

MOBILITY



To The Maxx

PART 3 — Motor Skills

⇒ by **Chris Cooper**

For more information on this product, visit www.machinebus.com/emaxx

Last month, we laid the foundation of the E-Maxx control system. After adding a module for servo control and another for wireless communication, we can effectively operate the E-Maxx using a USB joystick from a PC.

The existing Traxxas electronic speed controller (ESC) allows us to drive the two E-Maxx motors in forward and reverse, but only at relatively high speeds. This makes it difficult to position the E-Maxx accurately, read sensors, and move in tight spaces. The ESC was designed for racing. But moving fast is not the same as going somewhere, and our robot will need to go places.

In this article, we will replace the ESC with two motor controllers, which are better suited for robotics because they are capable of fine-grained speed increments and have a greater overall speed range. We will also add a quadrature rotational encoder to monitor

the speed and direction of the E-Maxx. After these modifications, the E-Maxx will have the motor skills required for precise, controlled, and consistent movement about its surroundings.

Replacing the ESC With Motor Controllers

Before removing the ESC, we need to understand its current functions. First of all, it uses pulse width modulation (PWM) to regulate the 14.4 V supply going to the motors. A standard R/C servo signal is used to vary the power. Secondly, the ESC regulates the 14.4 supply voltage to provide 5 V of power to

the steering and shifting servos. So when we replace the ESC, we will create another 5-6 V supply to power the servos.

Using a second LM2940-CT 5.0 low dropout voltage regulator with a heatsink proves to be a viable solution to power the steering and shifting servos. It also keeps the electrically noisy servos isolated from our digital electronics. Each servo has a no load operating current of around 350-450 mA, which falls under the one amp limit of the LM2940.

Photo Above: The E-Maxx RC monster truck makes an excellent robotics base. *Photo courtesy of Traxxas.*

Figure 1. The schematic for power regulation that we will duplicate for powering servos.

Finding the right motor controllers to replace the ESC's motor control functionality took some investigation. I had a difficult time finding information on the motors supplied with the E-Maxx. The supply voltage is 14.4 V, but I also needed to know the maximum current draw.

A motor draws the most current when stalled or starting. One way to determine the stall current without a data sheet is to measure the resistance of the brushes and winding when stationary. Using a standard ohmmeter, I found the resistance to be 0.65 ohms. I calculated the maximum current draw of each motor from the two known values using Ohm's Law:

$$14.4 \text{ V} / .65 \text{ Ohms} = 22.15 \text{ Amps}$$

Based on these motor characteristics, I found the new Parallax HB-25 Motor Controller to be a great match. The most impressive feature of the HB-25 is its excellent power dissipation — a feature that the alternative controllers I researched lacked. The HB-25 incorporates a large, machined aluminum heatsink, thermally bonded to the motor driver circuit. Attached to the heatsink is a fan that draws air through holes in the heatsink to provide outstanding cooling.

In addition to the impressive thermal characteristics, the HB-25 features over voltage, over current, and over temperature protection circuitry and can be controlled using standard R/C servo signals.

Two HB-25s can easily handle the E-Maxx motors, accepting motor supply voltages from 6-16 volts and a continuous current of 25 amps. Since the HB-25 is controlled using standard R/C servo signals, I can still use the servo control circuitry described in the March issue.

Installing the two HB-25 motor controllers is straightforward. There are two mounting holes pre-drilled in the aluminum blocks. I drilled matching holes

Figure 4. The six-wire connector keeps the wiring well organized.

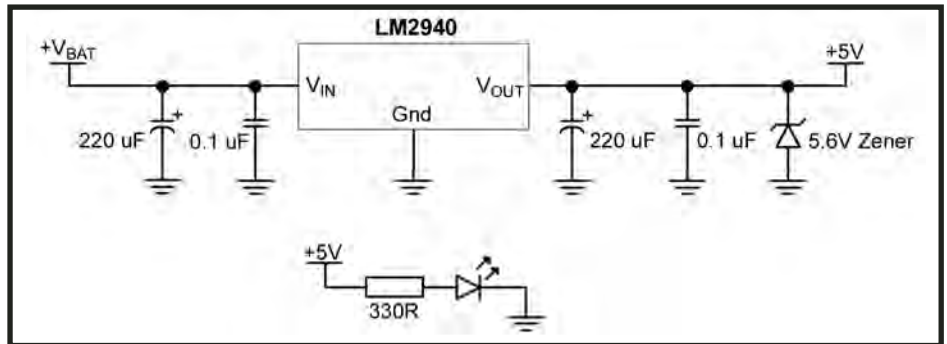


Figure 2. The Parallax HB-25 Motor Controller has an impressive thermal design to stay cool while handling the E-Maxx motors.

into the deck of the E-Maxx and fastened them using 4-40 screws. I wired the batteries, motors, and HB-25s using 14 gauge silicone wire, which is more flexible than plastic insulated wire.

I chose spade connectors to attach the wire to the HB-25's screw terminals, as opposed to ring connectors. Spade connectors make adding and removing connections easier than ring connectors, which would require removing the screw completely when adding or removing a connection. Check that the ends of the spade connectors do not make accidental contact with the fuse or

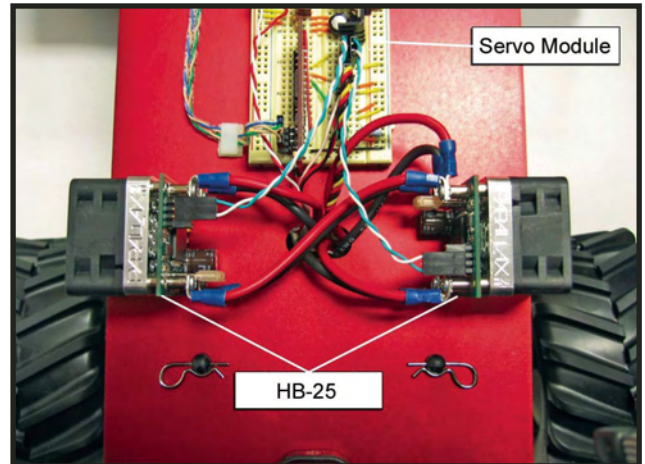
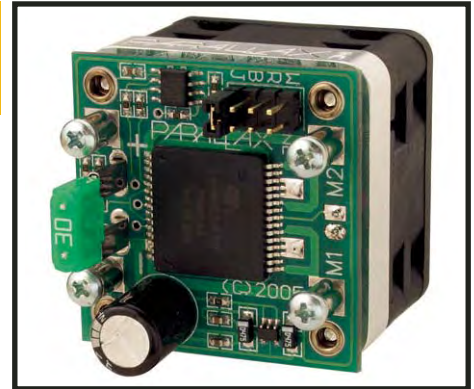
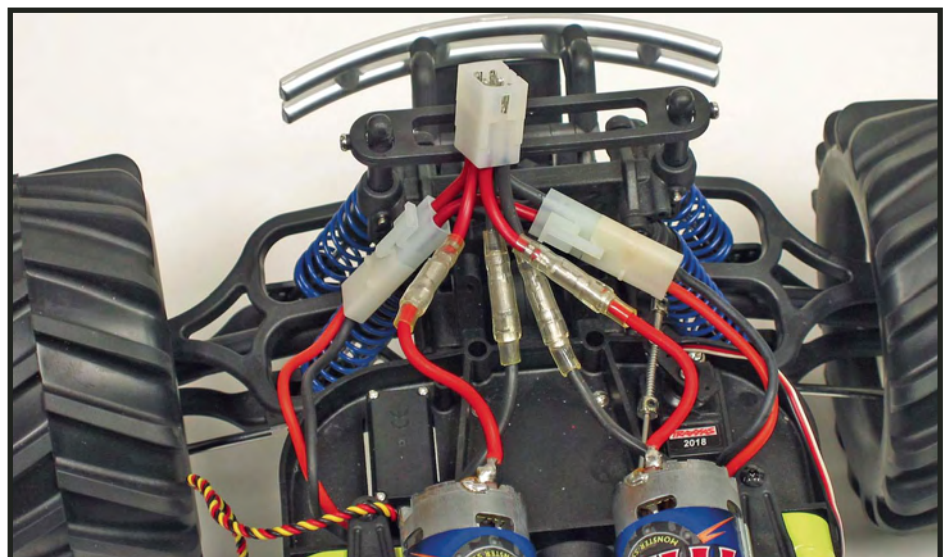


Figure 3. A pair of HB-25s mounted to the deck and wired to the Servo Module.



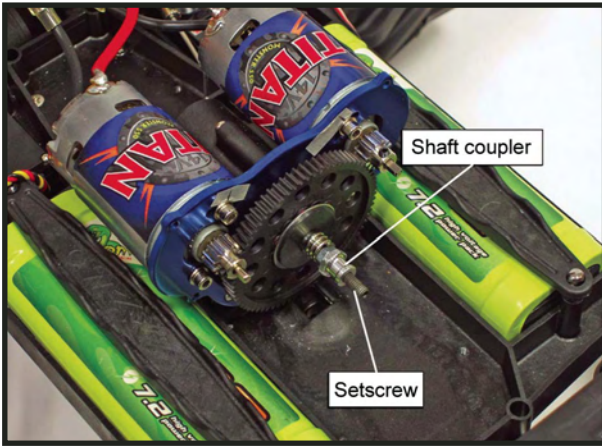


Figure 5. The center transmission shaft extended with coupler and set screw.

the top of the deck increased the amount of wiring between the deck and the chassis. In order to make the wiring more manageable, I used a six-wire 14 gauge connector. It worked so well that I plan on using a similar multi-pin connector for the servo and encoder wiring.

After installing the motor controllers on the deck of the E-Maxx, unscrew and remove the ESC and the radio receiver to reduce weight and free up some space on the chassis. The HB-25 motor controllers greatly expand the speed range of the E-Maxx. It can now crawl along at a snail's pace, race ahead like a rabbit, and hit any speed in between.

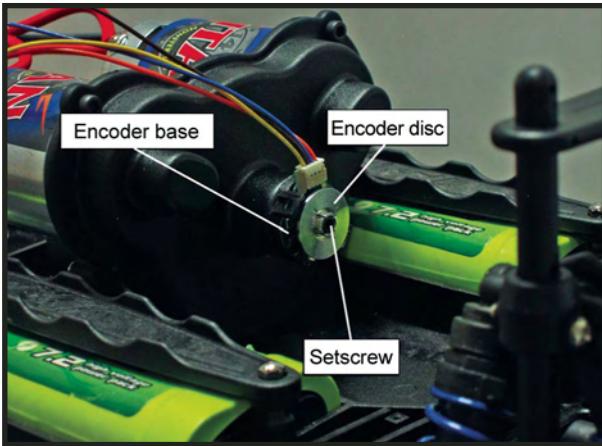


Figure 6. The encoder disc mounted with the lines facing in towards the encoder base.

the capacitor housing on the HB-25s. Trim the spade ends if necessary.

Mounting the motor controllers to

Adding the Rotational Encoder to the Transmission

Now, to monitor the speed, we

simply add an encoder. It would be very useful to know the rotational speed of the E-Maxx transmission. That way, we would be able to vary power to the motors in response to changes in the terrain. To measure transmission rotation, I added a rotational encoder, which translates motion into a stream of electrical pulses.

Knowing the rotational speed of the transmission is not always sufficient, though. For example, if the E-Maxx was on a steep incline, it is possible that it could be moving in a direction opposite of the power applied, especially now that we are able to supply very small amounts of power to the drive motors.

In this situation, I may not be applying enough power to overcome gravity, and our encoder would be reporting movement ... in the wrong direction. To solve this problem, we need a quadrature encoder to know the direction in which the transmission is rotating.

US Digital produces a wide range of optical encoders. I ordered an E4 — a miniature rotary optical encoder with quadrature that fits on the 4 mm drive shaft of the E-Maxx. The configuration I ordered has a shaft hole in the housing that allows the shaft to go through the encoder, so I do not have to be concerned with shaft length. I found that an E4 with transfer adhesive base mounting and a 100-count setscrew hub works perfectly.

To add the rotational encoder:

1. Remove the transmission case and the rubber plug on the case that covers the shaft hole. This will expose the center transmission shaft.
2. Remove the locknut and replace it with a thin 4 mm nut, being careful not to let the clutch plate separate.
3. Tighten the nut all the way and then back it off a quarter turn.

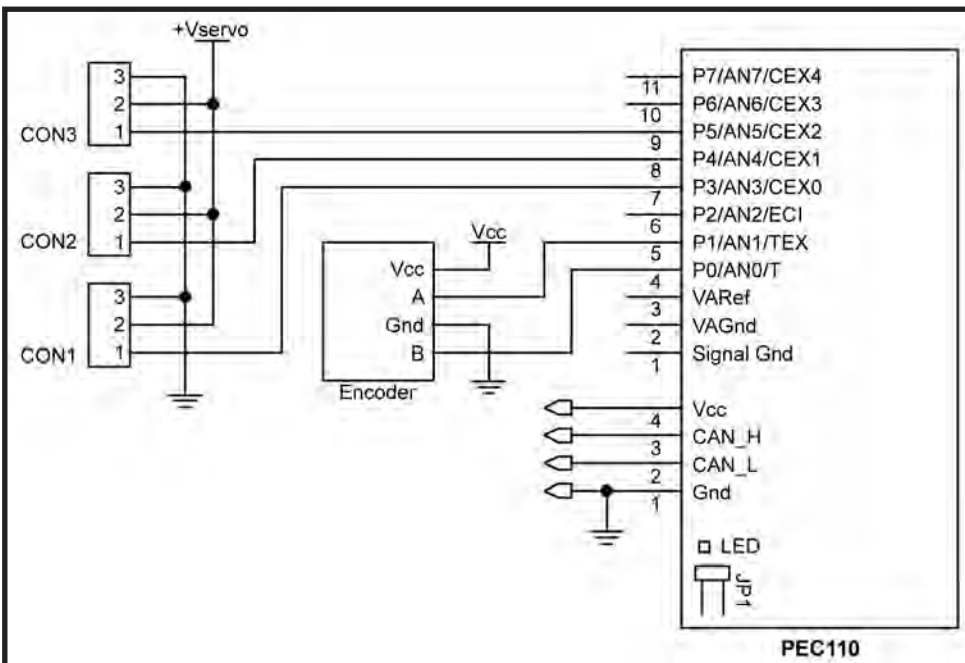


Figure 7. Schematic of the servo and encoder control circuitry.

Figure 8. An illustration of the Servo Module showing how to combine two independent five-volt regulated supplies, three servo ports, and an encoder on one small breadboard.

4. On the end of the center shaft, thread the coupler until it touches the nut, with the wider side facing inward so as not to touch the transmission cover plate. The end of the center shaft should thread half-way through the coupler. If it does not, try adding a washer or two to the shaft between the coupler and the nut to position it correctly.

5. Screw a 4 mm set screw into the coupler to extend the center shaft.

6. Replace the transmission cover.

7. Slip the centering tool over the shaft.

8. Peel the paper backing off of the encoder base and slide it down the centering tool. Make sure to apply pressure evenly until it adheres to the transmission cover plate.

9. Remove the centering tool and slide the encoder disc onto the shaft until it bottoms out against the spacer.

10. Tighten the setscrew.

11. Spin the shaft a few times to make sure the encoder disc is aligned properly.

12. Place the housing cover on the encoder and attach the connector.

The encoder is wired to the servo module, as shown in Figure 7, to process the encoder pulses. The A and B lines from the encoder are connected to the PEC-110's up/down counter and its power is connected to a five-volt supply.

In the last article, power regulation was on the Communications Module. I've

consolidated all of the power regulation circuitry to the Servo Module to avoid having to run multiple unregulated lines. I've also added a switch for the electronics and another for the motor controllers.

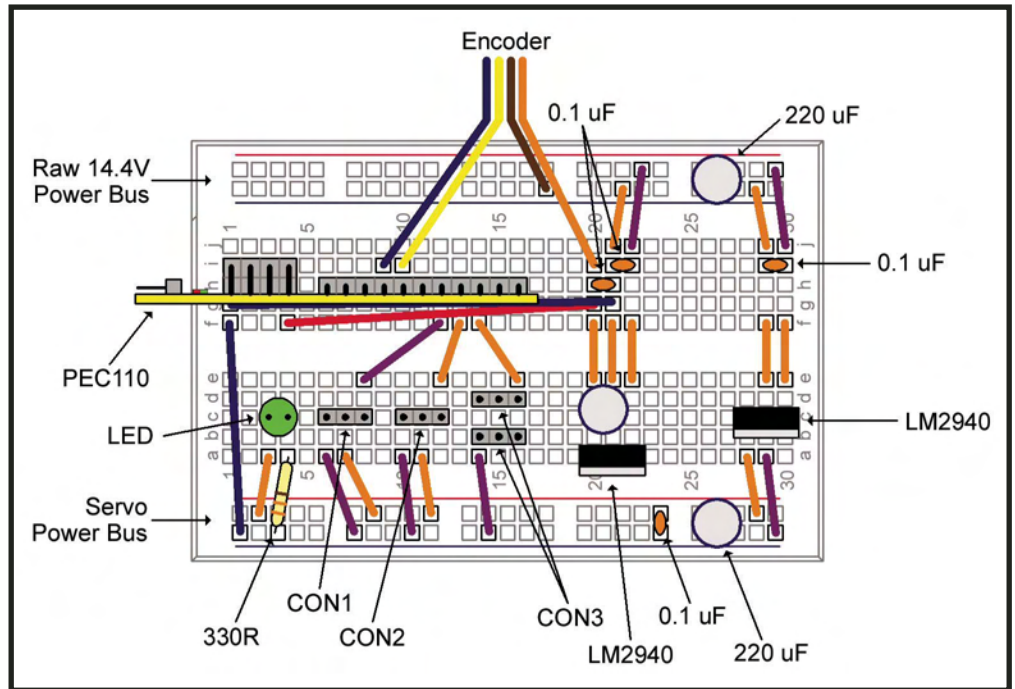
Now, all that's left is the software.

Controlling the Velocity of the E-Maxx

Supplying a constant amount of power to the motors will produce dif-

ferent speeds under different conditions. For example, it takes more power to move up an incline or through sand than it does to move across a hard, level surface. To maintain a constant velocity, the information from the encoder can be used to adjust power to the motors in response to changes in terrain.

The eight-pin port extender (PEC-110 by Machine Bus) I chose in the March article has a closed-loop control mode that will make the necessary



LISTING 1

```
// Create a new closed loop motor reference
EncodedMotor encodedmotor_createEncodedMotor(CommBus C, uint8_t id);

// Run the motor at the specified rate until commanded otherwise.
void encodedmotor_setRate(EncodedMotor E, int16_t rate);

// Run motor at the specified rate until its incremented/decremented count times.
void encodedmotor_setRelativePosition(EncodedMotor E, int16_t rate, int32_t count);

// Run the motor at the specified maximum rate until the encoder reached count.
void encodedmotor_setAbsolutePosition(EncodedMotor E, int16_t rate, int32_t count);

// Clear the count
void encodedmotor_resetCount(EncodedMotor E);

// Get the current count
void encodedmotor_getCount(EncodedMotor E);

// Get the current rate
int16_t encodedmotor_getRate(EncodedMotor E);

// Dispose of the encoded motor
void encodedmotor_disposeEncodedMotor(EncodedMotor E);
```

PARTS LIST

Item/Description

- ⇒ (2) Parallax HB-25 motor controllers
- ⇒ 14 gauge silicone insulated wire
- ⇒ 14 gauge bullet connectors
- ⇒ NiCad battery connectors (two pairs)
- ⇒ Six pin connector
- ⇒ 14 gauge spade connectors
- ⇒ 22 gauge spade connectors
- ⇒ E4-100-157-HT miniature optical kit encoder and cable
- ⇒ Molex .1" female housing, pins, and male header
- ⇒ 4 mm coupler
- ⇒ 4 mm hex head setscrew
- ⇒ 4 mm nut

Supplier

- Parallax, Inc. (www.parallax.com)
 Al's Hobby Shop (www.alshobbyshop.com)
 Al's Hobby Shop (www.alshobbyshop.com)
 Al's Hobby Shop (www.alshobbyshop.com)
 Jameco (www.jameco.com) Part No. 142228CE
 Any hardware store
 Any hardware store
 US Digital (www.usdigital.com)
 Jameco (www.jameco.com)
 Tower Hobbies (www.towerhobbies.com has it listed as 'Trinity Shock Spacer Nitro Spyder')
 Any hardware store
 Any hardware store

firmware on the PEC-110 measures the difference between the desired rate set point and the actual shaft rate. The velocity control algorithm on the PEC-110 uses that error to adjust the power to the motors.

With the encoder attached, the application software on the PC – to interface to an encoded motor – changes from “Servo” to “EncodedMotor,” as shown in Listing 1.

I have written a few applications to take advantage of velocity control. One

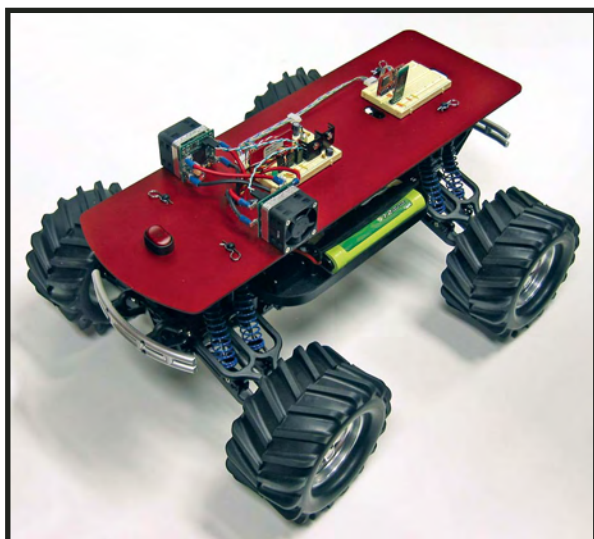
application maintains a constant velocity while driving across grass or pavement, as well as while traveling up and down inclines. Another application drives the E-Maxx 20 feet straight towards a wall and stops just short of it. I can then use the joystick application from the March article to drive the E-Maxx back to me without leaving my chair.

Next Up: Navigation

The modifications to control the speed of the E-Maxx are complete. The HB-25s provide fine-grained control over the motors, and the encoder helps determine how fast the E-Maxx is moving in forward or reverse. As a direct result of our modifications, the E-Maxx can:

- Maintain constant speed under all conditions.
- Move at very low speeds.
- Record the distance traveled.
- Move a specified distance and velocity.

In the next article, we will tackle dead reckoning. The information returned from the encoder will be used in odometry calculations. We will also add a digital compass to the E-Maxx so it knows exactly which direction it is heading. To test the hardware modifications and the software, we will have the E-Maxx traverse a set of waypoints using the digital compass and odometry. **SV**



power adjustments in response to the encoder data. By adding an encoder to the PEC-110, I can now drive by velocity.

In other words, instead of specifying a power level using servo position commands, I can send the Servo Module the desired rate for the motor shaft to rotate in counts per second. The


Figure 9. The E-Maxx with motor controllers and encoder, ready to go.

BUDGET ROBOTICS

THE BIG GRIPPER

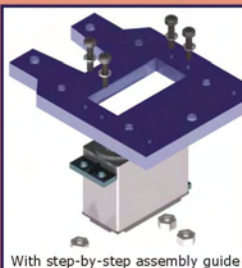
SERVO-OPERATED GRIPPER WITH THREE-INCH FINGERS

- Fingers open to over 3" wide
- Rubber finger tips for extra traction
- Available with or without servo
- Included servo works with microcontroller or RC receiver
- Solid construction using injected-molded plastic for fingers, and 1/4" expanded PVC
- Easy to mount to your robot or another servo to create a wrist or arm
- Versatile design
- Build in 20 minutes or less



Just \$24.95!

Version without servo only \$14.95



With step-by-step assembly guide

GREAT ROBOTS. GREAT PRICES.

WWW.BUDGETROBOTICS.COM