Build an Arduino ROV

https://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p052/robotics/arduino-underwater-ROV
(https://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p052/robotics/arduino-underwater-ROV)

Procedure PDF Date: 2023-10-26

Experimental Procedure

Note: This engineering project is best described by the **engineering design process,** as opposed to the **scientific method.** You might want to ask your teacher whether it's acceptable to follow the engineering design process for your project before you begin. You can learn more about the engineering design process in the Science Buddies Engineering Design Process Guide (http://www.sciencebuddies.org/science-fair-projects/engineering-design-process/engineering-design-process-steps).

1. *Before you purchase any parts or start building anything*, you should come up with a preliminary design for your ROV. It is OK if you end up changing your design later. While it can be tempting to start building something right away, especially when you are excited about a project, you should carefully go through the steps of the engineering design process (http://www.sciencebuddies.org/science-fair-projects/engineering-design-process/engineering-design-process-steps), which include defining the problem, doing background research, specifying requirements, and brainstorming solutions before you get to the prototyping stage. This is not an exhaustive list, but here are a few questions to help get you started when designing your ROV.
    a. Where do you plan to test and use your ROV? Can you do initial tests in a bathtub or other large tub of water? Do you eventually plan to use it in something larger, like a pool or a pond? We recommend avoiding bodies of water with strong currents.
    b. Will you always be able to see your ROV? Will you need a real-time video feed in order to operate it?
    c. What do you want the range of your ROV to be? How long should the tether be?
    d. What information do you want to gather with your ROV? Do you want to record still images or videos? Record environmental data, like temperature or light levels? Gather water or sediment samples?
    e. Do you want your ROV to interact with the environment somehow — for example, by grasping or scooping things up?
    f. What equipment is required to do the things you listed in points d and e?
    g. How big does your ROV need to be to fit all the required equipment?
    h. How will your ROV maneuver? Do you want to make it buoyant enough that it always floats on the surface? Heavy enough that it sinks and skims along the bottom? Neutrally buoyant?
    i. Do you plan to add more motors so your ROV can steer up and down?
    j. How do you want to control your ROV? The example code below uses "tank" steering (two separate joysticks control the left and right motors independently). Alternatively, you could control the forward/backward and left/right motion of your ROV with a single two-axis joystick. If you want to add more motors, you will need additional control inputs.
    k. Are there any potential obstacles or hazards in the environment that you need to watch out for? For example, it may be OK to build an ROV that skims along the bottom of a pool, but this might not work in a pond or lake where the tether could easily get snagged on rocks or sticks.
2. Charge your batteries so they are ready for use later.
3. Before you worry about physical construction and waterproofing your ROV, you should build and test the circuit. This will help you make sure that your motor control and any sensors are working properly. It will be much easier to change your circuit now than after everything is mounted to the ROV. Figure 4 shows a picture of the assembled example circuit. It has several main components: the Arduino, one breadboard for the H-bridge and motor connections, one breadboard for the joysticks, the two underwater thrusters, and the lithium battery (which powers the whole system). The next step includes detailed instructions for building the circuit.

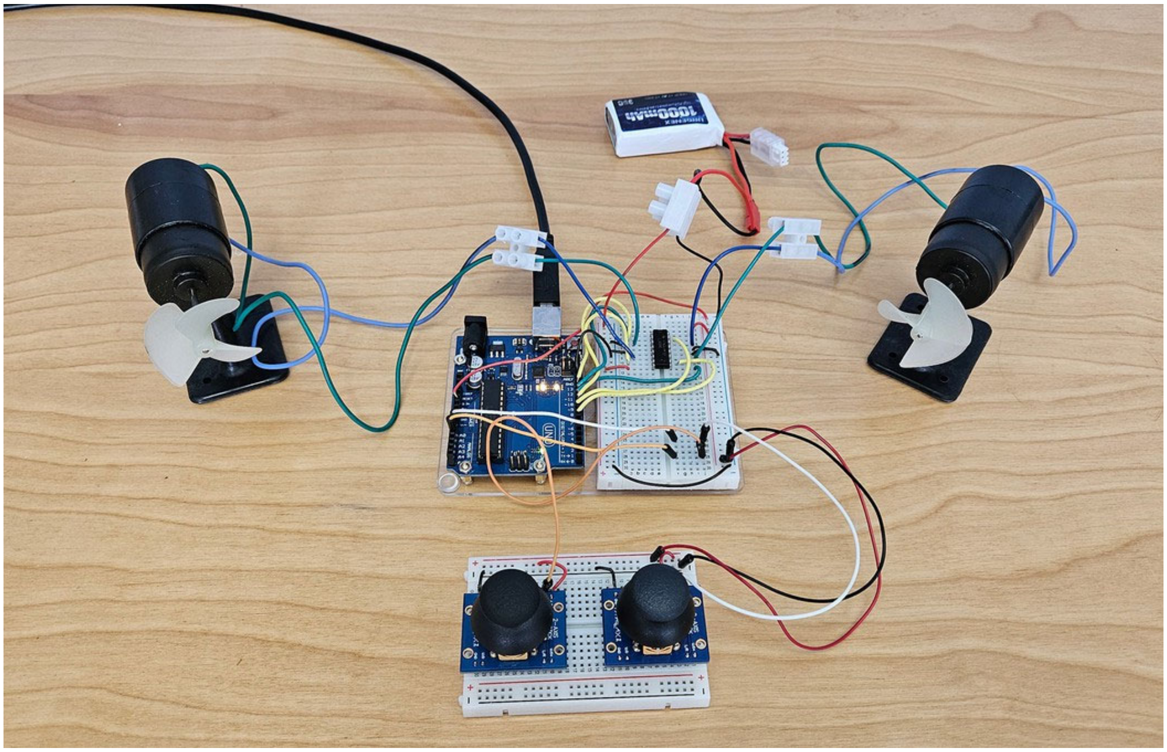

Figure 4. Circuit assembled for testing before mounting in the ROV case.

4. Build the circuit as shown in Figures 5 and 6. You can also watch a video overview of the circuit below. Figure 5 shows a breadboard diagram and Figure 6 shows the schematic. Use screw/spring terminals as needed to connect any large wires to smaller jumper wires that will fit in the breadboard. You can also access the Tinkercad simulation (https://www.tinkercad.com/things/aNekxu041Om) directly, where you can zoom in on the circuit. Note that Tinkercad does not have a joystick part, so the diagram uses potentiometers as a substitutes. You can also follow this list of connections:
   a. H-bridge (pins numbered counter-clockwise from the top left)
      i. Pin 1 to Arduino pin 11
      ii. Pin 2 to Arduino pin 12
      iii. Pin 3 to right motor negative wire
      iv. Pin 4 to ground
      v. Pin 5 to ground
      vi. Pin 6 to right motor positive wire
      vii. Pin 7 to Arduino pin 9
      viii. Pin 8 to 7.4 V from the battery
      ix. Pin 9 to Arduino pin 10
      x. Pin 10 to Arduino pin 8
      xi. Pin 11 to left motor positive wire
      xii. Pin 12 to ground
      xiii. Pin 13 to ground
      xiv. Pin 14 to left motor negative wire
      xv. Pin 15 to Arduino pin 7
      xvi. Pin 16 to 5 V from Arduino
   b. Left joystick
      i. L/R+ to 5 V from Arduino
      ii. L/R to Arduino analog pin A0
      iii. GND to ground
   c. Right joystick
      i. L/R+ to 5 V from Arduino
      ii. L/R to Arduino analog pin A1
      iii. GND to ground
   d. Battery
      i. Double-check all of your wiring before connecting the battery.
      ii. Connect the positive wire to the Arduino's Vin pin.
      iii. Connect the negative wire to the Arduino's GND pin.
      iv. Recommended: connect the switch in series with the battery's positive wire so you can easily turn your ROV on and off without unplugging wires.
      v. Make sure all breadboard ground buses are connected to Arduino GND so the entire circuit has a common ground.
      vi. Do **not** connect the left and right side power buses on the main breadboard to each other. This will create a short circuit between the 5 V supply from the Arduino and the 7.4 V supply from the lithium battery.

https://www.youtube.com/watch?v=jhRyistosQk (https://www.youtube.com/watch?v=jhRyistosQk)

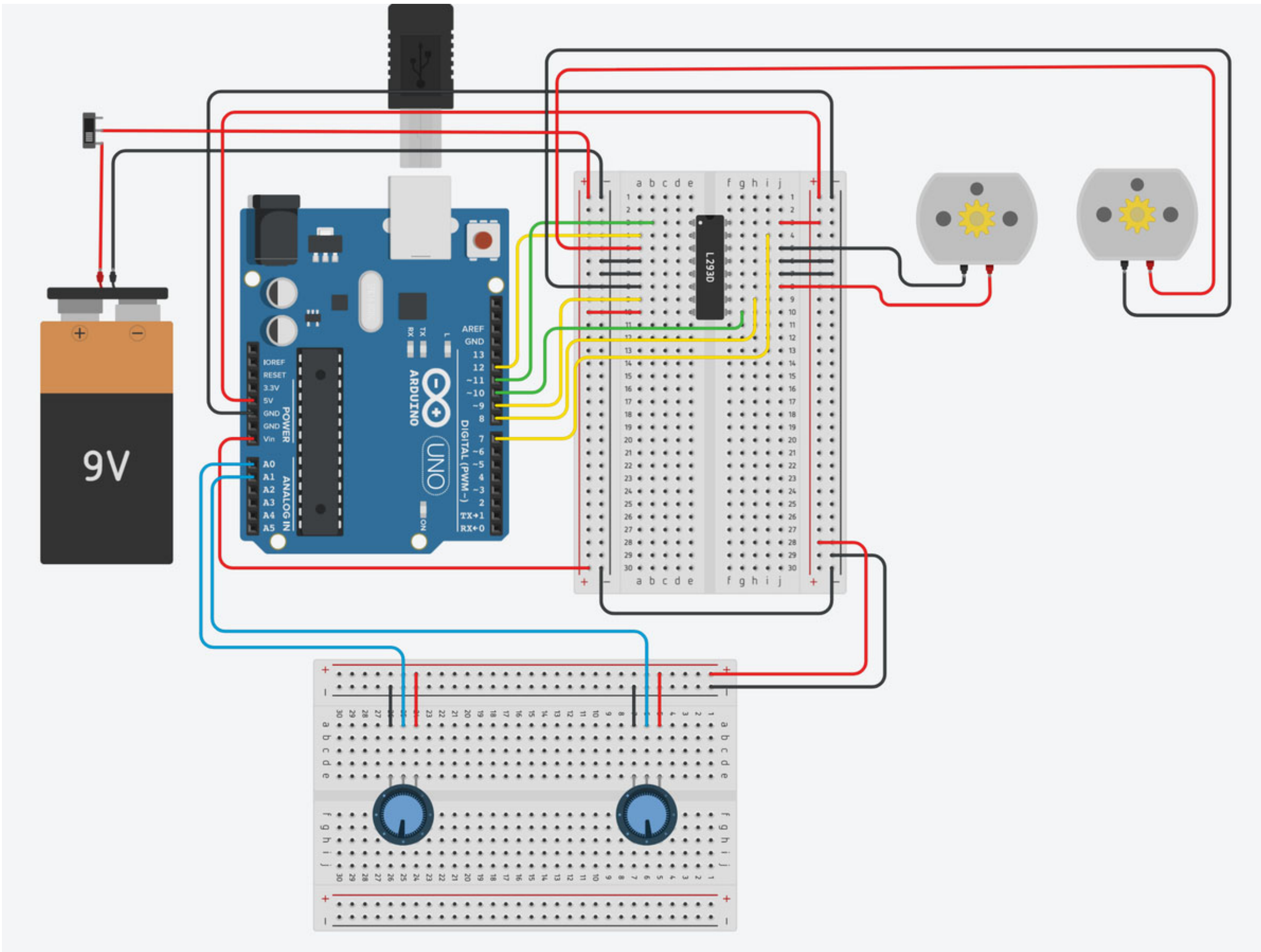

Figure 5. Breadboard diagram of the motor control circuit. Click here for a larger version of the diagram (https://www.sciencebuddies.org/cdn/Files/19622/4/arduino-ROV-breadboard-diagram.png) and click here to access the Tinkercad simulation (https://www.tinkercad.com/things/aNekxu041Om). Note that Tinkercad does not have a joystick part, so two potentiometers are used to represent the joysticks in this image. Each potentiometer has a connection to power, ground, and one of the Arduino analog pins.

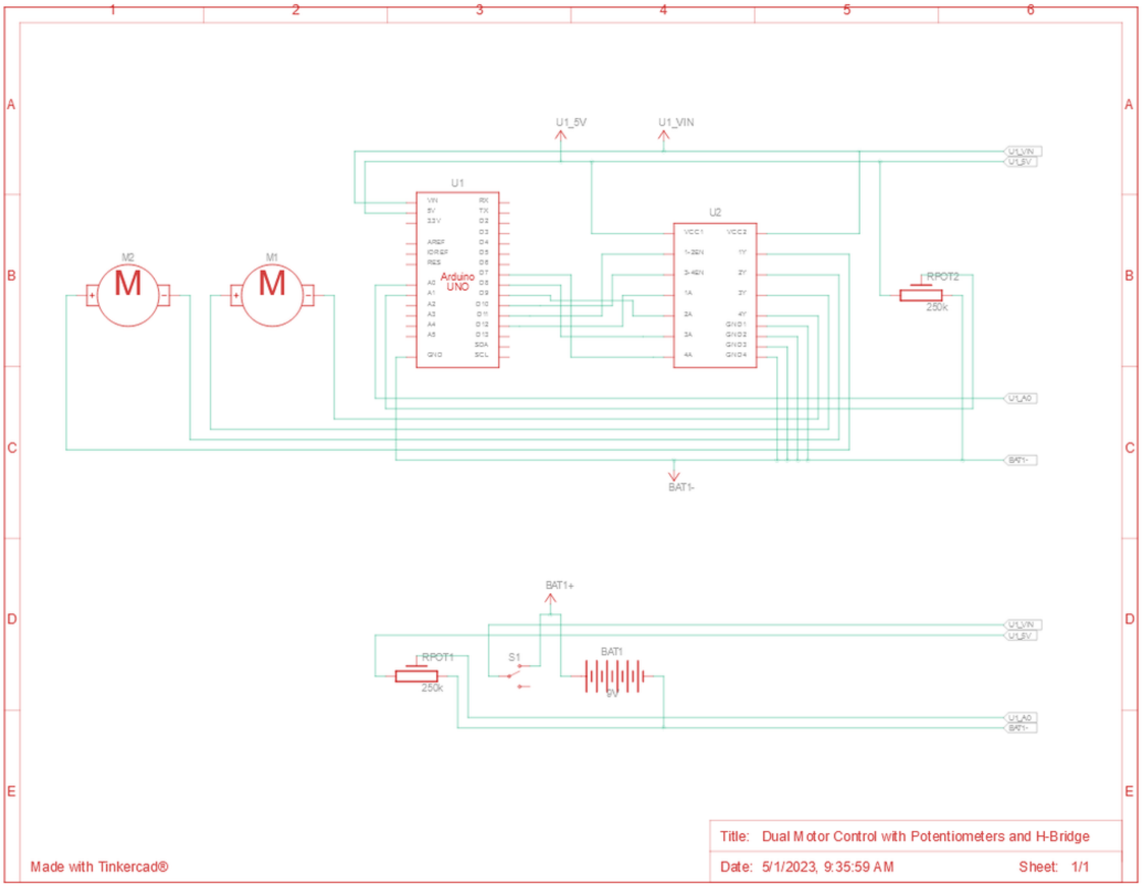

Figure 6. Schematic of the motor control circuit. Click here to download a PDF of the diagram (http://www.sciencebuddies.org/cdn/Files/19623/5/Dual-Motor-Control-with-Potentiometers-H-Bridge.pdf).

5. Download the example code (http://www.sciencebuddies.org/cdn/Files/19624/5/ROV_two_motors_two_joysticks.ino).
    a. Read through the commented code so you understand how it works.
    b. Make any required changes to the code, such as adding code to control additional motors or sensors.
    c. Upload the code to your Arduino.
    d. Test your motor controls. It can be hard to see which way the propellers are spinning, but you can put your hand near them to see which way they are blowing air.
    e. If needed, make any required corrections to your circuit or code. For example, if a motor is spinning backwards, you can reverse the two control wires connected to the Arduino or switch the two assigned pins in the code. Our Arduino troubleshooting video (http://www.sciencebuddies.org/science-fair-projects/references/how-to-use-an-arduino#step3) provides more tips for debugging your circuit if it is not working properly.
6. Once you have your circuit working, you are ready to start mounting parts to your waterproof container.
    a. You should carefully plan out where everything will go before you start drilling holes in your container.
    b. After drilling holes, remove any sharp burs from their edges, since these could cut the insulation on wires that you push through the holes.
    c. Figure 7 shows motors bolted to the sides of the container.
    d. Figure 8 shows the Arduino, breadboard, and battery placed inside the container, with wires connecting the circuit to the external controller, but holes have not been drilled for the wires yet. We recommend *not* permanently attaching the Arduino and breadboard inside the case, because you will want to remove them to conduct leak testing of your container.
    e. Figure 9 shows the sealed container with all the electronics placed inside and the wires passing through holes in the container. Note how cable clips and zip ties are used to help relieve stress on wires that pass through the container wall. This is especially important for the tether. You do not want the tether pulling directly on the silicone sealant that you will place around the hole.

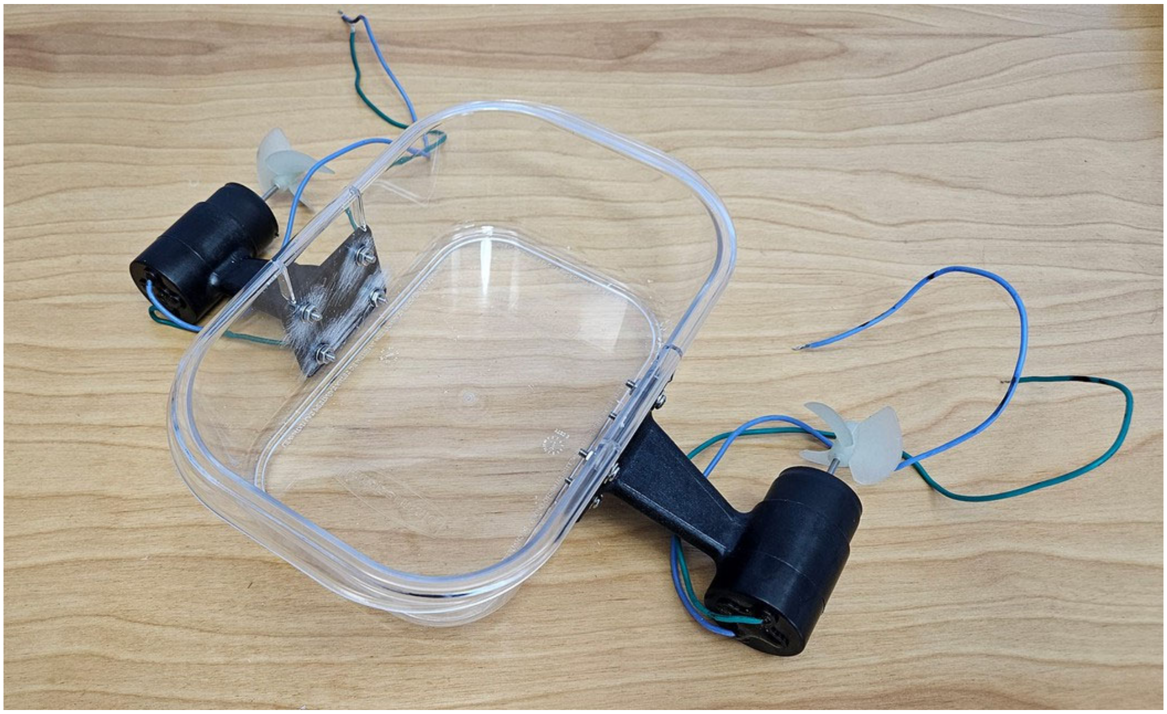

Figure 7. Motors bolted to the sides of the waterproof container.

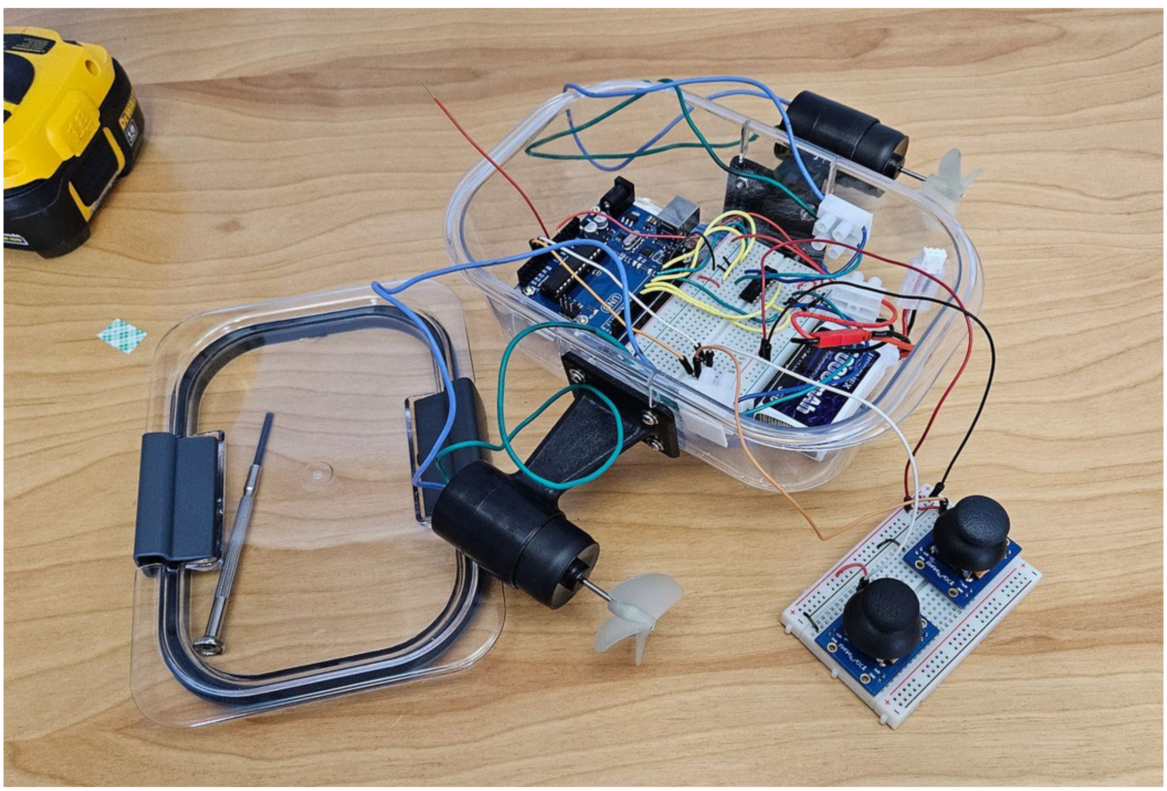

Figure 8. Electronics for the ROV placed inside a waterproof container. The motors are bolted to the sides of the container, but no holes have been drilled for the wires yet.

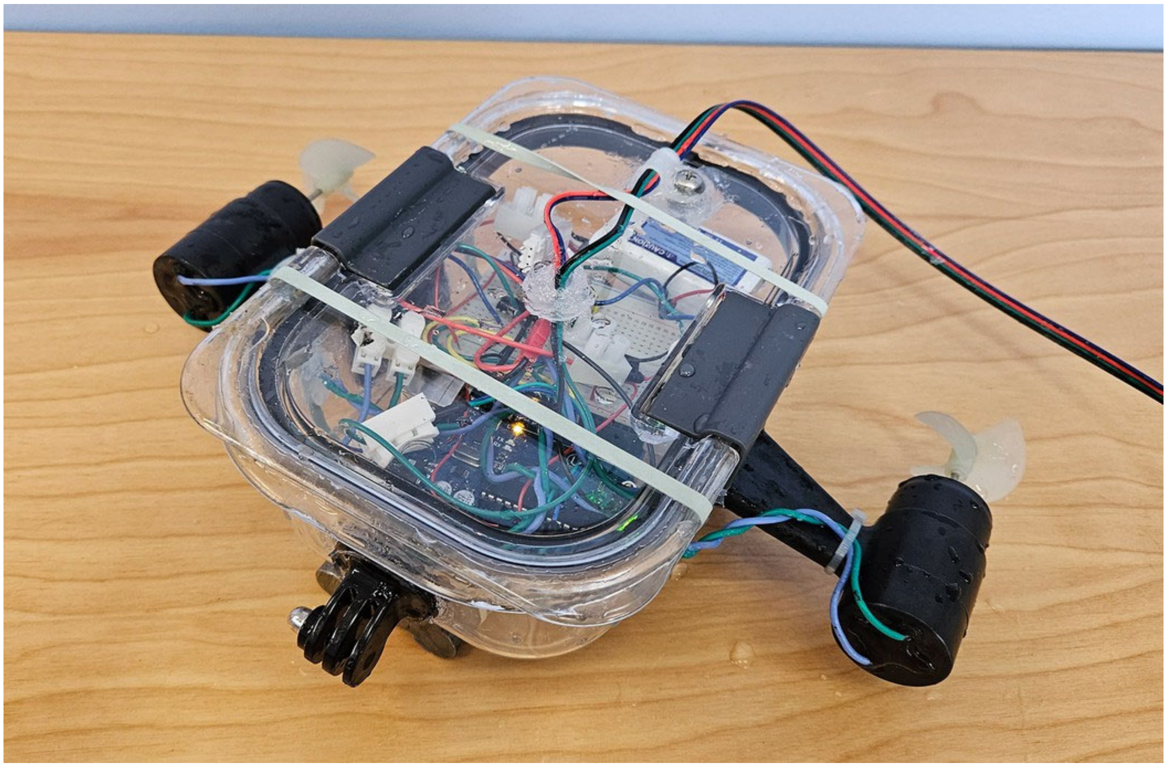

Figure 9. Assembled ROV with all the electronics inside the container.

7. After you have drilled all the required holes for mounting hardware, passing wires through, etc., follow the instructions for your silicone sealant to seal around the edges of the holes (Figure 10). Make sure you wait for the silicone to dry completely before you continue.

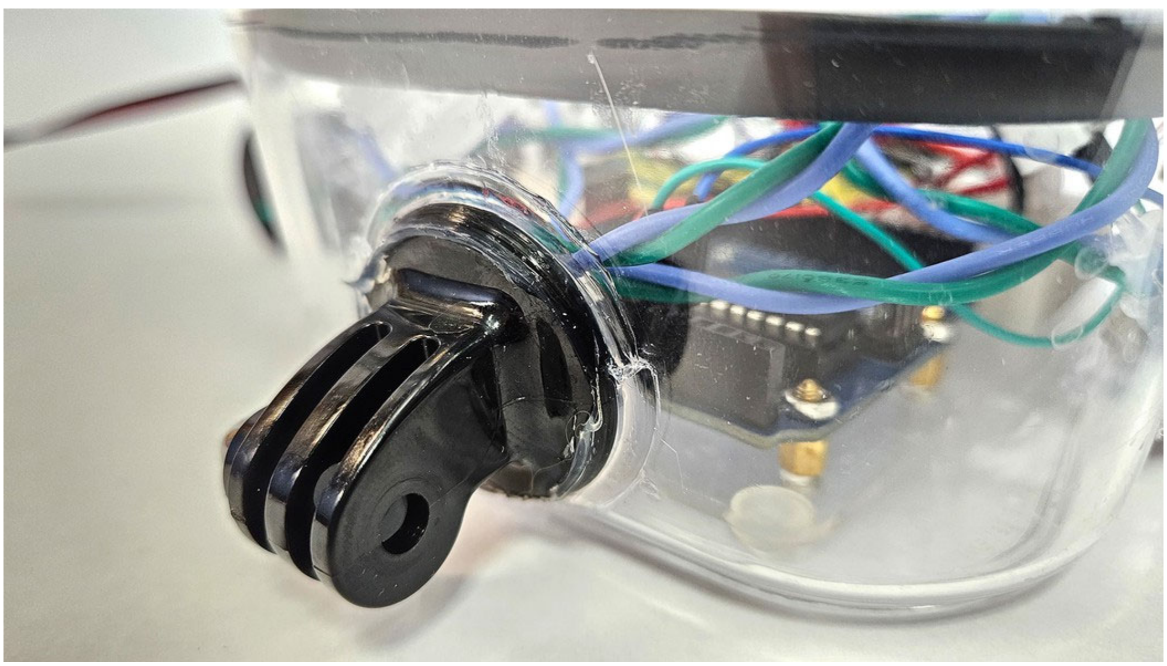

Figure 10. Silicone sealant around the edge of a camera mounting bracket.

8. After the silicone sealant has dried, it is time to conduct leak testing to make sure your container is completely watertight (Figure 11).
   a. Fill a container of water (sink, bathtub, etc.) large enough to completely submerge your ROV.
   b. **Remove the Arduino, breadboard, and battery from the container.**
   c. Put the lid on the container, making sure that it is sealed tightly.
   d. *Slowly* submerge your ROV in the water. Watch carefully for any air bubbles coming out of the container, or water leaking into the container, especially around any holes that you drilled and the edges of the lid.
   e. Hold your ROV underwater and gently move it around and rotate it as you continue to watch for leaks.
   f. Remove your ROV from the water. Use a towel to completely dry off the exterior. Check to see if any water has accumulated inside the container.
   g. If you find any leaks, you will need to patch them (by adding more silicone sealant) and repeat your testing until all leaks have been sealed. If your container leaks around the lid, you may need to use a different container.

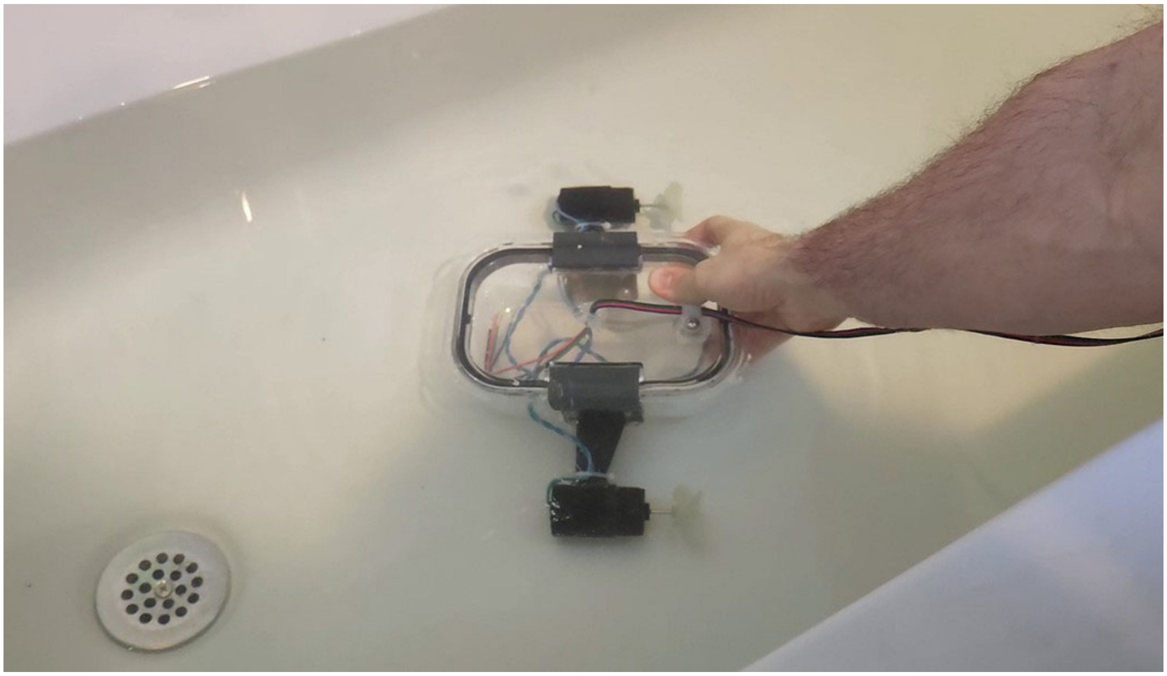

Figure 11. ROV submerged in a bathtub for leak testing.

9. When you are sure that the container does not leak, you can place the electronics inside. Remember that if you use a waterproof switch mounted to the container (Figure 12), you can turn the ROV on and off without having to remove the lid. You should turn your ROV off when not in use to conserve battery power.

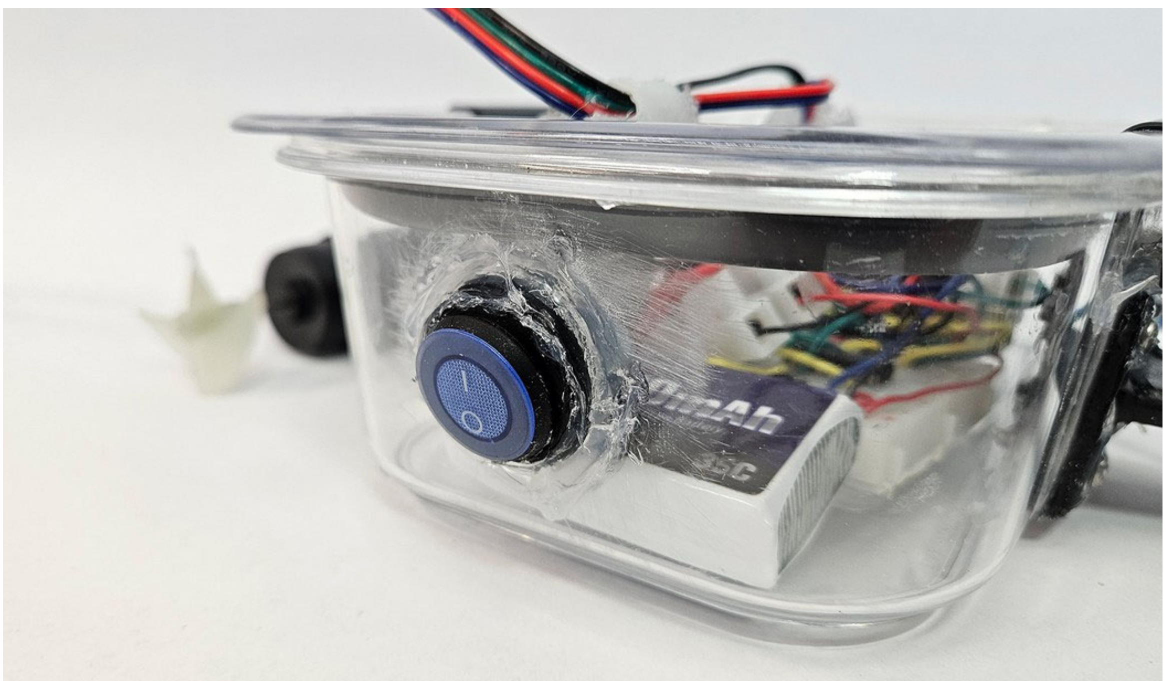

Figure 12. The waterproof power switch on the back of the ROV.

10. You may wish to build a handheld controller for your ROV so you are not just holding a breadboard. Figure 13 shows a very simple cardboard base for the controller. It uses screw terminals to connect the flexible tether wires to solid core jumper wires from the breadboard, and a cable clip to prevent the tether from pulling directly on the screw terminals. Note that your hands will probably get wet while handling the ROV, so while simple, cardboard may not be the best choice for a controller!

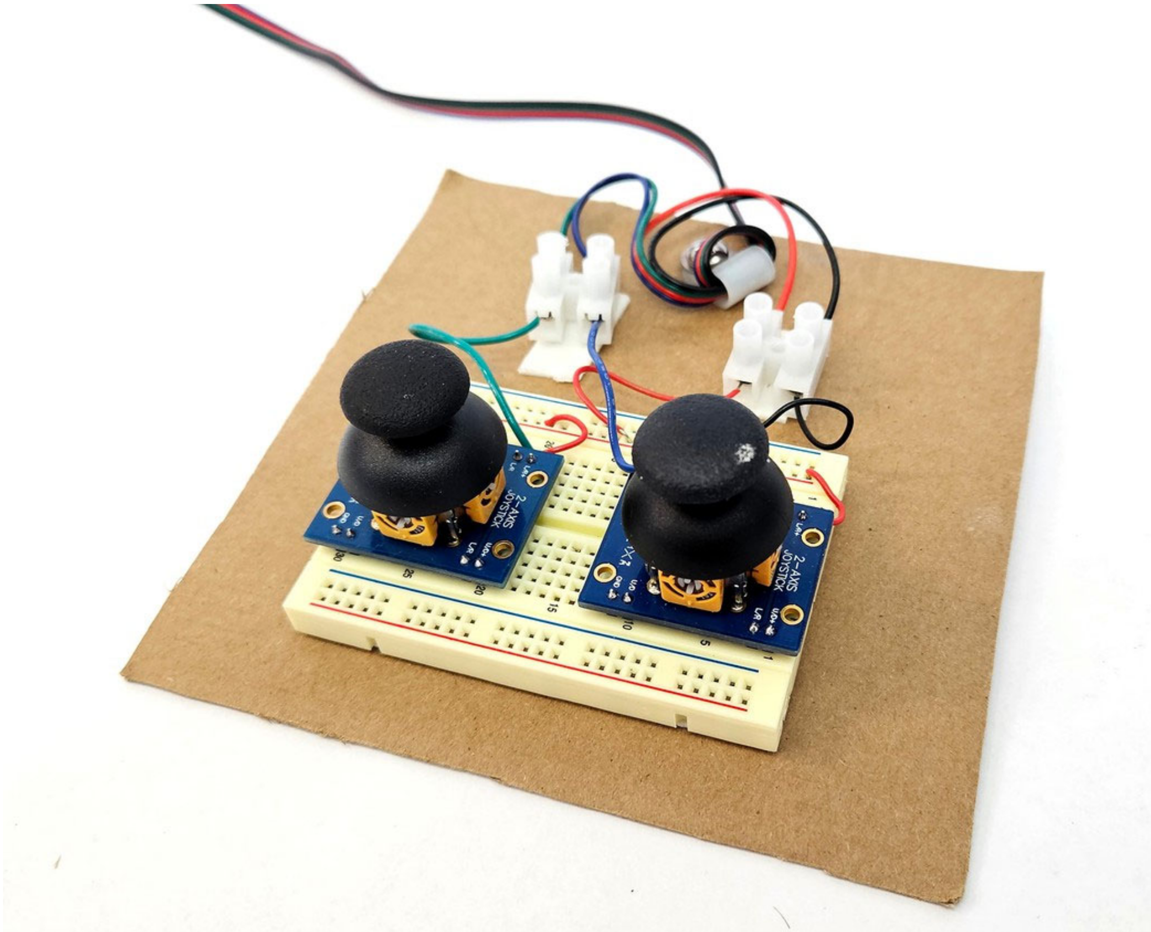

Figure 13. A simple controller made from cardboard.

11. Once you have your ROV sealed with all the electronics inside and you have finished your controller, you are ready to test your ROV! Turn your ROV on and place it in a tub of water.
    a. Start by *gently* pushing the joysticks to practice steering your ROV. Does it move too fast or too slow? If so, you can change the code to adjust how the analogRead values from the joysticks map to the analogWrite functions for motor speed.
    b. Do you need to adjust your ROV's buoyancy? If so, you can add weights or floats as needed. Figure 14 shows steel rods attached to the bottom of the ROV to make it sink to the bottom.
    c. Do you have trouble with the tether sinking or getting caught in the propellers? If so, you can consider adding one or more fishing bobbers to the tether to keep it afloat (Figure 15).
    d. Does your ROV have any attachments or accessories? If so, do they work as intended?

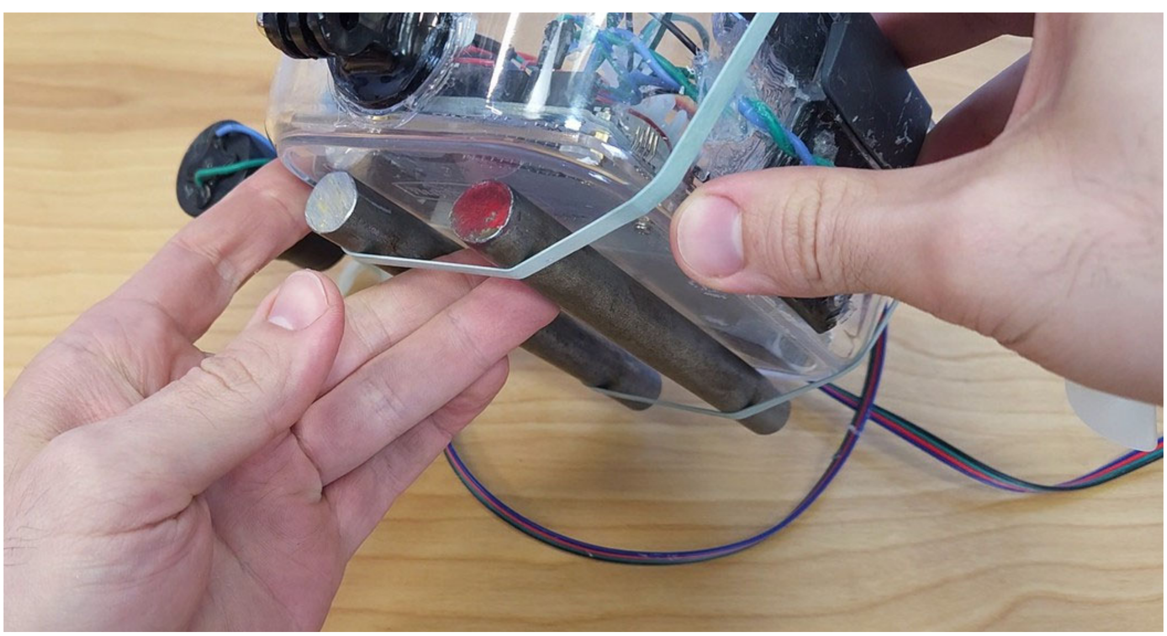

Figure 14. Steel rods attached to the bottom of the ROV with rubber bands to decrease its buoyancy.

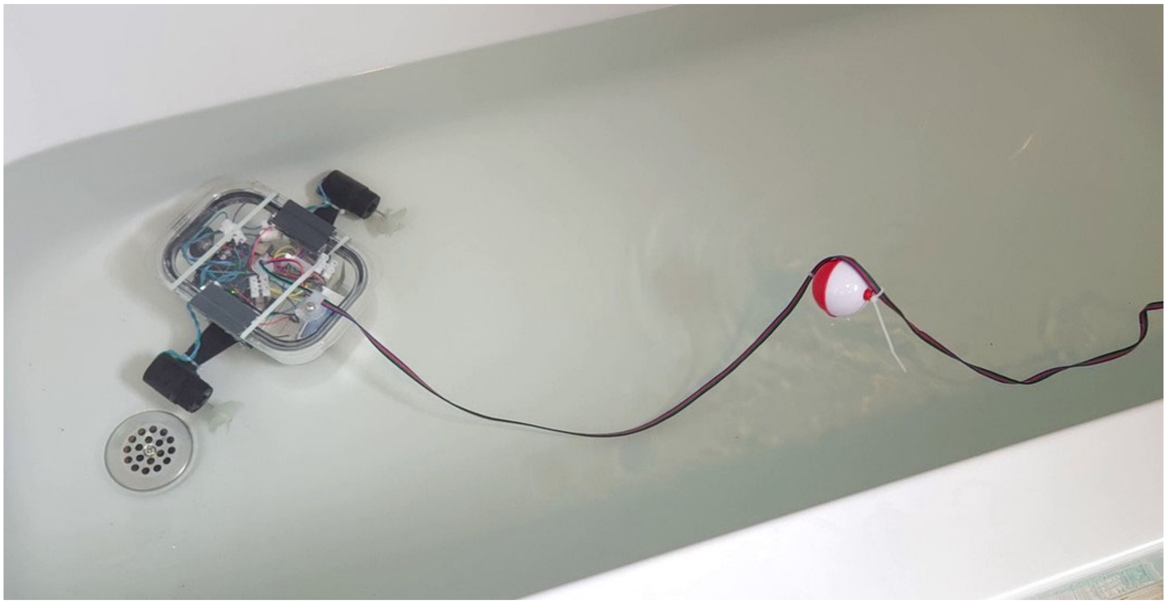

Figure 15. Fishing bobber attached to the ROV's tether to help prevent it from sinking.

12. Based on your observations, you may need to make changes to your ROV's physical design, code, or both. Remember that it is OK if everything does not work perfectly on the first try, and iteration is an important part of the engineering design process. If needed, improve your design and continue to test your ROV in a smaller container (like a bathtub or kiddie pool) before moving to a larger body of water.
13. Once you have your ROV working the way you want it, you are ready for some field testing! Try testing your ROV in a larger body of water (Figure 16). Now you might have to deal with some real-world obstacles, like wind and waves, or debris, like sticks and stones. How does your ROV perform under these conditions? Are there more changes you can make to improve your design?

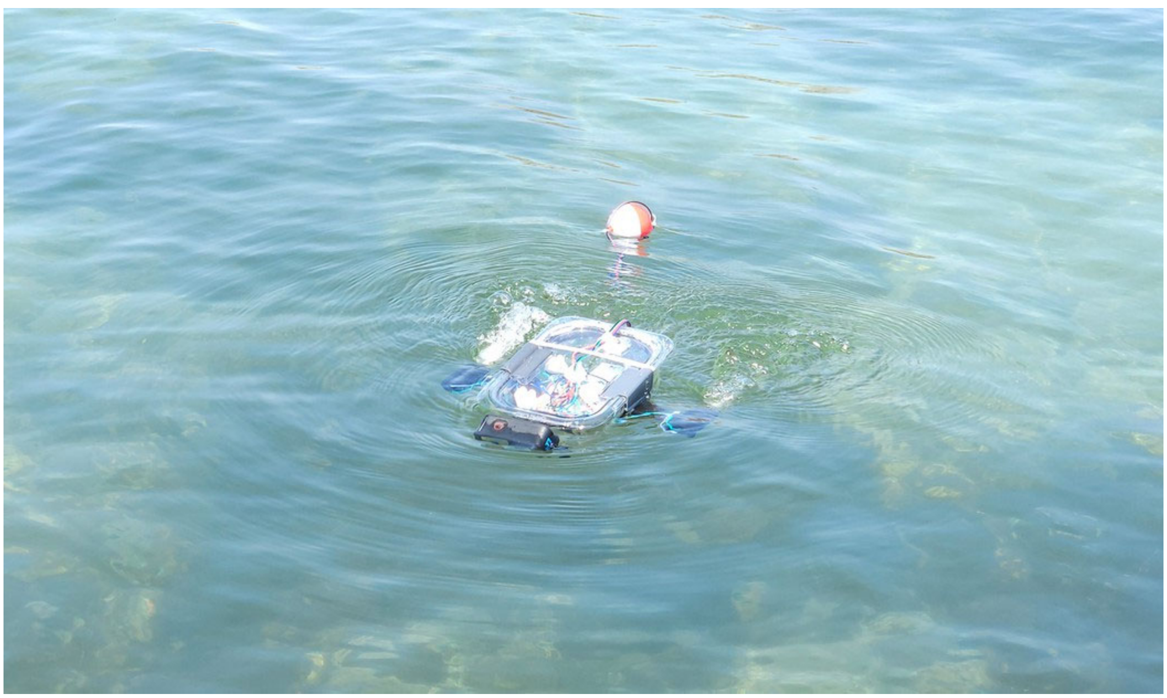

Figure 16. ROV operating in a lake.

14. Once you have your ROV working in "real world" conditions, can you use it to collect data or information about the environment? Figure 17 shows a picture of a school of fish we were able to capture with our ROV. What can you find when you explore? There are many other things you can try with your ROV! See the Variations (#makeityourown) section for some ideas.


Figure 17. Picture of a school of fish captured with the Science Buddies ROV.