Thư viện bốn LED 7 đoạn

Mục đích khai báo thư viện là để tiện cho việc viết code.Đó là thay thế cả một đoạn code dài chỉ bằng một dòng code gọi hàm trong thư viện ra để dùng.Không những vậy còn dễ quản lí và sửa lỗi bài code hơn.Ta có các bước để tạo một thư viện cho LED 7 đoạn đơn,đôi,bốn như sau:

File header “port.h” để khai báo các chân của vi điều khiển sử dụng

#ifndef \_PORT\_H\_ // Đánh dấu cho port.h để tránh lỗi khai báo lặp

#define \_PORT\_H\_

#define PORT P2 //Khai báo port dữ liệu ở port P2

sbit LED1 = P3 ^ 0; //Khai báo LED1 ở chân P3.0

sbit LED2 = P3 ^ 2; //Khai báo LED1 ở chân P3.2

sbit LED3 = P3 ^ 4; //Khai báo LED1 ở chân P3.4

sbit LED4 = P3 ^ 6; //Khai báo LED1 ở chân P3.6

#endif

File “main.h”khai báo tần số thạch anh,thanh ghi của vi điều khiển

#ifndef \_MAIN\_H\_

#define \_MAIN\_H\_

#include<regx52.h> //Khai báo thanh ghi của vi điều khiển

#define FREQ\_OSC 12000000 //Tần số thạch anh 12MHZ

#endif

**File “delay.c” và “delay.h” dùng để tiêu tốn thời gian**

**+)File h:**

#ifndef \_DELAY\_H\_

#define \_DELAY\_H\_

#include "main.h"

void Delay(unsigned int n); //Tên hàm Delay

#endif

**File c:**

#include"delay.h" //khai báo thư viện delay.h

#include"main.h" //khai báo thư viện main.h

void Delay(unsigned int n) {

unsigned int x, y;

for (x = 0; x < n; x++) { //Tiêu tốn n ms

for (y = 0; y < 123; y++); //Tiêu tốn 1ms

}

}

File “LED7DOAN4LED.h” và file “LED7DOAN4LED.c”

**+)File h:**

#ifndef \_LED7DOAN4LED\_H\_

#define \_LED7DOAN4LED\_H\_

void hienthi(unsigned int num);

#endif

**+)File c:**

#include"main.h"

#include"LED7DOAN4LED.h"

#include"port.h"

#include"delay.h"

unsigned char code Code7Seg[] = {//mảng mã code của LED7 đoạn từ 0 - 9

0xC0,

0xF9,

0xA4,

0xB0,

0x99,

0x92,

0x82,

0xF8,

0x80,

0x90

};

hienthi(unsigned int num) { //hàm hiển thị

if ((num < 10) &&(num>=0)){ //TH số hiển thị nhỏ hơn 10,lớn hơn hoặc bằng 0

LED1 = 0; // Cho LED1 bật

PORT = Code7Seg[num % 10]; //Truyền dữ liệu cho LED hàng đơn vị

Delay(1000); //delay 1s

}

if ((9 < num)&&(num < 100)) { // TH số hiển thị nhỏ hơn 100,lớn hơn 9

for(j=0;j<500;j++) //lặp lại 500 lần để tổng delay là 1s

{ LED1 = 0;

PORT = Code7Seg[num % 10]; //dữ liệu cho LED hàng đơn vị

Delay(1);

LED1 = 1;

LED2 = 0;

PORT = Code7Seg[num / 10]; //dữ liệu cho LED hàng chục

Delay(1);

LED2 = 1;

}

}

if (99 < num < 1000) { //TH num lớn hơn 99 và nhỏ hơn 1000

for(j=0;j<333;j++) //lặp 333 lần để tổng delay là 1s

{ LED1 = 0;

PORT = Code7Seg[num % 10]; //Truyền dữ liệu cho LED hàng đơn vị

Delay(1);

LED1 = 1;

LED2 = 0;

PORT = Code7Seg[(num % 100) / 10]; //dữ liệu cho LED hàng chục

Delay(1);

LED2 = 1;

LED3 = 0;

PORT = Code7Seg[num / 100]; //dữ liệu cho LED hàng trăm

Delay(1);

LED3 = 1;

}

}

if (999 < num < 10000) { //TH num lớn hơn 999 nhỏ hơn 10000

for(j=0;j<250;j++) //lặp 250 lần để tổng delay bằng 1 s

{

LED1 = 0;

PORT = Code7Seg[num % 10]; //Truyền dữ liệu cho LED hàng đơn vị

Delay(1);

LED1 = 1;

LED2 = 0;

PORT = Code7Seg[(num % 100) / 10]; //dữ liệu cho LED hàng chục

Delay(1);

LED2 = 1;

LED3 = 0;

PORT = Code7Seg[(num % 1000) / 100];//dữ liệu cho LED hàng trăm

Delay(1);

LED3 = 1;

LED4 = 0;

PORT = Code7Seg[num / 1000]; //dữ liệu cho LED hàng nghìn

Delay(1);

LED4 = 1;

}

}

}