.

Build a Motion-Activated Guard Robot (BlueBot Project #1)

https://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p024/robotics/guard-robot (http://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p024/robotics/guard-robot)

PDF date: 2019-12-03

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/engineering-design-process/engineering-design-process-steps.shtml).

Assembling Your BlueBot Chassis

1. Follow the instructions in the video to assemble your robot chassis.
    a. Note that your kit does come with printed directions for assembling the chassis, but we recommend watching the video so you fully understand how all the parts fit together.
    b. Note that we recommend using double-sided foam tape to attach the battery holder to the top of the chassis, as shown in Figure 4. The printed directions recommend putting the battery holder in-between the two chassis plates, but this makes it harder to change the batteries.

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

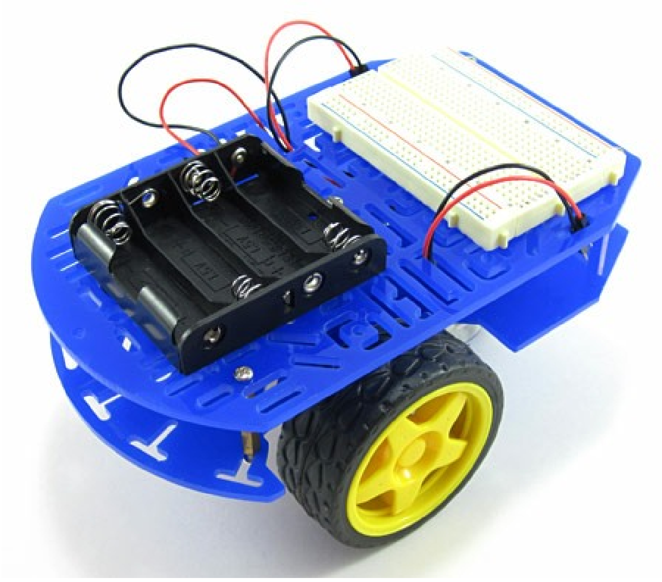


Figure 4. A completed BlueBot chassis with breadboard and battery pack on top.

Assembling Your Circuit

If you purchased your kit before December 2, 2019, use these instructions (https://www.sciencebuddies.org/Files/14072/8/Robotics_p024-Procedure-2019-06-03.pdf) to complete your assembly.

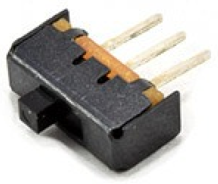
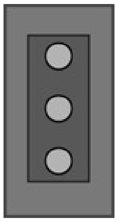
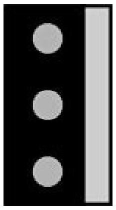
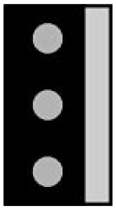
1. To build your circuit, you will need to know how to use a breadboard. Watch the video and see the Science Buddies reference How to Use a Breadboard (http://www.sciencebuddies.org/science-fair-projects/references/how-to-use-a-breadboard).

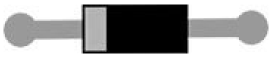
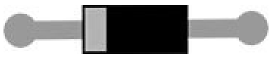
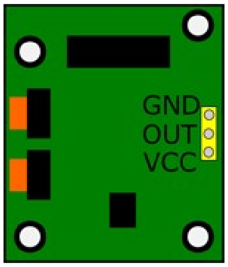
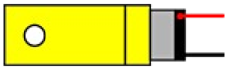
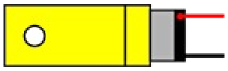
https://www.youtube.com/watch?v=6WReFkfrUIk (https://www.youtube.com/watch?v=6WReFkfrUIk)

2. Now that you know how to use a breadboard, you are ready to assemble your BlueBot circuit. Table 2 shows a list of all the components in the circuit and where they go on the breadboard. You can download and print a PDF (https://www.sciencebuddies.org/science-fair-projects/checklist-Robotics_p024-2019-4.pdf) of this table—complete with checkboxes to track each step—to use while you are building your robot. You can also view a slideshow (#breadboard-slideshow) that shows breadboard diagrams of the circuit. Follow along in the table and/or slideshow to build your circuit one component at a time. Your finished circuit should look like the one in Figure 5 (#figure5).

Pay attention to these notes:

a. Remember to push all components *firmly* into the breadboard.
b. All references to orientation (up, down, left, and right) assume you have the breadboard "right-side up," so the writing is facing you.
c. Your jumper wire kit comes with an assortment of colors, and the colors may vary. It does not matter what color jumper wires you use. Your colors do *not* need to match the colors in the diagrams. In general, you should use the shortest wires possible, to help keep your circuit neat.
d. You will use male-female jumper wires to connect the PIR sensor to the breadboard. These wires act like "extension cords" that allow you to attach the sensor to the front of your robot. You *do* need to keep track of the wire colors when connecting the PIR sensor, since you need to connect the three pins in the right order.
e. Insert the batteries *last*. If you see or smell smoke when you insert the batteries, you have a short circuit somewhere. Immediately remove the batteries and re-check your wiring.

Component	Picture	Symbol	Breadboard holes	Note
Power switch			F1, F2, F3	Direction it is facing does not matter, but make sure to slide switch down (toward row 30, away from row 1), this is the "off" position.
Jumper wire			J2 to (+) bus	Color does not matter.
Jumper wire			Left side (+) bus to right side (+) bus	Color does not matter.
Jumper wire			Left side (-) bus to right side (-) bus	Color does not matter.
MOSFET			C11, C12, C13	Writing should face to the left, large silver tab should face to the right. Note: the writing on your MOSFET may not match the picture exactly. This is OK.
MOSFET			C18, C19, C20	Writing should face to the left, large silver tab should face to the right. Note: the writing on your MOSFET may not match the picture exactly. This is OK.

Component	Picture	Symbol	Breadboard holes	Note
Jumper wire			A13 to (-) bus	Color does not matter.
Jumper wire			A20 to (-) bus	Color does not matter.
Diode			A12 to (+) bus	Gray band must face to the left. Optional: Shorten the leads (see Figure 5).
Diode			A19 to (+) bus	Gray band must face to the left. Optional: Shorten the leads (see Figure 5).
PIR sensor			GND pin to (-) bus OUT pin to J15 VCC pin to (+) bus	Color-code wires to make connections easier. Connect black wire to GND, blue wire to OUT, and red wire to VCC. See slides 13 and 14 in slideshow.
Jumper wire			E11 to G15	Color does not matter.
Jumper wire			E18 to F15	Color does not matter.
Top motor			Red lead to (+) bus Black lead to E12	When the robot is driving forward, this is the "right" motor
Bottom motor			Red lead to (+) bus Black lead to E19	When the robot is driving forward, this is the "left" motor.

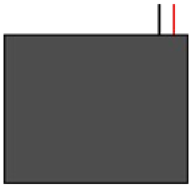
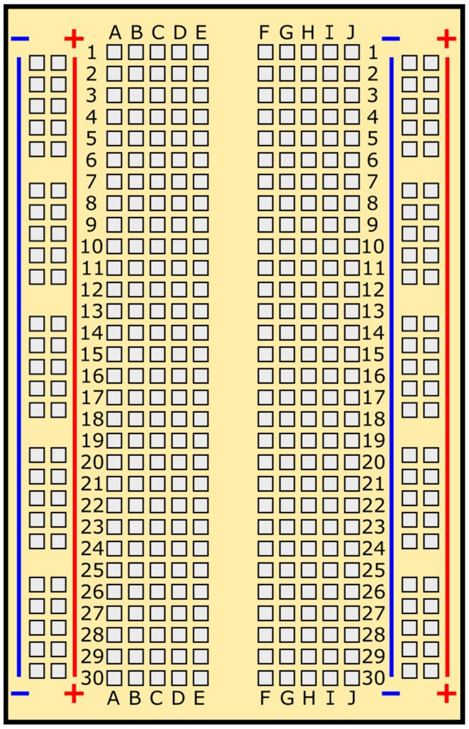
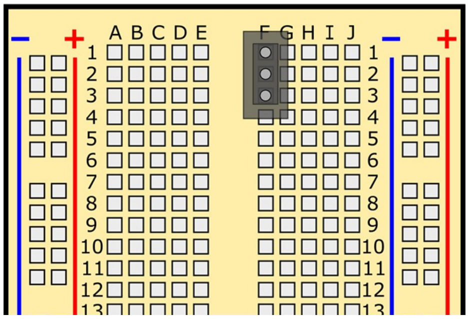
Component	Picture	Symbol	Breadboard holes	Note
Battery holder			Red lead to J1 Black lead to (-) bus	Do not insert batteries until circuit is complete.
AA battery			N/A	Insert into battery holder. Make sure (+) signs on batteries line up with (+) signs in battery holder.

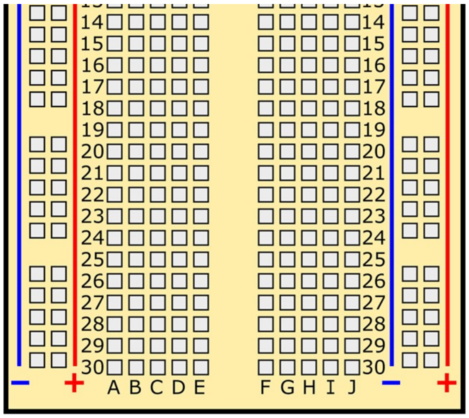
Table 2. List of circuit components and locations. A printable PDF version (https://www.sciencebuddies.org/science-fair-projects/checklist-Robotics_p024-2019-4.pdf) is available.

Slideshow Images

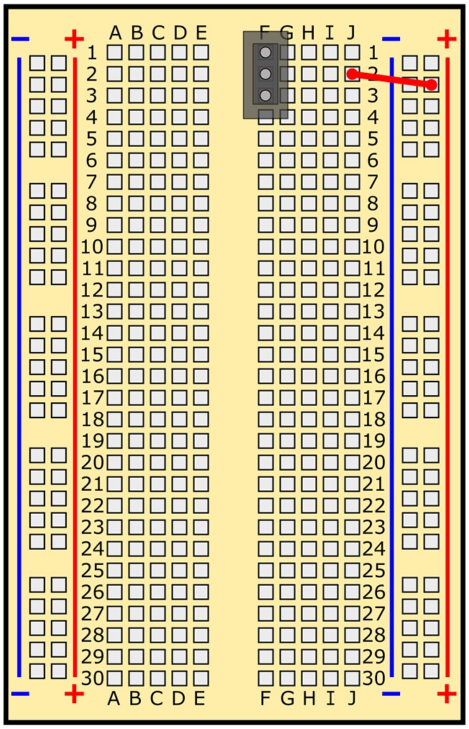


1. Blank breadboard.

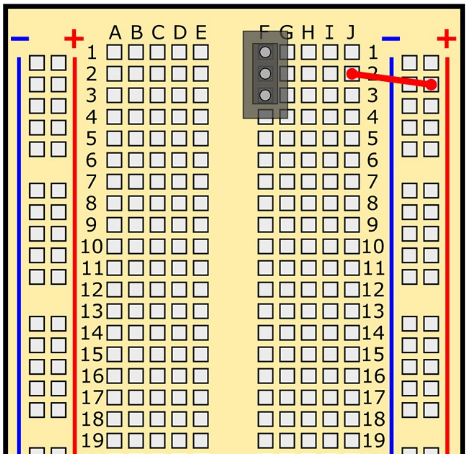


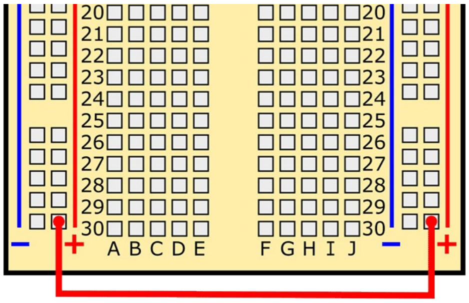


2. Insert power switch into holes F1, F2, F3. Direction of switch does not matter.

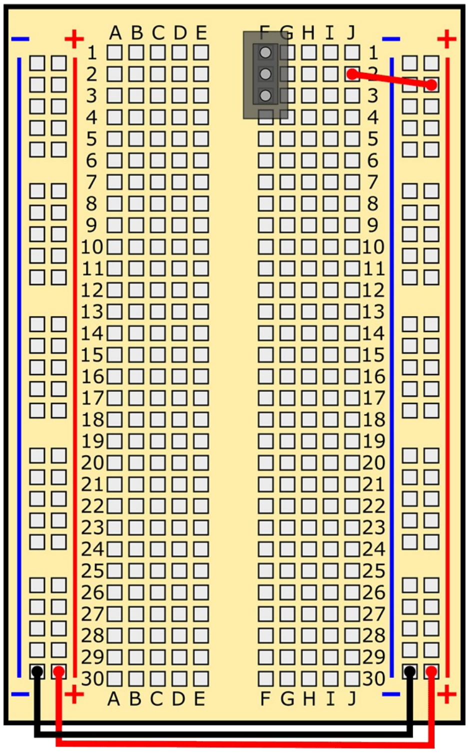


3. Connect jumper wire from hole J2 to (+) bus.

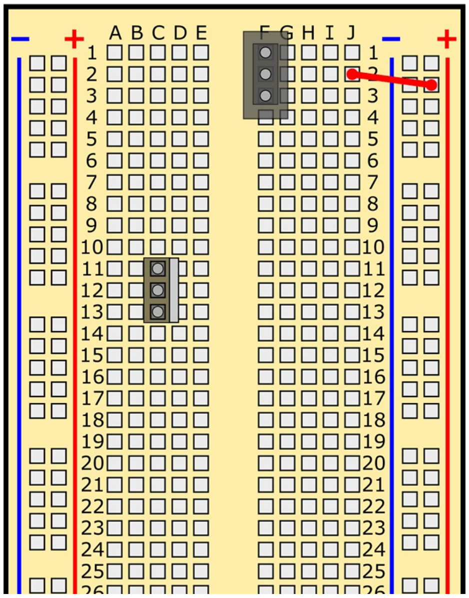


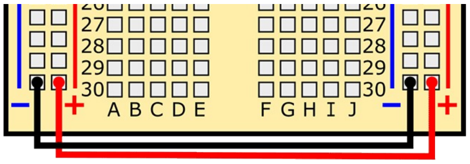


4. Connect jumper wire from left (+) bus to right (+) bus.

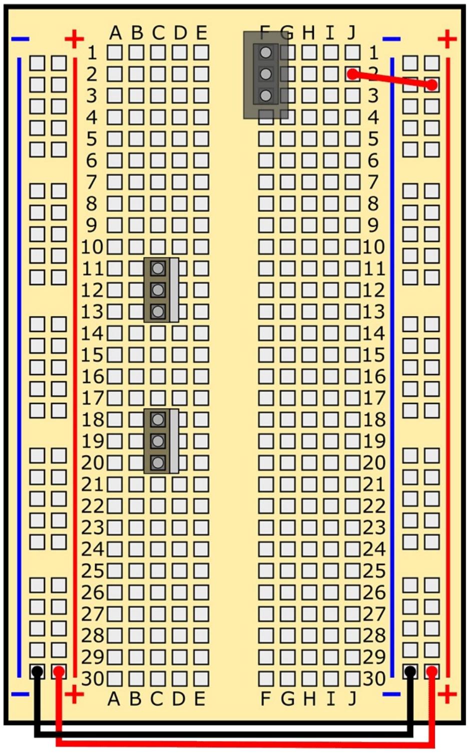


5. Connect jumper wire from left (-) bus to right (-) bus.

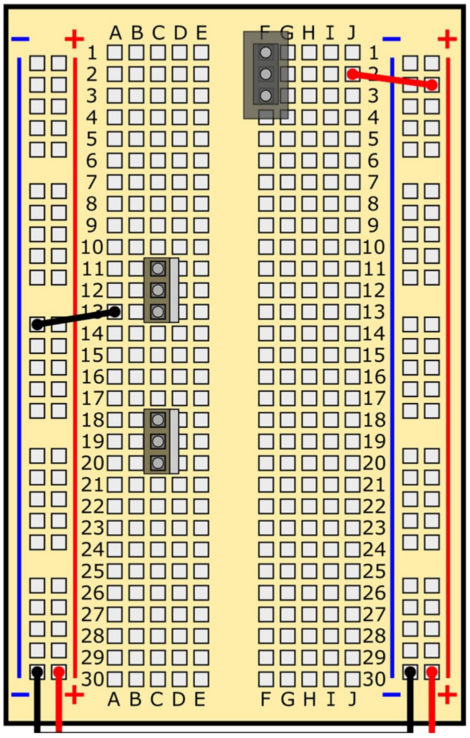




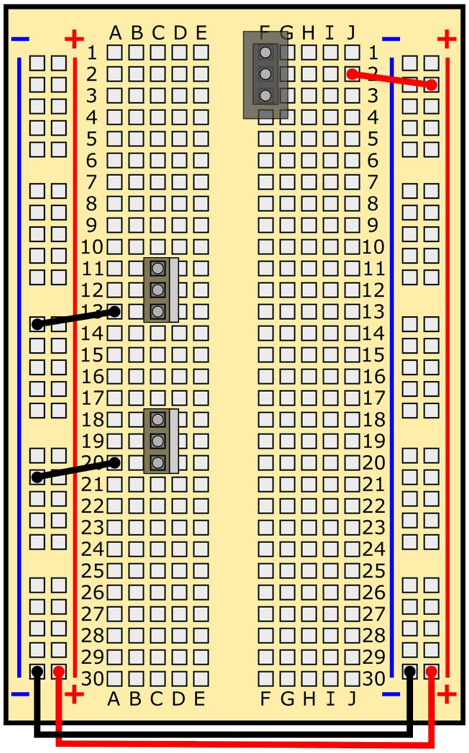
6. Insert MOSFET into holes C11, C12, C13. Writing must face to the left.



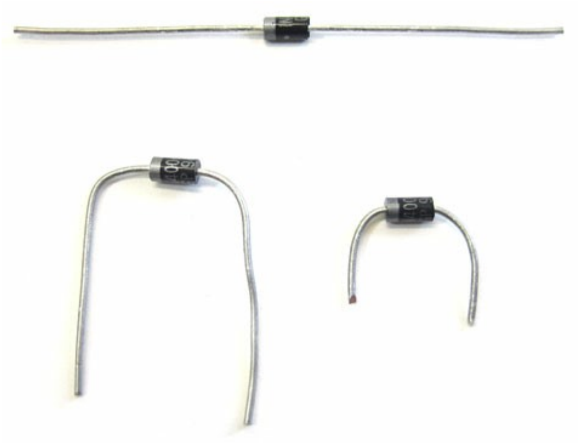
7. Insert MOSFET into holes C18, C19, C20. Writing must face to the left.



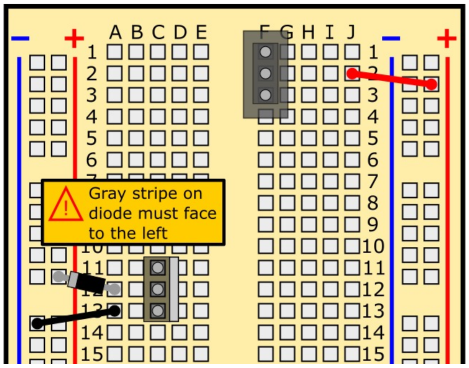
8. Connect jumper wire from hole A13 to (-) bus.



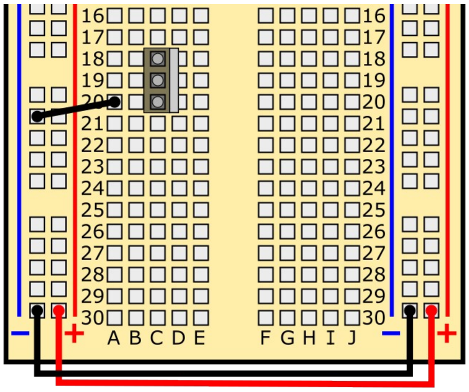
9. Connect jumper wire from hole A20 to (-) bus.



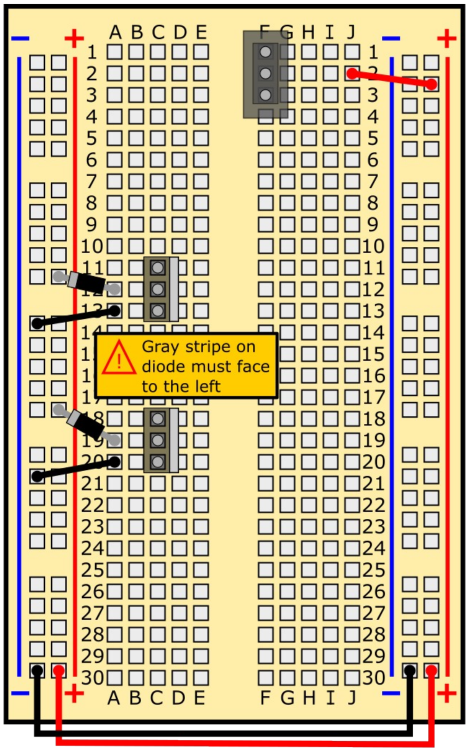
10. Bend the diode's leads down by 90 degrees and trim them so they fit into the breadboard.



Gray stripe on diode must face to the left

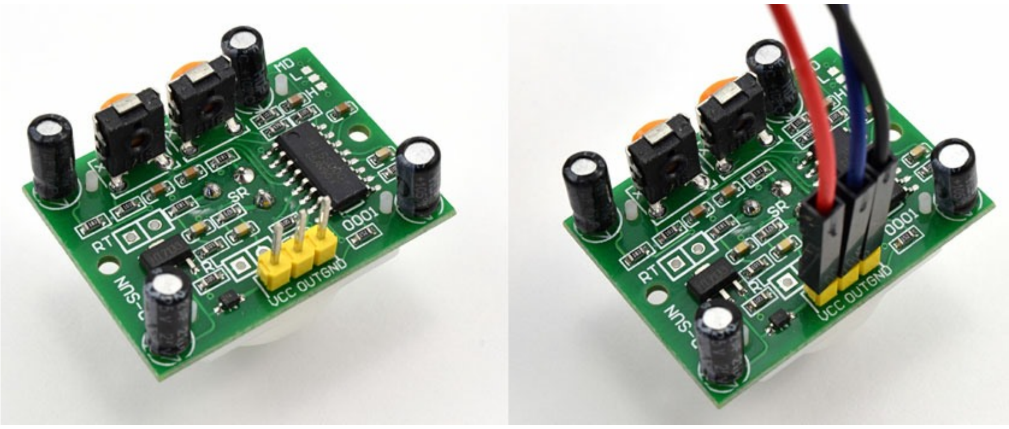


11. Insert diode into hole A12 and (+) bus. Gray stripe must face to the left.

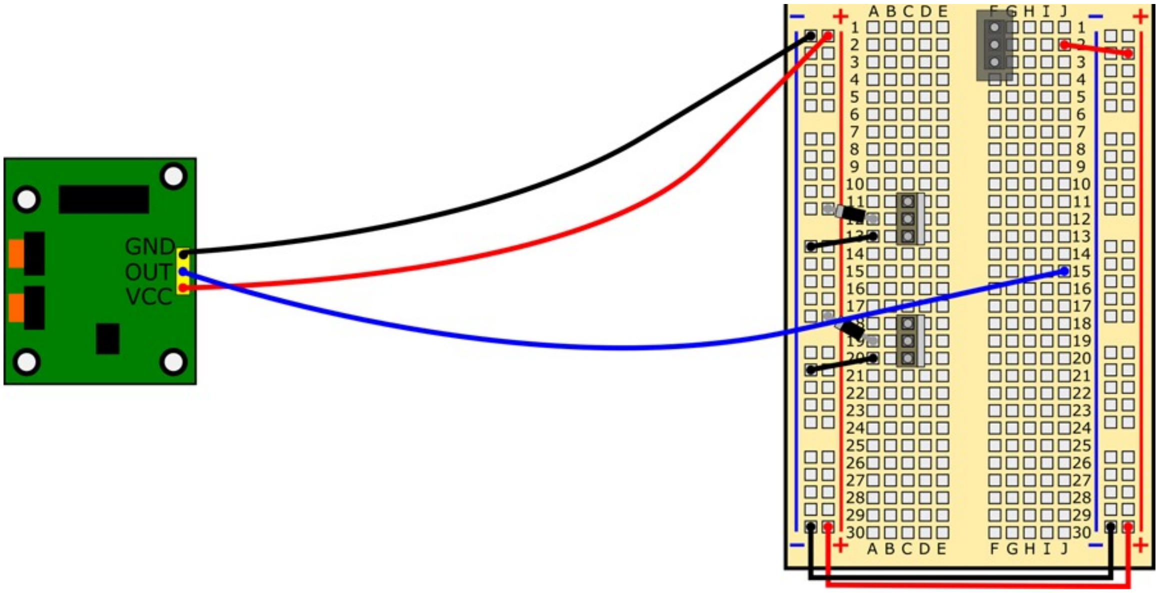


Gray stripe on diode must face to the left

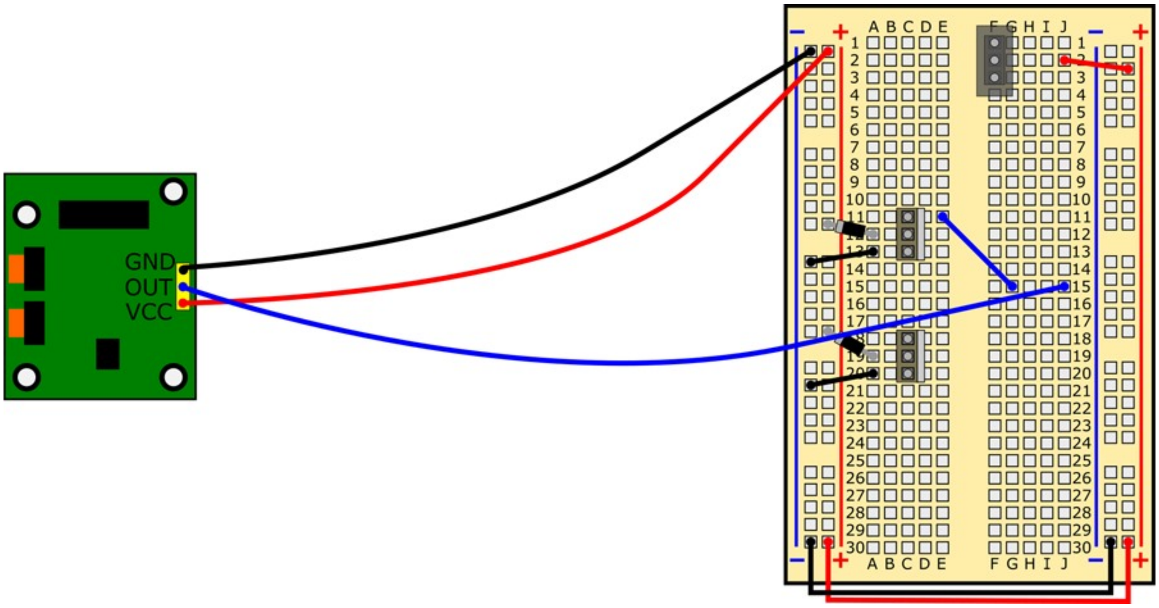
12. Insert diode into hole A19 and (+) bus. Gray stripe must face to the left.



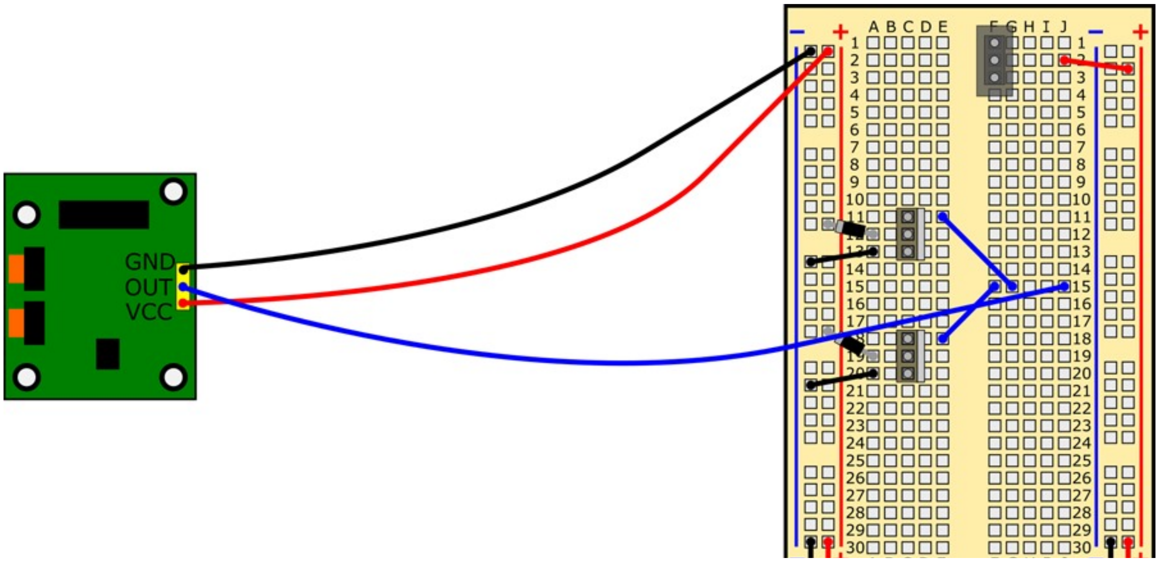
13. Connect a red male-female jumper wire to the VCC pin of the PIR sensor, a blue wire to the OUT pin, and a black wire to the GND pin.



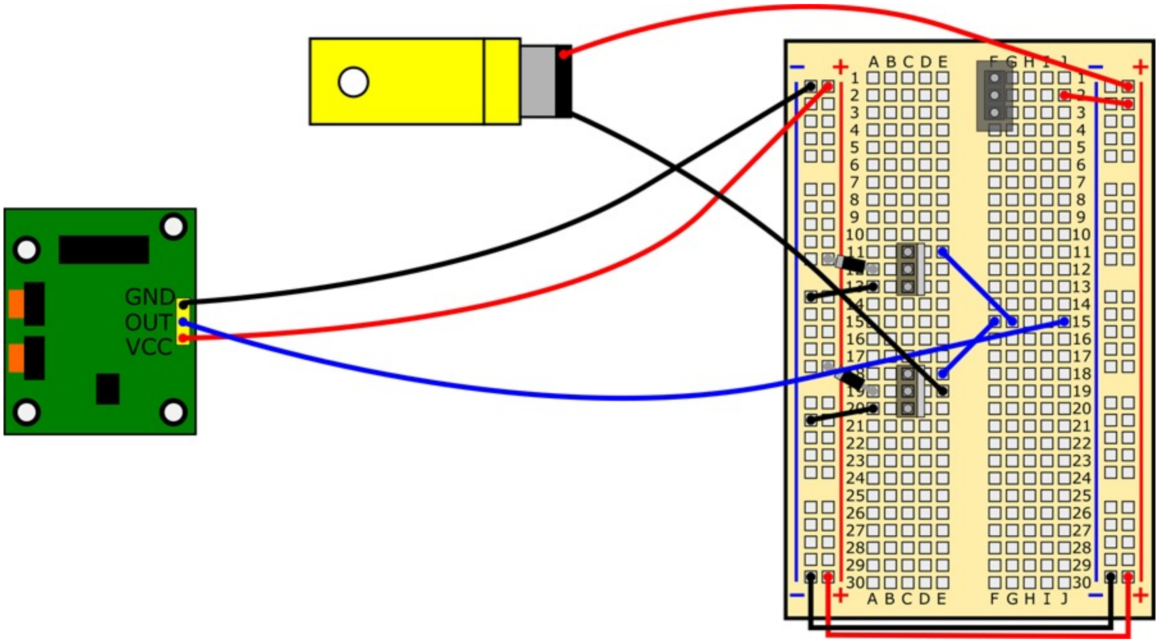
14. Connect PIR sensor: GND pin to (-) bus, OUT pin to J15, VCC pin to (+) bus.



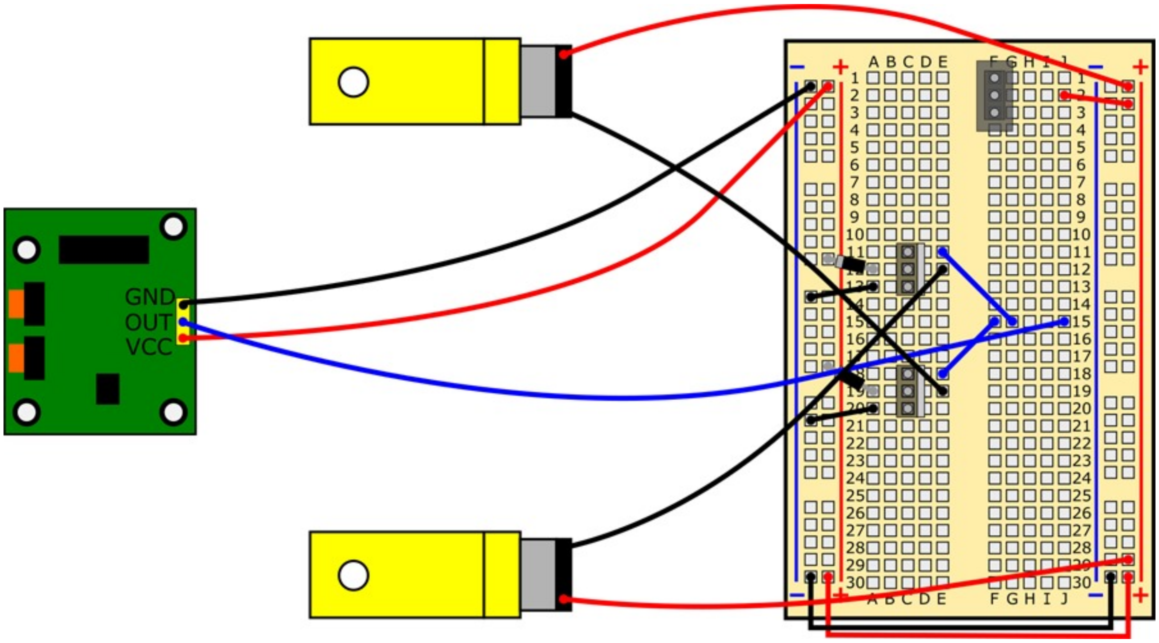
15. Connect jumper wire from hole E11 to G15.



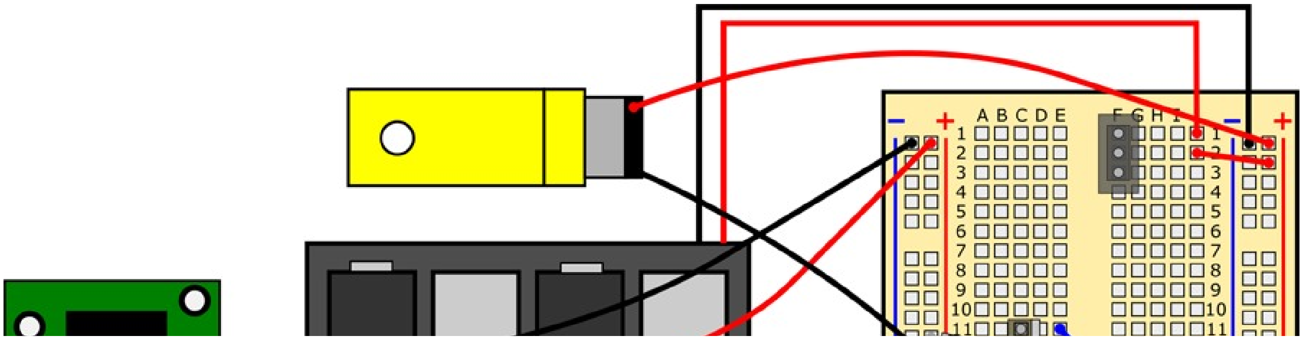
16. Connect jumper wire from hole E18 to F15.

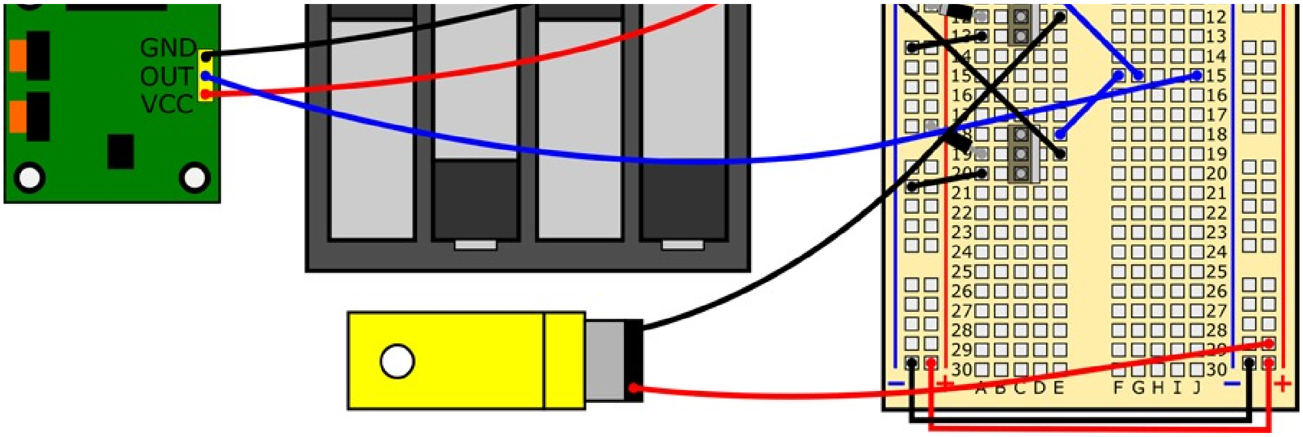


17. Connect top motor's red wire to (+) bus, black wire to hole E19.



18. Connect bottom motor's red wire to (+) bus, black wire to hole E12.





19. Connect battery holder's red wire to hole J1, black wire to (-) bus.

End of Slideshow Images

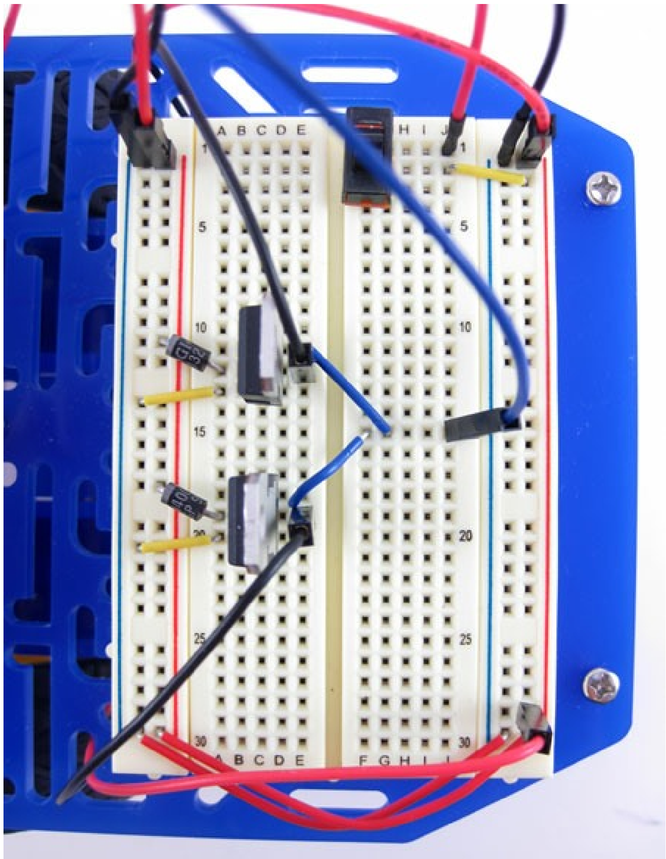


Figure 5. Your completed circuit should look like this.

3. After you have completed your circuit, mount the PIR sensor to the front of your robot chassis using double-sided foam tape, as shown in Figure 6.

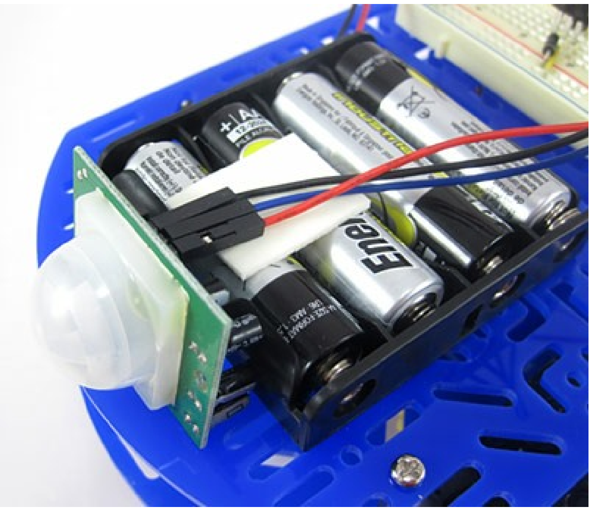


Figure 6. Mount the PIR sensor to the front of your robot using double-sided foam tape.

Testing Your Robot

You are finally ready to start testing your robot! Remember that now you will need to follow the engineering design process to get your robot working. Follow these steps to learn how to use your robot.

1. Double-check your circuit against the breadboard diagrams in the previous section. Remember that just *one* misplaced wire can prevent the circuit from working properly.
2. Your PIR sensor has two small orange knobs on it, labeled Sx and Tx (Figure 7). These knobs can be adjusted using a small Phillips-head screwdriver. Tx adjusts the "delay on time," or how long the sensor will remain on after detection motion. This value can range from 3 to 200 seconds. To start, we recommend turning this knob all the way counter-clockwise to set the delay to 3 seconds (otherwise, your robot will drive forward too far when it detects motion). You can increase the delay later if you want. Sx adjusts the sensor's sensitivity to motion.

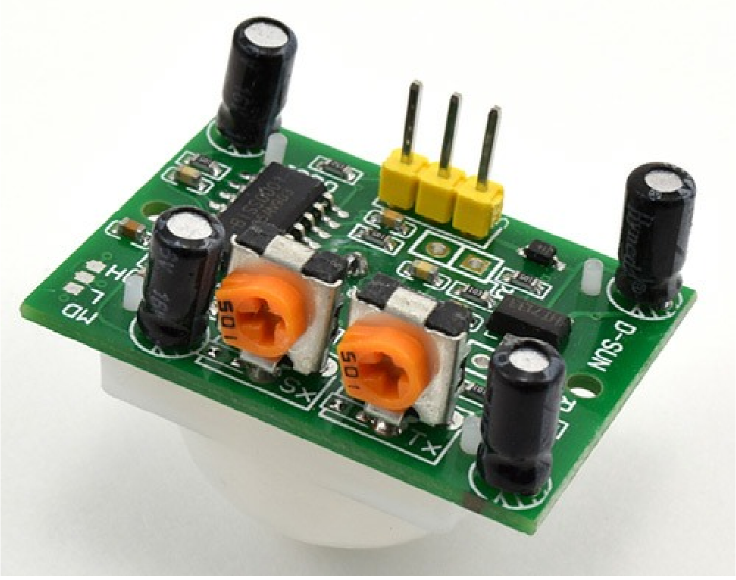


Figure 7. Adjustment knobs on the PIR sensor. Sx adjusts sensitivity and Tx adjusts delay time.

3. Hold the robot's chassis in one hand, so the wheels are off the ground, and the PIR sensor is facing away from you and any other sources of infrared light, like other people, animals, or an open window (sunlight contains some infrared light).
   a. Turn the robot's power switch "on" by sliding it up, toward row 1 on the breadboard.
   b. Your robot's wheels should spin briefly as the PIR sensor "boots up," then come to a stop. Check Table 3 to see what you should do next.

Observation	What to Do
I see or smell smoke.	Immediately turn your robot off. You have a short circuit somewhere. Recheck your wiring against the breadboard diagrams in the previous section.
Each wheel spins forward when I turn the robot on.	Your robot works! Move on to the next step.
One or both wheels spin backwards when I turn the robot on.	Reverse the red and black wires of the motor if the wheel is spinning backwards. See the Help (http://www.sciencebuddies.org/science-fair-projects/project_ideas/Robotics_p024.shtml#question2) section for details.
The wheels did not spin at all when I turned the robot on.	See the Help (#help) section for troubleshooting steps.

Table 3. Troubleshooting procedure for the first time you turn on your robot.

4. Now, try waving your hand in front of the PIR sensor. This should make your robot's wheels spin. If your robot is not working, see the Help (#help) section.
5. Try to discover other ways that you can activate the robot. What happens if you put the robot on the floor and then walk in front of it? What happens if you wave an inanimate object, like a ruler, in front of the sensor, or throw a crumpled piece of paper in front of the robot?
6. You might discover that it is actually too *easy* to activate the robot, because the PIR sensor has such a wide field of view. For example, if you try to hide the robot under a desk or table, it might jump out too early when someone walks by. This is where the engineering design process really comes into play. Can you figure out how to modify the sensor's field of view? For example, you could partially cover the sensor with a shortened paper towel tube, like in Figure 8. You can also adjust which way the PIR sensor is facing (forward, down, up, or to the side).

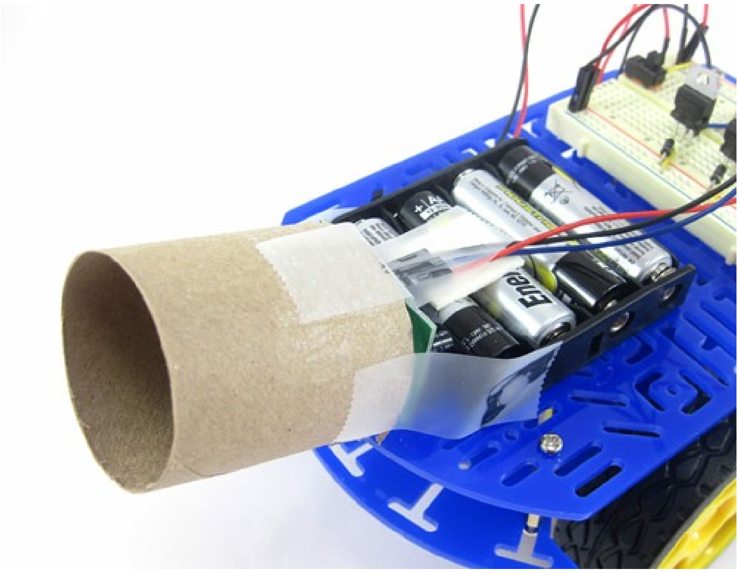


Figure 8. Partially covering the sensor with a shortened paper towel tube will help narrow its field of view.

7. Continue making adjustments to your PIR sensor, as needed. You do not want the robot to activate too early or too late. What works best for the situation in which you want to use the robot? For example, do you want the robot to wait in a room and drive toward whoever walks in? Do you want to hide it around a corner or under a table? Once you have your robot working just the way you like it, leave it set up to "guard" something, and scare away the next person who comes by!

.

If you like this project, you might enjoy exploring these related careers:

(http://www.sciencebuddies.org/science-engineering-careers/engineering/robotics-engineer)

Robotics Engineer (http://www.sciencebuddies.org/science-engineering-careers/engineering/robotics-engineer)

Have you watched "The Transformers" cartoon series or seen the "Transformers" movies? Both shows are about how good and evil robots fight each other and the humans who get in the middle. Many TV shows and movies show robots and humans interacting with each other. While this is, at present, fantasy, in real life robots play a helpful role. Robots do jobs that can be dangerous for humans. For example, some robots defuse landmines in war-stricken countries; others work in harsh environments like the bottom of the ocean and on the planet Mars. At the heart of every robot is a robotics engineer who thinks about what a robot needs to do and works with several engineering disciplines to design and put together the perfect piece of equipment. Read more (http://www.sciencebuddies.org/science-engineering-careers/engineering/robotics-engineer)

(http://www.sciencebuddies.org/science-engineering-careers/engineering/robotics-technician)

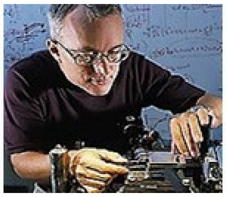
Robotics Technician (http://www.sciencebuddies.org/science-engineering-careers/engineering/robotics-technician)

Robots are no longer futuristic machines. Robots are here and now and are used in manufacturing, health care, service industries, and military applications. They perform tasks that are repetitive and hazardous—things that humans don't want to do or are unsafe to do. But robots are still machines, which means they require humans to build, maintain, program, and keep them functioning efficiently. Robotics technicians work with robotics engineers to build and test robots. They are responsible for installing and maintaining robots and keeping them in working order for their employers. If you are interested in working with robots, your future is here and now. Read more (http://www.sciencebuddies.org/science-engineering-careers/engineering/robotics-technician)

(http://www.sciencebuddies.org/science-engineering-careers/engineering/electrical-electronics-engineer)

Electrical & Electronics Engineer (http://www.sciencebuddies.org/science-engineering-careers/engineering/electrical-electronics-engineer)

Just as a potter forms clay, or a steel worker molds molten steel, electrical and electronics engineers gather and shape electricity and use it to make products that transmit power or transmit information. Electrical and electronics engineers may specialize in one of the millions of products that make or use electricity, like cell phones, electric motors, microwaves, medical instruments, airline navigation system, or handheld games. Read more (http://www.sciencebuddies.org/science-engineering-careers/engineering/electrical-electronics-engineer)



(http://www.sciencebuddies.org/science-engineering-careers/engineering/mechanical-engineer)

Mechanical Engineer (http://www.sciencebuddies.org/science-engineering-careers/engineering/mechanical-engineer)

Mechanical engineers are part of your everyday life, designing the spoon you used to eat your breakfast, your breakfast's packaging, the flip-top cap on your toothpaste tube, the zipper on your jacket, the car, bike, or bus you took to school, the chair you sat in, the door handle you grasped and the hinges it opened on, and the ballpoint pen you used to take your test. Virtually every object that you see around you has passed through the hands of a mechanical engineer. Consequently, their skills are in demand to design millions of different products in almost every type of industry. Read more (http://www.sciencebuddies.org/science-engineering-careers/engineering/mechanical-engineer)

Variations

- There are three other projects you can do with your BlueBot kit. Since you have already assembled your chassis, all you need to do is build a new circuit.
  - Build a Speedy Light-Tracking Robot (BlueBot Project #2) (http://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p022/robotics/light-following-robot)
  - Build a Zippy Line-following Robot (BlueBot Project #3) (http://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p023/robotics/line-following-robot)
  - Build an Obstacle-Avoiding Robot (BlueBot Project #4) (http://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p028/robotics/obstacle-avoiding-robot)
- The sensor's circuit board has a *jumper* that you can use to toggle the detection range between 15 or 30 feet. You can read more about how to do this on the sensor's datasheet (http://www.avantlink.com/click.php?tt=cl&mi=10609&pw=182414&ctc=guard-robot&url=http%3a%2f%2fwww.jameco.com%2fJameco%2fProducts%2fProdDS%2f2082927.pdf). How does adjusting the jumper affect the performance of your robot?
- The other robots in your BlueBot use something called *differential steering*. They use two different sensors to help the robot steer left and right. In this project, you only used one sensor to make the robot drive forward. Can you use two PIR sensors to make a motion-tracking robot that will actually follow a person around? This might be more difficult than it sounds, so be ready for a challenge!
- The Arduino (https://www.arduino.cc/) is a very popular type of *microcontroller* used in robotics. It lets you write a computer program that can read inputs from sensors and use them to control motors. This gives you more precise "control" over your robot's behavior. Can you build a programmable motion-activated guard robot by adding

an Arduino to your chassis? For example, using the other parts in your BlueBot kit (the H-bridge "motor driver" found in the Obstacle-Avoiding project), you can make a robot that drives forward when it detects motion, then goes in reverse to "hide" again. You could also make the robot wait a certain amount of time after detecting motion before it drives forward. See our Getting Started with Arduino (http://www.sciencebuddies.org/science-fair-projects/references/getting-started-with-arduino) page to learn more.

- You can also use a PIR sensor to make a homemade alarm system, using a tiny programmable computer called a Raspberry Pi. See the Science Buddies Raspberry Pi Projects Kit (http://www.sciencebuddies.org/science-fair-projects/project-ideas/raspberry-pi)—specifically the Make a Motion Sensor Alarm with a Raspberry Pi (http://www.sciencebuddies.org/science-fair-projects/project-ideas/CompSci_p055/computer-science/motion-sensor-alarm-raspberry-pi) project—to learn more.

Frequently Asked Questions (FAQ)

If you are having trouble with this project, please read the FAQ below. You may find the answer to your question.

- My robot's wheels do not spin at all when I turn it on, or when I wave my hand in front of the PIR sensor. What should I do? (#question1)
- Only one of my robot's wheels is spinning. What is wrong? (#question2)
- My robot is going backwards! What should I do? (#question3)
- My robot keeps moving on its own, even when no one walks in front of the sensor! What is happening? (#question4)
- I have double-checked everything and my robot still does not work. How can I check if something is broken? (#question5)
- How does the PIR sensor work? (#question6)
- How does a MOSFET work? (#question7)
- How does the circuit work? What is the circuit diagram? (#question8)
- What are the circuit diagram symbols for the components in this project? (#question9)
- How did you make the breadboard diagrams for this project? (#question10)

Q: My robot's wheels do not spin at all when I turn it on, or when I wave my hand in front of the PIR sensor. What should I do?
A: Try these steps:

- Double-check *all* of your breadboard connections. Make sure every lead and jumper wire is firmly pressed into the breadboard. Make sure all of your connections match the breadboard diagrams from the Procedure (#procedure). You can read about other common breadboard mistakes in the the Common Mistakes (http://www.sciencebuddies.org/science-fair-projects/breadboard-tutorial#common-mistakes) section of our breadboard tutorial.
- Make sure all your batteries are properly inserted into the battery holder, with the "+" symbols on the batteries lined up with the "+" symbols inside the holder.
- If you have been using your robot for a long time, or did some of the other BlueBot projects first, your batteries may be dead. Try putting fresh batteries in your robot if none of the other steps work.
- If your robot still does not turn on, follow these steps to see if the problem is with your power switch:
    - Remove the power switch from the breadboard, flip it around, and put it back into the same breadboard holes (F1, F2, and F3). If you are confused about which way to rotate the switch, see this video (https://youtu.be/zkGBW5JSqls).
    - Slide the power switch to the "on" position ("up" towards row 1 of the breadboard) and try testing your robot again. If your robot works after making this change, you can continue with the project, but please contact us at scibuddy@sciencebuddies.org (mailto:scibuddy@sciencebuddies.org?subject=Guard%20BlueBot:%20switch%20worked%20when%20flipped) to let us know this happened.
    - If your robot still does not work after flipping the power switch around, bypass the power switch entirely. Connect the battery pack's red lead directly to the breadboard's power bus instead of hole J1. This connects the battery pack directly to the rest of the circuit. To turn the robot off again, you will have to temporarily disconnect this wire.
    - Now, re-try testing your robot. If your robot works, you can continue with the project, but please contact us at scibuddy@sciencebuddies.org (mailto:scibuddy@sciencebuddies.org?subject=Guard%20BlueBot:%20switch%20did%20not%20work%20at%20all) to let us know the switch did not work in either direction.
    - If your robot still does not work, even after you have tried the previous steps, the problem is elsewhere in your circuit and not with the power switch. If you are having trouble figuring out what is wrong with your robot, you can ask a question in our Ask an Expert: Answers to Your Science Questions (http://www.sciencebuddies.org/science-fair-projects/ask-an-expert-intro) forums.

Q: Only one of my robot's wheels is spinning. What is wrong?
A: Your robot's circuit consists of two identical halves that drive the two motors. If only one wheel is spinning, chances are you just have something placed incorrectly on the breadboard in one half of the circuit. Very carefully double-check all of your writing on the side of the motor that is not working. It only takes *one* misplaced component or jumper wire to prevent the motor from spinning!

Q: My robot is going backwards! What should I do?
A: If one or both of your robot's wheels is spinning backwards, you just need to switch the red and black leads for the motor(s). This will cause the motor(s) to spin in the opposite direction. These connections are shown in Figure 9.

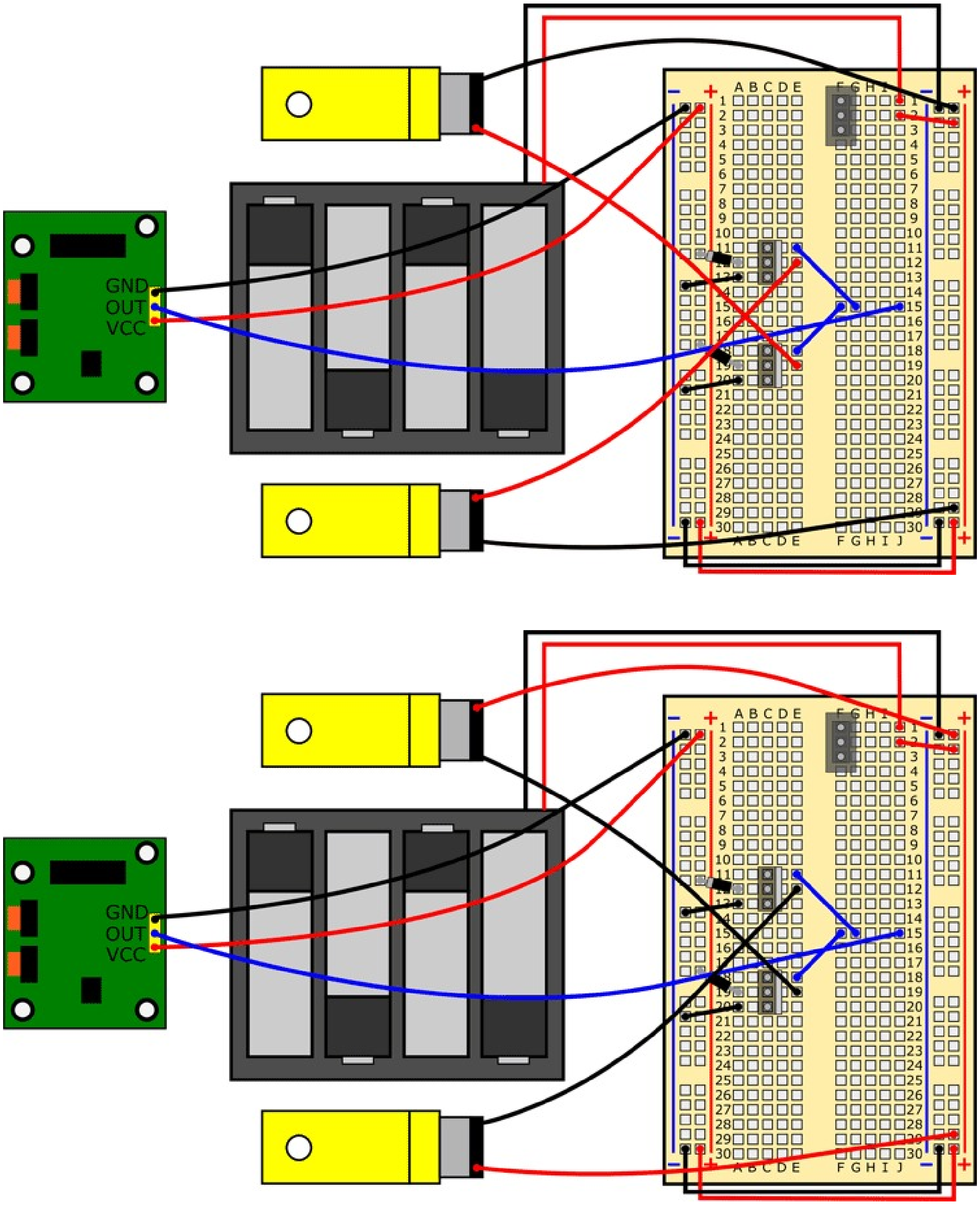


Figure 9. In the top image, the motors' red leads are connected to the power bus, and the black leads are connected to the MOSFETs' drain pins (holes E4 and E27). If this configuration causes your robot to go backwards, just switch the red and black wires for each motor, as shown in the bottom image. Connect the black wires to the power bus, and the red wires to holes E4 and E27. Do not worry about violating the color-coding convention for positive and negative; all this does is reverse the direction in which the motors spin. If only one motor is spinning backwards, only switch the wires for that motor.

Q: My robot keeps moving on its own, even when no one walks in front of the sensor! What is happening?
A: Remember that the PIR sensor is *very* sensitive, and has a rather wide field of view. So, it is possible to trigger the sensor even if you are to the side of the robot and not directly in front of it. There could also be other sources of infrared light in the room, or something in the room could reflect infrared light from your body towards the sensor. To make sure you are not triggering the sensor accidentally, try putting the robot under an upside-down cardboard box, or putting it in a dark closet to block out all possible sources of infrared light. If you can still hear the robot's wheels spinning, then there may be something wrong with your circuit that is causing them to spin independently of the sensor. If this is the case, you should double-check all of your wiring. If your robot stops moving when you put it under a box, then that means you were accidentally triggering the sensor somehow.

Q: I have double-checked everything and my robot still does not work. How can I check if something is broken?
A: If you are having trouble figuring out why your robot will not work, you can try asking a question in our Ask an Expert: Answers to Your Science Questions (http://www.sciencebuddies.org/science-fair-projects/ask-an-expert-intro) forums. If you have access to a multimeter, you can also use that to help troubleshoot your circuit. See the Science Buddies reference How to Use a Multimeter (http://www.sciencebuddies.org/science-fair-projects/references/how-to-use-a-multimeter) if you need help using a multimeter. There are several steps you can take to check individual parts of your circuit (depending on your multimeter's probes, you may need alligator clips and additional jumper wires to take these measurements).

- Try plugging your motor's leads directly into the buses on your breadboard (for each motor, one lead to the power bus, one lead to the ground bus). If the motors turn on, then you know that they are working, and the problem is elsewhere in your circuit. If the motors do not turn on, that does not necessarily mean they are broken. There might be a problem with your power supply (see next point).
- Check your breadboard's power supply.
  - Set your multimeter to measure DC volts. Four fresh AA batteries should provide just over 6 V.
  - Turn your power switch on and measure the voltage between the breadboard's buses. If you do not read a voltage, there may a problem with your power switch (see the first question in this FAQ), but you should also double check that your power switch and the red jumper wire from hole J2 to the power bus are in the right place.
  - Unplug the red and black battery pack leads and measure the voltage of the battery pack directly. If you get a reading, then you know your battery pack is working, and the problem is with your breadboard connections or the power switch. If you do not get a reading, make sure all the batteries are in the correct orientation in your battery pack, and that none of the metal clips and springs that hold the batteries in place are loose.
- Measure the voltage of your PIR sensor's output pin (labeled OUT on the sensor, connected to row 15 on the right side of the breadboard). It should read 0 V when the sensor is not triggered, and between 5–6 V when the sensor detects motion.
- If the motors work when you plug their leads directly into the buses, and the output voltage of the PIR sensor increases when you wave your hand in front of it, but the robot still does not work, then the problem is elsewhere in your circuit. You know the individual components are functioning so nothing is "broken." Double and triple-check all your breadboard connections. Remember that just one misplaced wire can prevent the whole circuit from working.

Q: How does the PIR sensor work?
A: See the references from LadyAda and Glolab Corporation in the Bibliography (#bibliography) for detailed descriptions of how PIR sensors work.

Q: How does a MOSFET work?
A: MOSFET stands for *metal-oxide-semiconductor field-effect transistor* (so you can see why it is a lot easier just to say "MOSFET"). The three pins of a field-effect transistor are called the **gate**, **drain**, and **source**. Unlike a *bipolar* transistor, which is controlled by a small current applied to the base pin, a *field-effect* transistor is controlled by a voltage applied to the gate pin, but the gate does not actually draw any current. A voltage applied to the gate causes current to flow between the drain and source pins.

Figure 10 shows a simplified explanation of how a MOSFET works. A voltage is applied to the gate pin in order to control the flow of current between the drain and source pins. When the voltage between the gate and source pins (V_{GS}) is below a certain limit, called the **threshold voltage** (V_{th}), no current flows. When V_{GS} exceeds V_{th}, the MOSFET begins to conduct, allowing current to pass through. This is what allows you to use the gate voltage of a MOSFET to turn a DC motor on and off. For this robot, the MOSFET's gate voltage is controlled by the PIR sensor's output.

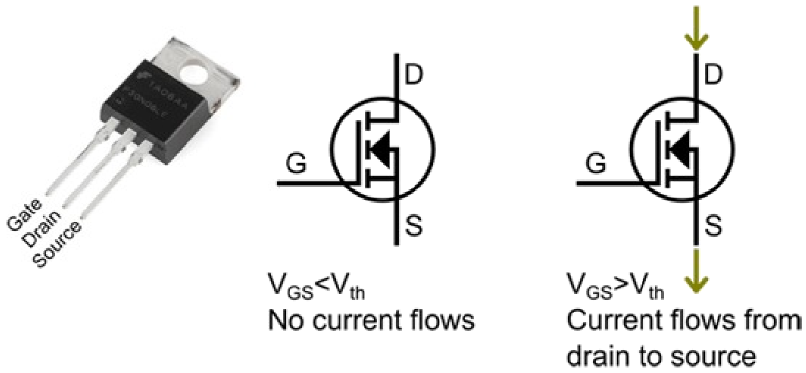


Figure 10. Simplified explanation of a MOSFET's operation.

The exact description of how a MOSFET works is more complicated than this. As V_{GS} increases past V_{th}, the current through the MOSFET will continue to increase. Eventually the MOSFET will reach **saturation**, where no additional current can flow, even if V_{GS} continues to increase. The MOSFET's behavior will also depend on the type of **load** to which it is attached. The MOSFET used in this project is an **N-channel MOSFET**, which requires a positive gate voltage to turn on. A **P-channel MOSFET** requires a negative gate voltage to turn on. Advanced users can refer to the Bibliography for more information on MOSFETs.

Q: How does the circuit work? What is the circuit diagram?
A: Figure 11 shows a complete circuit diagram for the robot. Refer to Table 4 for a list of individual circuit components and their respective circuit symbols. If you need help learning how to read a circuit diagram, refer to the SparkFun reference How to Read a Schematic (https://learn.sparkfun.com/tutorials/how-to-read-a-schematic). Here is a description of how the circuit works:

- The batteries supply a voltage, V_{batt} to the circuit. When the switch is closed, $V_1 = V_{batt}$. When the switch is open, V_1 is "floating", so the circuit does not receive any power.
- The PIR sensor receives power from its VCC pin and is connected to the circuit's ground through its GND pin. The sensor is *digital*—it outputs a logical "low" or "high" from its OUT pin depending on whether or not it detects motion. A "low" signal is 0 V, and a "high" signal is approximately 5 V.
- 5 V is above the MOSFET's threshold voltage. So, when the sensor goes "high," this turns on both MOSFETs (notice how the OUT pin of the sensor is connected to the gate pin of both MOSFETs). This allows current to flow through the motors, and the robot drives forward.
- Finally, each motor has a diode connected across its terminals. Motors can create large voltage spikes when they abruptly come to a stop (this has to do with the relationship between electrical current and magnetic fields, if you want to do more research on the explanation). The diodes help prevent damage to the MOSFET by safely discharging the current generated by the voltage spike.

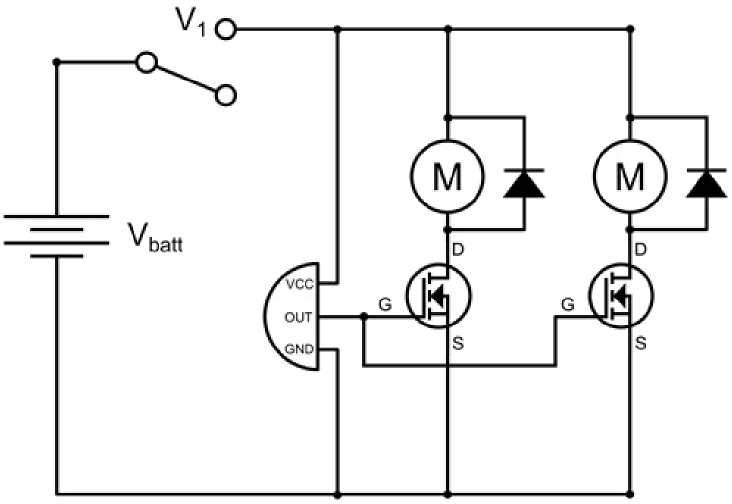
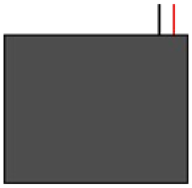
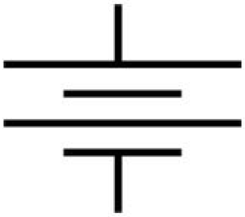
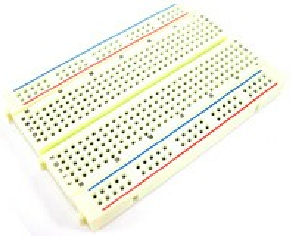
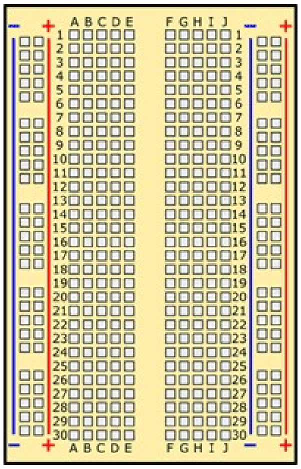
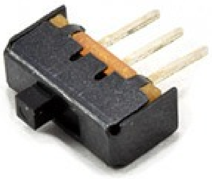
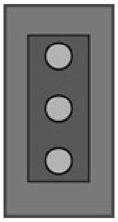
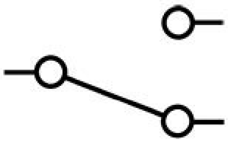


Figure 11. A complete circuit diagram for the motion-activated robot.

Q: What are the circuit diagram symbols for the components in this project?
A: Table 4 shows a picture, breadboard diagram symbol, and circuit diagram symbol (when applicable) for each component in the project.

Item name	Picture	Breadboard Diagram Symbol	Circuit Diagram Symbol
Battery pack			
Breadboard			n/a
Switch			

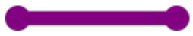
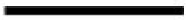
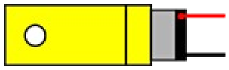
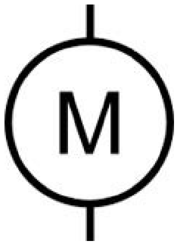
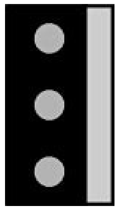
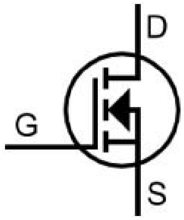
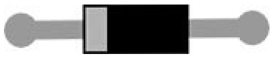
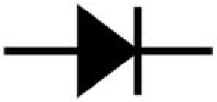
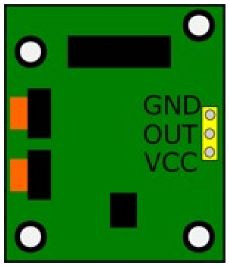
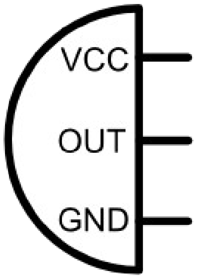
Item name	Picture	Breadboard Diagram Symbol	Circuit Diagram Symbol
Jumper wire			
DC Motor			
MOSFET			
Diode			
PIR Sensor			

Table 4. A picture, breadboard diagram icon, and circuit diagram symbol for each of the components used in this project.

Q: How did you make the breadboard diagrams for this project?

A: The breadboard diagrams for this project were created using Inkscape (http://inkscape.org/en/), a free vector graphics program. You can find free scalable vector graphic (SVG) files for many circuit components on Wikimedia Commons (http://commons.wikimedia.org/wiki/Main_Page). There are other free programs specifically for making breadboard diagrams, such as Fritzing (http://fritzing.org/home/).

Ask an Expert

The Ask an Expert Forum is intended to be a place where students can go to find answers to science questions that they have been unable to find using other resources.