

嵌入式系統作業

期末專案

姓名 : 岳紀伶| 學號 : 01057113|

# 功能說明

ESP32隨機產生一個每位數皆不同的四位數當作答案，並藉於UART傳到8051，用Iot MQTT Panel傳送猜測的四位數到ESP32，再透過UART傳到8051的七段顯示器左邊四位顯示，右邊顯示猜測結果。如果正確，七段顯示器的右邊四位會顯示Done並且依去猜測數字播放音樂，十次內猜到答案是Bingo，超過十次撥放叮咚叮咚。

# 程式碼

//final.py

from machine import Pin

from umqtt.simple import MQTTClient

import utime, xtools

from machine import Pin,PWM

from machine import UART

import random

xtools.connect\_wifi\_led()

#ledG = Pin(12, Pin.OUT)

#ledG.value(0)

pin = Pin(13, Pin.OUT)

ADAFRUIT\_IO\_USERNAME = "VivianYueh"

ADAFRUIT\_IO\_KEY = "aio\_DnTc34xrksJDJuNZpAbbxfbOcUUU"

FEED = "final"

com = UART(2, 9600, tx=17, rx=16)

com.init(9600)

# MQTT 客戶端

client = MQTTClient (

client\_id = xtools.get\_id(),

server = "io.adafruit.com",

user = ADAFRUIT\_IO\_USERNAME,

password = ADAFRUIT\_IO\_KEY,

ssl = False,

)

t=0

def sub\_cb(topic, msg):

global ledG

print("收到訊息: ", msg.decode())

if len(msg.decode())==4 and f==0: #輸入四位數且還沒猜到正確答案

print(msg.decode())

guess=""

guess=guess+str(msg.decode()[0])+str(msg.decode()[1])+str(msg.decode()[2])+str(msg.decode()[3])

print(guess)

com.write("g"+guess+"\r\n")

else: #不是輸入四位數

print("請輸入4個數字")

client.set\_callback(sub\_cb) # 指定回撥函數來接收訊息

client.connect() # 連線

topic = ADAFRUIT\_IO\_USERNAME + "/feeds/" +FEED

print(topic)

client.subscribe(topic) # 訂閱主題

def create\_num(): #產生答案

s=set()

while len(s)<4: #產生四個不同的數字

x=random.randint(0,9)

s.add(x)

n=""

for i in s: #串接四個數字

n=n+str(i)

print(n)

com.write("s"+n+"\r\n")

def welldone(): #十次內猜到答案

led\_pwm = PWM(pin, freq=494, duty=512)

utime.sleep(0.5)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=494, duty=512)

utime.sleep(0.5)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=523, duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=523, duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=523, duty=512)

utime.sleep(0.5)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=440 , duty=512)

utime.sleep(0.5)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=440 , duty=512)

utime.sleep(0.5)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=494, duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=494, duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=494, duty=512)

utime.sleep(0.5)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=392 , duty=512)

utime.sleep(0.5)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=392 , duty=512)

utime.sleep(0.5)

led\_pwm = PWM(pin, freq=440 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=440 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=440 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=392 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=370 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=294 , duty=512)

utime.sleep(0.25)

led\_pwm = PWM(pin, freq=330 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=370 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=392 , duty=512)

utime.sleep(0.75)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=392 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

create\_num()

f=0

while True:

if com.any() > 0:

a = com.readline()

print(a)

if a == b'Reset': #按下k1，重新開始

f=0

t=0

if a==b'DOIT SHIN\r\n': #紀錄猜測次數，輸入不合法不算

t=t+1

if a == b'Finish\r\n': #猜到答案

f=1

if t<=10: #十次內猜到

welldone()

else: #超過十次猜到

led\_pwm = PWM(pin, freq=659 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=523 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=659 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

led\_pwm = PWM(pin, freq=523 , duty=512)

utime.sleep(0.25)

led\_pwm.deinit()

client.check\_msg()

//Final.c

#include<reg52.h> //包含標頭檔，一般情況不需要改動，標頭檔包含特殊功能寄存器的定義

#include <string.h>

#define MAX 10

#define DataPort P0 //定義資料埠 程式中遇到DataPort 則用P0 替換

typedef unsigned char byte;

typedef unsigned int word;

byte buf[MAX];

byte head = 0;

byte get\_0d = 0;

byte rec\_flag = 0;

sbit k1 = P1^0;

sbit k2 = P1^1;

sbit LATCH1=P2^2;//定義鎖存使能埠 段鎖存

sbit LATCH2=P2^3;// 位鎖存

unsigned char code dofly\_DuanMa[]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f,

0x77,0x7c,0x39,0x5e,0x79,0x71,0x54,0x79};// 顯示段碼值0~F

unsigned char code dofly\_WeiMa[]={0xfe,0xfd,0xfb,0xf7,0xef,0xdf,0xbf,0x7f};//分別對應相應的數碼管點亮,即位碼

unsigned char TempData[10]; //存儲顯示值的全域變數

unsigned char Ans[5];

unsigned char g[5];

void Display(unsigned char FirstBit,unsigned char Num);//數碼管顯示函數

void DelayUs2x(unsigned char t);//us級延時函式宣告

void DelayMs(unsigned char t); //ms級延時

void delay(unsigned char t);

void Song(void);

/\*------------------------------------------------

函式宣告

------------------------------------------------\*/

void SendStr(unsigned char \*s);

/\*------------------------------------------------

串口初始化

------------------------------------------------\*/

void InitUART (void)

{

SCON = 0x50; // SCON: 模式 1, 8-bit UART, 使能接收

TMOD |= 0x21; // TMOD: timer 1, mode 2, 8-bit 重裝,timer 0 , mode 1

TH1 = 0xFD; // TH1: 重裝值 9600 串列傳輸速率 晶振 11.0592MHz

TR1 = 1; // TR1: timer 1 打開

EA = 1; //打開總中斷

ET0=1; //計時器中斷打開

TR0=1; //計時器開關打開

// ES = 1; //打開串口中斷

}

/\*------------------------------------------------

主函數

------------------------------------------------\*/

void main (void)

{

bit set=0;

unsigned char j,a,b;

word i;

InitUART();

SendStr("UART test，技術論壇：www.doflye.net 請在發送區輸入任意資訊\r\n");

ES = 1; //打開串口中斷

while (1)

{

if (rec\_flag == 1)

{

buf[head] = '\0';

rec\_flag = 0;

head = 0;

if(buf[0]=='s'){ //設置答案

for(j=0;j<4;j++){

Ans[j]=buf[j+1]-'0';

}

SendStr("HELLO SHIN\r\n");

}

else if(buf[0]=='g'){ //讀取猜測

a=0;

b=0;

g[0]=buf[1]-'0';

g[1]=buf[2]-'0';

g[2]=buf[3]-'0';

g[3]=buf[4]-'0';

TempData[0]=dofly\_DuanMa[g[0]];

TempData[1]=dofly\_DuanMa[g[1]];

TempData[2]=dofly\_DuanMa[g[2]];

TempData[3]=dofly\_DuanMa[g[3]];

SendStr("DOIT SHIN\r\n");

if(g[0]==Ans[0]) a++;

else if(g[0]==Ans[1]||g[0]==Ans[2]||g[0]==Ans[3]) b++;

if(g[1]==Ans[1]) a++;

else if(g[1]==Ans[0]||g[1]==Ans[2]||g[1]==Ans[3]) b++;

if(g[2]==Ans[2]) a++;

else if(g[2]==Ans[0]||g[2]==Ans[1]||g[2]==Ans[3]) b++;

if(g[3]==Ans[3]) a++;

else if(g[3]==Ans[0]||g[3]==Ans[1]||g[3]==Ans[2]) b++;

if(a==4){

TempData[4]=dofly\_DuanMa[13];

TempData[5]=dofly\_DuanMa[0];

TempData[6]=dofly\_DuanMa[16];

TempData[7]=dofly\_DuanMa[14];

SendStr("Finish\r\n");

}

else{

TempData[4]=dofly\_DuanMa[a]; //幾個A

TempData[5]=dofly\_DuanMa[10];

TempData[6]=dofly\_DuanMa[b]; //幾個B

TempData[7]=dofly\_DuanMa[11];

}

//Display(0,8);

}

/\*if (strcmp(buf,"HELLO") == 0)

{

SendStr("HELLO SHIN\r\n");

}

else if (strcmp(buf,"DOIT") == 0)

{

SendStr("DOIT SHIN\r\n");

}

else

{

SendStr(buf);

SendStr("\r\n");

} \*/

}

if (k1 == 0) //重置

{

for(i=0;i<2000;i++);

if(k1==0) SendStr("Reset");

while(k1 == 0);

for(i=0;i<2000;i++);

for(j=0;j<8;j++) TempData[j]=0;

Display(0,8);

}

}

}

/\*------------------------------------------------

發送一個位元組

------------------------------------------------\*/

void SendByte(unsigned char dat)

{

SBUF = dat;

while(!TI);

TI = 0;

}

/\*------------------------------------------------

發送一個字串

------------------------------------------------\*/

void SendStr(unsigned char \*s)

{

while(\*s!='\0')// \0 表示字串結束標誌，通過檢測是否字串末尾

{

SendByte(\*s);

s++;

}

}

/\*------------------------------------------------

串口中斷程式

------------------------------------------------\*/

void UART\_SER (void) interrupt 4 //串列中斷服務程式

{

unsigned char tmp; //定義臨時變數

if(RI) //判斷是接收中斷產生

{

RI=0; //標誌位元清零

tmp=SBUF; //讀入緩衝區的值

if (get\_0d == 0)

{

if (tmp == 0x0d) get\_0d = 1;

else

{

buf[head]=tmp;

head++;

if (head == MAX) head = 0;

}

}

else if (get\_0d == 1)

{

if (tmp != 0x0a)

{

head = 0;

get\_0d = 0;

}

else

{

rec\_flag = 1;

get\_0d = 0;

}

}

// SBUF=tmp; //把接收到的值再發回電腦端

}

// if(TI) //如果是發送標誌位元，清零

// TI=0;

}

/\*------------------------------------------------

顯示函數，用於動態掃瞄數碼管

輸入參數 FirstBit 表示需要顯示的第一位，如賦值2表示從第三個數碼管開始顯示

如輸入0表示從第一個顯示。

Num表示需要顯示的位元數，如需要顯示99兩位元數值則該值輸入2

------------------------------------------------\*/

void Display(unsigned char FirstBit,unsigned char Num)

{

static unsigned char i=0;

DataPort=0; //清空資料，防止有交替重影

LATCH1=1; //段鎖存

LATCH1=0;

DataPort=dofly\_WeiMa[i+FirstBit]; //取位碼

LATCH2=1; //位鎖存

LATCH2=0;

DataPort=TempData[i]; //取顯示資料，段碼

LATCH1=1; //段鎖存

LATCH1=0;

i++;

if(i==Num)

i=0;

}

/\*------------------------------------------------

計時器中斷副程式

------------------------------------------------\*/

void Timer0\_isr(void) interrupt 1

{

TH0=(65536-2000)/256; //重新賦值 2ms

TL0=(65536-2000)%256;

Display(0,8); // 調用數碼管掃瞄

}

# 結果

七段顯示器根據MQTT APP輸入顯示結果和Iot MQTT Panel的輸入，當猜測的答案和正確的答案相同時播放Bingo或叮咚叮咚。

|  |  |  |
| --- | --- | --- |
| 圖片 | | 說明 |
|  |  | 1. 程式執行，產生隨機的四位數 2. 輸入超過四位數，印出”請輸入4個數字” |
|  |  | 1. MQTT APP送出猜測的數字 2. 8051七段顯示器左邊顯示猜測的數字，右邊顯示幾A幾B |
|  |  | 1. 猜測正確，右邊顯示Done 2. 8051判斷正確，送出Finish |
|  |  | 按下k1，Reset遊戲，重新產生答案且清空七段顯示器 |

# 心得

在這個作業中，運用到8051、ESP32、UART和MQTT，由於上個作業已經練習過ESP32和8051藉由UART連接，在寫這個作業時十分順利。原本想結合第二題，但因為兩個timer都已經使用了，所以只能透過從ESP32接線到喇叭來撥放音樂。

感謝老師和助教這個學期的教導和幫助。