



Konu 1 : DC Motor Kontrol

Örnek Kod

```
#include<16f876a.h> // Pic Wizard ile otomatik olarak da ayarlanabilir.
#define ADC = 10 // ADC, 10 bit olacak şekilde ayarlanır.
//#fuses HS // High Speed - Yüksek Hızlı Kristal veya Seramik Resonatör kullanılacak.
#fuses XT // XTal - Kristal veya Seramik Resonatör
#fuses NOWDT // No Watch Dog Timer - WDT kullanılmayacak.
#fuses NOLVP // No Low Voltage Programming - LVP kullanılmayacak.
#fuses NOPROTECT // Program Belleğine (ROM) yazılan verilerin okunmasına izin verir.

#USE delay(clock = 4000000)

#define motor1_yon_belirle pin_a2 // 0 Geri :: 1 İleri ( Buton ile belirlenir )
#define motor_1_ileri pin_b4 // L293D için kullanılacak bacaklara özel isimler verilir.
#define motor_1_geri pin_b5

int motor_1_hiz; // Global değişken tanımlanır.

void main(){
    setup_adc(ADC_CLOCK_INTERNAL); // ADC aktif hale getirilir.
    setup_adc_ports(AN0); // A0 pin'i analog giriş(input) olarak ayarlanır.

    setup_ccp1(CCP_PWM); // CC1 bacağı PWM olarak kullanılacağı tanımlanır.
    setup_ccp2(CCP_OFF); // CC2 bacağı pasif hale getirilir.
    setup_timer_2(T2_DIV_BY_16,255,1); // Timer2 ayarları yapılır.
    // Kristal sinyali 16'ya bölünür.
    // Timer-2 sayıcısının sınır değeri 255 olarak belirlenir.

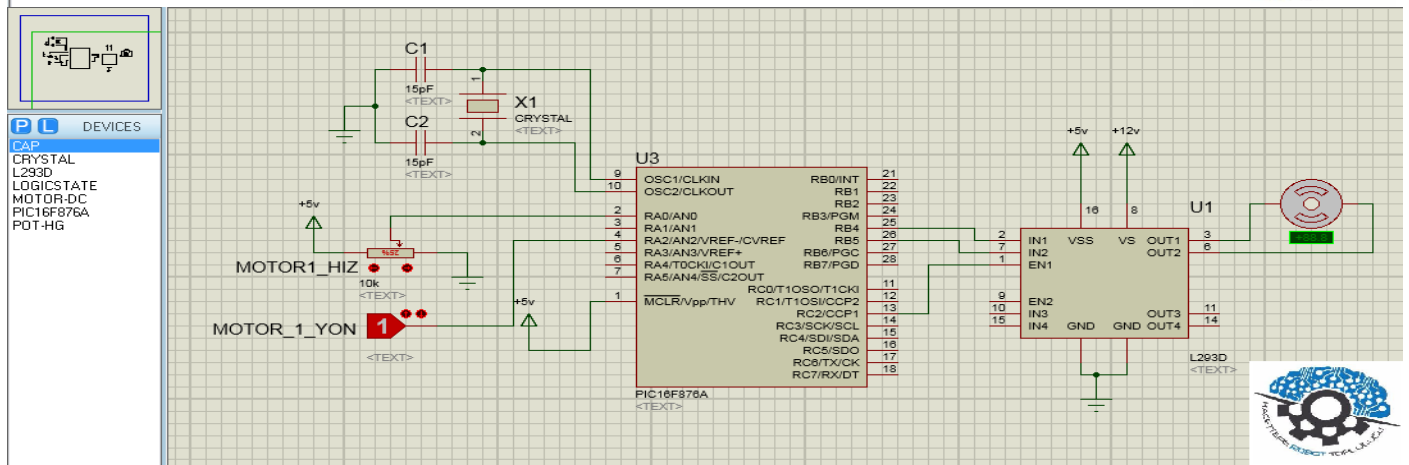
    set_tris_a(0x0f); // A0-3 pinleri Giriş(Input), A4-6 pinleri ise Çıkış(Output) olarak ayarlanır.
    set_tris_b(0x00); // B portu Çıkış(Output) olarak ayarlanır.
    set_tris_c(0x00); // C portu Çıkış(Output) olarak ayarlanır.

    output_b(0x00); // Portlar sıfırlanır.
    output_c(0x00);

    while(1){
        if( input(motor1_yon_belirle) ){ // 1. motorun hangi yönde gideceği belirlenir.
            output_high(motor_1_ileri);
            output_low(motor_1_geri);
        }
        else if( !input(motor1_yon_belirle) ){
            output_low(motor_1_ileri);
            output_high(motor_1_geri);
        }
        // Analog veri okunur.
        set_adc_channel(0); // AN0 analog bacağındaki veri okunur.
        delay_us(10); // 10 mikrosaniye gecikme oluşturulur( ADC işlemlerin yapılması için gereklidir).
        motor_1_hiz = read_adc()/4 ; // read_adc() ile 10bit lik okunan değer 4'e bölünüp değişkene eşitlenir.

        set_pwm1_duty( motor_1_hiz ); // 1. motorun hızı ayarlanır
        delay_ms(100); // 100 milisaniye gecikme oluşturulur.
    }
}
```

Proteus Çizimi



Konu 2 : Servo Motor Kontrol

Örnek Kod

```
// Ön işlemci fonksiyon ayarları yapılır
// Pic Wizard ile otomatik olarak da ayarlanabilir.
#include<16f876a.h> // High Speed - Yüksek Hızlı Kristal veya Seramik Resonatör kullanılacak.
//#fuses HS // XTal - Kristal veya Seramik Resonatör
//fuses XT // No Watch Dog Timer - WDT kullanılmayacak.
//fuses NOWDT // No Low Voltage Programming - LVP kullanılmayacak.
//fuses NOLVP // Program Belleğine (ROM) yazılan verilerin okunmasına izin verir.
//fuses NOPROTECT

#USE delay(clock = 4000000)

#define servo_bilgi pin_a0 // Servo motorun veri bacağı tanımlanır.

int i; // Global bir değişken tanımlanır.

void main(){
    setup_adc(ADC_OFF); // ADC pasif hale getirilir.
    setup_adc_ports(NO_ANALOGS); // Analog bacakların kullanılmayacağı belirtilir.

    setup_ccp1(CCP_OFF); // CC1 bacağı pasif hale getirilir.
    setup_ccp2(CCP_OFF); // CC2 bacağı pasif hale getirilir.

    setup_timer_1(T1_DISABLED); // Timer1 pasif hale getirilir.
    setup_timer_2(T2_DISABLED,0,1); // Timer2 pasif hale getirilir.

    set_tris_a(0x00); // A portu Çıkış(Output) olarak ayarlanır.
    set_tris_b(0x00); // B portu Çıkış(Output) olarak ayarlanır.
    set_tris_c(0x00); // C portu Çıkış(Output) olarak ayarlanır.

    output_b(0x00); // Portlar sıfırlanır.
    output_c(0x00);

    while(1){ // Servo motorda istenilen açılara döndürmek için tanımlanmış duty-cycle oranları oluşturularak
        // servo motor istenilen açığa döndürülür. Bu oranlar aşağıda belirtilmiştir.
        // Servo motor, PWM kullanılarak da istenilen açıda döndürülebilir. Fakat aşağıdaki gibi
        // high-low kullanılarak da kendi kare dalgamızı üretip servo motora açtı verebiliriz.
        for( i = 0; i < 5 ; i++){ // Periyodu 20ms için aç/süre karşılığı;
            output_high(servo_bilgi); // a = 1000 ==> -90 derece || a = 1500 ==> 0 derece || a = 2000 ==> +90 derece
            delay_us(1500);
            output_low(servo_bilgi);
            delay_us(20000 - 1000);
        }
        delay_ms ( 1000); // 1000 milisaniye gecikme oluşturulur.

        for( i = 0; i < 5 ; i++){ // Periyodu 20ms için aç/süre karşılığı;
            output_high(servo_bilgi); // a = 1000 ==> -90 derece || a = 1500 ==> 0 derece || a = 2000 ==> +90 derece
            delay_us(1000);
            output_low(servo_bilgi);
            delay_us(20000 - 1000);
        }
        delay_ms ( 1000); // 1000 milisaniye gecikme oluşturulur.
    }
}
```

Proteus Çizimi

