



Smart Door Security System With Audio Effect using Arduino 使用 Arduino 套件設計 智能多音效門禁系統

第二組

B812108030 陳婉安
B405110031 陳冠妤
B313109084 陳郁宣

目錄

01

專案目標

02

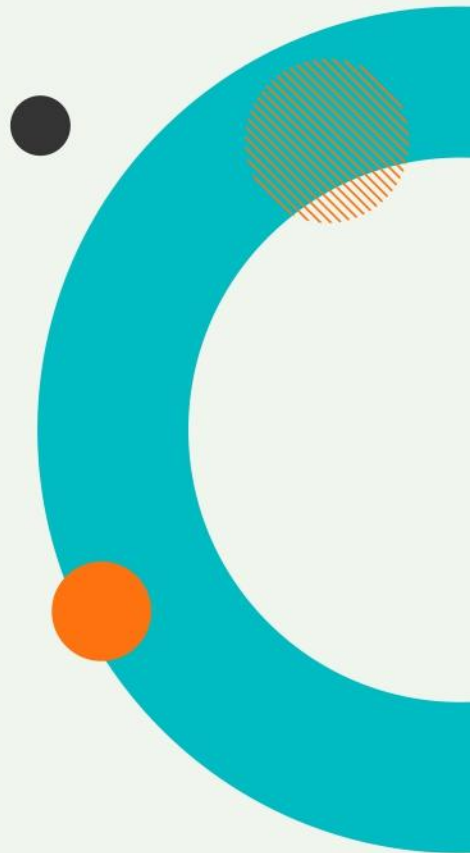
設計方法

03

實作流程
與成品 Demo

04

遇到困難
與可能解決方法





01

專案目標

人性化智能門禁系統



智能門禁系統

✦製作一個 user-friendly 的門禁GUI

- 。可感應多位不同使用者的門禁卡、磁扣
- 。可辨識多位不同使用者，並傳送客製化的 Line 訊息
- 。可根據不同情況，在液晶螢幕中顯示歡迎或是警告字樣
- 。若為外來入侵者，可透過 Line Notify 傳警告通知



智能音效：

- 若門禁系統辨識為外來入侵者 -> 播放「命運交響曲」
- 若門禁系統辨識為已知使用者 -> 播放「快樂頌」

02

設計方法

- 程式碼
 - 參考程式
 - 主程式
 - 副程式
- 線路示意圖
- 使用流程



設計方法: 流程&使用工具

1) 刷門禁卡、磁扣、輸入密碼

使用模組:

Arduino Code - MFRC 522
DumpInfo

3) 顯示使用者名稱

使用模組:

Arduino Code -
LiquidCrystal I2 C

2) pass / denied 的 buzzer 音樂

使用模組:
buzzer

4) Line訊息傳送

使用工具:
Arduino Code - Line
Notify



設計方法：欲使用的code & 模組

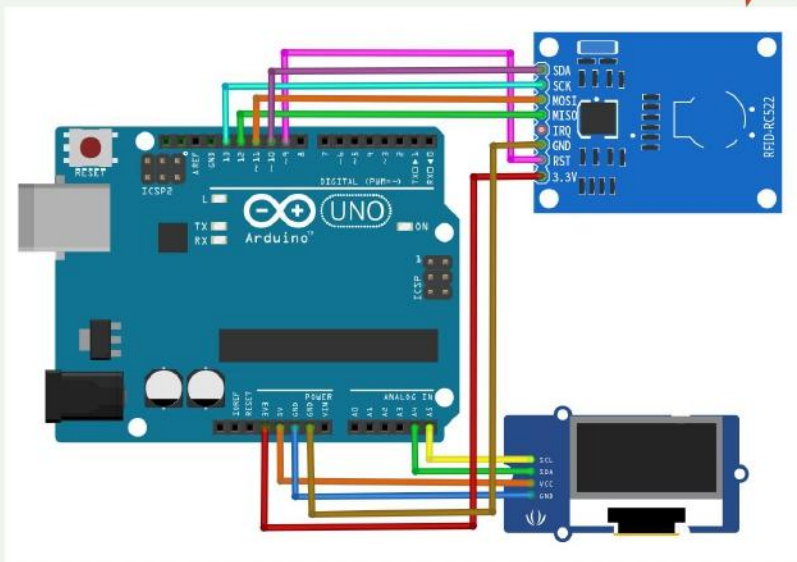
1-a) 刷門禁卡、磁扣、輸入密碼

參考模組：Arduino Code - MFRC 522 DumpInfo
主程式：Door_Security.ino

MFRC 522

1-b) 線路示意圖

SDA - pin 10
MOSI - pin 11
MISO - pin 12
SCK - pin 13
GND - GND
3.3 V - 3.3 V



設計方法: 欲使用的code & 模組

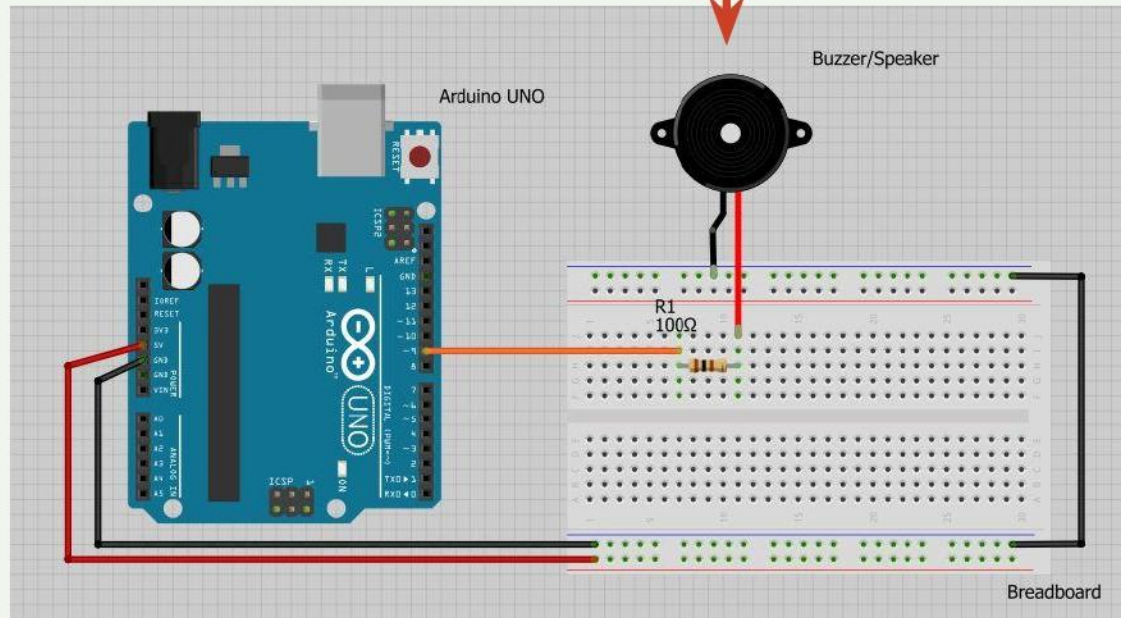
2-a) pass / denied 的 buzzer 音樂

副程式: buzzer.ino

無源
蜂鳴器

2-b) 線路示意圖

VCC - pin 7
GND - GND



設計方法: 欲使用的code & 工具

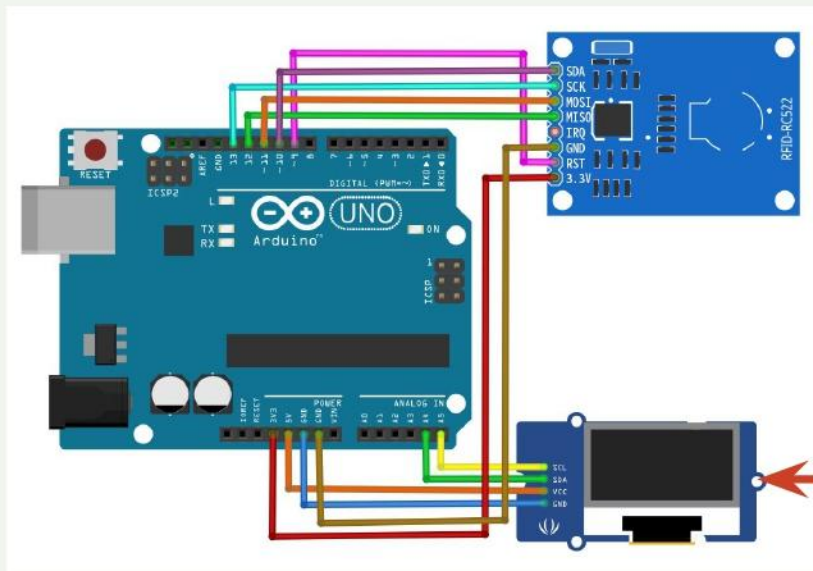
3 - a) 顯示使用者名稱

參考模組: Arduino Code - LiquidCrystal I2 C

主程式: Door_Security.ino

3 -b) 線路示意圖

SCL - pin A5
SDA - pin A4
GND - GND
VCC - 5V



Liquid
Crystal
I2C

設計方法:欲使用的code & 工具

4 - a) 顯示使用者名稱

參考模組: Arduino Code - Line Notify

主程式: Door_Security.ino

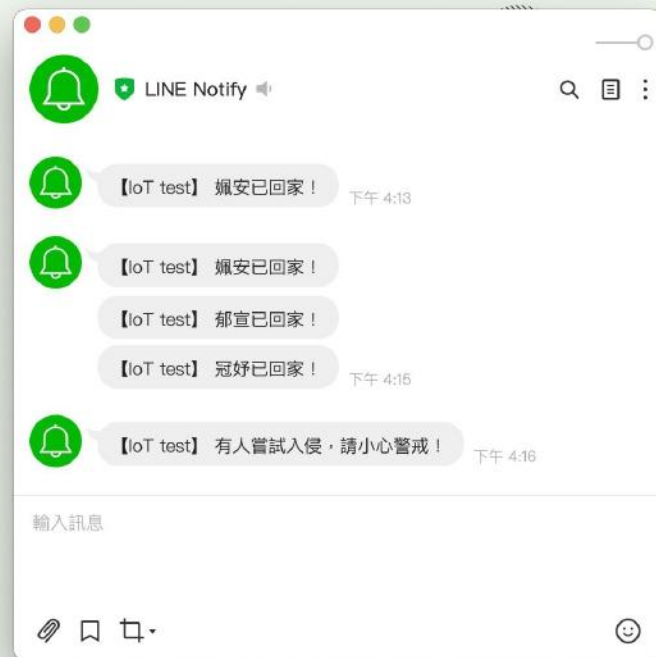
4 - b) Token & URL

Token :

QgDYw4n6sYvqrryofDqZnAYtvJoW4q1NgDgs9mMXxeL

URL :

http://3.114.197.59/test/line_notify.php?token=QgDYw4n6sYvqrryofDqZnAYtvJoW4q1NgDgs9mMXxeL&message=有人嘗試入侵, 請小心警戒!

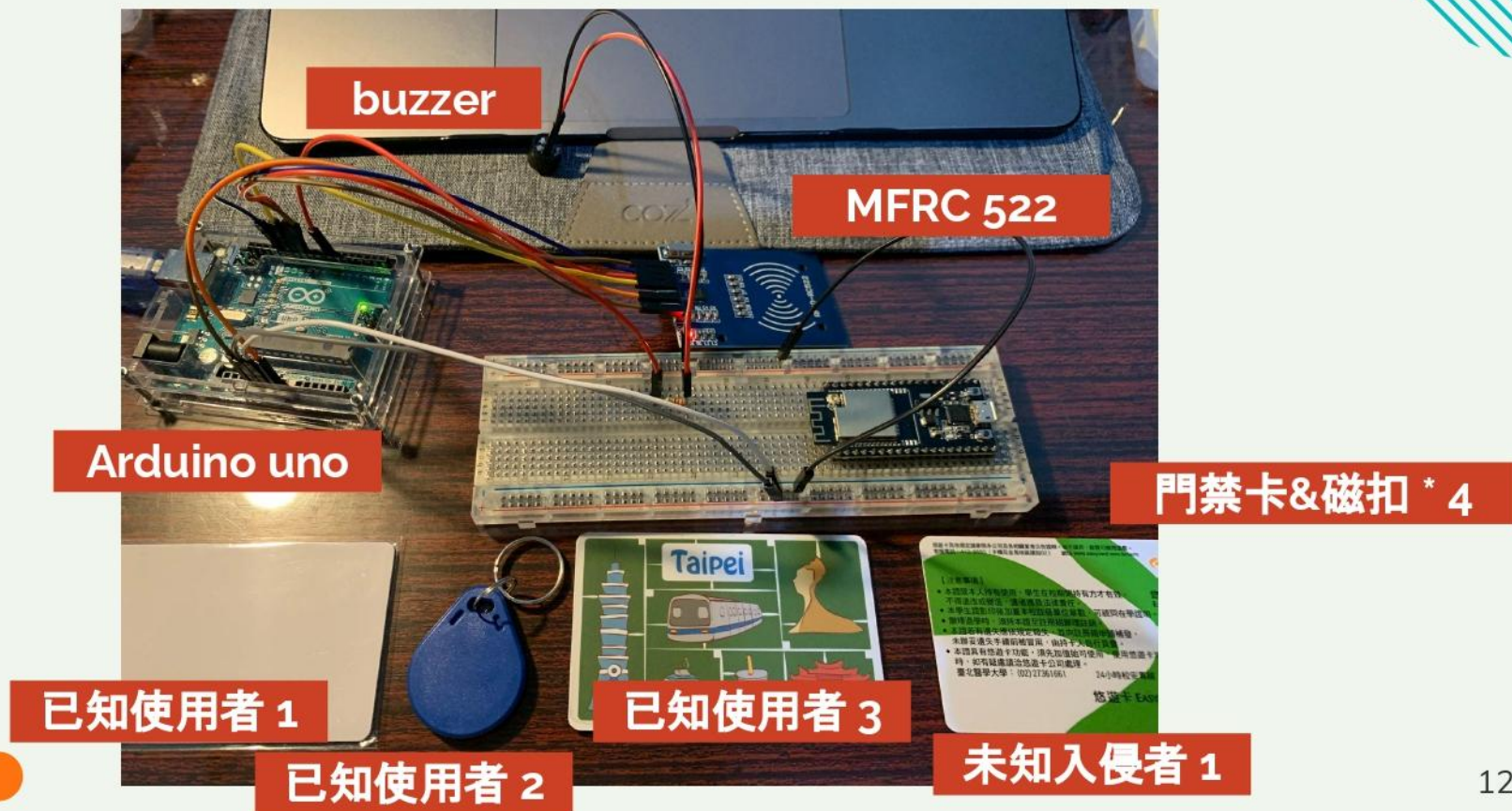




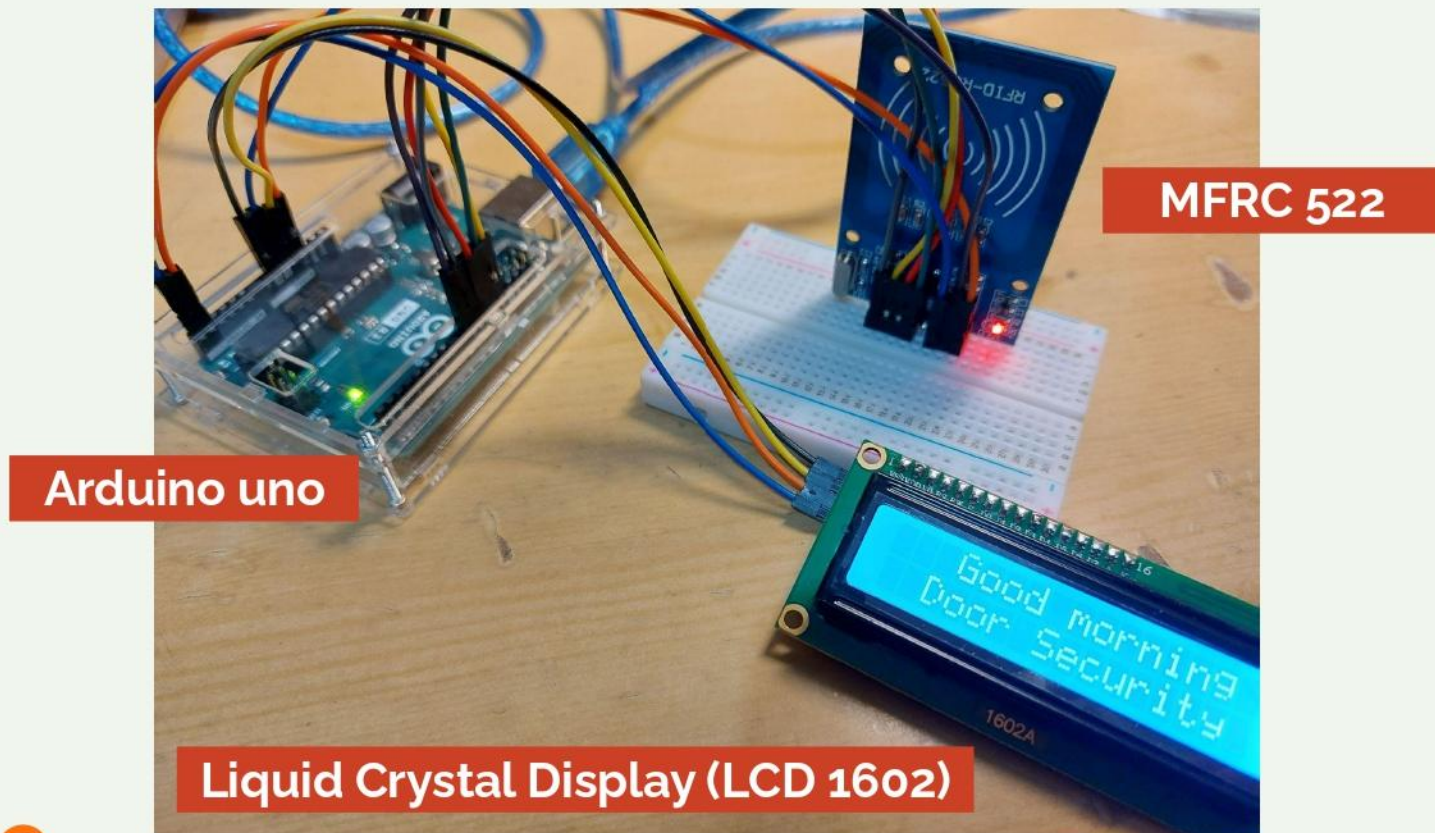
03

實作結果與成品 Demo

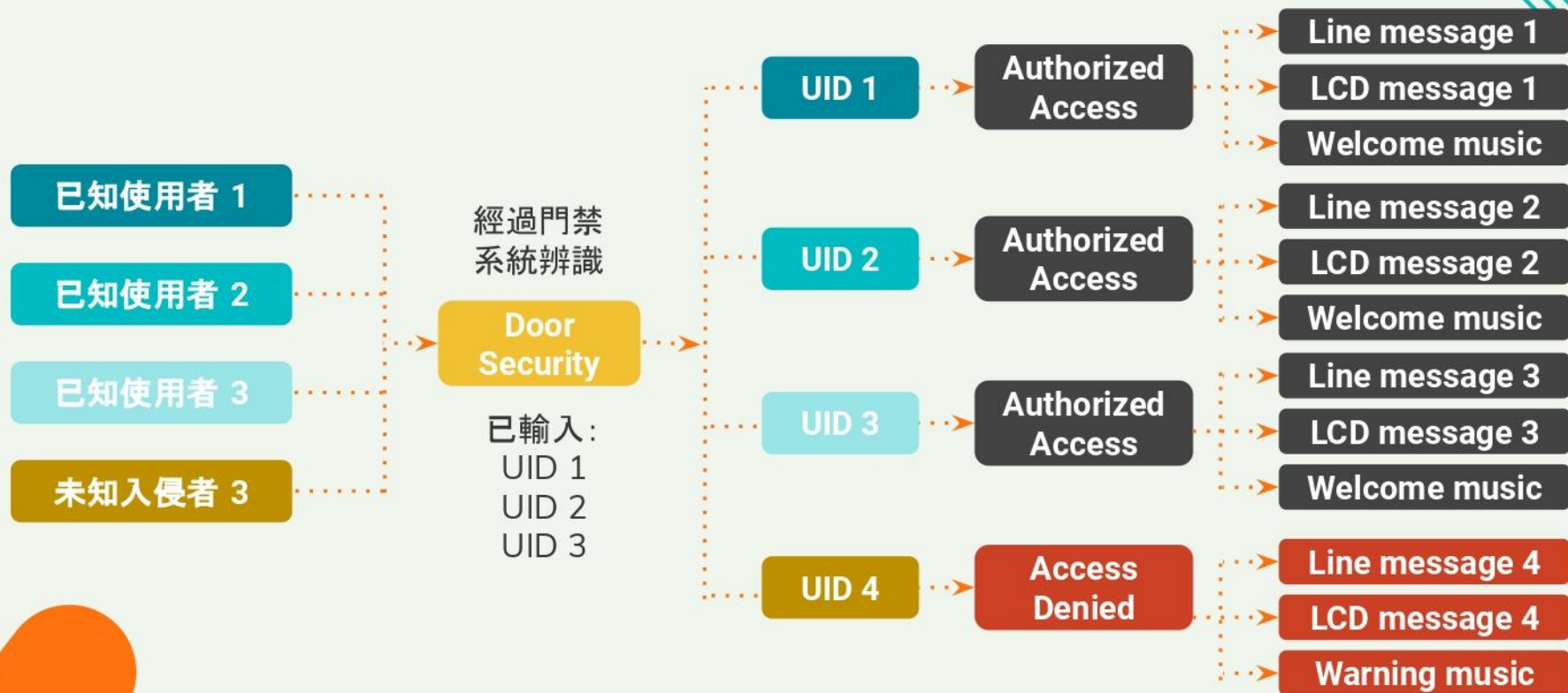
實作裝置圖 1



實作裝置圖 2



實作流程圖



實作成果 - UID recognition

The screenshot shows a terminal window titled `/dev/cu.usbmodem1401` with the following text:

```
Firmware Version: 0x92 = v2.0
Scan PICC to see UID, SAK, type, and data blocks...
UID tag : E5 8F 7A 48
Message : Access denied
UID tag : 75 2E 22 38
Message : Authorized access
UID tag : 03 54 25 97
Message : Authorized access
UID tag : E5 8F 7A 48
Message : Access denied
UID tag : 93 6C 3D 05
Message : Authorized access
UID tag : 75 2E 22 38
Message : Authorized access
UID tag : 03 54 25 97
Message : Authorized access
UID tag : E5 8F 7A 48
Message : Access denied
```

Overlaid on the terminal output is a diagram mapping specific UID tags to their access status:

- UID 1** (dark blue box) points to **Authorized Access** (dark grey box).
- UID 2** (teal box) points to **Authorized Access** (dark grey box).
- UID 3** (light blue box) points to **Authorized Access** (dark grey box).
- UID 4** (yellow box) points to **Access Denied** (red box).

The diagram uses orange dotted arrows to connect the UID boxes to the access status boxes. The UID boxes are positioned to the right of the terminal text, and the access status boxes are to their right. The UID boxes are labeled UID 1, UID 2, UID 3, and UID 4. The access status boxes are labeled Authorized Access and Access Denied.

At the bottom of the terminal window, there are controls: ☒ 自動捲動, ☐ Show timestamp, a dropdown menu set to NL & CR, a dropdown menu set to 9600 baud, and a Clear output button.

實作成果 - LCD display

LCD message 1



LCD message 2



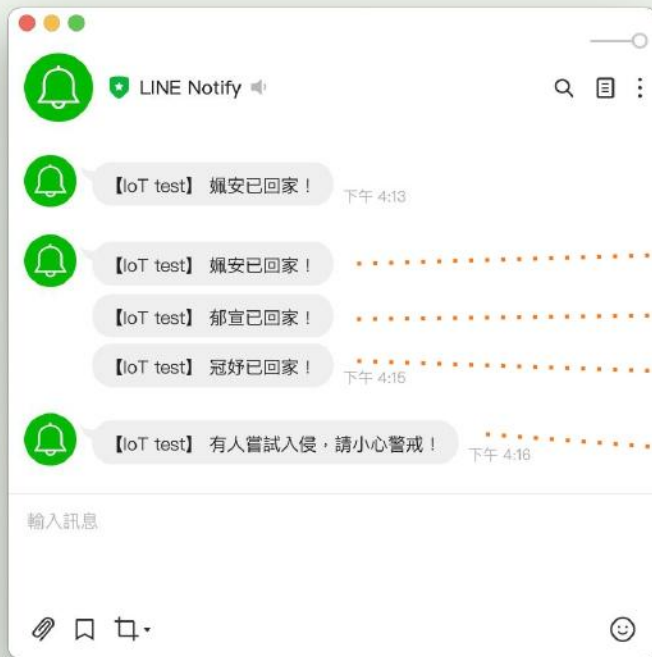
LCD message 3



LCD message 4



實作成果 - Line notification



Line message 1

Line message 2

Line message 3

Warning message

實作成果 - An die Freude 用 buzzer 編曲

```
54 //快樂頌 start/////
55 tone(buzzer, 320);
56 delay(300); // Mi...for 1 sec
57 noTone(buzzer); // Stop sound...
58 delay(200); // ...for 1sectone(buzzer, 500);
59 tone(buzzer, 320);
60 delay(300); // Mi...for 1 sec
61 noTone(buzzer); // Stop sound...
62 delay(200); // ...for 1sectone(buzzer, 500);
63 tone(buzzer, 341.3);
64 delay(300); //Fa ...for 1 sec
65 noTone(buzzer); // Stop sound...
66 delay(200); // ...for 1sec
67 tone(buzzer, 384);
68 delay(300); // So...for 1 sec
69 noTone(buzzer); // Stop sound...
70 delay(200); // ...for 1sectone(buzzer, 500);
71 tone(buzzer, 384);
72 delay(300); // So...for 1 sec
73 noTone(buzzer); // Stop sound...
74 delay(200); // ...for 1sectone(buzzer, 500);
75 tone(buzzer, 341.3);
76 delay(300); //Fa ...for 1 sec
77 noTone(buzzer); // Stop sound...
78 delay(200); // ...for 1sec
79 tone(buzzer, 320);
80 delay(300); // Mi...for 1 sec
81 noTone(buzzer); // Stop sound...
82 delay(200); // ...for 1sectone(buzzer, 500);
83 tone(buzzer, 288);
84 delay(300); // Re...for 1 sec
85 noTone(buzzer); // Stop sound...
86 delay(200); // ...for 1sectone(buzzer, 500);
87
```

```
88 tone(buzzer, 256);
89 delay(300); // Do...for 1 sec
90 noTone(buzzer); // Stop sound...
91 delay(200); // ...for 1sectone(buzzer, 500);
92 tone(buzzer, 256);
93 delay(300); // Do...for 1 sec
94 noTone(buzzer); // Stop sound...
95 delay(200); // ...for 1sectone(buzzer, 500);
96 tone(buzzer, 288);
97 delay(300); // Re...for 1 sec
98 noTone(buzzer); // Stop sound...
99 delay(200); // ...for 1sectone(buzzer, 500);
100 tone(buzzer, 320);
101 delay(300); // Mi...for 1 sec
102 noTone(buzzer); // Stop sound...
103 delay(200); // ...for 1sectone(buzzer, 500);
104 tone(buzzer, 320);
105 delay(400); // Mi...for 1 sec
106 noTone(buzzer); // Stop sound...
107 delay(200); // ...for 1sectone(buzzer, 500);
108 tone(buzzer, 288);
109 delay(250); // Re...for 1 sec
110 noTone(buzzer); // Stop sound...
111 delay(10); // ...for 1sectone(buzzer, 500);
112 tone(buzzer, 288);
113 delay(250); // Re...for 1 sec
114 noTone(buzzer); // Stop sound...
115 delay(1000); // ...for 1sectone(buzzer, 500);
116
```


實作成果 - Symphony No. 5 in C minor 用 buzzer 編曲

```
181 //命運交響曲 start/////
182 tone(buzzer, 384);
183 delay(300); // So...for 1 sec
184 noTone(buzzer); // Stop sound...
185 delay(200); // ...for 1sectone(buzzer, 500);
186 tone(buzzer, 384);
187 delay(300); // So...for 1 sec
188 noTone(buzzer); // Stop sound...
189 delay(200); // ...for 1sectone(buzzer, 500);
190 tone(buzzer, 384);
191 delay(300); // So...for 1 sec
192 noTone(buzzer); // Stop sound...
193 delay(200); // ...for 1sectone(buzzer, 500);
194 tone(buzzer, 311);
195 delay(600); // Mi...for 1 sec
196 noTone(buzzer); // Stop sound...
197 delay(1000); // ...for 1sectone(buzzer, 500);
198 tone(buzzer, 341.3);
199 delay(300); //Fa ...for 1 sec
200 noTone(buzzer); // Stop sound...
201 delay(200); // ...for 1sec
202 tone(buzzer, 341.3);
203 delay(300); //Fa ...for 1 sec
204 noTone(buzzer); // Stop sound...
205 delay(200); // ...for 1sec
206 tone(buzzer, 341.3);
207 delay(300); //Fa ...for 1 sec
208 noTone(buzzer); // Stop sound...
209 delay(200); // ...for 1sec
210 tone(buzzer, 288);
211 delay(600); // Re...for 1 sec
212 noTone(buzzer); // Stop sound...
213 delay(1000); // ...for 1sectone(buzzer, 500);
214 //命運交響曲 end/////
```

Demo 影片



可掃描 QR code
看 Demo 影片



<https://www.youtube.com/watch?v= QDMa7BvddY>



04

遇到困難 與可能解決方法

尚未串接好的功能



1. 用密碼解除門禁

目前查詢到的方法是能搭配 4x4 的 keypad, 還沒研究出完整串接的 code。



2. 輸出特定圖形

目前有用 Spyder 寫出聖誕樹的圖形, 還沒研究出可以顯示在液晶螢幕上的方法。



3. 即時攝影

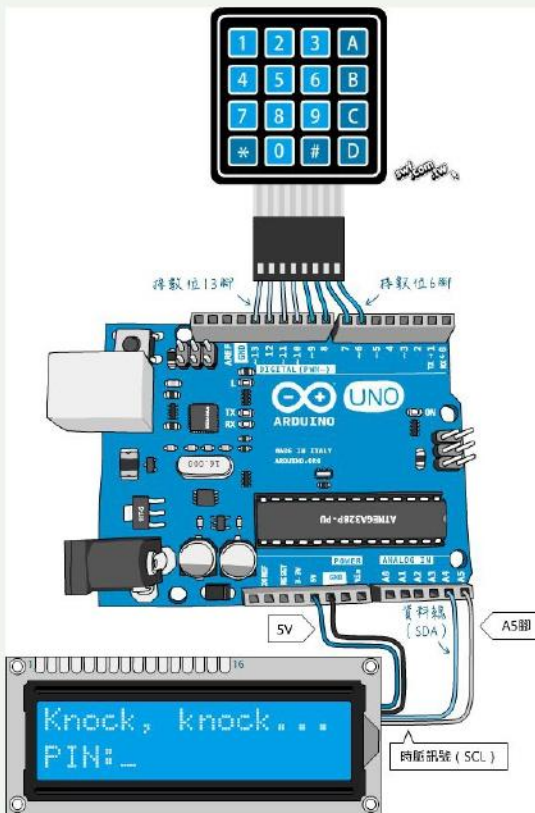
不知道 Arduino 能不能像樹莓派一樣外接攝影鏡頭。

用密碼解除門禁 - 可能解決方案

使用鍵盤模組：薄膜式的4x4鍵盤

準備材料：

1. Arduino Uno控制板
2. 4x4薄膜鍵盤
3. 1602文字型LCD顯示器模組
4. 自製LCD模組的序列轉接板或現成的 I2C 介面轉接板



用密碼解除門禁 - 可能解決方案

使用程式：

```
#include <Keypad.h>
#include <Wire.h>
#include <LiquidCrystal_SR.h> // 用LCD串列顯示器的程式庫

#define KEY_ROWS 4 // 鍵盤的列數
#define KEY_COLS 4 // 鍵盤的行數
#define LCD_ROWS 2 // LCD顯示器的列數
#define LCD_COLS 16 // LCD顯示器的行數

// 建立LCD顯示器物件：lcd(資料腳, 讀取腳, TWO_WIRE);
LiquidCrystal_SR lcd(A4,A5,TWO_WIRE);

// 設置鍵盤物件
char keymap[KEY_ROWS][KEY_COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};

byte rowPins[KEY_ROWS] = {13, 12, 11, 10}; // Rows 0 to 3
byte colPins[KEY_COLS] = {9, 8, 7, 6}; // Columns 0 to 3

Keypad keypad = Keypad(makeKeyMap(keymap), rowPins, colPins, KEY_ROWS, KEY_COLS);

String passcode = "4321"; // 預設密碼
String inputCode = ""; // 儲存用戶的按鍵字串
bool acceptKey = true; // 代表是否接受用戶按鍵輸入的提問, 預設為「接受」

void clearRow(byte n) {
  byte last = LCD_COLS - n;
  lcd.setCursor(n, 1); // 移動到第2行, "PIN:"之後
```

```
for (byte i = 0; i < last; i++) {
  lcd.print(" ");
}
lcd.setCursor(n, 1); // 移動到第2行, "PIN:"之後
}

// 顯示「歡迎光臨」後, 重設LCD顯示文字和輸入狀態。
void resetLocker() {
  lcd.clear();
  lcd.print("Knock, knock...");
  lcd.setCursor(0, 1); // 移動到第2行
  lcd.print("PIN:");
  lcd.cursor();

  acceptKey = true;
  inputCode = "";
}

// 判斷用戶輸入的密碼
void checkPinCode() {
  acceptKey = false; // 暫時不接受用戶按鍵輸入
  clearRow(0); // 從第0個字元開始清除LCD畫面
  lcd.noCursor();
  lcd.setCursor(0, 1); // 移動到第2行
  // 輸出密碼
  if (inputCode == passcode) {
    lcd.print("Welcome home!");
  } else {
    lcd.print("****WRONG!****");
  }
  delay(3000);
  resetLocker(); // 重設LCD顯示文字和輸入狀態
}
```

```
void setup() {
  Serial.begin(9600);

  lcd.begin(16,2); // 初始化LCD物件

  resetLocker();
}

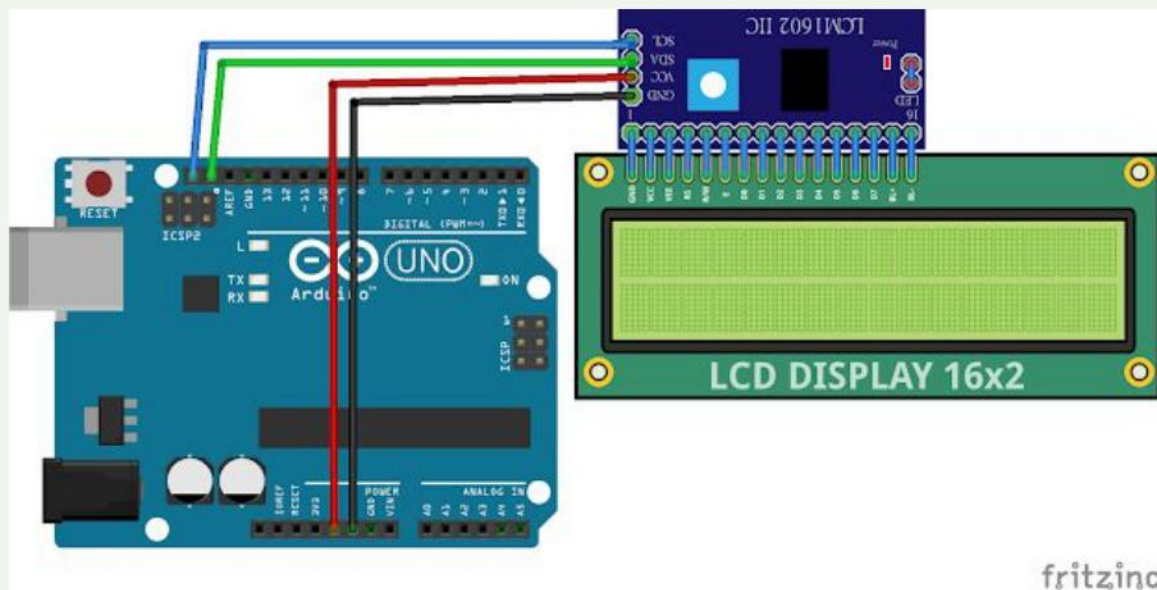
void loop() {
  char key = keypad.getKey();

  // 若目前接受用戶輸入, 而沒有新的字元輸入...
  if (acceptKey && key != NO_KEY) {
    if (key == '*') { // 清除畫面
      clearRow(4); // 從第4個字元開始清除
      inputCode = "";
    } else if (key == '#') { // 比較輸入密碼
      checkPinCode();
    } else {
      inputCode += key; // 儲存用戶的按鍵字元
      lcd.print('*');
    }
  }
}
```

輸出特定圖形 - 可能解決方案

使用跑馬燈模式

接線：



輸出特定圖形 - 可能解決方案

使用程式：

```
#include <LiquidCrystal_I2C.h> // 引用 LiquidCrystal_I2C Library

// 設定 LCD 位址為 0x27，有 16 個字元 2 列
LiquidCrystal_I2C lcd(0x27,16,2);
int row=0;

void setup()
{
    lcd.init();           // 初始化 lcd
    lcd.backlight();       // 設定背光為亮
}

void loop()
{
    lcd.clear();
    lcd.setCursor(0,row);
    lcd.print("hello!");
    for(int i=0;i<10;i++)
    {
        lcd.scrollDisplayRight(); // 往右移一格顯示
        delay(200);
    }
    // 判斷目前行數，是0行，將Row + 1，往下跳一行
    if(row == 0){
        row = row + 1;
        lcd.clear();
        lcd.setCursor(10,1);
        lcd.print("hello!");
        delay(200);
    }

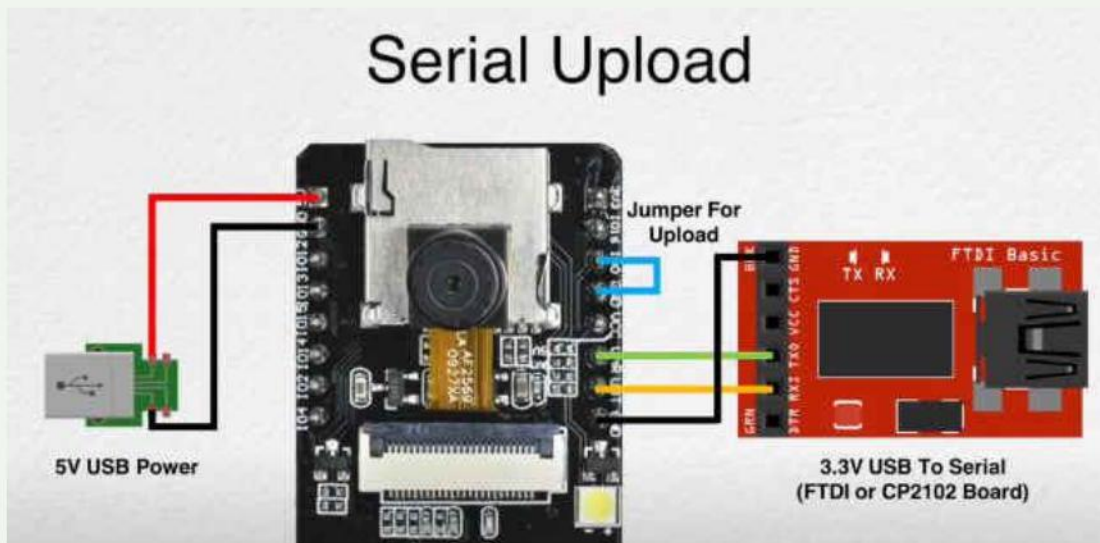
    for(int i=0;i<10;i++)
    {
        lcd.scrollDisplayLeft(); // 往左移一格顯示
        delay(200);
    }
    // 判斷目前行數，是1行，將Row 改為0行
    if(row == 1){
        row = 0;
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("hello!");
        delay(200);
    }
}
```

即時攝影 - 可能解決方案

使用**ESP32-Cam開發板**，透過網路可作為居家監控、嬰兒照護、即時錄影 ...等功能。

準備材料：

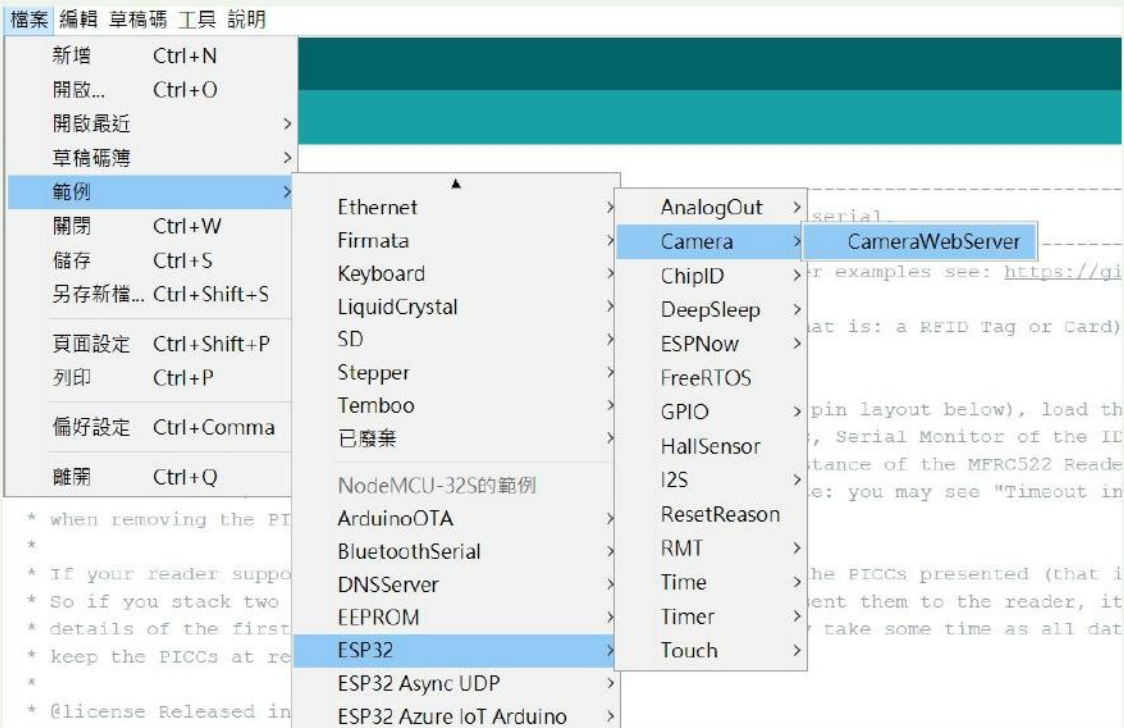
1. ESP32-Cam 開發板 *1
2. USB2TTL 模組 *1
3. 杜邦線母母頭 *5



即時攝影 - 可能解決方案

使用Arduino的範例程式：

下拉功能表 [檔案] > [範例] > [ESP32] > [Camera] > [CameraWebServer]。





05

小組分工

小組分工

- 程式運行：
 - MFRC 522 - 佩安
 - LCD - 冠妤
 - Line notification - 佩安
 - Buzzer - 佩安 & 郁宣
- 簡報製作：冠妤、郁宣、佩安
- 口頭報告：郁宣



Thank You !

Smart Door Security System With Audio Effect using Arduino
使用 Arduino 套件設計 智能多音效門禁系統

第二組

B812108030 陳嫻安、B405110031 陳冠妤、B313109084 陳郁宣

