

# 提款機

- Goal

完成 proposal 的內容

新增額外功能—修改輸入的密碼/金額

- Components

Interrupt: 接 Joystick 按鈕

Timer1: 判斷 Joystick 按下做提款機開關機

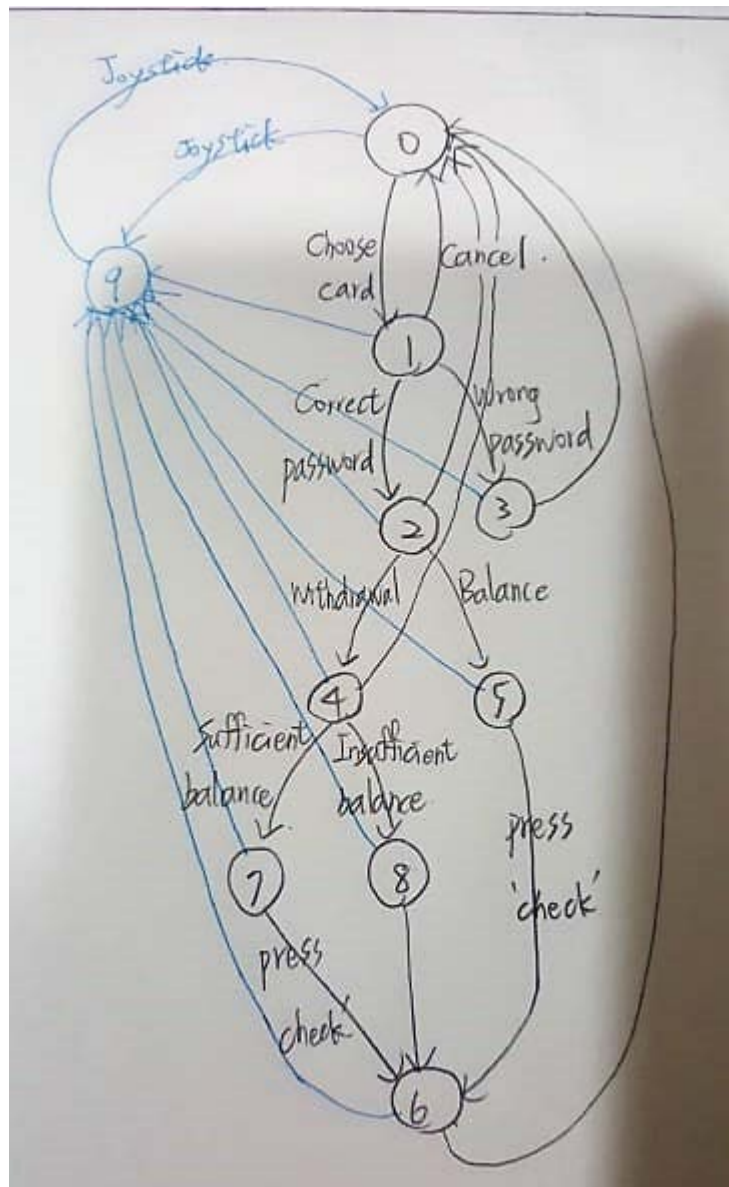
FreeRTOS: 控制流程

DAC: 顯示紅燈/綠燈

LCD: 顯示提款機螢幕

Sensor: 數字鍵盤/Joystick

- Program flow chart



## ● Detail

State 0 : 選擇卡片(有 5 張可選)

State 1 : 輸入密碼

State 2 : 選擇服務(1:提款 2:查詢餘額)

State 3 : 顯示密碼錯誤

State 4 : 輸入提款金額

State 5 : 顯示餘額

State 6 : 顯示 Thank you!

State 7 : 提領成功 顯示提出金額&餘額

State 8 : 顯示餘額不足

State 9 : 機器維修中

輸入密碼和金額時，按下#(修改鍵)可修改

## ● Code

我 create 了兩個 task。TaskIR()在鍵盤按鍵按下時會被 resume，用來判斷使用者在按下每個按鍵時，提款機應做出的相應動作，例如 state 的轉換、密碼輸入判斷、密碼修改等。TaskLCD()用來控制螢幕的顯示，在不同的 state 顯示不同的字幕。

我將鍵盤訊號的接收判斷直接放在 loop()內，當接收到訊號時直接 resume taskIR，再做處理。

我使用 Interrupt 去抓 Joystick 按鈕按下的訊號，用一個 isPress 變數去判斷按鈕按下的提款機狀態是開機或關機，並使用一個 timer 去判斷 isPress 變數來改變 state。

我將一些會重複用的功能包成 function 更清楚明瞭也方便在 code 中使用，以下介紹這些 function

1.

```
void writeRGB(int rValue, int gValue, int bValue){  
    analogWrite(redPin, rValue);  
    analogWrite(greenPin, gValue);  
    analogWrite(bluePin, bValue);  
}
```

用來寫入燈號的值。

2.

```
int cardMoney(char card){  
    if(card == '0') return money[0];  
    else if(card == '1') return money[1];  
    else if(card == '2') return money[2];  
    else if(card == '3') return money[3];  
    else if(card == '4') return money[4];  
}
```

回傳相對應卡內的金額。

3.

```
bool withdrawMoney(){
    int i = atoi(input_amount);
    int card = -1;
    if(CARD == '0') card = 0;
    else if(CARD == '1') card = 1;
    else if(CARD == '2') card = 2;
    else if(CARD == '3') card = 3;
    else if(CARD == '4') card = 4;

    if(money[card] - i < 0) return false;
    else{
        money[card] -= i;
        return true;
    }
}
```

判斷卡內金額是否足夠提領，若是則扣除金額並回傳是，否則直接回傳否。

4.

```
void initialToState0(){
    writeRGB(0,256,0);
    CARD = 0;
    input[0] = input[1] = input[2] = input[3] = 0;
    inputCount = 0;
    input_amount[0] = input_amount[1] = input_amount[2] = input_amount[3] = input_amount[4]
    amountCount = 0;
    state = 0;
}
```

回到 state 0 時候，將一些變數做初始化。

## ● Difficulties

一開始我本來 create 了三個 task，其中一個要用來判斷燈號要顯示紅燈或綠燈，但會一直因為記憶體的問題卡住，在之前寫 lab 的時候也常常因為記憶體問題一直在 debug，試了很多次不同

大小的記憶體後還是失敗，所以我直接把他改掉  
只用現在的兩個 task，燈號顯示的設定就把他放  
到這兩個 task 中。

我本來想用遙控器紅外線來做按鍵，但在接紅外  
線的時候沒有接好被燙到，就改成用 Keypad 來  
做按鍵。

## ● 心得

這學期的嵌入式課程讓我學到了很多有趣的東  
西，在做期末 project 前也在網路上看到很多別  
人用 Arduino 做的很酷的專案，以後時間允許的  
話也想再嘗試用 Arduino 來開發一些不一樣的  
project！