

ART385 Design Document for Interaction Design

The ART385 Design Document emerges from a few different sources, including traditional [software design documents](#) and interface design workflows. The idea is to convey a design and code structures that run along with it.

Document Info

Natalie ART385, Mapping Physical Interaction to and from the Virtual (software+hardware), 7 April 2020

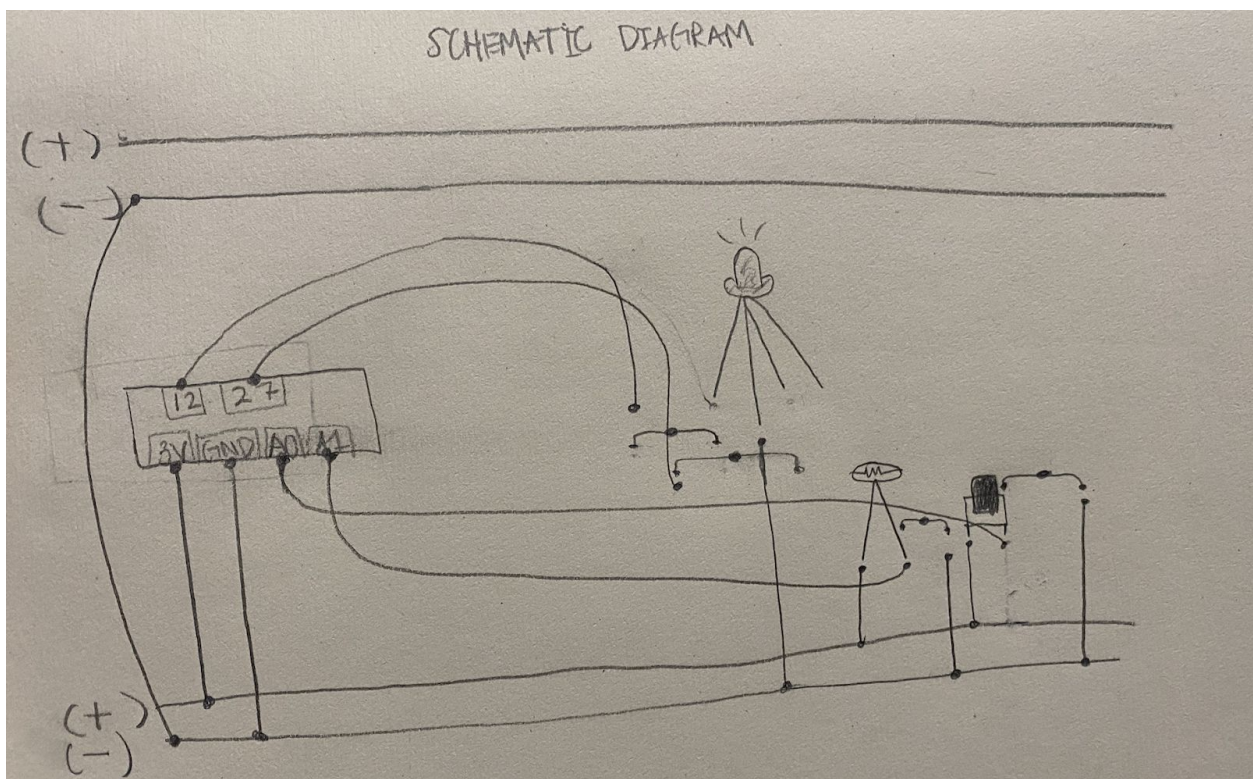
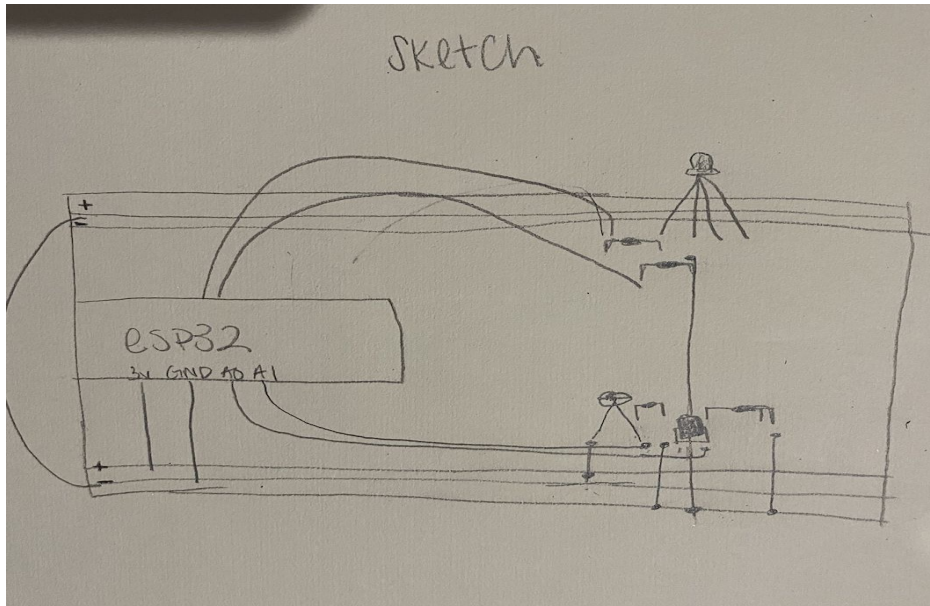
Re-state the Assignment

The purpose of this project is to develop our skills regarding code structure to make an interactive physical computing system. This project had us use two input and two output devices. This project used an esp32 and was coded in Arduino.

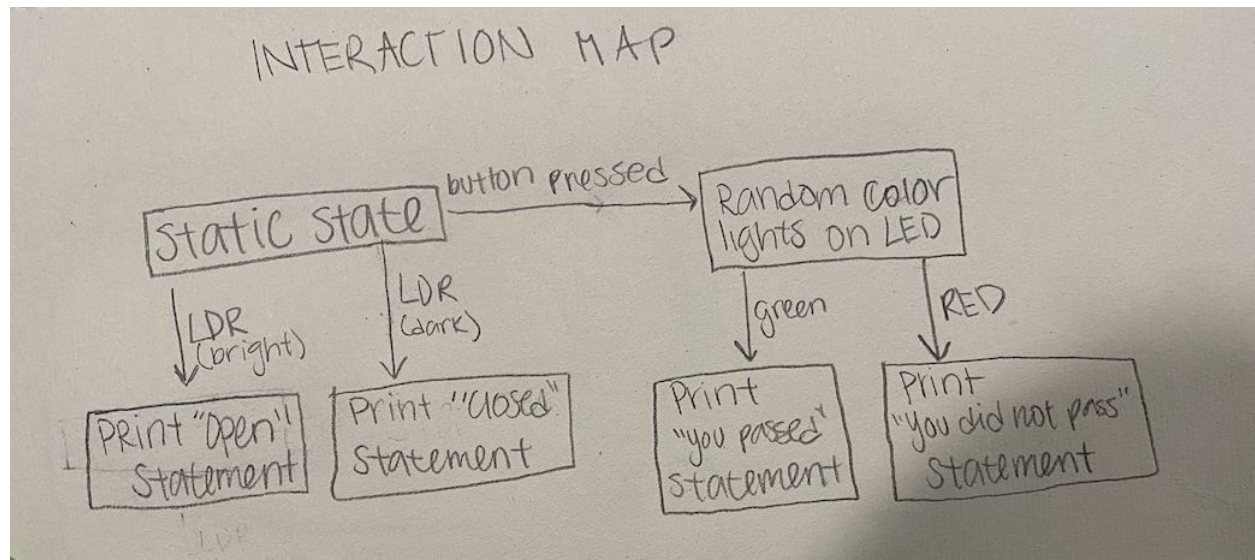
Audience

Who is your intended audience? Using the Design Thinking methodology, we start with Empathy and should be specific. If it is one of the **Weekly Assignments**, unless otherwise specified, the audience is the Professor and the other Students.

Hand-drawn sketches



Interaction Diagram (Interface Design)



Conveying Technical Information (Software Design)

This project was designed for airports. At the airport in Cabo San Lucas, where I went over spring break, I got the inspiration for this project. It is the only airport I have been too where they have around 10 kiosks where visitors push a button, and if it lights up red they have to get all of their bags checked, and if it lights up green, they get to exit the airport (fun fact: this was the first year I got red :(). My interactive computer uses the `random()` function to randomly pick a number 0 or 1 when the button is pushed. The number corresponds with the light that is illuminated. If the user gets 0, red will appear along with a printed message saying "You did not pass, Bag Check Required!" If the user gets 1, green will appear along with a printed message saying, "You Passed, Welcome!" This project also uses an LDR to detect whether the "station's" light is on or off. If the "station's" light is on, a message appears, "This station is open!" letting the user know they can go to that kiosk, and if the "station's" light is off they get a message saying, "This station is closed!" letting them know to pick a different one. This code looks to see if a button is pressed, and if it is it will implement the `random()` function then choosing a color to illuminate and the corresponding message to display. The LDR uses `analogRead()` to see if the brightness level is above 0, and if it is the open message appears, and if it is not the closed message appears. In addition, I used the `millis()` function to display these messages at a slower rate.

Reflections

This project actually reflects my original design idea. I came up with this idea and was super excited about it. When I began to code this project, I got super frustrated extremely fast. I wanted to give up a few times because I could not figure out how to randomly select a color without it

running funky. In addition, my print statements were all over the place which made me extremely irritated. Once I finally got this code up and running, it felt extremely rewarding!