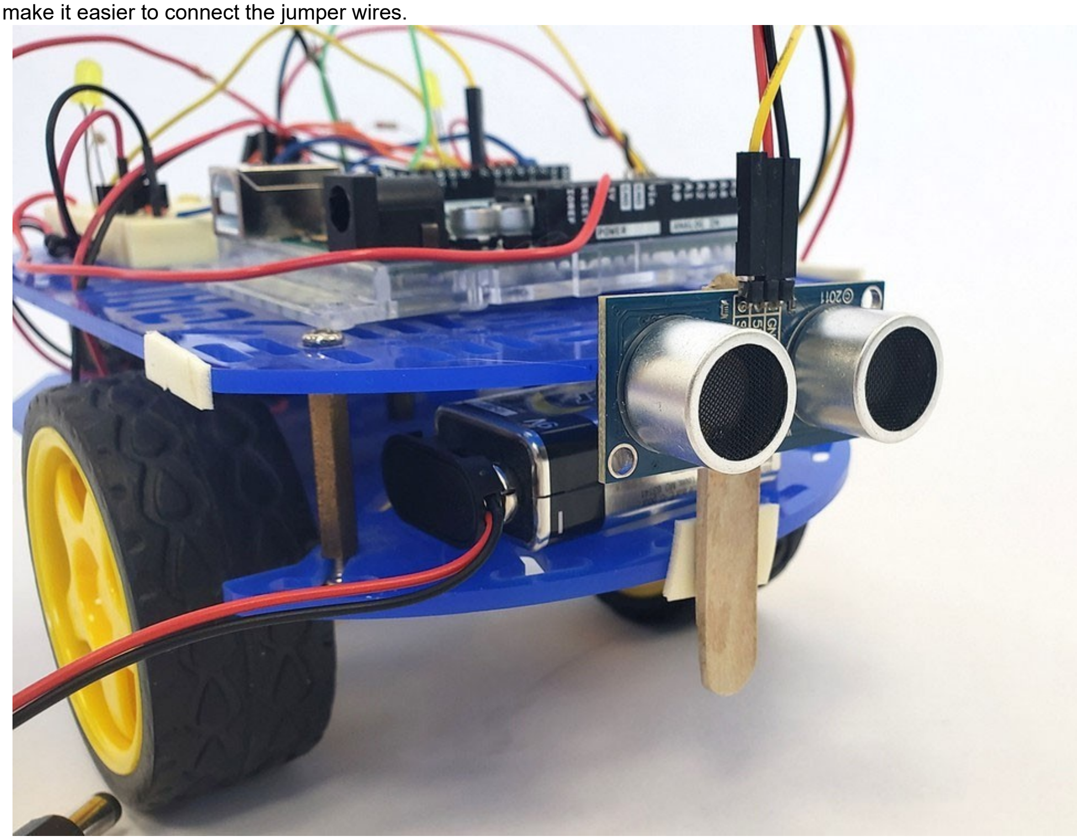
Experimental Procedure

1. Assemble your Bluebot chassis.

   Follow the instructions in this video; however, instead of mounting the 4xAA battery pack on top of the robot, mount it on the lower plate. Then mount your Arduino next to the breadboard on the top plate. This will give you easier access to the circuit when connecting it to the Arduino.

2. Attach the ultrasonic sensor.

   Mount the sensor to the front of your robot using a craft stick and double-sided tape, as shown in Figure 2. Make sure the pins are facing up; this will make it easier to connect the jumper wires.



3. Assemble your circuit.

   If you plan to use the example code provided by Science Buddies, assemble the circuit as shown in Figures 3 and 4. The Arduino pins used in this diagram match the ones used in our example code. If you try to use our example code without matching this wiring diagram exactly, the code will not work properly.

   Take your time and carefully double-check all of your wiring. This is a complicated circuit with a lot of connections. You should connect the batteries *last* to avoid the risk of creating short circuits while you work.

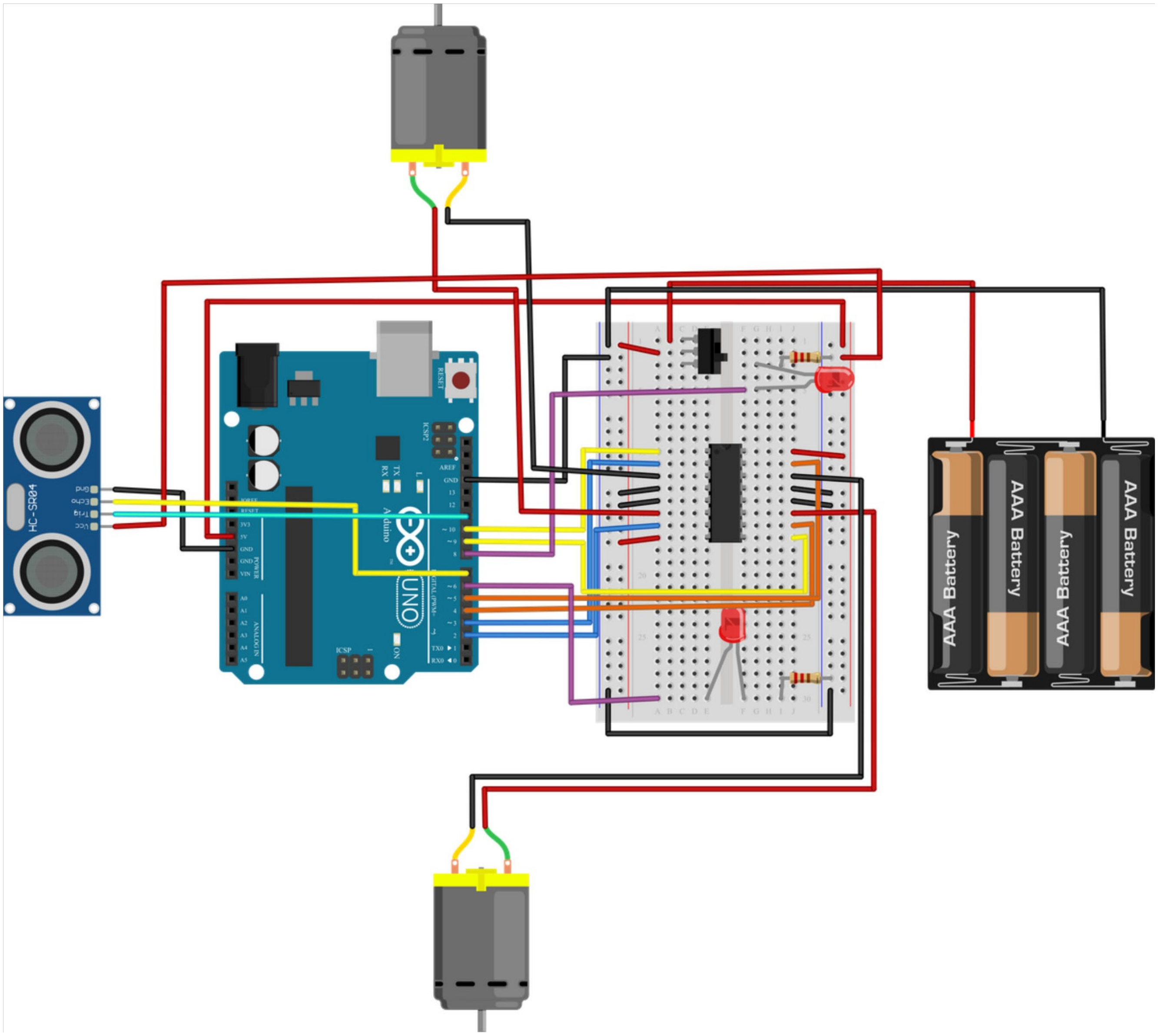
   Note: It is important to make sure all components in the circuit have a common ground, but you must be very careful ***not*** *to short different positive voltages together (e.g., 5 V from the Arduino and 6 V from the 4xAA battery pack).*

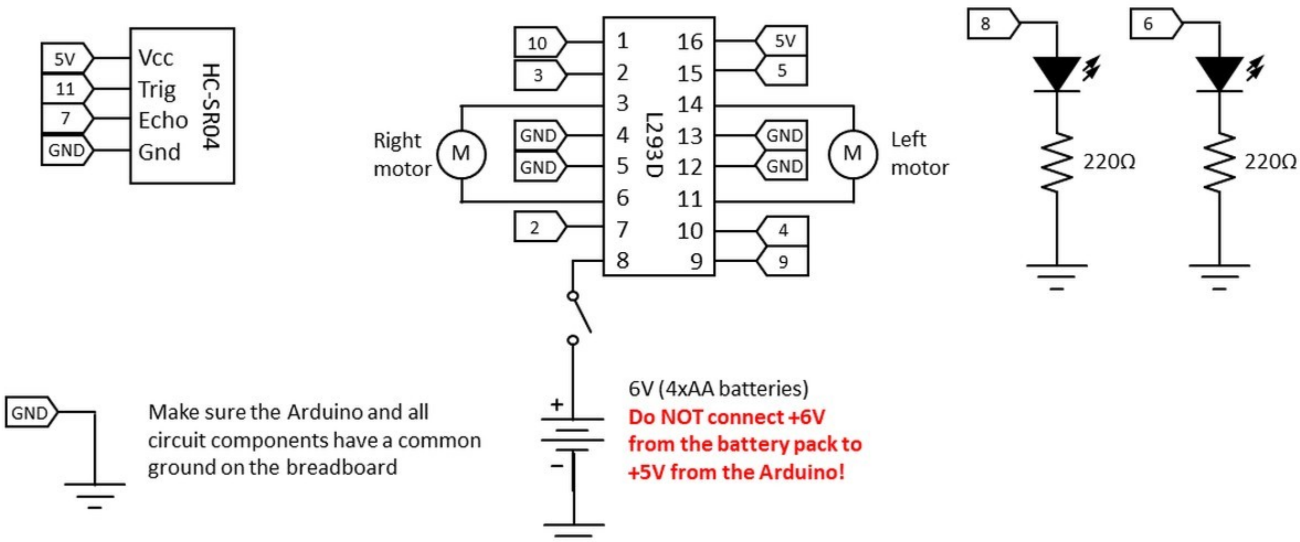
   Following is one suggested order in which you can build the circuit, but you do not need to do it in this order.
   Note: Connections are made with male-male jumper wires unless otherwise indicated.

   a. Connect the left and right ground buses on the breadboard. Do *not* connect the two power buses. You will use them for different voltages.
   b. Connect the Arduino's GND pin to the breadboard's ground bus.
   c. Connect the Arduino's 5 V pin to the breadboard's right-side power bus.
   d. Pins on the L293D H-bridge are numbered 1–16, counterclockwise from the top left. Connect them as follows:i. Pin 1 to Arduino pin 10
      ii. Pin 2 to Arduino pin 3
      iii. Pin 3 to the right motor's negative wireiv. Pin 4 to ground v. Pin 5 to ground
      vi. Pin 6 to the right motor's positive wire
      vii. Pin 7 to Arduino pin 2
      viii. Pin 8 to the left-side power bus (which will be connected to 6 V later)
      ix. Pin 9 to Arduino pin 9
      x. Pin 10 to Arduino pin 4

xi. Pin 11 to the left motor's positive wire
xii. Pin 12 to groundxiii. Pin 13 to ground xiv. Pin 14 to the left motor's negative wire
xv. Pin 15 to Arduino pin 5
xvi. Pin 16 to 5 V

e. Connect the ultrasonic sensor using male-female jumper wires.
    i. GND pin to ground
    ii. 5V V pin to 5 V
    iii. Trig pin to Arduino pin 11iv. Echo pin to Arduino pin 7
f. Connect an LED with a 220 Ω current-limiting resistor between Arduino pin 8 and ground.
g. Connect an LED with a 220 Ω current-limiting resistor between Arduino pin 6 and ground.
h. Finally, connect the 4xAA battery pack to your circuit.
i. Connect the negative wire to the breadboard's ground bus.
    ii. Instead of connecting the positive wire directly to the left-side power, connect it to a switch on the breadboard first, thenconnect the switch to the left-side power bus (assuming you previously connected the right-side power bus to 5 V; remember not to short the two voltages together). This will allow you to easily turn the battery pack on and off.



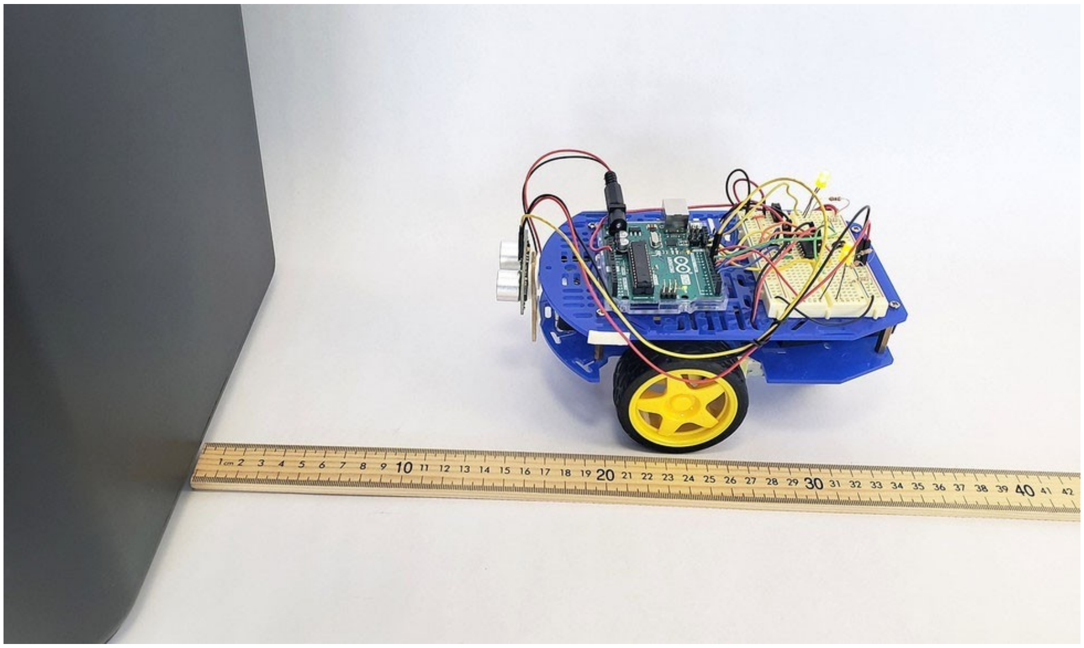


4. Download and review the automatic braking code (http://www.sciencebuddies.org/cdn/Files/18296/12/automatic_braking-code.ino) for your autonomous vehicle.

   Carefully read through the commented code. This code makes the robot drive forward at full speed until it detects an obstacle closer than a certain distance; then it will automatically stop.

5. Set up your experiment.

Place the meterstick against an obstacle as shown in Figure 5. Make sure the obstacle is large and has a flat surface so it can be easily detected by the sensor. Aim the robot toward the obstacle.



Calibrate your ultrasonic sensor.
  a. Make sure the power switch on the breadboard is turned off—that way your robot's wheels will not spin.
  b. Upload the code to your Arduino, but keep the Arduino connected to the USB cable.
  c. Open the serial monitor (Tools→Serial monitor). It will print out the reading from the ultrasonic sensor in centimeters.
  d. Place your robot next to the meterstick and aim it toward the obstacle. Measure the distance from the front of the sensor to theobstacle. Compare this to the reading displayed on the serial monitor. Record both values and note any difference between them.
  e. Manually move the robot back and forth. The LEDs should light up when the robot is less than 30 cm from the obstacle. (This is thedefault value in the example program, but you can change it).
6. Calibrate your robot's steering.
  a. Pick up the robot and turn the power switch on. The wheels should start spinning.
  b. Put the robot down far away from the obstacle (at least 60 cm) and watch it drive.
  c. If the robot does not drive straight, change either the *speedL* or *speedR* variable in the code as needed. For example, if the robot turns slightly to the right, decrease the *speedL* variable to make the left wheel spin slower.
  d. Re-upload your code and try again. Repeat until your robot drives straight.
7. Prepare to collect data.

You are almost ready to start collecting data! Make a data table like Table 1.

  a. Target Final Distance is the desired distance from the obstacle at which you want to stop. By default, this is 30 cm in the exampleprogram.
  b. Actual Final Distance is the final physical distance between the obstacle and the front of the ultrasonic sensor. Measure with themeterstick once the robot has come to a complete stop.
  c. Stopping Distance is the distance the robot traveled from the point where the sensor detected an obstacle at the target distance tothe point where the robot came to a complete stop. To find this value, subtract the actual distance from the target distance.

8. Test your robot's automatic braking.

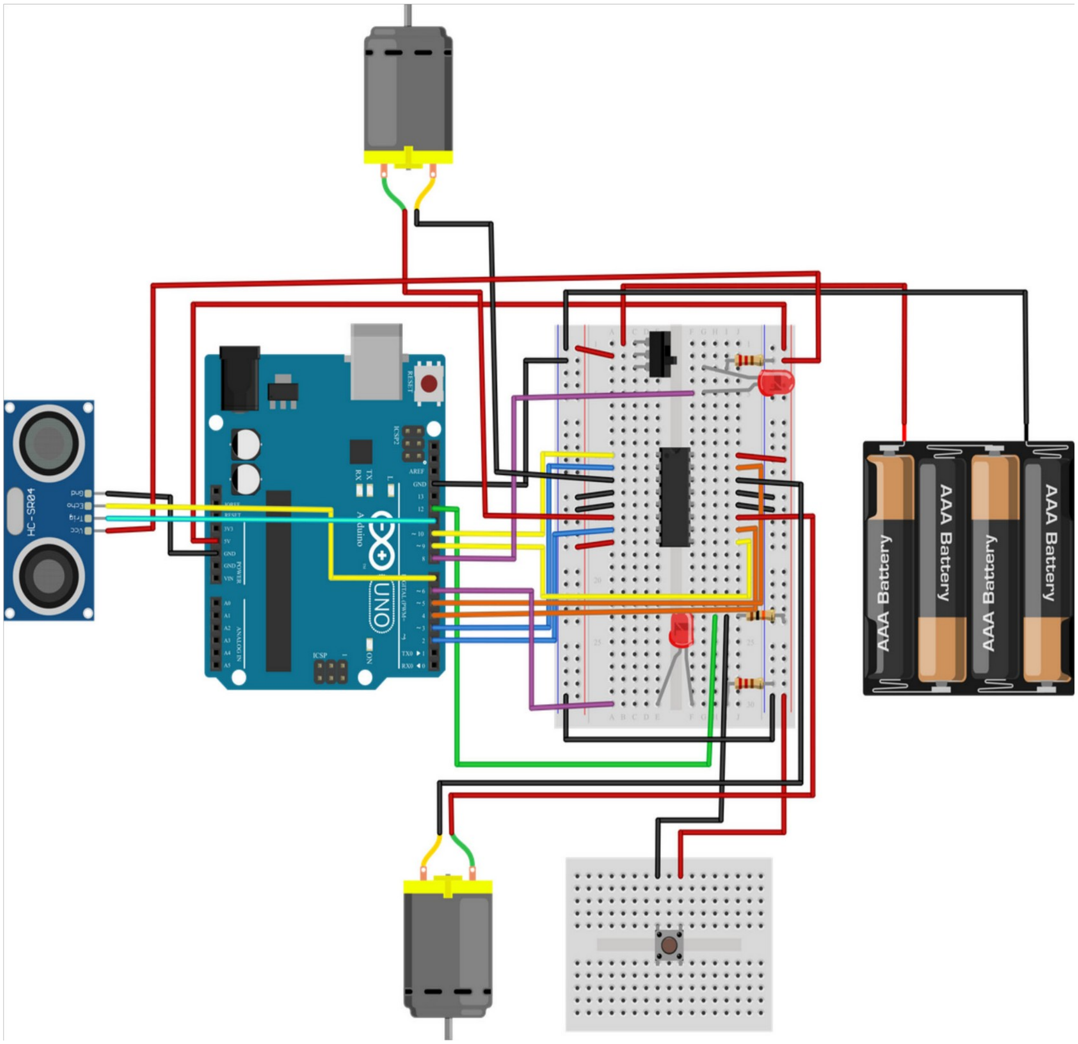
Pick your robot up and place it far enough back that it will have time to accelerate to full speed. Aim it straight toward the obstacle from about 50–60 cm away. Let the robot go. After it stops, measure the actual final distance to the obstacle and record this value in your data table.
Repeat this step at least four more times, for a total of five trials with automatic braking.

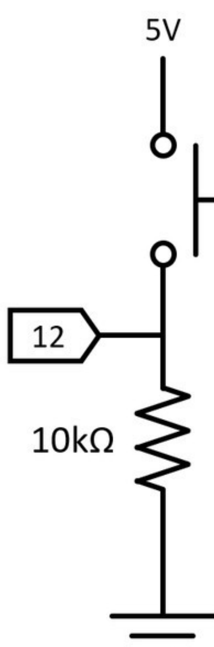
9. Build a manual braking switch.

This wired "remote" with a button will let you manually activate the brakes on your robot. When the robot detects an obstacle at the target distance, it will no longer brake manually; instead, its lights will come on and it will wait for you to push the button to make it stop.

  a. Cut two pieces of stranded wire that are each at least 1 m long. Strip the ends of the wires.
  b. Since the stranded wire is too flexible to push into a breadboard, use needle-nosed pliers to twist and crimp the ends of thestranded wire onto pieces of solid-core jumper wire (Figure 6). For a better connection, use a soldering iron if you have one available. You can also wrap the connections in electrical tape or heat-shrink tubing to prevent short circuits. If you do not, make sure the exposed wires do not bump into each other.

c. Place a button on a mini breadboard to use as a "remote" and connect it to your main breadboard as shown in Figures 7 and 8.i. Connect one of the button's wires directly to 5 V. ii. Connect the other wire to Arduino pin 12, with a 10 kΩ pull-down resistor to ground.





10. Download and review the manual braking code for your autonomous vehicle.

    Read through the commented code. Make sure you understand the differences between this program and the one for automatic braking.

11. Test your robot's manual braking.

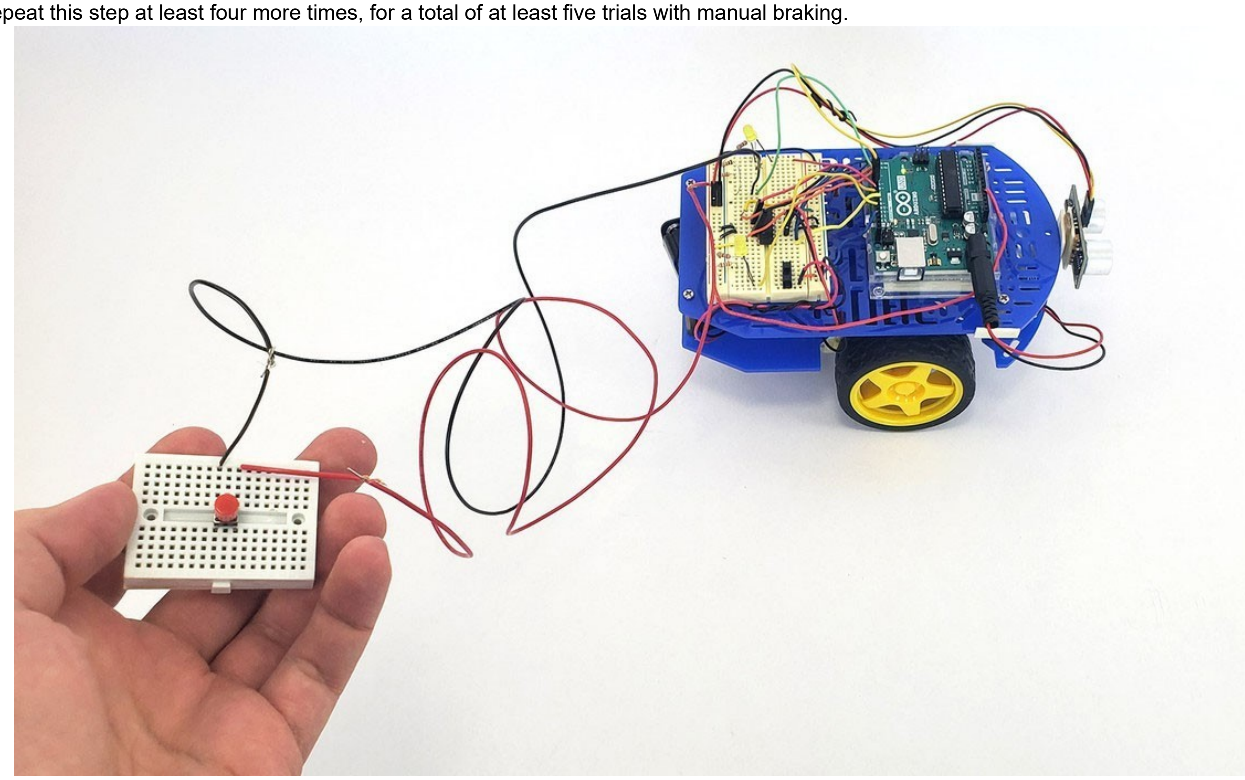
    Aim your robot toward the obstacle and hold on to the remote with one hand (Figure 9). Make sure the wires connecting the remote to the robot are loose and not tangled. Let go of the robot and carefully watch it drive forward.

    When you see the LEDs light up, press and hold the button.

    Note: Make sure you wait for the lights; do not try to anticipate the sensor. If it helps, you can remove the meterstick until the robot has stopped.

    Record the final distance to the obstacle in your data table.

    Repeat this step at least four more times, for a total of at least five trials with manual braking.



12. Analyze your results.
    a. Calculate average stopping distances for both automatic braking and manual braking. How do the two values compare?
    b. Think about your results in the context of a full-sized vehicle. How can human reaction time influence the stopping distance of a cartraveling at highway speeds?
    c. How could distractions that increase reaction time (e.g., cell phone use) affect stopping distance?