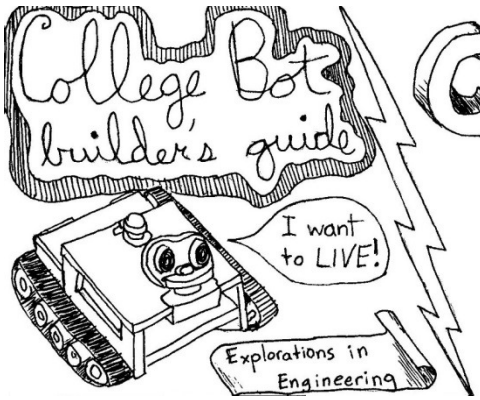


CHAPTER: ROBOT CONSTRUCTION 011: FINAL CONSTRUCTION

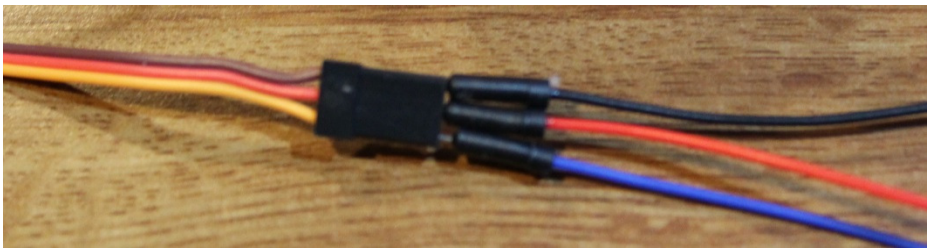
1. Prepare your parts

At this point, the robot should be mostly assembled. We'll just be adding a few wires and, eventually in the final step, throwing in the speaker shelf with 4 ½" bolts and 4 nuts to connect the top to the base.

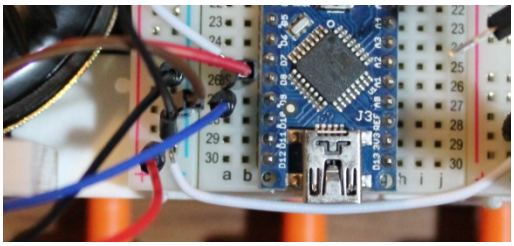
2. Connect and test the micro-servo (turret)

Assembly

First, lift the top off and set it aside (resting on the idlers on one side.) Then, find the cord running out of the microservo and insert 3 short flexible jumper wires into the three slots in the end as shown in this picture. The servo wires are color coded with the purple or brown wire being Ground (GND), the red wire being positive power (VCC), and the yellow wire being the control signal (SIG). We now need to connect these three wires to the breadboard.



- The wire going from GND should connect to –L (ground on the breadboard)
- The wire going from VCC should connect to +L (+6V on the breadboard)
- The wire going from SIG should connect to a27 (D9 on the Arduino Nano)



After the connection is made, put the Top back on loosely and push the Head onto the gear on the micro-servo.

Software

Connect the Arduino Nano to your computer and use the Arduino software to load the "ORPC_Step_2" program. After the upload is complete, detach the mini-USB cable from your Arduino Nano.

If you have time, look at the code used in this step and try to understand what it will do prior to moving on to the next step.

Test

Turn the robot on by flipping the power switch. It should do the following things in order:

1. Blink the LED for a second
2. Play “Shave & a Haircut”
3. Blink the LED for two seconds
4. Center the turret
5. Turn the turret left
6. Center the turret
7. Turn the turret right
8. Center the turret
9. Wait five seconds
10. Repeat at #1

If none of these things happen, go back and double-check the breadboard connections.

Calibration

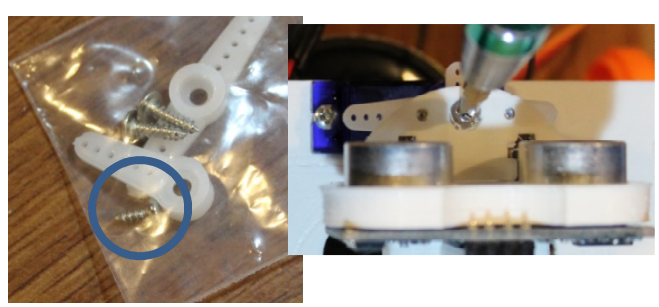
During the “wait five seconds” portion of the program, the face should be looking directly forward. If this is not the case, turn the robot off in the middle of that portion and then replace the face so that it is as close to centered as possible. This should be possible within a few degrees of center.

If you’d like to make sure the head is perfectly centered, you’ll need to adjust the following line in the code:

```
int turretCenter = 90; //set to (80-100) instead of 90 because arm of servo is
a little off center
```

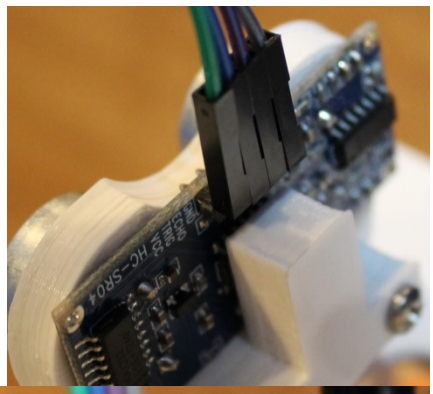
Note the direction off center prior to adjusting this code. Then, change the part that says “= 90” to “=85” and re-upload the program to the robot. Now see if it goes more off center or less (and whether it is now off center in the opposite direction.) Based on these observations, continue adjusting this number until the head is completely centered.

Finally, once calibrated, you’ll need to attach the head to the servo-motor with the small screw that was included in the servo-motor bag. Insert the screw into the top of the large hole in the Head Base and screw it into the hole in the servo-motor gear until the screw is tight. As you’re tightening the screw, gently hold the Head to make sure it doesn’t slip out.



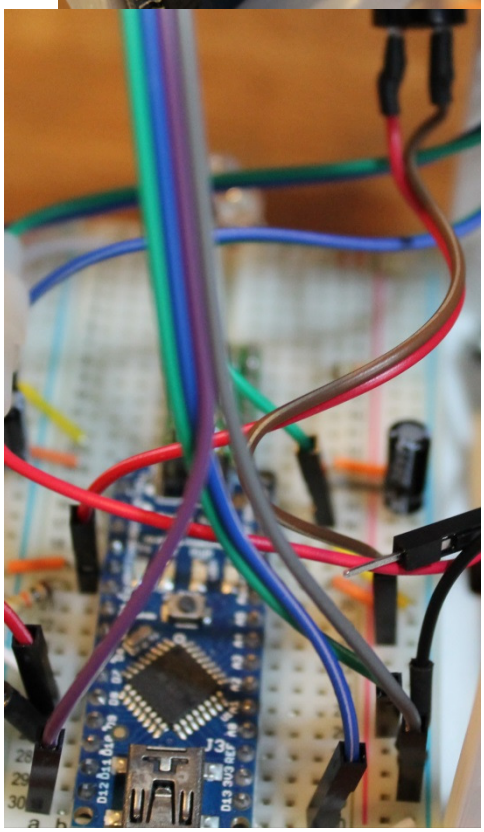
3. Connect and test the sonar sensor (eyes)

Assembly



Take the 4x Male-Female wire (P) and insert the female end (the part that accepts pins) onto the pins on the top of the sonar sensor. As you can see from the small labels printed on the back, the four pins from left to right (as viewed from the back) are "VCC" (power), "TRIG" (trigger or send a sonar ping), "ECHO" (listen for a sonar response), and "GND" (ground). Take note of which color wire is connected to which pin.

Now lift the Top off the robot body and set it sideways on the idlers such that you can still plug the pins from the sonar sensor into the breadboard. You'll need to connect all four in the following fashion:



- The wire coming from "VCC" should be connected to +R (+5V power on the breadboard)
- The wire coming from "TRIG" should be connected to j30 (D13 on the Arduino Nano)
- The wire coming from "ECHO" should be connected to a30 (D12 on the Arduino Nano)
- The wire coming from "GND" should be connected to -L or -R (ground on the breadboard)

In addition to connecting the sonar sensor, in this step, we will also connect the red button:

- Connect one lead (either one—the button will simply close the circuit when pressed, so either side is equivalent) to b20 (D2 on the Arduino Nano)
- Connect the other lead to +R (+5V power on the breadboard)

Software

Put the Top back on loosely. Then connect the Arduino Nano to your computer and use the Arduino software to load the "ORPC_Step_3" program. After the upload is complete, detach the mini-USB cable from your Arduino Nano.

If you have time, look at the code used in this step and try to understand what it will do prior to moving on to the next step.

Test

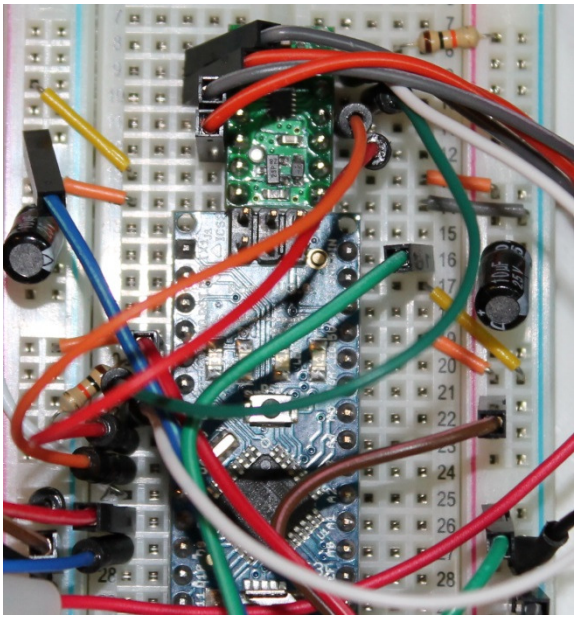
Turn the robot on by flipping the power switch. First test the sonar sensor—if you hold your hand in front of the eyes, it should make a sound. Check to see if the robot plays different sounds when your hand is 12-24 inches, 6-12 inches, and less than 6 inches away from the eyes. If this step does not work, double check the connections to the sonar and the speaker.

Next, test the button. After the first press, it should blink the LED a couple of times and play "Shave and a Haircut." After the second press, it will play a longer song, periodically moving the head from side to side. A third press will cause it to repeat the same behavior as the first press. If any of these actions fail to happen, double-check the relevant connections.

4. Test motors and motor driver

Assembly

In this step, we'll be connecting the motor driver to the Arduino Nano and the motor driver to the motors. (For more details on the motor driver behavior, see the appendix at the end of this document.) First, lift the Top off the robot body and set it on the idlers. Then, insert four flexible jumper wires into the breadboard to make the following connections:



- a21 (Arduino Nano D3) to g10 (AEnable on the motor driver)
- a22 (Arduino Nano D4) to g9 (APhase on the motor driver)
- a23 (Arduino Nano D5) to g12 (BEnable on the motor driver)
- a24 (Arduino Nano D6) to g11 (BPhase on the motor driver)

After you've connected the Arduino Nano to the motor driver, you need to plug the motors in. Make the following connections:

- Left motor black wire to d9 (AOut1 on the motor driver)
- Left motor white wire to d10 (AOut2 on the motor driver)
- Right motor black wire to d11 (BOut1 on the motor driver)
- Right motor white wire to d12 (BOut2 on the motor driver)

After these connections are made, loosely place the Top back on top of the robot base.

Software

Connect the Arduino Nano to your computer and use the Arduino software to load the "ORPC_Step_4" program. After the upload is complete, detach the mini-USB cable from your Arduino Nano.

If you have time, look at the code used in this step and try to understand what it will do prior to moving on to the next step.

Test

Turn the robot on by flipping the power switch. First make sure the sonar sensor still reacts as it did in the previous section, with different sounds based on how far away the detected object is.

Second cycle through each of the following steps by pressing the button and then waiting for the step to complete before pressing the button again:

1. Play “Shave and a Haircut”, blink LED
2. Drive forward for one second
3. Drive backward for one second
4. Turn left for one second (the left motor/gear should turn backwards and the right motor/gear should turn forwards)
5. <Repeats 1-4>

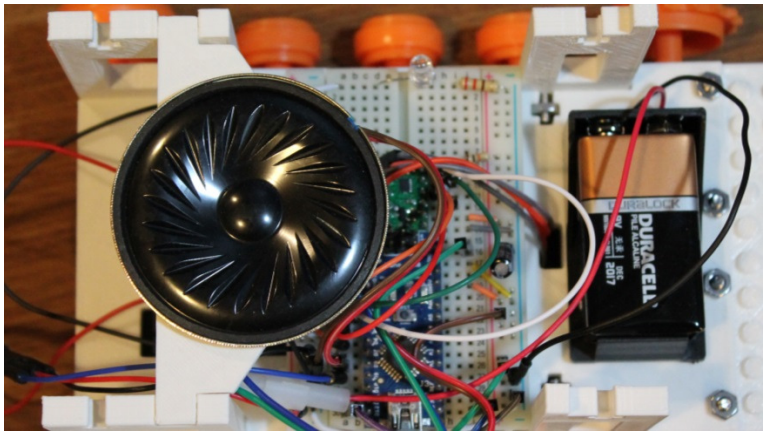
If the motors do not behave as expected, you will need to change their connections to the motor driver:

- If they run backwards (e.g. backward in step 2 and forward in step 3) swap the pins within the same motor—d9 swapped with d10 and d11 swapped with d12
- If they run fine in steps 2 & 3 but turn right instead of left, swap the pins to the different motors—d9 swapped with d11 and d10 swapped with d12
- If there is some combination of the above, double check that the connections from the Arduino Nano to the motor driver are correct and then try to swap the motor pins until they work correctly.

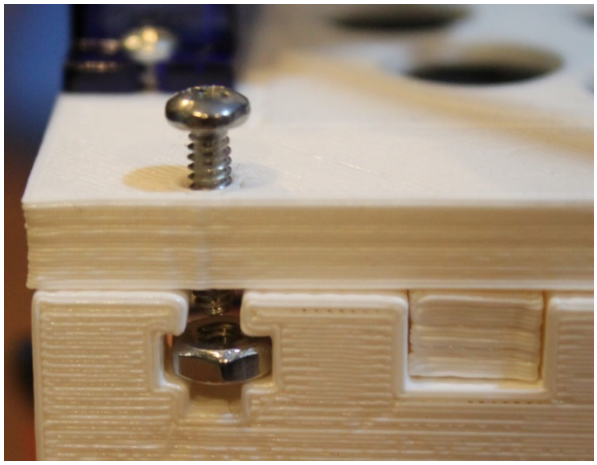
5. Final Assembly

Chassis assembly

Remove the Top and set it aside. Then place the Speaker Shelf into the notches on the two sides and place the speaker on top.



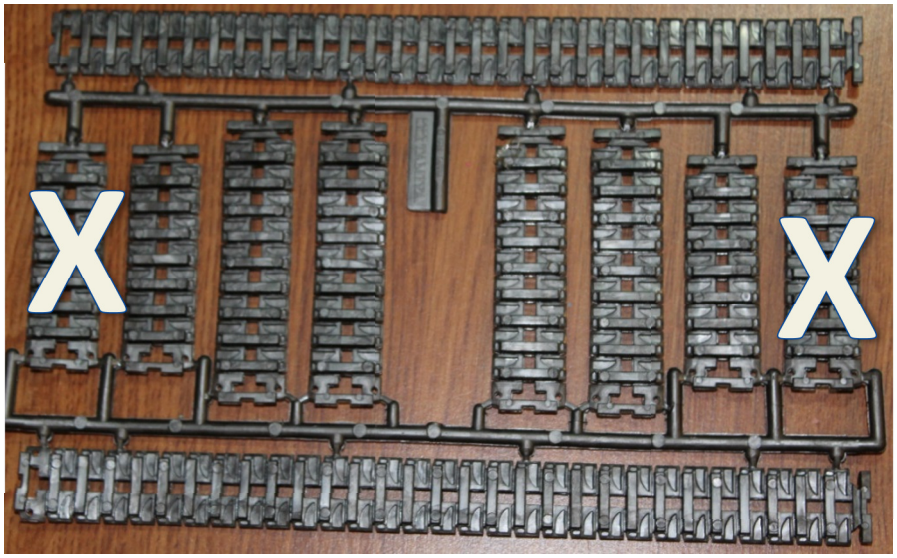
Slowly lower the Top into place—as you do so, the speaker should center into the cutout on the bottom of the Top. Now take 4 ½" bolts and 4 nuts to fasten the Top to the rest of the robot. You'll need to gently slide the nuts unto the crossing portion of the "T" slots before threading a bolt through the top.



As you're tightening all 4 bolts, gently adjust the sides in or out to make sure the assembled robot is "square." When complete, both sides will look like the following:

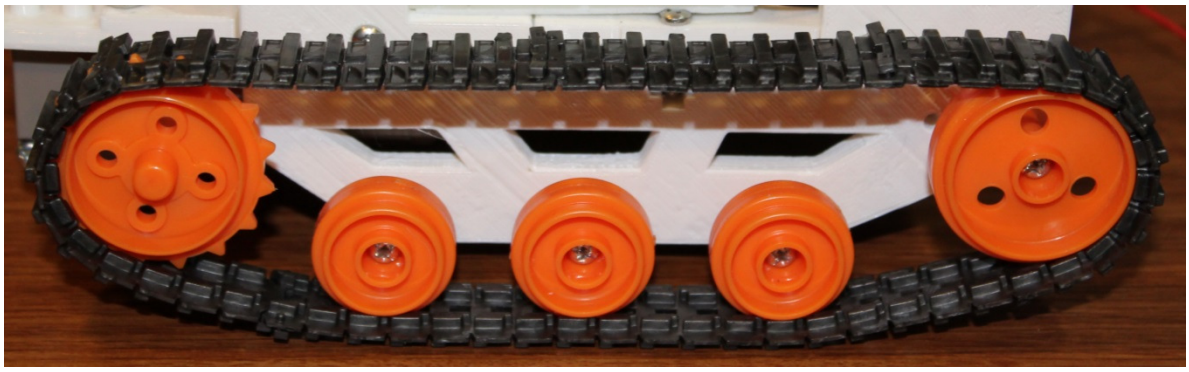
Tread assembly

Take the rubber treads from the Track & Wheel Set and remove all but two of the treads (shown with an "X" in the picture below) with a set of wire cutters.



You'll now need to assemble two separate loops of treads, each with one long strip, two medium strips, and one short strip. The pictures below show details for how to connect one chain link to the next.

After the tread is assembled, slip it over the idlers first and then slowly stretch it over the gear until the tread is completely in place. Repeat for both sides.



Software

Connect the Arduino Nano to your computer and use the Arduino software to load the “ORPC” program. After the upload is complete, detach the mini-USB cable from your Arduino Nano.

If you have time, look at the code used in this step and try to understand what it will do prior to moving on to the next step.

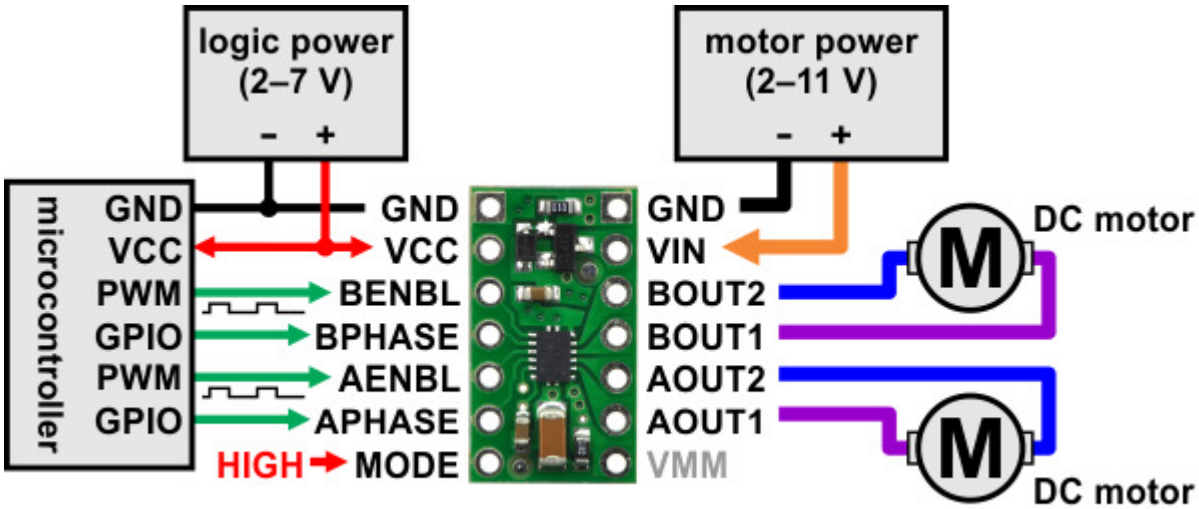
First, however, PLAY WITH YOUR COMPLETED ROBOT!!

Appendix: About the DRV 8835 motor driver carrier

Background on Motor Drivers and "driver carrier"

A motor driver is special set of chips that lets you turn a high current motor on and off using a low current microcontroller. In order to do this, it needs two sources of power—the "logic power" that is the same power connected to the microcontroller, and the "motor power" that is only connected to the driver.

Basic connection diagram



In our model, this is flipped 180 degrees with "VMM" in the upper left and "MODE" in the upper right.

In our program, we'll control it with the following commands:

Simplified drive/brake operation with MODE=1 (PHASE/ENABLE)				
xPHASE	**xENABLE**	**xOUT1**	**xOUT2**	**operating mode**
1	PWM	L	PWM	reverse/brake at speed *PWM %*
0	PWM	PWM	L	forward/brake at speed *PWM %*
X	0	L	L	brake low (outputs shorted to ground)