

Introduction to Embedded System Design

Lab Report

- Lab date: 2023-3-29 (year-month-day)
- Group number: _____
- Group members: (student ID) (name)

410921365 陳祁鎔

410921304 張芯翎

1. Lab Title:

猜數字遊戲

2. Lab Goal:

讓使用者輸入數字後，藉由蜂鳴器和 LED 燈猜測數字

3. Lab Description and Steps:

1. 首先由一使用者藉由 Keypad 輸入四位數字(該四位數不會有 0)，蜂鳴器會響一聲，代表輸入完成。

2. 另一使用者可以開始藉 Keypad 輸入數字來猜。

3. 蜂鳴器響幾聲代表有幾 A(也就是有幾個數字對，位置也對)，LED 亮幾顆代表有幾 B(也就是有幾個數字對，但位置錯)，蜂鳴器和 LED 燈會同時運作。

4.根據蜂鳴器和 LED 燈，使用者可推測正確的四位數。

5.當使用者猜出正確的四位數後，蜂鳴器會響四聲，以及 LED 燈會亮，最後七段顯示器會以跑馬燈的形式，由右到左顯示使用者總共猜測的次數。

6.按下 reset 可以重新開始遊戲。

4. Code:

```
// keypad input 4 digits and display on 7-segment LEDs
```

```
#include <math.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

```
#include <time.h>
```

```
#include "MCU_init.h"
```

```
#include "NUC100Series.h"
```

```
#include "SYS_init.h"
```

```
#include "Scankey.h"
```

```
#include "Seven_Segment.h"
```

```
// display an integer on four 7-segment LEDs
```

```
void Display_7seg(uint16_t value) {
```

```
    uint8_t digit;
```

```

    digit = value / 1000;
    CloseSevenSegment();
    ShowSevenSegment(3, digit);
    CLK_SysTickDelay(5000);
    value = value - digit * 1000;
    digit = value / 100;
    CloseSevenSegment();
    ShowSevenSegment(2, digit);
    CLK_SysTickDelay(5000);
    value = value - digit * 100;
    digit = value / 10;
    CloseSevenSegment();
    ShowSevenSegment(1, digit);
    CLK_SysTickDelay(5000);
    value = value - digit * 10;
    digit = value;
    CloseSevenSegment();
    ShowSevenSegment(0, digit);
    CLK_SysTickDelay(5000);
}

void displayRound(uint16_t round) {
    uint8_t digit;
    uint16_t tmp = round;

```

```
digit = round / 1000;
```

```
CloseSevenSegment();
```

```
if (digit != 0)
```

```
    ShowSevenSegment(3, digit);
```

```
CLK_SysTickDelay(500000);
```

```
round = round - digit * 1000;
```

```
digit = round / 100;
```

```
CloseSevenSegment();
```

```
if (digit != 0)
```

```
    ShowSevenSegment(2, digit);
```

```
CLK_SysTickDelay(500000);
```

```
round = round - digit * 100;
```

```
digit = round / 10;
```

```
CloseSevenSegment();
```

```
if (digit != 0)
```

```
    ShowSevenSegment(1, digit);
```

```
CLK_SysTickDelay(500000);
```

```
round = round - digit * 10;
```

```
digit = round;
```

```
CloseSevenSegment();
```

```

    if (digit != 0)
        ShowSevenSegment(0, digit);
    CLK_SysTickDelay(500000);

    displayRound((tmp * 10) % 10000);
}

void Buzz(uint16_t number) {
    int i;
    for (i = 0; i < number; i++) {
        PB11 = 0;  // PB11 = 0 to turn on Buzzer
        CLK_SysTickDelay(100000);  // Delay
        PB11 = 1;  // PB11 = 1 to turn off Buzzer
        CLK_SysTickDelay(100000);  // Delay
    }
}

int main(void) {
    int passcode;
    int tmppass = 1314;
    int i = 3;
    int j = 0;
    int k = 0;
    int s = 0;

```

```
int p = 0;

int code, tmpcode, number[10];

int round = 0;


for (j = 0; j < 50; j++)
    Display_7seg(s);


memset(number, 0, sizeof(number));

SYS_Init();

OpenSevenSegment(); // for 7-segment

OpenKeyPad(); // for keypad

GPIO_SetMode(PB, BIT11, GPIO_MODE_OUTPUT); // for
Buzzer

GPIO_SetMode(PC, BIT12, GPIO_MODE_OUTPUT); // for
Buzzer


while (1) {
    k = ScanKey();
    if (k != 0) {
        s = s * 10 + k;
        printf("key=%d i=%d s=%d\n", k, i, s);
        i--;
        if (i < 0) {
            i = 3;
        }
    }
}
```

```

        code = s;
        s = 0;
        for (j = 0; j < 50; j++)
            Display_7seg(code);
        tmppass = code;
        Buzz(1);
        break;
    }
}
for (j = 0; j < 10; j++)
    Display_7seg(s);
}

while (1) {
    k = ScanKey();
    if (k != 0) {
        s = s * 10 + k;
        printf("key=%d i=%d s=%d\n", k, i, s);
        i--;
        if (i < 0) {
            i = 3;
            tmpcode = s;
            s = 0;
            round++;
        }
    }
}

```

```
if (tmpcode == tmppass) {  
    Buzz(4);  
    PC12 = 0;  
    PC13 = 0;  
    PC14 = 0;  
    PC15 = 0;  
    CLK_SysTickDelay(100000);  
    PC12 = 1;  
    PC13 = 1;  
    PC14 = 1;  
    PC15 = 1;  
    displayRound(round);  
    break;  
}  
else {  
    int count = 0, B = 0;  
    code = tmpcode;  
    passcode = tmppass;  
    for (p = 0; p < 4; p++) {  
        number[passcode % 10]++;  
        if (code % 10 == passcode % 10) {  
            count++;  
            number[passcode % 10]--;  
        }  
    }  
}
```



```
        code = code / 10;
        passcode = passcode / 10;
    }
    code = tmpcode;
    passcode = tmppass;
    for (p = 0; p < 4; p++) {
        if (number[code % 10]) {
            B++;
            number[code % 10]--;
        }
        code = code / 10;
        passcode = passcode / 10;
    }
    switch (count) {
        case 1:
            Buzz(1);
            break;
        case 2:
            Buzz(2);
            break;
        case 3:
            Buzz(3);
            break;
    }
```

```
        switch (B) {
            case 1:
                PC12 = 0;
                break;
            case 2:
                PC12 = 0;
                PC13 = 0;
                break;
            case 3:
                PC12 = 0;
                PC13 = 0;
                PC14 = 0;
                break;
            case 4:
                PC12 = 0;
                PC13 = 0;
                PC14 = 0;
                PC15 = 0;
                break;
        }
    }
    for (j = 0; j < 50; j++)
        Display_7seg(tmpcode);
}
```

```
    }  
    for (j = 0; j < 10; j++)  
        Display_7seg(s);  
    PC12 = 1;  
    PC13 = 1;  
    PC14 = 1;  
    PC15 = 1;  
    memset(number, 0, sizeof(number));  
}  
}
```

5. Lessons:

在實作的過程中，有遇到些困難，例如怎麼計算 **A** 和 **B** 的數量，可以用陣列計算出現次數來解決，還有怎麼讓數字在七段顯示器以跑馬燈的形式跑過，最後是用遞迴和讓 **CLK** 延遲來解決。

其實開發板還滿有趣的，只是它的 **IDE** 和 **debugger** 很難用，也很難 **debug**，如果以後有機會會想再嘗試看看。