Lab 6-Integration

## Learning Outcomes:

* System integration
* Troubleshooting

## Background:

Up to this point we’ve only tested individual systems of the AutonoMouse (ie servos, ultrasonic range finder, PCB) today we’ll be integrating these subsystems into a whole. This process is called system integration. Integration can sometimes be frustrating because you might not know what system or component is not working. The best thing to do is be patient and logically troubleshoot one thing at a time until you can determine the thing that is not working.

### Part 1 Build:

First thing we’ll do is attach all our components to our acrylic base mechanically. Then we’ll start connecting the electronics.

1. Use a Phillips screwdriver to poke a hole through the center of your wheels
2. Use hot glue to glue your wheels to your servo mounts
3. Add the screw provided with your servos for extra support
4. Put a dot of hot glue around the circumference of your wheels about every 1cm (this helps the wheel maintain traction
5. Hot glue and zip tie your servos to the two wings of your board.
6. Zip tie your battery to the end of the board. Make sure that you insert the zip in the correct manner to create a caster. (From the bottom insert it through the hole closest to the edge) Check that it’s inserted correctly before tying it

Insert Picture of completed car mechanics

You may have noticed we haven’t attached the PCB yet. This is because we’ll be testing electronics next, and we might need access to the bottom of the PCB for troubleshooting and we don’t want it permanently on the board yet.

### Part 2 Electronics:

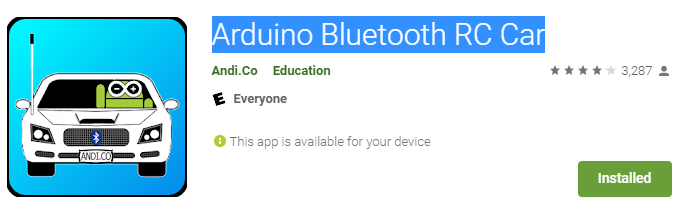
Next, we’ll attach the electronics to the PCB, upload some code, and see if it works.

1. Run the servo wires through the hole next to the servo, diagonally across the bottom of the board, then back over to the 3 header male pins. Make sure to attach the left servo to the left servo pins and vice versa
2. Upload the code provided for this lab
3. Unplug the Arduino from your device, flip the switch, and see if your car is operating correctly. (See Below)
4. If everything is working correctly (check that it is, it will be hard to remove your PCB) add some double-sided foam tape to your PCB and attach it to the base.

#### Operating Instructions

Note: this will only work with Android Phones.

First, download the app Arduino Bluetooth RC Car from the Play Store.



Next, we will connect the car with our phone via the new app. Make sure Bluetooth is on. Connect power to the esp32. In the app, select the gear button, then “Connect to car”. Once you do so you should see the red light in the left corner of the screen turn green indicating a proper connection.

Once connected, the car will start in an autonomous state. The car has two states. 1) autonomous, 2) remote control. To put the car into the remote-control state, click the triangle button near the top of the app. To switch back to the autonomous state, select the triangle button again.

Autonomous mode: the car drives forward until detecting an object. It will then turn 90 deg in each direction, determine which distance is longer via ultrasonic module, then drive in that direction.

Remote Control mode: This puts the car into remote control mode. You can move the car around by using the Bluetooth RC Controller app on the Play Store. (Only available for android phones). Connect by clicking the gear icon, then “Connect to car”. A list will appear. Select “ESP32”. Your phone and ESP32 must be paired via Bluetooth.

To stop the car from moving, put it into remote control mode and don’t press anything, disconnect it from Bluetooth, or flip the power switch on the PCB.

That's it, you've finished this course. Congratulations!

PICTURE of completed project

**Extra:**

Try changing the code to make the car do different things.