

## Group Members :

- 陳紅宏 F74097065
- 雷美心 F74095314
- 葉惟欣 F74109016
- 黃佳倫 E64096245

## a. Description of System Functions and Principles

Population aging of labor force in agriculture is big issue in Taiwan. Digging, sowing, and watering are very tiring thing for farmers . In order to reduce the burden of farmer, Agricultural Robot is designed. Farmer can control machine to move forward, back ,turn around ,sow, water, light the plant ,plant by self-controller. Farmer can know when to supply water when farmer see the amount of water. Additionally, Farmer can know where the Agricultural Robot is when farmer see the location display.

For our final project, we built a farming machine. The machine have 6 functions;

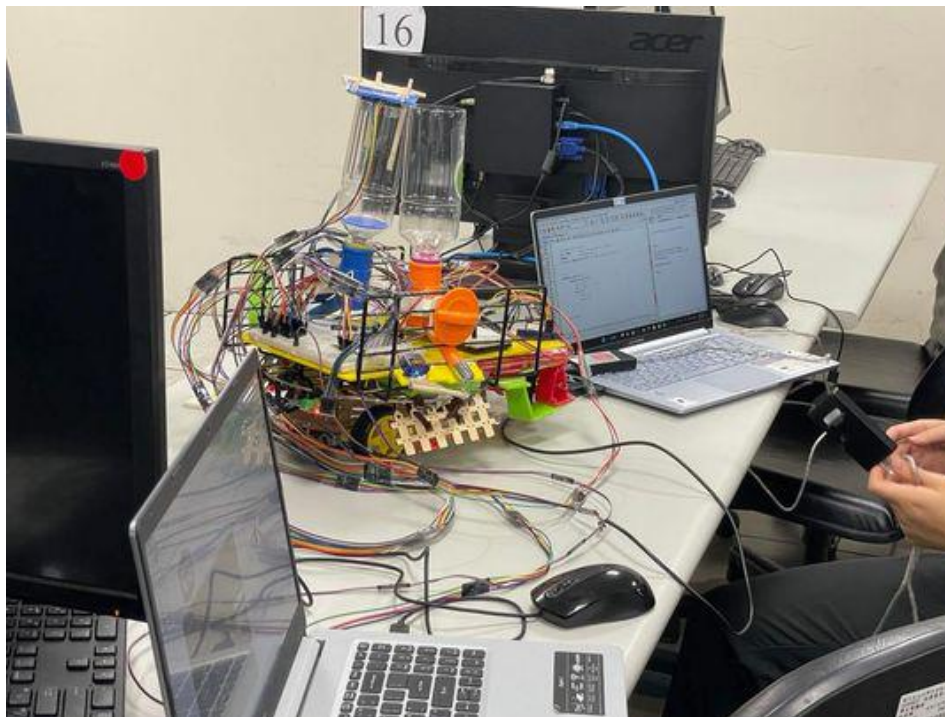
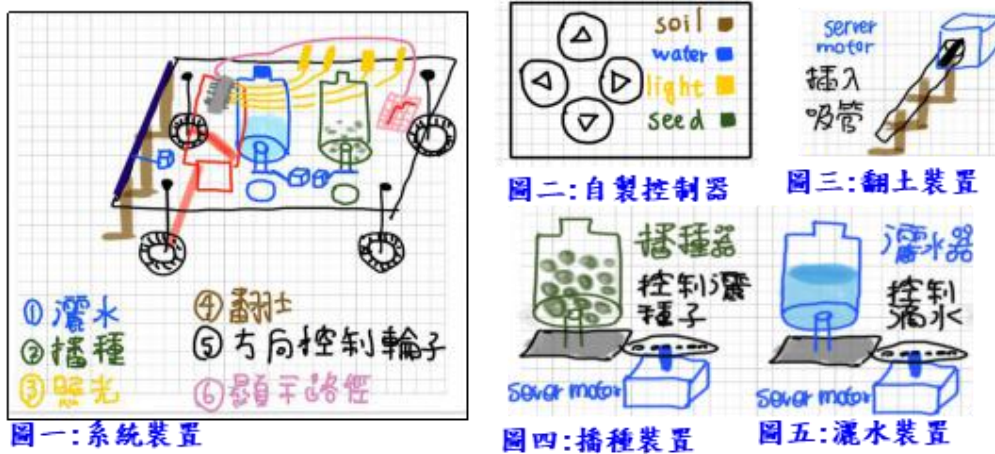
1. Watering, using a button and a servo motor to control the switch.
  - Press button to open the switch.
  - Press button one more time to close the switch.
2. Planting seeds, using a button and a servo motor to control the switch for releasing the seed.
  - Press button to open the switch.
  - Press button one more time to close the switch.
3. Lighting: using a button to control light up and off of the LED.
  - Press button turn on the LED.
  - Press button one more time to turn off the LED.
4. Digging the soil, using a DC motor for rotating the plow, and a button to control.
  - Press button to rotate the plow
5. Direction control, using DC motor to control only one side of the wheel to control the direction of the machine.
6. Automatically display the location of the robot on the 8\*8 dot matrix display.
7. Automatically display the amount of water the robot has.
8. The 1st to 5th can be controlled by the buttons on the self-made controller, and the 6th is automatically displayed

## b. System Usage Environment and Objects

- Environment: MPLAB X IDE
- Language: XC8
- Object: PicKit 4

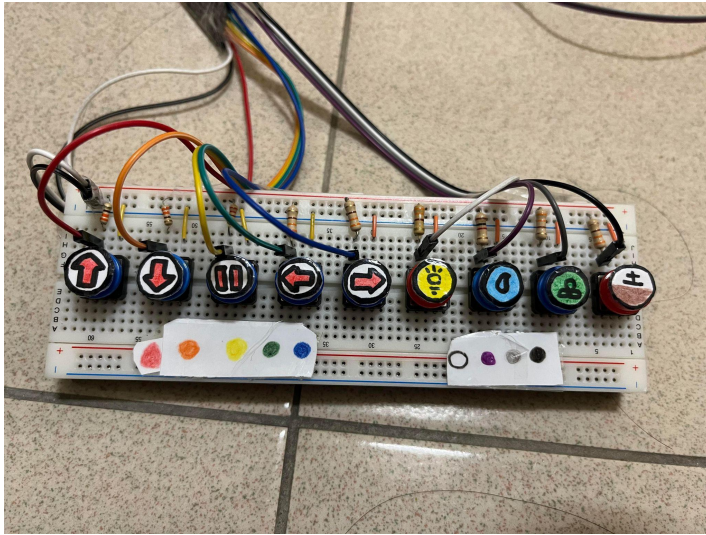
## c. Complete System Architecture Diagram, Flowchart, Circuit diagram, Design

### 1. System Architecture Diagram



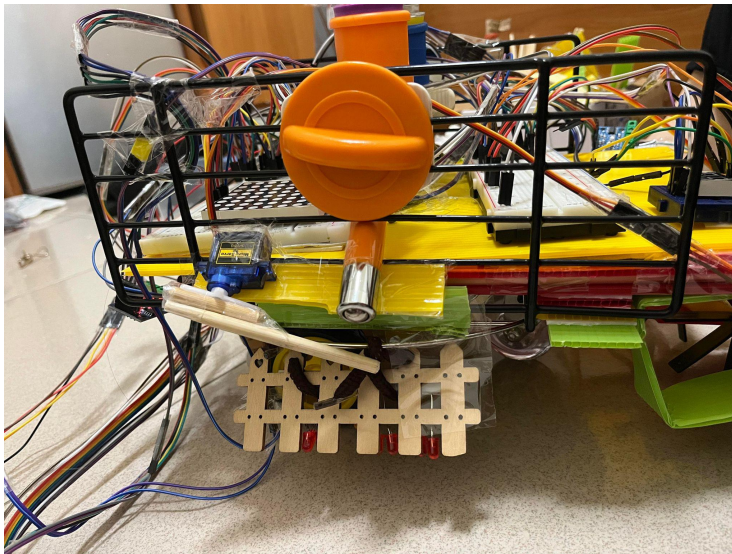
Demo video : <https://youtu.be/HZmYcK4nnb8>

## 1.1 self made controller



---

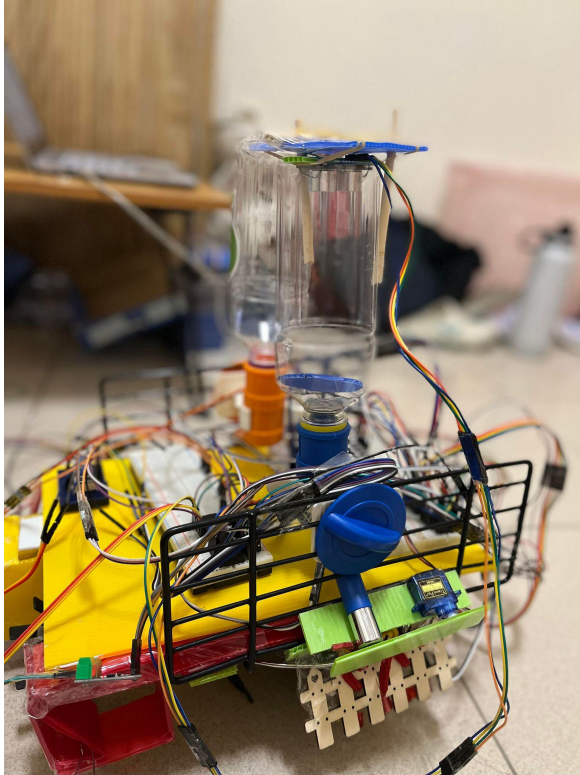
## 1.2 Watering machine



Demo video : <https://youtu.be/vNmy5nwkuUk>



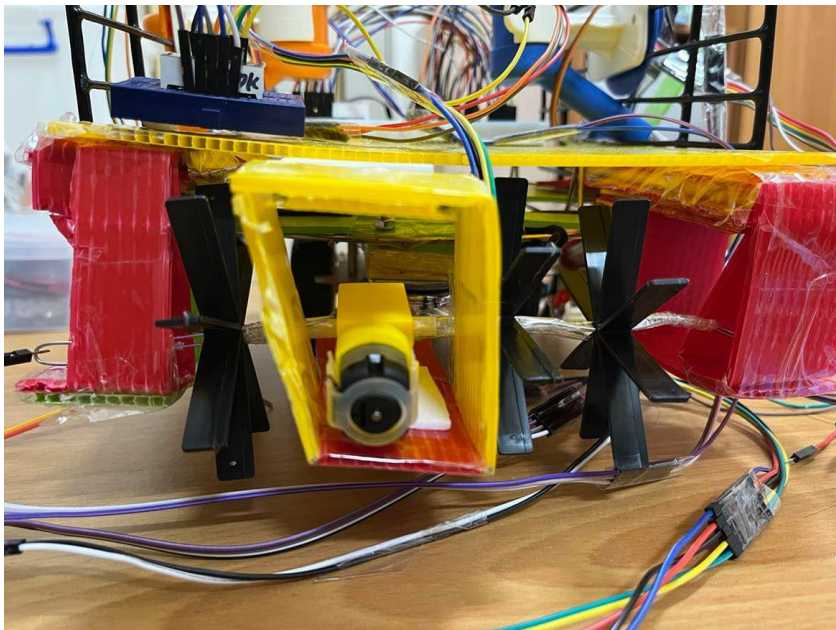
### 1.3 Plant Machine



Demo video : [https://youtu.be/Qj\\_ujI7H98E](https://youtu.be/Qj_ujI7H98E)

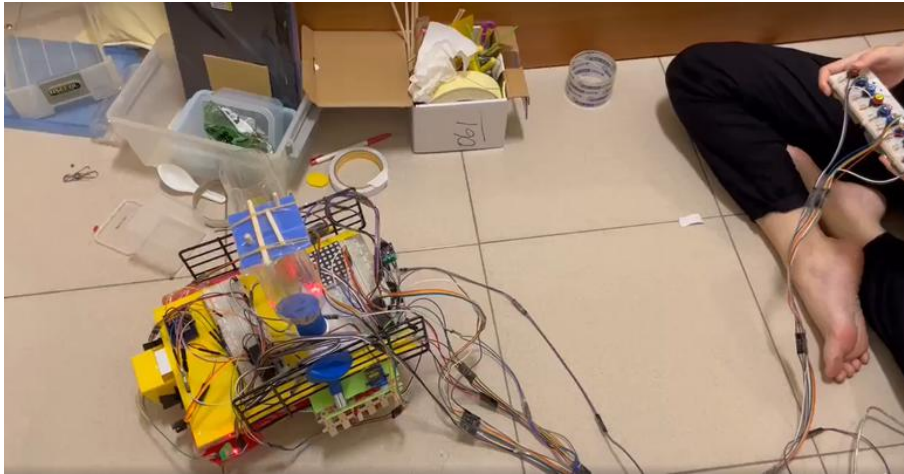
---

### 1.4 Digging the soil



Demo Video : [https://youtu.be/v89\\_Q-JRsBY](https://youtu.be/v89_Q-JRsBY)

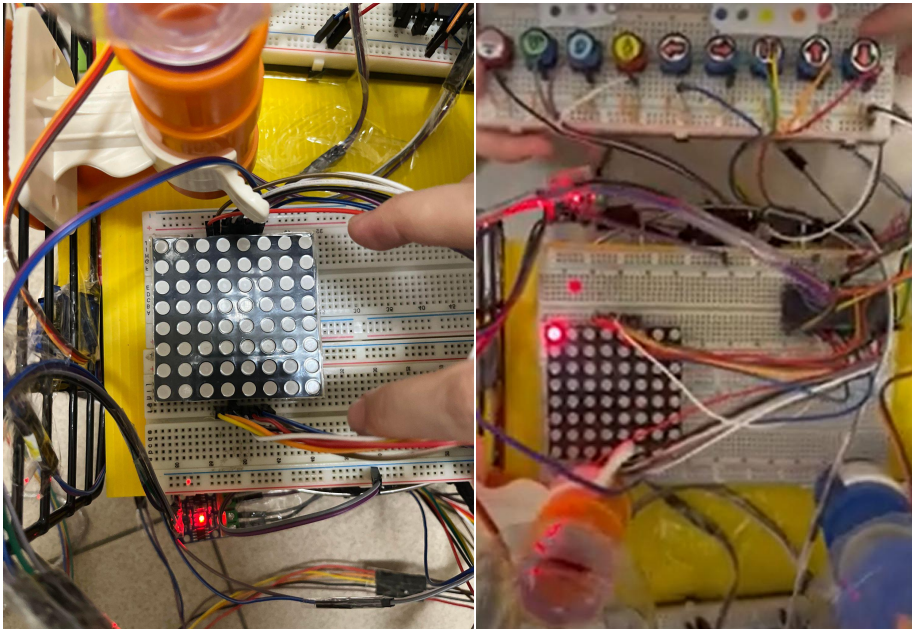
### 1.5 Robot moving control



Demo Video : [https://youtu.be/W0g9xHhyw\\_U](https://youtu.be/W0g9xHhyw_U)

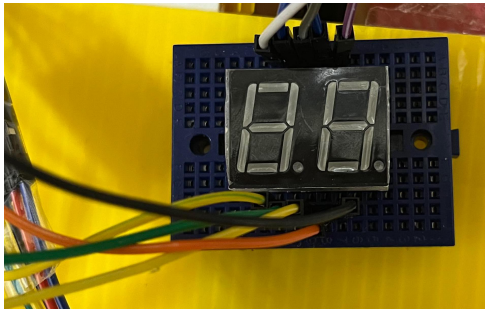
---

### 1.6 Location display on 8\*8 dot matrix



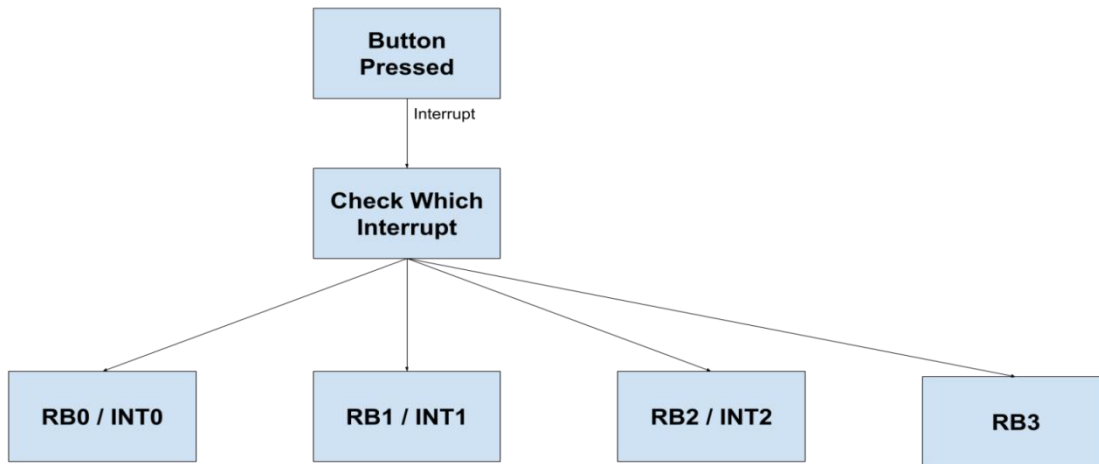
Demo video : <https://youtu.be/jXSN8A6g8iw>

### 1.7 the amount of water show on 7-Segment display

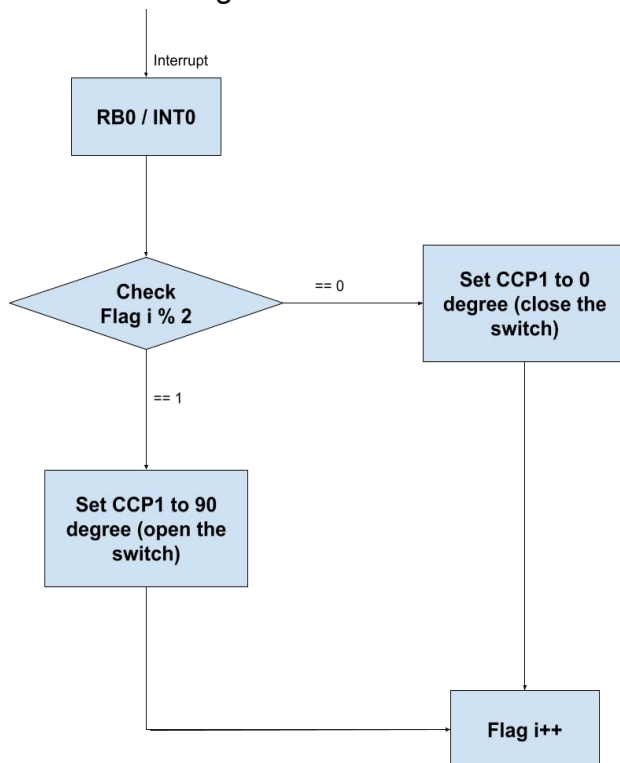


## 2. Flowchart

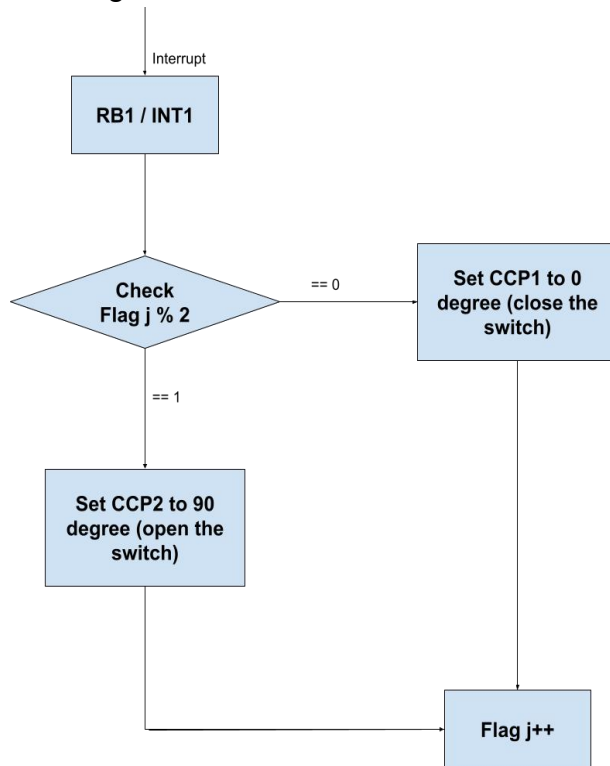
- Check which button is pressed (watering, planting seed, or lights)



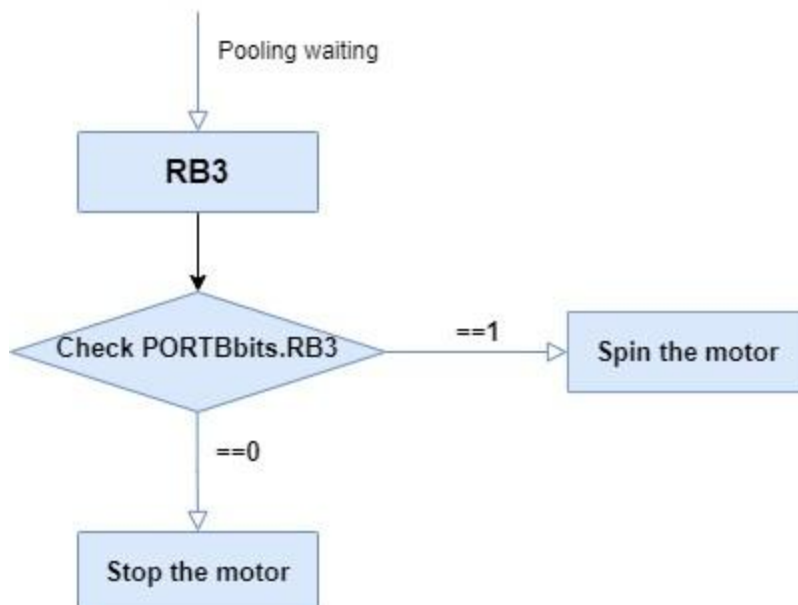
- Watering Flow Chart



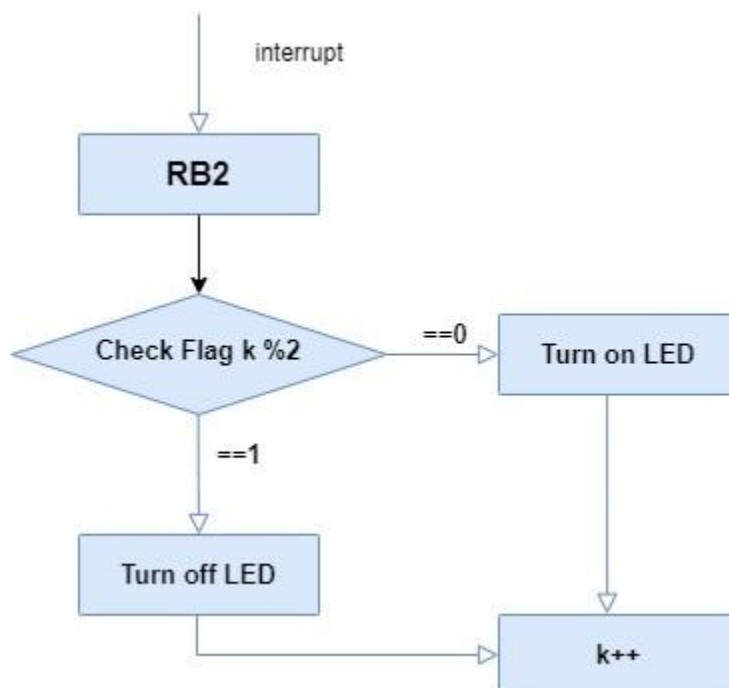
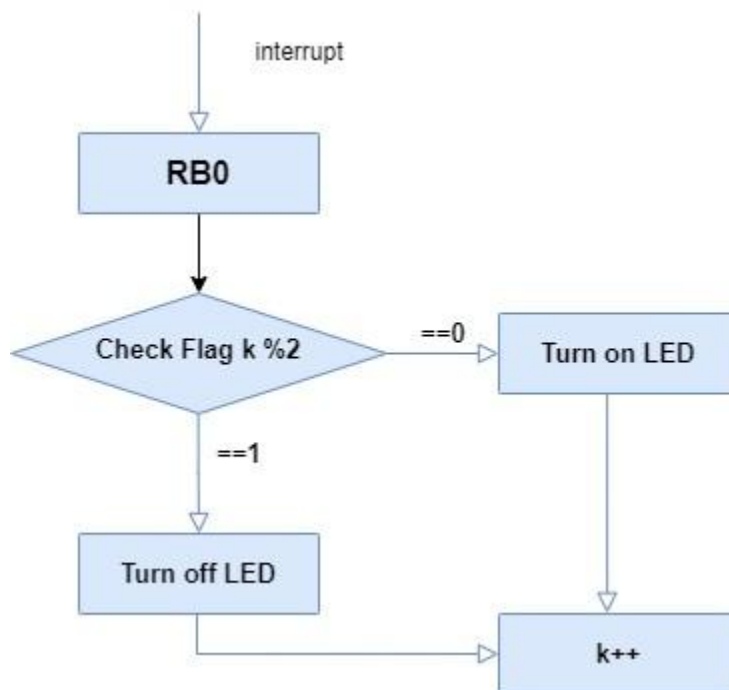
- Planting the seed Flow Chart



- Digging the soil Flow Chart

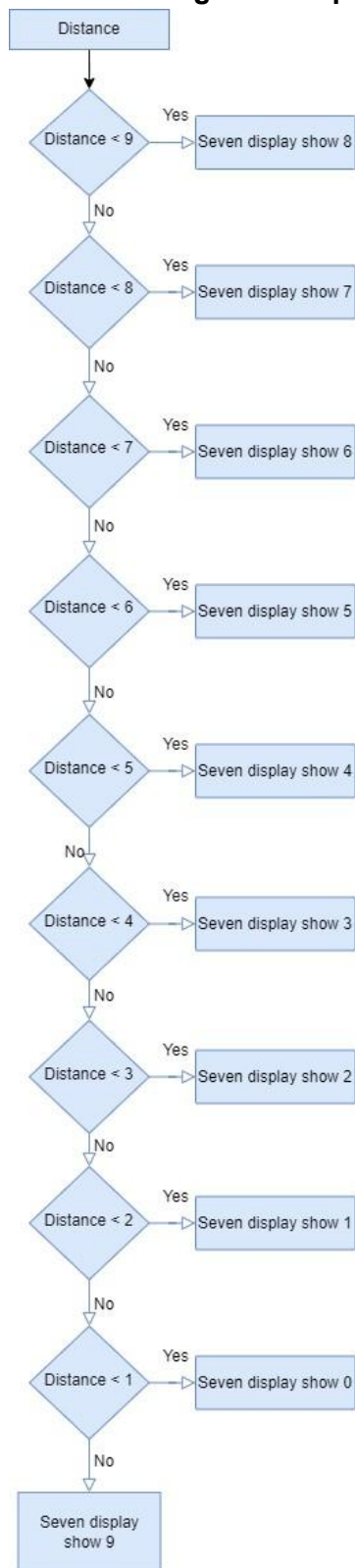


- Lighting (LED) Flow Chart

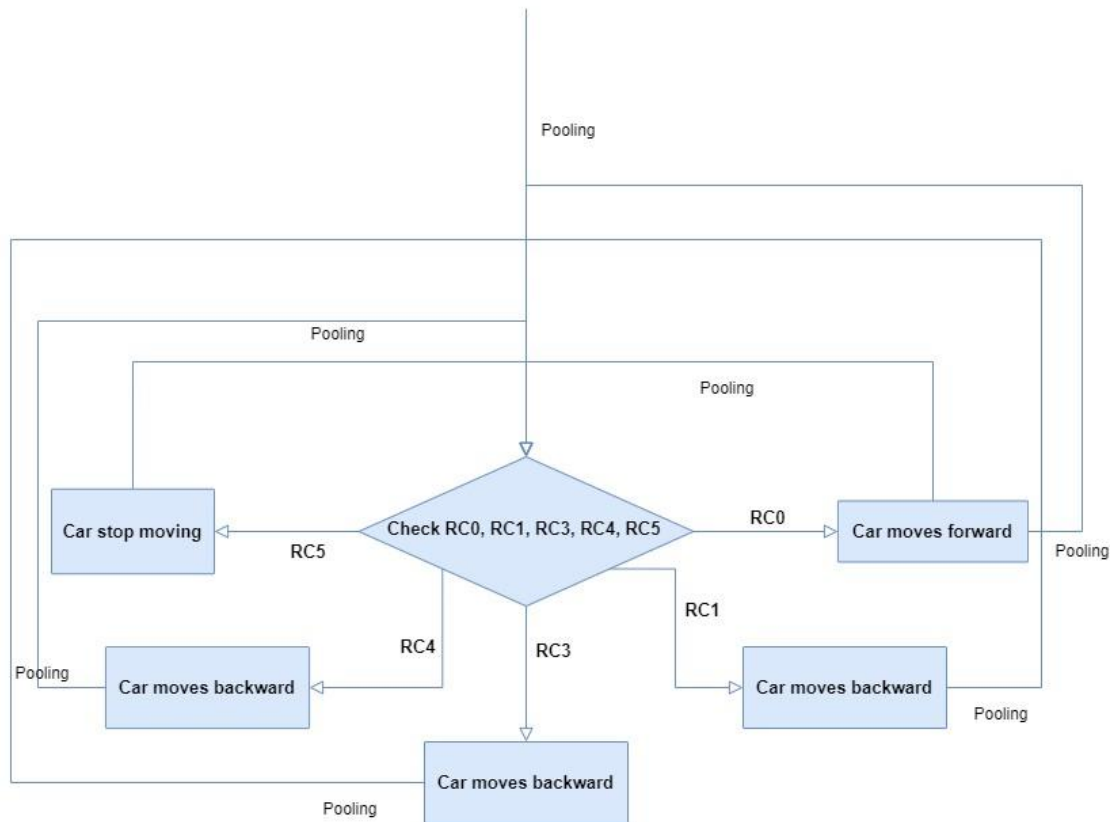




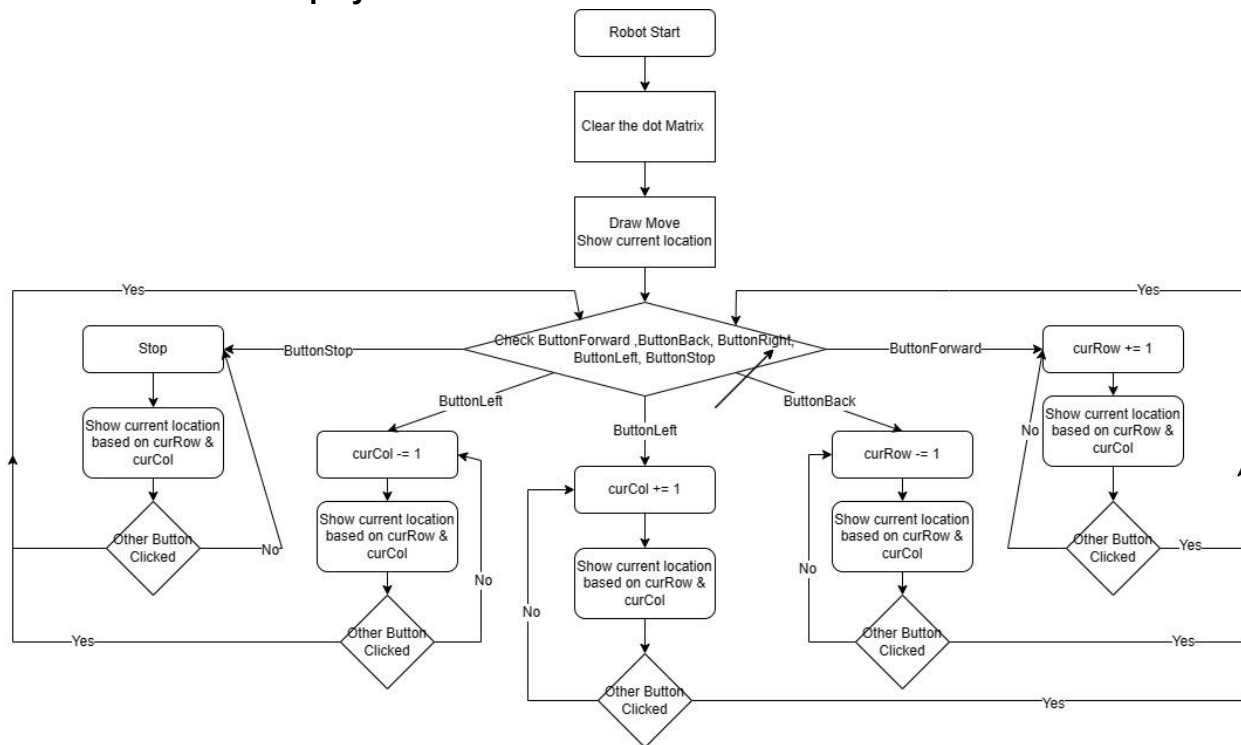
## ● Seven-Segment Display Flow Chart



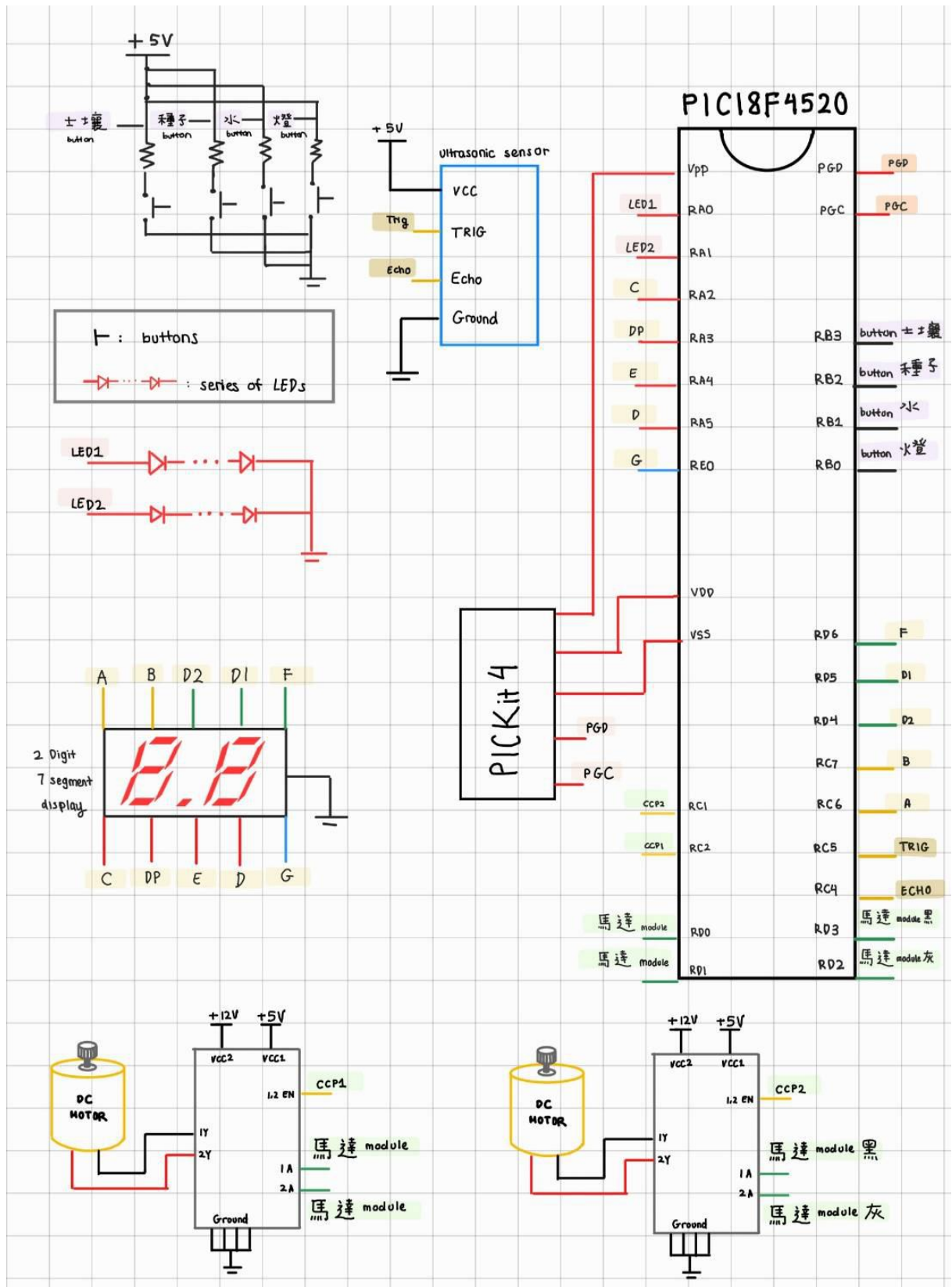
- **Moving Direction Control Flow Chart**



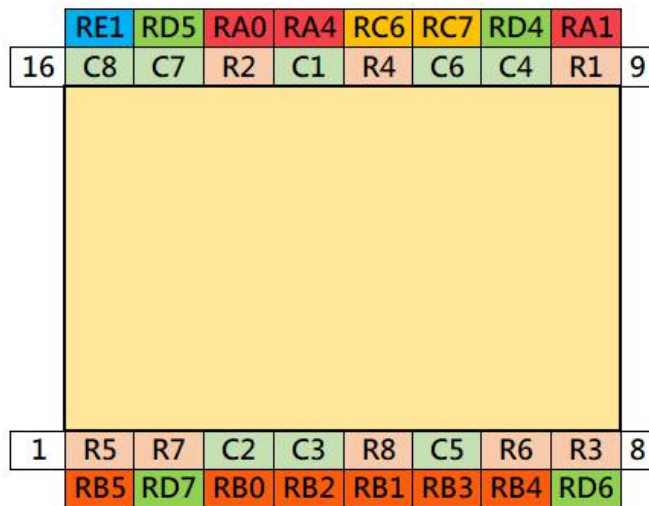
- **8\*8 Dot Matrix Display Location Flow Chart**



- Circuits:



8\*8 dot matrix for location display in main.c



8\*8 dot matrix (set row and column pin) tell the farmer where the agriculture robot is.

Reference : Interfacing 8×8 LED Matrix

[https://www.circuitstoday.com/interfacing-8x8-led-matrix-with-arduino?fbclid=IwAR1Dwr7snl5L6WABfz\\_edGxfDH6wiEoKvDJPkcil9DBSWRlvzOQmyx\\_M33Q](https://www.circuitstoday.com/interfacing-8x8-led-matrix-with-arduino?fbclid=IwAR1Dwr7snl5L6WABfz_edGxfDH6wiEoKvDJPkcil9DBSWRlvzOQmyx_M33Q)

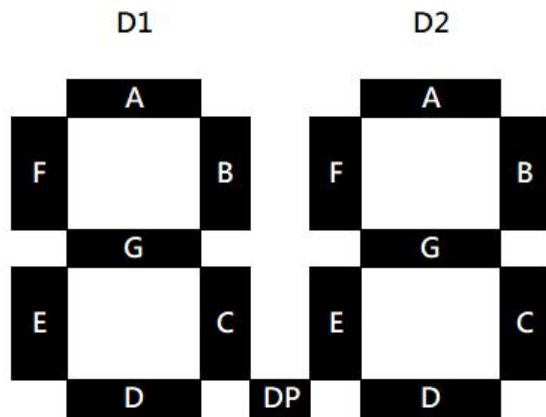
PIC18F4520 circuit pin setting in main.c

blue		1	40	RB7	yellow
R2	RA0	2	39	RB6	green
R1	RA1	3	38	RB5	R5
	RA2	4	37	RB4	R6
	RA3	5	36	RB3	C5
C1	RA4	6	35	RB2	C3
	RA5	7	34	RB1	R8
	RE0	8	33	RB0	C2
C8	RE1	9	32	VDD	
	RE2	10	31	VSS	
Red	VDD	11	30	RD7	R7
Black	VSS	12	29	RD6	R3
	RA7	13	28	RD5	C7
	RA6	14	27	RD4	C4
buttom Red	RC0	15	26	RC7	C6
buttom orange	RC1	16	25	RC6	R4
	RC2	17	24	RC5	buttom yellow
buttom green	RC3	18	23	RC4	buttom blue
motor module purple	RD0	19	22	RD3	motor module black
motor module blue	RD1	20	21	RD2	motor module gray



## PIC18F4520 function.c pin setting

seven-segment display  
show the amount of water in function.c



Seven-segment display the amount of water to tell the farmer when to supply water.

PIC18F4520 circuit pin setting in function.c

blue		1	40	RB7	yellow
LED (Output)	RA0	2	39	RB6	green
LED (Output)	RA1	3	38	RB5	
C (7-segment display)	RA2	4	37	RB4	
DP (7-segment display)	RA3	5	36	RB3	button (soil)(polling I/O)
E (7-segment display)	RA4	6	35	RB2	button (plant)(interrupt I/O)
D (7-segment display)	RA5	7	34	RB1	button (water)(interrupt I/O)
G (7-segment display)	RE0	8	33	RB0	button (LED)(interrupt I/O)
	RE1	9	32	VDD	
	RE2	10	31	VSS	
Red	VDD	11	30	RD7	
Black	VSS	12	29	RD6	F (7-segment display)
	RA7	13	28	RD5	D1 (7-segment display)
	RA6	14	27	RD4	D2 (7-segment display)
	RC0	15	26	RC7	B (7-segment display)
CCP2	RC1	16	25	RC6	A (7-segment display)
CCP1	RC2	17	24	RC5	TRIG
	RC3	18	23	RC4	ECHO
motor module	RD0	19	22	RD3	motor module black
motor module	RD1	20	21	RD2	motor module gray

## PIC18F4520 function.c pin setting

#### d. System Development Tools, Materials and Technologies

- LED
- Servo motor
- DC motor
- Button
- Pickit 4
- PIC18F4520
- Bread board
- 8\*8 dot matrix
- 7-segment display

#### e. Peripheral interface or Library and API instructions

- Pic18f4520.h
- xc.h

#### f. The division of labor of the actual team members

項目	單元項目	負責組員
1.灑水、2.播種、4.翻土：開關馬達	PWM, Interrupt	雷美心、陳紅宏
3.光照	Interrupt	黃佳倫、葉惟欣
5.輪子驅動：dc motor 轉彎與前進	Interrupt	
6.路徑顯示：dot display	Interrupt, Timer	
7.晶片整合(兩顆 pic18f4520)		全員

#### g. Difficulties encountered and how to solve them

One of the difficulties we face is to interface the buttons with the functions. We need a lot of buttons for different functions, and at first, when we set the function to run if  $\text{PORTx.RBx} = 1$  and if  $\text{INT0IF} = 1$ , it doesn't work.

In the end, we used interrupts INT0, INT1, and INT2 for each button. After we set the function to run if  $\text{INTxIF} = 1$ , the function will run. Because we could only use two PICKit 4, each PIC only has 3 interrupts, we had to limit the usage of our buttons.