3SENSE\*

\*Christmas Edition

Arduino Motion Sensor Audio LED Visualizer

Valencia Kaplinsky & Tyler Lynch

[vakaplin@buffalo.edu](mailto:vakaplin@buffalo.edu) | [tjlynch@buffalo.edu](mailto:tjlynch@buffalo.edu)

University at Buffalo: CSE 321 Fall 2016

Executive Summary

This prototype uses a motion sensor that reacts to movement and triggers a song to be played from a PC, and is accompanied by LED lights that illuminate in accordance to the kick, snare, and hat of the played song. When the device doesn’t detect motion for 6 seconds (or any assigned time period), the song and LEDs are paused until motion is further detected. This can be useful for a number of leisure activities, and could save energy in situations where music only needs to be played for short periods of time (elevators, waiting rooms, etc.) In future models, implementation of multiple motion sensors will be used to add next song/previous song/ pause/ play functionality. This could expand the usability of the device, and be useful around the house or in the kitchen. With the widespread normalization of wire free headphones, hands-free control over music will follow suit.

Project Objectives

**Required materials:**

Hardware:

* Arduino (Uno)
* 4 x Red LEDs
* 8 x White LEDs
* 4 x Green LEDs
* 6 resistors
* solderless breadboard
* Wires
* PIR Motion Sensor (Infrared)
* USB (A to B) cable
* Computer

Software:

* Arduino SE
* Processing SE
* Firmata library for Arduino
* Arduino library for Processing
* BeatWrite/BeatListener code from minim JavaSound Library examples

3Sense project will meet the following objectives:

* Play audio once motion sensor is triggered
* Pause audio track once no motion is detected after an allotted period of time
* Provide an audio synced LED light visualization

Project Approach

There were two general aspects to this project, one was linking the motion sensor to the audio output, and linking the audio to the LED lights. We must first ensure the proper operation of those individual components. Once the proper workings of individual components are tested, we will combine the two. We will add further extensions once the base prototype is working without any bugs.

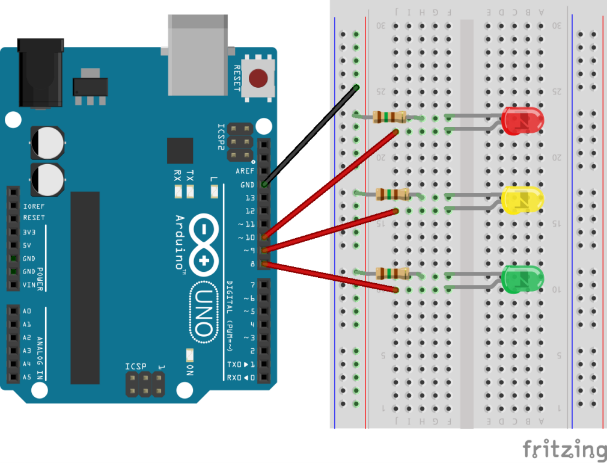
Project Description

3Sense is a localized product that has a large client base which ranges from business to home users. On the business aspect its energy conservation functionality creates great purpose for certain structures such as elevators-playing music only when someone is in the elevator, and pausing when vacant- or waiting rooms, the options are endless. It can also be used at home either at a desk with or without computer use. If a computer is being used, the user wouldn’t have to close out of current window or bring up the music player window to skip/pause/replay/etc a song. Instead he/she could simply swipe his/her hand over the motion sensor designated for the specific function, no hassle. It can be used in other rooms around the home such as the kitchen for example when the user’s hands are dirty and they cannot physically change a song. And the energy conservation comes into play here too.

Design Details

User: movement/motion/infrared





Senses motion

Lights up LEDs

Processing SE:

Motion Sensed:

Play song

No Motion Sensed after x time:

Pause song







User’s Manual

1. Select the MP3 file you’d like to play and enter it into the Processing program.
2. Execute the program
3. Create motion (wave your hand) in front of the PIR sensor, and the song will begin followed by in sync LED illumination.
4. To pause the music and LEDs, simply stop all motion in front of the PIR sensor for 6 seconds.
5. Once the music and LEDs have been paused, make motion in front of the PIR sensor again to start the song from the paused point.
6. Once song has completed, start again from step 1.

Programmer’s Manual

\*Refer to Project Objectives for required materials

**Build the Circuit:**

Motion Sensor to Arduino:

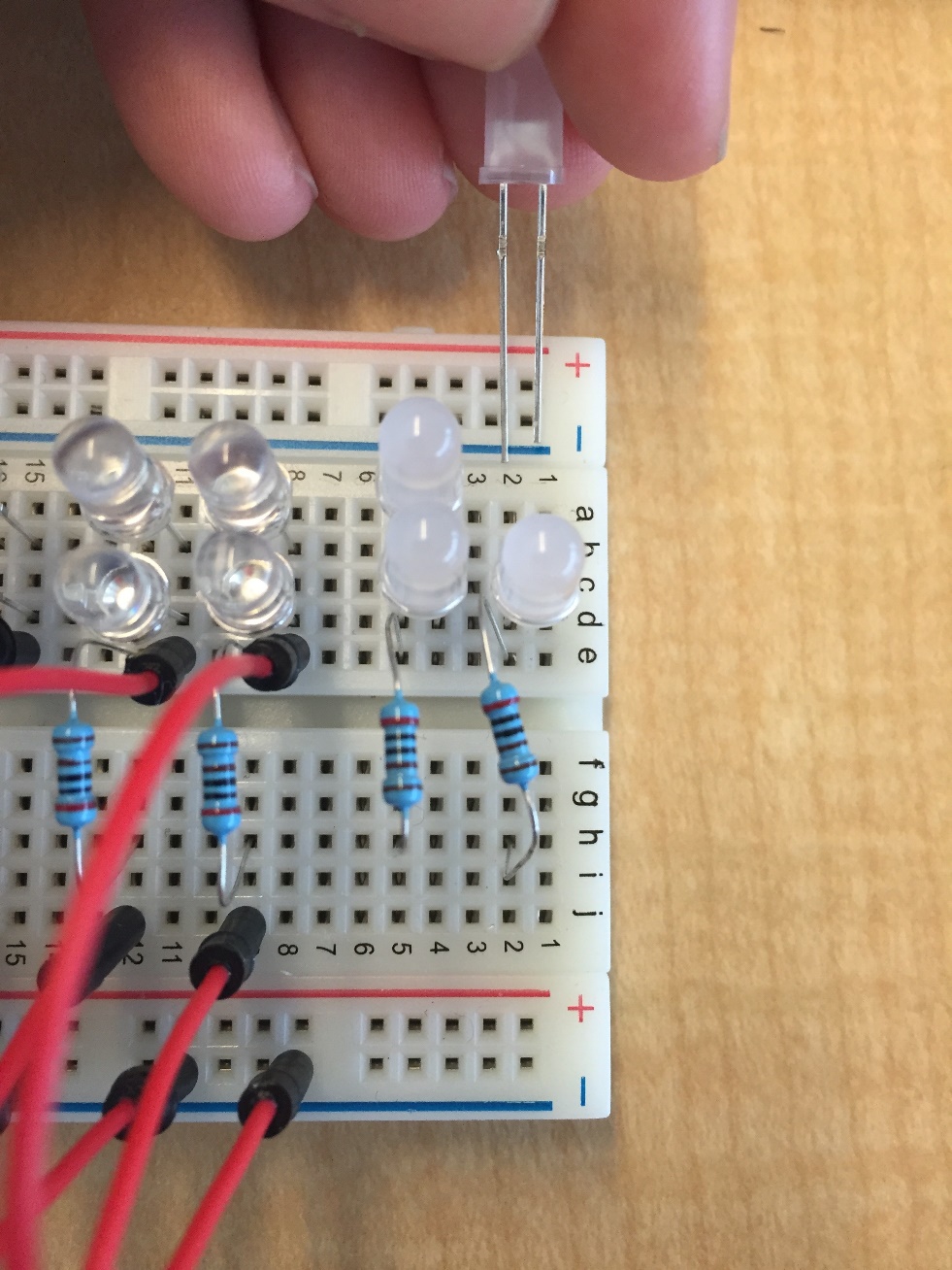
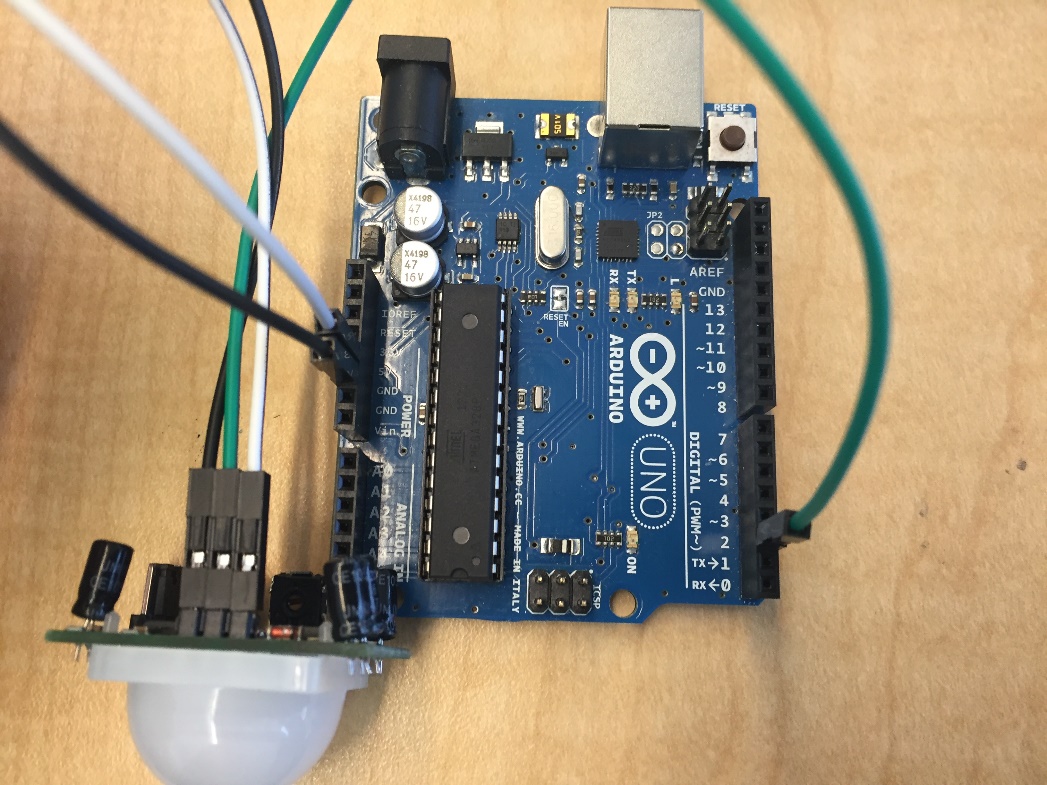
1. Run wires from Motion Sensor GND port to Arduino GND port, Output SRC to digital pin 2 in Arduino, and +5V to 5V in Arduino (Image 1)

LEDs to Arduino:

1. Run a wire from the negative outer rail of the breadboard to the digital GND port of the Arduino.
2. Place LED lights of the same color in groupings of 4 evenly across the breadboard. Make sure the anode (the longer end) is on the left. (Image 2)
3. Place your one resistor next to the LED anode for every two LEDs on the inner rails of the breadboard. Position the resistors so that they bridge the gap between the inner rails. (Image 3)
4. Run wires from the cathode (short end) rail of the LEDs to the ground outer rail.
5. Run wires from the anode side of the resistors to digital pins 3, 4, 6, 7, 8, 9, 11, 12 on the Arduino to complete the circuit. (Image 7)

**Programming:**

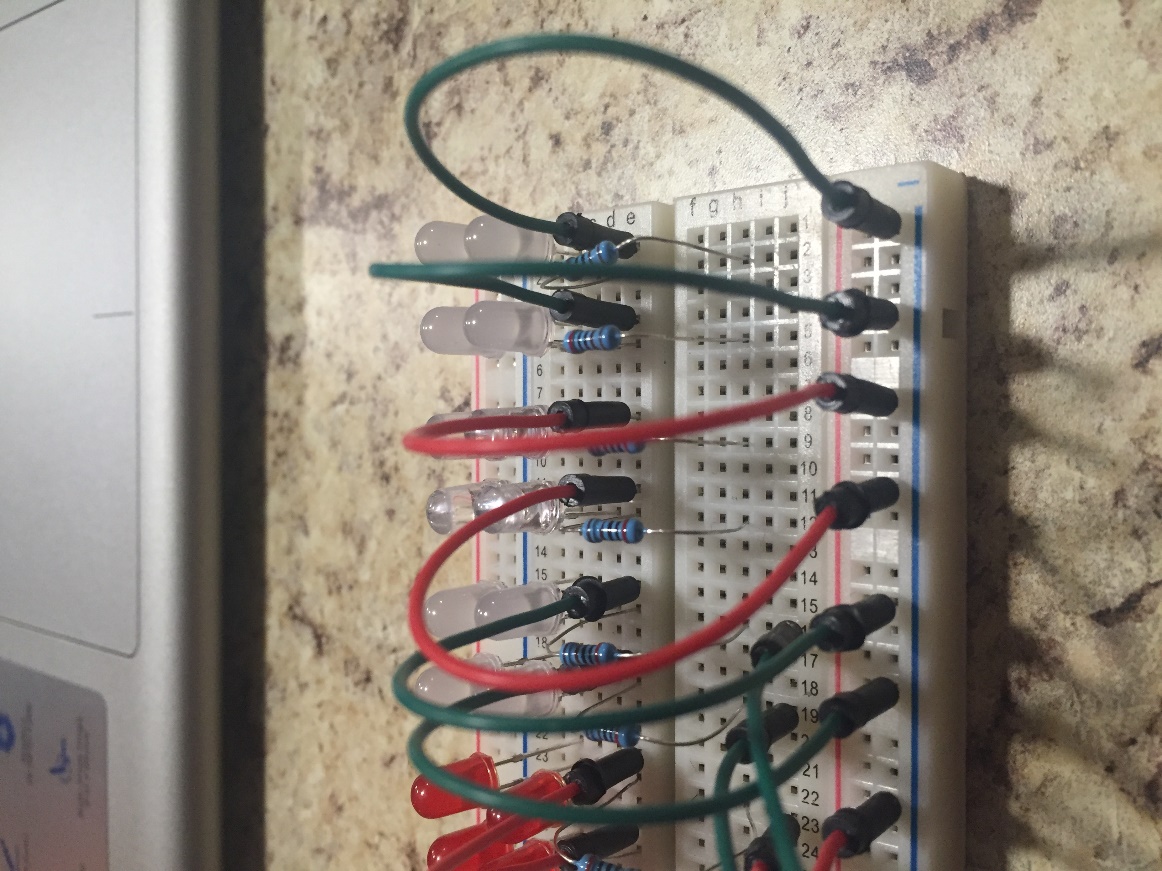
1. Connect the Arduino to computer using the A to B USB cable.
2. Download the Arduino and Processing Software Environments
3. Run Processing SE
4. Download the Arduino and Minim libraries for Processing
5. Run Arduino SE
6. Go to File -> Examples -> Firmata -> StandardFirmata, and press the Upload button on the Arduino SE to install the Firmata firmware onto the Arduino
7. Copy an .mp3 file into the "data" folder of the BeatWrite folder.
8. Open the BeatWrite.pde file in the BeatWrite folder in Processing.
9. Using the filename of your mp3, insert it between the quotes in the line of code "song = minim.loadFile(" ", 2048);".
10. Run the program and trigger the motion sensor once loaded.



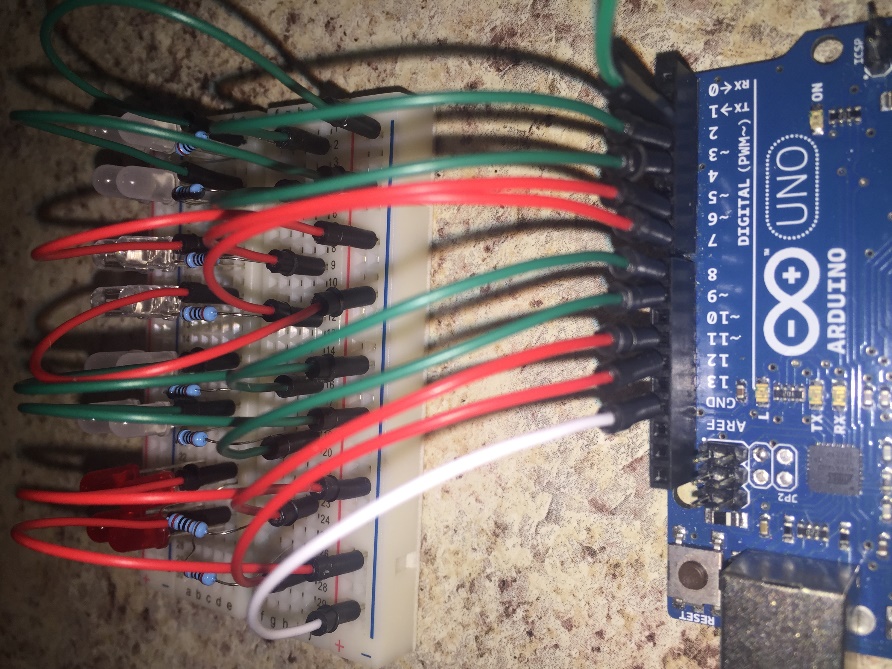
1

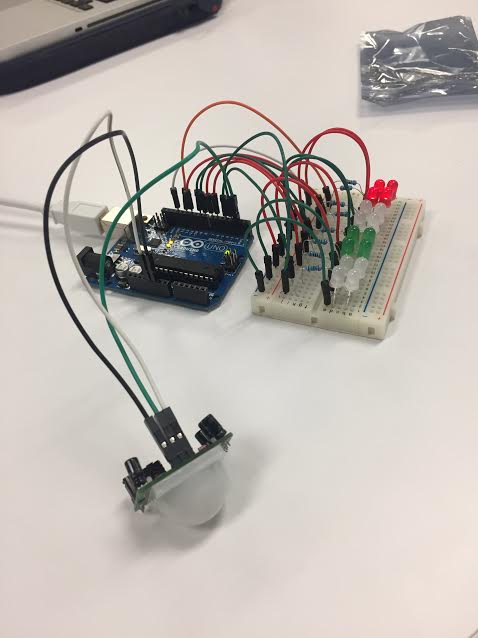
2

3



4





References

<http://www.instructables.com/id/How-to-Make-LEDs-Flash-to-Music-with-an-Arduino/>1

<http://playground.arduino.cc/Code/PIRsense>

<https://www.arduino.cc/en/Reference/Delay>

<http://forum.arduino.cc/index.php?topic=325291.0>

<http://code.compartmental.net/minim/javadoc/ddf/minim/analysis/BeatDetect.html>

1Our project code was adapted from existing code