

CSC 2463

# Final Project Documentation

By: Stephanie Mayberry

## Final Project Documentation Table of Contents

1. Project Outline.....	3
2. Original Project Proposal.....	6
3. Final Project Description.....	8
4. Project Images / Video.....	10
5. Conclusion and Future Development.....	11

## Project Outline

### Final Digital Media Project

#### I. Graphics / P5.js

##### a. Ping-Pong Game

##### i. Start Screen

1. Arduino Connect Button
2. Press to start Game

##### ii. Playing Screen

1. Player 1 and Player 2 scores
  - a. Increase when a player wins a round when the other player misses the ball
  - b. Does not increase or decrease if the timer runs out and no one wins the round
2. Player 1 and Player 2 paddles
  - a. Controlled on Arduino
3. One Ping-Pong Ball
  - a. Can bounce of the left and right side of the screen
  - b. Can bounce of either player's paddle
  - c. Stops moving if it goes past either player's paddle
4. Timer
  - a. 30 Second Countdown
  - b. Ends the round after 30 seconds
  - c. Resets for each round

- iii. Game Over Screen

1. Show who won the round
2. Show the current score between players
3. Press to restart a round and go to Playing screen

## II. Sound / Tone.js

- a. Ball Hit sound

- i. Freesong.org

1. Only 1 second long
2. Sound like a ball bouncing once

- ii. Played only when ball comes into contact with a paddle from either player

1. No sound if it bounces off the left and right wall
2. No sound when the player misses the paddle

- b. Background Song

- i. New Synthesizer sound

- ii. Set specific notes

1. Group those notes into certain play patterns

- iii. Loop it to continue to play the entire round on playing screen

- iv. Make loop play faster on each ball hit

1. Playback rate increases by .1

- v. Stop and restart playback rate at the start of each round

## III. Physical / Arduino

- a. Four Buttons

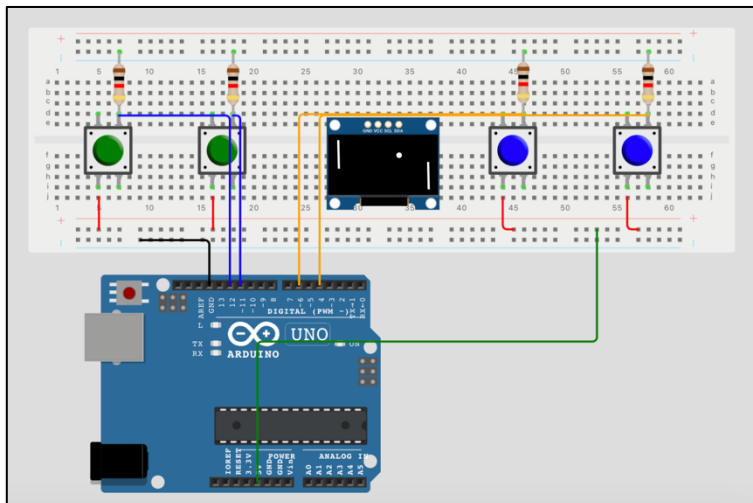
- i. Two Buttons for controlling Player 1 Movements

1. Move paddle from left to right
  - a. Move proper direction smoothly
- ii. Two Buttons for controlling Player 2 Movements
  1. Move paddle from left to right
    - a. Move proper direction smoothly
- b. RGB LED Light
  - i. Be able to produce 3 colors
    1. Blue
      - a. If Player 1 is in the lead LED will stay Blue
    2. Red
      - a. If Player 2 is in the lead LED will stay Red
    3. Green
      - a. If both players have equal scores then the LED will stay Green to indicate a tie

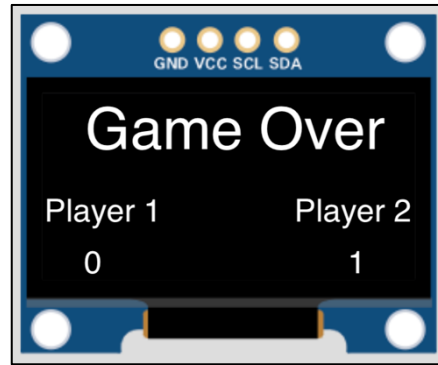
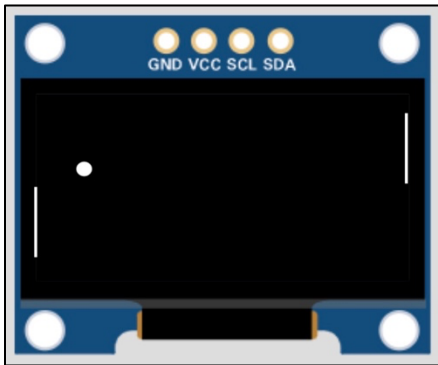
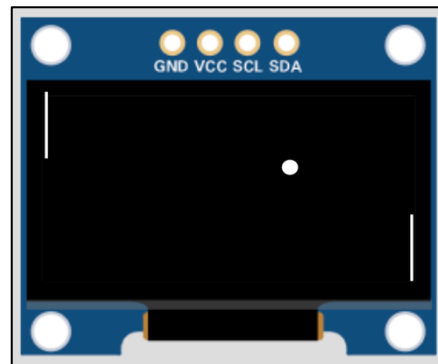
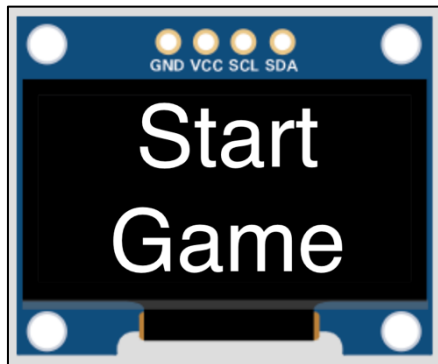
## Original Project Proposal

For my project, I wanted to create an engaging and accessible two-player simple game on my Arduino board. Using an Oled display and four buttons I want to create a back-and-forth motion of a traditional ping pong game. I believe this project will be both enjoyable and achievable, allowing me to explore the full capabilities of the Arduino platform. To bring this project to life I needed a physical, graphics, and sound aspect of the project.

For the physical aspect of the program, I will be using wires, Oled display, and buttons to control the program. The Oled display will serve as the game board, displaying two bars on either side and a small dot in between to simulate the movement of a ping pong ball. Also, the buttons will control the paddles, allowing players to move them up and down on the screen. For the graphic aspect of the program, on the Oled display there will be two bars on the left and right of the screen and a small dot in between them. The small dot will travel between the two bars to simulate a ping pong going back in forth between two paddles. Lastly, for the sound aspect, there will be a background music sound, sound every time the ball comes in contact with a paddle, and a losing sound if the ball goes out of bounds. The game will be programmed entirely on the Arduino board and will keep track of each player's score, creating a competitive and engaging gaming experience.



## Different versions of the Oled Display:



## Project Description

For my final project, in which I needed to implement graphics, sounds, and physical components, I created a two-player ping-pong game, also known as table tennis. I used p5.js, tone.js, and Arduino to assist me in the creation of this project. In my original proposal, I was going to have everything located on the Arduino circuit board, but after much consideration, I realized I would not have enough room on the circuit board to properly display my project, and I felt there would be better graphics and sound if I had it as a website that connected to the circuit board via serial communication. This project allowed me to demonstrate all the various technologies we learned this year and integrate them into one final project.

There is a difference between each game and each round in this project. Each game starts when the page is refreshed, the Arduino is initially connected, and the player is ready to start their first round. Once the player starts the first round of the game, there can be as many rounds in one game as the players want, but to start a new game, the screen needs to be refreshed to reset all score tracking. There is a 30-second timer per round, and if both players continue to hit the ball for the entire time, then the round ends and no one gets a point. From the playing screen, it goes to the game over screen after the timer goes out or the player does not hit the ball. Once the player reaches the game over screen, it will go back to the playing screen after the player presses on the computer screen with the mouse, and there is no way to get back to the start screen unless the player refreshes the page, indicating the player wants to start a new game.

For the p5.js graphics portion of this assignment, I started by creating three different screens for each element of the project. Firstly, I had a simple start screen. In the top left-hand corner of the screen is a connect button in order to have the graphics connect to the Arduino, and that needs to be connected before starting the game. There are no other interactions on that



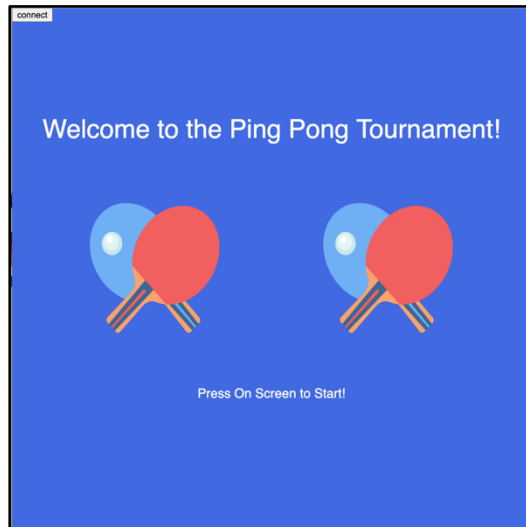
screen, so the user needs to use the computer mouse to press on the screen to start the game. The ball can bounce off both paddles and the left and right sides of the table, and if the ball passes the X and Y values of either paddle then the round ends and the player to hit the ball last gets the point. The controllers for moving the paddles are on the Arduino circuit board, and on this screen, there are no interactions between the keyboard or mouse and the game.

I decided to have two specific sounds integrated into my project. The first sound is for the background song going on during each playing round. This is made up of a string of notes separated into separate patterns inside an envelope. It continues to loop while the playing screen is activated. This sound also gets faster and loops faster as the ball is hit, in order to enhance the clock ticking down to the end of the round. The second sound is a sample I used from [freesound.org](https://freesound.org) and is used to represent a ball hit. This is played every time the ball comes into contact with either paddle. These two sounds are only played during the playing screen, and there is no sound present on the start screen or game over screen.

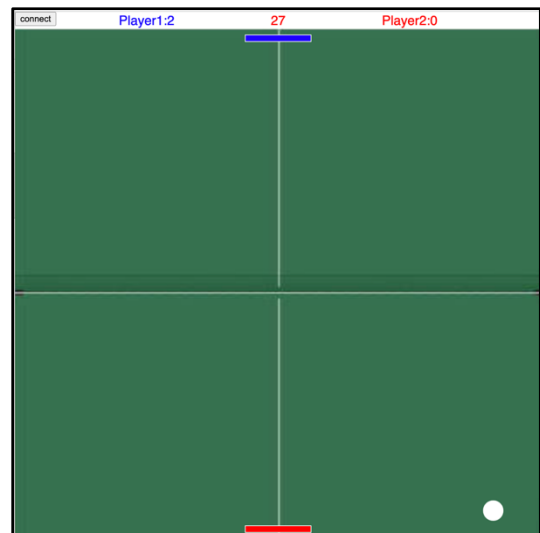
For the physical aspect of the project, I used an Arduino with multiple wires, buttons, and one RGB LED to complete the project. The four buttons are representative of the two paddles ability to move left and right. On the left side of the board, the two buttons are for controlling player 1's movements, and the buttons on the right side of the board are for controlling player 2's movements. In the middle of the four buttons, there is one RGB LED. This LED helps display who is currently winning. If the players' scores are equal or neither player has gained a point, the LED is green. Once points have been given, if Player 1 has more points, the LED is blue because in the game Player 1 is blue, and if Player 2 has more points, the LED is red because Player 2 is red in the game. The Arduino does not stop or start any game and is only affected based on the playing screen and actions.

## Project Images

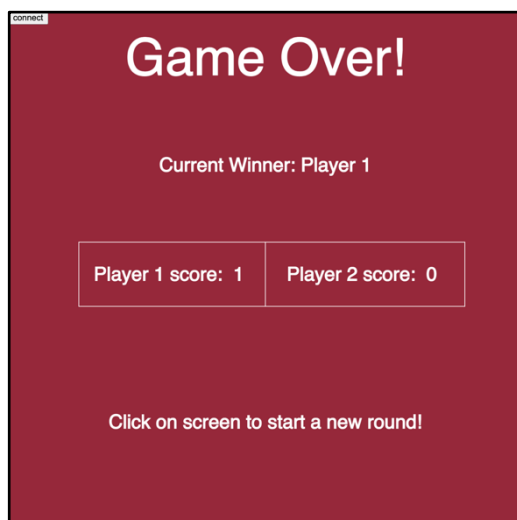
Start Screen:



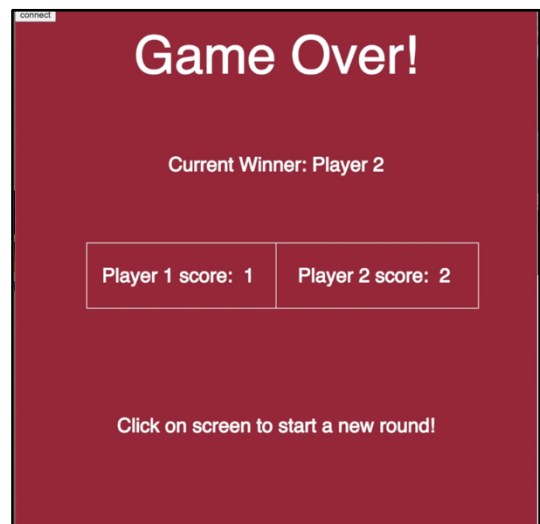
Playing Screen:



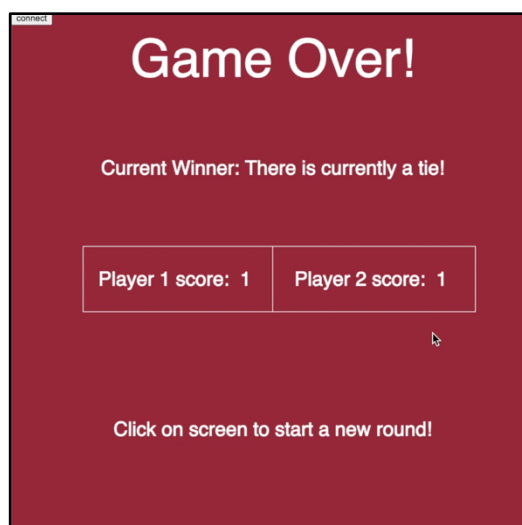
End Screen Player 1 Winning:



End Screen Player 2 Winning:



End Screen Tie:



Video Link:

YouTube Link: <https://youtu.be/sMvuE70A8po>

## Conclusion and Future Development

Overall, I felt that this project allowed me to successfully show the different processes we learned in this class, Programming Digital Media. As I mentioned before, I had to change my idea from my original proposal in order to make it more accessible and produce a quality product. I believe that since I changed some components of the project, I was able to be more detailed in my work and accomplish the tasks with quality. When I was starting this process, I started small by just making the ping-pong game work with buttons on the laptop, then continued to build it up by adding different game screens and sounds. I added the physical component last so I could confirm the game worked and just needed the physical component to control the movements instead of the buttons on the keyboard. This was a completely new process for me dealing with sound, wires, and buttons, but from this class and their teachings, I felt very successful in my final project.

If I were to continue to develop this game, there are a couple key points that I know I would want to add immediately. Firstly, for graphics, I would want to make the ping-pong ball get faster as the timer counts down in order to make it more difficult. I also like the idea of adding multiple balls going in opposite directions in the case of a tie. This could be controlled by the players to decide if they want to play on easy, medium, or hard difficulty. For sound, I would want to add a song on the start screen that is very simple and inviting, and then another song on the game over screen to confirm to players that the round has ended. Lastly, for physical components, I would want to make it simpler with fewer wires and add a button in order for the players to switch between screens with no contact with the keyboard of the computer. Even with these future ideas, I believe my current project was very successful for my level of experience in this field.