New York City College of Technology

Circuit Analysis Laboratory EET 2271 - OL51 Final Project

Phototransistor and PIR Motion Sensor Controlled LED

Nicholas Pillay, Amarjot Singh, Jason Alarcon Ali, Devesh Ramsingh

Professor Aale Naqvi

Due Date: 12/17/22

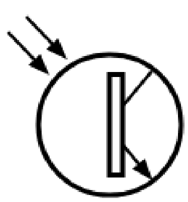
Table of Contents

Objective

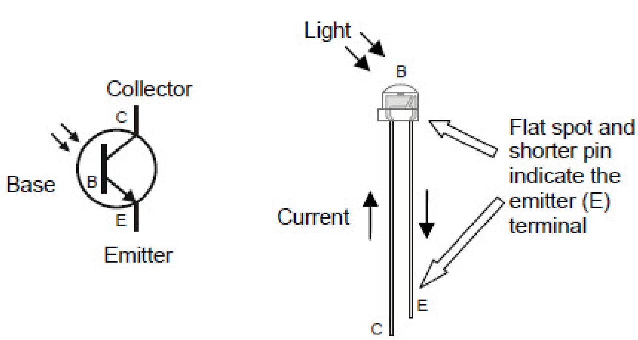
The goal of the project is to design a circuit where a LED is controlled either by a Phototransistor or a PIR Motion Sensor (Position 1 and 2 on Switch 2) when switch 1 is switched to on. And the LED will be off when switch 1 is switched off. So, the way the phototransistor should work is if it senses no light, then the LED should turn on and if the PIR sensor detects motion the LED should turn on. This circuit will be built using a Basic Stamp 2 Microcontroller and Parallax (Basic Stamp Software) for the coding.

Theory of Operation

For this experiment we will be working with two unfamiliar devices. The first is the phototransistor. This device will allow us to control our LED automatically based on if there is light hitting the phototransistor or not. Phototransistors are a type of a bipolar transistor that is sensitive to light. As we all know, all bipolar transistors are sensitive to light (That is why they are always covered by something like black paint), but what makes a phototransistor special is that these devices are enhanced to sense light more effectively. Specifically, phototransistors have a much larger base and collector area. Because phototransistors use a transistor, they have the advantage of being many more times sensitive to light than a photodiode. The way that a phototransistor works is it changes the current flowing between the emitter and collector according to how much light it is sensing. When the light hits the semiconductor, it then frees the electrons which in turn leads us to have more holes, and then this causes current to flow in the base region. Going into more detail, we understand that phototransistors are operated in their active mode, and the base connection will usually be left as an open circuit. Now the base of the phototransistor would only be used to bias the transistor where if any extra collector current was flowing, then this would cover any current flowing as a result of the light. For an NPN transistor, the collector is made positive with respect to the emitter and for a PNP transistor the collector is made negative. Below we can see the phototransistor circuit symbol (Using a NPN transistor).


[1]

And we can see a picture of how the phototransistor looks …



[2]

Some advantages to using a phototransistor are they have a high gain and this leads to them being more sensitive, and they can easily be used in an integrated circuit. Some disadvantages are that phototransistors cannot be exposed to high voltages like other semiconductor devices like photo-thyristors, and that they are not as fast as other light sensitive electronic components like photodiodes.

Some basic uses for a phototransistor include encoders where a rotating disc with light and dark stripes rotates, card readers, security systems, infrared detectors, and lighting control.

The next device we will be using is the PIR Motion Sensor. A PIR Motion sensor can also be called a "Passive Infrared", "Pyroelectric", or "IR motion" sensor. A PIR motion sensor basically allows you to sense motion. These devices are most often used to detect human movement in the sensors range. The way that a PIR motion sensor works is that it uses a pyroelectric sensor which detects different levels of infrared radiation. We know from prior knowledge that everything emits some level of radiation (most often very low), and the hotter something is, the more radiation it will emit. A pyroelectric sensor has parts made of crystalline material, which is material that generates electric charge when it senses infrared radiation. Depending on the amount of infrared that hits the material, a voltage is generated, that voltage is then measured by an amplifier in the circuit. It's also important to note that a PIR sensor uses a filter called a fresnel lens which focuses all the infrared signals onto this material. Then the last step is that the motion sensor will only indicate motion if the infrared signals are changing rapidly (It knows if the infrared signals are changing rapidly, thanks to the amplifier). Below we can see a picture of a PIR Motion Sensor.

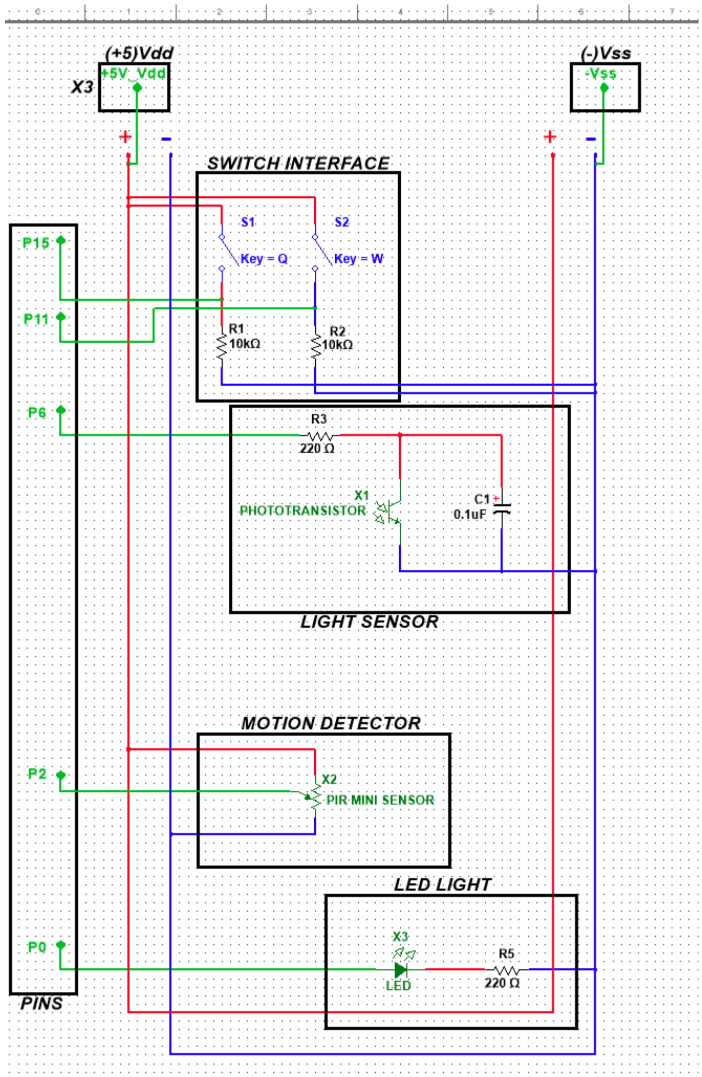

[3]

Preparation

The first step in this project was brainstorming the idea, so our group came together and settled on the idea of having two modes to turn on an LED, those modes being a daylight sensor and a motion sensor. Our interest in creating different modes using a switch was due to seeing how easy it is in Basic Stamp to test the state of a switch and jumping to another function using if-then statements. The idea for the different sensors came from a practical place, the most general use of a light is in homes and a useful application of mode switch would be for a location like a driveway or garage. With a light in that location only really being useful during times where you can not see (i.e. nighttime), it would be useful to turn it on in motion control mode when you are driving in or you can turn it to daylight sensor mode when you need to have it on for any other reason.

After deciding on the project, we had to decide on the parts to use to order them. We got the 850NM T1 ¾ Phototransistor for our daylight sensor and a PIR mini sensor for our motion sensor. We found light sensors online using this phototransistor so we decided that it was good to use for our project, and the PIR sensor worked for our 5V board and its properties of being an IR sensor made it have a good range and work at night as well, which is perfect for our application. The battery, resistors, switches, capacitor and LED were components we had from previous labs, so just to detail them we used: a 9V battery, 2x 220Ω resistors, 2x 10 kΩ resistors, 2x SPST switches, a 0.1 µF capacitor, and a green LED.

While waiting on the parts to arrive, we formulated the layout of the circuit and decided to have 4 separate small circuits that work together. The first is a circuit for the switches where they connect to the 5V and have a wire connected to a pin after each switch (P15 for the ON/OFF switch and P11 for the mode select switch) and there is a 10 kΩ current limiting resistor connected to ground at the end. The second is the daylight sensor circuit, where a wire from Pin 6 goes to a 220Ω resistor which connects to parallel branch of the phototransistor and capacitor both going to ground. The third circuit is a simple circuit for the motion sensor, where the Vin connects to the 5V, a ground pin connects to ground, and the output of the sensor goes to P2. The last circuit is just the LED circuit, so P0 connects to an LED which goes to a 220Ω current-limiting resistor and then ground. You can see the circuit below.



With the layout created, the only thing to do was to wait for the components to be delivered and write the code alongside the assembly process.

Manufacturing and Assembly Procedure

Once the components arrived, we implemented the circuit by interfacing the microcontroller with our breadboard so that we have enough space to comfortably create our project. There were no complications from doing this and the only thing was to connect the power lines to the 5V pin on the microcontroller and to do the same for the ground lines. After that was done we started to write the code in Basic stamp, and we needed to start with the ON/OFF switch proceeding to the mode select switch if the switch is on and being in an infinite loop of turning the LED off when the switch is off to keep the LED off if the switch was turned off while the LED was on. This was our InfiniteLoop function which can be seen below and it went to the Mode function, where it goes to the main daylight sensor mode when the switch is off and it and the motion control mode when the switch is on.

The daylight sensor function required the most care because we had to find out how to make it so that it will turn on the LED when there is no light, so we labeled it as our main function. We first charge the capacitor fully and then test to see the discharge time, which is done by setting P6 to high and pausing for 1 ms, then using RCTIME to measure how long it takes the capacitor to discharge. The 1 ms pause is more than enough because the time it takes for the capacitor to charge is which can be seen from the equation below.

$$t= 5*R*C= 5 * 220 * (0.1 *10^{-6}) = 110 \mu s$$

The discharge time using RCTIME is a bit involved so here is where we will explain. The RCTIME measures the time it takes for the capacitor to discharge, that is true, but that doesn't mean the time it takes to fully discharge. Basic Stamp sees the threshold for a logic value of 1.4V, so the RCTIME will go from the 5V of the fully charged capacitor and end when the capacitor voltage reaches 1.4V. This means that the time is

1.4= 5e^(-t/RC)

1.4 / 5 = e^(-t/RC)

ln(1.4 / 5) * RC = -t

t = -ln(1.4 / 5) * 220 * (0.1 *10^-6) = 2.8 * 10^-5 s

The units given to the result in our case is 2 µs because we are using a BS2 directive however, so that value we have to keep in mind in terms of the variable produced by Basic Stamp is t / 2 µs, which is 14. This would be when the phototransistor behaves as a completely short circuit, which it will not. When testing the code on the circuit and finding the value of the result from doing RCTIME, we were surprised to see how much the value changed from a small change in light. The value would go from 5000 with the light on in a room during daytime to 17000 with the light off. Since we want the LED to turn on at nighttime, we tried cupping our hand over the phototransistor to simulate it being night, and were surprised when the value returned to be 0. It turns out that Basic Stamp has an integer limit of 65536 (16-bits) for the RCTIME result value, so when we simulated nighttime we found that the result was over that limit and became 0. This meant that our circuit had to work in a region where that value would be 0, so we wrote the code so that if the result is over 0 it proceeds to the LED_Off function, otherwise it goes to the Light function.

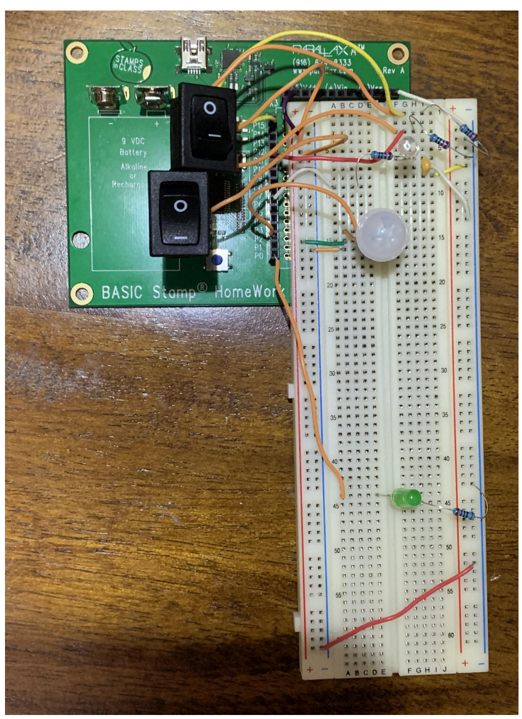
The MotionControl function was a simple function because all the sensor needs is Vin and ground and it outputs a voltage when it detects motion. This just requires the microcontroller to see a voltage greater than 1.4V from the sensor output and it proceeds to the light function, otherwise it goes to the LED_Off function.

The light function is just a few lines of code that turns the LED on for 5 seconds and then goes back to the InfiniteLoop function. The LED_Off function is similar in that it turns the LED off for 0.4 seconds and then goes back to the InfiniteLoop function.

The entire code we use can be seen in an image below.

```
' {$STAMP BS2}
' {$PBASIC 2.5}
Daylight CON 6
Led CON 0
result VAR Word

InfiniteLoop:
IF IN15=1 THEN Mode
LOW Led
GOTO InfiniteLoop

Mode:
IF IN11=1 THEN MotionControl
GOTO Main

Main:
HIGH Daylight
PAUSE 1
RCTIME Daylight,1,result
DEBUG? result
IF result > 0 THEN LED_Off
GOTO Light

MotionControl:
IF IN2=1 THEN Light
GOTO LED_Off

Light:
HIGH Led
PAUSE 5000
GOTO InfiniteLoop

LED_Off:
LOW Led
PAUSE 400
GOTO InfiniteLoop
```

With the code done, we ran it on our project and found that it worked as intended on both modes, with switching between the two modes and turning the main switch off working perfectly. You can see the build of the final project below.



Conclusion and Comment

Therefore, this experiment has taught us a lot about designing a LED controlled circuit using a Basic Stamp 2 Microcontroller and Parallax (Basic Stamp Software) for the coding. Two difficulties we ran into when performing this experiment were with determining the RCTIME result value being above the integer limit for nighttime and that for the motion control function we initially set the name "Motion" associated with pin 2, but checking for Motion being 1 did not work. We found out that Motion always equals 2 for the pin number, not 1 or 0 depending on the input signal from the sensor, which was fixed by just taking out the name and putting IN2 instead. Our progress with this experiment was that we successfully built a LED controlled circuit where the LED is controlled by either a phototransistor or a PIR sensor. We also learned the coding for using a phototransistor and a PIR sensor. The reason we know the circuit is working is because, when we placed the phototransistor in the dark and the phototransistor was set to control the LED, then the LED lit up. And we had the PIR sensor set to control the LED, and we applied motion to the PIR sensor, the LED again lit up. Overall, this experiment was a success.

Notes

Phototransistor information for theory of operation

- https://www.electronics-notes.com/articles/electronic_components/transistor/what-is-a-phototransistor-tutorial.php

Pic 1
https://www.electronics-notes.com/articles/electronic_components/transistor/what-is-a-phototransistor-tutorial.php

Pic 2
https://learn.parallax.com/tutorials/robot/shield-bot/robotics-board-education-shield-arduino/chapter-6-light-sensitive-15

PIR Motion Sensor information for theory of operation

- https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor
- https://cdn-learn.adafruit.com/assets/assets/000/010/136/original/PIRSensor-V1.2.pdf

Pic 3 https://www.parallax.com/product/pir-mini-sensor/

Reference for daylight sensor circuit

https://learn.parallax.com/tutorials/robot/activitybot/propeller-c-programming-activitybot/navigate-visible-light/build-light

Specifics on the RCTIME instruction

https://www.parallax.com/go/PBASICHelp/Content/LanguageTopics/Commands/RCTIME.htm

Bibliography

Ada, L. (2014, January 28). *Pir motion sensor*. Adafruit Learning System. Retrieved December 9, 2022, from https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor

Learn.parallax.com. Build the Light Sensor Circuits | LEARN.PARALLAX.COM. (n.d.). Retrieved December 9, 2022, from https://learn.parallax.com/tutorials/robot/activitybot/propeller-c-programming-activitybot/navigate-visible-light/build-light

Learn.parallax.com. Introducing the Phototransistor | LEARN.PARALLAX.COM. (n.d.). Retrieved December 9, 2022, from https://learn.parallax.com/tutorials/robot/shield-bot/robotics-board-education-shield-arduino/chapter-6-light-sensitive-15

Notes, E. (n.d.). *What is a phototransistor: Tutorial & primer*. Electronics Notes. Retrieved December 9, 2022, from https://www.electronics-notes.com/articles/electronic_components/transistor/what-is-a-phototransistor-tutorial.php

Parallax Inc. (n.d.). PIR Sensor (#555-28027). https://cdn-learn.adafruit.com/assets/assets/000/010/136/original/PIRSensor-V1.2.pdf

Pir Mini Sensor. Parallax. (2022, November 19). Retrieved December 9, 2022, from https://www.parallax.com/product/pir-mini-sensor/

RCTIME. (n.d.). Retrieved December 9, 2022, from https://www.parallax.com/go/PBASICHelp/Content/LanguageTopics/Commands/RCTIME.htm