Fly Dot

Custom Project Final Report

Spring 2017

Braeton Buckley

Table of Contents

[**Introduction**](#_n9o44neiquk5) **2**

[**Hardware**](#_vqmbsuo5p330) **2**

[Parts List](#_xtqh2tzibtrr) 2

[Pinout](#_8yptvhjzojq8) 3

[**Software**](#_w9543qe124on) **4**

[**Complexities**](#_sn48u4uktu3c) **4**

[**Youtube Link**](#_u9cxbem9510r) **5**

[**Known Bugs and Shortcomings**](#_grlzpb6vy2cq) **5**

[**Future work**](#_qbv6f31drpex) **5**

[**References**](#_1txxevdbm0qb) **5**

# Introduction

My project uses an accelerometer to play a game displayed on an 8x8 LED matrix. The character is a single lit LED whose movement is limited to the leftmost column of the LED matrix and whose upward and downward movement within that column would be controlled by the accelerometer. The accelerometer detects positioning on the z-axis, and moves the character accordingly. Enemies would be lit LEDs approaching from the right toward the character, which the character will have to catch. The starting position of the enemies is pseudorandom, each enemy spawns as the last one has finished crossing the matrix, and srand can be seeded with a button press to create real randomness in the enemy starting position. There are 4 levels, you must score 4 points on each level to reach the next one. Each time you catch an enemy, you gain a point, and each time you miss one, you lose a point. You need 4 points to proceed to the next level, and each new level the enemies get faster. After 4 levels, a celebration screen is displayed.

19125564_1558138860884651_573256267_o.jpg

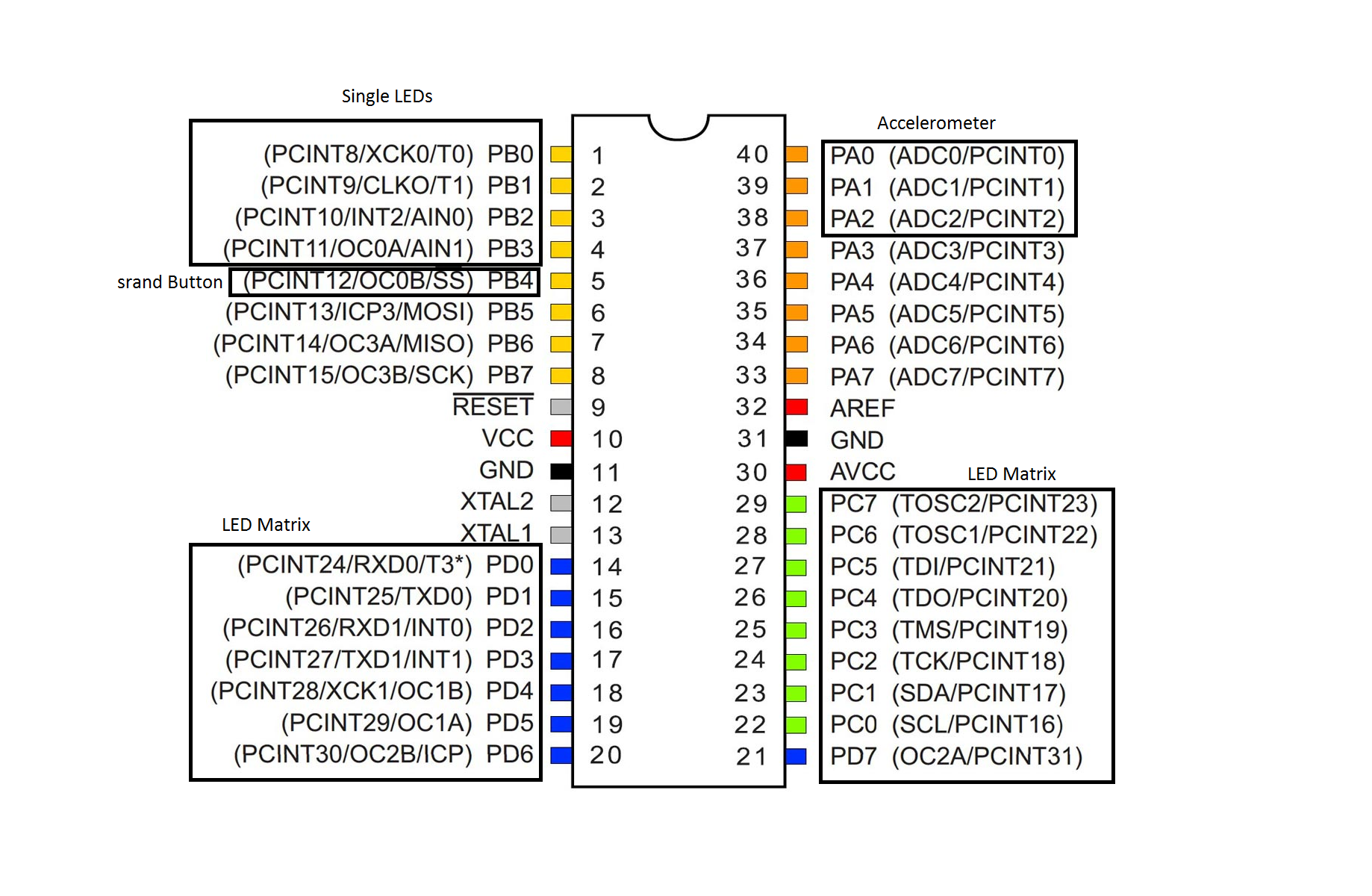
# Hardware

## Parts List

The hardware that was used in this design is listed below. The equipment that was not taught in this course has been bolded.

* ATMega1284 microcontroller
* Button
* 3 single LEDs
* **8x8 LED Matrix**
* **Accelerometer**

## Pinout

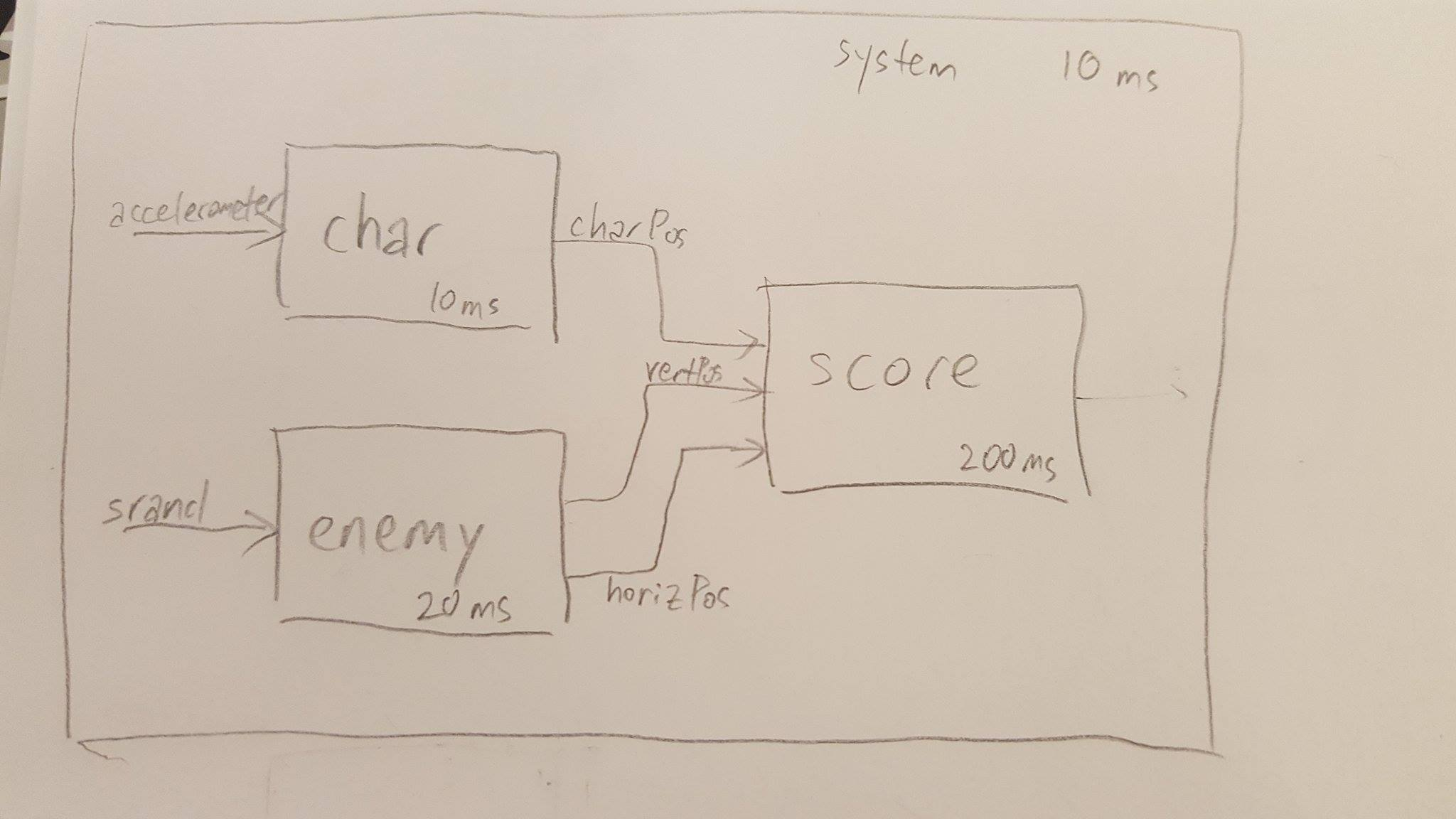


# 

# 

# Software

The software designed for this project was implemented using the PES standard. The overall design as a task diagram is included below.



Now I will write a short description of the tasks in the project. The appendix will include SM’s that I designed.

# Complexities

## Completed Complexities:

* Accelerometer as a controller
* LED Matrix
* Game Levels
  + 4 Defined Levels and a victory screen
* Random Map Generation
  + Real randomness of spawn positions with srand button

# Youtube Link

https://youtu.be/1tQ7aReRmcI

# Known Bugs and Shortcomings

* In order to have a victory screen I just created a display in main once the last level is defeated. This could be seen as a dead state but technically the game continues to play while the victory screen is displayed. The victory screen was just for fun so if I wanted to fix it I would just need to actually include it in the state machine.
* There is no way to reset the system except by doing a hard reset, I should include a button that does a soft reset
* The LED Matrix did not have enough power going through it to properly light up the middle 4 columns (or rows, I suppose). I tried to use a shift register to fix this since it has its own internal power. After many hours of not being able to get the shift register to work properly with the LED Matrix, I decided to use the lack of power as a feature. The middle 4 lights light up only when that row (or column) is enabled, so it actually helps to see if your character is lined up with the enemy. Without it, it is difficult to see if your character is in the same row because they are just fast moving dots. So in the end, I actually like it better this way. A happy accident :)

# Future work

If I were to continue the project I would make the game more interesting by creating more complicated enemy behavior in higher levels rather than just moving across, and have different colors to signal the different kinds of enemies. Maybe also create powerups that can be caught to make the character stronger like slow down time.

# References

Timer and especially task scheduler template code from the labs were used. Header files were included in the project folder.

I would also like to thank my friends for letting me use their parts since I’m cheap and never bought my own blue box.

Wesley Harada: Breadboard, Power Source, LED Matrix, Accelerometer

Jennifer Shin: Resistors, Single LEDs

Joshua Beto: Programming Chip

Chris Yee: USB to USB power cable, Button

Dan Zarate: Extra Jumper Wires