**T.C.**

**KOCAELİ ÜNİVERSİTESİ**

**MEKATRONİK MÜHENDİSLİĞİ**

**ROBOTİK VE OTOMASYON SİSTEMLERİ**



**ÖDEV-1**

**HAZIRLAYAN:**AHMET FURKAN HIDIR

**ÖĞRENCİ NO:** 150224015

**İçindekiler**

Kuka Firması Hakkında3

KUKA KR5 SIXX R850 Robot Tanıtımı4

Koordinat Sistemi Yerleştirilmesi5

D-H Tablosunun Çıkarılması7

Transformasyon Matrislerini Elde Edilmesi9

Matlab-GUI Arayüzü10

Matlab-GUI Editör Kodları11

Kaynakça22

**Kuka Firması Hakkında**



Şekil-1

KUKA dünya çapında edüstriyel robot üreticisi ve fabrika otomasyonu için çözümler üreten bir Alman şirketidir. KUKA Robotics Corporation’ın dünya genelinde, Amerika Birleşik Devletleri, Kanada, Meksika, Brezilya, Çin, Japonya, Kore, Tayvan, Hindistan ve çoğu Avrupa ülkesi başta olmak üzere birçok satış ve servis iştiraki olan 25 iştiraki bulunmaktadır. Şirket ismi, KUKA, Keller und Knappich Augsburg’un baş hariflerinin kısaltması olup, üretimi yapılan tüm endüstriyel robotlar ve ürünlerde geçerli olan tescilli ticari marka ismidir.[1]

Şirket, Johann Josef Keller ve Jacob Knappich tarafından Augsburg, Almanya’da 1898 yılında kurulmuştur. İlk başlarda bina ve sokak aydınlatmasına odaklanılmış ancak KUKA çok geçmeden 1966 yılında komünal araçlarda pazar lideri olma hedefiyle kaynak ekipmanları ve çözümleri, büyük konteynırlar gibi ürünlere yoğunlaşmıştır. KUKA, 1973 yılında dünyanın ilk endüstriyel robotu FAMULUS’u üretmiştir.[[2]](http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvS1VLQSNjaXRlX25vdGUtMg) O tarihte şirket Quandt grup bünyesindedir. Fakat, Quandt 1980 yılında şirketten çekilmiş ve şirket halka açık olarak kurulmuştur. 1995 yılında KUKA Robotics Corporation ve KUKA Schweißanlagen GmbH (şu anki adıyla KUKA Systems GmbH) olarak ikiye ayrılmıştır. KUKA bugün endüstriyel imalat süreçlerinin otomasyonu için yenilikçi çözümler üretmeye kendini adamıştır. Şu anda şirket, halka açık olan KUKA AG’ye (önceki ismiyle IWKA Group) aittir.

Şirketin genel merkezi [Almanya’nın](http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvQWxtYW55YQ) [Augsburg](http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvQXVnc2J1cmc" \o "Augsburg) şehrindedir. 30.09.2012 tarihi itibarıyla KUKA 3.150 kişiye istihdam sağlamaktadır. Şirket müşterileri çoğunlukla [otomotiv](http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvT3RvbW90aXZfZW5kJUMzJUJDc3RyaXNp) sanayinde yer almasına rağmen genel sanayi gibi diğer sektörlerde de giderek artan bir oranda yeni müşterileri portföyüne eklemektedir.[2]

**KUKA KR5 SIXX R850**



Şekil-2:Kuka KR-5 Sixx R850 robot

**Minimum alanda azami hız:** KUKA Küçük Robotlar, en yüksek düzeyde çağrı yapan neredeyse tüm uygulamalar için idealdir. Hassas bileşenleri kullanmak için hassas ve hızlıdır.

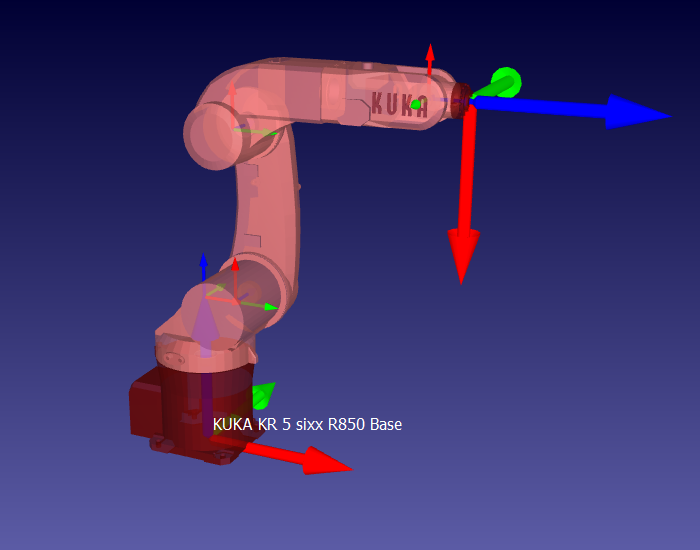
Özel aletler, tüm enerji ve sıvı besleme hatları, robotlara entegre edilmiştir,, böylece eksen hareket aralıklarında bir kısıtlama yoktur.

Tüm Küçük Robotlar, isteğe bağlı olarak, koruma sağlayan bir toz ve sıçrama korumalı varyant olarak da mevcuttur.Daha fazla dayanıklılık, kullanıcı dostu ve servis kalitesi kanıtlanmış KUKA denetleyicisidir. Bu, diğer KUKA modelleriyle sistem uyumluluğunu güvence altına alır, aynı anda tekdüze bir kontrol konsepti sağlar.Tüm ürün yelpazesi, hızlı devreye alma ve basit bakım sağlar. .[3]

Kuka firması tarafından endüstriyel amaçla üretilmiş olan KR-5 sixx R8500 modeli 6 serbestlik derecesine sahip bir robot koludur. Standart model olarak zemine veya tavana monte edilmiş olsun ya da Waterproof(WP ),Cleanroom(CR) gibi özel koşullar için olsun

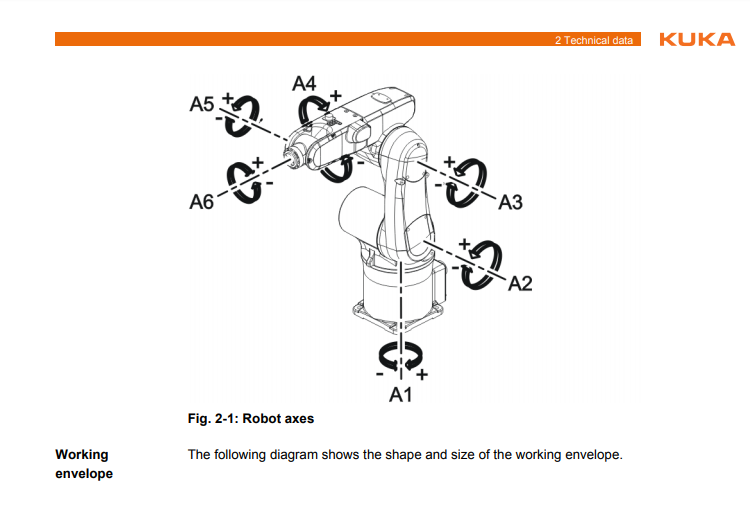
KR-5 sixx size yüksek hassasiyet sunar.Düşük ağırlığı sayesinde işini hızlı ve esnek bir şekilde gerçekleştirir.Ağırlığı yaklaşı 29 kg olup 5kg yük taşıma kapasitesine sahiptir.

Koordinat Sistemi Yerleştirilmesi



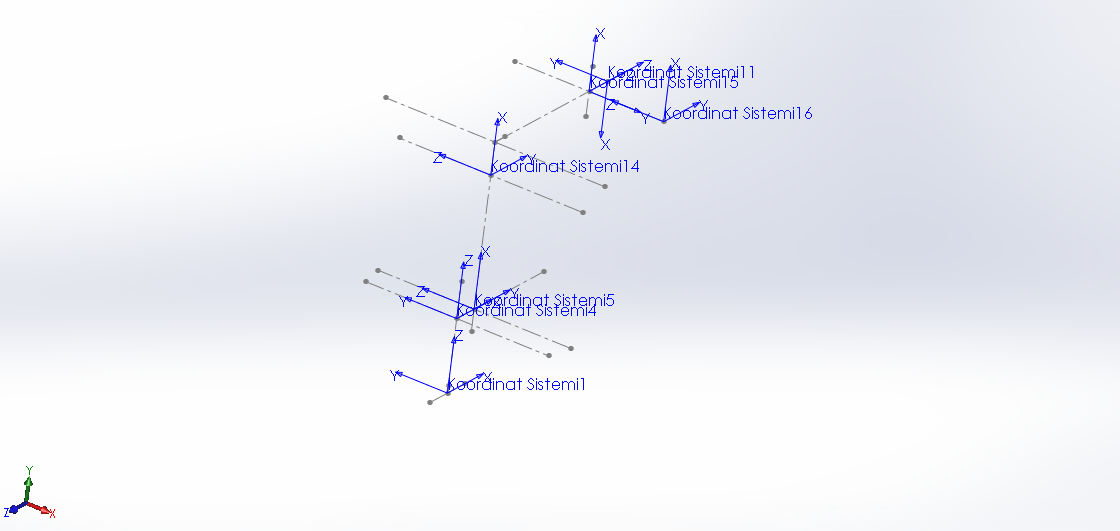
Şekil-3:RoboDK proğramından eksenlerin çıkarılması

Robotun dönme eksenleri robot dökümanlarından belirlenir.



Şekil-4:Robot eksenleri

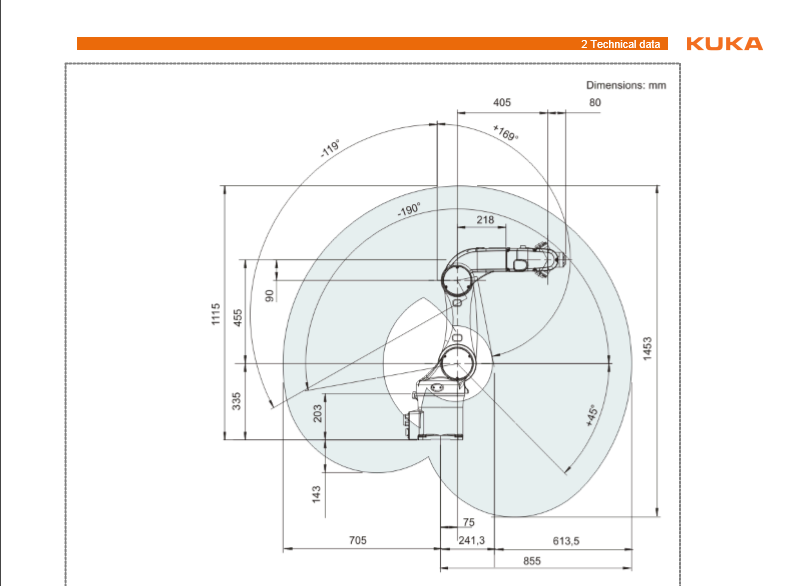
Derste ve kitapta öğrendiğimiz bilgilerle robotun eksenlerini elle çıkarabiliriz.Solidworks proğramı kullanarak ekesenler yerleştirildi.



Şekil-5:Solidworks çizimi

D-H Tablosunun Çıkarılması

Eklem uzunluklarını robot datasheet kullanarak bulabiliriz.



Şekil-6:Robot teknik resmi

Datasheet’e göre; L1=203 mm

L2=75 mm

L3=365 mm

L4=90 mm

L5=405 mm

L6=80 mm

bulunur.

Robot dinamiği kitabından yararlanılarak D-H tablosu değişkenleri bulunur.

Uzuv uzunluğu a*i* : x*i* boyunca o*i* ’den x*i* ve z*i*-1’nin kesişimlerine olan uzaklık

Uzuv ofseti d*i* : z*i*-1 boyunca o*i*-1’den x*i* ve z*i*-1’in kesişimlerine olan uzaklık.

Uzuv bükümü α*i* : x*i* etrafında z*i*-1 z*i* arasındaki açı

Eklem açısı θ*i* : z*i*-1 etrafında x*i*-1 x*i* arasındaki açı

D-H Tablosu;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i |  |  |  |  | D |
| 1 | 0 | 0 |  | +0 |  |
| 2 | -90 |  | 0 | -90 |  |
| 3 | 0 |  | 0 | +0 |  |
| 4 | -90 |  |  | +0 |  |
| 5 | +90 | 0 | 0 | +0 |  |
| 6 | -90 | 0 |  | +0 |  |

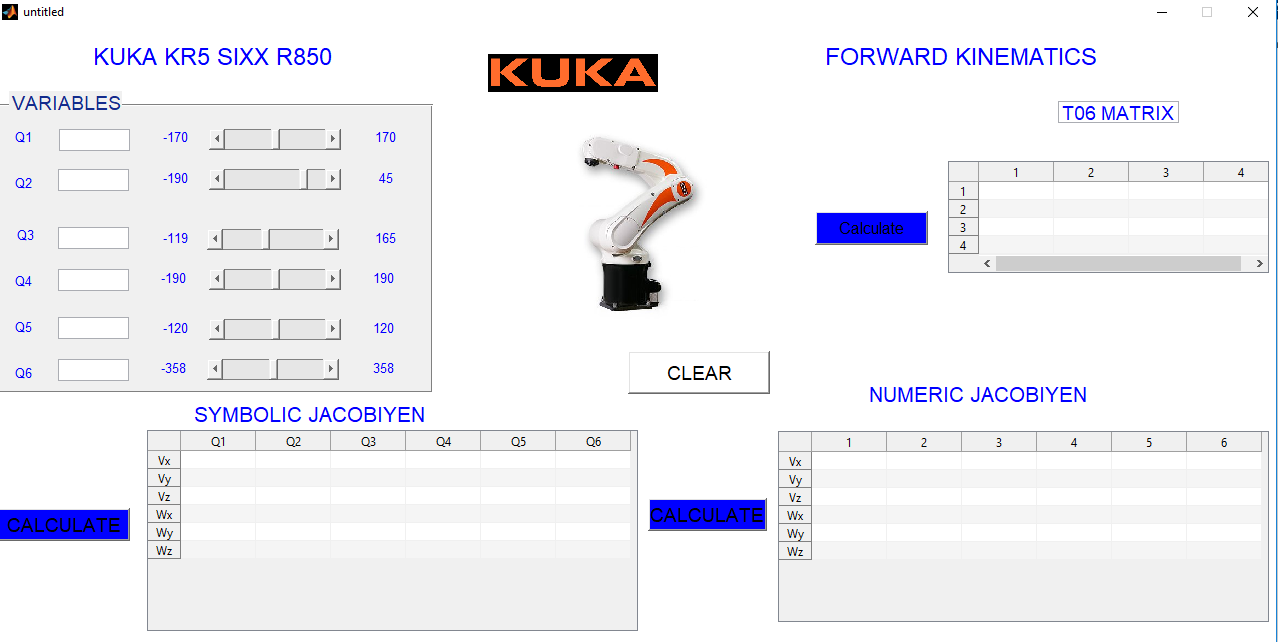
Şekil-6

Transformasyon Matrislerini Elde Edilmesi

D-H tablosundaki değerlere bakılarak her bir eklemin transformasyon matrisi elde edilmiştir.

Matlab-GUI Arayüzü

Matlab’a yukarıda çıkardığımız transformasyon matrislerini kullanarak ve tüm bulduğumuz transformasyon matrislerini çarparak T06 matrisini elde edilir. İleri kinematik matrislerini kullanarak sembolik Jacobiyen matrisleri bulunur sonra değişken değerlerimize göre nümerik Jacobiyen matrisi elde edilir.



Şekil-7:Gui Arayüzü

Matlab-GUI Editör Kodları

function varargout = untitled(varargin)

gui\_Singleton = 1;

gui\_State = struct('gui\_Name', mfilename, ...

'gui\_Singleton', gui\_Singleton, ...

'gui\_OpeningFcn', @untitled\_OpeningFcn, ...

'gui\_OutputFcn', @untitled\_OutputFcn, ...

'gui\_LayoutFcn', [] , ...

'gui\_Callback', []);

if nargin && ischar(varargin{1})

gui\_State.gui\_Callback = str2func(varargin{1});

end

if nargout

[varargout{1:nargout}] = gui\_mainfcn(gui\_State, varargin{:});

else

gui\_mainfcn(gui\_State, varargin{:});

end

function untitled\_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;

axes(handles.axes1);

imshow('C:\Users\afurk\Desktop\robot\_ödev1\KUKA\_logo.png'); %logo arayuze aktarildi.

axes(handles.axes2);

imshow('C:\Users\afurk\Desktop\robot\_ödev1\KUKA\_KR\_5\_sixx\_R850.jpg');%robot resmini arayuze aktarıldı

guidata(hObject, handles);

function varargout = untitled\_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;

function edit1\_Callback(hObject, eventdata, handles)

function edit1\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function slider\_q1\_Callback(hObject, eventdata, handles)

%Sliderdan alinan datalar texte aktarılmasi

angle=get(hObject,'Value');

set(handles.edit\_q1,'string',num2str(angle));

guidata(hObject,handles);

function slider\_q1\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function edit2\_Callback(hObject, eventdata, handles)

function edit2\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit3\_Callback(hObject, eventdata, handles)

function edit3\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit\_q1\_Callback(hObject, eventdata, handles)

edit=get(hObject,'string');

set(handles.slider,'value',str2num(edit));

guidata(hObject,handles);

function edit\_q1\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit5\_Callback(hObject, eventdata, handles)

function edit5\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit6\_Callback(hObject, eventdata, handles)

function edit6\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function slider2\_Callback(hObject, eventdata, handles)

function slider2\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function slider3\_Callback(hObject, eventdata, handles)

function slider3\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function slider4\_Callback(hObject, eventdata, handles)

function slider4\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function slider5\_Callback(hObject, eventdata, handles)

function slider5\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function slider6\_Callback(hObject, eventdata, handles)

function slider6\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function clear\_Callback(hObject, eventdata, handles)

%Datalarin temizlenmesi

set(handles.edit\_q1,'String','')

set(handles.edit\_q2,'String','')

set(handles.edit\_q3,'String','')

set(handles.edit\_q4,'String','')

set(handles.edit\_q5,'String','')

set(handles.edit\_q6,'String','')

set(handles.trans\_table,'Data','') %tablodaki degerler silindi.

set(handles.table\_jac,'Data','')

set(handles.symbol\_table,'Data','')

function calculate\_Callback(hObject, eventdata, handles)

L1=203; L2=75; L3=365; L4=90; L5=405; L6=80; %Robotun eklemlerinin uzunlukları

q1= str2num(get(handles.edit\_q1,'String')); %Edit texteki stringi q1 e aktarimi

q2= str2num(get(handles.edit\_q2,'String'));

q3= str2num(get(handles.edit\_q3,'String'));

q4= str2num(get(handles.edit\_q4,'String'));

q5= str2num(get(handles.edit\_q5,'String'));

q6= str2num(get(handles.edit\_q6,'String'));

if q1>=-170 && q1<=170 %Robotun sinirlari

T01 = [cosd(q1) -sind(q1) 0 0 ;

sind(q1) cosd(q1) 0 0 ;

0 0 1 L1 ;

0 0 0 1 ]

else

msgbox('Robot Cannot Reach to Position!!!)','ERROR','Error');

end

if q2>=-190 && q2<=45

T12 = [cosd(q2-90) -sind(q2-90) 0 L2 ;

0 0 1 0 ;

-sind(q2-90) -cosd(q2-90) 0 0 ;

0 0 0 1 ]

else

msgbox('Robot Cannot Reach to Position!!!)','ERROR','Error');

end

if q3>=-119 && q3<=165

T23 = [cosd(q3) -sind(q3) 0 L3 ;

sind(q3) cosd(q3) 0 0 ;

0 0 1 0 ;

0 0 0 1 ]

else

msgbox('Robot Cannot Reach to Position!!!)','ERROR','Error');

end

if q4>=-190 && q4<=190

T34 = [cosd(q4) -sind(q4) 0 L4 ;

0 0 1 L5 ;

-sind(q4) -cosd(q4) 0 0 ;

0 0 0 1 ]

else

msgbox('Robot Cannot Reach to Position!!!)','ERROR','Error');

end

if q5>=-120 && q5<=120

T45 = [cosd(q5) -sind(q5) 0 0 ;

0 0 -1 0 ;

sind(q5) cosd(q5) 0 0 ;

0 0 0 1 ]

else

msgbox('Robot Cannot Reach to Position!!!)','ERROR','Error');

end

if q6>=-358 && q6<=358

T56 = [cosd(q6) -sind(q6) 0 0 ;

0 0 1 L6 ;

-sind(q6) -cosd(q6) 0 0 ;

0 0 0 1 ]

else

msgbox('Robot Cannot Reach to Position!!!)','ERROR','Error');

end

T06 = T01\*T12\*T23\*T34\*T45\*T56

set(handles.trans\_table,'Data',T06); %ileri kinematiginin tabloya aktarılmasi.

function edit\_q2\_Callback(hObject, eventdata, handles)

edit=get(hObject,'string');

set(handles.slider,'value',str2num(edit));

guidata(hObject,handles);

function edit\_q2\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit\_q3\_Callback(hObject, eventdata, handles)

edit=get(hObject,'string');

set(handles.slider,'value',str2num(edit));

guidata(hObject,handles);

function edit\_q3\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit\_q4\_Callback(hObject, eventdata, handles)

edit=get(hObject,'string');

set(handles.slider,'value',str2num(edit));

guidata(hObject,handles);

function edit\_q4\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit\_q5\_Callback(hObject, eventdata, handles)

edit=get(hObject,'string');

set(handles.slider,'value',str2num(edit));

guidata(hObject,handles);

function edit\_q5\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function edit\_q6\_Callback(hObject, eventdata, handles)

edit=get(hObject,'string');

set(handles.slider,'value',str2num(edit));

guidata(hObject,handles);

function edit\_q6\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function slider\_q2\_Callback(hObject, eventdata, handles)

angle=get(hObject,'Value');

set(handles.edit\_q2,'string',num2str(angle));

guidata(hObject,handles);

function slider\_q2\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function slider\_q3\_Callback(hObject, eventdata, handles)

angle=get(hObject,'Value');

set(handles.edit\_q3,'string',num2str(angle));

guidata(hObject,handles);

function slider\_q3\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function slider\_q4\_Callback(hObject, eventdata, handles)

angle=get(hObject,'Value');

set(handles.edit\_q4,'string',num2str(angle));

guidata(hObject,handles);

function slider\_q4\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function slider\_q5\_Callback(hObject, eventdata, handles)

angle=get(hObject,'Value');

set(handles.edit\_q5,'string',num2str(angle));

guidata(hObject,handles);

function slider\_q5\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function slider\_q6\_Callback(hObject, eventdata, handles)

angle=get(hObject,'Value');

set(handles.edit\_q6,'string',num2str(angle));

guidata(hObject,handles);

function slider\_q6\_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor',[.9 .9 .9]);

end

function edit18\_Callback(hObject, eventdata, handles)

function edit18\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function jac\_button\_Callback(hObject, eventdata, handles)

%Sembolik olarak jacobiyen matrisi cikarilmasi

syms q1 q2 q3 q4 q5 q6;

syms L1 L2 L3 L4 L5 L6;

T01 = [cos(q1) -sin(q1) 0 0 ;

sin(q1) cos(q1) 0 0 ;

0 0 1 L1 ;

0 0 0 1 ]

T12 = [cos(q2-(pi/2)) -sin(q2-(pi/2)) 0 L2 ;

0 0 1 0 ;

-sin(q2-(pi/2)) -cos(q2-(pi/2)) 0 0 ;

0 0 0 1 ]

T23 = [cos(q3) -sin(q3) 0 L3 ;

sin(q3) cos(q3) 0 0 ;

0 0 1 0 ;

0 0 0 1 ]

T34 = [cos(q4) -sin(q4) 0 L4 ;

0 0 1 L5 ;

-sin(q4) -cos(q4) 0 0 ;

0 0 0 1 ]

T45 = [cos(q5) -sin(q5) 0 0 ;

0 0 -1 0 ;

sin(q5) cos(q5) 0 0 ;

0 0 0 1 ]

T56 = [cos(q6) -sin(q6) 0 0 ;

0 0 1 L6 ;

-sin(q6) -cos(q6) 0 0 ;

0 0 0 1 ]

T06=T01\*T12\*T23\*T34\*T56

pos=T06(1:3,4)

%%Jacobiyen %%

Vx\_q1=diff(pos(1,1),q1);%Hizlardan olusan jacobiyen matrisi icin pozisyon matrisini degiskenlere gore turevi alinir

Vx\_q2=diff(pos(1,1),q2);

Vx\_q3=diff(pos(1,1),q3);

Vx\_q4=diff(pos(1,1),q4);

Vx\_q5=diff(pos(1,1),q5);

Vx\_q6=diff(pos(1,1),q6);

Vy\_q1=diff(pos(2,1),q1);

Vy\_q2=diff(pos(2,1),q2);

Vy\_q3=diff(pos(2,1),q3);

Vy\_q4=diff(pos(2,1),q4);

Vy\_q5=diff(pos(2,1),q5);

Vy\_q6=diff(pos(2,1),q6);

Vz\_q1=diff(pos(3,1),q1);

Vz\_q2=diff(pos(3,1),q2);

Vz\_q3=diff(pos(3,1),q3);

Vz\_q4=diff(pos(3,1),q4);

Vz\_q5=diff(pos(3,1),q5);

Vz\_q6=diff(pos(3,1),q6);

V\_jac=[Vx\_q1 Vx\_q2 Vx\_q3 Vx\_q4 Vx\_q5 Vx\_q6;

Vy\_q1 Vy\_q2 Vy\_q3 Vy\_q4 Vy\_q5 Vy\_q6;

Vz\_q1 Vz\_q2 Vz\_q3 Vz\_q4 Vz\_q5 Vz\_q6 ]

V\_jacstr=arrayfun(@char,V\_jac,'uniform',0)

i3=[0;0;1];

R01=T01(1:3,1:3);

T02=T01\*T12;

R02=T02(1:3,1:3);

T03=T01\*T12\*T23;

R03=T03(1:3,1:3);

T04=T01\*T12\*T23\*T34;

R04=T04(1:3,1:3);

T05=T01\*T12\*T23\*T34\*T45;

R05=T05(1:3,1:3);

R06=T06(1:3,1:3);

Jw\_q1=R01\*i3 %acisal hizlardan olusan jacobiyen matrisini bulunmasi

Jw\_q2=R02\*i3;

Jw\_q3=R03\*i3;

Jw\_q4=R04\*i3;

Jw\_q5=R05\*i3;

Jw\_q6=R06\*i3;

W\_jac=[Jw\_q1 Jw\_q2 Jw\_q3 Jw\_q4 Jw\_q5 Jw\_q6];

W\_jacstr=arrayfun(@char,V\_jac,'uniform',0);

matris\_jac=[V\_jacstr;W\_jacstr];

set(handles.table\_jac,'Data',matris\_jac) %sembolik olarak jacobiyen matrisinin tabloya aktarilmasi

function symbol\_jac\_Callback(hObject, eventdata, handles)

L1=203; L2=75; L3=365; L4=90; L5=405; L6=80;

syms Q1 Q2 Q3 Q4 Q5 Q6;

T01 = [cos(Q1) -sin(Q1) 0 0 ;

sin(Q1) cos(Q1) 0 0 ;

0 0 1 L1 ;

0 0 0 1 ]

T12 = [cos(Q2-(pi/2)) -sin(Q2-(pi/2)) 0 L2 ;

0 0 1 0 ;

-sin(Q2-(pi/2)) -cos(Q2-(pi/2)) 0 0 ;

0 0 0 1 ]

T23 = [cos(Q3) -sin(Q3) 0 L3 ;

sin(Q3) cos(Q3) 0 0 ;

0 0 1 0 ;

0 0 0 1 ]

T34 = [cos(Q4) -sin(Q4) 0 L4 ;

0 0 1 L5 ;

-sin(Q4) -cos(Q4) 0 0 ;

0 0 0 1 ]

T45 = [cos(Q5) -sin(Q5) 0 0 ;

0 0 -1 0 ;

sin(Q5) cos(Q5) 0 0 ;

0 0 0 1 ]

T56 = [cos(Q6) -sin(Q6) 0 0 ;

0 0 1 L6 ;

-sin(Q6) -cos(Q6) 0 0 ;

0 0 0 1 ]

T06=T01\*T12\*T23\*T34\*T56

pos=T06(1:3,4)

%%Jacobiyen %%

Vx\_q1=diff(pos(1,1),Q1);

Vx\_q2=diff(pos(1,1),Q2);

Vx\_q3=diff(pos(1,1),Q3);

Vx\_q4=diff(pos(1,1),Q4);

Vx\_q5=diff(pos(1,1),Q5);

Vx\_q6=diff(pos(1,1),Q6);

Vy\_q1=diff(pos(2,1),Q1);

Vy\_q2=diff(pos(2,1),Q2);

Vy\_q3=diff(pos(2,1),Q3);

Vy\_q4=diff(pos(2,1),Q4);

Vy\_q5=diff(pos(2,1),Q5);

Vy\_q6=diff(pos(2,1),Q6);

Vz\_q1=diff(pos(3,1),Q1);

Vz\_q2=diff(pos(3,1),Q2);

Vz\_q3=diff(pos(3,1),Q3);

Vz\_q4=diff(pos(3,1),Q4);

Vz\_q5=diff(pos(3,1),Q5);

Vz\_q6=diff(pos(3,1),Q6);

V\_jac=[Vx\_q1 Vx\_q2 Vx\_q3 Vx\_q4 Vx\_q5 Vx\_q6;

Vy\_q1 Vy\_q2 Vy\_q3 Vy\_q4 Vy\_q5 Vy\_q6;

Vz\_q1 Vz\_q2 Vz\_q3 Vz\_q4 Vz\_q5 Vz\_q6 ]

i3=[0;0;1];

R01=T01(1:3,1:3);

T02=T01\*T12;

R02=T02(1:3,1:3);

T03=T01\*T12\*T23;

R03=T03(1:3,1:3);

T04=T01\*T12\*T23\*T34;

R04=T04(1:3,1:3);

T05=T01\*T12\*T23\*T34\*T45;

R05=T05(1:3,1:3);

R06=T06(1:3,1:3);

Jw\_q1=R01\*i3

Jw\_q2=R02\*i3;

Jw\_q3=R03\*i3;

Jw\_q4=R04\*i3;

Jw\_q5=R05\*i3;

Jw\_q6=R06\*i3;

W\_jac=[Jw\_q1 Jw\_q2 Jw\_q3 Jw\_q4 Jw\_q5 Jw\_q6];

matris\_jac=[V\_jac;W\_jac]

q1= str2num(get(handles.edit\_q1,'String'));

q2= str2num(get(handles.edit\_q2,'String'));

q3= str2num(get(handles.edit\_q3,'String'));

q4= str2num(get(handles.edit\_q4,'String'));

q5= str2num(get(handles.edit\_q5,'String'));

q6= str2num(get(handles.edit\_q6,'String'));

Q\_matrix=[Q1 Q2 Q3 Q4 Q5 Q6 ];

q\_matrix=[q1 q2 q3 q4 q5 q6];

ret\_jac=subs(matris\_jac,Q\_matrix,q\_matrix); %Q degiskenini edit textlerden aldigimiz q degiskenine atanmasi

jac\_num=double(ret\_jac);

set(handles.symbol\_table,'Data',jac\_num);%matrisin tabloya aktarimi

Kaynakça

1. Z. Bingül, S. Küçük, “Robot Tekniği I”, Birsen Yayınevi, pp. 104-200, 2005.
2. <http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly93d3cud2lraXBlZGlhLm9yZw>[1],[2]
3. [https://www.kuka.com/[3](https://www.kuka.com/%5b3)]
4. Mathworks