Bits and Bytes

—

Andrew Snell

05/11/2021

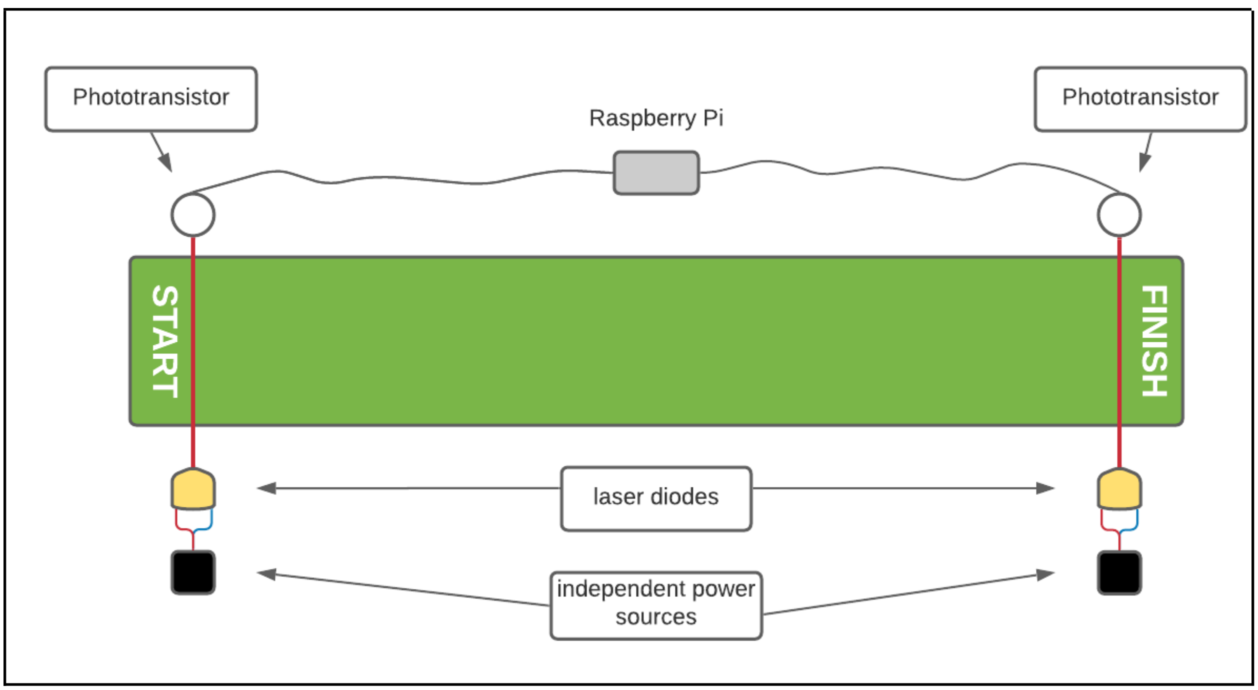
CS121 - B

Final Report

Modifications to Original Proposal

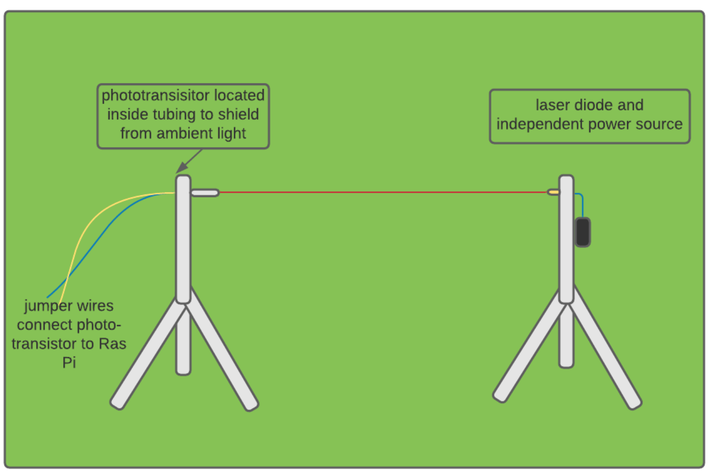
The General Set up

The final set up for the electronic timer was identical to the set up that was drafted in the original proposal. It featured two laser trip-wires - one at the start line and one at the finish line, forty yards apart. The trip-wires consisted of a laser diode, powered by a battery pack, directed at a phototransistor that connected to the Raspberry Pi via a breadboard and a system of jumper wires. The laser diode and phototransistor were mounted on separate structures approximately three feet apart, allowing the runner to pass between them freely. When the runner passed between the structures mounting the laser and phototransistors (located at the start/finish lines), the path of the was obstructed, decreasing the light intensity and triggering a response from the phototransistor to start or stop the timer. Jumper wires spanned from the start/finish line inward twenty yards, connecting the phototransistors to the RasPi in the middle of the track. The model below was included in the project proposal and remains an accurate representation of the final set up.



The Mounting System

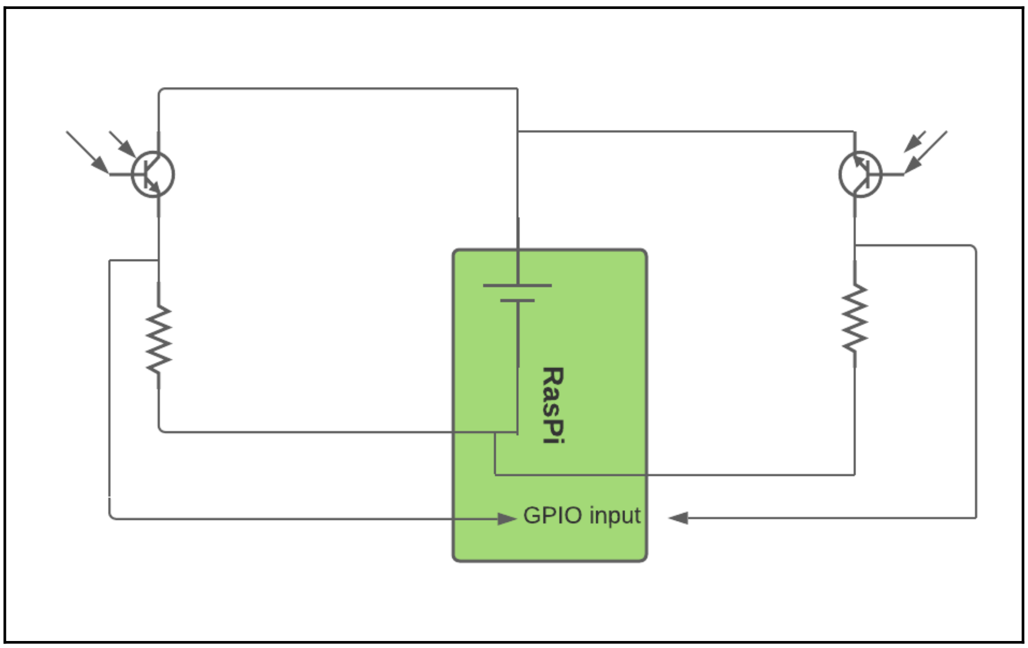
The mounting system featured in the final design was slightly different than originally proposed, but the concept remains the same. The mounting systems were similar in the fact that they secured the laser trip-wires in place, elevating them off the ground so they were approximately waist high (about 3 ft). Each laser diode and phototransistor was mounted on separate structures, so each structure held one component. Unlike the original proposal, the structures were constructed from a vertical steel shaft fastened to a heavy brick base. The steel shafts contained holes spanning the length of it from top to bottom. Because each structure featured an identical vertical shaft, it meant that the holes on one mount perfectly aligned with the holes on its counterpart. This allowed for easy alignment of the laser and phototransistor. As long as the laser diode and phototransistor were inserted into corresponding holes, they automatically aligned. As was mentioned in the project proposal, the phototransistors were placed several centimeters deep in tubing to prevent any ambient sunlight from affecting its readings. Below is the original design concept next to a photo from the final set up:



The Laser Trip-Wire

The laser trip-wire described in the project proposal worked to perfection in the final design. Phototransistors act as a valve, controlling the amount of current flowing through its circuit depending on the intensity of light it is exposed to. When the light intensity increases, the phototransistor allows more current to flow, and when light intensity decreases, it restricts the flow of current. It is the level of current in the circuit (determined by the light levels the phototransistor is exposed to) that the Ras Pi receives as input. When the laser diode is focused on the sensor, the current flowing through the circuit is nearly at its peak. The Ras Pi collects thousands of input values a second, allowing it to detect a break in the laser, even for a millisecond. This concept is the basis for our motion sensor timer. When the laser diode is focused on the phototransistor, an electrical current will flow through the circuit. When the subject impedes the path of the laser, a lack of light will cause the phototransistor to restrict the flow in the circuit. The RasPi will be able to detect this change of electrical flow via its GPIO pins, triggering the proper start/stop response for the timer.

The circuit containing the phototransistor draws its power from one of the 3.3V pins on the Raspberry Pi, and is grounded in a GND pin. The circuit, constructed on a breadboard, also includes a resistor and an analogue to digital converter (ADC). The ADC, which converts analogue signals to digital, is necessary for the GPIO pins to interpret input, as they can only understand digital signals. Below is an updated circuit diagram which features the model used in our final product design:



* model has been updated to reflect use of two phototransistors, as well receiving input before the resistor

Cost Analysis

Cost of Labor

Position	Hourly Wage	Hours Worked	Cost
Project Manager	$25.00	23.0	$575.00
Head of Software Development	$40.00	22.0	$880.00
Network Engineer	$18.00	15.5	$279.00
Lead Technician	$35.00	11.5	$402.50
Testing and Quality Assurance	$17.00	4.0	$68.00
		Total Cost	2,204.50

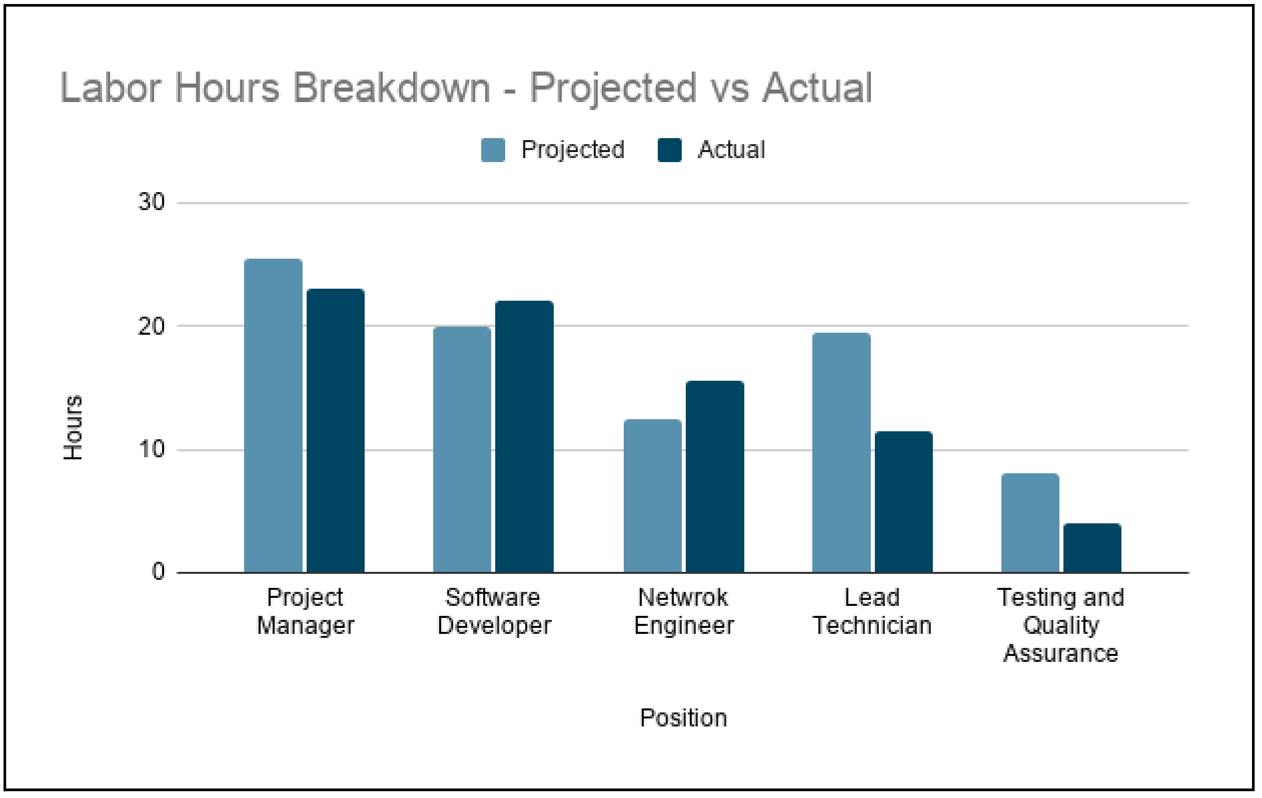
Cost of Materials

Materials	Count	Cost
Phototransistors	-	$4.99
MCP3008 Analogue to Digital Converter	1	$3.75
Jumper Wire	125 yds	$30.54
Breadboard	4	$5.99
Laser Diodes	-	$5.99
Battery Pack/Holder	2	$5.98
Double AA Batteries	24	$15.99
Mounting System	-	$42.59
	Total Cost	$115.82

Projected vs Actual Budget Costs:

	Projected	Actual
Labor	$2,481.00	$2,204.50

Materials	$83.68	$115.82
Total	$2,564.68	$2,320.32
Over/Under budget (+/-)		**-224.36**



Updated Gantt Chart and Team Contributions

The Bits and Bytes Team

Name	Role	Duty
Andrew Snell	Project Manager	Managed the due dates and expectations of the team, ensured proper materials were ordered, provide oversight in all fields, responsible for drafting proposals, reports, and logging time sheets
Andrew Snell	Head of Software Development	Wrote and tested programs to receive and interpret input data from sensors, implemented coded stopwatch to start/stop timing based on input received by GPIO pins
Andrew Snell	Network Engineer	Integrated user interface into product; worked with software developer to determine best approach; integrated software developer's code to be accessible via flask app, jquery
Andrew Snell	Lead Technician	Constructed proper circuitry and any physical systems in order for prototype to function properly, constructed mounting system, soldered necessary wires to ensure stability and longevity to product
Andrew Snell	Testing and Quality Assurance	Ensured prototype was functioning properly, diagnose and reported any and all problems, set fire to the UVM track with blazing speed

Gantt Chart

	March 26	April 1	April 8	Alrip 15	April 22	April 29	May 6	May 11
Publish Proposal								
Order Materials								
Test and Develop Software								
Test and Develop Circuitry								
Develop Working Prototype								
Construct Mounting System								
Assemble final design for testing								
Record Video								
Publish Final Report								

Reflection, Challenges, and Takeaways

Reflection

I really enjoyed the overall process of this final project. It felt really cool to design and construct something that started as an idea in my head. Every step along the way I knew it was closer and closer to being tangible. It was also fun pretending that I was a team, and that I had five different roles, each with different responsibilities. There were so many aspects to this project - coding, circuitry, engineering, construction - and to see them all align to form the product I had envisioned was super rewarding.

Challenges

There were several challenges throughout the process of drafting and implementing a design for the electric timer. The first was managing my time correctly in order to complete the project on time. There are a lot of steps to this final project, with equally as much headache and troubleshooting involved that I did not account for. That is why the current time is 10:56am, and this report is due in an hour. That being said, some of the hurdles I encountered while working on this project involved both coding and the circuitry. The first major challenge I faced was the implementation of the timer. My original plan was to display a running timer that updated every .01 seconds with a python GUI via the tkinter module. I had never used tkinter before so I was already struggling to design the program I had envisioned. Once I finally got a live timer to display, my next problem was controlling the timer with inputs from the Ras Pi. After doing some research, I discovered I would most likely need to use threading in order to circumvent my problem. Furthermore, I couldn't even display my GUI on the Raspberry Pi. Although it displayed when I ran the timer program on my computer, my Ras Pi was not configured to display the GUI and I was not prepared to go down another stackoverflow rabbit hole.

Another challenge I faced was soldering the laser diodes to the battery pack, as well as soldering jumper wires together. This was my first time ever soldering, so it came with a learning curve. The wires on the laser diodes were really small (30 awg), which made it really hard to solder a good connection without stripping the wire or leaving frayed ends.

The last problem I faced was during the testing of the final design. Up until this point, all the testing had taken place indoors - where light intensity is much less. After setting up outside for the first time, I could not figure out why the laser trip-wires were not triggering a response from the Ras Pi. I knew the lasers were focused on the phototransistors, and that the phototransistors detected changes in light. While the lasers were focused on the sensors, they had outputs of about 1000, and when the lasers were obstructed, the sensors had readings of about 400. Obstruction of the laser was clearly causing a decrease in the current of the circuit, yet it still wasn't triggering a start/stop response from the Ras Pi. It took me several minutes to determine that the issue was that the current in the circuit never dipped below the threshold required to trigger the start/stop response. While I was writing the code for this project, I set the threshold for the timer response arbitrarily at 250. However, now that I was outside, ambient light from the sun kept the input value from the sensors above 250, meaning the timer would never start/stop. So I changed the threshold to 450, fixing the problem.

Takeaways

My first takeaway is to never underestimate how long it takes to code something. It never works out as you plan, so you better have a plan B or even plan C. Even when it does finally work, you will continue to tinker with it for hours. My second takeaway is to value the process of making a schedule and following it the best you can. Although my timetable didn't follow the gantt chart in my proposal, I still tried to meet the deadlines I had set for myself. Without it, I would be even more rushed than I currently am.

References

Instructables. (2017, November 8). *Another Arduino Laser Tripwire*. https://www.instructables.com/Another-Arduino-Laser-Tripwire/

Raspberry Pi Analog to Digital Converters. (2016, February 9). Adafruit Learning System. https://learn.adafruit.com/raspberry-pi-analog-to-digital-converters/mcp3008

Lee. (n.d.). *JumpOff Web Development*. JumpOff. https://jumpoff.io/blog/laser-tripwire-experiment-phototransistor

What is a Phototransistor » Electronics Notes. (n.d.). Electronics Notes. https://www.electronics-notes.com/articles/electronic_components/transistor/what-is-a-phototransistor-tutorial.php#:%7E:text=Phototransistors%20are%20a%20form%20of,much%20more%20sensitive%20than%20photodiodes.

GPIO - Raspberry Pi Documentation. (n.d.). Raspberry Pi. Retrieved March 26, 2021, from https://www.raspberrypi.org/documentation/hardware/raspberrypi/gpio/README.md