Julia Schell
Eric Tan
EE 3 Spring 2016

Skittle Sorting Machine

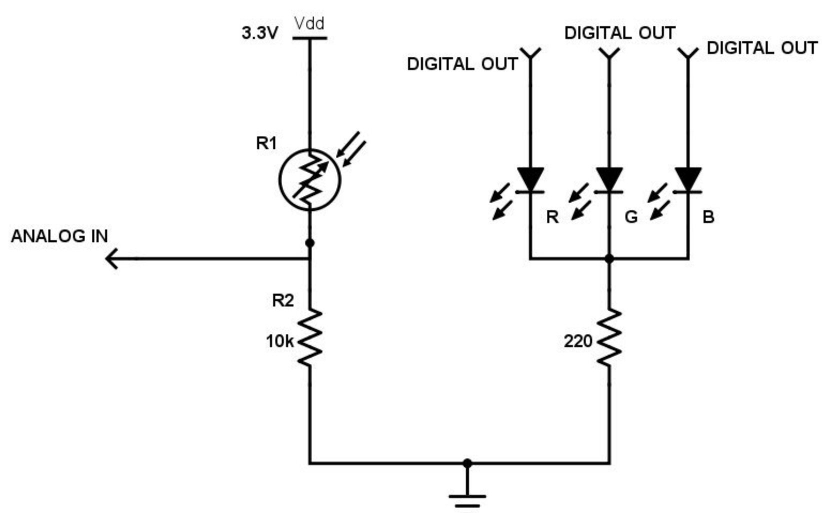
Introduction and Background

For our project, we built a machine that automatically sorts individual Skittles candy pieces by color. The goals of our project were to implement:

- Color detection sensor: determine the color of an object based on the concept of light reflection. We did this by reflecting light from an RGB LED to a photoresistor and measuring how much light hit the photoresistor.
- Isolation of a single object: be able to put in a cluster of objects and filter by individual objects. Our design for this part was a funnel that led to a tube about the circumference of a Skittle.
- Responsive catching mechanism: automate a way to group similar objects in their designated categories based on the color detected. We programmed two servo motors to achieve this, one to push the candy down a ramp and another to turn a turntable to the appropriate section that the candy should go.

Our project makes use of the voltage divider equation to measure readings from our sensor and pulse width modulation to control the servo motors. We programmed in Arduino with a Teensy 3.1 microcontroller.

Testing Methodology



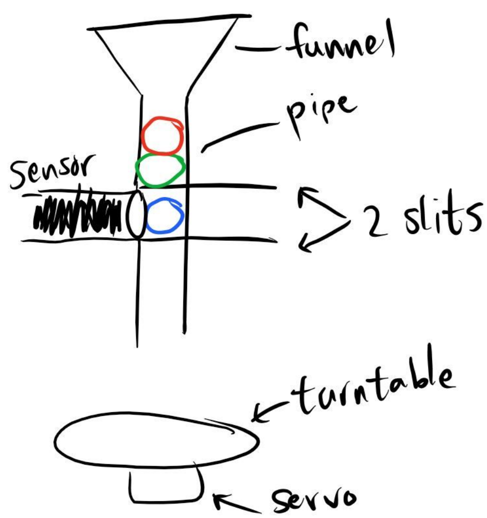
[1]

First, we had to figure out a way to measure how much light was shining on our photoresistor. The voltage divider equation states that:

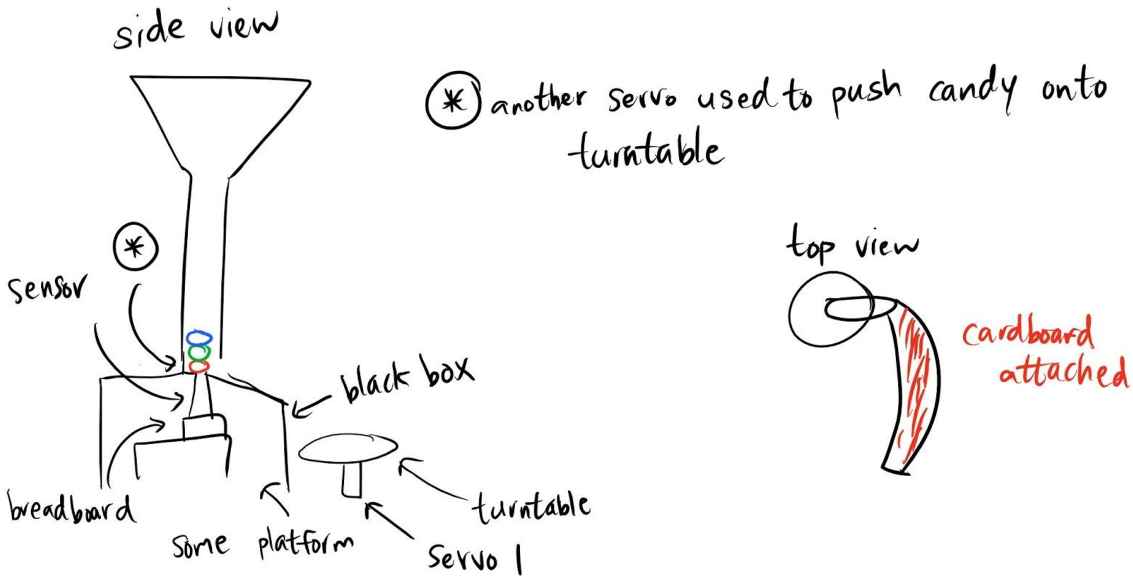
$$V\,out = V\,in * \frac{R2}{R1+R2}$$

According to the schematic diagram, Vout is the voltage measured by the analog input pin of the microcontroller, Vin is Vdd, R1 is the resistance of the photoresistor, and R2 is the 10k resistor. We measured the voltage across the 10k resistor instead of the photoresistor because we wanted a higher reading to correspond to more light shined on the photoresistor. A photoresistor decreases its resistance as more light is shined on it, so Vout will increase as R1 decreases. We chose R2 to be 10k ohms because in a dark environment, R1 is greater than 100k ohms, so Vout will be relatively small compared to Vin, whereas in a brighter environment R1 ranges from 2k to 20k ohms, so Vout will be on a more comparable scale relative to Vin.

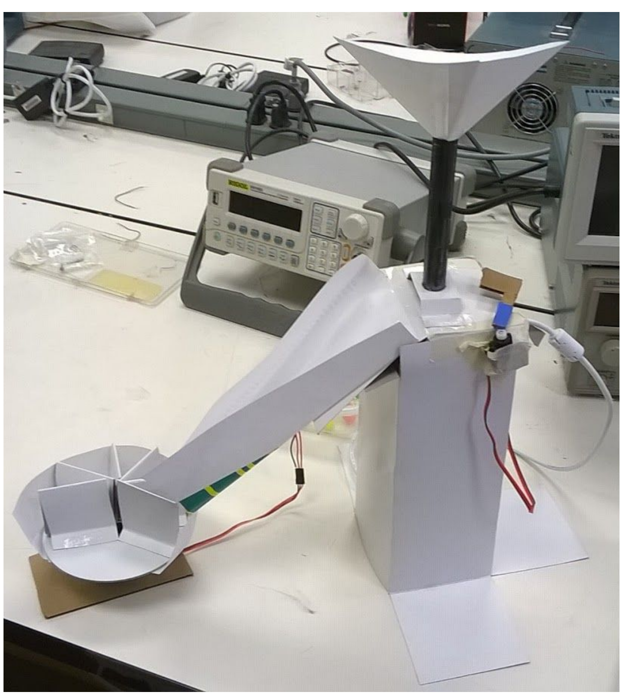
Next, we had to think of how to build the actual project apparatus, which we decided to make out of paper and cardboard. The responsive catching mechanism would be implemented with a servo-controlled turntable. To do the isolation of single objects, we initially thought of a servo-controlled double slit system in which a single Skittle would be filtered in the area between the two slits, as shown in the initial design below:



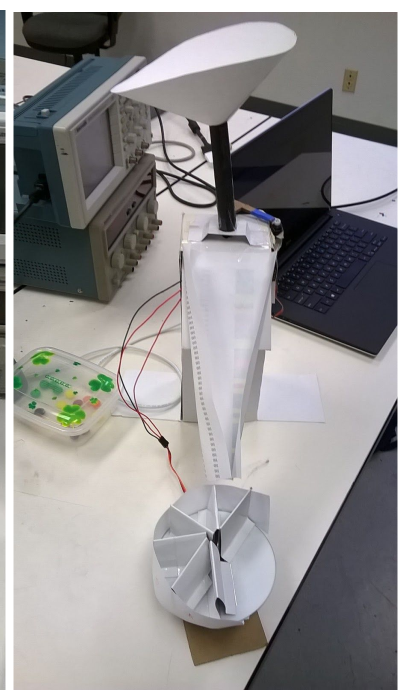
However, we soon realized that there were glaring issues with this design. First, it would be hard to use the servo to precisely control the pulling out and pushing in of the slits because of friction force. Also, the sensor readings would be difficult to interpret. Our sensor detection works for an object at a fixed position, so if the sensor moves a little then it could give a very different reading than the expected values. After some trial and error, below is what our final design came out to be:



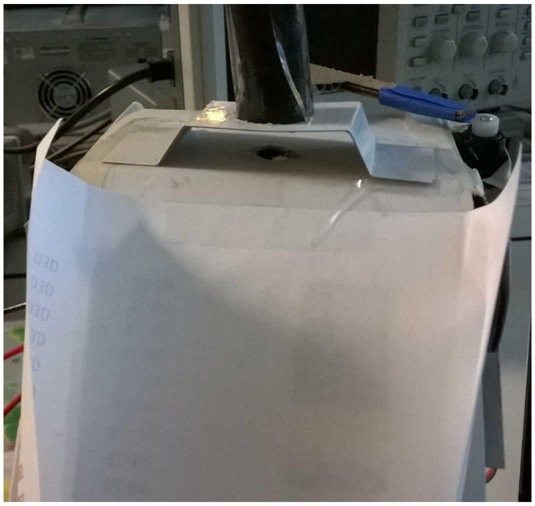
In the final design, one Skittle would fall onto a hole (that has a size a little smaller than the Skittle) that would be aligned with the black tube of the sensor. This fixes the moving sensor problem, as the object would land and stay still on the hole during the color detection process. Also, this gets rid of the need to use slits to filter out single pieces, as another servo with a cardboard arm would push the bottom piece of candy down a ramp into the turntable, with the next Skittle naturally falling down to the hole for the next color detection process. Below are some pictures of our apparatus.



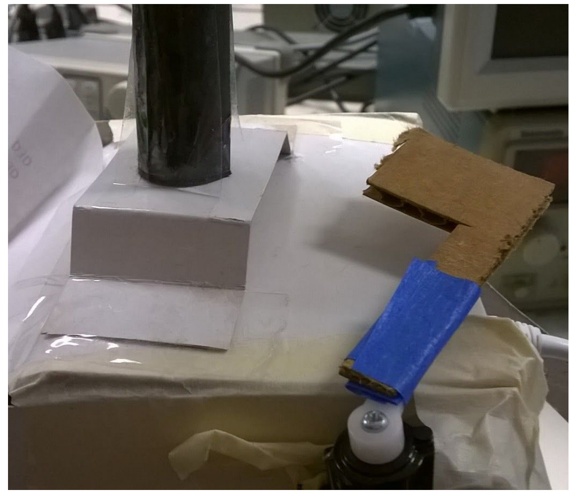
Side view



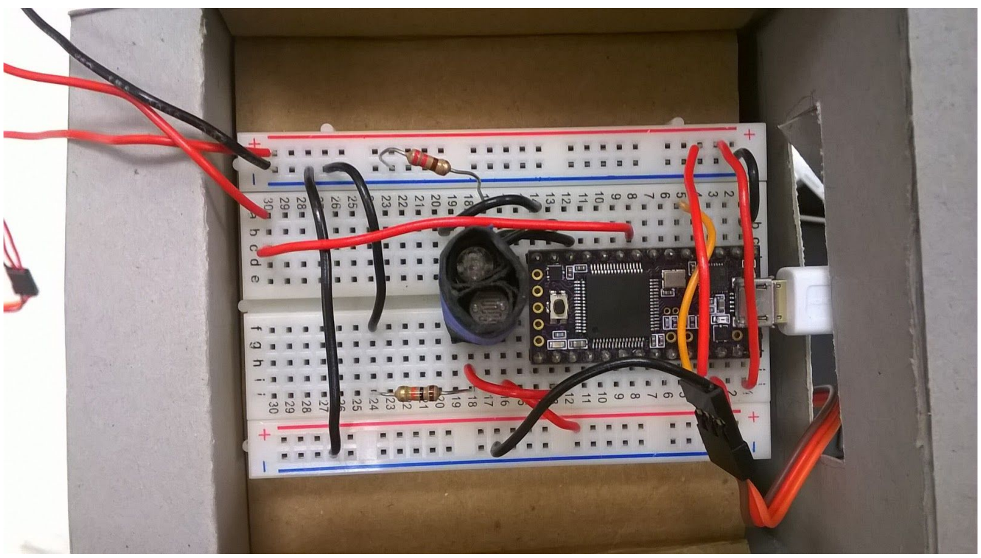
Front view



Color detection happens under the bridge



Servo's cardboard arm pushes candy down ramp



Our circuit for color detection controlling the servos. To block as much ambient light as possible, the LED and photoresistor are each wrapped with black paper, and the whole thing is wrapped with black paper again.

Results and Discussion

Our sensor measures the amount of light reflected from the LED light off a Skittle back to a photoresistor, so we had to measure the expected values for each Skittle color. The Arduino analogRead function maps the voltage range of 0 to Vmax to a number between 0 and 1023 , and the table below shows the average values read by our analog input pin for each color, done in a medium well-lit room. The titles in the first row are the colors of each Skittle, and the titles in the first column are the LED lights that were shined at the Skittle.

	Red Skittle	Orange Skittle	Yellow Skittle	Green Skittle	Purple Skittle
Red LED	440	542	520	288	252
Green LED	140	184	255	191	99
Blue LED	133	177	188	142	101

During the operation of the device, the color of a Skittle would be determined through similarity matching. The color chosen is the one with the set of expected values having the least total errors compared to the values (from R, G, B LED's) read by the sensor.

However, our sensor was still sensitive to ambient light, even with the black paper around it. Therefore, we couldn't hardcode these values because if we moved the apparatus to another environment with a different amount of ambient light, or even if the sensor slightly moved position, then we would have to manually change the numbers in our code in order for the right colors to be detected in that environment. For example, if we moved our apparatus to a brighter environment, then the range of values could be shifted by an increase of 40. Thus, a Skittle that is detected as orange in one environment might be detected as yellow in another.

To avoid the need to change the hardcoded values for each new environment, we instead came up with a calibration method for the software by using Arduino's EEPROM library. We would still need to put each Skittle color on top of the sensor for this calibration, but the expected values read by the microcontroller would be stored into the microcontroller's memory storage (like a tiny hard drive). When we interpret sensor readings for Skittles during the actual operation of the apparatus, the microcontroller can compare those values with the expected values that were stored in the microcontroller's memory storage during calibration.

For the turntable control, we spaced the 5 bins equally around a semicircle (this meant there was a 45 deg angle between each bin). Using the PWM pins on our microcontroller, we controlled the servos using pulse-width modulation. The length of the pulse determines what direction and how far the motor turns, and it looks for a new pulse every 20ms -- a standard repetition rate. Once the color of a skittle is determined, a neutral pulse is sent and the servo returns to center. Then, depending on the color, a pulse is sent to turn the servo to the current color's bin, as can be seen below.

	Red	Orange	Yellow	Green	Purple
Turn (deg)	0	45	90	135	180
Pulse (ms)	1	1.25	1.5	1.75	2

Conclusions and Future Work

Overall, our design met each of the three goals outlined above. The color sensor occasionally misreads the color of the skittle due to excess ambient light, but the calibration method significantly reduced the error frequency. Similarly, the skittle-isolation mechanism can get stuck as a result of a candy falling at an angle. Consequently, our next goal for this project would be to increase the reliability of the skittle separator. This would correspond to adapting the machine for other candy types, as both goals could be achieved by separating the candies independent of height and size -- the color sensor can already be set for candies of different colors. In addition, we would like to increase the speed and efficiency of the machine as a whole. Currently, one skittle is evaluated for color, the turntable is turned twice, and then the skittle is released and the process begins again. Ideally, the machine could look at the color of one candy while placing the previous candy in a bin, and could turn quickly and directly from the previous color's bin to the current bin.

The original focus of our project was to gain a deeper understand of variable resistors and LEDs. The process of building our color sensor and adapting it over time to fit new needs was extremely informative and allowed us to thoroughly explore the capabilities and inefficiencies of of these components. Additionally, we used two servos in our project for two very different functions. While implementing the servos, we gained hands-on experience with pulse-width modulation. Lastly, in a more general sense, the experience gained from designing, failing, and redesigning is extremely valuable and left us with a sense of confidence in applying engineering theory to real world problem solving.

References

[1] Schematic drawn using the Scheme-it drawing tool from Digi-Key.