

Side Scroller

Custom Project Final Report

Summer 2018

Sili Guo

Introduction

The game of Side Scroller test user the ability of precise controlling and timing during fast moving.

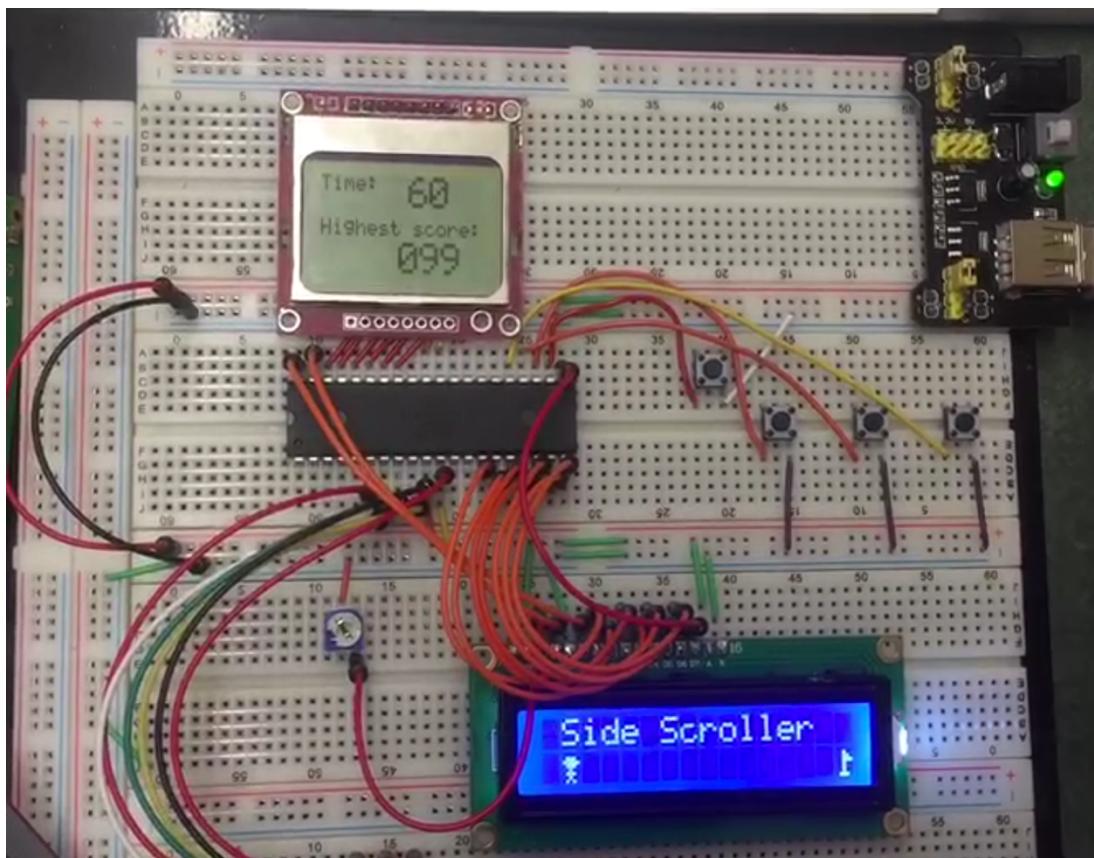
Basically, the player needs to move a character to avoid thorn and ghost to reach the final point to get the flag. There are four buttons, one for start/pause/reset game, the other three control the character to move left, right and jump. In the map, bricks won't hurt you, but you need to jump to avoid them. Also, there are thorns and a ghost that player should avoid fast, or one will lose the game.

There are two ways to lose the game: step on thorn or ghost and time up

Players will get a score after the game. The initial score is 100, based on different condition, the points will be deducted. if one loses, he loses all points; if he wins, he will receive a score based on the time he passed the game. Also, if he never died in the game, he can get 20 bonus points.

The highest soccer of history will also show on the screen; so try hard to be the No.1!!!

Picture:



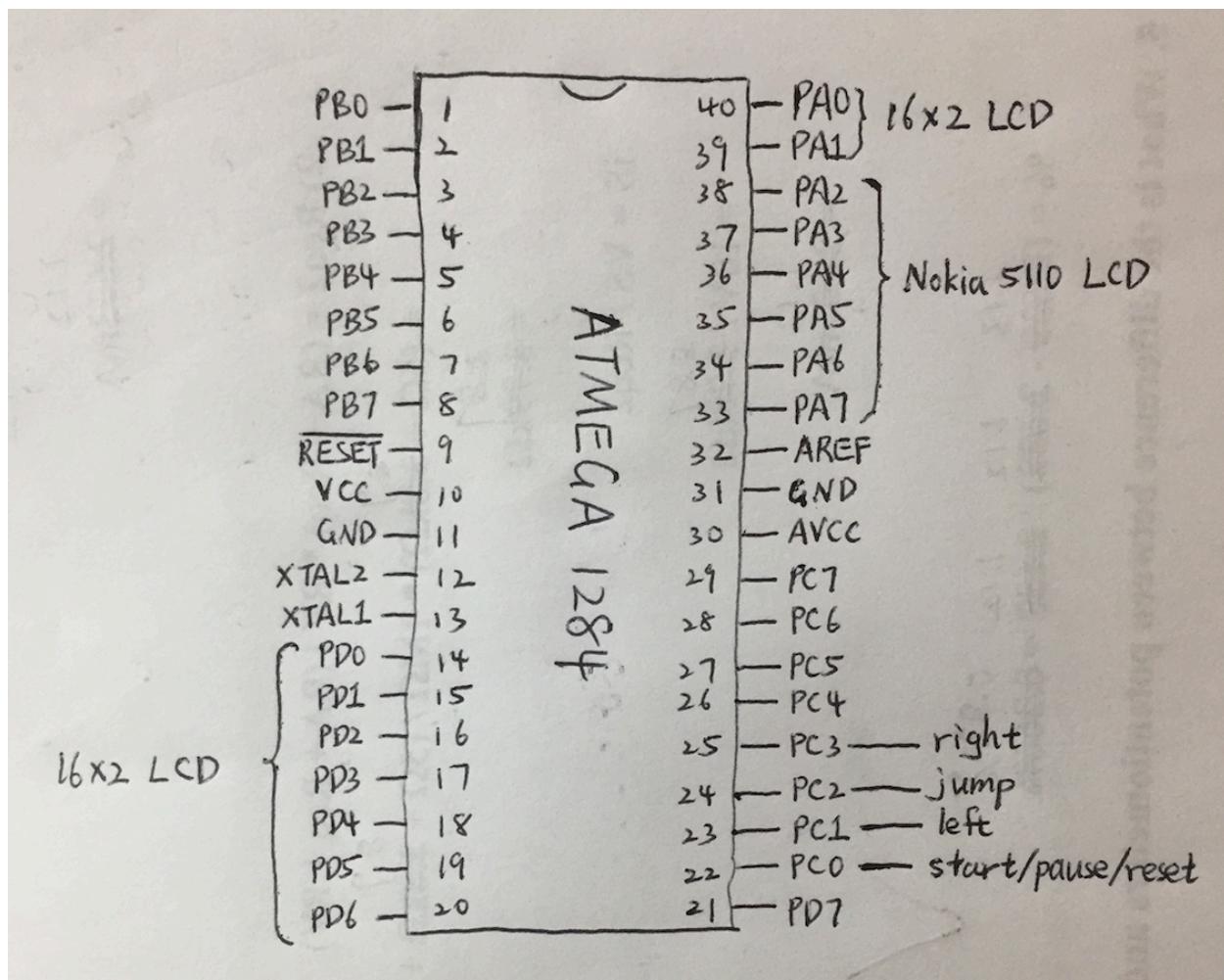
Hardware

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.

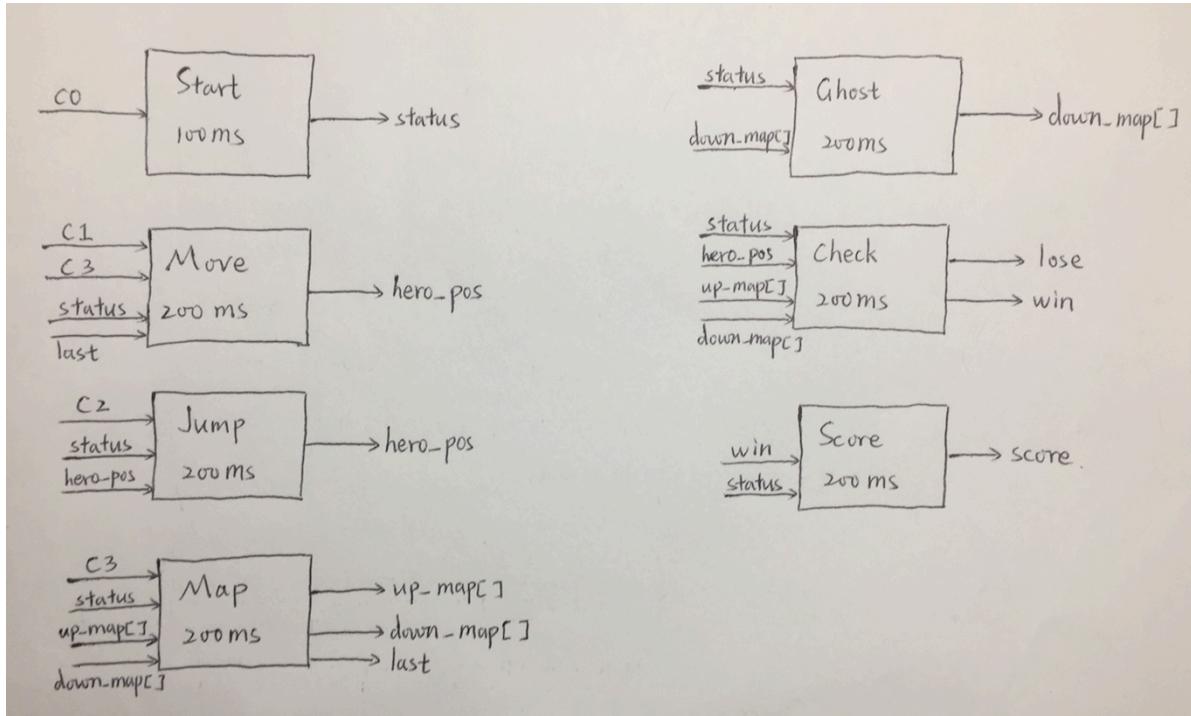
- ATMega1284p microcontroller
- 16 * 2 LCD Screen
- **Nokia 5110 LCD Screen**
- Buttons

Pinout



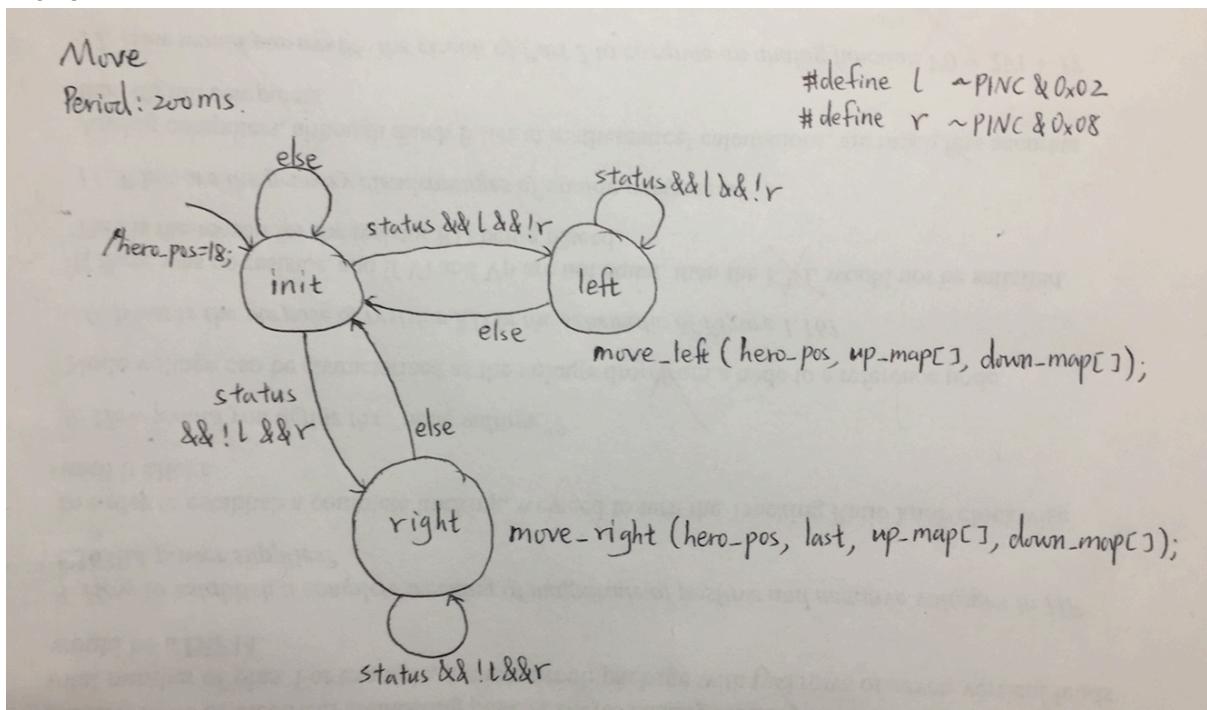
Software

Task Diagram:

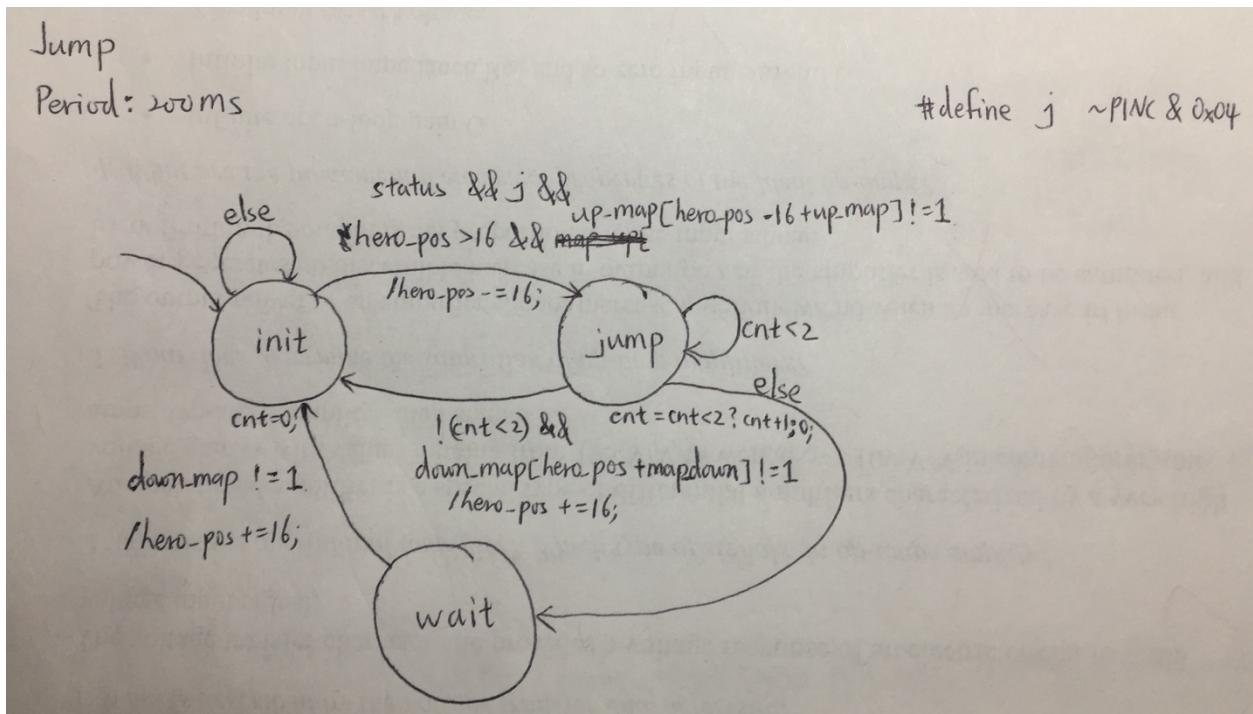


State Machine (Move, Jump, Map):

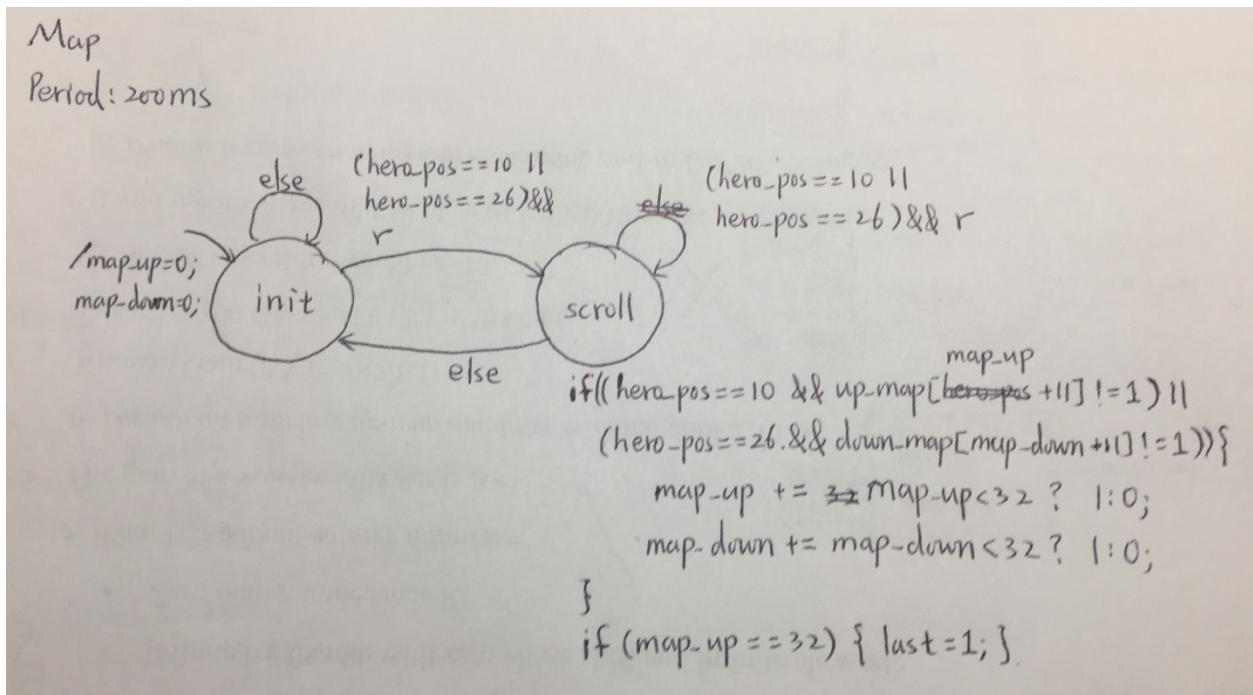
Move:



Jump:



Map:



Complexities

Completed Complexities:

- Game logic (Side Scroller)
- Connect and implement the function of Nokia 5110 LCD screen
- Using EEPROM to save the highest score (1/2)
- Creating custom characters on the LCD screen (1/2)

Incomplete complexities:

- Second player (due to the time)

Youtube Link

<https://youtu.be/WMHSQNN4zFw>

Github Link

<https://github.com/SiliGuo/UCR/tree/master/CS%20120B/Custom%20Project>

Known Bugs and Shortcomings

- The game I made is simpler than I have thought. I was planning to create several collections to increase the score, and one of the major props should hide below a brick. With that prop, the character can use weapon in the final battle with boss; otherwise, he can only move and jump to avoid the attack from boss and achieve the final goal.
- I have figured out the task diagram of that part, as well as logic. But due to the time limit, and I met several bugs when implementing the scene of attack, so I didn't achieve that part. And I think I can finish that part later.

What did you learn

During this project, I learned how to do the design with practical problems. I followed the steps that first analyze the problem and write down the function I need my game can do. Then I separate these functions into tasks and draw the task diagram and a brief state diagram for each task. And finally implement one by one in C code.

I realize it is a good way to implement each function individually, so you will not be confused by not knowing the bug come from which task.

I also learned to do the design before writing the code, because in the diagram, we can see the transition between states better, and will not left some conditions nowhere to go.