

## ARDUİNO İLE C#

### ARDUİNO İLE C#

- ✓ *Arduino ve C# ile Servo Motor*

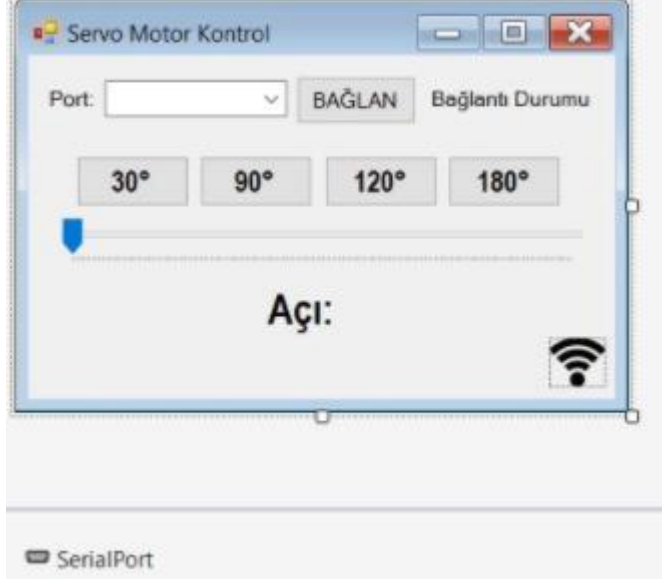
### KAZANIMLAR

- ✓ *Seri haberleşme ile birden fazla veriyi nasıl alabileceğini öğrenir.*



**Örnek:** Arduino ve C# haberleşmesi serisinin bu haftaki seri haberleşmeyi kullanarak **servo motor** kontrolünü göreceğiz. Ancak diğer yazılardan farklı olarak bu işlemi **Bluetooth UART** modülü kullanarak **kablosuz** gerçekleştireceğiz. Yani seri iletişimi **Bluetooth** yolu ile sağlaycaz.

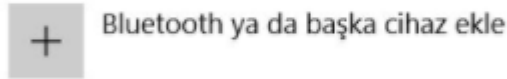
### C# Arayüz Tasarımı



### Seri İletişimi Bluetooth ile Sağlamak

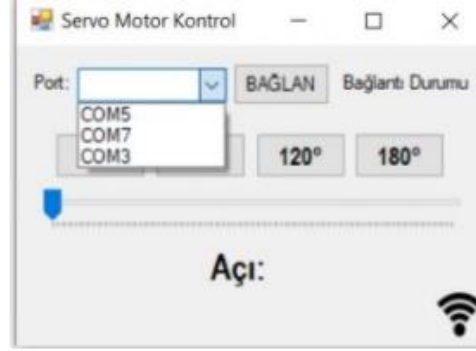
Devremizi **Bluetooth** üzerinden çalıştırmak için öncelikle bilgisayar ile modülü ilişkilendirmemiz gerekiyor. Bağlantıyı sağlamak için şifre gerekecektir. **HC-05** modülünün şifresi varsayılan ayarlarda **"1234"**tür. Dolayısıyla modül ayarlarında bir değişiklik yapmadıysanız bu şifreyi girerek modüle erişebilirsiniz.

## Bluetooth & diğer cihazlar



Bluetooth Ekleme

Bağlantıyı sağladıktan sonra kodunuzu çalıştırdığınızda arayüz üzerinde portların listelendiği comboBox'ta eski projelerden farklı olarak birden fazla **port** ile karşılaşacaksınız.



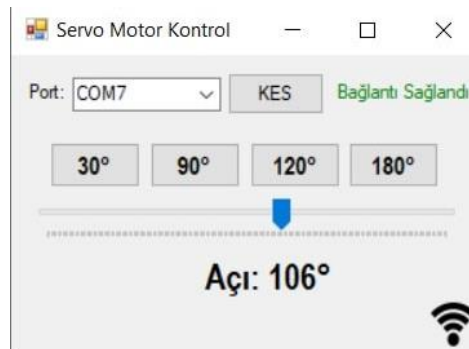
Seri Portlar

Bunun sebebi bağlantı noktaları arasına Bluetooth modülünün de dahil olmasıdır. Bu kısımda modülün hangi porta bağlı olduğunu bilgisayarınızdaki “**Aygıt Yönetici**”sinden görüntüleyebilirsiniz.

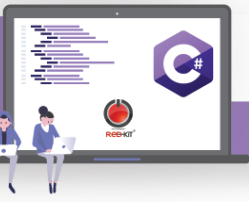


Aygıt Yöneticisi

Doğru port seçimini yaptıktan ve bağlantıyı sağladıktan sonra devreyi uzaktan kontrol edebiliriz. Bu derste önceki çalışmalardan farklı olarak devremizi uzaktan kontrol etmiş olduk. Kendi çalışmalarınızı da arayüz ile Bluetooth modülü kullanarak uzaktan haberleştirebilirsiniz.



Servo Motor Kontrol Arayüzü



## C# Kod Blokları

### Form1 Kod Blokları

```
using System;
using System.Windows.Forms;
using System.IO.Ports;

namespace arduino_servo_motor_kontrol
{
    public partial class Form1 : Form
    {
        int position;
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            string[] ports = SerialPort.GetPortNames(); //Seri portları diziye ekleme
            foreach (string port in ports)
                comboBox1.Items.Add(port);

            //TrackBar ayarları
            trackBar1.Minimum = 1;
            trackBar1.Maximum = 180;
        }

        private void button1_Click(object sender, EventArgs e)
        {
            try
            {
                if (!SerialPort.IsOpen)
                {
                    /* Seri Port Ayarları */
                    SerialPort.PortName = comboBox1.Text;
                    SerialPort.BaudRate = 9600;
                    SerialPort.Parity = Parity.None;
                    SerialPort.DataBits = 8;
                    SerialPort.StopBits = StopBits.One;
                    SerialPort.Open(); //Seri portu aç
                    label3.Text = "Bağlantı Sağlandı.";
                    label3.ForeColor = System.Drawing.Color.Green;
                    button1.Text = "KES"; //Buton1 yazısını değiştir
                }
                else
                {
                    label3.Text = "Bağlantı Kesildi.";
                    label3.ForeColor = System.Drawing.Color.Red;
                    button1.Text = "BAĞLAN"; //Buton1 yazısını değiştir
                    SerialPort.Close(); //Seri portu kapa
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message, "Hata"); //Hata mesajı
            }
        }
    }
}
```



```
private void trackBar1_Scroll(object sender, EventArgs e)
{
    position = trackBar1.Value;
    write_angle(position);
}

private void degree1_Click(object sender, EventArgs e) //30 derece
{
    position = 30;
    write_angle(position);
}

private void degree2_Click(object sender, EventArgs e) //90 derece
{
    position = 90;
    write_angle(position);
}

private void degree3_Click(object sender, EventArgs e) //120 derece
{
    position = 120;
    write_angle(position);
}

private void degree4_Click(object sender, EventArgs e) //180 derece
{
    position = 180;
    write_angle(position);
}

private void write_angle(int value)
{
    try
    {
        if (SerialPort.IsOpen)
        {
            SerialPort.WriteLine(value.ToString()); //Değeri port üzerinden gönder
            label2.Text = "Açı: " + value.ToString() + "°"; //Güncel değeri label2'ye yaz
        }
    }

    catch (Exception ex2)
    {
        MessageBox.Show(ex2.Message, "Hata"); //Hata mesajı
    }
}

private void Form1_FormClosed(object sender, FormClosedEventArgs e)
{
    if (SerialPort.IsOpen) SerialPort.Close(); //Seri port açıksa kapat
}
}
```



### Arduino Kod Blokları

```
#include <Servo.h>
#include <SoftwareSerial.h>
SoftwareSerial bt(8, 7); //RX, TX
Servo servo;
int data = 0;
int value;
void setup()
{
    bt.begin(9600); //Seri iletişimi başlat
    servo.attach(9); //Servo pin ayarlaması
}
void loop()
{
    if (bt.available() > 0) //Veri gelip gelmediğini kontrol et
    {
        data = bt.parseInt(); //Gelen veriyi data değişkenine eşitle
        if (data > 0)
        {
            value = data; //value değişkenini data değişkenine eşitle
        }
        servo.write(value); //Servo kontrolü
    }
    delay(20);
}
```









