

Design Manual

1. Introduction

This project need to develop a simulation system with AVR Development board, controlling a drone to search an accident location in the mountain. Since searching an accident is dangerous to human, this project can help us to complete this mission with drones.

2. Assumption

- 2.1 This mountain has no cave.
- 2.2 This mountain has no cliff.
- 2.3 We turn this mountain into a 2D array of surface points, and it is rough.
- 2.4 To simplify the design, I assume the mountain is like a pyramid in shape.

3. System Architecture

3.1 Program memory

The mountain is very large, and its x and y are in range of 64 meters. Besides, I need to read each point to get the height of the mountain in that location. So I use '.db' to load this mountain, and then read it by Z-register from program memory.

To generate the mountain, I wrote a simple python program to print a line, which consist of comma and numbers. This python program need to input 'number' and 'length', and it will generate a list consisting of that number. Besides, the length of the list is 'length'. After it prints a line, I combined several lines together to make the mountain.

4. Detail Design

4.1 System Control Overflow

(1) Start



Figure 1: start state

Figure 1 shows that what happened to the system when I pressed PB0, which is connected to RDX4 on my AVR Lab Board, the system will call external interrupt INT0, then the system run the following code, and LED starts to flash. After LED flashes three times, the system will execute LCD display code. Finally, it displays '(' on the LCD, waiting for the user to input the accident location x and y.

(2) Input accident location

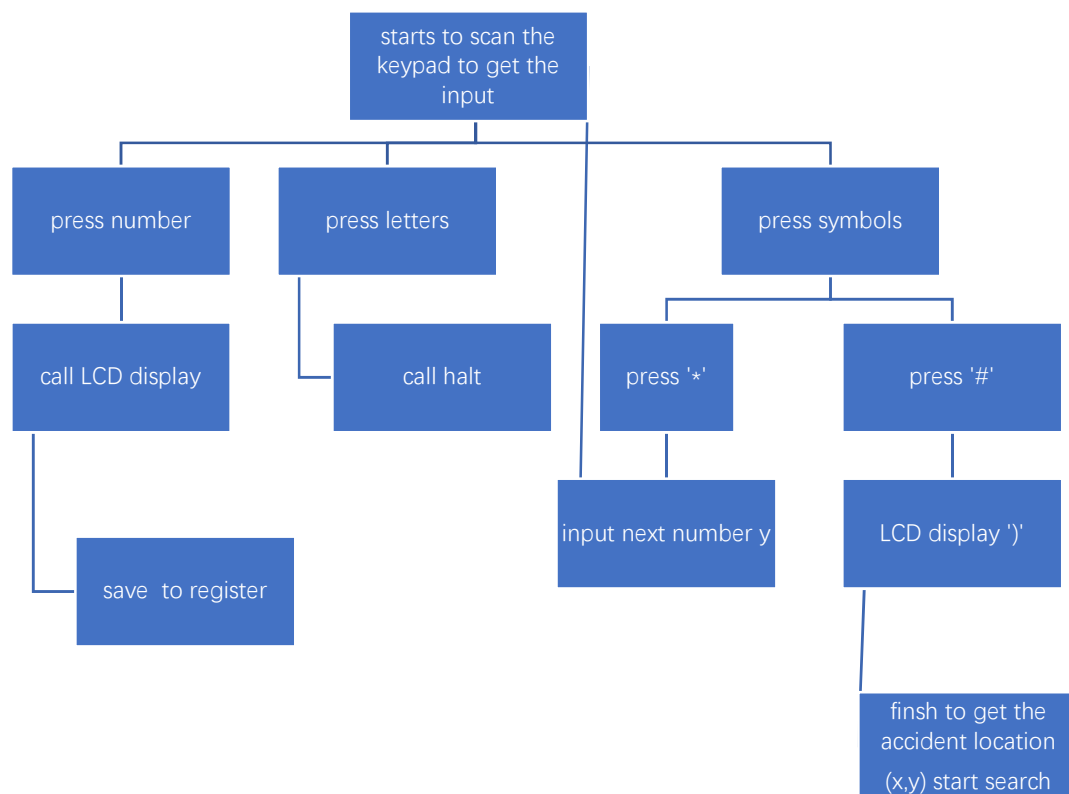


Figure 2: input accident location

Figure 2 shows the procedure of input accident location. I need to judge which key is user pressed. If user pressed number, then I get the tens and ones of that value, and save it to register to get the accident location. If user pressed letters, which I do not need, I call the halt to stop. If user press symbol, then I will check which symbol is he or she pressed. '*' means the end of the x input, while '#' means the end of the (x, y) input. Finally, after getting the accident location, I can start search.

(3) Search

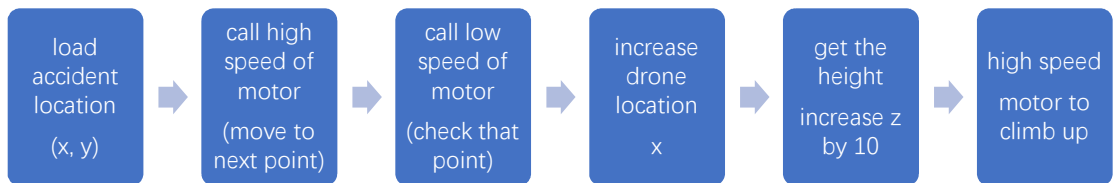


Figure 3: Check point by point

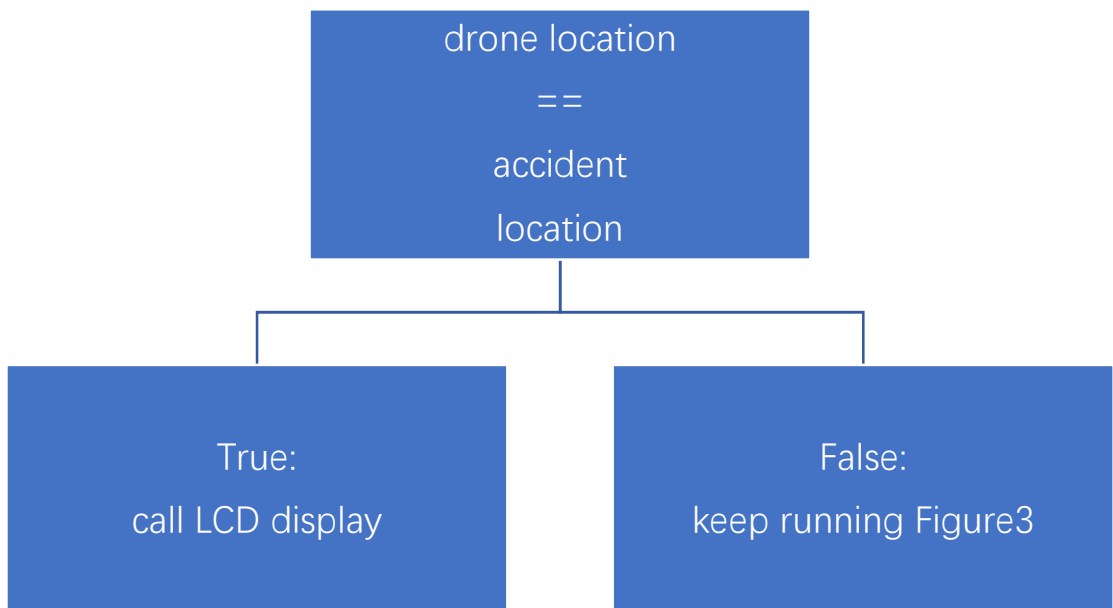
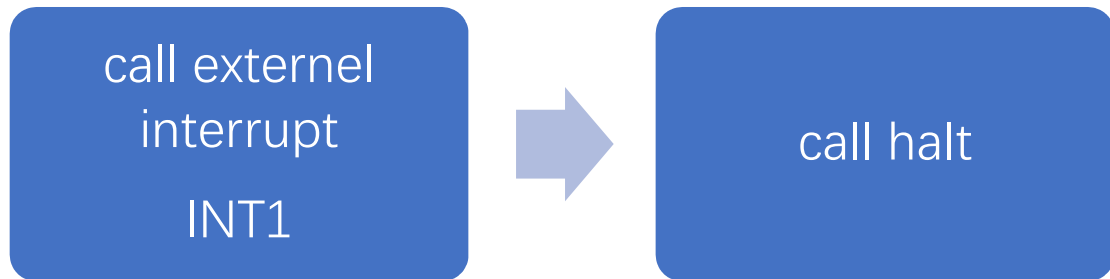


Figure 4: check if get the location or not

Search point by point. To be exactly, it check every point in the column, then check next column, until it finds the accident location or finish searching each point.

Figure 3 shows the procedure of searching, and Figure 4 illustrates the direction it goes after getting the searching result.

(4) Abort



When user press PB1, the program calls external interrupt, which calls halt to stop current operation.

4.2 Data structures and algorithm

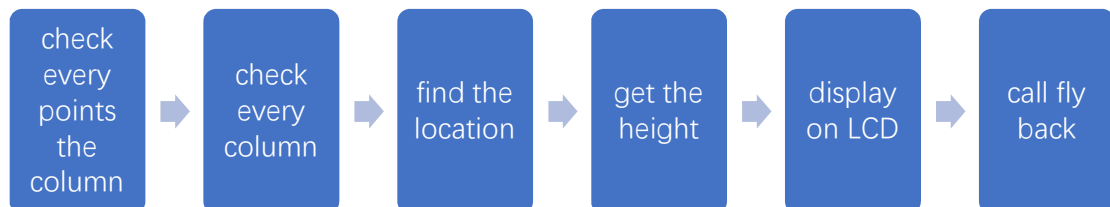


Figure 5: Searching Algorithm

The Figure 5 shows the searching algorithm, which is searching point by point.

I check the mountain from (0,0), then it checks the first column. After finish checking first column, the program goes to second column, and checks from (0,1). It keeps checking point by point until (64,64). Every time it checks, it also get the height of the mountain. If it finds the accident point, it will display it on the board.

5. Conclusion

During this Lab, I have learnt lots of knowledge about Microprocessors and I knows how to use AVR, which is really interesting. Besides, It is really fun to know how machine works when program starts. After finishing study this subject, I have already known what happens to memory, what happens to I/O devices, what is stack and so on. And these are all helpful to understand how computer works and following study on Information Technology. So I was really appreciate to choose this course.

Thanks for all the help from professor Guo, tutor Darshana and Mubashir . They are really patient, and helped me a lot on this subject.