

**T.C**

**İNÖNÜ ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Bitirme Projesi Raporu**

**Proje Adı**

ROBOT KOL

**HAZIRLAYAN**

**AD SOYAD:**HAMİT KÖROĞLU

**ÖĞRENCİ NO:**02140001042

**MALATYA**

**2019**

**Solidworks nedir?**

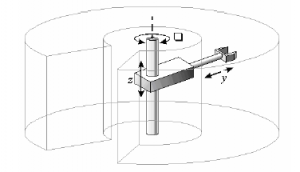
SolidWorks yenilikçi, kullanımı kolay, Windows'a özel üç boyutlu bir tasarım programıdır. SolidWorks her türlü makine, tesis, ürün tasarımında kullanıcıya Windows’un kolaylıklarını kullanarak hızlı bir şekilde çizim yapmasını sağlar.  
  
**Neler Yapılabilir?**  
  
[\*] SolidWorks kullanarak ölçülü ve vektör tabanlı üç boyutlu çizimler yapılabilir.  
[\*] Esnek yapısı sayesinde yaptığımız çalışmaları pek çok program üzerinde kullanabiliriz.  
[\*] Yaptığımız çizimler üzerinden teknik veriler alabiliriz.  
[\*] Montaj modülü sayesinde ayrı ayrı yaptığımız çalışmaları bir araya toplayabilir birlikte çalışabiliriz.  
[\*] Yaptığımız parçalar üzerinde darbe, statik v.b testler yapabilir çok ayrıntılı sonuçlara ulaşabiliriz.  
[\*] Çalışamalarımızı animasyon eklentisi sayesinde hareketlendirebiliriz.  
  
**Hangi Sektörlerde ve Nerelerde Kullanılır?**  
  
Çizim gerektiren her sektörde Solidworks kolayca ve rahatça kullanılabilir. Kullanım stili ve çalışma mantığının kolaylığı yanı sıra zengin içerikli bir program olduğu için uzun süre kullananlar bile her zaman Solidworks'ten öğrenecek bir şeyler bulabilmektedir.  
  
**PROGRAMIN GENEL İÇERİĞİ**  
  
**1. Sketcher (Taslak Oluşturma)**  
  
[\*] Profil çizimleri  
[\*] Profillere uygulanacak işlemler  
  
**2. Part Design (Parça Tasarımı)**  
  
[\*] Görüntü ayarlamaları  
[\*] Sketch tabanlı uygulamalar  
[\*] Katı modelleme  
[\*] Katı modelin translation ile koordinatların değiştirilmesi  
[\*] Sac metal şekillendirilmesi  
[\*] Yüzey oluşturma işlemleri  
[\*] Yüzey tabanlı katı oluşturma  
[\*] Parçaların imali için gerekli olan kalıplamanın uygulanması  
  
**3. Assembly Design (Montaj Tasarımı)**  
  
[\*] Parçaların montajlanması  
[\*] Kısıtlamaların oluşturulması  
[\*] Mekanizmaların hareketi  
[\*] Mekanizmalarda hareket engelleyiciler  
[\*] Mekanizmalarda bağımlı hareketler  
[\*] Montajlara Toolbox yardımı ile parça çağrımı  
[\*] Smart Mate ile hızlı parça ekleme  
  
**4. Drafting (Hazırlık)**  
  
[\*] Teknik resimlerin çıkarılması  
[\*] Görüntü oluşturma  
[\*] Ölçümlendirme  
[\*] Montaj numaralarının girilmesi  
[\*] Şekil ve ölçüsel toleranslar  
  
**5. Analiz**  
  
[\*] Cosmos Works ile basit bir parçanın lineer gerilme analizi  
[\*] Cosmos Works ile basit bir parçanın termal analizi  
[\*] Burkulma analizi  
[\*] Optimizasyon testleri  
[\*] Tasarlanan bir parçanın her türlü sınır şartları altında test edilmesi  
  
**6. Simülasyon**  
  
[\*]Cosmos Motion yardımı ile basit mekanizmanın bağlantılarının tanımlanması  
[\*]Bağlantılara sürtünme eklenmesi  
[\*]Çarpışma sonucu oluşması gereken itme momentum değerleri bulunması  
[\*]Mekanizmanın simülasyonu

**Robot Kol Nedir?**

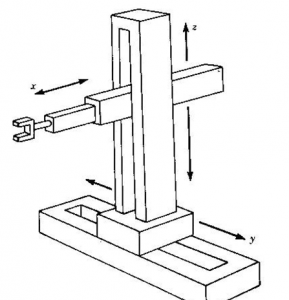
**Karmaşık bir robotun bir parçası veya mekanik parçaların devasa bütünü olabilecek, programlanabilir elektro-mekanik cihaz olarak tanımlayabiliriz. Robotik kollar endüstrinin ayrılmaz bir bütünü haline gelmiştir. Hassas ve yoğun işçilik isteyen kısımlarda kullanılırlar.Hata payı oranını azaltarak, insana duyulan ihtiyacı en az indirgeyen, üretim miktarını ve hızını en yüksek seviyelere çıkaran bu kollar, teknolojide kullanılmaya ve yarar sağlamaya devam edecektir.**

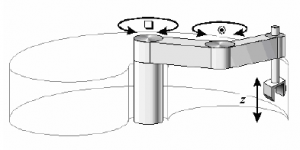
**ROBOTİK KOLLARIN ÇEŞİTLERİ**

**a)Silindirik Robot Kolu.  
Yüksek dönüş kabiliyetleri sayesinde kartezyen robot kollarına göre hareket etme kapasitesi daha yüksektir. Kolun biri dikey, diğeri yatay vaziyettedir. Bundan dolayı tut ve taşı komutlarını yerine getirirler. Ancak esnek bir kullanıma sahip değillerdir. Hidrolik, pnömatik ve elektrik motorlu çeşitleri bulunmaktadır.**

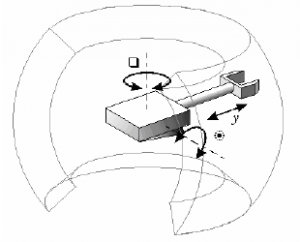
[**[](http://www.mekatronikmuhendisligi.com/wp-content/uploads/2016/02/silindirik-1.png)**](http://www.mekatronikmuhendisligi.com/wp-content/uploads/2016/02/silindirik-1.png)

**b)Kartezyen Robot Kolu  
x,y,z koordinat düzleminde her kol bir önceki kola göre dik açıyla hareket eder. Belirli noktalara ulaşımı kolay olmasına rağmen hareketi bir o kadar kısıtlıdır. Tut ve taşı komutlarını yerine getirirler. Büyük boyutlu ve ağır yükleri taşımak için uygun bir robot koludur.**

[](http://www.mekatronikmuhendisligi.com/wp-content/uploads/2016/02/kartezyen-1.png)

**c)Scara Robot Kolu  
Yüksek hız ve iş kapasitesine sahiptir. Üzerinde (eklem noktalarında ) bulunan elektrik motorları, kolun kendi ekseni etrafında dönmesini sağlar; (ağız kısmında ) bulunan pnömatik motorlar ise kola esneklik kazandırır.  
[](http://www.mekatronikmuhendisligi.com/wp-content/uploads/2016/02/scara-1.png)**

**d) Küresel Robot Kolu  
Bir diğer adı Polar Koordinatlı Robot Koludur. Programlanması zordur.  
Kollarının büyüklüğü ile hareket alanları birbirine zıttır. Yani kolun büyüklüğü arttıkça hareket alanı kısıtlanır. Kaynak, eğme, bükme,yapıştırma vb. işlemlerde kullanılır.**

[](http://www.mekatronikmuhendisligi.com/wp-content/uploads/2016/02/45rf-1.png)

**Robot Kol Kullanım Alanları**

**Sinai Robot Kollar**

Günümüzde robotik kollar endüstrinin ayrılmaz bir parçası haline gelmiştir. Robotik kollar insan gücünü en aza indirerek, hata payı oranını azaltır. Böylece üretim miktarını üst seviyelere getirerek, günümüz teknolojisine önemli derecede katkı sağlar.

**Robot Kolun Kullanım Alanları**

* Püskürtme, Boyama, Cilalama
* Toplama, Paketleme, Paletleme
* Otomasyonlu Makine Desteği
* Montaj
* Tutkallama, Yapıştırma
* Malzeme Taşıma



**Da Vinci robotik cerrahisi**

Da Vinci Robotik Cerrahi ya da diğer adıyla robot teknolojisi, tıp dünyasında cerrahinin ulaştığı en ileri teknolojidir ve başta ABD’deki önemli merkezler olmak üzere Türkiye’de de özellikle üroloji, kadın hastalıkları, genel cerrahi, KBB, göğüs cerrahisi, kalp cerrahisi ameliyatlarında kullanılmaktadır.

Da Vinci robot sadece cerrahın istediği şeyleri yapıyor yani kendi başına ameliyat yapması söz konusu değil. Başta üroloji, genel cerrahi ve kadın hastalıklarında olmak üzere pek çok alanda kullanılan ameliyat robotunun sağladığı en büyük avantajlar operasyonun başarı oranını artırması ve kanama oranını azaltması.

Ameliyat robotunu geliştiren ilk firma olan Intuitive’in meşhur ameliyat robotu “Da Vinci” ile gerçekleştirilen işlemler ‘robot yardımlı laparoskopi ameliyatı’ olarak tanımlanıyor. Yani bugüne kadar duymaya alıştığımız, ‘açık ameliyat’ ve ‘laparoskopik ameliyat’ın yanında yeni geliştirilmiş üçüncü bir ameliyat yöntemi.



Intuitive da Vinci’nin en son modeli Da Vinci SI HD’yi piyasaya sürerken, patent koruma süresinin dolmasıyla birlikte çok sayıda firmanın ürettikleri ameliyat robotlarını çok daha uygun maliyetlerle piyasaya çıkarması bekleniyor.



**Robot Kol projesi**

**Proje içeriği:**

Dış iskelet 3d yazıcı yardımıyla oluşturulacağı kararlaştırıldı.

Projenin dış iskelet çizimlerim 3d yazıcıdan çıktı alınacak şekilde yapıldı.

İç mekanizma ise 1 arduino ve bağlantıları sağlamak için ekstra bir elektronik devre kartı tasarlanacak.

Bluetooth shieldi kullanarak android cihazla arduino arasındaki iletişim sağlanacak.

Robot koldaki hareketler servo motorlar yardımıyla sağlanacak.

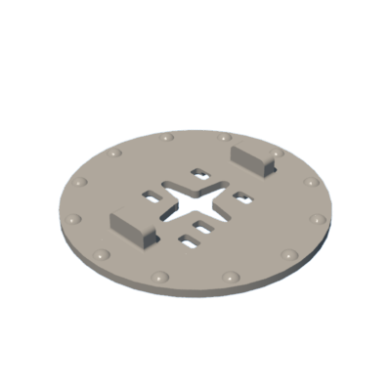
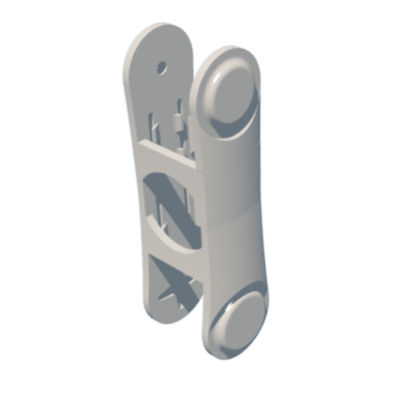
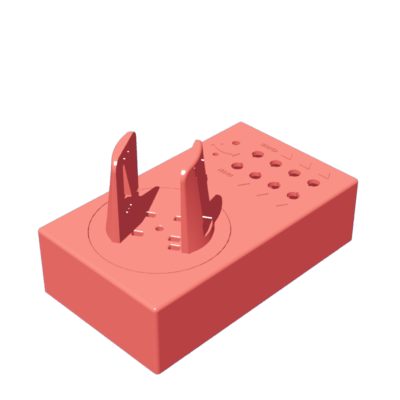
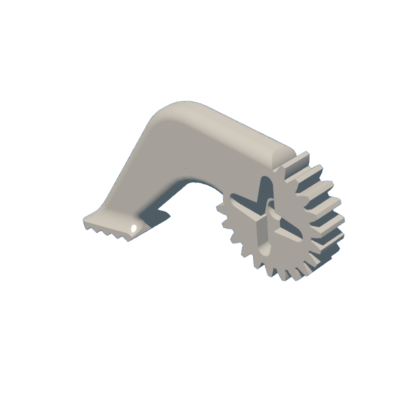
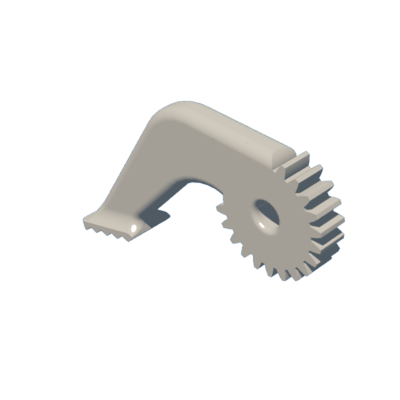
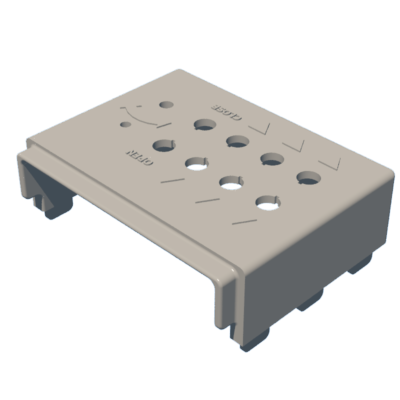
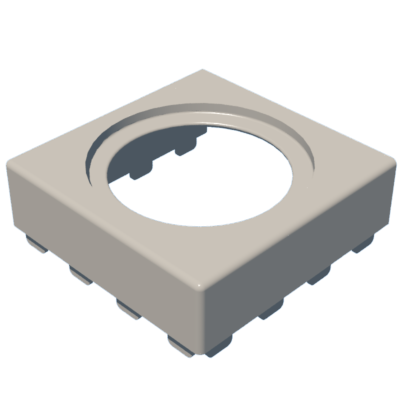
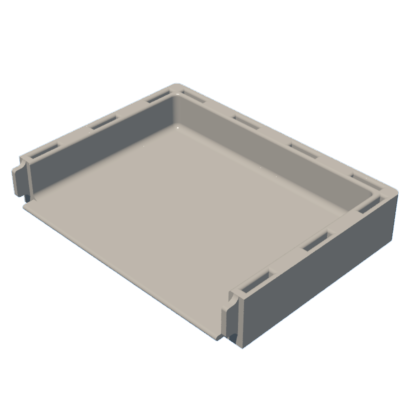
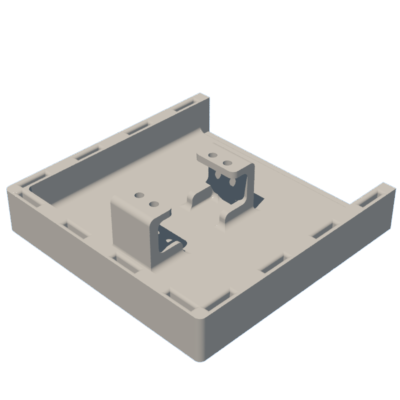
Android'de projenin kontrolü için program tasarlanacak.

Robot kol android program üzerinden gelen komutlara göre servo motorların hareketiyle kontrol edilecek.

**Kullanılacak ekipman listesi;**

* **1 adet Arduino mega/uno**
* **HC06 Arduino Bluetooth Modül**
* **4 adet Standard servo motor**
* **2 adet Mikro servo motor**
* **Baskı devre kartı**
* **Farklı değerlerde dirençler**
* **Bağlantılar için kablo ve konnektörler somun ve civatalar**
* **Dış iskelet için yaklaşık 1 kg filament**

Robot Kol Çizimleri

**PROJENİN ORTAMLARI**

1-)Android Studio

2)Arduino İDE

**Android proje geliştirme ortamı;**

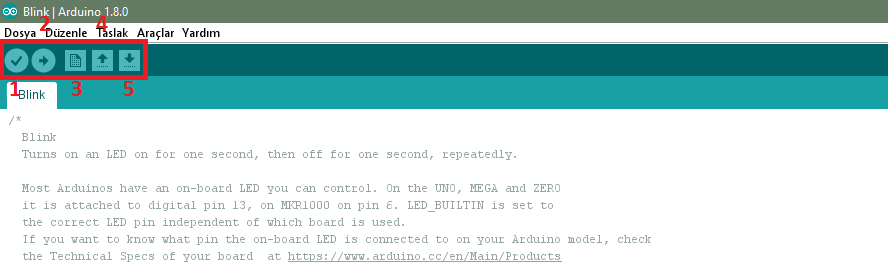
Android Studio, Android uygulamaları geliştirmek için kullanılan resmi bir yazılımdır. Resmi olması Google tarafından geliştirilmesidir.[IntelliJ IDEA](https://www.jetbrains.com/idea/) tabanlı olarak geliştirilen özel bir geliştirme ortamıdır. Sadece Android uygulamaları geliştirmek için tasarlanmıştır.

**Android Studio ile ilgili temel özellikler**

* Gradle tabanlı esnek bir yapıya sahip olması (Android Studio, yazdığımız kodları derlemek için Gradle kullanır.)
* Uygulamalarımızı kontrol edip çalıştırabileceğimiz emülatör ortamına sahip olması
* Tüm Android cihazlarına uygun uygulama geliştirme ortamına sahip olması
* GitHub entegrasyonuna sahip olması
* Kapsamlı test araçlarına sahip olması
* C++ ve NDK(Native Development Kit) desteğinin olması gibi birçok özelliğe sahiptir.

**Arduino proje geliştirme ortamı;**

Türkçe dil desteği de bulunan Arduino IDE oldukça basit, anlaşılır ve kullanışlı bir geliştirme ortamına sahip. Kısaca inceleyecek olursak;

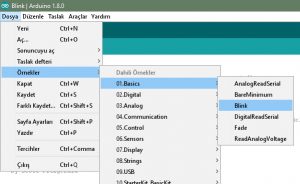
[](https://www.mobilhanem.com/wp-content/uploads/2017/01/Ekran-Al%C4%B1nt%C4%B1s%C4%B1-1.png)

Yukarıdaki görüntüye göre;

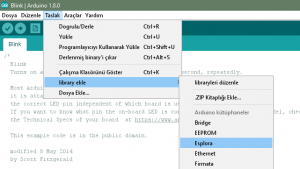
Kontrol et: Kodumuz da bir yazım hatasının bulunup bulunmadığını kontrol etmemizi,

1. Yükle: Kodumuzu önce kontrol edip ardından aygıtımıza yüklememizi,
2. Yeni: Yeni bir .ino dosyası oluşturmamızı,
3. Aç: Kayıtlı bir dosyayı açmamızı,
4. Kaydet: Kodumuzun son halini kaydetmemizi sağlamaktadır.

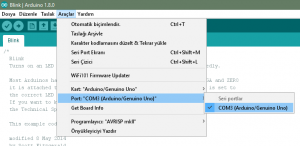
Ayrıca menümüzden bahsedecek olursak;

[](https://www.mobilhanem.com/wp-content/uploads/2017/01/dosya-1.png)

**Dosya > Örnekler** ile bizler için hazırlanmış çok sayıda örnek uygulamaya erişebiliriz.

[](https://www.mobilhanem.com/wp-content/uploads/2017/01/ekle.png)

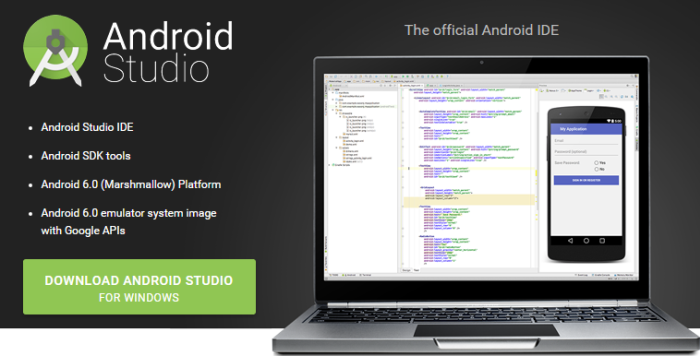
**Taslak > Library Ekle** ile hali hazırda Arduino IDE ile gelen ve kendi oluşturabileceğimiz kütüphaneleri projemize ekleyebiliriz.

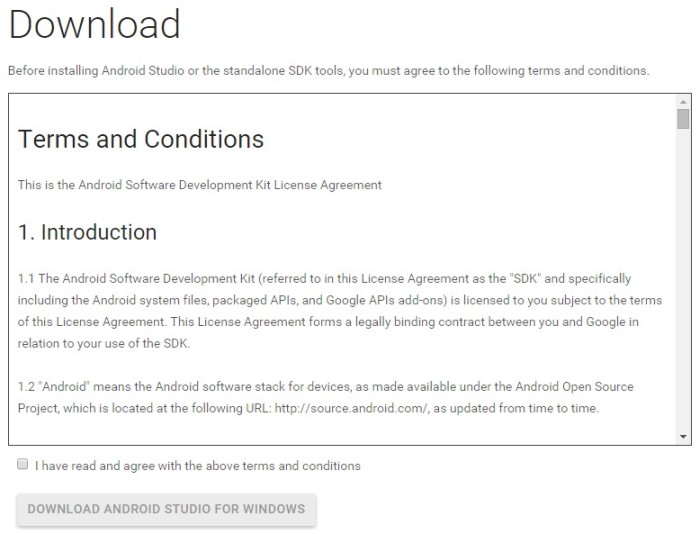
[](https://www.mobilhanem.com/wp-content/uploads/2017/01/port.png)**Araçlar > Kart** ile kullanmış olduğumuz Arduino kartımızın modelini seçebilir,

**Araçlar > Port** ile bilgisayarımıza bağlı olan kullanmak istediğimiz Arduino kartlarını seçebiliriz.

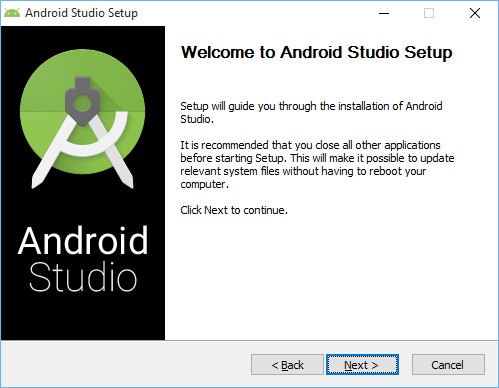
**Android Studio Kurulumu**

Öncelikle https://developer.android.com/studio/ indirme sayfasına gidiyoruz:

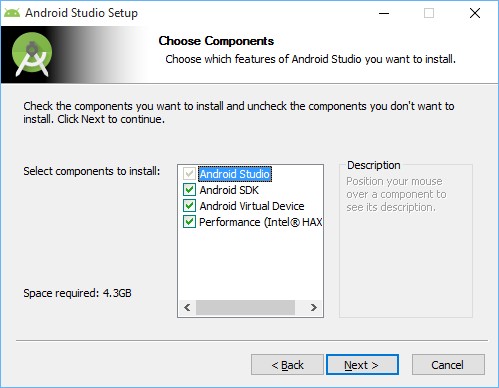
Yeşil renkli "Download Android Studio for Windows" düğmesine tıkladıktan sonra lisans sözleşmesi karşımıza gelecek. Sözleşmeyi okuduktan sonra, kabul etmek için "I Agree" düğmesine basmamız gerekiyor.



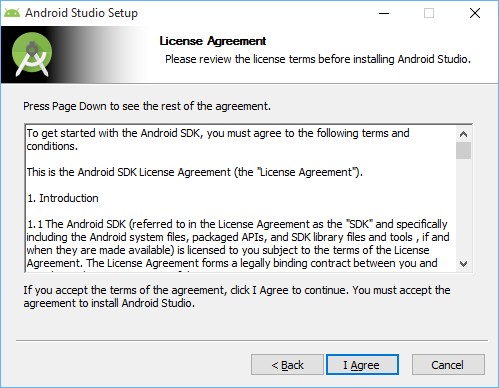
Sözleşmeyi onayladıktan sonra indirme işlemi otomatik olarak başlayacak. İndirme işlemi tamamlandıktan sonra dosyamıza çift tıklıyoruz. UAC etkinse, Windows öncelikle bizden cihaz üzerinde kurulu yönetimsel izinleri isteyecek. Yönetici izni verildikten sonra, Android Studio kurulumu için ilk adım görülecek. Next düğmesine basarak kurulum işlemine başlıyoruz.



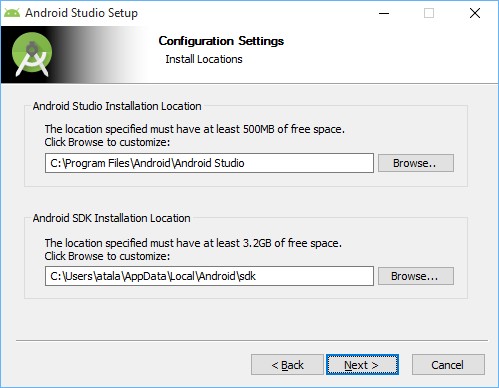
Next düğmesi sonrası karşımıza çıkacak ekranda yüklemek istediğimiz bileşenleri seçiyoruz ve tekrar Next düğmesine basıyoruz.



Next düğmesine bastıktan sonra yine lisans sözleşmesi karşımıza gelecek. Sözleşmeyi okuduktan sonra eğer kabul ediyorsak "I Agree" düğmesine basmamız gerekiyor.

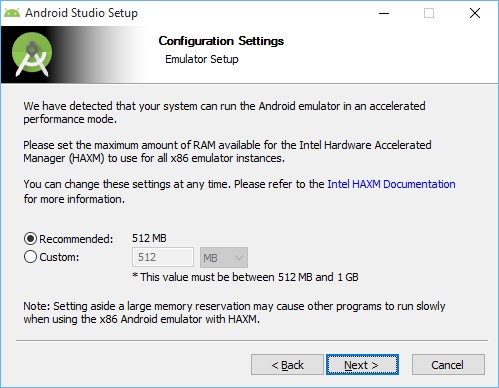


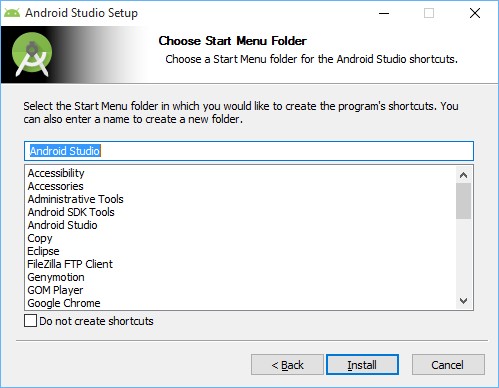
Sözleşmeyi onayladıktan sonra karşımıza Android Studio'nun ve SDK'nın nereye yükleneceğini soran bir pencere gelecek. Uygun yerleri belirttikten sonra tekrar "Next" düğmesine basıyoruz.



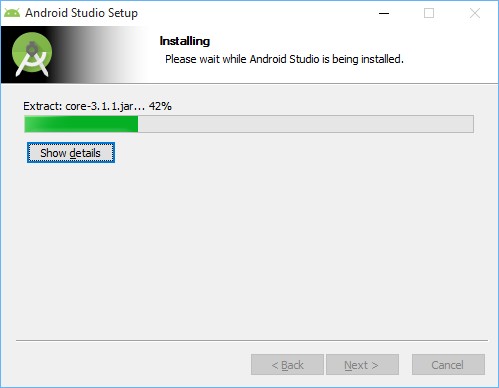
Eğer Intel Donanım Hızlandırıcı'ya sahipseniz, Bir sonraki adımda ise karşımıza Android uygulamanız için kullanabileceğiniz emülatörün kurulum ayarlarının bulunduğu ekran çıkacak. Buradan emülatörün kullanabileceği bellek miktarını ayarlayabilirsiniz.

Intel Donanım Hızlandırıcı desteği bulunmayan bilgisayarlarda, bu pencere görüntülenmeyip, doğrudan bir sonraki adıma geçilmektedir.

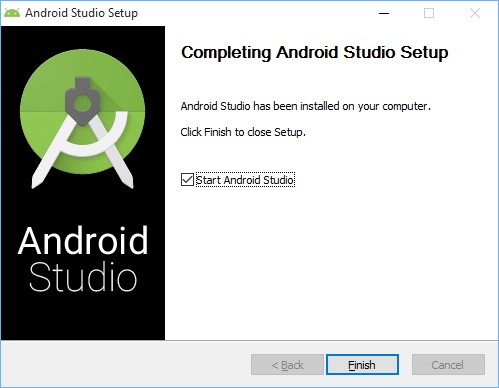


Android Studio'nun "Başlat" çubuğunda kısayol olarak gözükmesini istemiyorsak "Do not create shortcuts" seçeneğini işaretliyoruz ve "Install" düğmesine basıyoruz.

Ardından kurulum işlemimiz başlıyor.

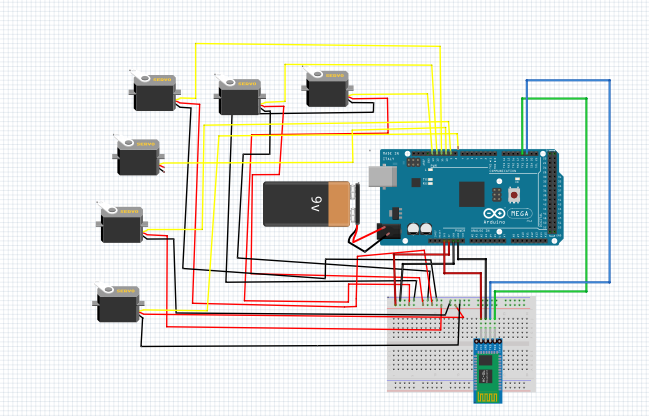


Kurulum bittikten sonra "Next" düğmesine tıklıyoruz. Çıkan pencereden "Finish" düğmesine tıklıyoruz ve kurulumu sonlandırıyoruz.



**PROJENİN ELEKTRONİK DEVRESİNİN KURULMASI**

Robot kolumuzun hareketleri servo motorlar yardımıyla kontrol edileceği için servo motor bluetooth sensörü ve arduinonun bağlantıları aşağıdaki şekildeki gibi yapılır;



**Arduino Kodu;**

**#include <SoftwareSerial.h> // TX RX library bluetooth modülü için eklenir.**

**#include <Servo.h> // servo kütüphanesi eklenir**

**Servo myservo1, myservo2, myservo3, myservo4,myservo5,myservo6; // servo motorlarımız tanımlanır.**

**void setup()**

**{**

**myservo1.attach(8); // bağlandığı pinler belirlenir.**

**myservo2.attach(9);**

**myservo3.attach(10);**

**myservo4.attach(11);**

**myservo5.attach(12);**

**myservo6.attach(13);**

**Serial1.begin(9600);**

**}**

**void loop()**

**{**

**if(Serial1.available()>= 2 )//bluetoothtan okunan bir değer varsa eğer**

**{**

**unsigned int servopos = Serial1.read();**

**unsigned int servopos1 = Serial1.read();**

**unsigned int realservo = (servopos1 \*256) + servopos;**

**Serial1.println(realservo);**

**if (realservo >= 1000 && realservo <1180) {**

**int servo1 = realservo;**

**servo1 = map(servo1, 1000, 1180, 0, 180);**

**myservo1.write(servo1);**

**delay(10);**

**}**

**if (realservo >= 2000 && realservo <2180) {**

**int servo2 = realservo;**

**servo2 = map(servo2, 2000, 2180, 0, 180);**

**myservo2.write(servo2);**

**delay(10);**

**}**

**if (realservo >= 3000 && realservo <3180) {**

**int servo3 = realservo;**

**servo3 = map(servo3, 3000, 3180, 0, 180);**

**myservo3.write(servo3);**

**delay(10);**

**}**

**if (realservo >= 4000 && realservo <4180) {**

**int servo4 = realservo;**

**servo4 = map(servo4, 4000, 4180, 0, 180);**

**myservo4.write(servo4);**

**delay(10);**

**}**

**if (realservo >= 5000 && realservo <5180) {**

**int servo5 = realservo;**

**servo5 = map(servo5, 5000, 5180, 0, 180);**

**myservo5.write(servo5);**

**delay(10);**

**}**

**if (realservo >= 6000 && realservo <6180) {**

**int servo6 = realservo;**

**servo6 = map(servo6, 6000, 6180, 0, 180);**

**myservo6.write(servo6);**

**delay(10);}}}**

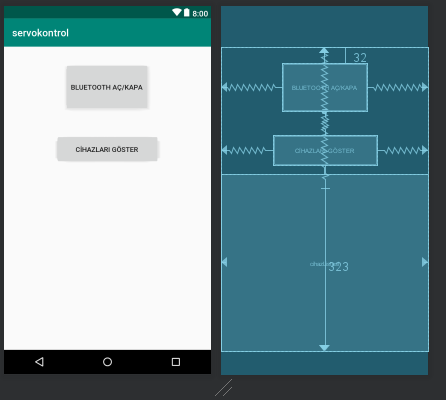
**Android Kısmı;**

**Projemizde android kısmında ise android studio'da yeni bir empty project oluştururuz.**

**Projemiz oluştuğunda androidmanifest kısmına gelip bluetooth kullanacağımız için kullanıcıdan bluetooth izni almamız gerekir.Android manifeste şu 2 kodu yazarız.**

<uses-permission android:name="android.permission.BLUETOOTH" />  
<uses-permission android:name="android.permission.BLUETOOTH\_ADMIN" />

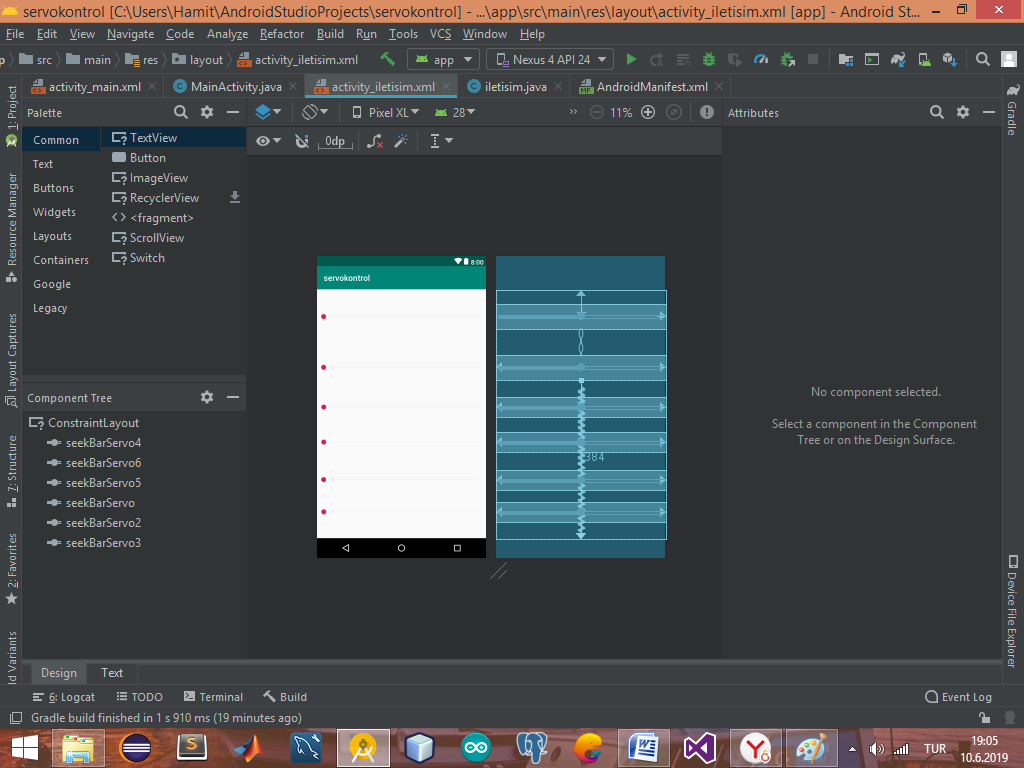
**Mainactivity tasarımımızı şu şekilde yapıyoruz;**

****

**Daha sonrada MainActivity.java içerisine aşağıdaki kodları yazıyoruz**

package com.example.servokontrol;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.bluetooth.BluetoothAdapter;  
import android.bluetooth.BluetoothDevice;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.ListView;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import java.util.ArrayList;  
import java.util.Set;  
  
public class MainActivity extends AppCompatActivity {  
  
 BluetoothAdapter myBluetooth;//kendi android cihazımızı temsil eder.  
 private Set<BluetoothDevice> eslesılmısCıhaz;//etraftaki android cihazları temsil eder.  
 Button btAcKapa,cihazlariGoster;  
 ListView cihazListesi;  
 public static String ekstra\_adres="cihaz\_adres";  
 ArrayAdapter<String> adapter;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 myBluetooth=BluetoothAdapter.getDefaultAdapter();//cihazın bluetoothunun olup olmadıgını gösterir.  
 btAcKapa=findViewById(R.id.btnOnOff);  
 cihazlariGoster=findViewById(R.id.cihazGoster);  
 cihazListesi=findViewById(R.id.cihazListesi);  
  
  
 btAcKapa.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 btDurumDegıs();  
 }  
  
  
 });  
 cihazlariGoster.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 fonkCihazlariListele();  
 }  
 });  
 }  
  
 private void btDurumDegıs()  
 {  
 //bluetooth özelliğinin açık olup olmadıgını test edicez.  
 if(myBluetooth==null)  
 {  
 Toast.*makeText*(getApplicationContext(),"Bluetooth cihazı yok.",Toast.*LENGTH\_SHORT*).show();  
 }  
 if(!myBluetooth.isEnabled())  
 {  
 //pasifse aktif et  
 Intent enableBtIntent=new Intent(BluetoothAdapter.*ACTION\_REQUEST\_ENABLE*);  
 startActivity(enableBtIntent);  
 }  
 if(myBluetooth.isEnabled())  
 {  
 //aktifse pasif et  
 myBluetooth.disable();  
 }  
 }  
  
  
 private void fonkCihazlariListele()  
 {  
 //cihazları listeme işlemi  
 eslesılmısCıhaz=myBluetooth.getBondedDevices();  
 ArrayList list=new ArrayList();  
 if(eslesılmısCıhaz.size()>0)  
 {//liste boyutu 0 dan büyükse yani daha önceden bağlantı yapılmışsa  
 for (BluetoothDevice bt:eslesılmısCıhaz)  
 {//ismini ve adresini listeye eekle  
 list.add(bt.getName()+"\n"+bt.getAddress());  
 }  
 }  
 else  
 {  
 Toast.makeText(getApplicationContext(),"Eşleşmiş cihaz yok",Toast.LENGTH\_SHORT).show();  
 }  
 final ArrayAdapter adapter=new ArrayAdapter(this,android.R.layout.simple\_list\_item\_1,list);//listelemek için array adapter oluşturduk  
 cihazListesi.setAdapter(adapter);  
 cihazListesi.setOnItemClickListener(cihazSec);  
  
 }  
 public AdapterView.OnItemClickListener cihazSec=new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id)  
 {  
 String bilgi=((TextView) view).getText().toString();  
 String adres=bilgi.substring(bilgi.length()-17);  
 Intent iletisimIntent=new Intent(MainActivity.this,iletisim.class);  
 iletisimIntent.putExtra(ekstra\_adres,adres);//herhangi bir veriyi bir classtan diğerine aktarmayı sağlar.  
 startActivity(iletisimIntent);  
 }  
 };  
  
  
}

Daha sonra kontrol ekranımızı tasarlarız.Bunun için iletişim adında bir activity oluştururuz.İletişim tasarım kısmını ise aşağıdaki gibi yaparız.



İletişim.java içerisine ise Bağlantı threadlerini oluşturmak için şu kodları yazarız.Değişken olarak aşağıdaki ddeğişkenler tanımlanır.

String adres=null;  
private ProgressDialog progress;  
BluetoothAdapter myBluetooth=null;  
  
BluetoothSocket btSocket=null;  
BluetoothDevice remoteDevice;  
BluetoothServerSocket mmServer;  
  
private boolean isbtConnected=false;  
static final UUID *myUUID*=UUID.*fromString*("00001101-0000-1000-8000-00805F9B34FB");

Oncreate metodumuzun dışına ise;

private void Disconnect()  
{  
 if(btSocket!=null)  
 {  
 try  
 {  
 btSocket.close();  
 }catch (IOException e)  
 {  
 //msg ("error");  
 }  
 }  
 finish();  
}  
@Override  
public void onBackPressed()  
{  
 super.onBackPressed();  
 Disconnect();  
}  
  
  
  
private class btBaglan extends AsyncTask<Void,Void,Void>  
{  
 private boolean connectSuccess=true;  
 @Override  
 protected void onPreExecute()  
 {  
 //önceki yapması gereken işlem  
 progress=ProgressDialog.*show*(iletisim.this,"bağlanıyor...","lütfen bekleyiniz.");  
 }  
 @Override  
 protected Void doInBackground(Void... devices)  
 {  
 //ve sürekli yapması gereken işlemler  
 try {  
 if(btSocket==null||!isbtConnected)  
 {  
 myBluetooth=BluetoothAdapter.*getDefaultAdapter*();  
 BluetoothDevice cihaz=myBluetooth.getRemoteDevice(adres);  
 btSocket=cihaz.createInsecureRfcommSocketToServiceRecord(*myUUID*);  
 BluetoothAdapter.*getDefaultAdapter*().cancelDiscovery();  
 btSocket.connect();  
 }  
 }catch (IOException e)  
 {  
 connectSuccess=false;  
 }  
 return null;  
 }  
 @Override  
 protected void onPostExecute(Void result)  
 {  
 //daha sonraa yapması gereken işlemler  
 super.onPostExecute(result);  
 if(!connectSuccess)  
 {  
 Toast.*makeText*(getApplicationContext(),"Bağlantı hatası tekrar deneyin.",Toast.*LENGTH\_SHORT*).show();  
 finish();  
 }else  
 {  
 Toast.*makeText*(getApplicationContext(),"Bağlantı başarılı",Toast.*LENGTH\_SHORT*).show();  
 isbtConnected=true;  
  
 }  
 progress.dismiss();  
 }

seekbarlara erişim ve veri bluetooth hattına veri göndermek için ise oncreate dışında şu değişkenler tanımlanır.

SeekBar servoKontrol;  
SeekBar servoKontrol2;  
  
SeekBar servoKontrol3;  
SeekBar servoKontrol4;  
SeekBar servoKontrol5;  
SeekBar servoKontrol6;  
int servo1Deger;  
int servo2Deger;  
int servo3Deger;  
int servo4Deger;  
int servo5Deger;  
int servo6Deger;

Oncreate metodumuz içine ise şu kodlar yazılır;

Intent newInt=getIntent();  
adres=newInt.getStringExtra(MainActivity.*ekstra\_adres*);  
new btBaglan().execute();  
  
servoKontrol=findViewById(R.id.*seekBarServo*);  
servoKontrol2=findViewById(R.id.*seekBarServo2*);  
servoKontrol3=findViewById(R.id.*seekBarServo3*);  
servoKontrol4=findViewById(R.id.*seekBarServo4*);  
servoKontrol5=findViewById(R.id.*seekBarServo5*);  
servoKontrol6=findViewById(R.id.*seekBarServo6*);  
servo1Deger=servoKontrol.getProgress();  
servo2Deger=servoKontrol2.getProgress();  
servo3Deger=servoKontrol3.getProgress();  
servo4Deger=servoKontrol4.getProgress();  
servo5Deger=servoKontrol5.getProgress();  
servo6Deger=servoKontrol6.getProgress();  
servoKontrol.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
 servo1Deger=progress;  
 // servo1Deger=servo1Deger\*10;  
 if (btSocket!=null)  
 {  
 try  
 {  
 btSocket.getOutputStream().write(servo1Deger+1000);  
 }  
 catch (IOException e)  
 {  
  
 }  
 }  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
});  
  
servoKontrol2.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
 servo2Deger=progress;  
 if (btSocket!=null)  
 {  
 try  
 {  
 btSocket.getOutputStream().write(servo2Deger+2000);  
 }  
 catch (IOException e)  
 {  
  
 }  
 }  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
});  
  
  
servoKontrol3.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
 servo3Deger=progress;  
 if (btSocket!=null)  
 {  
 try  
 {  
 btSocket.getOutputStream().write(servo2Deger+3000);  
 }  
 catch (IOException e)  
 {  
  
 }  
 }  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
});  
  
servoKontrol4.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
 servo4Deger=progress;  
 if (btSocket!=null)  
 {  
 try  
 {  
 btSocket.getOutputStream().write(servo2Deger+4000);  
 }  
 catch (IOException e)  
 {  
  
 }  
 }  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
});  
  
servoKontrol5.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
 servo5Deger=progress;  
 if (btSocket!=null)  
 {  
 try  
 {  
 btSocket.getOutputStream().write(servo2Deger+5000);  
 }  
 catch (IOException e)  
 {  
  
 }  
 }  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
});  
  
servoKontrol6.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
 servo6Deger=progress;  
 if (btSocket!=null)  
 {  
 try  
 {  
 btSocket.getOutputStream().write(servo2Deger+6000);  
 }  
 catch (IOException e)  
 {  
  
 }  
 }  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
});