

P.A.R.T.S

Portland Area Robotics Society

Issue # 01 By. Marvin Green (503) 656-8367.

I would like to start a **Robotics club here in Portland Ore.** If you think that's a good idea, and you want to get together with other robot hobbyists in the Portland area, give me a call!

My Robot Zippy had a chance to run in the Y.O.R. line following competition. It was a great place to exchange ideas and put robots to the test. Zippy was one of the smallest robots there, but that didn't stop him from showing off. One of the most asked questions about Zippy was about the drive motor. I used two modified airplane servo's to drive my robot for a few basic reasons:

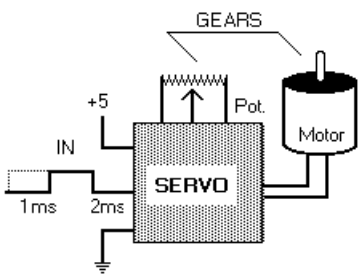


Fig. 1

1. All the drive circuitry is built in. (No Relays etc.)
2. Low power, and connects to one I/O line.
3. Low cost. (\$11.00 per servo.) Easy to find.
4. Pulse width controls speed and direction.
5. Only a few lines of code to run. (**Forth on 68HC11**)
6. Servo is easy to modify.

The R/C servo is light, durable and strong, perfect for a model airplane or a small robot.

Basically, a servo works by reading an incoming pulse width signal. The servo measures the pulse width (1ms to 2ms) and compares it with an internal pulse, (controlled by a potentiometer). The servo controls the motors speed and direction that adjusts the connected potentiometer.

When the internal pulse (potentiometer) and the external pulse (input signal) are equal, then the motor stops. This is a bit over simplified, but ... See figure 1.

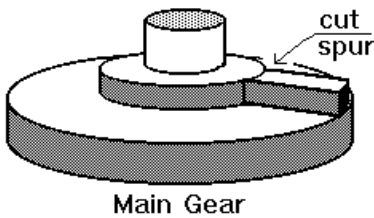


Fig. 2

The servo can turn $\cong 270$ degrees. To modify it for continuous rotation cut off the stopping spur on the main gear, and disconnect the potentiometer. Open up the gear box taking out the four small screws. Locate the biggest gear, this gear is also part of output shaft, and underneath is the potentiometer. Using a razor blade, cut off the spur, so that it can rotate freely. See figure 2.

When putting the servo back together, apply some grease to the gears.

The potentiometer should be removed, and replaced with 2 resistors. For example my potentiometer had a value of 10K, I replaced it with two 5k resistors. Doing this makes the internal pulse width a constant. Now the input pulse can adjust the motor speed and direction. See Figure 3.



The input of the servo can be an output pin of the CPU. The 68HC11, can control 4 servos directly with only its built in hardware, no software timing loops! Writing the code in FORTH makes controlling a servo simple. (See code.) Also refer to Motorola's 68HC11 specs.

The hardware is setup to create a pulse 30 times a second with OC1, then OC4 and OC5 act as a one-shot to create the variable pulse width. In your code, you must find correct value to center the servo. (Servo not turning in either direction, but centered in the 'dead band').

For example the centering value for the right servo may be (HEX) A000. To turn the servo slowly CCW enter A010, to turn slowly CW enter BF0F.

This is just an overview to show how easy running a servo from a computer can be. Refer to other servo articles in Radio Electronics, the Computer Journal, etc. Experiment, and have fun.

```

: INIT-SERVO          ( Two servos connected to PA3 and PA4      )
  18 B00C C!         ( Effect OC4 OC5 on OC1                    )
  00 B00D C!         ( OC4 OC5 set 0 on OC1                     )
  0F B020 C!         ( OC4 OC5 set 1 on compare                 )
  00 B016 ! ;       ( TIC1 set to 0 = OC1 - Trigger             )

: RIGHT_SERVO B01E ! ; ( 0 - FFFF CONTROLS PULSE WIDTH )
: LEFT_SERVO B01C ! ; ( 0 - FFFF CONTROLS PULSE WIDTH )

```

Simple code sample. (FORTH) * Remember that FORTH is cool.

This is my first attempt at a news letter, or trying to organize a Robotics Club. I hope that it starts the ball rolling in getting some robotics interest going here in Portland

area. If any part of robotics interest you, theory, design, programming, building, AI, Sci Fi, or just thinking about them. Please call or write.

Marvin Green 637 Hartke Loop, Oregon City, OR 97045 (503) 656-83687