**Lesson 31: Photoresistors**

Needed

* Boe-Bot
* Computer with BASIC Stamp Editor program
* USB cable
* Piezospeaker
* Jumper wires
* 0.01μF capacitors
* 220Ω resistor

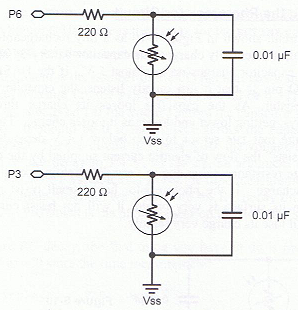
Photoresistors

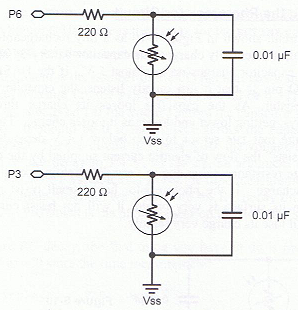
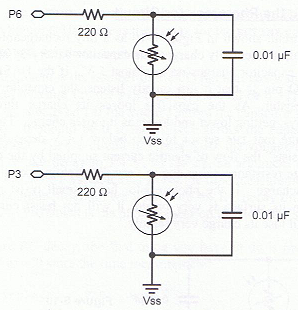
Last class we discussed the circuit associated with the piezospeakers. In this class we will implement a photoresistor circuit in conjunction with a piezospeaker to have the Boe-Bot change frequency output based upon the amount of light intensity the photoresistor detects.

In order to complete this task, lets first look at the photoresistor circuit. If the piezospeaker circuits are still set up on the students Boe-Bots, have the students leave the circuit and install the photoresistor circuit around the piezospeaker circuit.

First, what exactly is a photoresistor? Photoresistors are light dependent resistors (LDR). LDRs have resistance values that vary dependent upon the amount of light that is reflecting off the light detecting surface. In many instances a photoresistor can be used. Some common uses of photoresistors are in camera light meters and street lights. Photoresistors can also be used as a means of providing an input to the Boe-Bot so that a certain output is performed. Photoresistors can be used to aid the Boe-Bot in identifying a path to travel. Next class we will have our Boe-Bots use the photoresistor to measure light intensity which will then tell the Boe-Bot the frequency output it should provide.

Before we begin controlling the sound output due to light intensity, let’s look quickly at the circuitry behind the photoresistor setup. The following schematic applies to the pohotoresistor circuitry. Students should create this circuit on their Boe-Bot breadboards.



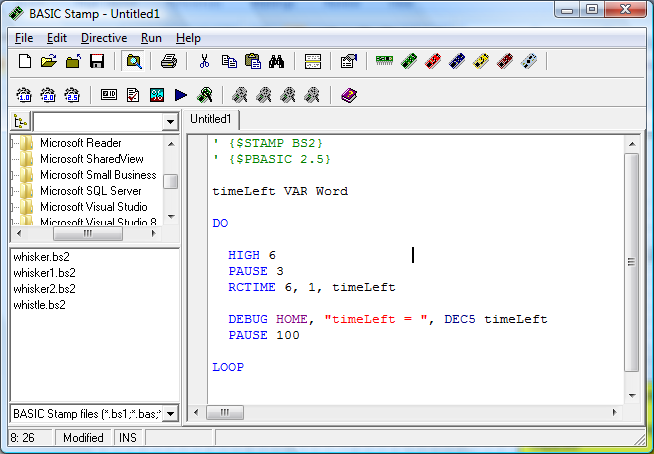
This symbol represents the photoresistor. Another unfamiliar symbol, , represents a capacitor. Capacitors are mainly used to store charge. There are many sizes of capacitors; knowing the right one to use is important. Capacitance is represented in units of Farads. For the purpose of our class, capacitors of degree Farads(F) are too large for practical use with the Boe-Bots. Therefore, we will use capacitors containing a fraction of the Farad to the millionth degree, a microfarad (µF). For the above schematic a 0.01μF capacitor is used.

In order to use the photoresistor to make the Boe-Bot perform a function, the programming must be such that the capacitor conducts a sequence of charging and discharging. In order to do so the command utilized is the RCTIME command. The syntax for the command reads as follows.

RCTIME Pin, State, Duration

Dissecting the syntax, the pin represents the pin at which the circuit is connected to and is being measured. The state will either be a 1 or 0. If the state is a 1, then the voltage across the capacitor starts at a value greater than 1.4V and will decay. On the other hand, if the state is indicated as a 0, then the voltage across the capacitor is less than 1.4V and will show growth upward. The duration is the variable that will store the time that is being measure. This time measurement will be stored in units of 2µs.

Students should test the photoresistor using the RCTIME command. Performing this test will not only aid in testing if the photoresistor is working, but will also identify the decay time of the capacitor. Students should type the following program into their Boe-Bots (Notice the state associated with the RCTIME command is set to 1 for decay):



**High:** starts charging the capacitor

**Pause:** provides enough time to charge the capacitor

**RCTIME:** measures the decay time

Students can cover the photoresistors to reduce the amount of light shining on it. The timeLeft value should change (If this is the case the photoresistor is working). Does the timeLeft get larger or smaller as it is covered (ans. It should get larger when the photoresistor is covered)? What does the increase/decrease mean?

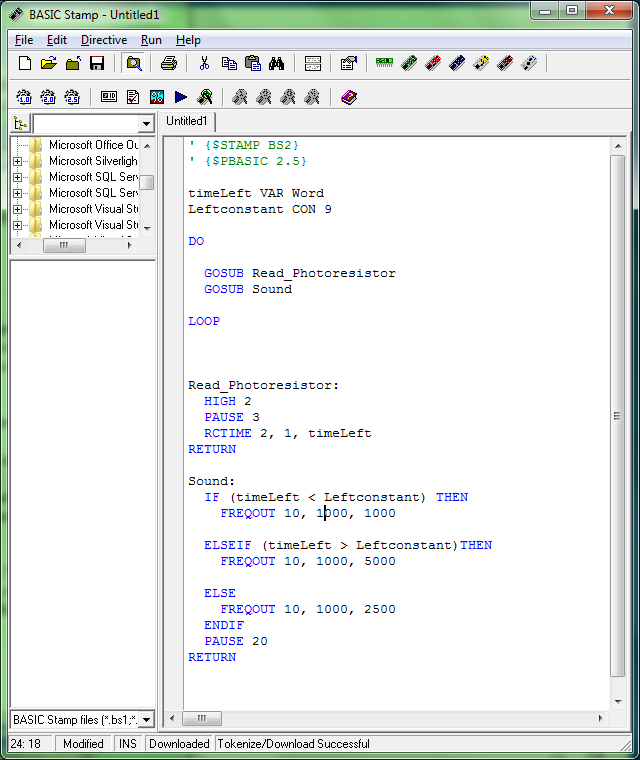
It is important to note the value of the timeLeft when the photoresistors are exposed to the ambient light, students will use this value for the calibration of the photoresistors.

Light Intensity Controlled Piezospeaker Output

Now, that the photoresistor circuit is properly installed. Let’s look back at the piezospeaker circuit (if circuit is not installed, have students install it now around the photoresistor circuit). We want to control the frequency output of the piezospeaker based upon the light intensity that the photoresistor detects.

In order to do so, students must use the timeLeft value determined from the calibration of the photoresistors. This value will become a constant that is used for the basis of comparison of light change, and in turn frequency output. (Note constants are stated at the beginning of a program using the following statement: NameOfConstant CON Value). Provide the student with the hint that using subroutines may be helpful: one that reads the light intensity and another that provides frequency output for the various light intensities.

A solution to this exercise is below. This solution only looks at three values of light intensity, greater than the constant, equal to the constant, and less than the constant. Task the student will developing their own method. Is there a more elegant way of writing the program? Can you make the frequency play more fluidly? Can you account for more than just three variation in light intensity, etc?

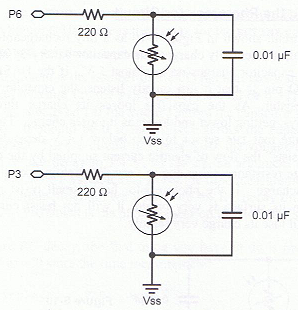
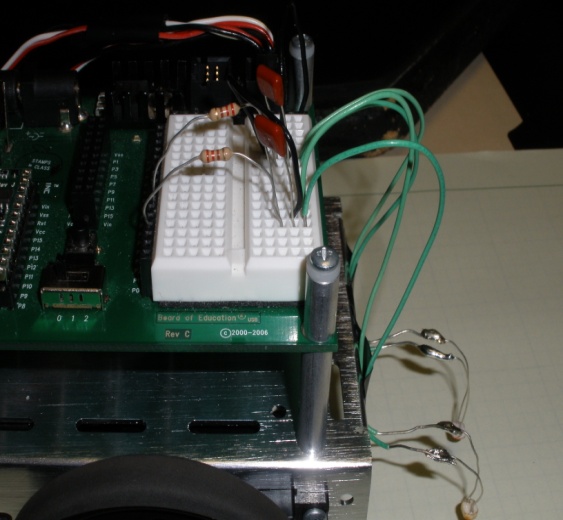


Following a Line with Two Photoresistors

In four class periods, students will compete in a task that requires their Boe-Bots to use photoresistors to navigate over a black line. Introduce the task to the students here. Throughout the week students can work on their program and be ready to make their Boe-Bot complete the task in four classes.

The task follows the below procedure:

1. Students should install a second photoresistor circuit as shown in the below schematic (they can remove the piezospeaker now if they want).

**Teachers:** Our intention here is that prior to this class you would have constructed this course for the students. A suggestion is to use white poster board material as the base for the course. The bright with color of the boards will provide a good contrast and will make it easier for the Boe-Bot to stay on task.

1. Obtain a course made from black electrical tape that is intended for the Boe-Bot to follow. It must be a continuous line. A suggestion is to Use 3-4 lines of tape to make the line wider. This will help in navigation. The line should have at least one straight section, one right turn, and one left turn.
2. Calibrate your Boe-Bot to the light reflected by the tape and the light reflected by surface the course is mapped out on.
3. Place the Boe-Bot with both photoresistor over the line of tape and run same calibration program from the following a flashlight activity. Record the values of timeLeft and timeRight.
4. Place the Boe-Bot with both photoresistor over the floor or table. Record the values of timeLeft and timeRight
5. Average the two values for each photoresistor.
6. These values will be the constant values for the left and right, respectively (similar to the light intensity frequency program).

**Teachers:** Because you will have the students working on the calibration at the same time, it might be helpful to provide each student/student group with a small sample of the course. For instance cut out pieces of a poster board and have the students place some black electrical tape on the poster board for their calibration. It is important that the calibration is performed on as similar of a surface as the actual course so that no other influences are introduced during the calibration. To expand on this, if the students were to perform the calibration on a dark wooden desk and the course is on poster board then the Boe-Bot would not perform the operation properly, because it would be reading the light differently.

1. Write a program that will make the Boe-Bot follow the line on the course. The program should resemble the frequency due to light intensity program. For this activity however, the Boe-Bot is following the darkness instead of outputting a frequency due to light intesity.

Important Questions/Hints for the Program:

* Does the fact that the Boe-Bot is following the darkness instead of the brightness affect the inequality signs in the IF/THEN statements?
* How does it affect the direction the Boe-Bot turns when the light changes?
* Examine the two situations and find the differences between them?
* Reflect these differences in your program and the Boe-Bot will follow the line.