微處理機系統實習　　Lab5

班級：資訊三甲 學號：D1109023 姓名：楊孟憲.

一、【實驗目的】：

撰寫 C 程式控制 GPIO 的 LCD、LED、Keypad 和 Seven-Segment Display 的邏輯操作。

二、【遭遇的問題】：

沒有遇到問題。

三、【解決方法】：

Lab5.1簡易計算機

```c
#include <stdio.h>
#include <string.h>
#include "NUC100Series.h"
#include "MCU_init.h"
#include "SYS_init.h"
#include "Scankey.h"
#include "LCD.h"
#include "Seven_Segment.h"

int a, b, onClear;

void Display_7seg()
{
    int i;
    int arr[3];
    arr[0] = a;
    arr[1] = b;
    arr[2] = onClear;
    // 3 2 1 0
    for(i = 0; i < 3; i++) {
        CloseSevenSegment();
        ShowSevenSegment(i, arr[i]);
        CLK_SysTickDelay(2000);
    }
  return;
}

void solve(int a, int b) {
    char line[4][128];
    int i = 0;
    memset(line, 0, sizeof(line));
    sprintf(line[0], "%d + %d = %d", a, b, a + b);
    sprintf(line[1], "%d - %d = %d", a, b, a - b);
    sprintf(line[2], "%d * %d = %d", a, b, a * b);
    sprintf(line[3], "%d / %d = %d", a, b, a / b);
    for(i = 0; i < 4; i++) {
        print_Line(i, line[i]);
    }
    return;
}
```

```c
int main(void)
{
    int i = 0, keyPressed = 0;
    onClear = 0, a = 0, b = 0;
    SYS_Init();
    OpenKeyPad();
    GPIO_SetMode(PB, BIT11, GPIO_MODE_OUTPUT);
    init_LCD();


    while(1) {
        i=ScanKey();
        Display_7seg();

        if( i == 0 ) {
            keyPressed = 0;
            continue;
        }

        if(keyPressed) {
            continue;
        }

        keyPressed = 1;

        if ( onClear ) {
            onClear = 0;
            a = b = 0;
            clear_LCD();
            continue;
        }

        if ( a == 0 ) {
            a = i;
        } else if( b == 0 ) {
            b = i;
            solve(a, b);
            onClear = 1;
        }
    }

    return 0;
}
```

Lab5.2 四位電子鎖：

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "NUC100Series.h"
#include "MCU_init.h"
#include "SYS_init.h"
#include "Scankey.h"
#include "LCD.h"
#include "Seven_Segment.h"

#define INF (int)(1e9)

int password[4];
int input[4];
int currentIndex = 0;
int seed = 0;

// LCD
char line[4][128];
int lcdIndex = 0;

// LED
int currentLED = 0;
int led[4];
int showLED = 0;

char *result[3] = {
    "PASS",
    "ERROR",
    "NULL"
};


void LCD(int status) {
    if ( lcdIndex >= 4 ) {
        return;
    }

    printS(64, 16 * lcdIndex++, result[status]);
    return;
}
```

```c
void BinLED() {
    memset(led, 1, sizeof(led));
    led[currentLED] = 0;
    currentLED = ( currentLED + 1 ) % 4;
    PC12 = led[0];
    PC13 = led[1];
    PC14 = led[2];
    PC15 = led[3];
    return;
}

void Buzz()
{
    PB11=0; // PB11 = 0 to turn on Buzzer
    CLK_SysTickDelay(10000);     // Delay
    PB11=1; // PB11 = 1 to turn off Buzzer
    CLK_SysTickDelay(10000);     // Delay
}

void Display_7seg()
{
    int i = 0;
    if ( password[0] == -1 ) {
        CloseSevenSegment();
        return;
    }

    for(i = 0; i < 4; i++) {
        CloseSevenSegment();
        ShowSevenSegment(i, password[i]);
        CLK_SysTickDelay(2000);
    }
  return;
}

void userInput(int number) {
    char C[1];
    sprintf(C, "%d", number);
    if ( currentIndex >= 4 || password[0] == -1 ) {
        return;
    }

    input[currentIndex] = number;
    printS(8 * currentIndex, 16 * lcdIndex, C);
    currentIndex++;
    return;
}

void generatePassword() {
    int i = 0;
    for(i = 0; i < 4;i++) {
        password[i] = rand() % 5 + 1;
        seed = (seed + 1) & INF;
    }
    return;
}

void clearPassword() {
    //memset(password, -1, sizeof(password));
    memset(input, -1, sizeof(input));
    clear_LCD();
    currentIndex = 0;
    lcdIndex = 0;
    return;
}
```

```c
void validate() {
    int ok = 1, i = 0, j = 3;

    if ( password[0] == -1 ) return;

    if ( currentIndex == 0 ) {
        LCD(2);
        return;
    }

    for (i = 0, j = 3; i < 4; i++, j--) {
        if ( password[j] != input[i] ) {
            ok = 0;
            break;
        }
    }

    if ( ok ) {
        LCD(0);
        showLED = 1;
    } else {
        Buzz();
        LCD(1);
    }

    return;
}

void Init_GPIO() {
    GPIO_SetMode(PC, BIT12, GPIO_MODE_OUTPUT);
    GPIO_SetMode(PC, BIT13, GPIO_MODE_OUTPUT);
    GPIO_SetMode(PC, BIT14, GPIO_MODE_OUTPUT);
    GPIO_SetMode(PC, BIT15, GPIO_MODE_OUTPUT);
    PC12 = PC13 = PC14 = PC15 = 1;
    return;
}

int main(void)
{
    int i = 0, keyPressed = 0;
    SYS_Init();
    OpenKeyPad();
    GPIO_SetMode(PB, BIT11, GPIO_MODE_OUTPUT);
    Init_GPIO();

    init_LCD();
    clearPassword();


    while(1) {
        i=ScanKey();
        Display_7seg();
        if ( showLED ) BinLED();
        seed++;
        seed %= INF;
        srand(seed);

        if( i == 0 ) {
            keyPressed = 0;
            continue;
        }

        if(keyPressed) {
            continue;
        }

        keyPressed = 1;

        if ( i == 7 ) {
            generatePassword();
        } else if ( i == 8 ) {
            clearPassword();
        } else if ( i == 9 ) {
            validate();
            currentIndex = 0;
        } else {
            userInput(i);
        }
    }

    return 0;
}
```

四、【未能解決的問題】：

沒有未能解決的問題。