be 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*	50–6000 rpm *50–6000*	Defines the maximum reverse speed of the motor (in motor rpm), when emergency reverse is active.
EMR Accel Rate *EMR_Accel_Rate*	0.1–3.0 sec *100–3000*	Sets the rate (in seconds) at which the vehicle accelerates in the opposite direction after it has been 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*	0.1–3.0 sec *100–3000*	Sets the rate (in seconds) at which the vehicle brakes to a stop when emergency reverse is activated and 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.

INTERLOCK BRAKING MENU		
PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Enable *Interlock_Brake_Enable* *OptionBits3* [Bit 7]	On/Off *On/Off*	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 Sequencing Delay expires and allow the vehicle to roll freely when the interlock signal is removed. This option is typically used only when there is a user controlled mechanical or hydraulic brake system.
Decel Rate HS *Interlock_Brake_Decel_Rate_HS*	0.1–30.0 *100–30000*	Sets the rate (in seconds) that is used to slow down the vehicle when the interlock is released at high vehicle speeds. Larger values represent slower response.
Decel Rate LS *Interlock_Brake_Decel_Rate_LS*	0.1–30.0 *100–30000*	Sets the rate (in seconds) that is used to slow down the vehicle when the interlock is released at low vehicle speeds. Larger values represent slower response.
Interlock Brake Timeout *Interlock_Brake_Timeout*	0–8.0 sec *0–1000*	Controls the maximum allowable duration of an interlock braking event. The timer starts as soon as the interlock signal is removed. If the time expires before the vehicle has slowed below the Set_Speed_Threshold, the EM brake will engage automatically. This parameter can be used to allow parallel usage of regen braking and the EM brake to reduce stopping distance. If 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 EM_Brake_Type = 1 or 2 (see page 42).*

CAN INTERFACE MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
CANopen Interlock *CANopen_Interlock_Enable* *OptionBits3* [Bit 2]	On/Off *On/Off*	When programmed On, CAN NMT State must = 5 (operational state) in order for the interlock to be set; see Monitor » CAN Status menu, page 73.
CAN Node ID *CAN_Node_ID*	0–127 *0–127*	Sets the Node ID of the CANopen Slave system. The Node ID is the first 7 bits of the 11-bit identifier (the COB ID).
Baud Rate *CAN_Baud_Rate*	-3 – 2 *-3 – 2*	Sets the CAN baud rate for the CANopen Slave system: -3=20Kbps, -2=50Kbps, -1=100Kbps, 0=125Kbps, 1=250Kbps, 2=500Kbps.
Heartbeat Rate *CANopen_Heart_Beat_Rate*	16–200 msec *4–50*	Sets the rate at which the CAN heartbeat messages are sent from the CANopen Slave system.
PDO Timeout Period *CAN_PDO_Timeout_Period*	0–200 msec *0–50*	Sets the PDO timeout period for the CANopen Slave system. After the slave controller has sent a PDO MISO, it will declare a PDO Timeout Fault if the master controller has not sent a reply PDO MOSI message within the set time. Either PDO1 MOSI or PDO2 MOSI will reset the timer. Setting the PDO Timeout Period = 0 will disable this fault check.
Emergency Message Rate *CANopen_Emergency_ Message_Rate*	16–200 msec *4–50*	Sets the minimum rate between CAN emergency messages from the CANopen Slave system. This prevents quickly changing fault states from generating so many emergency messages that they flood the CAN bus.
Suppress CANopen Init *Suppress_CANopen_Init*	0–1 *0–1*	When Suppress CANopen Init is set = 1, at KSI On the initialization of the CANopen system is suppressed. Typically this is done so that the VCL program can make changes to the CANopen system before enabling it (by setting the variable Suppress_CANopen_Init = 0 and running the Setup_CAN() function).

MOTOR CHARACTERIZATION TESTS MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
		Contact your Curtis customer support engineer if you will be running the AC motor characterization tests yourself. See Initial Setup, step ⑩, page 78.

FIELD WEAKENING CONTROL MENU

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
FW Base Speed *FW_Base_Speed*	200–6000 rpm *200–6000*	This parameter needs to be reset each time the Motor Type is changed or the low speed current limit is changed. For example, if you lower Drive_Current_Limit (page 36) or PL_Drive_Nominal (page 37), you should consider adjusting this parameter. To determine the correct value, perform this tuning test. The test should be run with batteries that have a reasonable charge. In either Torque Control Mode or Speed Control Mode, set your accel rates to be fast—so that you'll be accelerating at full current during the test. Next, set the Base Speed parameter to the maximum value (so that it will not interfere with the test result). From a stop, apply full throttle and accelerate to high speed and then stop. After stopping, note the value displayed in Monitor » Controller » Motor Tuning » Base Speed Captured, and enter this value for the Base Speed setting. The test restarts each time the vehicle comes to a stop and the throttle is released, so be sure to note the value before driving away.
Field Weakening *Field_Weakening*	0–100 % *0–1024*	Determines the amount of high speed power the controller will allow, while still maintaining maximum effficiency at the allowed power. Reducing this parameter effectively reduces controller current at high speeds, which can reduce energy consumption and motor heating, but at the expense of reduced available torque from the motor.
Weakening Rate *Field_Weakening_Rate*	0–100 % *0–500*	Determines 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.

MOTOR TYPE PARAMETER

PARAMETER	ALLOWABLE RANGE	DESCRIPTION
Motor Type *Motor_Type*	0–200 *0–200*	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.

CLONING (for copying parameter settings to multiple controllers)

Once a controller has been programmed to the desired settings, these settings can be transferred as a group to other controllers, thus creating a family of "clone" controllers with identical settings. **Cloning only works between controllers with the same model number and software version.** For example, the programmer can read all the information from a 1298-2205 controller and write it to other 1298-2205 controllers; however, it cannot write that same information to 1298-2206 controllers.

To perform cloning, plug the programmer (1313 or 1314) into the controller that has the desired settings. Select the Program menu; follow the prompts to copy the settings into the programmer.

Plug the programmer into the controller that you want to have these same settings, and follow the Program menu prompts to write these settings into the controller.

Note: For cloning Dual Drive controllers, see the separate Dual Drive addendum, p/n 38272-DD.

4a

MONITOR MENU

Through its Monitor menu, the 1313 programmer provides access to real-time data during vehicle operation. This information is helpful during diagnostics and troubleshooting, and also while adjusting programmable parameters.

Monitor Menu: INPUTS

VARIABLE	DISPLAY RANGE	DESCRIPTION
Throttle Command *Throttle_Command*	-100–100 % *-32768–32767*	Throttle request to slew rate block.
Mapped Throttle *Mapped_Throttle*	-100–100 % *-32768–32767*	Mapped throttle request.
Throttle Pot *Throttle_Pot_Raw*	0–5.5 V *0–36044*	Voltage at throttle pot wiper (pin 16).
Brake Command *Brake_Command*	0–100 % *0–32767*	Brake request to slew rate block.
Mapped Brake *Mapped_Brake*	0–100 % *0–32767*	Mapped brake request.
Pot2 Raw *Pot2_Raw*	0–5.5 V *0–36044*	Voltage at pot2 wiper (pin 17).
Mapped Hyd Throttle *Mapped_Hyd_Throttle*	-100 – 100 % *-32767 – 32767*	Mapped hydraulic throttle request to slew block (from lowering valve or pump).
Pump Throttle *Pump_Throttle*	0–100 % *0–32767*	Pump throttle request after slew block.
PD Throttle *PD_Throttle*	0–100 % *0–32767*	Proportional driver current request.
Steer Pot *Steer_Pot_Raw*	0–5.5 V *0–36044*	Voltage at steer pot wiper (pin 17) on Dual Drive traction slave.
Steer Angle *Steer_Angle*	-90 – 90 deg *-90 – 90*	Steer angle calculated in Dual Drive traction master.

Monitor Menu: INPUTS, cont'd

VARIABLE	DISPLAY RANGE	DESCRIPTION
Interlock *Interlock_State* *System_Flags1* [Bit 0]	On/Off *On/Off*	Interlock input on or off. The source of the interlock input is determined by the Interlock Type parameter: from Switch 3 (pin 9) if Interlock Type = 0 from VCL function if Interlock Type = 1 from KSI (pin 1) if Interlock Type = 2.
Emer Rev *EMR_State* *System_Flags1* [Bit 1]	On/Off *On/Off*	Emergency reverse input on or off. The source of the emergency reverse input is determined by the EMR Type parameter: from Switch 1 (pin 24) if EMR Type = 0 from VCL function if EMR Type = 1.
Analog 1 *Analog1_Input*	0–10.0 V *0–1023*	Voltage at analog 1 (pin 24).
Analog 2 *Analog2_Input*	0–10.0 V *0–1023*	Voltage at analog 2 (pin 8).
Switch 1 *Sw_1* *Switches* [Bit 0]	On/Off *On/Off*	Switch 1 on or off (pin 24).
Switch 2 *Sw_2* *Switches* [Bit 1]	On/Off *On/Off*	Switch 2 on or off (pin 8).
Switch 3 *Sw_3* *Switches* [Bit 2]	On/Off *On/Off*	Switch 3 on or off (pin 9).
Switch 4 *Sw_4* *Switches* [Bit 3]	On/Off *On/Off*	Switch 4 on or off (pin 10).
Switch 5 *Sw_5* *Switches* [Bit 4]	On/Off *On/Off*	Switch 5 on or off (pin 11).
Switch 6 *Sw_6* *Switches* [Bit 5]	On/Off *On/Off*	Switch 6 on or off (pin 12).
Switch 7 *Sw_7* *Switches* [Bit 6]	On/Off *On/Off*	Switch 7 on or off (pin 22).
Switch 8 *Sw_8* *Switches* [Bit 7]	On/Off *On/Off*	Switch 8 on or off (pin 33).

Monitor Menu: INPUTS, cont'd

VARIABLE	DISPLAY RANGE	DESCRIPTION
Driver 1 Input *Sw_9* *Switches* [Bit 8]	On/Off *On/Off*	Driver 1 input on or off (pin 6).
Driver 2 Input *Sw_10* *Switches* [Bit 9]	On/Off *On/Off*	Driver 2 input on or off (pin 5).
Driver 3 Input *Sw_11* *Switches* [Bit 10]	On/Off *On/Off*	Driver 3 input on or off (pin 4).
Driver 4 Input *Sw_12* *Switches* [Bit 11]	On/Off *On/Off*	Driver 4 input on or off (pin 3).
PD Input *Sw_13* *Switches* [Bit 12]	On/Off *On/Off*	Proportional driver on or off (pin 2).
DigOut6 Input *Sw_14* *Switches* [Bit 13]	On/Off *On/Off*	Digital Out 6 input on or off (pin 19).
DigOut7 Input *Sw_15* *Switches* [Bit 14]	On/Off *On/Off*	Digital Out 7 input on or off (pin 20).
Switch 16 *Sw_16* *Switches* [Bit 15]	On/Off *On/Off*	Switch 16 on or off (pin 14).

Monitor Menu: OUTPUTS

VARIABLE	DISPLAY RANGE	DESCRIPTION
Analog Out *Analog_Output*	0–10.0 V *0–32767*	Voltage at Analog output (pin 30).
Digital Out 6 *Dig6_Output*	On/Off *On/Off*	Digital Out 6 output on or off (pin 19).
Digital Out 7 *Dig7_Output*	On/Off *On/Off*	Digital Out 7 output on or off (pin 20).
Driver 1 PWM *PWM1_Output*	0–100 % *0–32767*	Driver 1 PWM output (pin 6).
Driver 2 PWM *PWM2_Output*	0–100 % *0–32767*	Driver 2 PWM output (pin 5).
Driver 3 PWM *PWM3_Output*	0–100 % *0–32767*	Driver 3 PWM output (pin 4).
Driver 4 PWM *PWM4_Output*	0–100 % *0–32767*	Driver 4 PWM output (pin 3).
PD PWM *PD_Output*	0–100 % *0–32767*	Proportional driver PWM output (pin 2).
Pump PWM *Pump_Output*	0–100 % *0–32767*	PWM output of the DC pump motor.
PD Current *PD_Current*	0–2.0 A *0–607*	Current at proportional driver (pin 2).
Pump Current *Pump_Current*	-100 – 400 A *-1000 – 4000*	Current in the DC pump motor.
5 Volts *Five_Volts_Output*	0–6.25 V *0–1023*	Voltage at +5V output (pin 26).
Ext Supply Current *Ext_Supply_Current*	5–200 mA *52–800*	Combined current of the external +12V and +5V voltage supplies (pins 25 and 26).
Pot Low *Pot_Low_Output*	0–6.25 V *0–1023*	Voltage at pot low (pin 18).

Monitor Menu: BATTERY		
VARIABLE	DISPLAY RANGE	DESCRIPTION
BDI *BDI_Percentage*	0–100 % *0–100*	Battery state of charge.
Capacitor Voltage *Capacitor_Voltage*	0–105 V *0–6720*	Voltage of controller's internal capacitor bank at B+ terminal.
Keyswitch Voltage *Keyswitch_Voltage*	0–105 V *0–10500*	Voltage at KSI (pin 1).

Monitor Menu: MOTOR		
VARIABLE	DISPLAY RANGE	DESCRIPTION
Motor RPM *Motor_RPM*	-12000–12000 rpm *-12000–12000*	Motor speed in revolutions per minute.
Temperature *Motor_Temperature*	-100–300 °C *-1000–3000*	Temperature sensor readout.
MotorSpeed A *MotorspeedA*	0–12000 rpm *0–12000*	Motor encoder phase A speed in revolutions per minute. This can be used to verify that phase A of the encoder is operating correctly. MotorSpeed A should equal MotorSpeed B in a properly operating motor encoder. MotorSpeed A does not indicate direction.
MotorSpeed B *MotorspeedB*	0–12000 rpm *0–12000*	Motor encoder phase A speed in revolutions per minute. This can be used to verify that phase B of the encoder is operating correctly. MotorSpeed B should equal MotorSpeed A in a properly operating motor encoder. MotorSpeed B does not indicate direction.

Monitor Menu: CONTROLLER

VARIABLE	DISPLAY RANGE	DESCRIPTION
Current (RMS) *Current_RMS*	0–1000 A *0–10000*	RMS current of the controller, taking all three phases into account.
Modulation Depth *Modulation_Depth*	0–100 % *0–1182*	Percentage of available voltage being used.
Frequency *Frequency*	-300–300 Hz *-18000–18000*	Controller electrical frequency.
Temperature *Controller_Temperature*	-100–300 °C *-1000–3000*	Controller internal temperature.
Main State *Main_State*	0–10 *0–10*	Main contactor state: 0= open 1= precharge 2= weldcheck 3= closingdelay 4= missingcheck 5= closed (when Main Enable = On) 6= delay 7= arccheck 8= opendelay 9= fault 10= closed (when Main Enable = Off).
Regen *Regen_State* *System_Flags1* [Bit 2]	On/Off *On/Off*	On when regen braking is taking place; Off when it is not.
VCL Error Module *Last_VCL_Error_Module* **VCL Error** *Last_VCL_Error*	0–65536 *0–65536* 0–65536 *0–65536*	A VCL Runtime Error (fault code 68) will store additional information about the cause of a VCL runtime error in the VCL Error Module and VCL Error variables. The resulting non-zero values can be compared to the runtime VCL module ID and error code definitions listed in the controller's OS SysInfo file, which should help pinpoint the VCL error that caused the runtime error.
Motor Characterization Error *Motor_Characterization_Error*	0–65536 *0–65536*	A Motor Characterization Error (fault code 87) will store additional information in the Motor Characterization Error variable: 0= none 1= encoder signal seen but unable to determine step size; must set up Encoder Step Size manually 2= motor temp sensor fault 3= motor temp hot cutback fault 4= controller overtemp cutback fault 5= controller undertemp cutback fault 6= undervoltage cutback fault 7= severe overvoltage fault 8= encoder signal not seen, or one or both channels missing 9= motor parameters out of characterization range.

Monitor Menu: CUTBACKS		
VARIABLE	DISPLAY RANGE	DESCRIPTION
Motor Temp Cutback *MotorTempCutback*	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 Temp Cutback *ControllerTempCutback*	0–100 % *0–4096*	Displays the current available as a result the controller temperature cutback function. A value of 100% indicates no cutback in current.
Undervoltage Cutback *UndervoltageCutback*	0–100 % *0–4096*	Displays the current available as a result the undervoltage cutback function. A value of 100% indicates no cutback in current.
Overvoltage Cutback *OvervoltageCutback*	0–100 % *0–4096*	Displays the current available as a result the overvoltage cutback function. A value of 100% indicates no cutback in current.

Monitor Menu: MOTOR TUNING		
VARIABLE	DISPLAY RANGE	DESCRIPTION
Base Speed Captured *Base_Speed_Captured*	0–8000 rpm *0–8000*	Displays the value of the motor base speed captured in the most recent acceleration. This value is used to set the Base Speed parameter (Program » Motor Control Tuning » Field Weakening Control menu), using the Base Speed set procedure described on page 62.

☞ *Note: All vehicle calculations assume no tire slippage.*

Monitor Menu: VEHICLE

VARIABLE	DISPLAY RANGE	DESCRIPTION
Vehicle Speed *Vehicle_Speed*	-327.7–327.7 *-32768–32767*	Vehicle speed, in units of MPH or KPH, depending on the setting of the Metric Units parameter (see Program » Vehicle menu, page 58). For accurate speed estimates, the Speed to RPM parameter must be set correctly (page 58).
Vehicle Odometer *Vehicle_Odometer*	0–429496729.5 *0–4294967295*	Vehicle distance traveled, in units of miles or km, depending on the setting of the Metric Units parameter (page 58). For accurate distance measurements, the Speed to RPM parameter must be set correctly (page 58).
Vehicle Acceleration *Vehicle_Acceleration*	-10–10 g *-10000–10000*	Vehicle acceleration. The Speed to RPM parameter must be set correctly for accurate measurement.
Time to Speed 1 *Time_to_Capture_Speed_1*	0–128 sec *0–32000*	Time taken for the vehicle to go from zero rpm to the programmed Capture Speed 1 (see Program » Vehicle menu, page 58) during its most recent such acceleration.
Time to Speed 2 *Time_to_Capture_Speed_2*	0–128 sec *0–32000*	Time taken for the vehicle to go from zero rpm to the programmed Capture Speed 2 (see Program » Vehicle menu) during its most recent such acceleration.
Time Between Speeds *Time_Between_Capture_Speeds*	0–128 sec *0–32000*	Time taken for the vehicle to go from the programmed Capture Speed 1 to the programmed Capture Speed 2 (see Program » Vehicle menu, page 58) during its most recent such acceleration.
Time to Dist 1 *Time_to_Capture_Dist_1*	0–128 sec *0–32000*	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 1 (see Program » Vehicle menu, page 58) during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly (page 58).
Time to Dist 2 *Time_to_Capture_Dist_2*	0–128 sec *0–32000*	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 2 (see Program » Vehicle menu) during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly (page 58).

Monitor Menu: VEHICLE, cont'd		
VARIABLE	DISPLAY RANGE	DESCRIPTION
Time to Dist 3 *Time_to_Capture_Dist_3*	0–128 sec *0–32000*	Time taken for the vehicle to travel from zero rpm to the programmed Capture Distance 3 (see Program » Vehicle menu) during its most recent such trip. For accurate distance measurements, the Speed to RPM parameter must be set correctly (page 58).
Braking Distance Captured *Braking_Distance_Captured*	0–1000000.0 *0–400000000*	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 (page 58). For accurate distance measurements, the Speed to RPM parameter must be set correctly (page 58).
Distance Since Stop *Distance_Since_Stop*	0–1000000.0 *0–400000000*	Distance traveled by the vehicle starting from a stop. In effect, the vehicle is used 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 (page 58). Units are meters or feet, depending on the setting of the Metric Units parameter (page 58).
Distance Fine *Distance_Fine_Long*	-214748364.8–214748364.7 *-2147483648–2147483647*	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. Units are decimeters or inches, depending on the setting of the Metric Units parameter (page 58). For accurate distance measurements, the Speed to RPM parameter must be set correctly (page 58).

Monitor Menu: CAN STATUS		
VARIABLE	DISPLAY RANGE	DESCRIPTION
CAN NMT State *CAN_NMT_State*	0–127 *0–127*	Controller CAN NMT state: 0=initialization, 4=stopped, 5=operational, 127=pre-operational.
PDO1 MOSI Byte Map*	$0 – 2^{32}$	Mapping objects for PDO1 MOSI's eight bytes.
PDO1 MISO Byte Map*	$0 – 2^{32}$	Mapping objects for PDO1 MISO's eight bytes.
PDO2 MOSI Byte Map*	$0 – 2^{32}$	Mapping objects for PDO2 MOSI's eight bytes.
PDO2 MISO Byte Map*	$0 – 2^{32}$	Mapping objects for PDO2 MISO's eight bytes.

* Each of these byte maps is a submenu containing 8 variables, one for each byte. Each variable is 32 bits. For example, the PDO1 MOSI Byte Map menu looks like this:

PDO1 MOSI Byte Map

1	$0 – 2^{32}$	Mapping object for byte 1 of PDO1 MOSI.
2	$0 – 2^{32}$	Mapping object for byte 2 of PDO1 MOSI.
3	$0 – 2^{32}$	Mapping object for byte 3 of PDO1 MOSI.
4	$0 – 2^{32}$	Mapping object for byte 4 of PDO1 MOSI.
5	$0 – 2^{32}$	Mapping object for byte 5 of PDO1 MOSI.
6	$0 – 2^{32}$	Mapping object for byte 6 of PDO1 MOSI.
7	$0 – 2^{32}$	Mapping object for byte 7 of PDO1 MOSI.
8	$0 – 2^{32}$	Mapping object for byte 8 of PDO1 MOSI.

4b

CONTROLLER INFORMATION MENU

This menu provides ID and version numbers for your controller hardware and software.

CONTROLLER INFORMATION MENU

VARIABLE	DISPLAY RANGE	DESCRIPTION
Model Number *Model_Number*	0–4294967295 *0–4294967295*	Model number. For example, if you have a controller with the model number 1298-2207, the Model Number variable will have a value of 12982207.
Serial Number *Serial Number*	0–4294967295 *0–4294967295*	Serial number. For example, if the serial number printed on your controller is 08045L.11493, the Serial Number variable will have the value of 11493.
Mfg Date Code *Manuf_Date*	0–32767 *0–32767*	Controller date of manufacture, with the first two digits indicating the year and the last three indicating the day. For example, if the serial number printed on your controller is 08045L.11493, the Mfg Date Code variable will have the value of 08045 (45th day of 2008).
Hardware Version *Hardware_Ver*	0–32.767 *0–32767*	The hardware version number uniquely describes the combination of power base assembly and the logic, cap, and IMS board assemblies used in the controller.
OS Version *OS_Ver*	0–32767 *0–32767*	Version number of the operating system software that is loaded into the controller. This variable specifies the major version number of the controller's operating system.
Build Number *Build_Number*	0–32767 *0–32767*	Build number of the operating system software that is loaded into the controller. This variable specifies the minor version number of the controller's operating system.
SM Version *SM_Ver*	0–327.67 *0–32767*	Version number of the Start Manager software that is loaded into the controller.
Param Blk Version *Param_Blk_Ver*	0–327.67 *0–32767*	Version number of the parameter block that is loaded into the controller.
VCL App Version *VCL_App_Ver*	0–327.67 *0–32767*	Version number of the VCL application software that is loaded into the controller. This value is set in the VCL program by assigning a value to the VCL_App_Ver variable.

5

INITIAL SETUP

The 1298 controller can be used in a variety of vehicles, which differ widely in characteristics. Before driving the vehicle, it is imperative that these initial setup procedures be carefully followed to ensure that the controller is set up to be compatible with your application. The first step is to contact Curtis:

✻ ✻ ✻ BEFORE YOU START ✻ ✻ ✻

Correct values for the AC motor parameters (Motor Type, FW Base Speed, and Field Weakening) **must be determined individually for each AC motor.** You can determine these values in any one of the following ways:

- ➡ Contact Curtis with the manufacturer's part number for your motor. We have a database of many AC motors for which we have already determined the correct motor parameter settings.
- ➡ Use the AC Motor Characterization Procedure, which has the controller "learn" the AC motor parameter data. The procedure should take about 30 minutes to complete and can be done on the vehicle. Contact Curtis for the procedure. Go ahead and complete setup steps ① through ⑨ before conducting the characterization procedure.
- ➡ Send your AC motor to Curtis for testing on the motor dyno. Your motor's data will be entered into the Curtis database and we will send you the appropriate parameter values to enter into your controller. Contact Curtis before shipping your motor.

Once you have obtained the correct values for Motor Type, FW Base Speed, and Field Weakening and have set them on your controller (see Motor Control menus, page 62), you can start conducting the setup procedures. Note: If you will be conducting the AC Motor Characterization Procedure, that will come later.

CAUTION

Before starting the setup procedures, **jack the vehicle drive wheels up off the ground** so that they spin freely. Double-check all wiring to ensure it is consistent with the wiring guidelines presented in Section 2. Make sure all connections are tight. Turn on the controller and plug in the 1313 programmer.

① Motor encoder *(see page 48)*

Set the Encoder Steps parameter to the correct setting for your motor's position encoder. This information is typically available from the motor manufacturer. If the AC Motor Characterization Procedure is used, it can determine the encoder steps (but only for encoders with 32, 64, or 80 ppm).

CAUTION

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

② **Motor temperature sensor** *(see page 49)*

Set the Sensor Type parameter to the predefined type (1–5) that corresponds to your motor temperature sensor. Typically, the motor temperature sensor will be a thermistor that should be connected from Analog 2 (pin 8) to ground (pin 7) as shown on page 10.

To check whether the parameter settings and the motor thermistor connections yield the correct motor temperature, read the Temperature value displayed in the 1313's Monitor»Motor menu (page 68). This is typically done when the motor has not been run for many hours, to ensure the motor is at a known (room) temperature. If the 1313 does not display the correct motor temperature, contact your Curtis customer support engineer for help. If the correct motor temperature is not displayed, or if there is no motor temperature sensor, this setup procedure can continue only if the Sensor Enable is set to Off.

If the 1313 displays the correct motor temperature, continue with the procedure and set up the Sensor Enable, Temperature Hot, and Temperature Max parameters.

③ **Current limits** *(see page 36)*

The Drive, Regen, Brake, EMR, Interlock, and DC Pump Current Limit parameters are a percentage of the controller's full rated current. The controller's full rated current is printed on the label of the controller. Set the six current limit parameters to your desired values.

④ **Battery** *(see page 55)*

Set the Nominal Voltage parameter to match the nominal battery pack voltage of your system.

⑤ **Main contactor** *(see page 44)*

Set up the parameters in the Main Contactor menu.

⑥ **EM brake** *(see pages 42–43)*

Set up the parameters in the EM Brake Control menu.

⑦ **Throttle** *(see pages 11–15 and 39–41)*

Before the throttle can be set up the interlock must be verified as Off, by reading the Interlock value displayed in the Monitor»Inputs menu (page 65). If the 1313 indicates the interlock is On, review how you set the Interlock Type parameter (Main Contactor menu) and turn the interlock off. Verify that the

1313 displays that the interlock is now Off. Contact your Curtis customer support engineer to resolve any issues about the interlock before continuing with the setup procedure.

Once you have verified the interlock is off, you can set up the throttle input. The Throttle Type parameter must be set to match the type of throttle (1–5) and wiring that you are using, as described on pages 11–15. Adjust the Forward Deadband, Forward Max, Reverse Deadband and Reverse Max parameters to match the range of your throttle. The Throttle Pot value displayed on the Monitor » Inputs menu (page 64) is useful when setting up these parameters. For the forward and reverse directions, read the displayed throttle pot voltage at the point when the throttle moves out of neutral and at the point just before full throttle and enter these values for the deadband and max settings for that direction. Set up the other parameters in the Throttle menu as required by the application.

You will be able to verify that your throttle settings are correct by checking the Mapped Throttle value displayed in the Monitor » Inputs menu (page 64) over the entire range of throttle pot movement. The value displayed for Mapped Throttle should be = 0% through the range of throttle motion that is considered neutral. The displayed Mapped Throttle should be = 100% through the range of motion that is considered forward throttle max and should be = -100% through the range considered reverse throttle max. Contact your Curtis customer support engineer to resolve any issues about the throttle setup before continuing with the setup procedure.

⑧ Faults *(see Section 8)*

Turn the KSI input Off and then On (to clear any parameter change faults) and use the 1313 to check for faults in the controller. All faults must be cleared before continuing with the setup procedure. Use Section 8 for help in troubleshooting. Contact your Curtis customer support engineer to resolve any issues about the faults before continuing with the setup procedure.

⑨ Setting encoder direction and direction of rotation *(see page 48)*

With the vehicle drive wheels still jacked up, no faults present in the controller, the interlock Off (as verified in the Monitor » Inputs menu, page 65), and both the throttle and brake in neutral (Mapped Throttle = 0% and Mapped Brake = 0% in the Monitor » Inputs menu), the encoder direction can be checked. Use the Monitor » Motor menu (page 68) to view the Motor RPM display. Turn the motor by hand and observe the sign of Motor RPM. Positive is forward and negative is 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 Encoder Direction parameter is correct and should not be changed. If you are getting negative Motor RPM when rotating the motor forward, the Swap Encoder Direction parameter must be changed. Cycle KSI power and repeat the procedure until you are satisfied that the Swap Encoder Direction setting is correctly set. If the vehicle will use the emergency reverse feature, the reverse direction (negative Motor RPM) must be correctly selected so that when the Emergency reverse input is active the motor will rotate in the reverse direction. Contact your Curtis customer support engineer to resolve any issues about encoder direction or emergency reverse before continuing with the setup procedure.

Now that you have the encoder direction set correctly, you can test to see which direction the motor will spin due to how the three phase cables (**U**, **V**, and **W**) are connected to the motor.

Cycle KSI input Off and then On (to clear any parameter change faults) and use the 1313 to check for faults in the controller. All faults must be cleared before continuing with the setup procedure.

Apply the interlock input and verify that the interlock = On (as verified in the Monitor » Inputs menu).

Then, while keeping the brake in neutral, select a direction and apply throttle. The motor should begin to turn, but it may turn in the wrong direction. Observe the direction of rotation of the motor and if it is turning in the wrong direction return the throttle to neutral, and change the setting of the Swap Two Phases parameter. Cycle power, turn on interlock, and turn on direction. Apply throttle and verify that the direction of rotation of the motor matches the direction input. If the motor is turning in the correct direction but appears to be "fighting itself" (struggling at full current while jerkily turning at very low speed), change the direction of the Swap Encoder Direction parameter. If the motor still does not respond properly you should contact your Curtis customer support engineer to resolve any issues about encoder direction or emergency reverse before continuing with the setup procedure.

CAUTION

Do not take the vehicle down off the blocks until the motor is responding properly.

Once the motor is responding properly, lower the vehicle to put the drive wheels on the ground.

⑩ **Motor characterization** *(refer to procedure sent by Curtis)*

If you will be conducting the AC Motor Characterization Procedure, do it now. This procedure will determine the values you should set for the Motor Type, FW Base Speed, and Field Weakening parameters (see Motor Control menus, page 62). Note: If you obtained these values and set them before starting the Initial Setup, skip this step.

⑪ **Emergency reverse** *(see page 59)*

Set up the parameters in the Emergency Reverse menu.

⑫ **Interlock braking** *(see page 60)*

Set up the parameters in the Interlock Braking menu.

Setting up the hydraulic system

Before beginning the setup procedures for the hydraulics,

- Double-check the hydraulic system wiring to ensure it is consistent with the wiring guidelines presented in Section 2.
- Confirm that the hydraulic system is consistent with with the system diagram shown in either Figure 15 or 16.
- Make sure all electrical and hydraulic connections are tight, and the hydraulic fluid filled to the appropriate level.

Ⓗ-1 **Hydraulic throttle inputs** *(see pages 50–54)*

For simple digital (on/off) inputs using the Lift and Lower switches, set the Lift Switch Only Enable and/or Lower Switch Only Enable parameters to On. If both are set On, the hydraulic throttle is not used and the Hyd Throttle Type should be set to 5.

For variable lift or lower, the hydraulic throttle must be set up. Still with the interlock verified to be off, the Hyd Throttle Type must be set up to match the type of throttle (1–5) and wiring that you are using, as described in the Hydraulic Throttle menu and on pages 11–15.

Adjust the Lift Deadband, Lift Max, Lower Deadband and Lower Max parameters to match the range of your hydraulic throttle. The Pot2 Raw value displayed on the Monitor»Inputs menu (page 64) is useful when setting up these parameters. For lift and lower, read the displayed Pot2 Raw voltage at the point when the throttle moves out of neutral and at the point just before full throttle and enter these values for the deadband and max settings for that direction. Set up the other parameters in the Hydraulic Throttle menu as required by the application.

You will be able to verify that your hydraulic throttle settings are correct by checking the Mapped Hyd Throttle value displayed in the Monitor»Inputs menu (page 64) over the entire range of throttle movement. The value displayed for Mapped Hyd Throttle should be = 0% through the range of throttle motion that is considered neutral. The displayed Mapped Hyd Throttle should be = 100% through the range of motion that is considered Lift throttle max and should be = -100% through the range considered Lower throttle max. Contact your Curtis customer support engineer to resolve any issues about the throttle setup before continuing with the setup procedure.

Ⓗ-2 **Load hold valve** *(see page 51)*

If your application uses a hydraulic load hold valve, set the Load Hold Enable parameter On; otherwise set it Off.

Ⓗ-3 **Proportional lowering valve** *(see page 46)*

If your application uses a proportional lowering valve, set the PD Max Current, PD Min Current, PD Dither %, and PD Dither Period parameters based on the valve manufacturer’s ratings.

Ⓗ-4 **Adjustments** *(see pages 46 and 51–52)*

Test the hydraulic system and adjust the Pump Max PWM and PD Max Current (if a proportional valve is used) to give the desired performance.

To further tune the Lift response, adjust the Pump Accel Rate and Pump Decel Rate.

To further tune the Lower response, adjust the Lower Accel Rate and Lower Decel Rate.

If a bump is felt at the end of Lift or Lower operation, increase the Load Hold Delay value to allow the hydraulic fluid to stop flowing before the load hold valve closes.

Set the Lift BDI Lockout and Hyd Inhibit Type (HPD) parameters as required by the application.

6

TUNING GUIDE

Many aspects of vehicle performance can be optimized, using the wide variety of adjustable parameters available to the 1298 controllers. Once a vehicle/motor/controller combination has been tuned, the parameter values can be made 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 tuned again to provide optimum performance.

Selecting the control mode *(see page 25)*

Before starting to tune your vehicle's performance, you must select which control mode you use. Set the Control Mode Select parameter = 0 (Speed Mode Express) or = 1 (Speed Mode) or = 2 (Torque Mode). Cycle KSI input Off and then On (to clear any parameter change faults) and use the 1313 to check for faults in the controller. Then proceed to the tuning steps for the control mode you have selected.

Conduct the steps in the sequence given, because successive steps build upon the ones before. It is important that the effect of these programmable parameters be understood in order to take full advantage of the 1298's powerful features. Please refer to the descriptions in Section 3 if there is any question about what any of the parameters do.

0 - SPEED MODE EXPRESS tuning *(see page 26)*

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 your control mode to Speed Mode (i.e., set Control Mode Select =1).

a. Adjust Max Speed to the maximum speed the motor should turn in the vehicle application; this speed setting corresponds to an input of full throttle.

b. Adjust Typical Max Speed (page 48) 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 you set the Typical Max Speed parameter you should not readjust it without adjusting all the rate parameters as well.

c. 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 “c” in the Speed Mode tuning section.

d. 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.).

e. 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.).

1 - SPEED MODE tuning *(see pages 27–30)*

a. Adjust Max Speed to the maximum speed the motor should turn in the vehicle application; this speed setting corresponds to an input of full throttle.

b. Adjust the Typical Max Speed (page 48) 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 you set the Typical Max Speed parameter you should not readjust it without adjusting all the rate parameters as well.

c. 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 you want 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 (default) settings so you can return them to these original values at the end of this procedure.
  - * If your vehicle has an EM Brake, set the Brake Type (page 42) to 1. This setting will release the EM Brake as soon as interlock is asserted.
  - * 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.

* In the Speed Mode » Speed Controller menu, set the Max Speed to low value (≈1000 rpm), as high speed operation is not needed to observe system response.
* Set Soft Stop Speed parameter to 0 rpm to disable the soft stop speed function (see Restraint menu, page 35).

- 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 in the 1313 by about one third (i.e., final Kp = marginal stability Kp * 2/3).
- If you will be using Speed Mode Express, enter this Kp value for the Kp parameter in the Speed Mode Express menu.
- Set the Brake Type, Accel/Decel Rates, Max Speed, and Soft Stop Speed back to their original values.

d. In the Speed Mode » Response menu, adjust the five 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.).

e. In the Speed Mode » Response menu, adjust the remaining three 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 three brake rates until you are satisfied with the response when brake is applied.

f. 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 *(see pages 31–34)*

a. Set Max Speed to the maximum speed you want to limit the motor to.

b. Kp, Ki, and Kd 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.

c. Set Typical Max Speed (page 48) to the expected maximum speed of the motor.

d. Adjust the parameters shown in Figure 9 (page 34) to set up the throttle mapping:

- Regen Current Limit *(Current Limits menu, page 36)*
- Drive Current Limit *(Current Limits menu, page 36)*
- Restraint Forward, Restraint Back *(Restraint menu, page 35)*
- Neutral Braking *(Torque Mode » Response menu, page 32)*
- Neutral Taper Speed *(Torque Mode » Response menu, page 32)*
- Creep Torque *(Torque Mode » Response » Fine Tuning menu, page 33)*.

e. In the Torque Mode » Response menu, adjust the four accel, decel, 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.).

f. The other parameters in the Torque Mode » Response » Fine Tuning menu typically may need to be changed for some applications. Read the parameter descriptions and adjust as necessary.

7 VEHICLE CONTROL LANGUAGE (VCL)

The 1298 controller has a built-in programmable logic controller with application-specific functions. VCL (Vehicle Control Language) software provides a way to implement unique and complex vehicle control functions.

VCL is a simple programming language that will feel very familiar to anyone who has worked with BASIC, Pascal, or C. Working with VCL requires the installation of the WinVCL program onto a PC. WinVCL will compile VCL programs and flash download the software into the controller through the computer's serial port. The install process for WinVCL will also install two important manuals on your PC—the VCL Programmer's Guide and the VCL Common Functions Manual. These two manuals, which are in PDF format, include more detailed information about VCL than is included here.

This section of the manual summarizes VCL and also describes aspects and functions of VCL that are specific to the 1298 controller. For a more complete understanding of the functions and capabilities of VCL, see the the WinVCL User's Guide, VCL Programmer's Guide, and VCL Common Functions Manual.

Summary of VCL Basics

- VCL is not case-sensitive:
  put_pwm(), Put_PWM(), and PUT_PWM() are identical.

- Spaces in variable names are not allowed in VCL; use underscores in place of spaces.
  Example: Forward_Offset is the VCL name for the 1313 parameter Forward Offset.

- Functions are followed by parentheses; for example:
  Reset_Controller() is a function
  Reset_Voltage is a variable.

- Logical statements must be inside parentheses; examples:
  IF (setpoint >50)
  ELSE IF ((setpoint <20) & (temperature >100)).

- Comments are preceded by semicolons.

The VCL functions described in the VCL Common Functions Manual are available on 1298 controllers. The 1298 also has these additional functions:

These functions, which are not included in the VCL Common Functions Manual, are described at the end of Section 7.

VARIABLE TYPES

VCL provides dedicated space in which to store custom variables. There are four types of variables, based on their type of storage: volatile storage (RAM) and three types of non-volatile storage (EEPROM) are available.

RAM variables are stored only while power is on; they are lost at power-down. They must be initialized on power-up by explicit VCL assignments (i.e., User1 = 12).

NVUser1–15 EEPROM variables are 15 variables stored at power-down and recalled by the operating system when the NVM_NVUser_Restore function is used. Thus, they can then be recalled at the next power-on cycle, which restores their previous values. See the section on non-volatile memory access in the VCL Common Functions manual for more information.

Block EEPROM are 38 blocks of 15 variables (total of 570 variables), which are stored and recalled using the functions NVM_Block_Read and NVM_Block_Write. The 38 blocks are called NVM3–NVM40. The read and write functions must point to the RAM variables that the EEPROM blocks should be written from or read to. For example, NVM_Block_Read(NVM10,0,15,User20) will read the 15 variables stored in EEPROM block NVM10 and restore those variables to the 15 variables starting with RAM variable User20 (so the 15 EEPROM variables would be restored to User20–34). See the section on non-volatile memory access in the VCL Common Functions manual for more information.

Parameters EEPROM variables are a special type of EEPROM variable that is intended to be used to create OEM defined 1313 parameters. These 1313 parameters can be defined as 16-bit by using the P_User variables or they can be defined as bit (On/Off) by using the P_UserBit variables. These variables are typically written to EEPROM through the 1313 programmer interface (i.e., when a 1313 user changes a parameter setting using the 1313). They can be used in the VCL code, but changing a P_User (or P_UserBit) value with VCL will only change the variable value in RAM and will **not** change the value in EEPROM. Thus, these variables are intended for creating and defining 1313 parameters only.

TYPE	QUANTITY	RANGE
RAM	120 variables	User1 – User120
NVUser EEPROM	15 variables	NVUser1 – NVUser15
Block EEPROM	38 blocks (15 variables each)	NVM3 – NVM40
Parameters EEPROM	100 variables and 20 variables of 8 bits each (160 bits)	P_User1 – P_User100 P_UserBit1 – P_UserBit20

VCL can modify the 1313 control mode parameters in RAM by using the VCL variable name for the 1313 parameter. For example,

```
Brake_Rate_SpdM = 3000 ;Change Brake Rate to 3.0sec
```

will change the RAM value of the speed control mode's Brake Rate; the new value will be used in determining the Controller Torque Command. However, the value of the stored EE value of this parameter remains unchanged; when the controller is turned off, the RAM value will be lost. The next time the controller is powered back on, the "old" value of Brake Rate will be restored from EE memory. VCL cannot write to the EE memory. The 1313 parameter settings in EE memory can be changed by using the 1313 to change the values in the program menus.

VCL RUNTIME RATES

VCL is an interpreted language. Each line of VCL code is converted (compiled) into a set of codes and then flash loaded into the controller. The controller interprets these codes one line at a time while the system is powered up. Here are the processing rates of the various functions:

FUNCTION	FUNCTION FULL NAME	INSTANCES	SERVICE RATE
ABS	Absolute Value	2	4 ms
ADC	Analog to Digital Converter Input	2	1 ms
CAN	CAN Communications	15	4 ms
CPY	Copy	8	4 ms
DLY	Delay	32	1 ms
FLT	Filter	4	1 ms
LIM	Limit	4	4 ms
MAP	Map	4	4 ms
MTD	Multiply then Divide	4	4 ms
NVM	Non-Volatile Memory	38	2 ms
PID	Proportional Integral Derivative	2	4 ms
POT	Potentiometer Input	2	8 ms
PWM	Pulse Width Modulated Output	6	4 ms
RMP	Ramp	4	1 ms
SCL	Scaling	4	4 ms
SEL	Selector, 2-position switch	8	4 ms
SEL_4P	Selector, 4-position switch	8	32 ms
SW	Switch Input	1*	4 ms
TMR	Timers (hourmeters)	3	1 ms

* There is only one Switch variable; it has 16 associated bit-variables.

I/O CONTROL WITH VCL

Digital Inputs

The 1298 controller has a total of 16 digital inputs. Eight are switch inputs (Sw_1 through Sw_8, plus Sw_16). These switch inputs are shown on the standard wiring diagram (Figure 3, page 10). The remaining seven digital inputs are less obvious: one on each driver and digital output (Sw_9 through Sw_15). These can be used as digital inputs or to sense the state of the output or its wiring (e.g., open coil check).

To address a digital input in a VCL program, use the desired input label (Sw_1 through Sw_16). You must use On or Off in the code when determining a switch state; using true/false or 1/0 will give erroneous results.

```
if   (Sw_1 = ON)
     {
     ;put code here to run when switch 1 is On
     }
if  (Sw_16 = OFF)
     {
     ;put code here to run when switch 16 is Off
     }
```

All switch inputs are automatically debounced by the VCL operating system. This prevents noisy contacts or contact bounce from causing erroneous events in your VCL code. The debounce time can be varied from 0 to 32 milliseconds in 4ms steps, using this function:

```
Setup_Switches(5); 20 milliseconds
```

If this line is not in the VCL code, the debounce time is set at 16 ms.

Driver and Digital Outputs

There are five driver outputs (PWM1 through PWM5) and two digital outputs (DigOut6 and DigOut7). These outputs have variations in current and frequency range. For their specifications, see "digital outputs" on page 17.

The driver outputs have high current FET output stages and can be pulse width modulated (PWM) to vary the average output to inductive loads such as contactors and relays. This is useful when the battery voltage needs to be brought down for lower voltage coils. The two digital outputs are 1-amp drivers that are only On or Off.

Drivers use a special VCL function to set their PWM level. This PWM level can be set up in a signal chain to update automatically or can be set directly in the main loop. PWM can be set from 0–100% using the digital range of 0 to 32767.

```
Put_PWM(PWM2,16384)
```

will output a 50% waveform on Driver 2.

```
Automate_PWM(PWM2,@user1)
```

will continually update the Driver 2 output with the present value of variable User1. This automate statement needs only to be run once, usually in the initialization section of the VCL program. VCL can monitor the present value of a PWM driver: the variable PWMx_Output (where "x" is the PWM channel number) is automatically filled with the present value of the driver output.

The proportional driver (Driver 5) is different from Drivers 1–4. It can be controlled in two ways: with the proportional driver processing function (see Figure 20, page 100) or with the VCL Put_PWM() function. The VCL statement Put_PWM(PWM5, 16383) will result in a 50% PWM output on pin 2 only if the parameter PD Enable is set to Off. See page 94 for more information on interfacing the proportional driver.

Control of the two digital outputs (Digital Outputs 6 and 7) is done using the VCL functions Set_Digout() and Clear_Digout().

```
Set_DigOut(DigOut6)
```

will set Digital Output 6 On (active). VCL can monitor the present value of a digital output driver: the bit variable Digx_Output (where "x" is the digital output channel number) is automatically filled with the present value of the driver output (On or Off).

It is important to note that all outputs are active Low. With 100% PWM or an output of "On," the FET or transistor will be pulling hard to ground. A DVM on the output will measure near 0 volts.

Potentiometer Inputs

The 1298 controller has two potentiometer inputs, which are typically used for a traction motor throttle and a pump motor throttle. Many features (mapping, acceleration rates, etc.) are built in as 1313 parameters. Still, there are times that these potentiometer inputs may be needed for other functions such as steering angle or height sensing, or simply as data inputs.

The standard way to input pot information is to set the 1313 parameter Throttle Type (or Hyd Throttle Type) to an appropriate value of 1–4 as shown on pages 11–14. When set to a value of 1–4, the resulting signal chain can operate without the use of any VCL.

However, if an OEM wishes to control the throttle (or hydraulic throttle) signal chain in VCL or use either of the throttle inputs for signals that are not throttle signals, then the 1313 parameter Throttle Type (or Hyd Throttle Type) should be set to a value of 5. Setting the 1313 parameter Throttle Type (or Hyd Throttle Type) to a value of 5 changes the routing of the appropriate signal chain (drive throttle or hydraulic throttle) and allows the VCL programmer access to the Throttle_Pot_Output or Brake_Pot_Output variables; see Figure 17, page 93. (Note that in VCL, the hydraulic throttle output is named Brake_Pot_Output.)

When the Throttle Type = 5, the Throttle_Pot_Output is a VCL variable that the OS will update with the current value of the throttle pot input. Similarly, when the Hyd Throttle Type = 5, the Brake_Pot_Output is a VCL variable that the OS will update with the current value of the hydraulic throttle pot input. However, the value of the Throttle_Pot_Output (or Brake_Pot_Output) will remain clamped to zero until the VCL function Setup_Pot() is executed.

Typically the Setup_Pot() function is executed at the beginning of a VCL program to define the potentiometer input connection as THREE_WIRE (uses Pot High and Pot Low connections), TWO_WIRE (variable resistor, or rheostat, uses Pot Low but no connection to Pot High), or ONE_WIRE (a voltage input, no connection to either Pot High or Pot Low). THREE_WIRE potentiometer connections are the same as the 3-wire potentiometer connections shown on page 13 for a Throttle Type 2. TWO_WIRE potentiometer connections are the same as the 2-wire potentiometer connections shown on page 12 for a Throttle Type 1. ONE_WIRE potentiometer connections are the same as the Voltage Source or Current Source connections shown on page 13 for a Throttle Type 2.

Note that the Setup_Pot() function will only work (and is only needed) if the corresponding Type is set to 5 (Throttle Type = 5 or Hyd Throttle Type = 5).

```
Setup_Pot(THROTTLE_POT,THREE_WIRE)
```

will set up the throttle pot input for wiring using all three connections (pins 15, 16, 18).

To set up the hydraulic throttle pot input for use in VCL, use the Brake_Pot constant in place of the Thottle_Pot constant in the Setup_Pot function.

```
Setup_Pot(BRAKE_POT,TWO_WIRE)
```

will set up the hydraulic throttle pot input for wiring using two connections (pins 17, 18).

The 0–100% position of the potentiometer is represented by a value from 0–32767 in VCL. Once set up (through the VCL Setup_Pot() function) the potentiometer value is automatically and continuously loaded into the variable Throttle_Pot_Output or Brake_Pot_Output. It is important to use the correct setup (ONE_WIRE, TWO_WIRE, or THREE_WIRE) since the input is automatically re-scaled for 0–100% based on the wiring used; for example, the voltage at the Pot Low pin is automatically subtracted and re-scaled on a THREE_WIRE pot.

Another effect of setting Throttle Type = 5 is that the signal chain for the throttle now gets its input from a different source. The input to the throttle chain is now a VCL variable called VCL_Throttle instead of the throttle pot. Similarly, Hyd Throttle Type = 5 means that the hydraulic throttle signal chain will get its input from a VCL variable called VCL_Hyd_Throttle rather than from the hydraulic throttle pot. The VCL_Throttle and VCL_Hyd_Throttle variables will need to be controlled in the VCL program.

One of the unique features of the potentiometer inputs (as opposed to the analog inputs) is that they have automatic pot fault detection functions running in the motor controller OS. The VCL programmer has access to the pot detection functions with the Setup_Pot_Faults() function. With this function, VCL can set the high and low threshold at which a fault occurs. This function also forces the pot value to a definable level if a fault occurs. Note that the Setup_Pot_Faults() function will work for all throttle Types (1–5). See page 111 for more detail on this function.

Analog Inputs

These controllers have two generic analog inputs (pins 24 and 8). These are shared as switch inputs 1 and 2 (Sw_1, Sw_2). The values of the analog inputs are automatically placed in VCL variables Analog1_Input and Analog2_Input every 1 millisecond. Scaling is 0–10V = 0–1023.

```
User2 = Analog2_Input
```

will fill the User2 RAM variable with the value of the voltage at pin 8.

The filtered values of the analog inputs are also available and are automatically placed in VCL variables Analog1_Filtered and Analog2_Filtered. Scaling is 0–10V = 0–1023. The default filter value is 328 (10 Hz) and can be changed in VCL by changing the Analog1_Filter and Analog2_Filter values. Scaling is 0–999Hz = 0–32767.

Analog Output

This controller has one analog output (pin 30). This output is a special driver output. The switching stage is filtered to provide a smooth average voltage, instead of the actual PWM waveform seen on Drivers 1–5. However, AnalogOut uses the same Put_PWM() and Automate_PWM() used by these other drivers. The scaling is 0–10V = 0–32767.

```
Put_PWM(PWM6,6553)
```

will generate 2.0 volts at the analog output. VCL can monitor this output using the variable Analog_Output.

INTERFACING THE THROTTLE AND BRAKE COMMANDS

VCL can interface and modify the throttle, brake, and hydraulic throttle signals at several points; it can be used to create a completely unique command, adjust parameters to provide MultiMode, or modify the command based on steering angle, height, etc.

These signal chains within the controller are sophisticated and flexible. Before applying VCL to modify these chains, it is important to fully understand the ramifications of these changes. The AC motor command diagram is presented in Figure 17, and the hydraulic command diagram in Figure 19.

Throttle Processing (Fig. 17)

The throttle signal chain flows left to right starting with the physical throttle pot. The voltage on the throttle wiper input (pin 16) is input into the controller and has the VCL variable name Throttle_Pot_Raw which is displayed in the 1313 Monitor»Inputs menu. This throttle signal is then modified by the Throttle Type Processing and Throttle Mapping blocks.

The Throttle Type Processing block combines the Throttle Type parameter (see page 39) and the throttle potentiometer input (Throttle_Pot_Raw) to create a 16-bit variable containing the magnitude of the raw command. This raw command passes to the Throttle Mapping block, which re-shapes the throttle signal magnitude and direction based on the various Throttle menu parameters (see pages 39–41) and the direction inputs.

Following the Throttle Mapping block are two switches whose purpose is to give the throttle signal a small value (1 for the forward switch, and -1 for the reverse switch) to indicate that a direction switch is On—but only if the throttle signal output from the Throttle Mapping block is = 0.

The signal then passes through a selector switch. If the Throttle_Type parameter is set to 5 (Throttle Type = VCL input, see page 39), the Throttle Mapping block output signal is ignored and the command comes from the VCL variable VCL_Throttle. The VCL program manipulates the VCL_Throttle variable to get a throttle command. When the Throttle Type is set to 1–4, the variable VCL_Throttle does nothing, and the Throttle Mapping block output signal passes through.

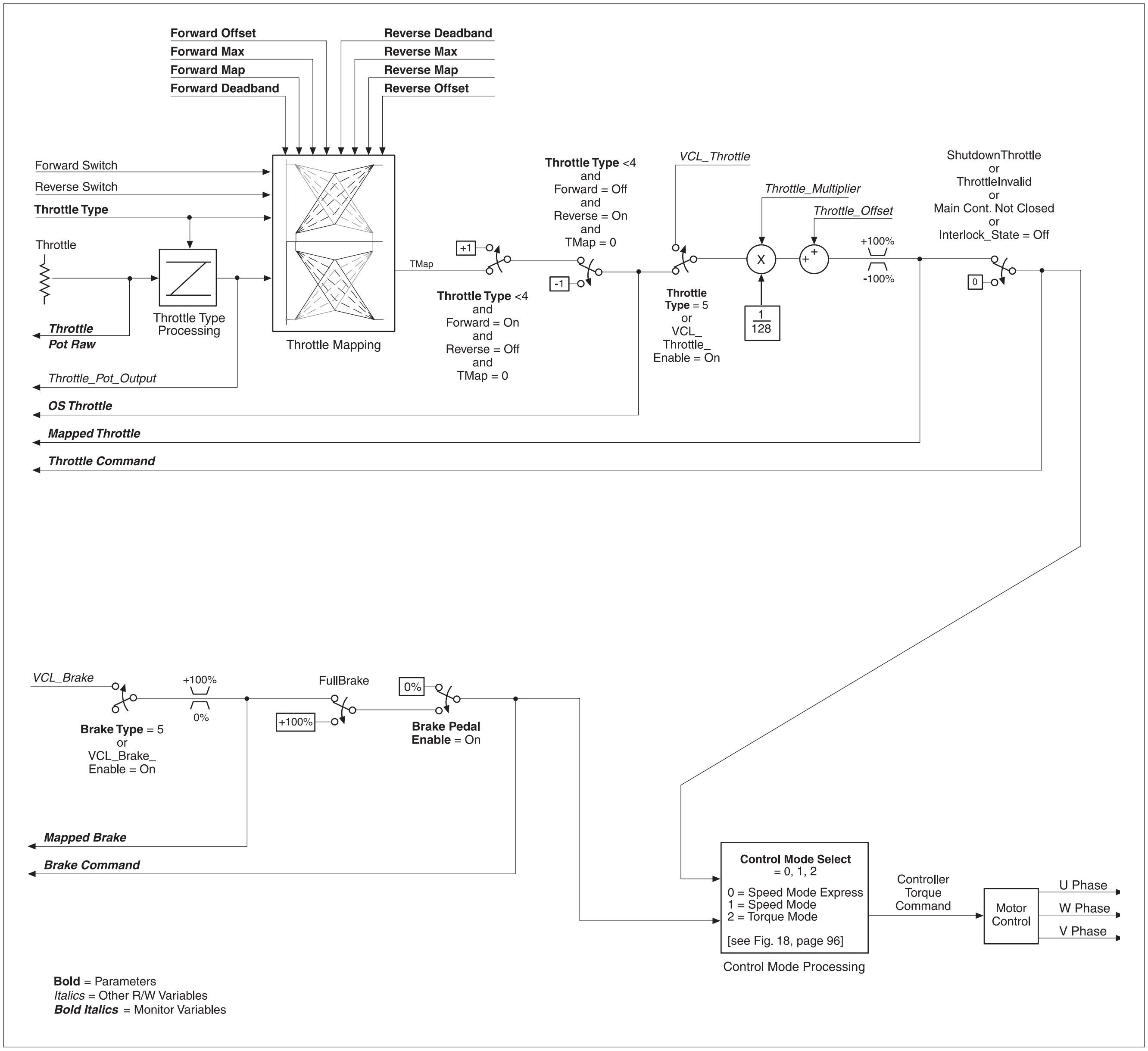


Fig. 17 *Motor command diagram, AC traction motor.*

After the "Throttle Type = 5" switch, the throttle signal is modified by the multiplying and summing nodes. These nodes can be adjusted by VCL through the variables Throttle_Multiplier and Throttle_Offset. This is the basic input point for creating functions like MultiMode, dual drive algorithms, and height vs. speed control. Note that the throttle multiplier has a built-in "divide by 128." This allows the VCL to either multiply (Throttle_Multiplier > 128) or divide (Throttle_Multiplier < 128) the nominal throttle value. Typically the default multiplier is set to 128, thus having no net effect. Both Throttle_Multiplier and Throttle_Offset can be positive or negative.

The output of the multiplying and summing nodes is a VCL variable called Mapped_Throttle, which is displayed in the 1313 Monitor » Input menu. Checking the value of Mapped_Throttle using the 1313 is a good way to see if your Throttle menu parameters are set correctly. A VCL program can control the throttle by changing the variables VCL_Throttle (only if Throttle Type = 5), Throttle_Multiplier, and Throttle_Offset. The effect of these variables can be observed as Mapped_Throttle in the 1313 Monitor » Inputs menu.

The throttle signal continues to a selector switch that will set the throttle signal = 0% if any of the following conditions is present: Interlock_State = Off (see page 65), a fault has set throttle request = 0% (see the Troubleshooting Chart, Table 5), or if Main_State ≠ 5 or 10 (see page 69).

After this selector switch the throttle signal is a VCL variable called Throttle_Command, which is displayed in the 1313 Monitor » Inputs menu. Throttle_Command is the final value of the throttle signal chain that is input to the Control Mode Processing block; see Figure 18. Checking the value of Throttle_Command using the 1313 is a good way to see the final throttle signal. If ABS(Throttle_Command) >1 count, the motor controller will output signals to the motor to make it spin.

For investigating why a motor is not spinning, it is useful to use the 1313 to check the state of the throttle signal from beginning to end: using Throttle_Pot_Raw, Mapped_Throttle, and Throttle_Command. Once these values are known, the Motor Command Diagram (Figure 17) can be used to find how that signal progressed from input to final value.

The following throttle processing variables are accessible by VCL:

VCL VARIABLE	ACCESS	DESCRIPTION
Throttle_Pot_Raw	Read Only	Voltage measurement at pin 16, scaled for the proper wiring
Throttle_Pot_Output	Read Only	Throttle pot input value after being scaled for the proper wiring; for use in VCL program when Throttle Type = 5
OS_Throttle	Read Only	Throttle pot value after mapping, to be used in VCL when VCL Throttle Enable = On and Throttle Type = 1–4
Mapped_Throttle	Read Only	Throttle pot value after mapping
VCL_Throttle	Read/Write	VCL-accessible throttle command
Throttle_Multiplier	Read/Write	Multiplies or divides the throttle signal
Throttle_Offset	Read/Write	Provides a +/- offset to the throttle signal
Throttle_Command	Read Only	Command resulting from throttle processing

Brake Processing (Fig. 17)

Braking can be programmed in the 1298 through VCL, as shown in the lower portion of Figure 17, rather than through a physical brake pot. The VCL program manipulates the VCL_Brake variable to get a brake command. Custom braking functions can be set up in this fashion; e.g., braking based on a switch position or an internal fault.

After the initial switch, 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 is displayed in the 1313 Monitor Inputs menu.

The brake signal then goes through a second selector switch that will set the brake signal = 0% if the Brake Pedal Enable parameter is set Off. If set On, the brake signal will pass through. The brake signal after this second selector switch is a VCL variable called Brake_Command, which is displayed in the 1313 Monitor Inputs menu. Brake command is the final value of the brake signal chain that is input to the Control Mode Processing block; see Figure 18. If Brake_Command is non-zero, the throttle signal will be set to 0%.

The following brake processing variables are accessible by VCL:

VCL VARIABLE	ACCESS	DESCRIPTION
VCL_Brake	Read/Write	VCL-accessible brake command
Mapped_Brake	Read Only	Brake signal value after mapping
Brake_Command	Read Only	Command resulting from brake processing

Control Mode Processing (Fig. 18) **and Final AC Motor Control**

Figure 18 begins with the Throttle_Command input. The signal chain is then directed to Speed Mode Express or Speed Mode or Torque Mode, based on Control Mode Select.

The control mode function uses algorithms to convert the incoming throttle signal and the motor rpm input into a Controller Torque Command.

The selected control mode calculates the desired Controller Torque Command, which is passed to the Motor Control block (see Figure 17). The Motor Control block uses its mathematical model of the specific AC induction motor used to generate the high efficiency three-phase outputs that are output to the AC motor via the cables connected to the **U**, **V**, and **W** terminals.

Note: In Speed Mode Express it is possible to include some of the Speed Mode parameters. You can use any Speed Mode parameters except those with the same names as their Speed Mode Express counterparts (for example, Max Speed) or with names including the words Accel Rate, Decel Rate, or Brake Rate.

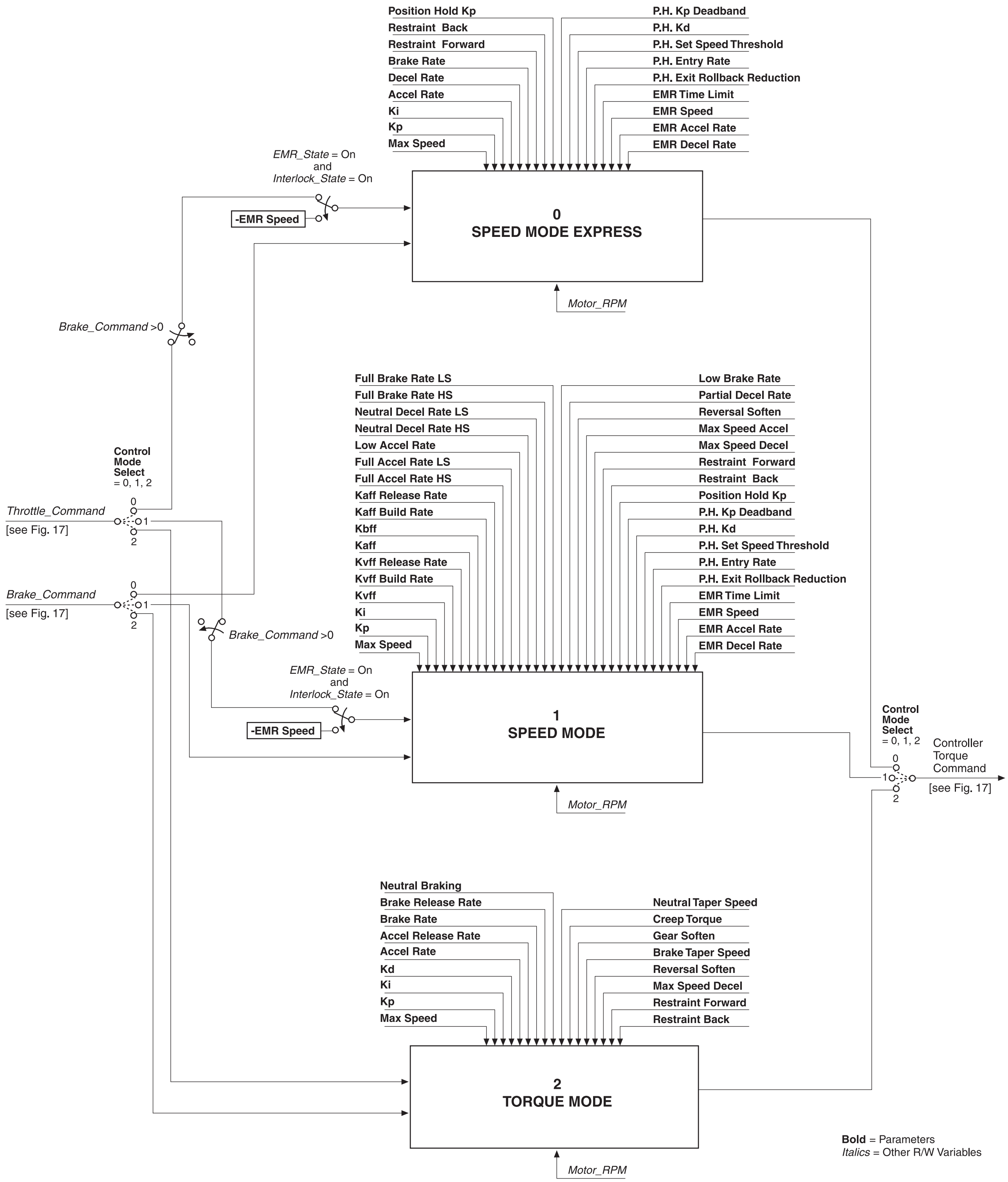


Fig. 18 *Control Mode processing.*

Hydraulic Throttle Processing (Fig. 19)

Hydraulic throttle processing is the throttle chain that controls the commands to the lowering valve and the pump motor. Hydraulic throttle processing is very flexible and can be done with or without the hydraulic throttle pot or VCL, as shown in Figure 19. The hydraulic throttle signal chain flows from left to right starting with the physical hydraulic throttle pot. The voltage on the wiper input (pin 17) is input into the controller and has the VCL variable name Pot2_Raw which is displayed in the 1313 Monitor»Inputs menu. The Throttle Type Processing block then combines the Hyd Throttle Type

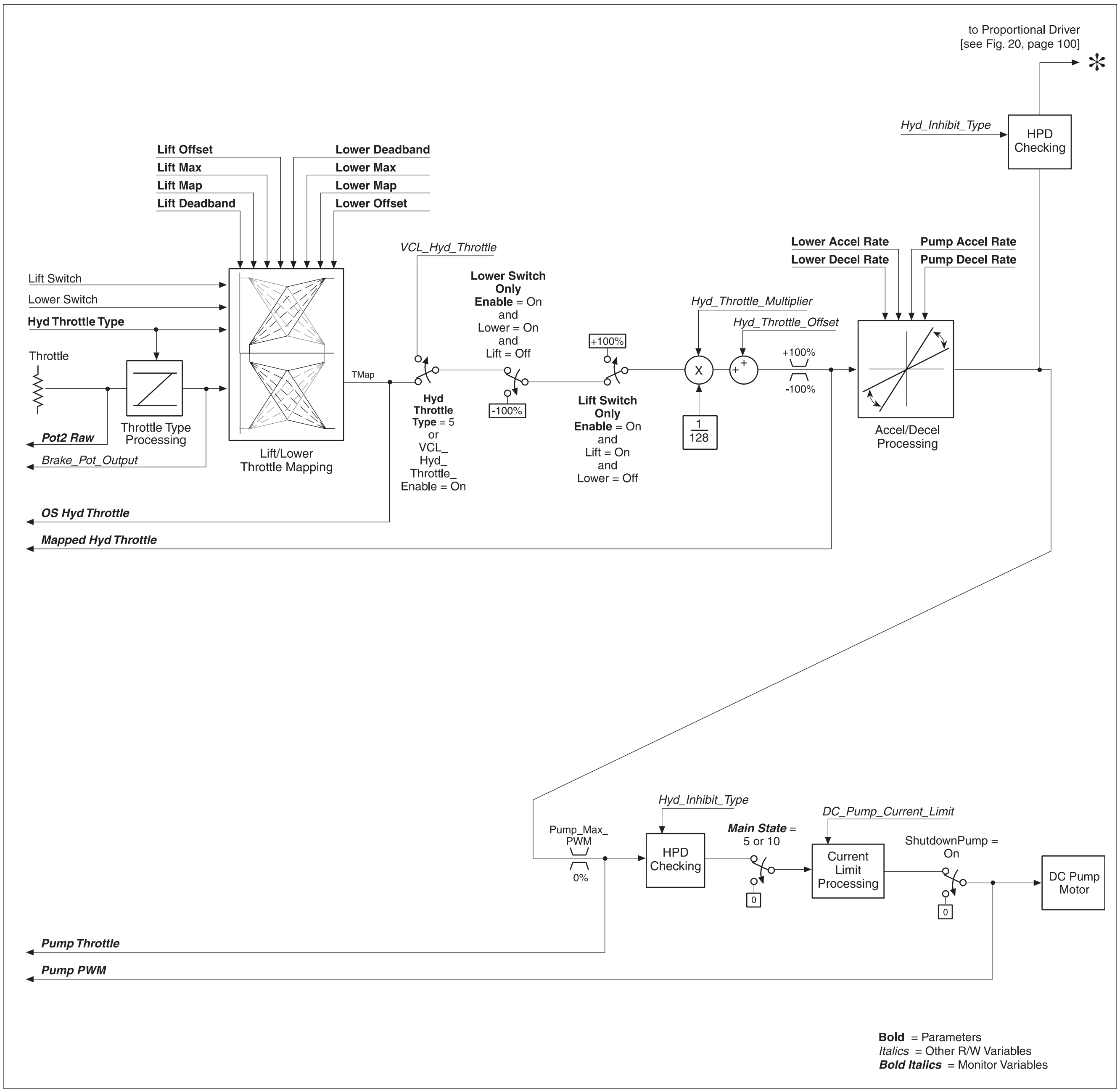


Fig. 19 *Hydraulic command diagram.*

parameter and the hydraulic throttle pot input (Pot2_Raw) to create a 16-bit variable containing the magnitude of the raw command, which has the VCL name Brake_Pot_Output. This signal and the Lift and Lower switch inputs go to the Lift/Lower Throttle Mapping block which re-shapes the signal according to the various Hydraulic Throttle menu parameters (page 53) and outputs the VCL variable OS_Hyd_Throttle.

The signal then passes through a selector switch. If VCL_Hyd_Throttle_Enable = On (or Hyd Throttle Type parameter = 5), the Lift/Lower Throttle Mapping block output signal is not used and the command comes from the VCL variable VCL_Hyd_Throttle. The VCL program can manipulate the VCL_Hyd_Throttle variable to get a command. Custom hydraulic functions can be set up in this fashion: e.g., Lift, Lower, or aux hydraulic functions (reach, sideshift, tilt) based on a switch position or CAN bus communication. The hydraulic throttle pot can still be used with the OS_Hyd_Throttle variable and VCL programming.

The signal then passes through the Lower Switch Only Enable and Lift Switch Only Enable switches. These parameters and Lift and Lower switch inputs can be used for non-variable lift and lower inputs (either On (100%) or Off (0%)) as an alternative to using the variable pot input.

The throttle signal is then modified by the multiplying and summing nodes. These nodes can be adjusted by VCL through the variables Hyd_Throttle_Multiplier and Hyd_Throttle_Offset. Note that the throttle multiplier has a built-in divide by 128. This allows the VCL to either multiply (Hyd_Throttle_Multiplier > 128) or divide (Hyd_Throttle_Multiplier < 128) the nominal throttle value. Typically the default multiplier is set to 128 (thus having no net effect). Both Hyd_Throttle_Multiplier and Hyd_Throttle_Offset can be positive or negative.

The output of the multiplying and summing nodes is a VCL variable called Mapped_Hyd_Throttle, which is displayed in the 1313 Monitor » Inputs menu. Checking the value of Mapped_Hyd_Throttle using the 1313 is a good way to see if your Hydraulic Throttle menu parameters are set correctly. A VCL program can control the throttle by changing the variables VCL_Hyd_Throttle, Hyd_Throttle_Multiplier, and Hyd_Throttle_Offset. The effect of these variables can be observed as Mapped_Hyd_Throttle in the 1313 Monitor » Inputs menu. Mapped_Hyd_Throttle then passes to the Accel/Decel Processing block, which sets the slew rate of the signal according to various Hydraulic menu parameters (page 51).

The signal then gets split in two different directions: negative throttle values drive the lowering valve, and positive values drive the hydraulic pump.

For negative throttle values, the signal goes through an HPD checking block, which checks for HPD conditions according to the setting of the Hyd_Inhibit_Type parameter, and then continues in the signal chain shown in Figure 20 (page 100).

For positive throttle values (following the path of the Hydraulic pump), this signal goes through a limiter that limits the max PWM according to the parameter Pump_Max_PWM. The VCL variable Pump_Throttle is displayed in the 1313 Monitor»Inputs menu. Next the signal goes through the HPD checking block, which checks for HPD conditions according to the setting of the Hyd_Inhibit_Type parameter. The signal will be switched to zero if the main contactor has not been pulled in (Main State = 5) or if the 1298 FET bridge has not been enabled (Main State = 10 and Main Enable = Off). The signal then continues to the Current Limit Processing block and to a final switch block where a fault action (Shutdownpump) can shut down the pump output. The final output to the pump motor is a VCL variable called Pump_Output and is displayed in the 1313 Monitor»Inputs menu as Pump PWM.

The following hydraulic throttle processing variables are accessible by VCL:

VCL VARIABLE	ACCESS	DESCRIPTION
Pot2_Raw	Read Only	Voltage measurement at pin 17, scaled for the proper wiring
Brake_Pot_Output	Read Only	Hydraulic pot input value afterbeing scaled for the proper wiring; for use in VCL program when Hyd Throttle Type = 5
OS_Hyd_Throttle	Read Only	Hydraulic pot value after mapping, to be used in VCL when VCL Hyd Throttle Enable = On and Hyd Throttle Type = 1–4
VCL_Hyd_Throttle	Read/Write	VCL-accessible hydraulic throttle command
Hyd_Throttle_Multiplier	Read/Write	Multiplies or divides the throttle signal
Hyd_Throttle_Offset	Read/Write	Provides a +/- offset to the throttle signal
Mapped_Hyd_Throttle	Read Only	Hydraulic throttle pot value after mapping
Pump_Throttle	Read Only	Pump command resulting from hydraulic throttle processing
Hyd_Inhibit_Type	Read/Write	Decides which type of HPD check is performed on the hydraulic throttle signal
DC_Pump_Current_Limit	Read/Write	Current limit for the DC pump
Pump_Output	Read Only	Final pump command

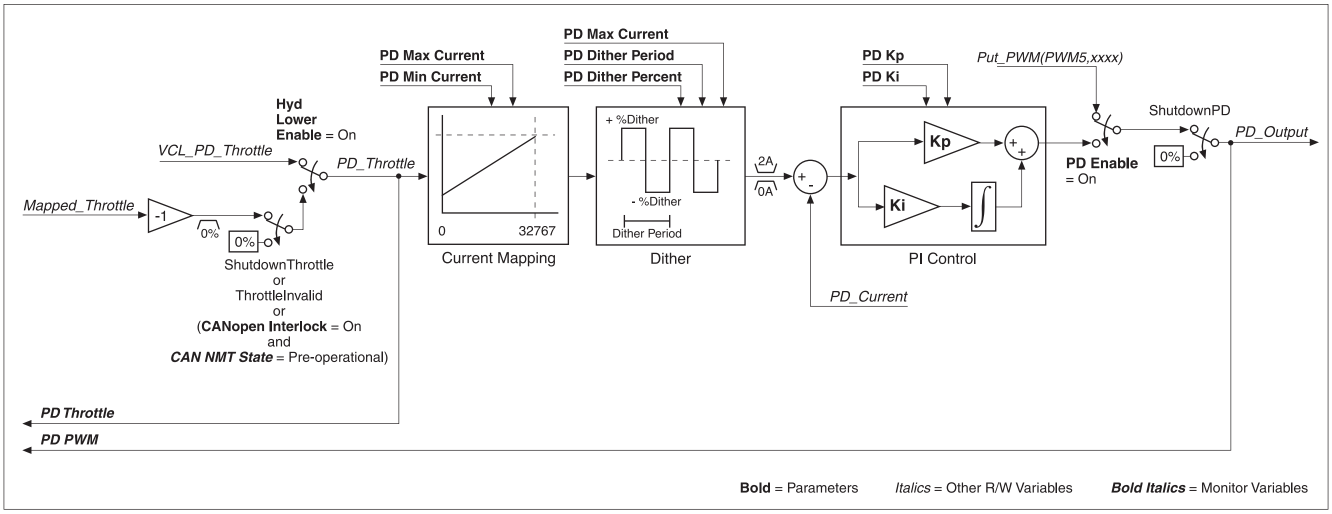


Fig. 20 *Proportional driver processing.*

INTERFACING THE PROPORTIONAL CURRENT DRIVER

In the 1298, the proportional current driver is used for the lowering valve; see Figures 15 and 16, on page 50. For other connections, VCL can directly interface the proportional current driver (PD), as shown in Figure 20. VCL can change the working parameters of the PD and can provide the command.

Depending on how the PD system is to be used, certain parameters need to be set; they can be set via the 1313 handheld programmer or via VCL.

1. PD_Enable must be set On for current control, otherwise the PD_Output will be controlled by the VCL function Put_PWM(PWM5,xxxx), which is voltage control.
2. Hyd_Lower_Enable must be set On to use the throttle input to control lowering in a hydraulic lift/lower system.
3. Hyd_Lower_Enable must be set Off to allow using a VCL variable (VCL_PD_Throttle) as the PD command.

Once the PD parameters are set, the PD_Throttle variable will be mapped between PD_Min_Current and PD_Max_Current and sent to the dither function. Note that Mapped_Throttle is inverted; lowering its value (making it more negative) increases the PD_Throttle value.

The Dither function adds and subtracts from the current command to the PD based on PD_Dither_Percent, at a rate set by PD_Dither_Period.

The dithered current command is compared to the present PD_Current and the error is fed into a PI controller. The feedback gains are set by the parameters PD Kp (proportional gain) and PD Ki (integral gain). The output of the PI controller becomes the driver's pulse width modulation, thus controlling the average current of the driver. This final output is the VCL variable PD_Output, which is displayed in the 1313 Monitor » Inputs menu as PD PWM.

The following PD processing variables are accessible by VCL:

VCL VARIABLE	ACCESS	DESCRIPTION
Mapped_Throttle	Read Only	Command from throttle section
VCL_PD_Throttle	Read/Write	VCL-accessible PD command
PD_Throttle	Read Only	Resultant command to the PD
PD_Current	Read Only	Average current flowing in the PD
PD_Output	Read Only	Resultant PWM at PD output

USING THE FAULT HANDLER IN VCL

The operating system of the controller detects various faults and takes appropriate fault actions to protect the controller. These faults have fault codes that are flashed on the controller status LEDs, and fault text is displayed in the 1313 System Faults and Fault History menus. These operating system faults are covered in Section 8. Additionally, the operating system makes the status of the operating system faults available for use in VCL programs in the form of seven variables called Status1, Status2, Status3, Status4, Status5, Status6, and Status7. Each of these 16-bit variables contains the status of 8 faults in the lower byte (the upper byte is always set to 0). These Status1–7 variables are read only (RO) and can be used in a VCL program to trigger additional fault actions such as sending fault text messages to a display or blinking a dashboard LED.

Here are the bit locations of each of the operating system faults in the Status1–7 variables:

```
Status1
*      Bit0 = Main Contactor Welded (Code 38)
*      Bit1 = Main Contactor Did Not Close (Code 39)
*      Bit2 = Pot Low Overcurrent (Code 45)
*      Bit3 = Throttle Wiper Low (Code 42)
*      Bit4 = Throttle Wiper High (Code 41)
*      Bit5 = Pot2 Wiper Low (Code 44)
*      Bit6 = Pot2 Wiper High (Code 43)
*      Bit7 = EEPROM Failure (Code 46)

Status2
*      Bit0 = HPD/Sequencing Fault (Code 47)
*      Bit1 = Severe Undervoltage (Code 17)
*      Bit2 = Severe Overvoltage (Code 18)
*      Bit3 = Undervoltage Cutback (Code 23)
*      Bit4 = Overvoltage Cutback (Code 24)
*      Bit5 = Not Used
*      Bit6 = Controller Overtemp Cutback (Code 22)
*      Bit7 = Controller Severe Undertemp (Code 15)

Status3
*      Bit0 = Controller Severe Overtemp (Code 16)
*      Bit1 = Coil1 Driver Open/Short (Code 31)
*      Bit2 = Coil2 Driver Open/Short (Code 32)
*      Bit3 = Coil3 Driver Open/Short (Code 33)
```

```
*      Bit4 = Coi14 Driver Open/Short (Code 34)
*      Bit5 = PD Open/Short (Code 35)
*      Bit6 = Main Open/Short (Code 31)
*      Bit7 = EMBrake Open/Short (Code 32)

Status4
*      Bit0 = Precharge Failed (Code 14)
*      Bit1 = Digital Out 6 Overcurrent (Code 26)
*      Bit2 = Digital Out 7 Overcurrent (Code 27)
*      Bit3 = Controller Overcurrent (Code 12)
*      Bit4 = Current Sensor Fault (Code 13)
*      Bit5 = Motor Temp Hot Cutback (Code 28)
*      Bit6 = Parameter Change Fault (Code 49)
*      Bit7 = Motor Open (Code 37)

Status5
*      Bit0 = External Supply Out of Range (Code 69)
*      Bit1 = Motor Temp Sensor Fault (Code 29)
*      Bit2 = VCL Run Time Error (Code 68)
*      Bit3 = +5V Supply Failure (Code 25)
*      Bit4 = OS General (Code 71)
*      Bit5 = PDO Timeout (Code 72)
*      Bit6 = Encoder Fault (Code 36)
*      Bit7 = Stall Detected (Code 73)

Status6
*      Bit0 = Not Used
*      Bit1 = Not Used
*      Bit2 = Emer Rev HPD (Code 47)
*      Bit3 = Not Used
*      Bit4 = Motor Type Fault (Code 89)
*      Bit5 = Not Used
*      Bit6 = Motor Characterization Fault (Code 87)
*      Bit7 = Pump Hardware Fault (Code 97) – model 1298 controllers only

Status7
*      Bit0 = Not Used
*      Bit1 = VCL/OS Mismatch (Code 91)
*      Bit2 = EM Brake Failed to Set (Code 92)
*      Bit3 = Encoder LOS (Limited Operating Strategy) (Code 93)
*      Bit4 = Emer Rev Timeout (Code 94)
*      Bit5 = Dual Severe Fault (Code 75)
*      Bit6 = Fault On Other Traction Controller (Code 74)
*      Bit7 = Illegal Model Number (Code 98)

Status8
*      Bit0 = Pump Overcurrent (Code 95) – model 1298 controllers only
*      Bit1 = Pump BDI (Code 96) – model 1298 controllers only
*      Bit2 = Pump HPD (Code 47) – model 1298 controllers only
*      Bit3 = Dualmotor Parameter Mismatch (Code 99)
*      Bit4 = Not Used
*      Bit5 = Not Used
*      Bit6 = Not Used
*      Bit7 = Not Used
```

The operating system also provides the capability to create OEM-defined custom faults using VCL. Just as with system faults, the VCL fault codes are flashed on the controller Status LEDs and fault text is displayed on the 1313 System Faults and Fault History menus. Optionally, the VCL can assign fault actions to occur automatically when the associated fault is set. Sixteen VCL faults are available, stored in the VCL variables UserFault1 and UserFault2. The UserFault1,2 variables are Read/Write (R/W) and the 16 faults are stored in the lower byte of each variable like this:

```
UserFault1
*      Bit0 = VCLfault1 (Code 51)
*      Bit1 = VCLfault2 (Code 52)
*      Bit2 = VCLfault3 (Code 53)
*      Bit3 = VCLfault4 (Code 54)
*      Bit4 = VCLfault5 (Code 55)
*      Bit5 = VCLfault6 (Code 56)
*      Bit6 = VCLfault7 (Code 57)
*      Bit7 = VCLfault8 (Code 58)

UserFault2
*      Bit0 = VCLfault9  (Code 59)
*      Bit1 = VCLfault10 (Code 61)
*      Bit2 = VCLfault11 (Code 62)
*      Bit3 = VCLfault12 (Code 63)
*      Bit4 = VCLfault13 (Code 64)
*      Bit5 = VCLfault14 (Code 65)
*      Bit6 = VCLfault15 (Code 66)
*      Bit7 = VCLfault16 (Code 67)
```

The "Code" numbers result in the flashing of the controller status LEDs that help identify the fault without use of a 1313. Setting the fault in VCL is done by setting the associated fault bit in the UserFault1 and 2 variables. Clearing the fault must also be handled by VCL and is done by clearing the associated fault bit. For example:

```
if (BDI-Percentage < 10)
       {
       UserFault1.2 = ON          ;Set VCL Fault bit
       }
else
       {
       UserFault1.2 = OFF         ;Clear VCL Fault bit
       }
```

This VCL will check to see if the Battery Discharge Indicator is less than 10%. If it is, UserFault1 Bit1 (Code 52) is set. If the BDI is not less than 10%, the fault is cleared. Using just the VCL above in a program will only result in the flashing of a code 52 on controller status LEDs and no fault actions will result nor will the 1313 display any text about the fault.

To add automatic fault actions to the VCL faults, the VCL programmer must define the desired fault actions by using the sixteen VCL variables: User_Fault_Action_01 through User_Fault_Action_16. Each of the UserFault bits has a corresponding User_Fault_Action_xx variable (where "xx" is the number of the VCL fault bit). When a VCL fault bit is set, the actions defined in the corresponding User_Fault_Action_xx variable will be automatically executed by the operating system. Here are the fault actions available in the User_Fault_Actions_xx variables:

```
          VARIABLE                              FAULT ACTION
User_Fault_Action_xx
*     Bit0 = ShutdownMotor           Disable the motor
*     Bit1 = ShutdownMainContactor   Shut down the main contactor
                                     (only if Main Enable = On)
*     Bit2 = ShutdownEMBrake         Shut down the EM brake
                                     (only if EM Brake Disable Upon
                                     Fault = On)
*     Bit3 = ShutdownThrottle        Set the Throttle_Command = 0%
*     Bit4 = ShutdownInterlock       Set the Interlock_State = Off
*     Bit5 = ShutdownDriver1         Shut down Driver1
*     Bit6 = ShutdownDriver2         Shut down Driver2
*     Bit7 = ShutdownDriver3         Shut down Driver3
*     Bit8 = ShutdownDriver4         Shut down Driver4
*     Bit9 = ShutdownPD              Shut down Proportional Driver
*     Bit10 = FullBrake              Set the Brake_Command = 100%
*     Bit11 = ShutdownPump           Shut down DC pump
*     Bit12 = TrimDisable            Disable Dual Drive trim
                                     calculation.
*     Bit13 = SevereDual             For Dual Drive system, one con-
                                     troller has a severe fault but
                                     the main contactor must stay
                                     closed so the other controller
                                     can continue to operate.
*     Bit14 = ShutdownSteer          Steer angle = 0°
*     Bit15 = LOSDual                For Dual Drive system, set the
                                     max speed to Dual_LOS_Max_Speed
                                     parameter for operation in
                                     Limited Operating Strategy.
```

The User_Fault_Action_xx variables should be set up at the beginning of a VCL program (before the main loop) as these fault actions should be defined only once in a program. Here is another example:

```
User_Fault_Action_02 = 24   ;Set fault action to ShutdownInterlock
                            ;and ShutdownThrottle

MainLoop:
if (BDI_Percentage < 10)
       {
       UserFault1.2 = ON         ;Set User Fault bit
       Put_Spy_Text("BDI Low")   ;Send message to Model 840 display
       }
else
       {
       UserFault1.2 = OFF        ;Clear User Fault bit
       }
goto MainLoop
```

This time when UserFault1.2 is set, the operating system will ShutdownInterlock and ShutdownThrottle (which will result in a Throttle_Command = 0%) in addition to flashing the code 52 on the controller status LEDs. An additional VCL line was added (Put_Spy_Text ("BDI Low")) to show how additional actions beyond those provided in the User_Fault_Action_xx can be programmed using VCL. In this example the Put_Spy_Text("BDI Low") will result in the message "BDI Low" appearing on the model 840 display (presumably as a message to the vehicle operator). This example will still not result in any display on the 1313 System Faults and Fault History menus.

To add the fault text on the 1313 System Faults and Fault History menus it is necessary to create a Fault Definition for the 1313. Creating fault definitions is a subject that is covered in detail in the VCL Programmer's Guide (Section 5, Support for the 1313 Handheld Programmer). Here is an example of a fault definition:

```
User_Fault_Action_02 = 24   ;Set fault action to ShutdownInterlock
                            ;and ShutdownThrottle

MainLoop:
if (BDI_Percentage < 10)
       {
       UserFault1.2 = ON        ;Set User fault bit
       Put_Spy_Text("BDI Low")  ;Send message to Model 840 display
       }
else
       {
       UserFault1.2 = OFF       ;Clear User fault bit
       }
goto MainLoop

;  PARAMETER_ENTRY   "BDI Low Fault"
;  TYPE              FAULTS
;  WIDTH             8BIT
;  ALT_ADDRESS       Hist_UserFault1
;  ADDRESS           UserFault1
;  BITSELECT         1
;  BITACTIVELOW      NO
;  END
```

This example will result in the exact same actions as the last example, except now the fault will be displayed in the 1313 System Faults menu (only while the fault is set) and this fault will be logged into the 1313 Fault History menu after being set. The text displayed in either of these 1313 menus will be the text defined in the fault definition (in this example "BDI Low Fault" will be displayed). The variable Hist_UserFault1 is listed as an ALT_ADDRESS. This line controls whether the fault gets logged into fault history (and thus appears in the 1313 Fault History menu). The two variables that can be used in the fault definitions for ALT_ADDRESS are Hist_UserFault1 and Hist_UserFault2; these should be used in the fault definitions with the corresponding UserFault1 and UserFault2 variables. If you wish to use VCL to clear fault history, use the VCL function Clear_Diaghist(). Also note that this example fault definition was for bit 1 of UserFault1. The VCL example set and cleared this bit by using the UserFault1.2 notation (".2" being the mask that defines bit 1).

ADDITIONAL VCL FUNCTIONS

Function descriptions are provided here for the nine 1298 functions that are not included in the VCL Common Functions Manual. They are presented in the same format that is used in the VCL Common Functions Manual for the common functions.

ENABLE_PRECHARGE()

This function is designed to precharge the capacitor bank before engaging a main contactor thereby preventing current surges and to protect controller internal components and main contactor tips. This function turns on the request for precharge of the capacitor bank from KSI. When the precharge function is enabled, power will be supplied to the capacitor bank until the voltage is within 3 volts of KSI, or one second has expired, or the precharge resistor energy range has been exceeded. The current state of precharge is shown by the precharge variable (Precharge_State), which has the following values:

0 – Precharge has not yet been done.
1 – Precharge is in progress.
2 – Precharge has passed.
3 – Precharge has been aborted by the Disable_Precharge() function.
4 – Precharge has exceeded the precharge resistor energy limit.
5 – Precharge has exceeded the one-second time limit.

Syntax **`Enable_Precharge()`**

Parameters None.

Returns
0 – Precharge not enabled.
1 – Precharge successfully enabled.

Error Codes None.

Example `Enable_Precharge()`
This will attempt to precharge the capacitor bank.

DISABLE_PRECHARGE()

This function is designed to abort the precharge function and clear any precharge fault. This function aborts the request for precharge of the capacitor bank from KSI. The resultant state of the precharge variable (Precharge_State) will be set to = 3 (for precharge aborted). The precharge states are:

0 – Precharge has not yet been done.
1 – Precharge is in progress.
2 – Precharge has passed.
3 – Precharge has been aborted by the Disable_Precharge() function.
4 – Precharge has exceeded the precharge resistor energy limit.
5 – Precharge has exceeded the one-second time limit.

Syntax **`Disable_Precharge()`**

Parameters None.

Returns
0 – Precharge not aborted.
1 – Precharge successfully aborted.

Error Codes None.

Example `Disable_Precharge()`
This will attempt to abort the precharge of the capacitor bank and will clear any precharge fault.

SET_DIGOUT()

This function turns on the selected digital output. The digital outputs are active low (On = driver on and pulled to ground, Off = open circuit at the pin).

The low power digital output is protected from excessive current (current over 15 mA); a fault will occur when this current has been exceeded, and the driver will be shut off (open). Running the function again will reactivate the driver, and will attempt to clear the fault.

Syntax **`Set_Digout(DigOut_ID)`**

Parameters

`DigOut_ID` is the low power digital I/O identification.
DigOut6 = Digital Output 6 (pin 19).
DigOut7 = Digital Output 7 (pin 20).

Returns

0 – Selected digital output not set.
1 – Selected digital output successfully set.

Error Codes

Bad_ID is returned when DigOut_ID is not in the range of DigOut6 to DigOut7.

Example `Set_Digout(DigOut6)`

This example will set Digital Output 6 (pin 19) On (active low, pulled to ground).

CLEAR_DIGOUT()

This function turns off the selected digital output. The digital outputs are active low (On = driver on and pulled to ground, Off = open circuit at the pin).

Syntax **`Clear_Digout(DigOut_ID)`**

Parameters

`DigOut_ID` is the low power digital I/O identification.
DigOut6 = Digital Output 6 (pin 19).
DigOut7 = Digital Output 7 (pin 20).

Returns

0 – Selected digital output not cleared.
1 – Selected digital output successfully cleared.

Error Codes

Bad_ID is returned when DigOut_ID is not in the range of DigOut6 to DigOut7.

Example `Clear_Digout(DigOut6)`

This example will set Digital Output 6 (pin 19) Off (open circuit).

ENABLE_EMER_REV()

This function is used to engage emergency reverse using VCL. The 1313 EMR Type must be set to = 1 in order for the Enable_Emer_Rev() function to operate. If the system emergency reverse state is enabled (EMR_State bit variable = On), the emergency reverse function will operate according to the Emergency Reverse parameter settings; see page 59. To view the current emergency reverse state, see the 1313 menu Monitor » Inputs: Emer Rev.

When the EMR_Type is set to = 1 and neither the Enable_Emer_Rev() nor the Disable_Emer_Rev() function has been called, the state for emergency reverse is Off (EMR_State bit variable = Off).

Syntax **`Enable_Emer_Rev()`**

Parameters None.

Returns

0 – Emergency reverse not enabled.
1 – Emergency reverse successfully enabled.

Error Codes None.

Example `Enable_Emer_Rev()`

This will enable the emergency reverse function.

DISABLE_EMER_REV()

This function is used to disengage emergency reverse using VCL. The 1313 EMR Type must be set to = 1 in order for the Disable_Emer_Rev() function to operate. If the system emergency reverse state is disabled (EMR_State bit variable = Off), the emergency reverse function will stop operating and normal motor control function will resume (including an HPD/SRO check if the HPD/SRO Enable parameter is set to On). To view the current emergency reverse state, see the 1313 menu Monitor » Inputs: Emer Rev.

When the EMR_Type is set to = 1 and neither the Enable_Emer_Rev() nor the Disable_Emer_Rev() function has been called, the state for emergency reverse is Off (EMR_State bit variable = Off).

Syntax **`Disable_Emer_Rev()`**

Parameters None.

Returns

0 – Emergency reverse not disabled.
1 – Emergency reverse successfully disabled.

Error Codes None.

Example `Disable_Emer_Rev()`

This will disable the emergency reverse function.

SET_INTERLOCK()

This function is used to engage the system interlock using VCL. The 1313 Interlock Type parameter must be set to = 1 in order for the Set_Interlock() function to operate. If the system interlock is set (Interlock_State bit variable = On), the throttle input signal is allowed to pass along the throttle chain; see Figure 17. Additionally, if the main contactor is used (1313 Main Enable parameter = On), setting the interlock will request the main closed state from the main contactor state machine. To view the current interlock state, see the 1313 menu Monitor » Inputs: Interlock. To view the current main contactor state, see the 1313 menu Monitor » Controller: Main State.

When the Interlock_Type is set to = 1 and neither the Set_Interlock() nor the Clear_Interlock() function has been called, the state for the interlock is Off (Interlock_State bit variable = Off).

Syntax `Set_Interlock()`

Parameters None.

Returns

0 – Interlock not set.
1 – Interlock successfully set.

Error Codes None.

Example `Set_Interlock()`

This will engage the system interlock.

CLEAR_INTERLOCK()

This function is used to disengage the system interlock using VCL. The 1313 Interlock Type parameter must be set to = 1 in order for the Set_Interlock() function to operate. If the system interlock is cleared (Interlock_State bit variable = Off), the throttle input signal is not allowed to pass along the throttle chain; see Figure 14. Additionally, if the main contactor is used (1313 Main Enable parameter = On), clearing the interlock will request the main open state from the main contactor state machine. To view the current interlock state, see the 1313 menu Monitor » Inputs: Interlock. The view the current main contactor state, see the 1313 menu Monitor » Controller: Main State.

When the Interlock_Type is set to = 1 and neither the Set_Interlock() nor Clear_Interlock() function has been called, the default state for the interlock is Off (Interlock_State bit variable = Off).

Syntax `Clear_Interlock()`

Parameters None.

Returns

0 – Interlock not cleared.
1 – Interlock successfully cleared.

Error Codes None.

Example `Clear_Interlock()`

This will disengage the system interlock.

SETUP_POT_FAULTS()

This function sets the upper and lower wiper fault voltages for a given pot input and sets the replacement wiper voltage value that will be used if there is a fault. The valid range for the function parameters is 0–6.25 V (0–400 counts). If this function is not run, the default thresholds depend on the 1313 Throttle Type (or Brake Type or Hyd Throttle Type) parameter setting; see table below. If the Throttle Type = 5 (or Brake Type = 5 or Hyd Throttle Type = 5), the VCL function Setup_Pot will determine what fault thresholds are used.

THROTTLE TYPE	LOW FAULT THRESHOLD	HIGH FAULT THRESHOLD
1	0.1 V	5.5 V
2	none	5.5 V
3	0.1 V	5.5 V
4	0.1 V	5.5 V
5 (ONE_WIRE)	none	5.5 V
5 (TWO_WIRE)	0.1 V	5.5 V
5 (THREE_WIRE)	0.1 V	5.5 V

Syntax **`Setup_Pot_Faults(Pot_ID,Low_Fault,High_Fault, Fault_Value)`**

Parameters

`Pot_ID` identifies the throttle whose fault limits are being set:
THROTTLE_POT
BRAKE_POT (Note: In 1298 controllers, this is the hydraulic throttle pot.)

`Low_Fault`
Specifies the lower threshold voltage limit.
Scaling: 1 V = 64 counts.

`High_Fault`
Specifies the upper threshold voltage limit.
Scaling: 1 V = 64 counts.

`Fault_Value`
The value that is used for the pot input when there is a fault (0–32767).
Scaling: ±32767 = ±100%.

Returns

0 – Setup did not execute.
1 – Setup successful.

Error Codes

Bad_ID is returned when an incorrect pot ID is used.
Param_Range is returned when the voltage value is not within range.

Example `Setup_Pot_Faults(THROTTLE_POT,19,320,4000)`

For the throttle pot, this will set the lower pot voltage at 0.3 volts (19/64) and the upper pot voltage at 5.0 volts (320/64). When there is a pot fault, the value of 4000 will be used. That is 4000/32767 of the full output, or roughly 12%.

8 DIAGNOSTICS AND TROUBLESHOOTING

These controllers detect a wide variety of faults or error conditions. Faults can be detected by the operating system or by the VCL code. This section describes the faults detected by the operating system.

Faults detected by VCL code (faults 51–67 in Table 5) cannot be defined here as they will vary from application to application. Refer to the appropriate OEM documentation for information on these faults.

DIAGNOSTICS

Diagnostics information can be obtained in either of two ways: (1) by reading the display on a 1313 programmer or (2) by observing the fault codes issued by the Status LEDs. See Table 4 for a summary of LED display formats.

The 1313 programmer will display all faults that are currently set as well as a history of the faults that have been set since the history log was last cleared. The 1313 displays the faults by name.

The pair of LEDs built into the controller (one red, one yellow) produce flash codes displaying all the currently set faults in a repeating cycle. Each code consists of two digits. The red LED flashes once to indicate that the first digit of the code will follow; the yellow LED then flashes the appropriate number of times for the first digit. The red LED flashes twice to indicate that the second digit of the code will follow; the yellow LED flashes the appropriate number of times for the second digit.

Example: Battery Undervoltage (code 23).

In the Fault menu of the 1313 programmer, the words **Undervoltage Cutback** will be displayed; the real-time battery voltage is displayed in the Monitor menu ("Keyswitch Voltage").

The controller's two LEDs will display this repeating pattern:

RED	YELLOW	RED	YELLOW
*	* *	* *	* * *
(first digit)	(2)	(second digit)	(3)

The numerical codes used by the yellow LED are listed in the troubleshooting chart (Table 5), which also lists possible fault causes and describes the conditions that set and clear each fault.

Summary of LED display formats

The two LEDs have four different display modes, indicating the type of information they are providing.

Table 4 TYPES OF LED DISPLAY

DISPLAY	STATUS
Neither LED illuminated	Controller is not powered on; or vehicle has dead battery; or severe damage.
Yellow LED flashing	Controller is operating normally.
Yellow and red LEDs both on solid	Controller is in Flash program mode.
Red LED on solid	Watchdog failure or no software loaded. Cycle KSI to restart, and if necessary load software.
Red LED and yellow LED flashing alternately	Controller has detected a fault. 2-digit code flashed by yellow LED identifies the specific fault; one or two flashes by red LED indicate whether first or second code digit will follow.

TROUBLESHOOTING

The troubleshooting chart, Table 5, provides the following information on all the controller faults:

- fault code
- fault name as displayed on the programmer's LCD
- the effect of the fault
- possible causes of the fault
- fault *set* conditions
- fault *clear* conditions.

Whenever a fault is encountered and no wiring or vehicle fault can be found, shut off KSI and turn it back on to see if the fault clears. If it does not, shut off KSI and remove the 35-pin connector. Check the connector for corrosion or damage, clean it if necessary, and re-insert it.

Table 5 TROUBLESHOOTING CHART

CODE	PROGRAMMER LCD DISPLAY / *EFFECT OF FAULT*	POSSIBLE CAUSE	SET/CLEAR CONDITIONS
12	Controller Overcurrent *ShutdownMotor;* *ShutdownMainContactor;* *ShutdownEMBrake;* *ShutdownThrottle;* *FullBrake;* *ShutdownPump.*	1. External short of phase U,V, or W motor connections. 2. Motor parameters are mis-tuned. 3. Controller defective. 4. Speed encoder noise problems.	*Set:* Phase current exceeded the current measurement limit. *Clear:* Cycle KSI.
13	Current Sensor Fault *ShutdownMotor;* *ShutdownMainContactor;* *ShutdownEMBrake;* *ShutdownThrottle;* *FullBrake;* *ShutdownPump.*	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:* Cycle KSI.
14	Precharge Failed *ShutdownMotor;* *ShutdownMainContactor;* *ShutdownEMBrake;* *ShutdownThrottle;* *FullBrake;* *ShutdownPump.*	1. See Monitor menu » Battery: Capacitor Voltage. 2. External load on capacitor bank (B+ connection terminal) that prevents the capacitor bank from charging.	*Set:* Precharge failed to charge the capacitor bank to the KSI voltage. *Clear:* Cycle Interlock input or use VCL function *Enable_Precharge()*.
15	Controller Severe Undertemp *ShutdownMotor;* *ShutdownMainContactor;* *ShutdownEMBrake;* *ShutdownThrottle;* *FullBrake;* *ShutdownPump.*	1. See Monitor menu » Controller: Temperature. 2. Controller is operating in an extreme environment.	*Set:* Heatsink temperature below -40°C. *Clear:* Bring heatsink temperature above -40°C, and cycle interlock or KSI.
16	Controller Severe Overtemp *ShutdownMotor;* *ShutdownMainContactor;* *ShutdownEMBrake;* *ShutdownThrottle;* *FullBrake;* *ShutdownPump.*	1. See Monitor menu » Controller: Temperature. 2. Controller is operating in an extreme environment. 3. Excessive load on vehicle. 4. Improper mounting of controller.	*Set:* Heatsink temperature above +95°C. *Clear:* Bring heatsink temperature below +95°C, and cycle interlock or KSI.
17	Severe Undervoltage *Reduced drive torque.*	1. Battery menu parameters are misadjusted. 2. Non-controller system drain on battery. 3. Battery resistance too high. 4. Battery disconnected while driving. 5. See Monitor menu » Battery: Capacitor Voltage. 6. Blown B+ fuse or main contactor did not close.	*Set:* Capacitor bank voltage dropped below the Severe Undervoltage limit (see page 55) with FET bridge enabled. *Clear:* Bring capacitor voltage above Severe Undervoltage limit.

Table 5 TROUBLESHOOTING CHART, continued

CODE	PROGRAMMER LCD DISPLAY *EFFECT OF FAULT*	POSSIBLE CAUSE	SET/CLEAR CONDITIONS
18	Severe Overvoltage *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; FullBrake; ShutdownPump.*	1. See Monitor menu » Battery: Capacitor Voltage. 2. Battery menu parameters are misadjusted. 3. Battery resistance too high for given regen current. 4. Battery disconnected while regen braking.	*Set:* Capacitor bank voltage exceeded the Severe Overvoltage limit (see page 55) with FET bridge enabled. *Clear:* Bring capacitor voltage below Severe Overvoltage limit, and then cycle KSI.
22	Controller Overtemp Cutback *Reduced drive and brake torque.*	1. See Monitor menu » Controller: Temperature. 2. Controller is performance-limited at this temperature. 3. Controller is operating in an extreme environment. 4. Excessive load on vehicle. 5. Improper mounting of controller.	*Set:* Heatsink temperature exceeded 85°C. *Clear:* Bring heatsink temperature below 85°C.
23	Undervoltage Cutback *Reduced drive torque.*	1. Normal operation. Fault shows that the 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. See Monitor menu » Battery: Capacitor Voltage. 7. Blown B+ fuse or main contactor did not close.	*Set:* Capacitor bank voltage dropped below the Undervoltage limit (see page 55) with the FET bridge enabled. *Clear:* Bring capacitor voltage above the Undervoltage limit.
24	Overvoltage Cutback *Reduced brake torque.*	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 Monitor menu » Battery: Capacitor Voltage.	*Set:* Capacitor bank voltage exceeded the Overvoltage limit (see page 55) with the FET bridge enabled. *Clear:* Bring capacitor voltage below the Overvoltage limit.
25	+5V Supply Failure *None, unless a fault action is programmed in VCL.*	1. See Monitor menu » outputs: 5 Volts and Ext Supply Current. 2. External load impedance on the +5V supply (pin 26) is too low.	*Set:* +5V supply (pin 26) outside the +5V±10% range. *Clear:* Bring voltage within range.
26	Digital Out 6 Overcurrent *Digital Output 6 driver will not turn on.*	1. External load impedance on Digital Output 6 driver (pin 19) is too low.	*Set:* Digital Output 6 (pin 19) current exceeded 15 mA. *Clear:* Remedy the overcurrent cause and use the VCL function *Set_DigOut()* to turn the driver on again.

Table 5 TROUBLESHOOTING CHART, continued

CODE	PROGRAMMER LCD DISPLAY / *EFFECT OF FAULT*	POSSIBLE CAUSE	SET/CLEAR CONDITIONS
27	Digital Out 7 Overcurrent *Digital Output 7 driver will not turn on.*	1. External load impedance on Digital Output 7 driver (pin 20) is too low.	*Set:* Digital Output 7 (pin 20) current exceeded 15 mA. *Clear:* Remedy the overcurrent cause and use the VCL function *Set_DigOut()* to turn the driver on again.
28	Motor Temp Hot Cutback *Reduced drive torque.*	1. Motor temperature is at or above the programmed Temperature Hot setting, and the requested current is being cut back. 2. Motor Temperature Control menu parameters are mis-tuned. 3. See Monitor menu » Motor: Temperature and » Inputs: Analog2. 4. If the application doesn't use a motor thermistor, Temp Compensation and Temp Cutback should be programmed Off.	*Set:* Motor temperature is at or above the Temperature Hot parameter setting. *Clear:* Bring the motor temperature within range.
29	Motor Temp Sensor Fault *MaxSpeed reduced (LOS, Limited Operating Strategy), and motor temperature cutback disabled.*	1. Motor thermistor is not connected properly. 2. If the application doesn't use a motor thermistor, Motor Temp Sensor Enable should be programmed Off. 3. See Monitor menu » Motor: Temperature and » Inputs: Analog2.	*Set:* Motor thermistor input (pin 8) is at the voltage rail (0 or 10V). *Clear:* Bring the motor thermistor input voltage within range.
31	Coil1 Driver Open/Short *ShutdownDriver1.*	1. Open or short on driver load. 2. Dirty connector pins. 3. Bad crimps or faulty wiring.	*Set:* Driver 1 (pin 6) is either open or shorted. This fault can be set only when Main Enable = Off. *Clear:* Correct open or short, and cycle driver.
31	Main Open/Short *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; FullBrake; ShutdownPump.*	1. Open or short on driver load. 2. Dirty connector pins. 3. Bad crimps or faulty wiring.	*Set:* Main contactor driver (pin 6) is either open or shorted. This fault can be set only when Main Enable = On. *Clear:* Correct open or short, and cycle driver.
32	Coil2 Driver Open/Short *ShutdownDriver2.*	1. Open or short on driver load. 2. Dirty connector pins. 3. Bad crimps or faulty wiring.	*Set:* Driver 2 (pin 5) is either open or shorted. This fault can be set only when EM Brake Type = 0. *Clear:* Correct open or short, and cycle driver.
32	EMBrake Open/Short *ShutdownEMBrake; ShutdownThrottle; FullBrake.*	1. Open or short on driver load. 2. Dirty connector pins. 3. Bad crimps or faulty wiring.	*Set:* Electromagnetic brake driver (pin 5) is either open or shorted. This fault can be set only when EM Brake Type >0. *Clear:* Correct open or short, and cycle driver.
33	Coil3 Driver Open/Short *ShutdownDriver3.*	1. Open or short on driver load. 2. Dirty connector pins. 3. Bad crimps or faulty wiring.	*Set:* Driver 3 (pin 4) is either open or shorted. *Clear:* Correct open or short, and cycle driver.
34	Coil4 Driver Open/Short *ShutdownDriver4.*	1. Open or short on driver load. 2. Dirty connector pins. 3. Bad crimps or faulty wiring.	*Set:* Driver 4 (pin 3) is either open or shorted. *Clear:* Correct open or short, and cycle driver.

Table 5 TROUBLESHOOTING CHART, continued

CODE	PROGRAMMER LCD DISPLAY *EFFECT OF FAULT*	POSSIBLE CAUSE	SET/CLEAR CONDITIONS
35	PD Open/Short *ShutdownPD.*	1. Open or short on driver load. 2. Dirty connector pins. 3. Bad crimps or faulty wiring.	*Set:* Proportional driver (pin 2) is either open or shorted. *Clear:* Correct open or short, and cycle driver.
36	Encoder Fault *ShutdownEMBrake; ShutdownThrottle.*	1. Motor encoder failure. 2. Bad crimps or faulty wiring. 3. See Monitor menu » Motor: Motor RPM.	*Set:* Motor encoder phase failure detected. *Clear:* Cycle KSI.
37	Motor Open *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; FullBrake; ShutdownPump.*	1. Motor phase is open. 2. Bad crimps or faulty wiring.	*Set:* Motor phase U, V, or W detected open. *Clear:* Cycle KSI.
38	Main Contactor Welded *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; FullBrake; ShutdownPump.*	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 precharge resistor) 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 for a short time and the voltage did not discharge. *Clear:* Cycle KSI
39	Main Contactor Did Not Close *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; FullBrake; ShutdownPump.*	1. Main contactor did not close. 2. Main contactor tips are oxidized, burned, or not making good contact. 3. External load on capacitor bank (B+ connection terminal) that prevents capacitor bank from charging. 4. Blown B+ fuse.	*Set:* With the main contactor commanded closed, the capacitor bank voltage (B+ connection terminal) did not charge to B+. *Clear:* Cycle KSI.
41	Throttle Wiper High *ShutdownThrottle.*	1. See Monitor menu » Inputs: Throttle Pot. 2. Throttle pot wiper voltage too high.	*Set:* Throttle pot wiper (pin 16) voltage is higher than the high fault threshold (can be changed with the VCL function *Setup_Pot_Faults()*). *Clear:* Bring throttle pot wiper voltage below the fault threshold.
42	Throttle Wiper Low *ShutdownThrottle.*	1. See Monitor menu » Inputs: Throttle Pot. 2. Throttle pot wiper voltage too low.	*Set:* Throttle pot wiper (pin 16) voltage is lower than the low fault threshold (can be changed with the VCL function *Setup_Pot_Faults()*). *Clear:* Bring throttle pot wiper voltage above the fault threshold.
43	Pot2 Wiper High *ShutdownPump* (if Lift_Switch_Only_Enable=Off); *ShutdownPD* (if Lower_Switch_Only_Enable=Off).	1. See Monitor menu » Inputs: Pot2 Raw. 2. Pot2 wiper voltage too high.	*Set:* Pot2 wiper (pin 17) voltage is higher than the high fault threshold (can be changed with the VCL function *Setup_Pot_Faults()*). *Clear:* Bring Pot2 wiper voltage below the fault threshold.

Table 5 TROUBLESHOOTING CHART, continued

CODE	PROGRAMMER LCD DISPLAY / *EFFECT OF FAULT*	POSSIBLE CAUSE	SET/CLEAR CONDITIONS
44	Pot2 Wiper Low *ShutdownPump* (if Lift_Switch_Only_Enable=Off); *ShutdownPD* (if Lower_Switch_Only_Enable=Off).	1. See Monitor menu » Inputs: Pot2 Raw. 2. Pot2 wiper voltage too low.	*Set:* Pot2 wiper (pin 17) voltage is lower than the low fault threshold (can be changed with the VCL function Setup_Pot_Faults()). *Clear:* Bring Pot2 wiper voltage above the fault threshold.
45	Pot Low Overcurrent *ShutdownThrottle;* *ShutdownPump* (if Lift_Switch_Only_Enable=Off); *ShutdownPD* (if Lower_Switch_Only_Enable=Off).	1. See Monitor menu » Outputs: Pot Low. 2. Combined pot resistance connected to pot low is too low.	*Set:* Pot low (pin 18) current exceeds 10mA. *Clear:* Clear pot low overcurrent condition and cycle KSI.
46	EEPROM Failure *ShutdownMotor;* *ShutdownMainContactor;* *ShutdownEMBrake;* *ShutdownThrottle;* *ShutdownInterlock;* *ShutdownDriver1;* *ShutdownDriver2;* *ShutdownDriver3;* *ShutdownDriver4;* *ShutdownPD;* *FullBrake;* *ShutdownPump.*	1. Failure to write to EEPROM memory. This can be caused by EEPROM memory writes initiated by VCL, by the CAN bus, by adjusting parameters with the programmer, or by loading new software into the controller.	*Set:* Controller operating system tried to write to EEPROM memory and failed. *Clear:* Download the correct software (OS) and matching parameter default settings into the controller and cycle KSI.
47	HPD/Sequencing Fault *ShutdownThrottle.*	1. KSI, interlock, direction, and throttle inputs applied in incorrect sequence. 2. Faulty wiring, crimps, or switches at KSI, interlock, direction, or throttle inputs. 3. See Monitor menu » Inputs.	*Set:* HPD (High Pedal Disable) or sequencing fault caused by incorrect sequence of KSI, interlock, direction, and throttle inputs. *Clear:* Reapply inputs in correct sequence.
47	Emer Rev HPD *ShutdownThrottle;* *ShutdownEMBrake.*	1. Emergency Reverse operation has concluded, but the throttle, forward and reverse inputs, and interlock 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.
47	Pump HPD *ShutdownPump.*	1. KSI and the hydraulic throttle input applied in incorrect sequence; hydraulic throttle input was active when KSI was turned on. 2. Faulty wiring, crimps, or switches at KSI, lift, lower, or hydraulic throttle inputs. 3. See Monitor menu » Inputs.	*Set:* Incorrect sequence of KSI and lift, lower, or hydraulic throttle inputs according to Hyd_Inhibit_Type parameter. *Clear:* Set inputs to neutral and re-apply in correct sequence.

Table 5 TROUBLESHOOTING CHART, continued

CODE	PROGRAMMER LCD DISPLAY / *EFFECT OF FAULT*	POSSIBLE CAUSE	SET/CLEAR CONDITIONS
49	Parameter Change Fault *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; FullBrake; ShutdownPump.*	1. This is a safety fault caused by a change in certain parameter settings so that the vehicle will not operate until KSI is cycled. For example, if a user changes the Throttle Type this fault will appear and require cycling KSI before the vehicle can operate.	*Set:* Adjustment of a parameter setting that requires cycling of KSI. *Clear:* Cycle KSI.
51–67	OEM Faults *(See OEM documentation.)*	1. These faults can be defined by the OEM and are implemented in the application-specific VCL code. See OEM documentation.	*Set:* See OEM documentation. *Clear:* See OEM documentation.
68	VCL Run Time Error *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; ShutdownInterlock; ShutdownDriver1; ShutdownDriver2; ShutdownDriver3; ShutdownDriver4; ShutdownPD; FullBrake; ShutdownPump.*	1. VCL code encountered a runtime VCL error. 2. See Monitor menu » Controller: VCL Error Module and VCL Error. This error can then be compared to the runtime VCL module ID and error code definitions found in the specific OS system information file.	*Set:* Runtime VCL code error condition. *Clear:* Edit VCL application software to fix this error condition; flash the new compiled software and matching parameter defaults; cycle KSI.
69	External Supply Out of Range *None, unless a fault action is programmed in VCL.*	1. External load on the 5V and 12V supplies draws either too much or too little current. 2. Fault Checking Menu parameters Ext Supply Max and Ext Supply Min are mis-tuned. 3. See Monitor menu » Outputs: Ext Supply Current.	*Set:* The external supply current (combined current used by the 5V supply [pin 26] and 12V supply [pin 25]) is either greater than the upper current threshold or lower than the lower current threshold. The two thresholds are defined by the External Supply Max and External Supply Min parameter settings (page 47). *Clear:* Bring the external supply current within range.
71	OS General *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; ShutdownInterlock; ShutdownDriver1; ShutdownDriver2; ShutdownDriver3; ShutdownDriver4; ShutdownPD; FullBrake; ShutdownPump.*	1. Internal controller fault.	*Set:* Internal controller fault detected. *Clear:* Cycle KSI.

Table 5 TROUBLESHOOTING CHART, continued

CODE	PROGRAMMER LCD DISPLAY / *EFFECT OF FAULT*	POSSIBLE CAUSE	SET/CLEAR CONDITIONS
72	PDO Timeout *ShutdownInterlock; CAN NMT State set to Pre-operational.*	1. Time between CAN PDO messages received exceeded the PDO Timeout Period.	*Set:* Time between CAN PDO messages received exceeded the PDO Timeout Period. *Clear:* Cycle KSI or receive CAN NMT message.
73	Stall Detected *ShutdownEMBrake; ShutdownThrottle; Control Mode changed to LOS (Limited Operating Strategy).*	1. Stalled motor. 2. Motor encoder failure. 3. Bad crimps or faulty wiring. 4. Problems with power supply for the motor encoder. 5. See Monitor menu » Motor: Motor RPM.	*Set:* No motor encoder movement detected. *Clear:* Either cycle KSI, or detect valid motor encoder signals while operating in LOS mode and return Throttle Command = 0 and Motor RPM = 0.
74	Fault On Other Traction Controller	Dual Drive fault: see Dual Drive manual.	
75	Dual Severe Fault	Dual Drive fault: see Dual Drive manual.	
87	Motor Characterization Fault *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; FullBrake; ShutdownPump.*	1. Motor characterization failed during characterization process. See Monitor menu » Controller: Motor Characterization Error for cause: 0=none 1=encoder signal seen, but step size not determined; set Encoder Step Size manually 2=motor temp sensor fault 3=motor temp hot cutback fault 4= controller overtemp cutback fault 5=controller undertemp cutback fault 6=undervoltage cutback fault 7=severe overvoltage fault 8=encoder signal not seen, or one or both channels missing 9=motor parameters out of characterization range.	*Set:* Motor characterization failed during the motor characterization process. *Clear:* Correct fault; cycle KSI.
89	Motor Type Fault *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; FullBrake; ShutdownPump.*	1. The Motor_Type parameter value is out of range.	*Set:* Motor_Type parameter is set to an illegal value. *Clear:* Set Motor_Type to correct value and cycle KSI.
91	VCL/OS Mismatch *ShutdownMotor; ShutdownMainContactor; ShutdownEMBrake; ShutdownThrottle; ShutdownInterlock; ShutdownDriver1; ShutdownDriver2; ShutdownDriver3; ShutdownDriver4; ShutdownPD; FullBrake; ShutdownPump.*	1. The VCL software in the controller does not match the OS software in the controller.	*Set:* VCL and OS software do not match; when KSI cycles, a check is made to verify that they match and a fault issued when they do not. *Clear:* Download the correct VCL and OS software into the controller.

Table 5 TROUBLESHOOTING CHART, continued

CODE	PROGRAMMER LCD DISPLAY / *EFFECT OF FAULT*	POSSIBLE CAUSE	SET/CLEAR CONDITIONS
92	EM Brake Failed to Set *ShutdownEMBrake;* *ShutdownThrottle.*	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:* Activate the throttle.
93	Encoder LOS (Limited Operating Strategy) *Enter LOS control mode.*	1. Limited Operating Strategy (LOS) control mode has been activated, as a result of either an Encoder Fault (Code 36) or a Stall Detect Fault (Code 73). 2. Motor encoder failure. 3. Bad crimps or faulty wiring. 4. Vehicle is stalled.	*Set:* Encoder Fault (Code 36) or Stall Detect Fault (Code 73) was activated, and Brake or Interlock has been applied to activate LOS control mode, allowing limited motor control. *Clear:* Cycle KSI **or**, if LOS mode was activated by the Stall Fault, clear by ensuring encoder senses proper operation, Motor RPM = 0, and Throttle Command = 0.
94	Emer Rev Timeout *ShutdownEMBrake;* *ShutdownThrottle.*	1. Emergency Reverse was activated and concluded because the EMR Timeout timer has 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 Off.
95	Pump Overcurrent *ShutdownPump.*	1. External short of pump motor. 2. Controller defective.	*Set:* Pump current exceeded the current measurement limit. *Clear:* Cycle KSI.
96	Pump BDI *ShutdownPump.*	1. Battery is fully discharged. 2. BDI parameters are mis-tuned.	*Set:* BDI Percentage charge at 0% when the pump was activated. *Clear:* Charge the battery.
97	Pump Hardware Fault *ShutdownMotor;* *ShutdownMainContactor;* *ShutdownEMBrake;* *ShutdownThrottle;* *FullBrake;* *ShutdownPump.*	1. External short of the pump motor. 2. Controller defective.	*Set:* Controller hardware indicated inappropriate voltage at pump motor M- connection, or internal supply voltage problem. *Clear:* Cycle KSI.
98	Illegal Model Number *ShutdownMotor;* *ShutdownMainContactor;* *ShutdownEMBrake;* *ShutdownThrottle;* *FullBrake;* *ShutdownPump.*	1. Model_Number variable contains illegal value (not 1298, and not 1234, 1236, or 1238, which are also legal for the 1298 controller). 2. Software and hardware do not match. 3. Controller defective.	*Set:* Illegal Model_Number variable; when KSI cycles, a check is made to confirm a legal Model_Number, and a fault is issued if one is not found. *Clear:* Download appropriate software for your controller model.
99	Dualmotor Parameter Mismatch	Dual Drive fault: see Dual Drive manual.	

9

MAINTENANCE

There are no user serviceable parts in Curtis 1298 controllers. **No attempt should be made to open, repair, or otherwise modify the controller.** Doing so may damage the controller and will void the warranty.

It is recommended that the controller and connections be kept clean and dry and that the controller's fault history file be checked and cleared periodically.

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 and that normally exist in battery powered systems.

CAUTION

When working around any battery powered system, proper safety precautions should be taken. These include, but are not limited to: proper training, wearing eye protection, and avoiding loose clothing and jewelry.

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.
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. The controller should be wiped clean with a moist rag. Dry it before reconnecting the battery.
4. Make sure the connections are tight. Refer to Section 2, page 5, for maximum tightening torque specifications for the battery and motor connections.

FAULT HISTORY

The 1313 programmer can be used to access the controller's fault history file. The programmer will read out all the faults the controller has experienced since the last time the fault history file was cleared. Faults such as contactor faults may be the result of loose wires; contactor wiring should be carefully checked. Faults such as overtemperature may be caused by operator habits or by overloading.

After a problem has been diagnosed and corrected, it is a good idea to clear the fault history file. This allows the controller to accumulate a new file of faults. By checking the new fault history file at a later date, you can readily determine whether the problem was indeed fixed.

APPENDIX A

THEORY OF OPERATION

Curtis 1298 controllers convert DC battery power to 3-phase AC power by precisely controlling the induction drive for high bandwidth, high efficiency, and low ripple torque generation. To realize this level of precise torque control of induction motor drives in electric vehicles, Curtis engineers carefully evaluated and incorporated the latest technology in microprocessors, power electronics, and motor control.

Invented by Nikola Tesla in 1888, the induction motor became a workhorse that contributed to the vast industrial growth in the twentieth century. Until recently relegated to non-dynamic applications where transient response wasn't a critical concern, induction motors are now a common motor of choice in high performance control applications. This shift was facilitated by the enormous advancements in microprocessors and power silicon devices in the last thirty years, coupled with intense research and development.

The 3-phase induction motor has three sets of distributed windings in the stator winding slots. The standard induction motor has a rotor with aluminum bars short-circuited by cast aluminum end-rings. There are no brushes, commutators, or slip-rings, and—unlike DC and synchronous motors—there is no need for permanent magnets or a separate current supply for the rotor. The brushless construction of the induction motor and the rugged rotor provide high reliability, fault tolerance, low maintenance, and low cost.

Three-phase sinusoidal voltages, electrically displaced by 120°, are applied to the phase windings to create the stator magnetic field. The field rotates at the stator voltage frequency divided by the number of pole pairs. This rotating stator field induces currents in the conductive rotor bars by transformer action which, in turn, create a second rotor magnetic field. The rotor field reacts to the stator field to generate torque. The differential speed, or slip frequency, between the stator field and rotor speed is critical to the torque and speed control of an induction motor.

Motor Control Algorithms

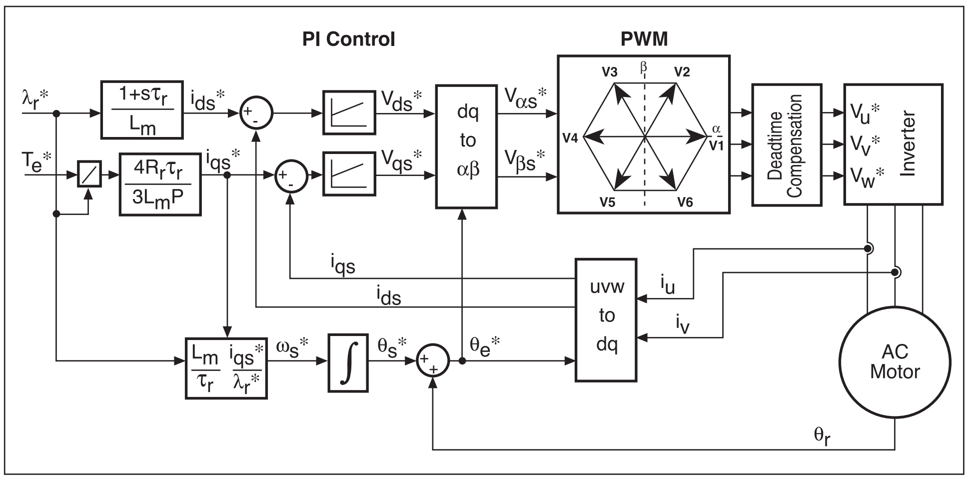
Two main approaches are commonly used for induction motor control: scalar control and vector control.

Scalar control (e.g., volts/Hz) modulates only the magnitude and frequency of the applied voltage or current. Although scalar control has the advantage of being simpler than vector control, it has poor dynamic response and lower operation efficiency. The various methods used to improve performance require extensive characterization of the motor and loads.

Vector control (e.g., indirect rotor flux orientation, stator flux orientation, etc.) manipulates the magnitude, frequency, and phase of the control variables to provide better control. The mathematical model of an induction motor is complex. Using a series of reference frame transformations, vector control simplifies the model to enable precise control of torque and flux, similar to a SepEx motor controller.

Figure A-1 shows a typical diagram of indirect rotor flux orientation. The instantaneous 3-phase currents are transformed to the rotor flux reference frame, using rotor speed and slip frequency—which means that the motor currents are now observed from the viewpoint of rotating with the rotor flux. As a result of this transformation the currents, now in what is called the d/q reference frame, lose their sinusoidal nature and look like DC signals. In the d/q reference frame, q-axis current controls torque and d-axis current controls flux. If properly oriented, the torque and flux remain independent of each other, and the motor can achieve high efficiency and dynamic response.

Fig. A-1 *Diagram of Indirect Field Orientation (IFO) technique.*



The controller uses an advanced pulse width modulation technique to maximize the utilization of battery voltage, minimize harmonic losses, and increase system efficiency. This method achieves 15% greater linear utilization of battery voltage, thereby effectively getting more usable motor power than standard PWM at the same battery voltage.

Power Section

The power section efficiently generates high current 3-phase AC signals from the DC battery voltage to drive the AC motor as requested by the motor control algorithms.

The power section is implemented as three high frequency MOSFET half-bridge power stages controlled by three pulse width modulators, as shown in Figure A-2. Each half-bridge power stage is a parallel array of high-side and low-side power MOSFETs mounted to Insulated Metal Substrate (IMS) circuit board. This technology provides a very low thermal resistance to the heatsink and enables high power capability in a compact area.

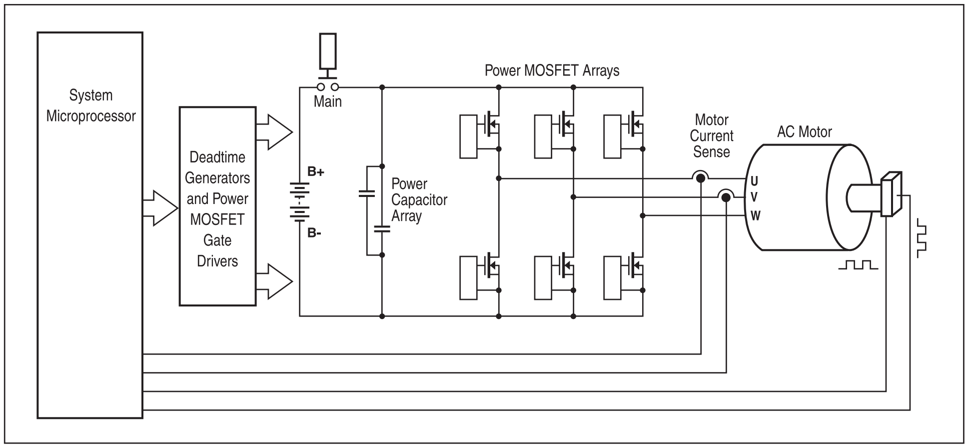


Fig. A-2 *Power section topology.*

Heavy busbars connect the IMS modules to the external motor connection studs. A bank of power capacitors keeps DC bus levels stable during high frequency MOSFET switching and also reduces EMI on the external B+ and B- cables.

Motor currents and motor speed and direction are the primary feedback signals used in the motor control algorithms. Accurate Hall sensors detect the motor currents; they do this by sensing the flux created by the motor currents on the U and V motor output busbars where they pass through the capacitor board on their way to the external motor connections. Motor speed and direction are simultaneously sensed by a quadrature-type speed encoder mounted on the motor shaft.

APPENDIX B

VEHICLE DESIGN CONSIDERATIONS REGARDING ELECTROMAGNETIC COMPATIBILITY (EMC) AND ELECTROSTATIC DISCHARGE (ESD)

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 normally in the presence of RF energy.

EMC is ultimately a system design issue. Part of the EMC performance is designed into or inherent in each component; another part is designed into or inherent in end product characteristics such as shielding, wiring, and layout; and, finally, a portion is a function of the interactions between all these parts. The design techniques presented below can enhance EMC performance in products that use Curtis motor controllers.

Emissions

Signals with high frequency content can produce significant emissions if connected to a large enough radiating area (created by long wires spaced far apart). Contactor drivers and the motor drive output from Curtis controllers can contribute to RF emissions. Both types of output are pulse width modulated square waves with fast rise and fall times that are rich in harmonics. (Note: contactor drivers that are not modulated will not contribute to emissions.) The impact of these switching waveforms can be minimized by making the wires from the controller to the contactor or motor as short as possible and by placing the wires near each other (bundle contactor wires with Coil Return; bundle motor wires separately).

For applications requiring very low emissions, the solution may involve enclosing the controller, interconnect wires, contactors, and motor together in one shielded box. Emissions can also couple to battery supply leads and throttle circuit wires outside the box, so ferrite beads near the controller may also be required on these unshielded wires in some applications. It is best to keep the noisy signals as far as possible from sensitive wires.

Immunity

Immunity to radiated electric fields can be improved either by reducing overall circuit sensitivity or by keeping undesired signals away from this circuitry. The controller circuitry itself cannot be made less sensitive, since it must accurately detect and process low level signals from sensors such as the throttle potentiometer. Thus immunity is generally achieved by preventing the external RF energy from coupling into sensitive circuitry. This RF energy can get into the controller circuitry via conducted paths and radiated paths.

Conducted paths are created by the wires connected to the controller. These wires act as antennas and the amount of RF energy coupled into them is generally proportional to their length. The RF voltages and currents induced in each wire are applied to the controller pin to which the wire is connected. Curtis controllers include bypass capacitors on the printed circuit board's throttle wires to reduce the impact of this RF energy on the internal circuitry. In some applications, additional filtering in the form of ferrite beads may also be required on various wires to achieve desired performance levels.

Radiated paths are created when the controller circuitry is immersed in an external field. This coupling can be reduced by placing the controller as far as possible from the noise source or by enclosing the controller in a metal box. Some Curtis controllers are enclosed by a heatsink that also provides shielding around the controller circuitry, while others are partially shielded or unshielded. In some applications, the vehicle designer will need to mount the controller within a shielded box on the end product. The box can be constructed of just about any metal, although steel and aluminum are most commonly used.

Most coated plastics do not provide good shielding because the coatings are not true metals, but rather a mixture of small metal particles in a non-conductive binder. These relatively isolated particles may appear to be good based on a dc resistance measurement but do not provide adequate electron mobility to yield good shielding effectiveness. Electroless plating of plastic will yield a true metal and can thus be effective as an RF shield, but it is usually more expensive than the coatings.

A contiguous metal enclosure without any holes or seams, known as a Faraday cage, provides the best shielding for the given material and frequency. When a hole or holes are added, RF currents flowing on the outside surface of the shield must take a longer path to get around the hole than if the surface was contiguous. As more "bending" is required of these currents, more energy is coupled to the inside surface, and thus the shielding effectiveness is reduced. The reduction in shielding is a function of the longest linear dimension of a hole rather than the area. This concept is often applied where ventilation is necessary, in which case many small holes are preferable to a few larger ones.

Applying this same concept to seams or joints between adjacent pieces or segments of a shielded enclosure, it is important to minimize the open length of these seams. Seam length is the distance between points where good ohmic contact is made. This contact can be provided by solder, welds, or pressure contact. If pressure contact is used, attention must be paid to the corrosion characteristics of the shield material and any corrosion-resistant processes applied to the base material. If the ohmic contact itself is not continuous, the shielding effectiveness can be maximized by making the joints between adjacent pieces overlapping rather than abutted.

The shielding effectiveness of an enclosure is further reduced when a wire passes through a hole in the enclosure; RF energy on the wire from an external field is re-radiated into the interior of the enclosure. This coupling mechanism can be reduced by filtering the wire where it passes through the shield boundary.

Given the safety considerations involved in connecting electrical components to the chassis or frame in battery powered vehicles, such filtering will usually consist of a series inductor (or ferrite bead) rather than a shunt capacitor. If a capacitor is used, it must have a voltage rating and leakage characteristics that will allow the end product to meet applicable safety regulations.

The B+ (and B-, if applicable) wires that supply power to a control panel should be bundled with the other control wires to the panel so that all these wires are routed together. If the wires to the control panel are routed separately, a larger loop area is formed. Larger loop areas produce more efficient antennas which will result in decreased immunity performance.

Keep all low power I/O separate from the motor and battery leads. When this is not possible, cross them at right angles.

ELECTROSTATIC DISCHARGE (ESD)

Curtis motor controllers contain ESD-sensitive components, and it is therefore necessary to protect them from ESD (electrostatic discharge) damage. Most of these control lines have protection for moderate ESD events, but must be protected from damage if higher levels exist in a particular application.

ESD immunity is achieved either by providing sufficient distance between conductors and the ESD source so that a discharge will not occur, or by providing an intentional path for the discharge current such that the circuit is isolated from the electric and magnetic fields produced by the discharge. In general the guidelines presented above for increasing radiated immunity will also provide increased ESD immunity.

It is usually easier to prevent the discharge from occurring than to divert the current path. A fundamental technique for ESD prevention is to provide adequately thick insulation between all metal conductors and the outside environment so that the voltage gradient does not exceed the threshold required for a discharge to occur. If the current diversion approach is used, all exposed metal components must be grounded. The shielded enclosure, if properly grounded, can be used to divert the discharge current; it should be noted that the location of holes and seams can have a significant impact on ESD suppression. If the enclosure is not grounded, the path of the discharge current becomes more complex and less predictable, especially if holes and seams are involved. Some experimentation may be required to optimize the selection and placement of holes, wires, and grounding paths. Careful attention must be paid to the control panel design so that it can tolerate a static discharge.

MOV, transorbs, or other devices can be placed between B- and offending wires, plates, and touch points if ESD shock cannot be otherwise avoided.

APPENDIX C

CURTIS WEEE / RoHS STATEMENT, MARCH 2009

WEEE

The Directive 2002/96/EC on Waste Electrical and Electronic Equipment (WEEE) was adopted by the European Council and Parliament and the Council of the European Union on January 27, 2003. The aim of the directive was to improve the collection and recycling of WEEE throughout the EU, and to reduce the level of non-recycled waste. The directive was implemented into law by many EU member states during 2005 and 2006. This document provides a general description of Curtis's approach to meeting the requirements of the WEEE legislation.

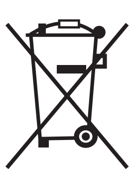
Note that the directive gave some flexibility to the member states in implementing their individual WEEE regulations, leading to the definition of varying implementation requirements by country. These requirements may involve considerations beyond those reflected in this document. This statement is not intended and shall not be interpreted or construed to be legal advice or to be legally binding on Curtis or any third party.

Commitment

Curtis is committed to a safe and healthy environment and has been working diligently to ensure its compliance with WEEE legislation. Curtis will comply with WEEE legislation by:

- Designing its equipment with consideration to future dismantling, recovery and recycling requirements;
- Marking its products that fall within the scope of the directive with the required symbol and informing users of their obligation;
- To separate WEEE from general waste and dispose of it through the provided recycling systems;
- Reporting information as required by each member state;
- Facilitating the collection, recycling and disposal of WEEE from private households and other than private households (businesses) as defined by the applicable member state regulation;
- Providing information to treatment centres according to the requirements defined in the local regulation.

WEEE symbol on Curtis products



The requirement to mark equipment with the WEEE symbol (the crossed-out wheeled bin) went into effect as of August 13, 2005. As of this date, Curtis Instruments began the process of marking all products that fall within scope of this directive with the WEEE symbol, as shown opposite.

Obligations for buyers of electrical and electronic equipment

As of 13 August 2005, in each EU member state where the WEEE directive has been implemented, disposal of EEE waste other than in accordance with the scheme is prohibited. Generally, the schemes require collection and recycling of a broad range of EEE products. Certain Curtis products fall within the scope of the directive and the implemented member state regulations. Affected Curtis products that have reached end-of-life must not be disposed as general waste, but instead, put into the collection and recycling system provided in the relevant jurisdiction.

RoHS

For several years now, Curtis has been implementing a rigorous program with the aim of achieving full compliance with the Restrictions on the use of Hazardous Substances (RoHS) Directive, 2002/95/EC.

Curtis has taken all available steps to eliminate the use of the six restricted hazardous substances listed in the directive wherever possible. As a result of the Curtis RoHS program, many of our instrumentation product lines are now fully RoHS compliant.

However, Curtis's electronic motor speed controller products are safety-critical devices, switching very large currents and designed for use in extreme environmental conditions. For these product lines, we have successfully eliminated five out of the six restricted hazardous substances. The single remaining issue preventing full RoHS compliance is the unsuitability of the lead-free solders available to date, due to the well-documented issues such as tin whiskers, and premature failure (compared with leaded solder) due to shock, vibration, and thermal cycling.

Curtis is closely monitoring all RoHS developments globally, and in particular is following the automotive industry's attempts to introduce lead-free solder as a result of the End of Life Vehicle (ELV) Directive, 2003/53/EC. To date, the automotive industry has rejected all lead-free solder pastes due to a significant reduction in reliability compared to leaded soldering.

Curtis firmly believes that the operating environments, safety requirements, and reliability levels required of automotive electronics are directly analogous to that of our speed controller products. As such, Curtis will not be switching to a lead-free solder process until lead-free solder pastes and techniques are available that meet the requirements of the RoHS study groups and ELV Automotive Industry bodies. That is, when all known issues, including that of tin whiskers, are satisfactorily resolved.

At this moment in time, all Curtis motor speed controllers used on industrial vehicle applications are also regarded as exempt under EEE category 9 of the RoHS directive 2002/95/EC. This means there is no requirement at this time for Curtis control systems used on such equipment to comply with the directive. Curtis will work closely with all key customers to ensure that whenever possible, we are in a position to continue the supply of products should these exemptions expire.

APPENDIX D
PROGRAMMING DEVICES

Curtis programmers provide programming, diagnostic, and test capabilities for the 1298 controller. The power for operating the programmer is supplied by the host controller via a 4-pin connector. When the programmer powers up, it gathers information from the controller.

Two types of programming devices are available: the 1314 PC Programming Station and the 1313 handheld programmer. The Programming Station has the advantage of a large, easily read screen; on the other hand, the handheld programmer (with its 45×60mm screen) has the advantage of being more portable and hence convenient for making adjustments in the field.

Both programmers are available in User, Service, Dealer, and OEM versions. Each programmer can perform the actions available at its own level and the levels below that—a User-access programmer can operate at only the User level, whereas an OEM programmer has full access.

PC PROGRAMMING STATION (1314)

The Programming Station is an MS-Windows 32-bit application that runs on a standard Windows PC. Instructions for using the Programming Station are included with the software.

HANDHELD PROGRAMMER (1313)

The 1313 handheld programmer is functionally equivalent to the PC Programming Station; operating instructions are provided in the 1313 manual. This programmer replaces the 1311, an earlier model with fewer functions.

PROGRAMMER FUNCTIONS

Programmer functions include:

Parameter adjustment — provides access to the individual programmable parameters.

Monitoring — presents real-time values during vehicle operation; these include all inputs and outputs.

Diagnostics and troubleshooting — presents diagnostic information, and also a means to clear the fault history file.

Programming — allows you to save/restore custom parameter settings files and also to update the system software (not available on the 1311).

Favorites — allows you to create shortcuts to your frequently-used adjustable parameters and monitor variables (not available on the 1311).

APPENDIX E

SPECIFICATIONS

Table E-1 SPECIFICATIONS: 1298 CONTROLLER

Nominal input voltage	24V
PWM operating frequency	10 kHz
Maximum encoder frequency	10 kHz
Maximum controller output frequency	300 Hz
Electrical isolation to heatsink	500 V ac (minimum)
Storage ambient temperature range	-40°C to 95°C (-40°F to 203°F)
Operating ambient temp. 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)
Heatsink overtemperature cutoff	linear cutback starts at 85°C (185°F); complete cutoff at 95°C (203°F)
Heatsink undertemperature cutoff	complete cutoff at -40°C (-40°F)
Package environmental rating	IP65
Weight	2.84 kg (6.3 lbs)
Dimensions (W×L×H)	155 × 212 × 75 mm (6.1" × 8.3" × 3.0")
EMC	Designed to the requirements of EN12895
Safety	Designed to the requirements of EN1175
UL	UL recognized component per UL583

Note: Regulatory compliance of the complete vehicle system with the controller installed is the responsibility of the OEM.

MODEL NUMBER	NOMINAL BATTERY VOLTAGE (volts)	2 MIN TRACTION CURRENT RATING (amps rms)	1 HOUR TRACTION CURRENT RATING (amps rms)	2 MIN DC PUMP CURRENT RATING (amps)	1 HOUR DC PUMP CURRENT RATING (amps)
1298-220X	24	250	165	300	215

Notes: Internal algorithms automatically reduce maximum current limit when heatsink temperature is >85°C or battery voltage is outside the allowed limits. Heatsink temperature is measured internally near the power MOSFETs.

2-minute ratings are based on an initial controller heatsink temperature of 25°C and a maximum heatsink temperature of 85°C. No addtitional external heatsink is used for the 2-minute rating test.

1-hour ratings assume that only one motor (traction or pump) is running for the test. For a combined rating, contact your Curtis customer support engineer for power dissipation charts.