EV Driver Controls
User's Manual

31 August 2015

©2015 Tritium Pty Ltd

Brisbane, Australia

http://www.tritium.com.au

TABLE OF CONTENTS

1   INTRODUCTION

This document describes the interface, installation, and usage requirements for the Tritium EV Driver Controls product.   It also provides information on programming the device to implement custom functionality.

The EV Driver Controls provide an easy way to control a Tritium WaveSculptor motor controller.   The driver controls come programmed from the factory configured with sensible default values that will work in a plug-and-play manner with a motor controller.

The EV Driver Controls provide analog and digital inputs, used to generate commands to the motor controller.   It also provides both analog and digital outputs to drive dashboard gauges and error and status LEDs.

The microcontroller firmware for the device (written in 'C') is available on the Tritium website under an open-source BSD license, as are the hardware schematics and component overlays.

2   PHYSICAL LAYOUT

The following illustration shows the connections and indicators on the front panel of the EV Driver Controls. A CAN bus connection, 12V in and out, and two status LEDs are present.
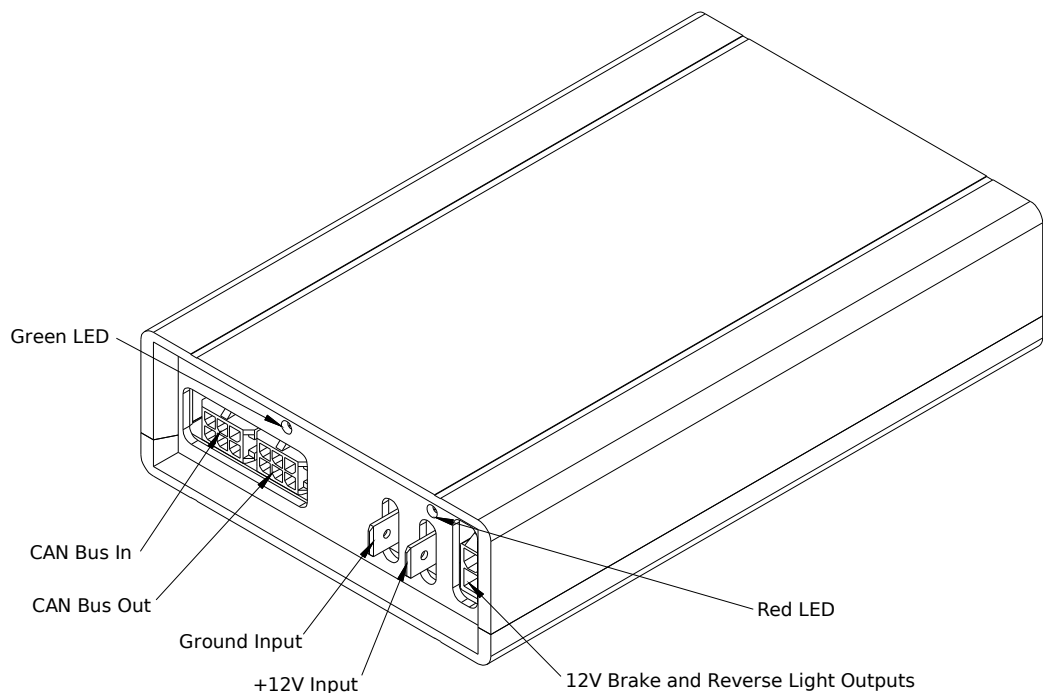

The rear panel contains a single DB37 connector, connecting all the analog and digital I/O connections required to drive a vehicle.

3 CAN BUS

3.1 CAN NETWORK TOPOLOGY

The CAN bus is structured as a linear network, with short stubs branching from 'T' connectors on the main bus backbone to each device. The CAN bus data lines must be terminated at each end of the main bus with 120 ohm resistors between the CAN-H and CAN-L signals.

In the range of Tritium EV products, including the WaveSculptor 200, EV Driver Controls, and BMS Pack Master, the CAN connections are implemented with an 'in' and an 'out' connector, therefore placing the 'T' on the device, resulting in a very short fixed-length stub on the circuit board of each device. This is ideal from a signal integrity and network performance point of view.
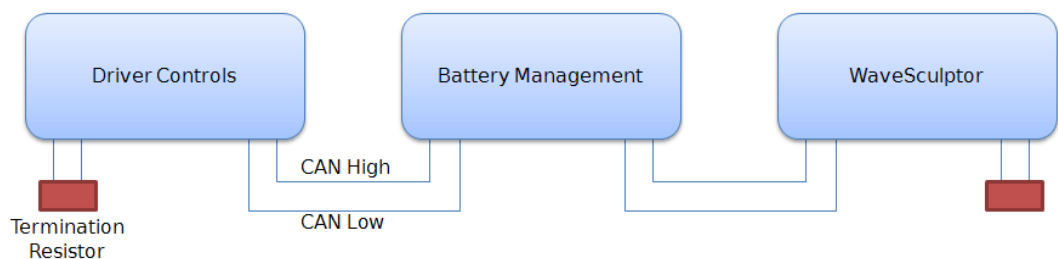
The WaveSculptor motor controller uses the CAN bus to receive commands and transmit telemetry, as well as to provide low-voltage DC power to operate the controller electronics.

Therefore, a basic system consists of three major components:

1. WaveSculptor motor controller
2. Driver controls interface to pedals, switches & gauges
3. Power supply (nominally 12V DC) to operate the system

Other components can also be added to the network. Items typically found in an electric vehicle would include:

4. LCD and other driver information displays
5. Battery management system
6. Multiple motor controllers
7. Datalogging

3.2 CAN WIRING

The CAN data lines (CAN-H and CAN-L) must be implemented with twisted-pair wire for proper data integrity. The wire should have a characteristic impedance of 120 ohms.

Power should also be provided along the CAN cable, ideally with another twisted pair to minimise noise pickup. An overall shield can also be advantageous.

From a performance perspective, the optimal choice of cable is 7mm Devicenet CANbus 'thin' cable, with 24AWG (data) + 22AWG (power) twisted pairs and a braided shield. Using this cable will result in a robust installation. However, it is

not cheap, and using it for a large network with many devices will quickly add up to a significant cost.

For those on a budget, standard CAT5 network cabling (which has an impedance of 100 ohms) can be used, but may become unreliable in longer networks or in the presence of electrical noise from DC/DC converters and other electrical devices in the system – use this type of cable at your own risk.

3.3 CAN CONNECTOR

The connector used on the EV Driver Controls and other Tritium devices is a 6-way 3mm pitch Molex MicroFit connector. The pinout is shown below, as viewed from the wire side – as you would look at it while inserting crimps. The colours shown match those in the standard DeviceNet CAN cabling pairs.

	(BARE)	(BLUE)	(WHITE)
	SHIELD GROUND	CAN-L	CAN-H
	SHIELD	GROUND	+12V
	(BARE)	(BLACK)	(RED)
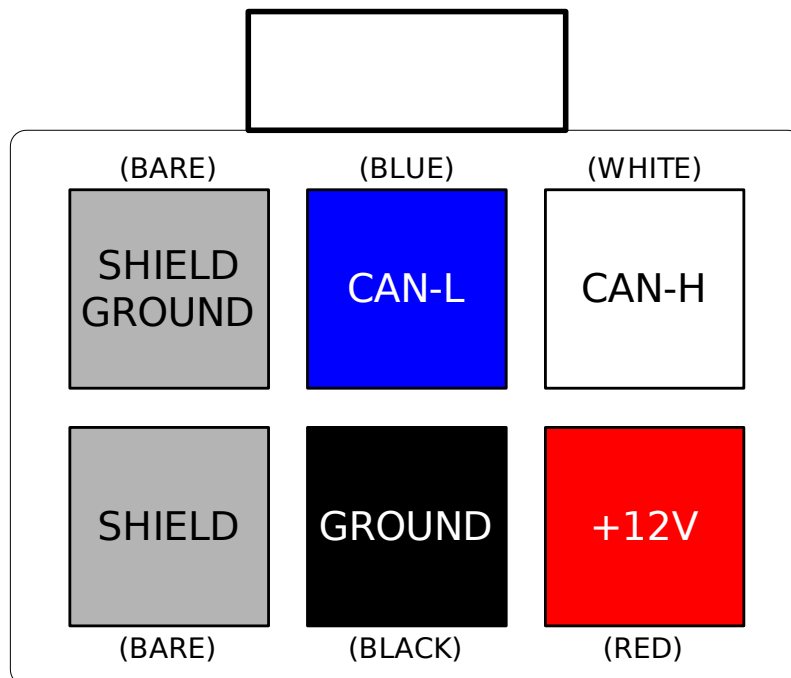
3.4 CAN SHIELDING

If the recommended braided shield is used in the cable, then terminate it to the SHIELD pin (lower-left corner on the connector) on both CAN IN and CAN OUT connectors on each device.

On **one device only** in the network, instead of using the SHIELD pin, terminate the shield to the SHIELD GROUND pin (upper-left corner on the connector) on both CAN IN and CAN OUT connectors, to ground the shield for the entire network at this single point. The usual place to do this is where power is fed into the network, typically at Tritium's **EV Driver Controls** product.

3.5 CAN TERMINATION

To implement the required 120 Ohm termination resistor at each end of the CAN bus, plug a connector into the unused CAN connector on the last device at each

end of the network with a resistor crimped into the appropriate locations.

3.6 COMMUNICATIONS

The CAN standard does not specify high-level message protocols. Tritium devices use a custom protocol, outlined in the communication specification document for each device.

By default, each device operates at 500 kbits/second, one step below the maximum possible data rate of 1 Mbit/second, and comes programmed from the factory with a CAN base address that will allow it to work without problems with other Tritium devices. Currently, modifying the driver controls data rate or CAN base address requires reprogramming the microcontroller, although hardware support is provided to allow updates over the CAN bus in the future.

The WaveSculptor motor controller expects regular messages from the driver controls device. If a message is not received within a set timeout period then the controller will change to a safe mode and will stop driving the motor. This protects against faults where either a connector is loose or broken, the cable has been damaged, or the driver controls have failed.

3.7 POWER SUPPLY

The EV Driver Controls provides a means to connect power from the 12V car battery into the CAN bus, to power other devices on the network. The default software switches power to the CAN bus based on the state of the ignition key – the CAN bus will be powered if the key is in either the Accessories or the Run position.

3.8 SYSTEM EXPANSION

Tritium can provide a CAN bus LCD display capable of showing up to four different telemetry values (one at a time) on a 3.5 digit sunlight-readable screen. Multiple displays can be used if desired.

3.9 MULTIPLE MOTORS

Multiple motors/controllers are accommodated easily with the CAN bus system. All that is required is for each WaveSculptor controller to be programmed to receive messages from the same driver controls base address, and then to run the vehicle in current-control mode (the default setup). Each motor will now operate at the same current, thus giving automatic wheel speed differences for cornering, with the system acting as an electronic differential.

4   **12V INPUT & OUTPUT**

4.1   **POWER SUPPLY**

The Driver Controls requires a 12V power connection to the spade terminals on the front panel. This connection should be capable of supplying enough current to run all devices on the CAN bus, plus the expected current draw of the brake and reversing lights (if connected). It should be fused externally to the driver controls.

This power can be from an 'always on' connection, however the driver controls draws around 30mA continuously, so will drain your accessory battery if your DC/DC converter from the high voltage traction pack does not operate continuously.   This current draw will be lower in the future, however sleep mode functions are not implemented in the default software at this point.

Supplying the driver controls from a relay activated by the 'Run' position of the ignition key is a good compromise.

Supplying the driver controls from the 'Accessories' position of the ignition key is not recommended as this power is usually cut when the key is moved to the start position.

4.2   **BULB DRIVE OUTPUTS**

The driver controls has two 12V 3A outputs, and the default software is suitable for connecting these two outputs to your brake lights and your reversing lights.

This is so that the brake lights can activate if regenerative braking is used, and the reversing lights can be activated if you are using a fixed ratio gearbox and are performing vehicle reversing by changing the motor rotation direction.

If you are not using regen braking then you can leave the brake output from the driver controls disconnected, and leave the vehicle wiring untouched.

If you are using a conventional gearbox to perform reversing, then you can leave the reversing output from the driver controls disconnected, and leave the vehicle wiring untouched.

5 VEHICLE CONTROL CONNECTIONS

The vehicle control connector is a female DB37 socket. The following illustration shows the pinout as viewed when looking into the driver controls, the same view you have when soldering or inserting crimps into the rear of a mating connector housing.

Top row (left to right):
- Pin 19: Gauge Drive Ground
- Pin 18: Gauge Drive Output 3
- Pin 17: Gauge Drive Output 1
- Pin 16: LED Output Fault 3
- Pin 15: LED Output Fault 1
- Pin 14: LED Output Backlighting
- Pin 13: LED Output Gear 3
- Pin 12: LED Output Gear 1
- Pin 11: Gear Switch 6 (Active Low)
- Pin 10: Gear Switch 4 (Active Low)
- Pin 9: Gear Switch 2 (Active Low)
- Pin 8: Fuel Door Switch Ground
- Pin 7: Brake Lights (Active High)
- Pin 6: Ignition Switch Run (Active High)
- Pin 5: Analog C Ground
- Pin 4: Analog C +5V Output
- Pin 3: Analog B signal Input
- Pin 2: Analog A Ground
- Pin 1: Analog A +5V Output

Bottom row (left to right):
- Pin 37: Gauge Drive Output 4
- Pin 36: Gauge Drive Output 2
- Pin 35: LED Output Fault Ground
- Pin 34: LED Output Fault 2
- Pin 33: LED Output Gear Ground
- Pin 32: LED Output Gear 4
- Pin 31: LED Output Gear 2
- Pin 30: Gear Switch Ground
- Pin 29: Gear Switch 5 (Active Low)
- Pin 28: Gear Switch 3 (Active Low)
- Pin 27: Gear Swtich 1 (Active Low)
- Pin 26: Fuel Door Switch (Active Low)
- Pin 25: Ignition Switch Start (Active High)
- Pin 24: Ignition Switch Accy (Active High)
- Pin 23: Analog C Signal Input
- Pin 22: Analog B Ground
- Pin 21: Analog B +5V Output
- Pin 20: Analog A Signal Input

All ground pins on the connector are identical, and connected together internally to the driver controls. They are provided separately on the connector to simplify wiring, as a ground can be run out from the driver controls along with the bundle of other wires going to each input or output device.

5.1 ANALOG INPUTS

The driver controls provide three analog inputs. These are suitable for connecting to a potentiometer (1k to 10k linear recommended), with the driver controls supplying a stable 5V supply, capable of up to 75mA shared between all three outputs.

They are also suitable for connection to a voltage output device such as a hall-effect 'drive by wire' pedal assembly, as long as it is capable of operating from 5V and has outputs that do not go below ground or above 5V. Most pedals meet these requirements.

The default software uses the Analog A channel as the motor current input, suitable for connection to your accelerator (gas) pedal. It has a small dead-band at either end of the pedal travel, but basically maps 0 to 100% throttle to the full range of potentiometer movement.

Analog B and Analog C are planned to be used for the 2nd cross-check output from a hall pedal, and the regenerative braking strength slider control, respectively. However neither is currently implemented in the default software.

5.2   DIGITAL INPUTS

Multiple digital switch or voltage inputs are provided. Each input accepts a voltage between ground and the 12V supplied to the driver controls. Inputs have a wide hysteresis band to provide noise immunity, and a 10k pull-up resistor to avoid floating inputs.

Inputs can be connected to a signal that will either be 12V or ground, such as the ignition switch position outputs.

Inputs can also be connected to a toggle or pushbutton switch to ground, with the internal pull-up resistor providing the alternative logic level.

Inputs are configured in four groups, but they are all identical in hardware and can be re-mapped as desired by modifying the device software.

The default firmware is configured with the input polarity suitable for the following inputs:

•   Ignition Accessories, Run, and Start positions (voltages: 12V = active)

•   Brake Status (voltage: 12V = brakes on)

•   Fuel Door switch (switch to ground: closed to drive, open to charge)

•   Gear selection (switches to ground: closed to select specific mode)

The default software provides power to the CAN bus if the ignition switch Accessories or Run inputs are active.

The default software will only issue commands to the motor controller if the ignition switch Run input is active, and the fuel door is closed.

The default software will send 0% torque commands to the motor controller if the brake switch is active.

The default software will drive in the forwards direction if Gear Switch 1 or 2 is selected, will switch to Neutral if Gear Switch 3 is selected, and will drive in reverse if Gear Switch 4 is selected. Do not select multiple gears simultaneously.

5.3   ANALOG OUTPUTS

Four analog outputs are provided to drive dashboard gauges or other devices in the vehicle. All hardware is functionally identical, and capable of providing either a 0-12V pulse train output, or a 'variable resistance' to ground. This allows driving either a tachometer type gauge, or a fuel type gauge.

The default software drives Gauge outputs 1 and 2 as pulse outputs, with Gauge 1 showing engine RPM, and Gauge 2 showing system power.

The default software drives Gauge outputs 3 and 4 as resistance outputs, with Gauge 3 showing battery voltage, and Gauge 4 showing motor controller

temperature.

5.4 DIGITAL OUTPUTS

Eight digital outputs are provided, each capable of driving to either Ground or 12V, and capable of up to 100mA per output. These outputs are not protected against overcurrent, and are designed to drive current-limited LED loads or small relays. Take care to avoid overloading these outputs, and make sure to use a diode clamp across the coil if driving a relay.

If used, each output should be connected through a LED and a current-limiting resistor to Ground. Select the resistor value to set the LED current to the brightness required in your situation. For a typical LED, a 560 Ohm resistor will give approximately 20mA LED current, giving high brightness from a standard 3mm or 5mm LED.

The default software shows the currently selected gear mode by illuminating one of the Gear LED 1-4 outputs. The Gear LED backlight is illuminated if either the ignition Accessories or Run positions are active.

The Fault LED outputs are unused by the default software.

6 BASIC CONNECTION DIAGRAM

The following schematic shows the simplest possible connection, when using the default software.
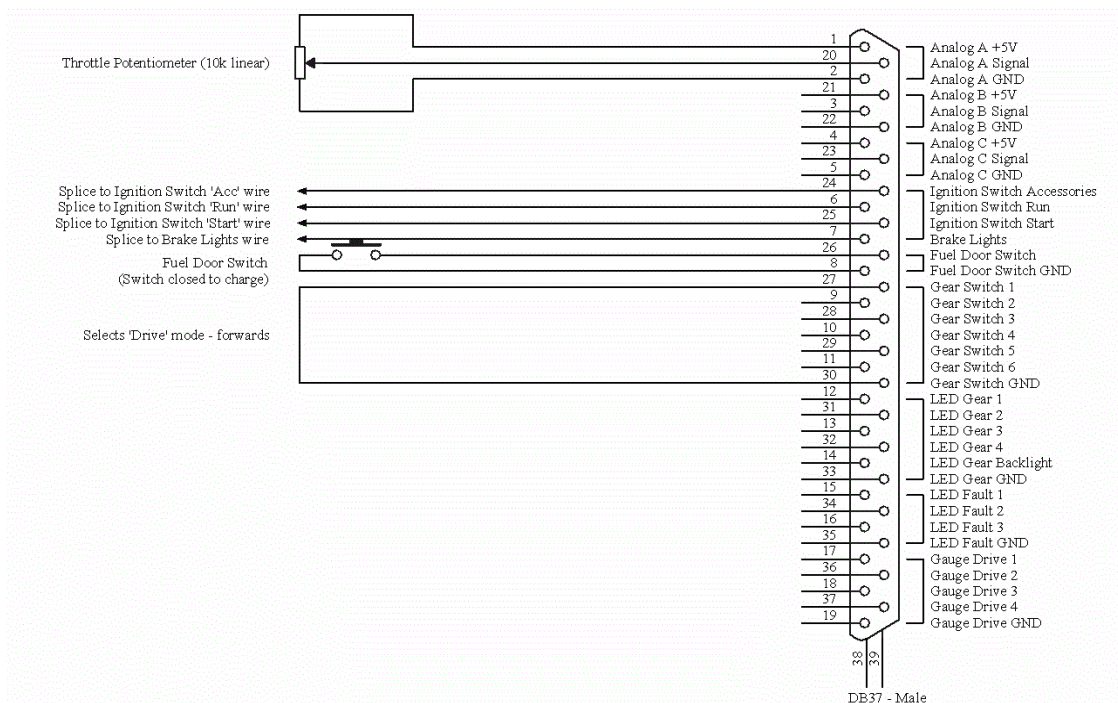
7 SOFTWARE PROGRAMMING

7.1 OVERVIEW

The driver controls are based around a Texas Instruments (TI) MSP430 16-bit embedded low-power microcontroller, operating from a 16 MHz clock as default. CAN bus support is provided via a Microchip MCP2515 CAN controller and TI SN65HVD234 CAN transceiver. The driver controls are not isolated from the CAN bus.

7.2 SCHEMATICS & SOURCE CODE

Schematics and PCB component position overlays in PDF format are available for download on the Tritium website. A zip file is also provided containing the default source code for the microcontroller, written in 'C' and available under a BSD open-source license.

Please refer to these references if developing custom firmware for the driver controls.

7.3 TOOLCHAIN

The example default code provided is configured to work with the freely-available open-source MSP430 GCC toolchain, which provides a command-line driven compiler, binutils, download, and real-time debug capability through a JTAG header present on the driver controls PCB. The debug functionality requires purchase of a USB programmer from Texas Instruments. Please refer to the README file with the source code for download and installation instructions.

7.4 CODE DOWNLOAD

The EV Driver Controls contains a facility to load new software into the device via the CAN bus. This works in conjunction with our CAN-Ethernet bridge. The bootloader utility is available on the Tritium website. Using this method of programming means that a debugger is not possible, but it works immediately and does not require the purchase of additional hardware.

8 REVISION RECORD

REV	DATE	CHANGE
1	15 May 2010	Document creation (JMK)
2	12 July 2010	Updated for v2 hardware
3	9 December 2010	Modified software section to include bootloader (JMK)
4	31 August 2015	Fixed figures (AKR)