

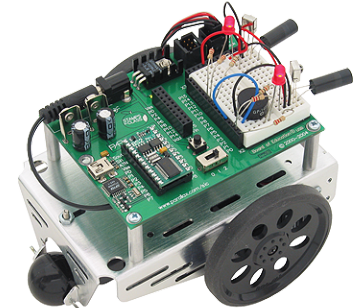
The Parallax Bot

Using the Parallax Bot

The Parallax Bot comes mostly preassembled for you. However, we need to test and calibrate the operation of this bot!

Step-by-step instructions are in the Parallax Bot User Manual at the link below

<http://learn.parallax.com/node/125>



Test the Servos

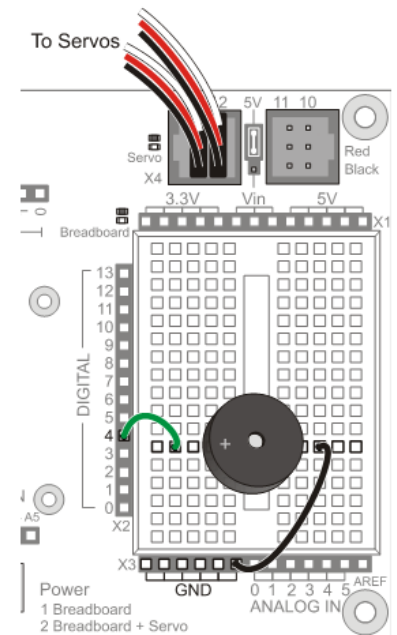
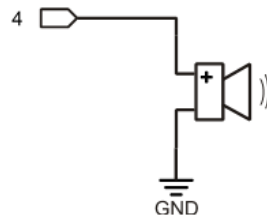
- Set the BOE Shield-Bot on its nose so that the drive wheels are suspended above the ground.
- Connect the programming cable and battery pack to the Arduino.
- Verify and upload **RightServoTest** to your Arduino
- Set the 3-position switch to position-2 and press/release the **RESET** button.
- Verify that the right wheel turns clockwise for three seconds, stops for one second, then turns counterclockwise for three seconds.
- Now do the same using the **LeftServoTest** program

Add the Start/Reset Indicator

Imagine your BOE Shield-Bot is navigating through a maze, and suddenly it stops for a moment and then goes in a completely unexpected direction. How will you know if it is a mistake in your code, or if it is losing power? One simple, effective solution is to add a speaker to your BOE Shield-Bot and make it play a “start” tone at the beginning of every sketch.

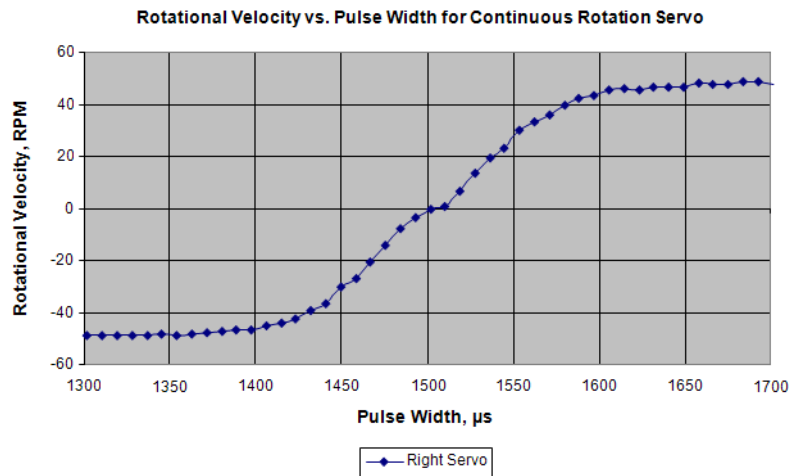
- Obtain the piezospeaker and jumper wires
- Assemble the speaker according to the diagram on the right
- Add the following command to your program

```
tone(4, 3000, 1000);  
delay(1000);
```

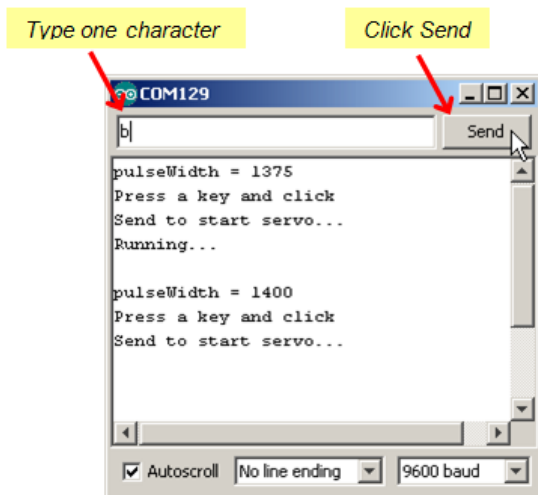


Test Speed Control

This graph shows pulse time vs. servo speed. The graph's horizontal axis shows the pulse width in microseconds (μs), and the vertical axis shows the servo's response in RPM. Clockwise rotation is shown as negative, and counterclockwise is positive. This particular servo's graph, which can also be called a *transfer curve*, ranges from about -48 to +48 RPM over the range of test pulse widths from 1300 to 1700 μs . A transfer curve graph of your servos would be similar.



- Place a mark (like a piece of black tape) on the wheel so that you can see how revolutions it turns during the wheel speed tests.
- Set the Parallax Bot on its nose so that the wheels can spin freely.
- Verify and upload **TestServoSpeed** program to the Arduino.
- Open the Serial Monitor, and set the drop-down menus to **“No line ending”** and **“9600 baud.”**
- Click the transmit pane at the top, type any character, and click the Send button.
- Count the number of turns the wheel makes, and multiply by 10 for RPMs. (Don't forget to make a note of direction; it will change after the 5th test.)
- If you were to add your data points to the graph, would they fit the overall shape?



Change the for statement in TestServoSpeed from:

```
for(int pulseWidth=1375; pulseWidth <= 1625; pulseWidth += 25)
```

to

```
for(int pulseWidth=1300; pulseWidth <= 1700; pulseWidth += 20)
```

Load the modified program into the Arduino and use it to fill in every other table entry.

Use Excel to plot the pulse width vs. wheel RPM.

Calculating Distances

You might remember asking your parents this question, over and over again, while on your way to a vacation destination or relatives' house:

“Are we there yet?”

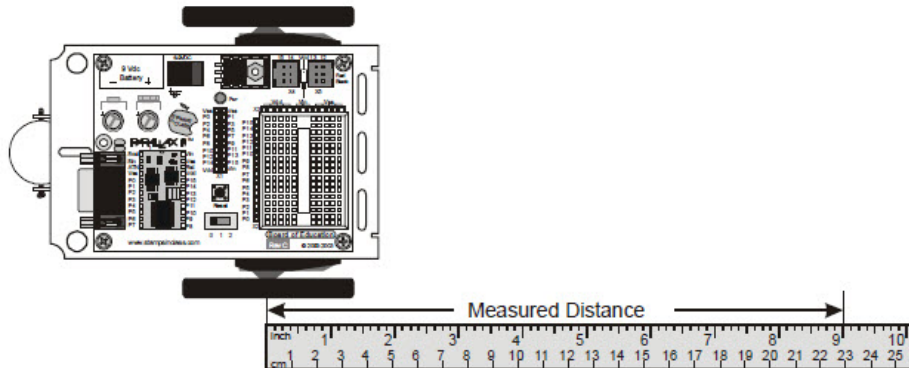
Perhaps when you got a little older, and learned division in school, you started watching the road signs to see how far it was to the destination city. Next, you checked the car’s speedometer.

$$time = \frac{distance}{speed}$$
$$time = \frac{140 \text{ miles}}{70 \text{ miles/hour}}$$
$$= 140 \text{ miles} \times \frac{1 \text{ hour}}{70 \text{ miles}}$$
$$= 2 \text{ hours}$$

You can do the same exercise with the BOE Shield-Bot, except you have control over how far away the destination is. Here’s the equation you will use:

$$servo \text{ run time} = \frac{BOE \text{ Shield - Bot distance}}{BOE \text{ Shield - Bot speed}}$$

- Verify and run **ForwardOneSecond**.
- Place your Parallax Bot next to a ruler or marks on the floor
- Make sure to line up the point where the wheel touches the ground with the 0 in/cm position on ruler.



- Press the Reset button on your board to re-run the program.
- Measure how far your Parallax Bot traveled by recording the measurement where the wheel is now touching the ground here:_____.
- The distance you just recorded is your Parallax Bot speed, in units per second. Now, you can figure out how many seconds your Parallax Bot has to travel to go a particular distance.