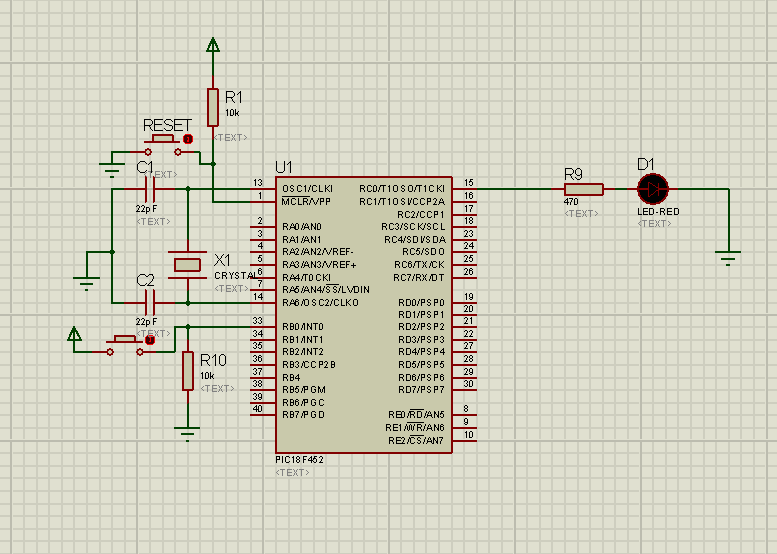
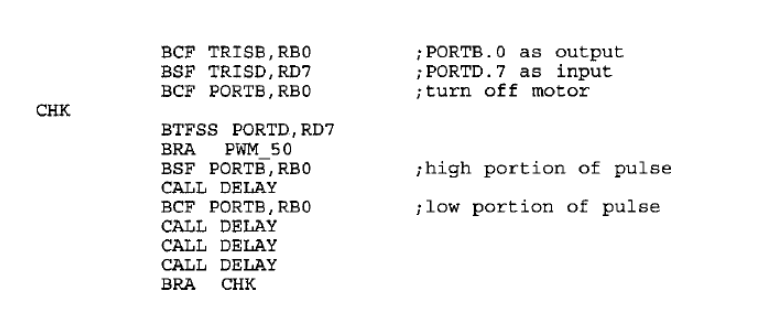
**PWM modülü kullanmadan PWM sinyalinin elde edilmesi**



PWM modülü kullanılmadan PWM sinyali elde edebilmek için, çıkış portuna verilen sinyal istenilen değere çekilerek (High ya da Low) gereken süre kadar bekleme komutu çağırılır. Aynı işlem sinyalin diğer değeri için de periyodun geri kalanı kadar bekleme komutu çağırılarak yapılır. Aşağıda bu durumun assembly kodu ile yazılışı bulunmaktadır. Sinyal önce High değerine çekilmiş ve bir kez delay çağırılmıştır. Bir delay’lik High değerinden sonra sinyal Low’a çekilmiş ve üç kez delay eklenerek sinyalin bir periyodu tamamlanmıştır.



**Delay tabanlı PWM için C kodu:**

#pragma config[1] = 0xF1 // Osilatör: XT#pragma config[1] = 0xF1 // Osilatör: XT

void ayarlar();

void bekle(unsigned long t); // t milisaniye gecikme sağlayan fonksiyon tanımı

void kesme();

void kesme() //kesme gelince yapilacak komutlar, kesmede calisacak fonksiyon main fonksiyonunun ustunde yazilir...

{

INTCON=0x90; // kesmeler acilir RBO/INT0 girisi interrupt enable edilir.

//GO=1; // adc cevrimi baslar

//while(GO); //cevirme bitene kadar calisir, cevirme bitince go=0 olur

INT0IF=0; // yeni kesmeler gelmesi icin butona bagli olan INT0 portundaki interrupt flagi kapatilir.

GIE=1; //kesmeler acilir, yeni kesme gelmesine musade edilir

PWM\_dongu:

PORTC.0=1;

bekle(0x7F); // %50 High – ADC kullanıldığında ADRESH değeri olmalıdır

PORTC.0=0;

bekle(0xFF-0x7F); // %50 Low – ADC kullanıldığında 0xFF-ADRESH değeri olmalıdır

goto PWM\_dongu;

}

void main()

{

ayarlar();

anadongu:

//bekle(1); // Acquisition Time(Sample & Hold kapasitörünün şarj olması için gerekli zaman)

INTCON=0x90;

nop();

goto anadongu;

//-----------------------------------------------

}

void bekle(unsigned long t) //t milisaniye gecikme sağlar

{

unsigned x;

t=t/2;

for(;t>0;t--)

for(x=140;x>0;x--)

nop();

}

void ayarlar() // Bütün başlangıç ayarlarının tamamlandığı kısım

{

GIE=1; // Bütün kesmeleri ac

TRISA=0xFF;

TRISB=0xFF;

TRISC=0x00;

TRISD=0x00;

TRISE=0xFF;

PORTC=0x00;

PORTD=0x00;

//ADCON0=0b.0100.0001;// Anlog kanal 0 aktif, A/D conversion is not in progress

//ADCON1=0b.0000.0000;

}

//////////////////////////////////////////////////////////////////////////////////////////////////

Yukarıdaki program interrupt kullanılarak yazılım içerisindeki kesme fonksiyonunda belirtilen aralıklarda PWM sinyali üretmektedir. Üretilen PWM sinyali, C portunun ilk pininden led’e bağlanmıştır. Program PIC’e gönderilip çalıştırıldığında verilen aralıklarla led’in yanıp söndüğü görülmektedir. Programdaki ADC ile ilgili kısımlar yorum satırı şeklinde bulunmaktadır. PWM süresi POT değerine bağlı ayarlanmak istenirse, bu kısımlar kullanılacaktır. PWM\_dongu döngüsü ise bu durumda her zaman çalışacağından main fonksiyonundaki anadongu içerisine alınmalıdır.

Bu laboratuvar çalışmasında istenen, yukarıdaki örnek programdan da yararlanarak, PWM sinyalindeki High ve Low sürelerinin PIC’in analog kanalına bağlanan POT ile ayarlanması ve elde edilen sinyal ile interrupt tabanlı olarak motor hızının kontrol edilmesidir. POT’dan alınan değerler ADC kullanılarak dijital veriye çevrilecek, bu çevrilen veriler ADRESH yazmacından okunarak bekle komutuna uygulanacaktır. Elde edilen sinyal motor ile PIC arasında koruma amaçlı bulunan Optocoupler’ın girişine verilecek ve motor hızı PWM sinyalindeki High süresine bağlı olarak ayarlanacaktır. Aşağıda PIC ile motor sürebilmek için gerekli koruyucu devrenin şeması verilmiştir.

