2022 微算機期末專題文件

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 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;

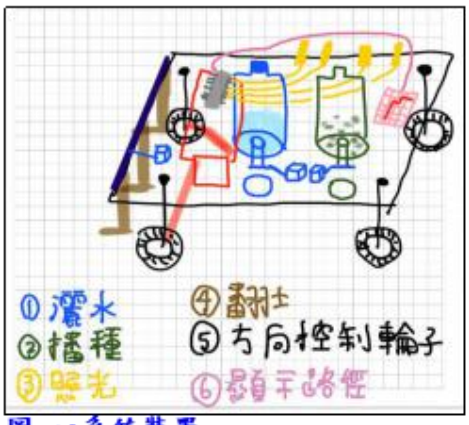
1. Watering, using a button and a servo motor to control the switch.
    - Press button to open the switch.
    - Press button one more time to close the switch.
2. Planting seeds, using a button and a servo motor to control the switch for releasing the seed.
    - Press button to open the switch.
    - 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

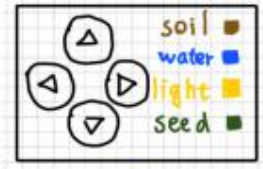
- Environment: MPLAB X IDE
- Language: XC8
- Object: PicKit 4

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

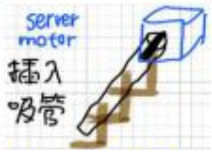
1. System Architecture Diagram



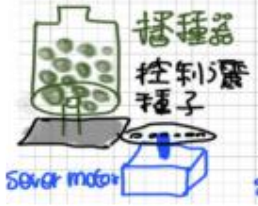
圖一：系統裝置



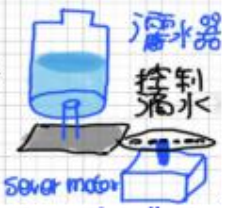
圖二：自製控制器



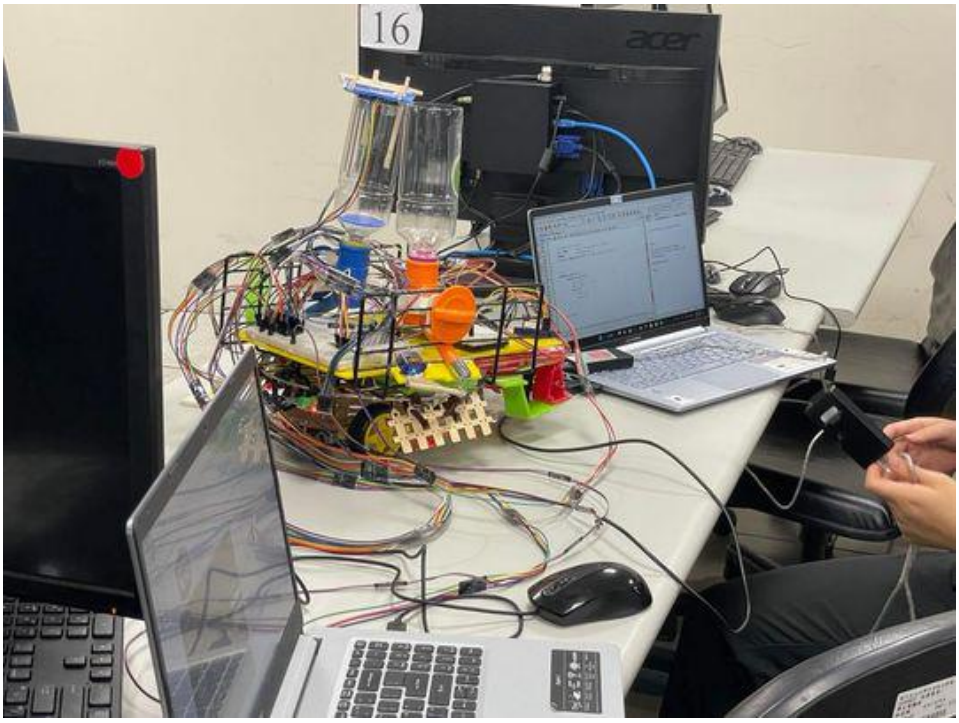
圖三：翻土裝置



圖四：播種裝置

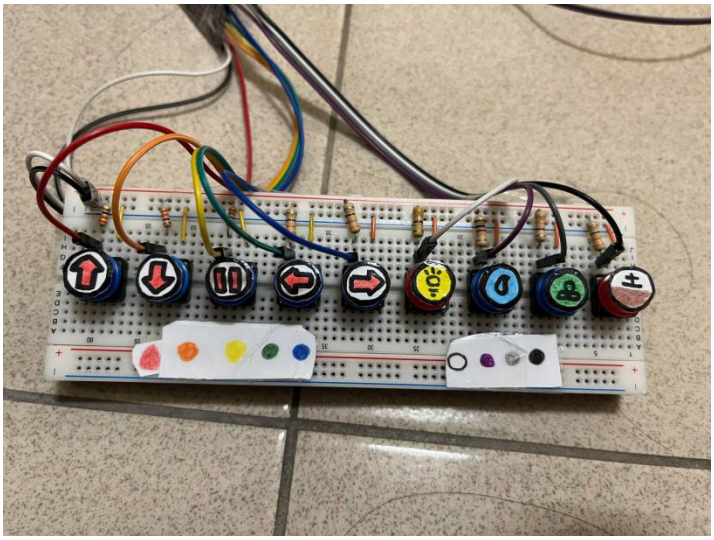


圖五：灑水裝置

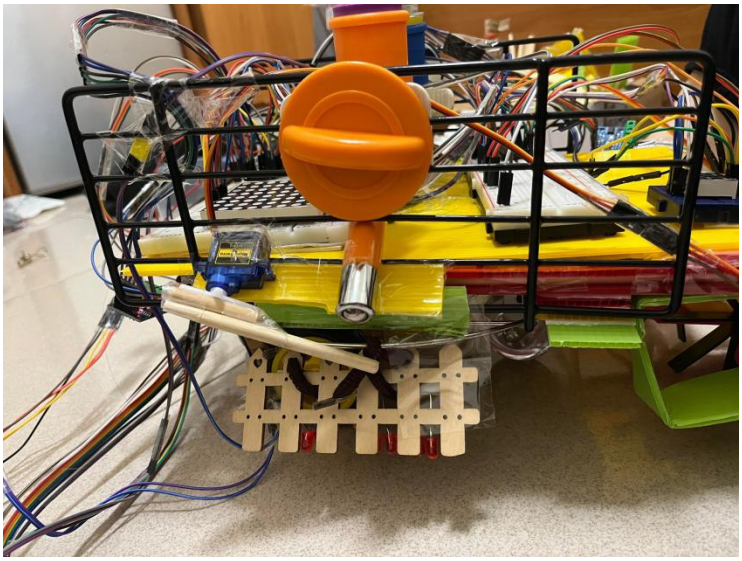


Demo video : https://youtu.be/HZmYcK4nnb8

1.1 self made controller

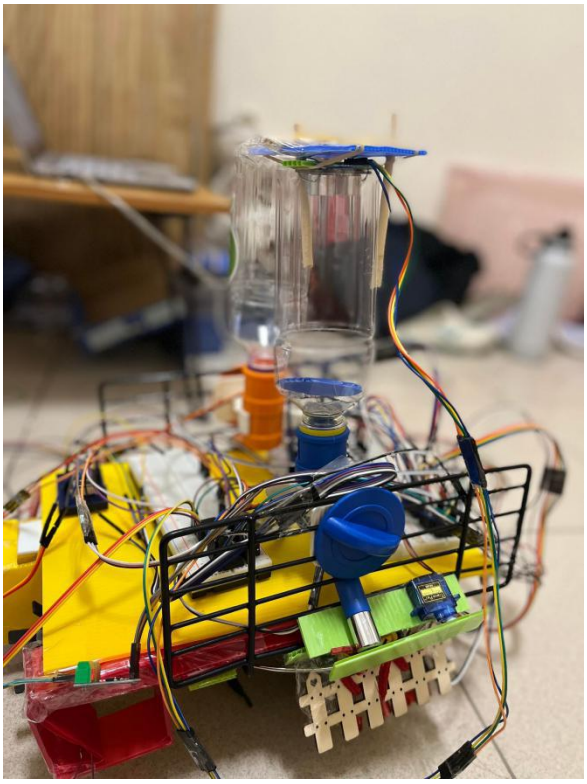


1.2 Watering machine



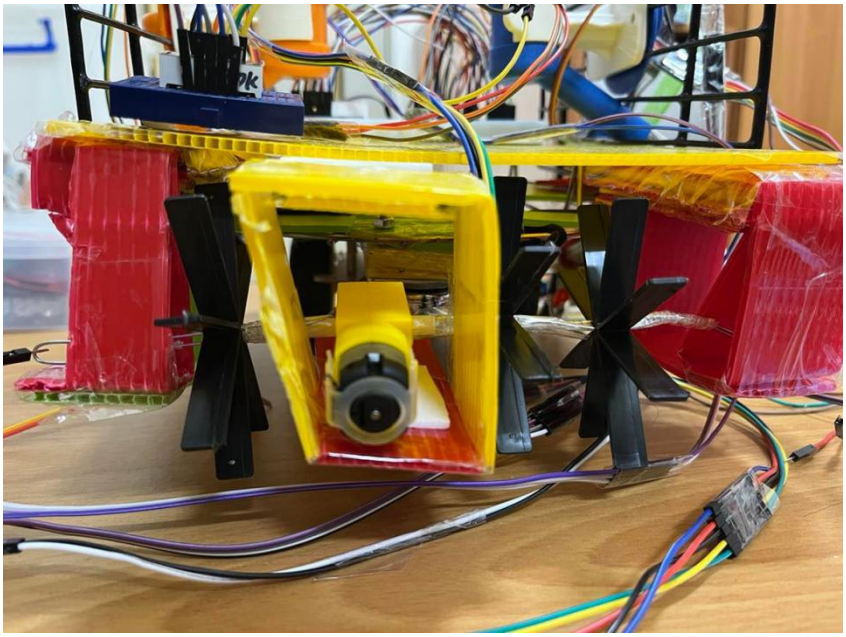
Demo video : https://youtu.be/vNmy5nwkuUk

1.3 Plant Machine



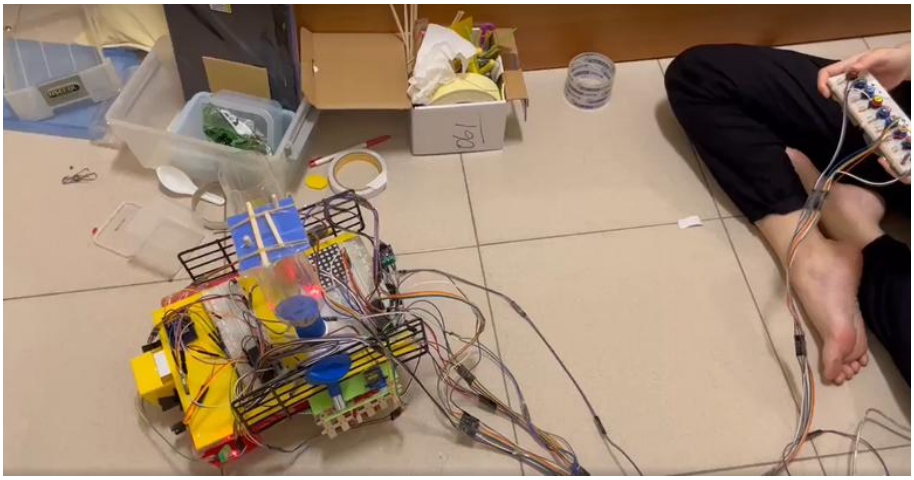
Demo video : https://youtu.be/Qj_ujl7H98E

1.4 Digging the soil



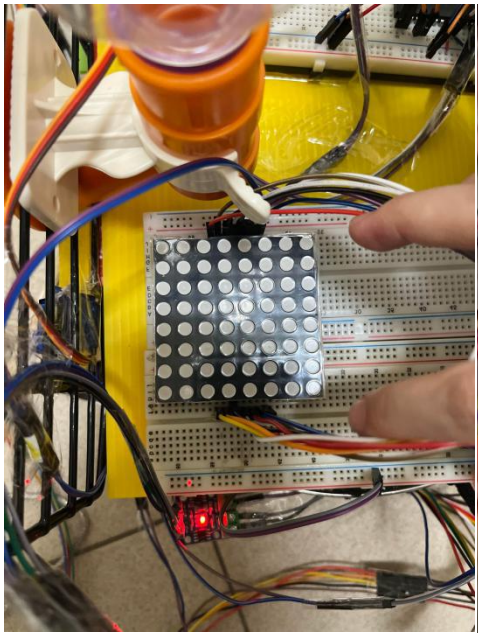
Demo Video : https://youtu.be/v89_Q-JRsBY

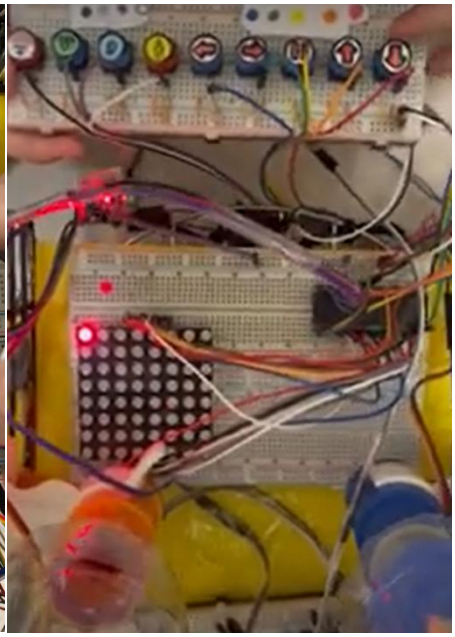
1.5 Robot moving control



Demo Video : https://youtu.be/W0g9xHhyw_U

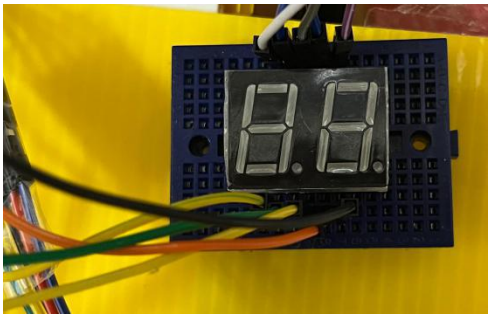
1.6 Location display on 8*8 dot matrix





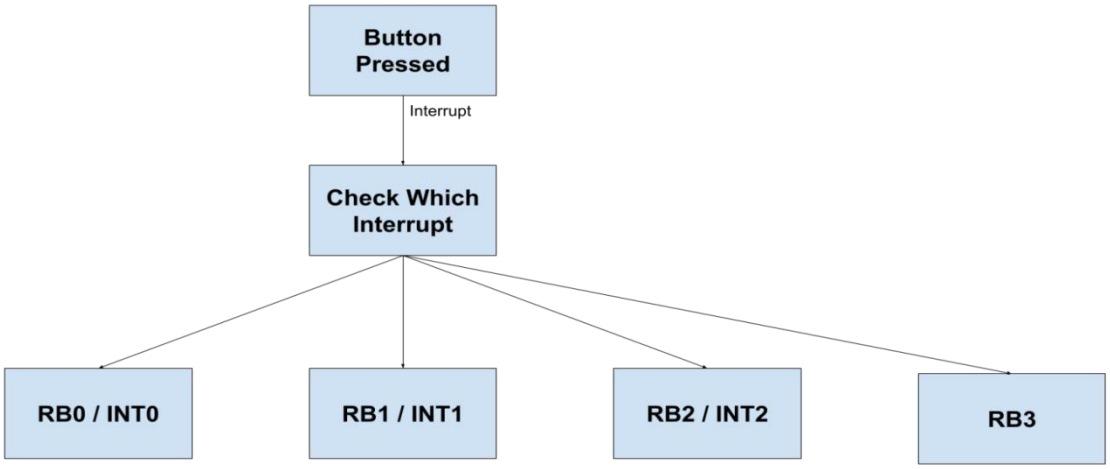
Demo video : https://youtu.be/jXSN8A6g8iw

1.7 the amount of water show on 7-Segment display

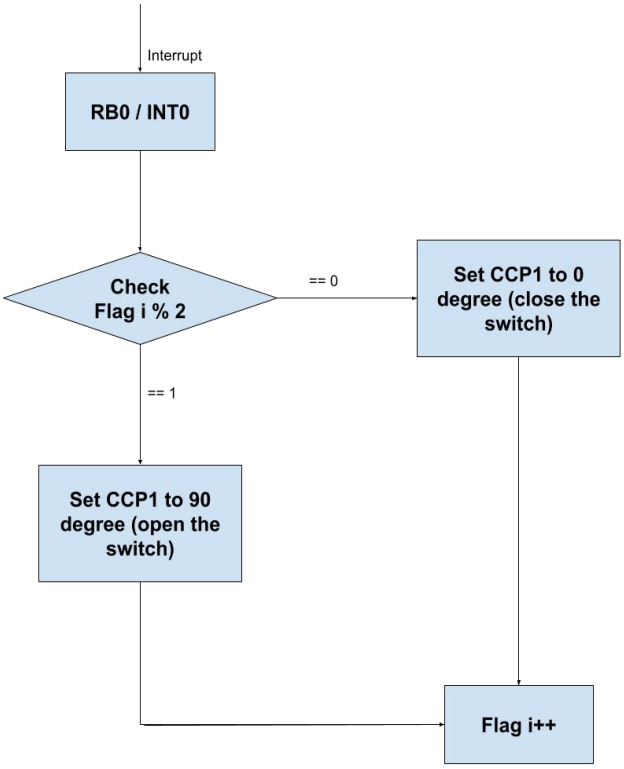


2. Flowchart

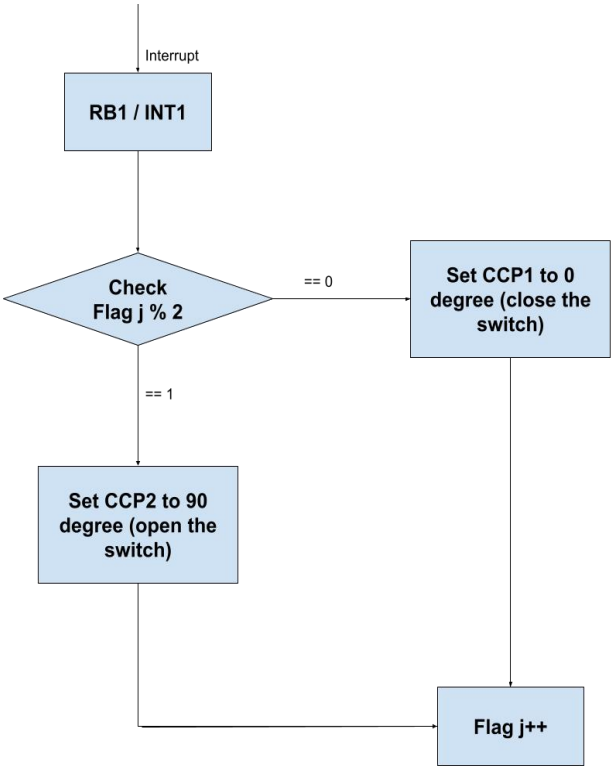
- Check which button is pressed (watering, planting seed, or lights)



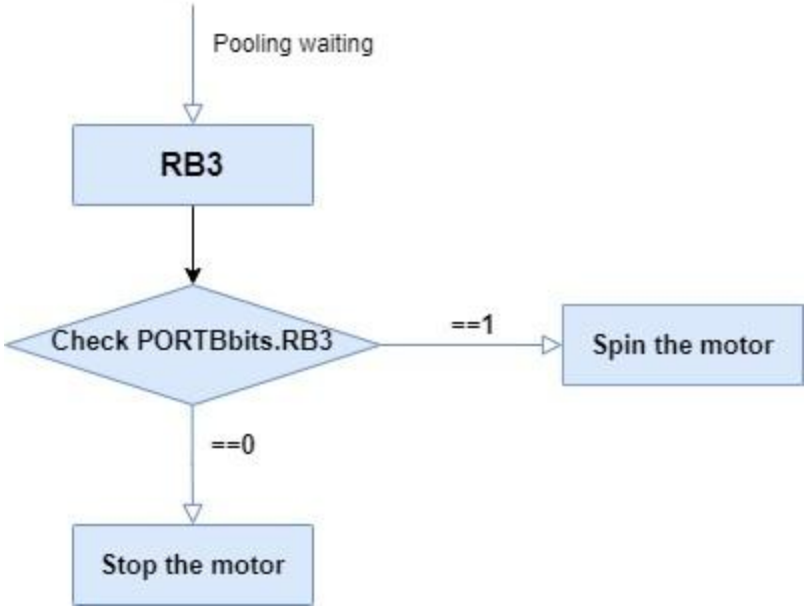
- Watering Flow Chart



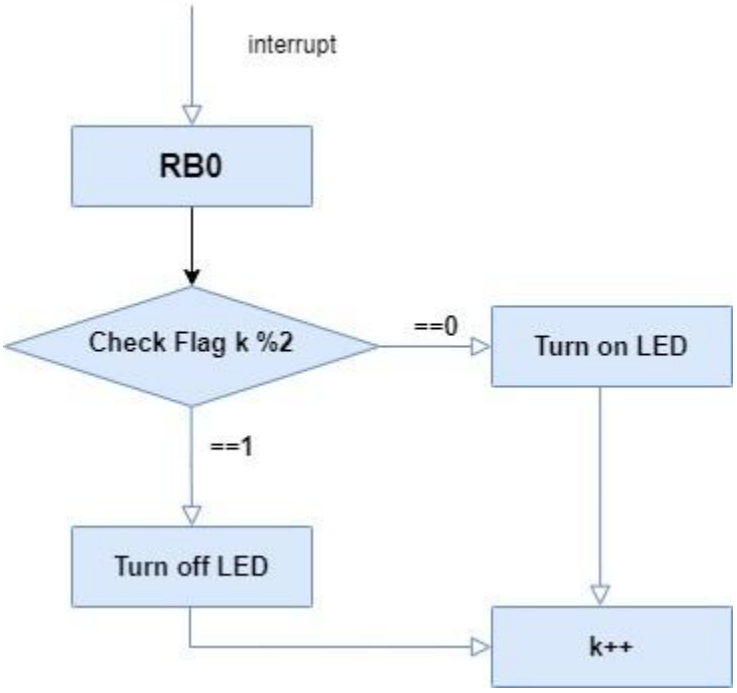
- Planting the seed Flow Chart

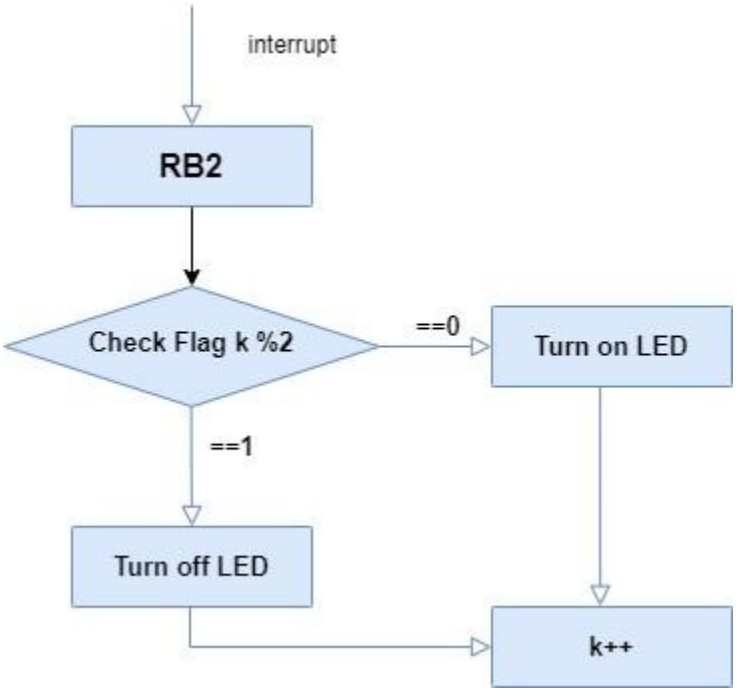


- Digging the soil Flow Chart

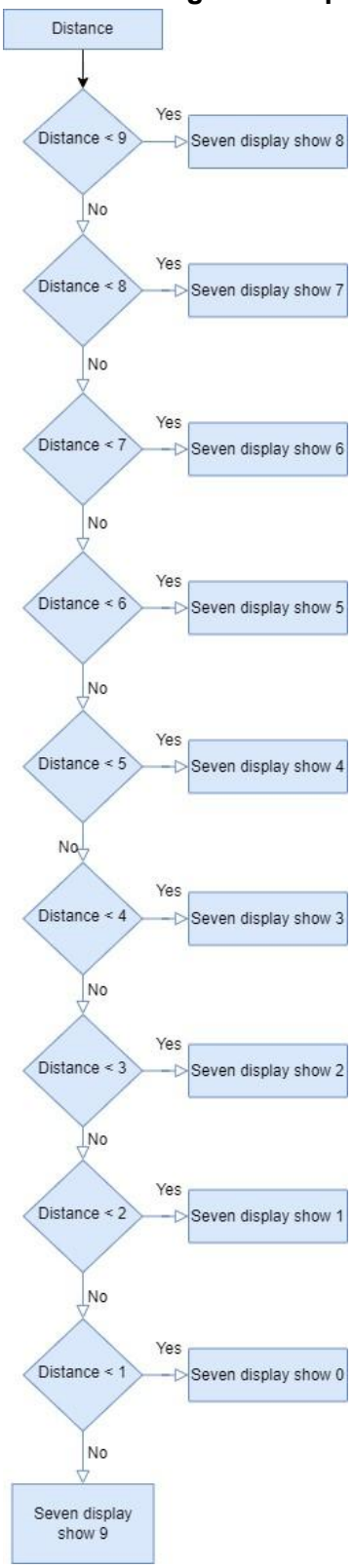


- Lighting (LED) Flow Chart

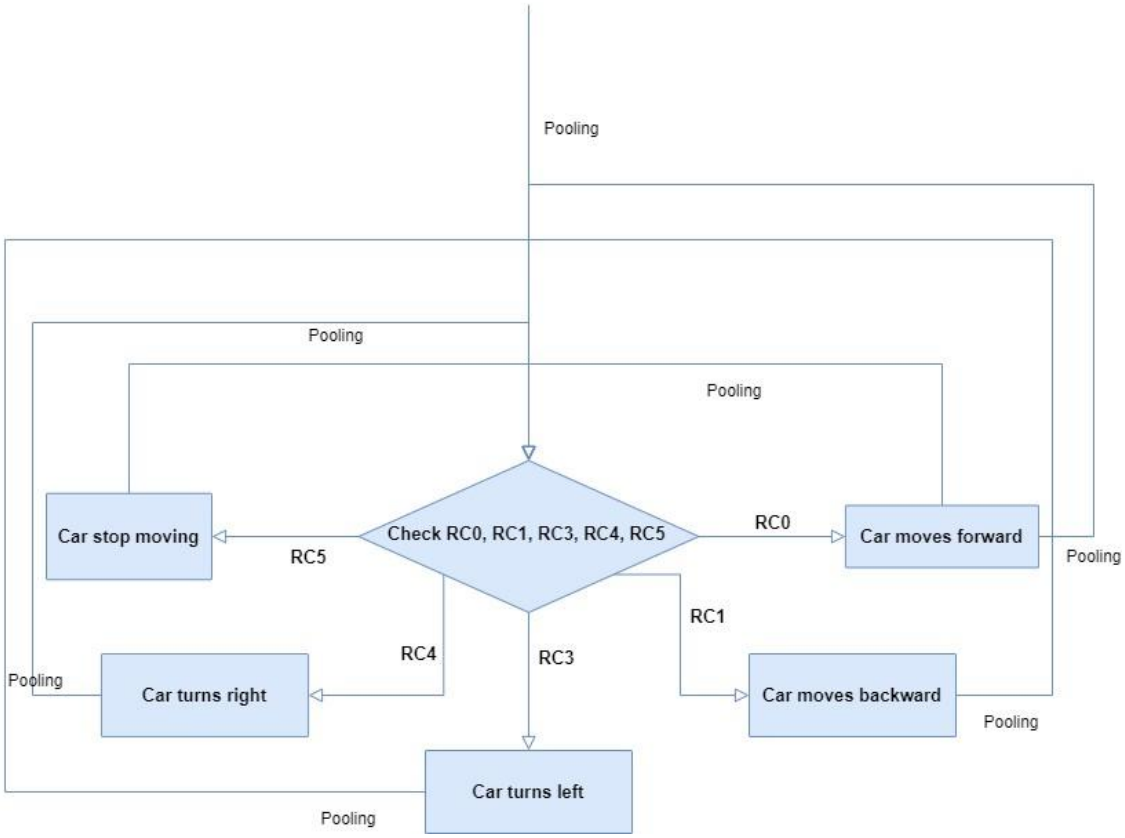




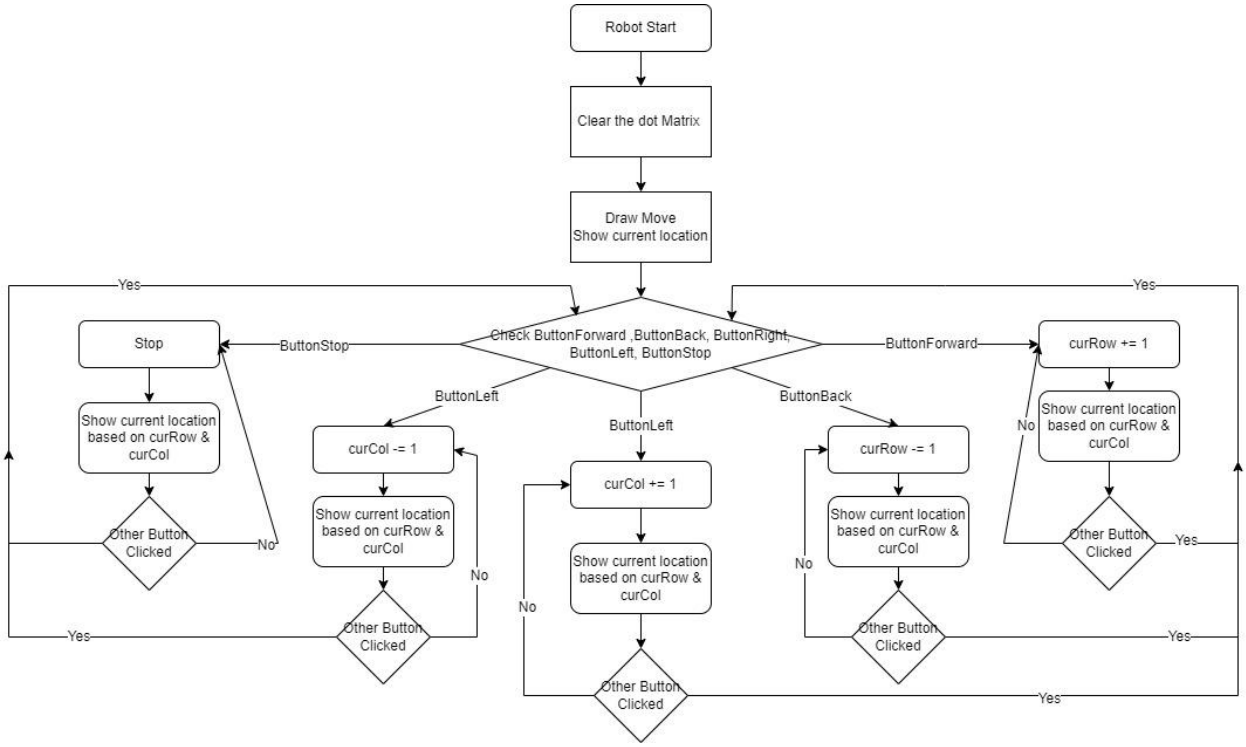
- **Seven-Segment Display Flow Chart**



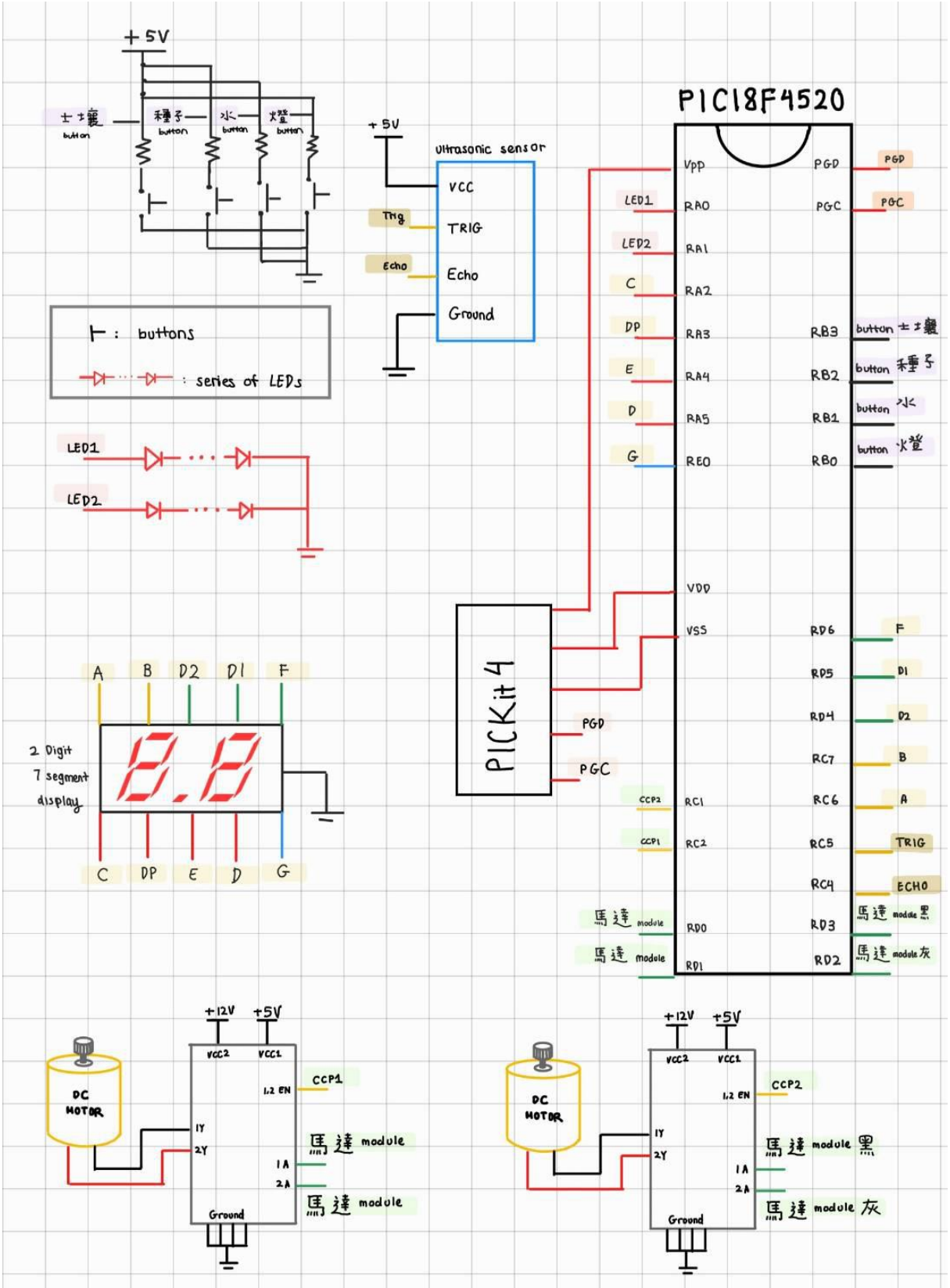
- **Moving Direction Control Flow Chart**



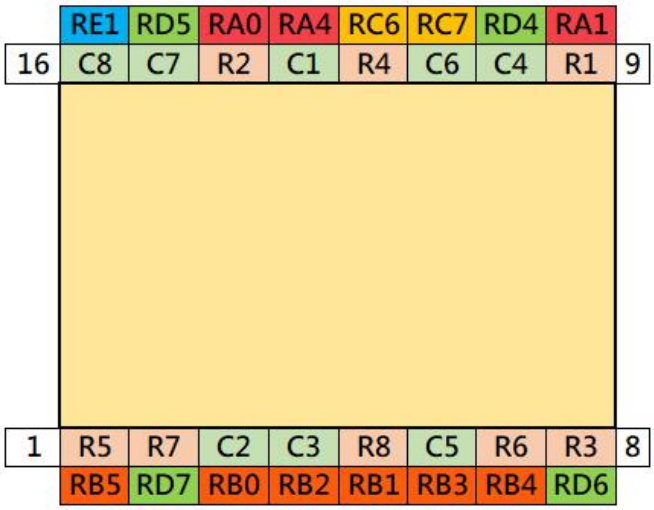
- **8*8 Dot Matrix Display Location Flow Chart**



- Circuits:


+5V
button
種子 button
水 button
燈 button
+5V
Ultrasonic sensor
VCC
Trig
TRIG
Echo
Echo
Ground
PIC18F4520
Vpp
PGD
PGC
LED1
RA0
LED2
RA1
C
RA2
DP
RA3
E
RA4
D
RA5
G
RE0
RB3
RB2
RB1
RB0
button 土壤
button 種子
button 水
button 燈
: buttons
: series of LEDs
LED1
LED2
VDD
VSS
PICKit 4
PGD
PGC
RD6
RD5
RD4
RC7
RC6
RC5
RC4
RD3
RD2
F
D1
D2
B
A
TRIG
ECHO
馬達 module 黑
馬達 module 灰
A
B
D2
D1
F
2 Digit
7 segment
display
C
DP
E
D
G
CCP2
RC1
CCP1
RC2
馬達 module
RD0
馬達 module
RD1
+12V
+5V
VCC2
VCC1
1,2 EN
CCP1
DC MOTOR
1Y
2Y
馬達 module
1A
2A
馬達 module
Ground
+12V
+5V
VCC2
VCC1
1,2 EN
CCP2
DC MOTOR
1Y
2Y
馬達 module 黑
1A
2A
馬達 module 灰
Ground

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.

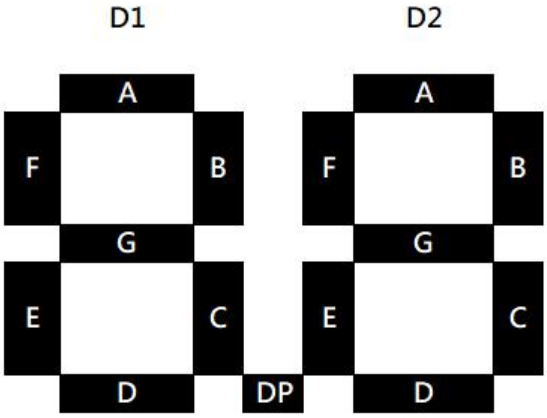
Reference : Interfacing 8×8 LED Matrix
https://www.circuitstoday.com/interfacing-8x8-led-matrix-with-arduino?fbclid=IwAR1Dwr7snI5L6WABfz_edGxfDH6wiEoKvDJpHciI9DBSWRIvzOQmyx_M33Q

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	buttom (soil)(polling I/O)
E (7-segment display)	RA4	6	35	RB2	buttom (plant)(interrupt I/O)
D (7-segment display)	RA5	7	34	RB1	buttom (water)(interrupt I/O)
G (7-segment display)	RE0	8	33	RB0	buttom (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
- L9110

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 (開關sever motor)	PWM,Interrupt I/O	雷美心、陳紅宏
2. 播種 (開關sever motor)		
3. 光照		
4. 翻土 (開關DC motor)	Polling I/O	葉惟欣、黃加侖
5. 輪子驅動 : DC motor 轉彎與前進		
6. 路徑顯示	Dot Matrix Display	
7. 水量顯示	7-segment display 超聲波距離偵測	黃加侖
8. 晶片整合	兩顆PIC18F4520	葉惟欣、黃加侖
9. Document		全員

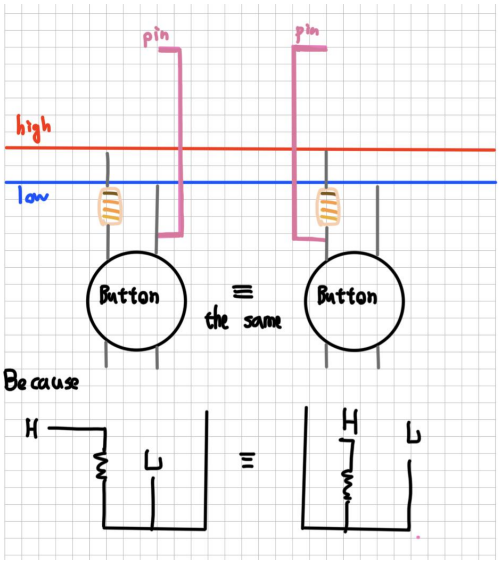
g. Difficulties encountered and how to solve them

1. Difficulties about function of watering , plant , lighting
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 PORTx.RBx = 1 and if INT0IF =1, it doesn’t work.

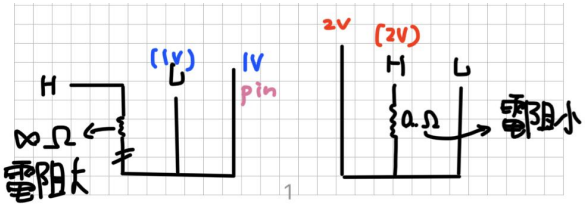
In the end, we used interrupts INT0, INT1, and INT2 for each button. After we set the function to run if INTxIF = 1, the function will run. Because we could only use two PICKit 4, each PIC only has 3 interrupts, we had to limit the usage of our buttons.

2. Difficulties about high/low voltage

當我們在寫 code 時候，我們以一般認知認為按鈕按下 1 ，則會給我們輸入端 high voltage。但神奇的是，經過我們一番努力的 debug 後，我們發現我們的按鈕按下，是給輸入端 low voltage。我原本以為是我們 button 的電路接法問題，而導致如此的結果。但後來發現其實這沒有差。



最後，我認為是電阻的問題。由以下圖來說明。



Case1 : 如果電阻很大的時候則當我按下 button 則，輸入到 pin 的電壓相當於 low (2V 被電阻吃掉剩趨近於 0V，0V 與 1V 並聯，所以輸出為 1V)。(如左圖)

Case2 : 如果電阻很小的時候則當我按下 button 則 ,輸入到 pin 的電壓相當於 high (2V 與 1V 並聯，所以輸出為 2V)。(如右圖)

而我們 button 有接電阻 ,因此我認為我們是 Case1 ,所以判斷 button 按下後的電位 check 是否==0。而平時我們認為的是未接電阻的 button(相當於 Case2 電阻很小)，所以判斷 button 按下後的電位 check 是否==1。

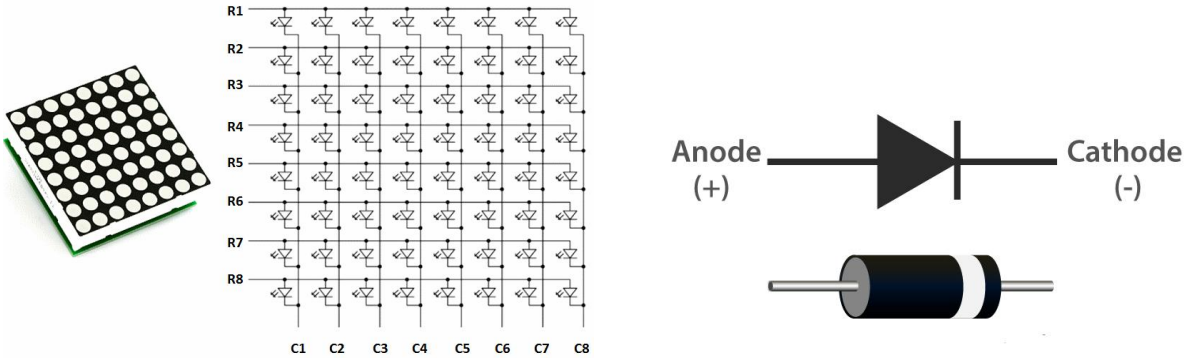
3. Difficulties about control direction

當我們在用馬達的時候邏輯基本上都正確，但在設定馬達的時候我們一直搞不清楚馬達的 delay 要設在哪裡，我們參考別人的 code，但用在我們的 code 上面卻執行的不是那麼順利。最後我們理清思緒，決定用 polling I/O 假如按下某個方向控制的 button 就可以進到相對應控制的 function，而在那個 function 中有一個無限迴圈，只會終止於按下其他 button 的事件。設一個無限迴圈的原因是因為，我們發現只給他左右馬達高低電位，他似乎只 run 了微乎其微的一瞬間 ,因此要設定讓他一直給他高低電位來讓馬達可以讓它移動。與此同時無限迴圈也要設定終止條件，此為其他 button 按下。如果此事件觸發，則將自己的 button = 1(根據困難 2 說明)。然後跳出此無限迴圈回到 main function 去檢查是哪個 button 再次被按下，再進到相對應控制方向的 function 中。

4. Difficulties about location display

一開始我們在設定 8*8 dot matrix，我以為 row 都是在一排，而 column 都是在一排，所以我們剛開始跑出來都是錯的。因為真正的 dot matrix 上面 8 個 pin 角，與下面 8 個 pin 角的設定都是交錯的。去查了之後才從新更正。

第二個困難的部份是我原先不知道 8*8 dot matrix 一個 row 要設定為 high column 要設定為 low，二極體才會流過，所以我們一定要有一個設為 1 一個設為 0，才會使想要的位置的燈泡發光。



第三個困難的部分是原先我們想要做到顯示路徑，但後來我思考過後我們只有顯示一個位置一個點才有可能實現，因為我們不是所有的 code 都在跑 dot matrix display ，而是有改變方向或經過幾秒才進去跑，如果要實現大範圍的點矩陣顯示，就要透過一排一排刷的去顯示，因為 dot matrix 只能實現一行 或 一列的顯示。而其中切換一排一排的間距時間是奈秒，因為人的肉眼看不到所以就認為他是同時顯示，但其實不竟然。

要實現大面積路徑顯示就要一排排刷，而一排排的效果就要所有的 code 都在 run 點矩陣，但我們是只有在改變路徑或到行經特定時間才會再做 location 定位的移動 ,所以我就改成只做 location 定位，而不是 record the path.

5. Difficulties about amount of water display