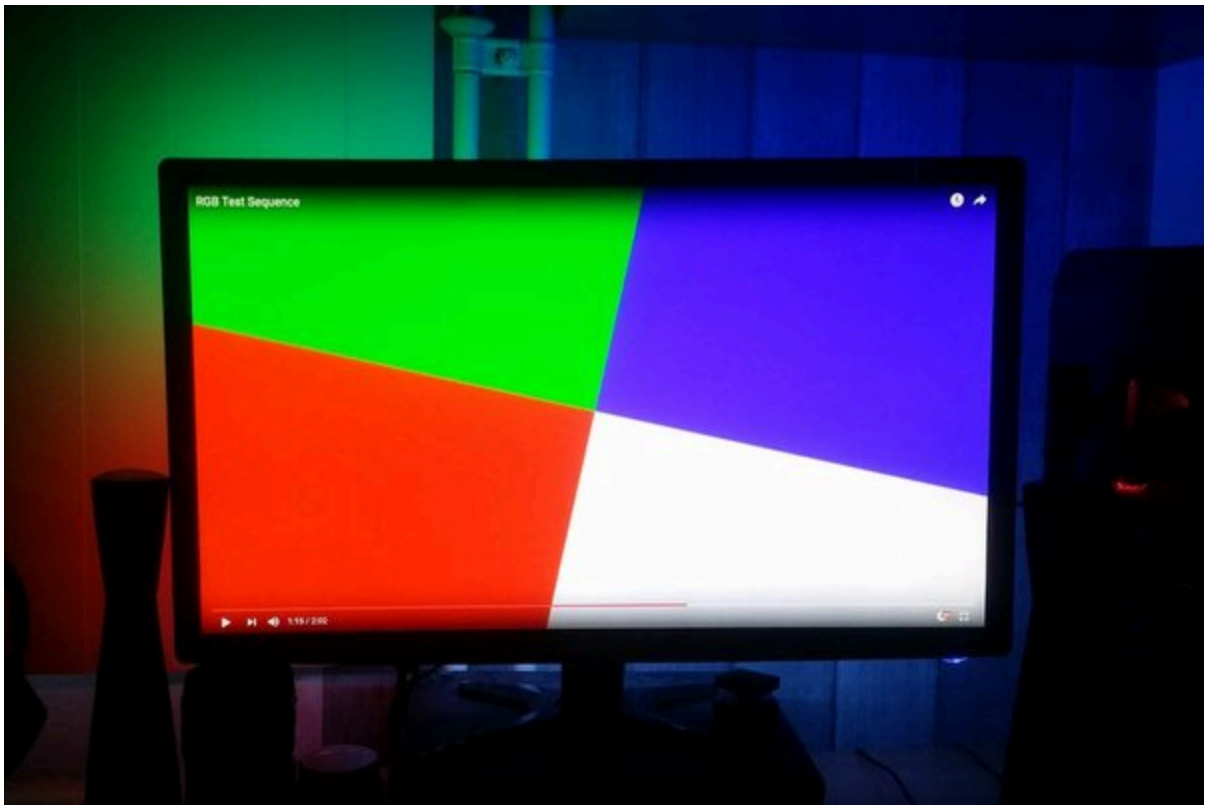
Roberto Kern & Eugenia Carmona
Group 10
ESE 190

Final Project Write-up: Ambilights

Philip Hue smart bulbs served as the inspiration for our Arduino-based Ambilights. Our idea was to create a smart RGB LED strip that would emit the same colors as those in the screen of a monitor, computer or television to which it was connected. The idea is that the LED strip could be pasted or taped in the back of the screen and serve to illuminate a TV room in a house. Such would entail having the lights changing with the colors being projected in the screen, to create a soothing visual ambient enhancer that would work with Arduino code and therefore require no work for the user once installed.



Picture credits. *Notice how the LED strips pasted on the back edges of the screen emit the same color tones projected on their corresponding edges of the screen.*

As an internal screen sensor based on Processing (simplified Java IDE) reads the colors being projected by the screen, this sensor would write the different tones of colors into the Arduino code for the LED strip so that the lighting of the room changes accordingly with the screen. We used two sensors to address the two problems in the original design:

- Problem 1: If we were to use a physical sensor to read the colors being projected by the screen, any other light source in the room where the lights are placed could interfere with the sensor readings, and the LED strips would therefore be unable to accurately reflect the RGB values being projected by the screen. For example, white light coming in through the window would make the sensor read as if the screen is whiter than it actually is.
- Solution 1: For such reason, we were able to use Processing code for an internal sensor that could read the RGB values emitted by pixels in the surrounding edge of the screen and then write these values on the LED strip to emit the same ones. This would solve the problem of potential effect of other light sources that would distort the RGB readings.
- Problem 2: The different lighting in the room where the Ambilight are placed would make it challenging to estimate a default light intensity value for the LED strip.
- Solution 2: By integrating an light-dependent resistor (LDR) to the project, brighter LED strips would remain visible in open and well-illuminated rooms while not being too bright for darker rooms. Essentially, the LDR would spit out a coefficient that would directly control the intensity of the LED strip to work as a function of the room's light condition.

Finally, while the Processing code would feed in directly into the Arduino IDE with the RGB readings based off of what is being projected in the screen, the LDR sensor would send its readings over bluetooth to the Arduino feeding the LED strips so that the LDR sensor and the strips could be in different places within the same room.

For our first demo, we started by preparing the sensors given that we were told a few days inside the project that the lab did not have any individually-addressable LED strips – meaning it took a few days to get ours from Amazon. For the demo we had a working LDR sensor and a functioning Processing internal color sensor; nonetheless, because we did not have an LED strip we could not show the lights. After that, by the day of the final project we had the LED strips projecting different patterns of highly visually-appealing light combinations, a working Processing internal sensor (nonetheless, we experienced issues having these readings being sent to the Arduino IDE for the values to be inputted into the LED strips), a working LDR sensor, and a working sound sensor that would measure the loudness of the room using a microphone (this component came as a way to integrate a second working sensor into the design given the Arduino IDE was not reading the values being sent by the working Processing sensor). Because the LED strip that we purchased consisted of 60 LED bulbs, we used the single-output power supply given the Arduino 5V source would not provide sufficient current for the lights to work correctly. The only problem we experienced during the final demo was hardware issues with the HC 05 for the Bluetooth, resulting in complications sending the readings from the LDR sensor into the Arduino connected to the LED strips.

Even though this has been discussed earlier, I will carefully describe how the different components of this design are meant to interact with each other.

- The Processing-based internal sensor would use Java to read the RGB values being emitted by the pixels in the border of the monitor. These values would be read with

Processing's equivalent of a serial monitor and would be inputted into the Arduino IDE so that the RGB values being written by its code would be the same as those in the screen, causing the screen and Ambilights to emit the same tones of light.

- The LDR sensor would read the luminosity of the room so that a brighter room would have brighter LED strip and viceversa; that way the LED strip is never too bright to be disturbing or too dim to be overshadowed by more light in the room. These readings would be sent over using HC 05's over Bluetooth to create a coefficient that controls the light intensity of the LED strip in real time.
- The LED strip is powered through a single-unit power supply at 5V and 0.50A. Furthermore, the individually-addressable LEDs would be lit using a combination of visually appealing light patterns using different codes, and would also be affected by the Processing and LDR sensors to adjust the tone of the light being projected and their corresponding light intensity.

In terms of selling this product, there are ways in which the design and underlying infrastructure can be enhanced and leveraged to sell this product for a variety of purposes. One way is to take advantage of the sensors to combine it with brighter, more potent lights. Once the coding infrastructure has been set to allow the sensors to adjust the light intensity, changing LED strips for more advanced light bulbs only requires minor tweaks. For such reason, such design could be used to decorate TV rooms in households, as well as for lighting in stages for concerts and musical events (even though this would require significantly more potent lights). For the latter use, this would cater to a sector where buyers are usually willing to invest hefty sums as they are used for expensive events.

Finally, some improvements I would propose are the following:

- Send the readings from the LDR and audio sensor through Wi-Fi (IoT) instead of Bluetooth so that the distance between the sensors and the LED strips are not restricted by the distance range of the HC 05.
- Finding a better tool to integrate the Processing and Arduino IDE's so that the readings by the internal processing sensor are effectively sent to the Arduino that writes the RGB values being emitted by the screen.
- Designing a visual controller for the patterns and intensity of the LED strips that would allow users with no coding proficiency to play with the lights in real time.

Link to final video: https://vimeo.com/265682568