201803010 Ahmet CITAK

#include <16f877.h> // program dosyasi ismi tanimlaniyor

#fuses HS, NOWDT // WDT kullanimi ve fuse tanimlamasi yapiliyor

#use delay (clock=4000000) // osilatör frekansi ayarlaniyor

#define display\_1 pin\_a1 // pin tanimlamasi yapiliyor

#define display\_2 pin\_a0 // pin tanimlamasi yapiliyor

unsigned int kesme=0; // degisken tanimlamasi yapiliyor

char birler=0, onlar=0, sayi=0; // degisken tanimlamasi yapiliyor

int i,z, led; // degisken tanimlamasi yapiliyor

const int digit[10]={0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F};

// dizi tanimlamasi yapiliyor

#int\_timer1 // timer1 kesmesi baslatiliyor

void timer1\_kesme () // timer1 kesmesi fonksiyonu tanimlaniyor

{

clear\_interrupt(int\_timer1);

set\_timer1(65532); // timer 1’e 65532 sayisi yukleniyor

kesme++; // kesme degiskeni 1 arttiriliyor

sayi++; // sayi degiskeni 1 arttiriliyor

if (sayi==100) // eger sayi 100’se

sayi=0; // sayi 0’a esitleniyor

for( z=0; z<200; z++) // 200 kere dönecek bir for dongusu baslatiliyor

{

birler=sayi%10; // sayi degiskeninin mod 10 aliniyor ve birler degiskenine esitleniyor

onlar=sayi/10; // sayi degiskeni 10’a bolunuyor ve onlar onlar degiskenine esitleniyor

output\_high(display\_2); // display\_2 degiskenine cikis veriliyor

output\_d(digit[birler]); //digit dizisinde birler degiskeninin değeri getiriliyor ve d portuna cikis olarak veriliyor

delay\_ms(5); // 5ms gecikme

output\_low(display\_2); // display 2 cikisi 0’laniyor

output\_high(display\_1); // display\_1 cikis veriliyor

output\_d(digit[onlar]); //digit dizisinde onlar degiskeninin değeri getiriliyor ve d portuna cikis olarak veriliyor

delay\_ms(5); // 5ms gecikme

output\_low(display\_1); // display 1 cikisi 0’laniyor

if (kesme==100) // eger kesme 100 ise

kesme=0; // kesmeyi 0’a esitle

}

}

void main ( ) // ana program baslangici

{

set\_tris\_a(0xFC); // a portunda cikis ve girişler ayarlaniyor

set\_tris\_b(0x00); // b portu tamamen cikis ayarlaniyor

set\_tris\_c(0x01); // c portunda cikis ve girişler ayarlaniyor

set\_tris\_d(0x00); // d portu tamamen cikis ayarlaniyor

output\_low(display\_1); // display 1 cikisi 0’laniyor

output\_low(display\_2); // display 2 cikisi 0’laniyor

output\_b(0x00); // b portu cikisi 0’laniyor

setup\_timer\_1(T1\_EXTERNAL | T1\_DIV\_BY\_2); // timer1 ayarlari yapiliyor

set\_timer1(65532); // timer1’e deger yukleniyor

enable\_interrupts(INT\_timer1); // kesmeler aktif hale getiriliyor

enable\_interrupts(GLOBAL); // kesmeler aktif hale getiriliyor

while(1) // sonsuz döngüye giriliyor

{

led=1; // led degiskeni 1’e esitleniyor

for(i=0;i<=7;i++) // 7 kez donen for dongusu baslatiliyor

{

output\_b(led); // b portuna led degiskeni cikisi veriliyor

led=led<<1; // led degiskeni 1 bit sola kaydirilir

delay\_ms(200); // 200 ms gecikme

}

led=0x80; // led degiskeni hex 0x80’e esitlenir

output\_b(led); // b portuna led degiskeni değeri cikisi verilir

delay\_ms(200); // 200 ms gecikme

for(i=0;i<=6;i++) // 6 kez donen for degiskeni baslatilir

{

led=led>>1; // led degiskeni saga 1 bit kaydirilir

output\_b(led); // b portuna led degiskeni cikisi verilir

delay\_ms(200); // 200 ms gecikme

}

}

}

Program Aciklamasi: Bu program RC0 bacagina bagli butona basimadigi surece kara simsek uygulamasidir. RC0 butonuna 8 kere basildigi zaman karasimsek uygulamasindan cikilir ve kesmeye gidilir. Kesme fonksiyonu içerisindeki işlemler gerçekleştirilir. Ardindan kesmeden cikilir ve karasimsek kaldigi yerden devam eder.

1. Display’de gösterilen sayının, gösterilme sayısını 200 yerine 350 ve 500 olarak tekrar denediğimizde kesme fonksiyonunun suresinin uzadigini görmekteyiz. Displayde gösterilmesi daha da uzamaktadır. Ancak bu durumda hata alinir. Cunku for içerisindeki ‘z’ degiskeni int olarak tanimlanmistir ve alabileceği maksimum deger 255’tir.