CS/EE 120B

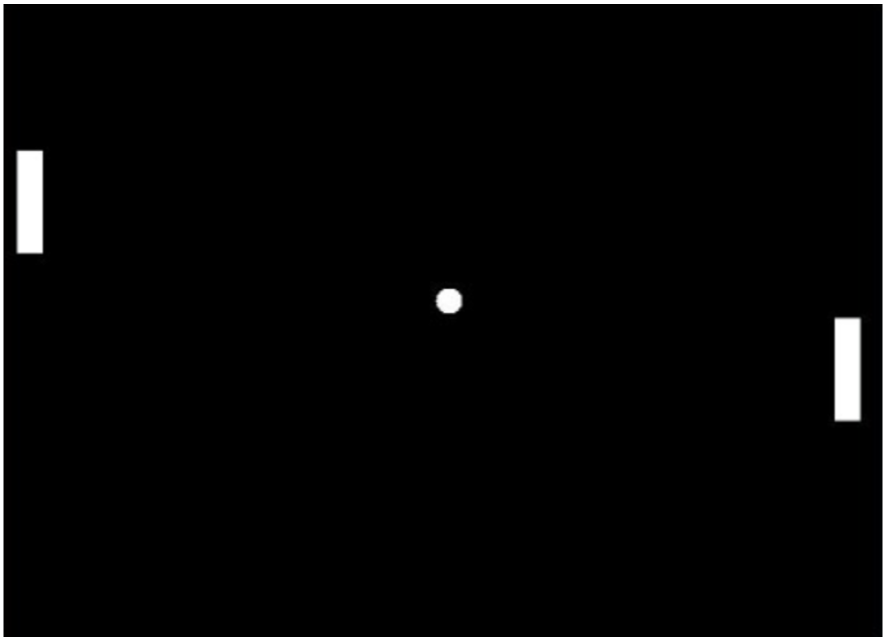
Custom Project: Pong

Alex Madera

March 17, 2019

Introduction

Pong is a two-dimensional sports game that simulates table tennis. The player controls an in-game paddle by moving it vertically across the left or right side of the screen. They can compete against another player controlling a second paddle on the opposing side. Players use the paddles to hit a ball back and forth. The goal is for each player to reach seven points before the opponent; points are earned when one fails to return the ball to the other



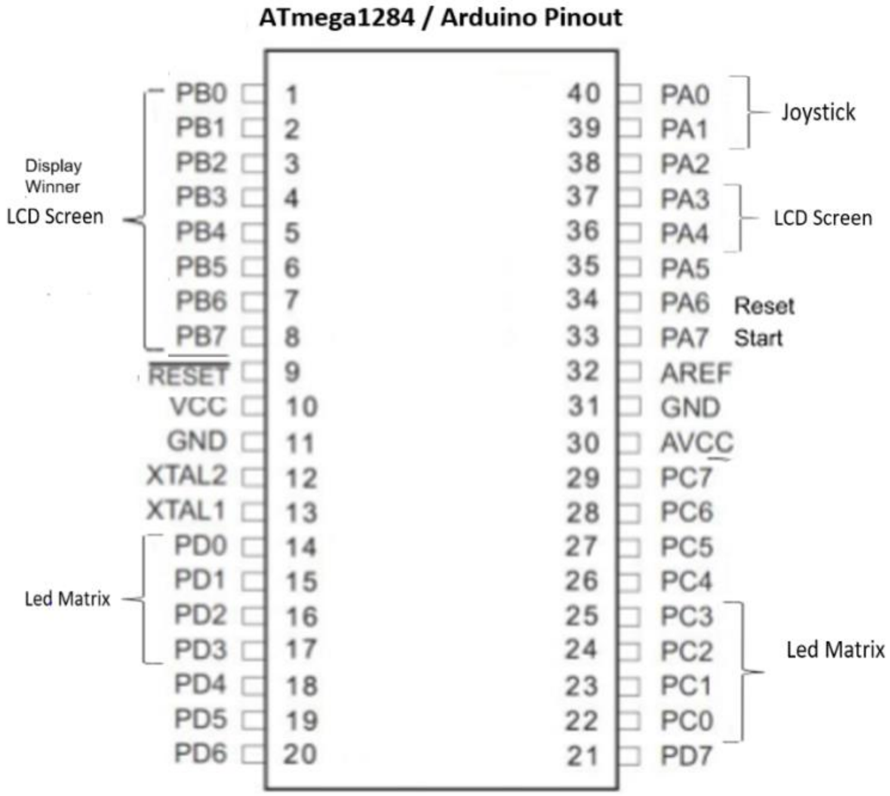
Technologies (Components)

- **Atmel Studio 7**
  Integrated development platform (IDP) for developing and debugging all **AVR** microcontroller applications
- **ATmega1284**
  High-performance Microchip 8-bit AVR RISC-based microcontroller

Components (Pin-out)

- **Inputs**
  - PA0 (2-axis joystick control)
  - PA1 (2-axis joystick control)
  - PA6-PA7 Reset and start buttons

- **Outputs**
  - PD0-PD3 LED Matrix (y-axis) – display game
  - PC0-PC3 LED Matrix (x-axis) – display game
  - PB0-PB7, PA3-PA4 LCD Screen – display messages, score, and winner



Complexities/Build-Upons

1. LED Matrix – display game
2. Shift Registers (2) – maximize port usage
3. 2-axis Joystick (2) – player controls

User Guide (How it works?)

- Game starts with ball on (right) paddle 1
- Ball moves back and forth vertically between both paddles (opponents)
- Paddles can be moved left/right by user
- If ball move past the paddle of either player opponent scores one point
- The game resets after each point at paddle 1 till either player scores 7 points
- Winner (informational messages) is shown on LCD Screen and Game on LED Matrix
- Player can reset game with button connected to PA6 on the AVR at any time.

User Engaged Controls

1. 2-axis Joystick

- User can shift left and right to move paddle (up/down disabled)

2. Button

- Use can press to reset at any time during game

3. Display

- Displays Game

Link to Demo Video

- https://www.youtube.com/watch?v=8TPLRwJhwkQ&t=41s

Link to Source Files

- https://github.com/iperson98/Pong

  Three Files:

  Pong.c

  Description: (Logic)
  - Here I have a method (int main) that calls multiple functions that carry out commands asynchronously on a timer of 10 ms.

  - The functions are: (1) Function to Move paddle 1, (2) Function to move paddle 2, (3) Function to move ball from player to player, and (4) Function to reset the game. All these functions are checked every 10 ms and carry out state-by-state.

  - The sub-functions are: (1) Function to Shift Registers in PORTC, (2) Function to shift registers in PORTD, (3) Function to move Joystick1, (4) Function to move Joystick2, and function to read ADC Value in PA0-PA1.

  - Other functions include the timer function for setup.

  io .c/.h (helper files)

  Description:
  - These files define the methods that can be called and setup the actions each method takes on the LCD Screen.

Acquired Skills

- Throughout my custom laboratory project I was able to learn a variety of new ideas and skills. I learned how to program a microcontroller (ATmega1284) using AVR programming software (Atmel Studio 7) and make it engage with hardware via its ports on a breadboard.
- When using Atmel Studio I wrote code that would execute functionalities on my hardware (complexities). The code was organized using the high level understanding of synchSMs, which set timed instructions to be carried out to the ports of my selection on the microcontroller.
- Through this also learned to read data sheets to adequately wire and program the components used.