1. Introduction

a. Statement of Purpose

We aim to give people an interactive solution to control their water intake. We create a lid that measures water level within a bottle to monitor people's hydration level throughout the day. Our system has an advantage in that our whole system will be on the lid, and therefore will be easy to make multiple cases compatible for different types of water bottles (even if we don't make the actual bottles).

b. Features and Benefits

Our current water bottle cap design can tell how much water is in the water bottle and makes recommendation on whether or not you should drink water by detecting the water level inside the bottle. If you choose to refill your water bottle you can simply hit the reset button to detect the new level of water.

The importance of hydrating yourself throughout the day is often overlooked by college students as some don’t have water pitcher in their dorm room. But even beyond college campuses, dehydration is a big problem. 75 percent of Americans suffer from chronic dehydration. The hydration monitor helps you take in 8 ounces of water 8 times throughout the day. [2]

Although there is a Kickstarter project called Gululu that targets kids' market, it contains lots of extraneous features, which only drains battery. We are trying to make a simpler solution to allow durability while not losing its value. [1]

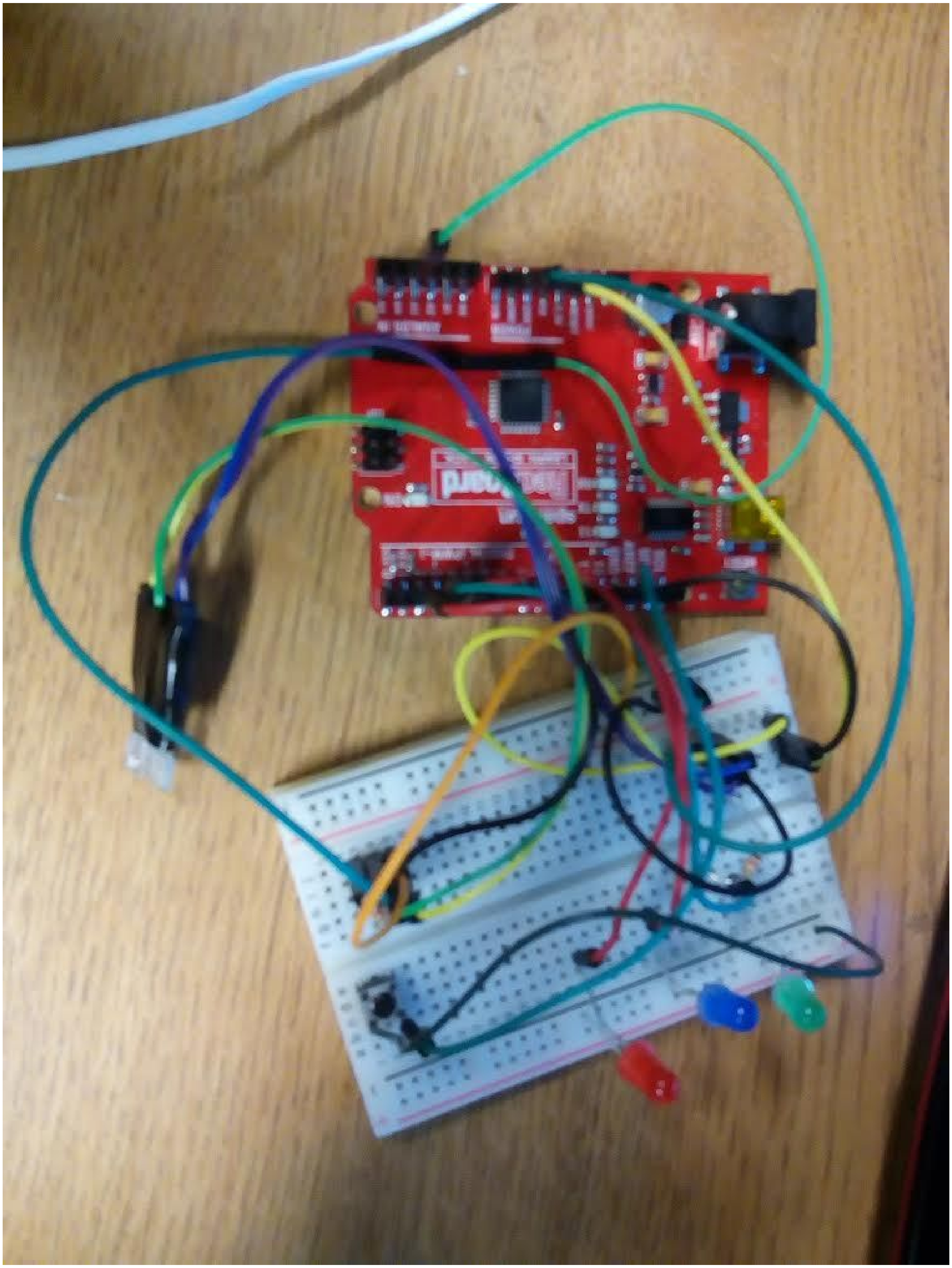
2. Design

The following page depicts our device. As you can see we used a breadboard to wire the hydration sensor. The breadboard allowed us to actively experiment with the different electrical configurations in order to find what worked best.

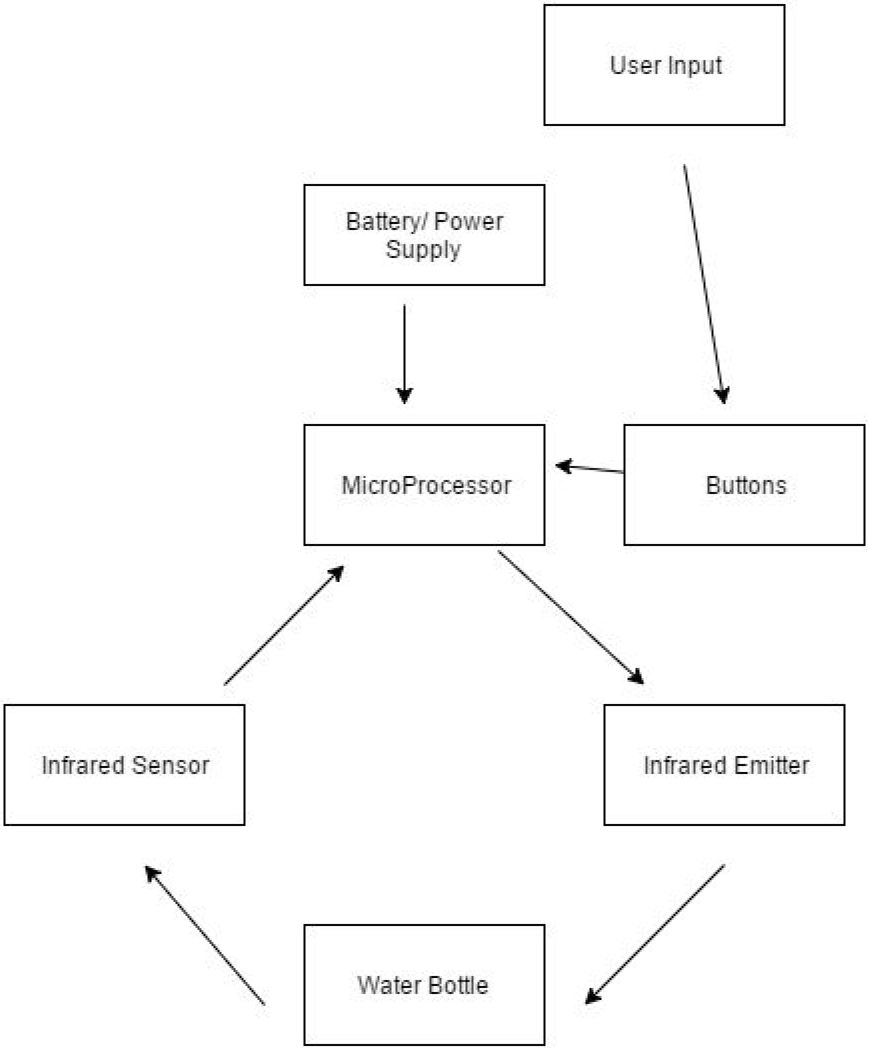
The jumper cables are a useful tool that allowed us to easily change where the wires were connected on the bread board. The colors can also be helpful in organization. By attaching a male to female jumper to a male to male you could extend the length of a jumper cable greatly which often helped when characterizing the sensors at different distances.

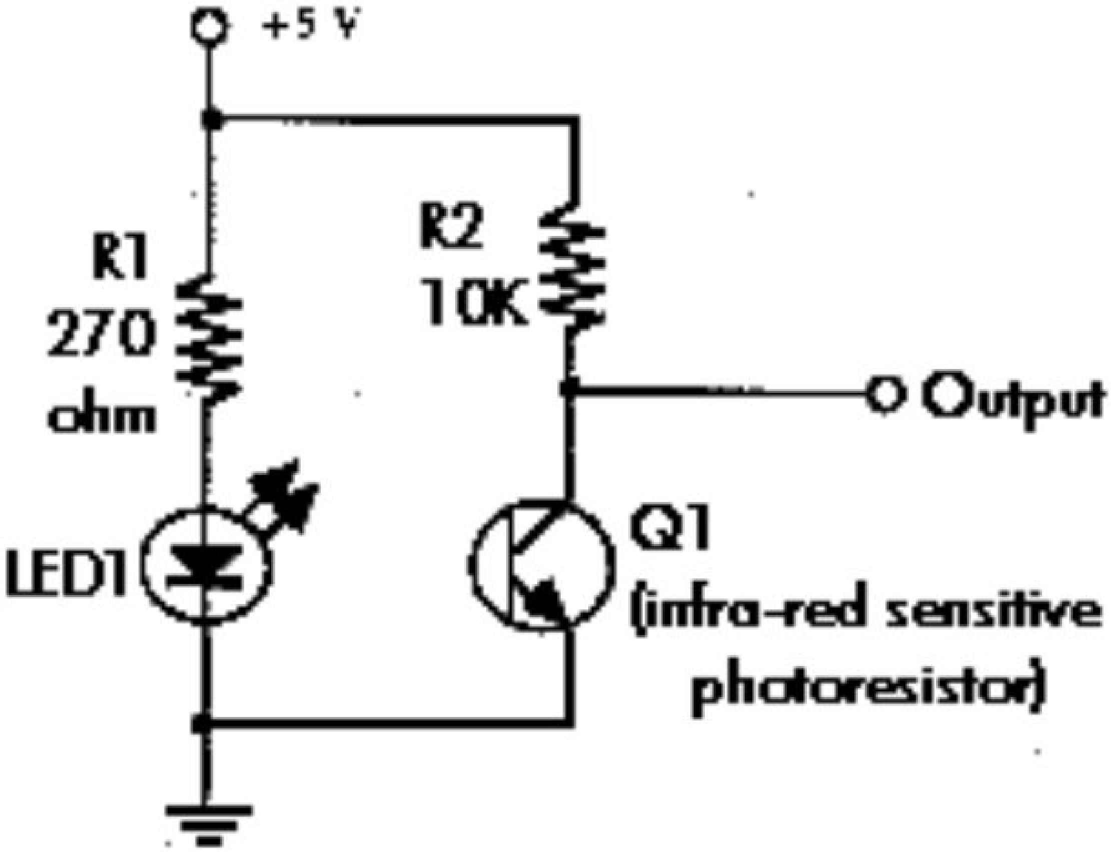
The LEDs and resistors did not always fit properly in the bread board. The leads can be trimmed on either to make a lower connection that is less likely to fall out. The resistor leads can also be bent in on themselves to provide a better connection. The button leads must be bent slightly inwards before they can be inserted into the breadboard.

The Red Board was not our first choice. The Arduino mini did not ever properly download our code even with the proper configurations. The Red Board was easily insertable in the minis place. Although it is larger and more bulky, it worked for the level of experimentation we completed. Ideally we would put this on all onto the small chip in the middle of the arduino.



a. System Overview


User Input
Battery/ Power Supply
MicroProcessor
Buttons
Infrared Sensor
Infrared Emitter
Water Bottle



[3]

b. Design Details

Infrared emitter emits infrared light while the receiver changes the voltage depending on how much IR light is received. In the water bottle, the emitter emits the IR light, which will bounce back from the top surface of the water back to the IR receiver that is located next to the emitter. This setup allows us to measure the distance of the water from the inside of the cap. In our experiments we found that we could detect reflections up to around 20 centimeters.

There are three LEDs with different colors: red, blue, and green. Each of them are connected to a digital pin on the Arduino. The negative side of the three LEDs are all connected to a single button before going to ground. This is to allow the LEDs to be off when they’re not needed, but to be turned on anytime with a single button without having to know which LED the Arduino is powering. (Only one LED will be on at a time).

The microprocessor converts the voltage into 4-bit data, ranging from 0000 to 1023 depending on the voltage. The higher the voltage is, the higher the number will be. The voltage is at its highest when it is exposed to the greatest amount of Infrared light. This occurs when the water level is at its highest. It will take an initial reading upon reset. Based on the voltage reading that it receives after a given time increment, it decides whether there is too much/too little water inside the water bottle.

The Arduino takes 7-12V through the DC power jack and 5V through the USB cable. During the lab, we mostly used the USB cable, but during the demonstration, we connected it to a 9V battery through the DC power jack.

Code appendix

```
int receiverAnalogPin = 3; // IR sensor receiver
int emitterDigitalPin = 5;
int val = 0; //store the value read
int value = 0;//difference in values
int redLED = 2;
int greenLED = 3;
int blueLED = 4;\
int initialvalue;
void setup() {
  // put your setup code here, to run once:
  pinMode(redLED, OUTPUT);
  pinMode(greenLED, OUTPUT);
  pinMode(blueLED, OUTPUT);
  pinMode(emitterDigitalPin, OUTPUT);
  initialvalue = analogRead(receiverAnalogPin);
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  // calibrate to the bottom/top water level
  // turn on emitter
  digitalWrite(emitterDigitalPin, HIGH);
  // read the adc data
  val = analogRead(receiverAnalogPin);
  value = val - initialvalue;
  Serial.println(val);
  Serial.println(value);
  delay(600);
  // if the voltage is in certain interval, turn on water level
  // TESTING: blink LEDs
  if (value < 150)
  {
    digitalWrite(redLED, 2.5);
    digitalWrite(redLED, LOW);
  }
  if (150 <= value && value < 300) {
    digitalWrite(greenLED, HIGH);
    digitalWrite(greenLED, LOW);
  }
  if (300 <= value) {
    digitalWrite(blueLED, 3.5);
    digitalWrite(blueLED, LOW);
  }
  /*
   * int amount = 0;
   * 1. Set up with initial amount at the beginning of the day
   * 2. If there is a decrease in water level, add the volume to the amount
   * Decrease in water level indicated by change in water level
   * The change in voltage should be negative
   * In a second loop, check water level every two hours 8 oz is the ideal amount.
   */
}
```

3. Results

The final experiments were done with the IR detector and emitter attached together. The water bottle that we were using carried approximately 16 ounces and is around 20 cm tall This means that every two hours the sensor must detect half of the water level disappear. We filled the water bottle entirely and recorded voltage reading of around 700-800 out of 1023. After emptying half we received reading from 500-600 and when almost empty 300-400.

After these findings we programmed the lights to read the initial value which we knew was around 800. After the given delay(which we kept minimal for experimental purposes) it detected the difference in water level. Being that from full to empty is 800 to 400, a change in 200 would correspond to a water decrease in approximately 8 ounces. Knowing this if the the difference was between 150 and 250 the light would flash green, less would flash red, and more would flash blue.

We repeated the original experiment with the programming now tailored to the sensor's characterization and found that for the most part that it would flash the correct color after the given time. Our delay was only several milliseconds instead of 2 hours in consideration of our time and for the demonstration.

The LEDs are simply activated with the logic statement previously stated. The arduino then sets the corresponding pin to high based off the logic. The secondary switch exists so that the lights are not constantly flashing. The sensor will pick up the reading and when you press the switch you can then see what data it has gathered and what it recommends you do next.

4. Problems and Challenges

The IR sensor that we used was fairly unreliable. Testing the IR sensor in our laboratory setting was challenging, because the IR sensor picked up signals from the ambient light. When isolating the system away from light sources readings were more successful. Ambient reflections also caused a problem, so readings were more consistent when the reflective surface was not water. Water would prove to be more reliable if contained in an opaque container.

We tried using an optical reflection sensor that was part of the ECE 110 Kit. However, the sensor's detection range was too small (less than 8cm) so we decided to stick with our original plan of using a separate IR receiver/emitter.

It took us a considerable amount of type figuring out that the Arduino mini board wasn't executing the code properly. We found out by probing the voltage throughout the circuit until we realized that the LEDs aren't getting the right amount of voltage. We decided to abandon the mini board and go with the Sparkfun board.

5. Future Plans

The current functions of our water bottle cap can be done with our naked eyes by opening the water bottle. The crucial next step in terms of capabilities would be to find a way to measure how much water is being drinken throughout the day. The Arduino doesn't have a permanent storage space, so we'll need a microSD card to save the data, and a computer program that can calculate the water drinken based solely on water level.

We plan to achieve by using the following algorithm:

If the water level rises after the bottle is capped, the total amount of water drinken doesn't change (since the water bottle is being refilled). We save the new water level as the reference point when the water level changes next time.

If the water level decreases after the bottle is capped, we can find the volume of water drinken using the height difference of the water. This can be different depending on the water bottle used, but it can easily be tailored with quick measurements.

We plan to ensure that the person drinks 8 ounces of water every two hours. We'll turn the blue LED on when the person has had more than 8 ounces of water in the past two hours. We'll turn the red LED on when the person has had less than 8 ounces of water in the past two hours. We'll turn on the green LED if the person is drinking water in the right pace.

6. References

1 "Gululu | the interactive bottle that keeps kids hydrated", *Kickstarter*, 2016. [Online]. Available: https://www.kickstarter.com/projects/gululu/gululu-the-interactive-bottle-that-keeps-kids-hydr/description. [Accessed: 22- Sep- 2016].

2. Ericson, "75% of Americans May Suffer From Chronic Dehydration, According to Doctors", *Medical Daily*, 2016. [Online]. Available: http://www.medicaldaily.com/75-americans-may-suffer-chronic-dehydration-according-doctors-247393. [Accessed: 12- Dec- 2016].

3. A. Wilson, "Schematics Depot (TM) - basic ir detector/emitter", *Reconnsworld.com*, 2016. [Online]. Available: http://www.reconnsworld.com/ir_ultrasonic_basicirdetectemit.html. [Accessed: 12- Dec- 2016].