RBE 3005
วิศวกรรมหุ่นยนต์ (Robotics Engineering)

สาขาวิศวกรรมหุ่นยนต์

คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม

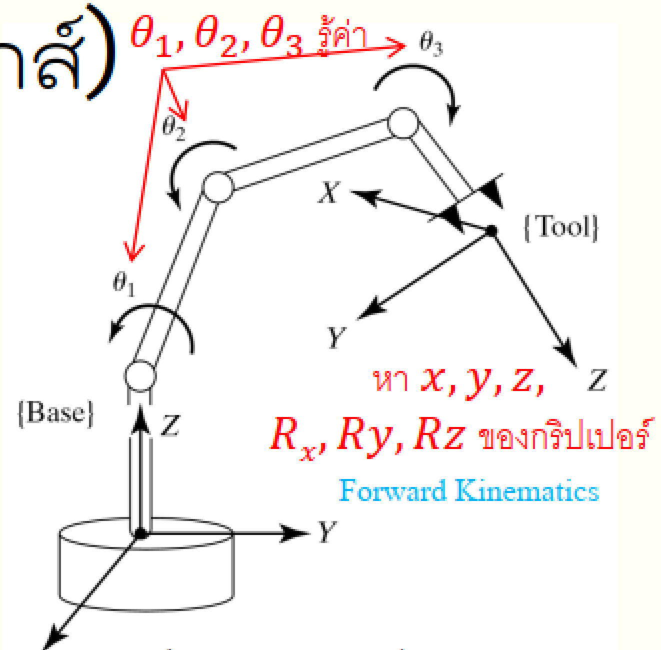
มหาวิทยาลัยราชภัฏสวนสุนันทา

บทที่ 3 จลนศาสตร์ข้างหน้าและผกผัน
Forward Kinematics and Inverse Kinematics

- สายโซ่จลนศาสตร์ (kinematic chain)
- จลนศาสตร์ข้างหน้าโดยวิธีการของเดนาวิท-ฮาร์เทนบอร์ก (Denavit-Hartenberg)

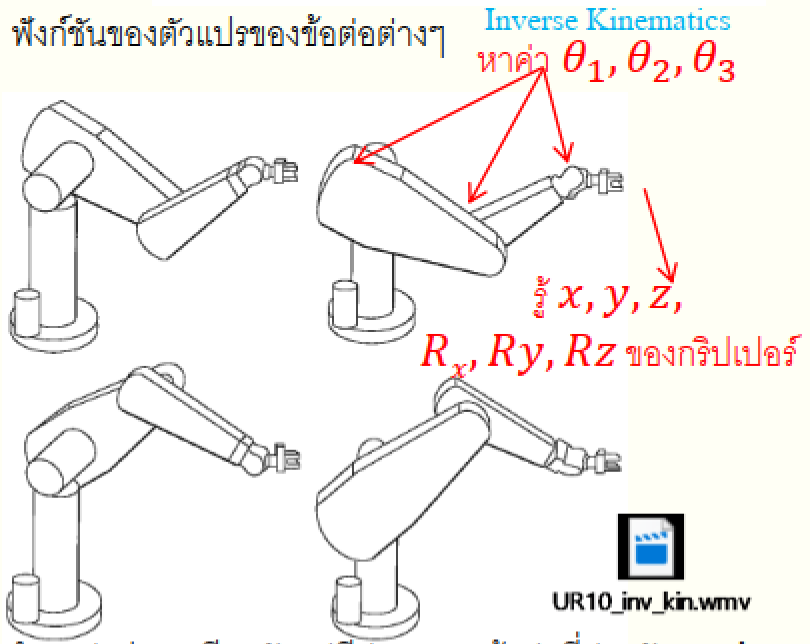
Kinematics (คิเนเมติกส์)

- **Forward Kinematics** : การคำนวณหาตำแหน่ง (position: x,y,z) และทิศทาง (orientation: R_x, R_y, R_z) หรือเรียกสั้นๆ ว่า Pose ของกริปเปอร์หรือทูล (tool) ของหุ่นยนต์ที่สัมพัทธ์กับเบสเตชั่น (Base) โดยทราบค่ามุมของแกนต่างๆ (Axes) ของหุ่นยนต์ (joint angles : $\theta_1, \theta_2, \theta_3$)



รูปที่ 1 ทูลเฟรมสัมพัทธ์กับเบสเฟรมโดยเป็นฟังก์ชันของตัวแปรของข้อต่อต่างๆ

- **Inverse Kinematics** : การคำนวณหาค่ามุมของแกนต่างๆ ของหุ่นยนต์ เมื่อทราบตำแหน่ง และทิศทางของกริปเปอร์หรือทูลของหุ่นยนต์ที่สัมพัทธ์กับเบสเตชั่น



รูปที่ 2 ตำแหน่งปลายเดียวกันแต่มีค่ามุมของข้อต่อที่ต่างกันของ Inv.

Link Description

- Links : a set of bodies connected by joints.
- Joint : forming a connection between two links
- Lower pair : being used to describe the connection between a pair of bodies when the relative motion is characterized by two surfaces sliding over one another.

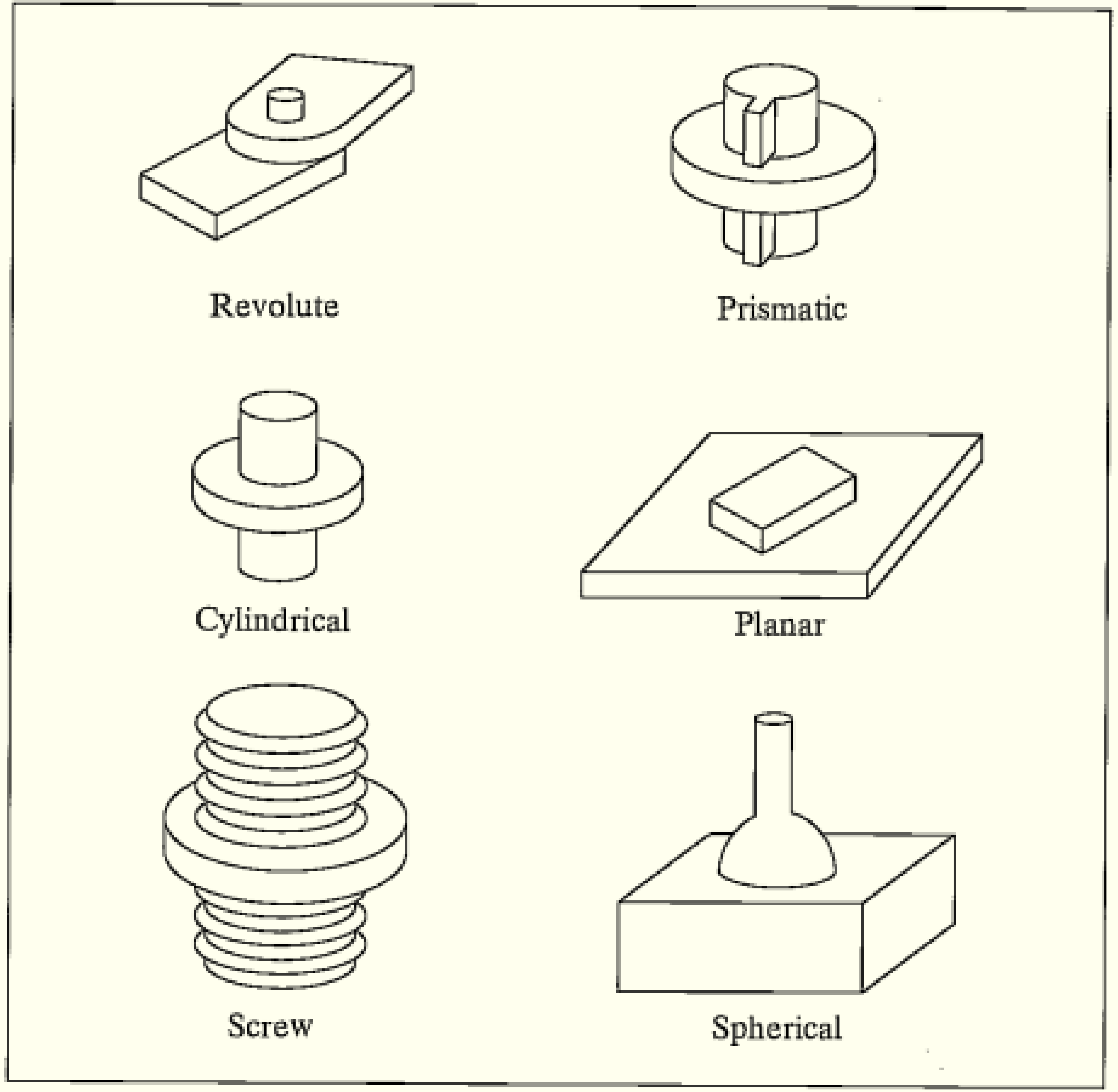


FIGURE 3.1: The six possible lower-pair joints.

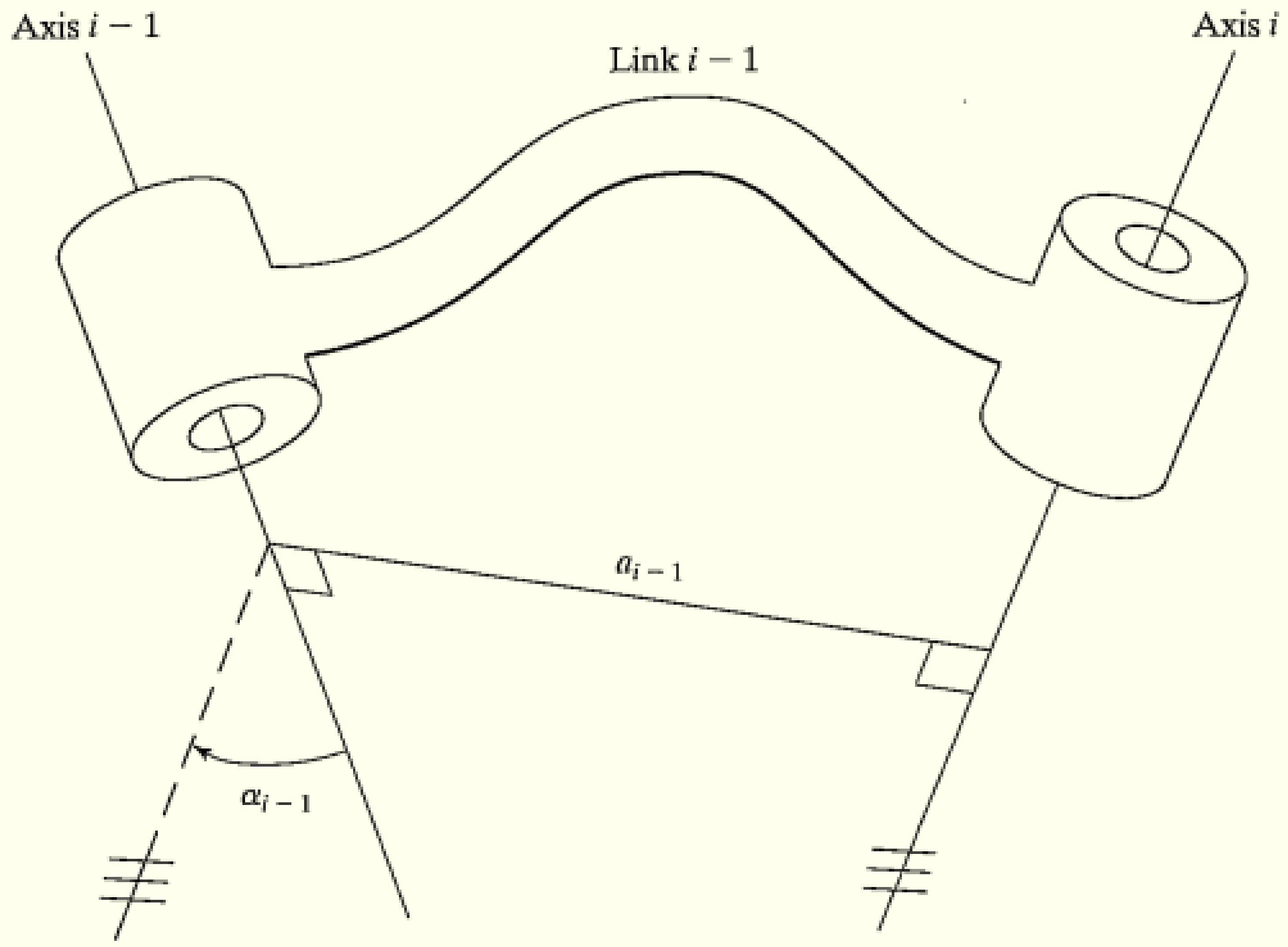


FIGURE 3.2: The kinematic function of a link is to maintain a fixed relationship between the two joint axes it supports. This relationship can be described with two parameters: the link length, a, and the link twist, α.

Two Parameters define the relationship between two lines

- **Link length** = a_{i-1} measured between two joint axes. The line is perpendicular to both axes.
- **Link twist** = α_{i-1} angle between axis i-1 and axis I when both axes lay down on a plan whose normal is the perpendicular line.

EXAMPLE 3.1

Figure 3.3 shows the mechanical drawings of a robot link. If this link is used in a robot, with bearing "A" used for the lower-numbered joint, give the length and twist of this link. Assume that holes are centered in each bearing.

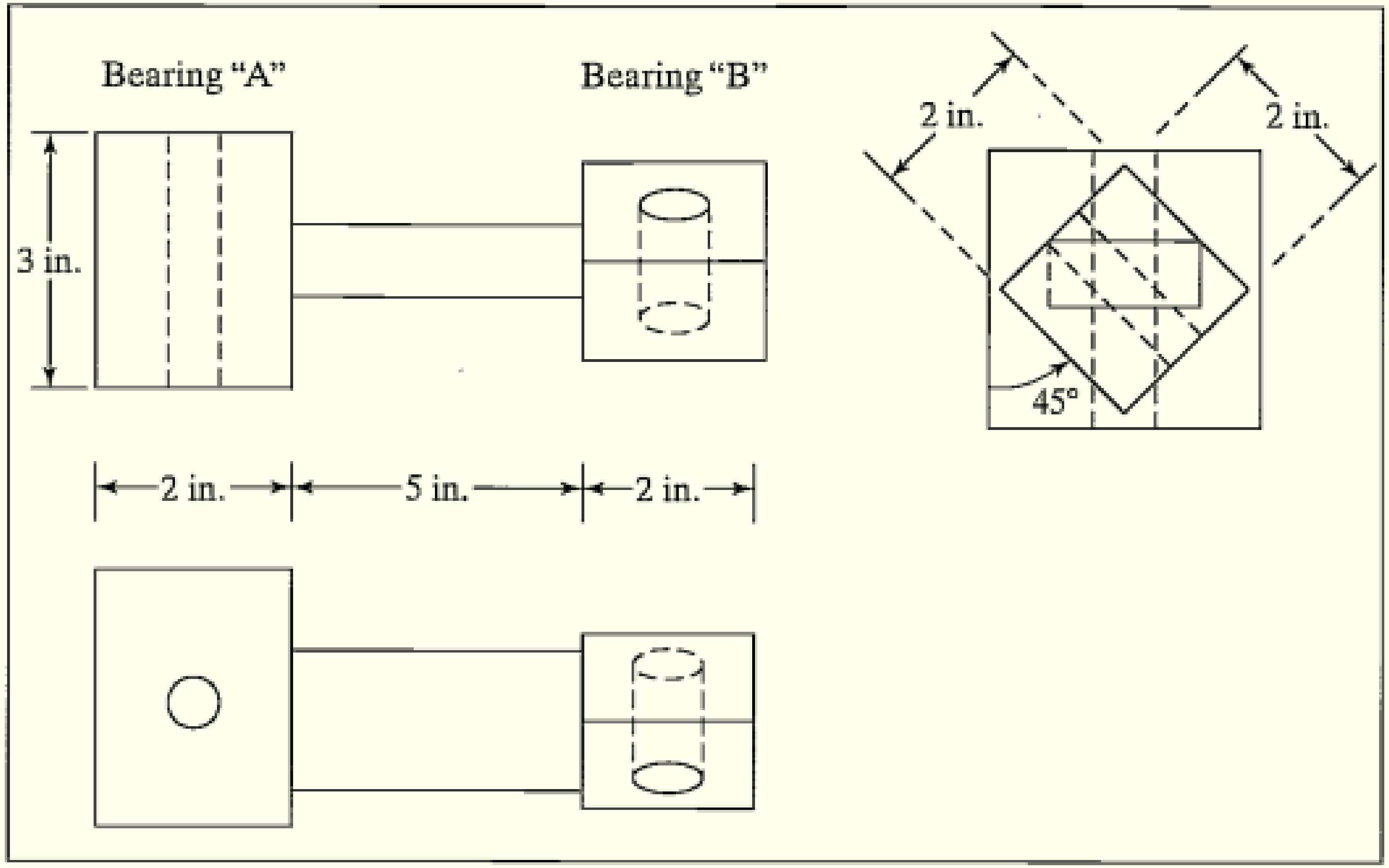


FIGURE 3.3: A simple link that supports two revolute axes.

EXAMPLE 3.1

Figure 3.3 shows the mechanical drawings of a robot link. If this link is used in a robot, with bearing "A" used for the lower-numbered joint, give the length and twist of this link. Assume that holes are centered in each bearing.

By inspection, the common perpendicular lies right down the middle of the metal bar connecting the bearings, so the link length is 7 inches. The end view actually shows a projection of the bearings onto the plane whose normal is the mutual perpendicular. Link twist is measured in the right-hand sense about the common perpendicular from axis $i-1$ to axis i, so, in this example, it is clearly +45 degrees.

Link-Connection Description

The problem of connecting the links of a robot together is again one filled with many questions for the mechanical designer to resolve. These include the strength of the joint, its lubrication, and the bearing and gearing mounting. However, for the investigation of kinematics, we need only worry about two quantities, which will completely specify the way in which links are connected together.

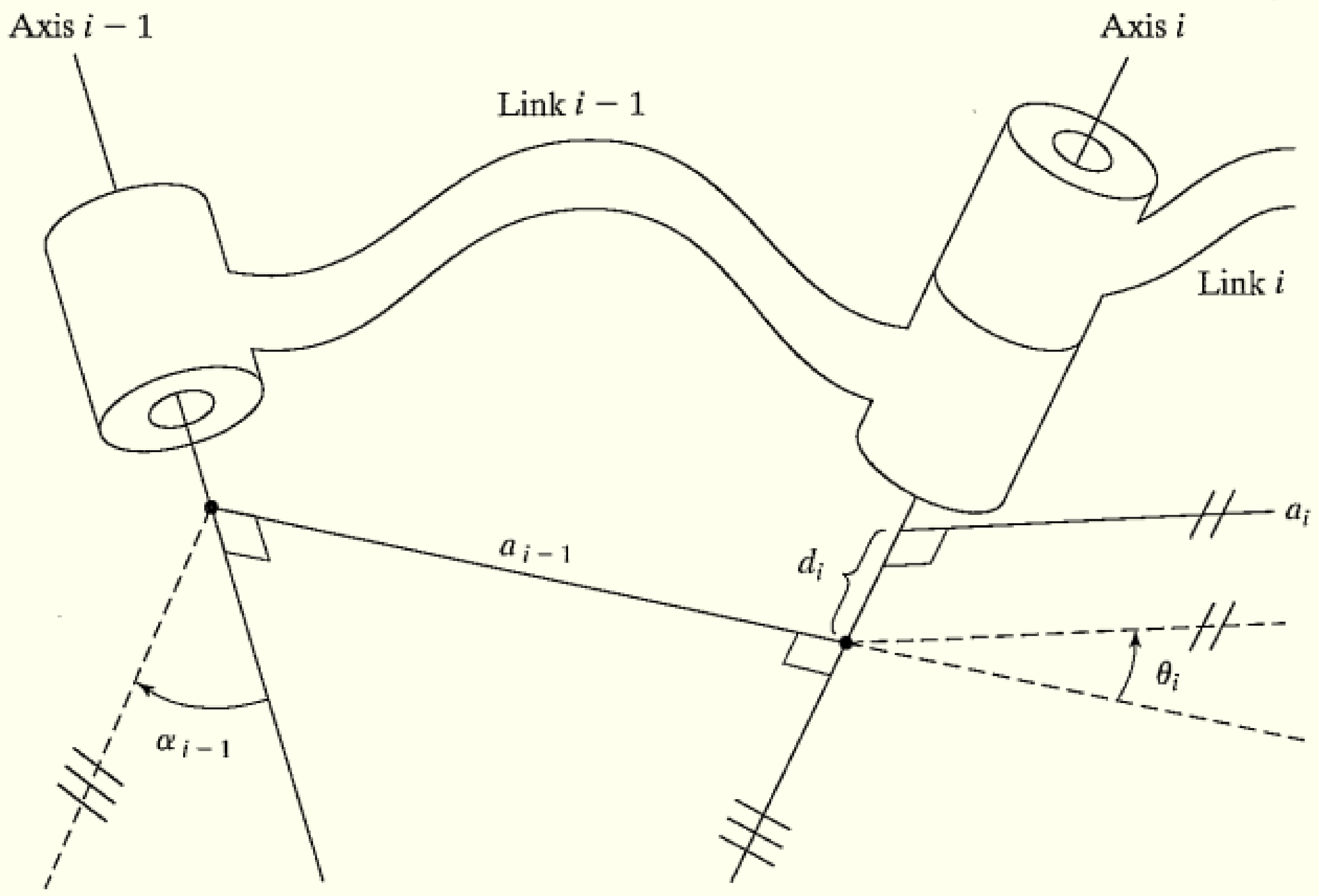


FIGURE 3.4: The link offset, d, and the joint angle, θ, are two parameters that may be used to describe the nature of the connection between neighboring links.

Intermediate links in the chain

Neighboring links have a common joint axis between them. One parameter of interconnection has to do with the distance along this common axis from one link to the next. This parameter is called the **link offset**. The offset at joint axis i is called d_i. The second parameter describes the amount of rotation about this common axis between one link and its neighbor. This is called the **joint angle**, θ_i.

Figure 3.4 shows the interconnection of link $i-1$ and link i. Recall that a_{i-1} is the mutual perpendicular between the two axes of link $i-1$. Likewise, a_i is the mutual perpendicular defined for link i. The first parameter of interconnection is the link offset, d_i, which is the signed distance measured along the axis of joint i from the point where a_{i-1} intersects the axis to the point where a_i intersects the axis. The offset d_i is indicated in Fig. 3.4. The link offset d_i is variable if joint i is prismatic. The second parameter of interconnection is the angle made between an extension of a_{i-1} and a_i measured about the axis of joint i. This is indicated in Fig. 3.4, where the lines with the double hash marks are parallel. This parameter is named θ_i and is variable for a revolute joint.

First and last links in the chain

Link length, a_i, and link twist, α_i, depend on joint axes i and $i+1$. Hence, a_1 through a_{n-1} and α_1 through α_{n-1} are defined as was discussed in this section. At the ends of the chain, it will be our convention to assign zero to these quantities. That is, $a_0 = a_n = 0.0$ and $\alpha_0 = \alpha_n = 0.0$.[3] Link offset, d_i, and joint angle, θ_i, are well defined for joints 2 through $n-1$ according to the conventions discussed in this section. If joint 1 is revolute, the zero position for θ_1 may be chosen arbitrarily; $d_1 = 0.0$ will be our convention. Similarly, if joint 1 is prismatic, the zero position of d_1 may be chosen arbitrarily; $\theta_1 = 0.0$ will be our convention. Exactly the same statements apply to joint n.

These conventions have been chosen so that, in a case where a quantity could be assigned arbitrarily, a zero value is assigned so that later calculations will be as simple as possible.

Link parameters

Hence, any robot can be described kinematically by giving the values of four quantities for each link. Two describe the link itself, and two describe the link's connection to a neighboring link. In the usual case of a revolute joint, θ_i is called the **joint variable**, and the other three quantities would be fixed **link parameters**. For prismatic joints, d_i is the joint variable, and the other three quantities are fixed link parameters. The definition of mechanisms by means of these quantities is a convention usually called the **Denavit–Hartenberg notation** [1].[4] Other methods of describing mechanisms are available, but are not presented here.

At this point, we could inspect any mechanism and determine the Denavit–Hartenberg parameters that describe it. For a six-jointed robot, 18 numbers would be required to describe the fixed portion of its kinematics completely. In the case of a six-jointed robot with all revolute joints, the 18 numbers are in the form of six sets of $(\alpha_i, \alpha_i, d_i)$.

EXAMPLE 3.2

Two links, as described in Fig. 3.3, are connected as links 1 and 2 of a robot. Joint 2 is composed of a "B" bearing of link 1 and an "A" bearing of link 2, arranged so that the flat surfaces of the "A" and "B" bearings lie flush against each other. What is d_2?

The link offset d_2 is the offset at joint 2, which is the distance, measured along the joint 2 axis, between the mutual perpendicular of link 1 and that of link 2. From the drawings in Fig. 3.3, this is 2.5 inches.

CONVENTION FOR AFFIXING FRAMES TO LINKS

In order to describe the location of each link relative to its neighbors, we define a frame attached to each link. The link frames are named by number according to the link to which they are attached. That is, frame $\{i\}$ is attached rigidly to link i.

Intermediate links in the chain

The convention we will use to locate frames on the links is as follows: The $\hat{Z}$-axis of frame $\{i\}$, called $\hat{Z}_i$, is coincident with the joint axis i. The origin of frame $\{i\}$ is located where the a_i perpendicular intersects the joint i axis. $\hat{X}_i$ points along a_i in the direction from joint i to joint $i+1$.

In the case of $a_i = 0$, $\hat{X}_i$ is normal to the plane of $\hat{Z}_i$ and $\hat{Z}_{i+1}$. We define α_i as being measured in the right-hand sense about $\hat{X}_i$, and so we see that the freedom of choosing the sign of α_i in this case corresponds to two choices for the direction of $\hat{X}_i$. $\hat{Y}_i$ is formed by the right-hand rule to complete the ith frame. Figure 3.5 shows the location of frames $\{i-1\}$ and $\{i\}$ for a general manipulator.

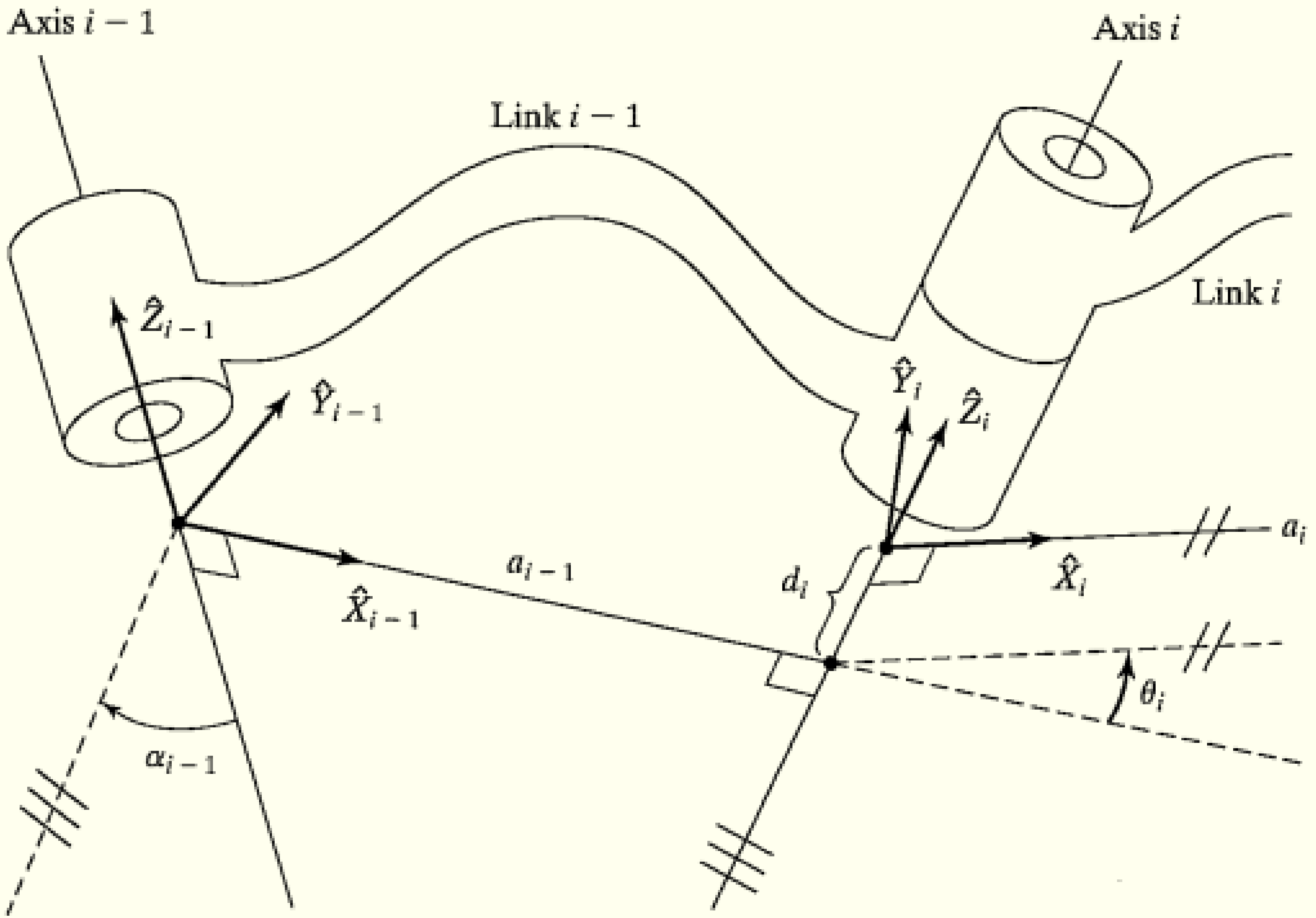


FIGURE 3.5: Link frames are attached so that frame $\{i\}$ is attached rigidly to link i.

First and last links in the chain

We attach a frame to the base of the robot, or link 0, called frame {0}. This frame does not move; for the problem of arm kinematics, it can be considered the reference frame. We may describe the position of all other link frames in terms of this frame.

Frame {0} is arbitrary, so it always simplifies matters to choose $\hat{Z}_0$ along axis 1 and to locate frame {0} so that it coincides with frame {1} when joint variable 1 is zero. Using this convention, we will always have $a_0 = 0.0$, $\alpha_0 = 0.0$. Additionally, this ensures that $d_1 = 0.0$ if joint 1 is revolute, or $\theta_1 = 0.0$ if joint 1 is prismatic.

For joint n revolute, the direction of $\hat{X}_N$ is chosen so that it aligns with $\hat{X}_{N-1}$ when $\theta_n = 0.0$, and the origin of frame $\{N\}$ is chosen so that $d_n = 0.0$. For joint n prismatic, the direction of $\hat{X}_N$ is chosen so that $\theta_n = 0.0$, and the origin of frame $\{N\}$ is chosen at the intersection of $\hat{X}_{N-1}$ and joint axis n when $d_n = 0.0$.

Summary of the link parameters in terms of the link frames

If the link frames have been attached to the links according to our convention, the following definitions of the link parameters are valid:

$$a_i = \textit{the distance from } \hat{Z}_i \textit{ to } \hat{Z}_{i+1} \textit{ measured along } \hat{X}_i;$$

$$\alpha_i = \textit{the angle from } \hat{Z}_i \textit{ to } \hat{Z}_{i+1} \textit{ measured about } \hat{X}_i;$$

$$d_i = \textit{the distance from } \hat{X}_{i-1} \textit{ to } \hat{X}_i \textit{ measured along } \hat{Z}_i; \text{ and}$$

$$\theta_i = \textit{the angle from } \hat{X}_{i-1} \textit{ to } \hat{X}_i \textit{ measured about } \hat{Z}_i.$$

We usually choose $a_i > 0$, because it corresponds to a distance; however, α_i, d_i, and θ_i are signed quantities.

Summary of link-frame attachment procedure

The following is a summary of the procedure to follow when faced with a new mechanism, in order to properly attach the link frames:

1. Identify the joint axes and imagine (or draw) infinite lines along them. For steps 2 through 5 below, consider two of these neighboring lines (at axes i and $i+1$).
2. Identify the common perpendicular between them, or point of intersection. At the point of intersection, or at the point where the common perpendicular meets the ith axis, assign the link-frame origin.
3. Assign the $\hat{Z}_i$ axis pointing along the ith joint axis.
4. Assign the $\hat{X}_i$ axis pointing along the common perpendicular, or, if the axes intersect, assign $\hat{X}_i$ to be normal to the plane containing the two axes.
5. Assign the $\hat{Y}_i$ axis to complete a right-hand coordinate system.
6. Assign $\{0\}$ to match $\{1\}$ when the first joint variable is zero. For $\{N\}$, choose an origin location and $\hat{X}_N$ direction freely, but generally so as to cause as many linkage parameters as possible to become zero.

EXAMPLE 3.3

Figure 3.6(a) shows a three-link planar arm. Because all three joints are revolute, this manipulator is sometimes called an ***RRR*** (or **3*R***) **mechanism**. Fig. 3.6(b) is a schematic representation of the same manipulator. Note the double hash marks indicated on each of the three axes, which indicate that these axes are parallel. Assign link frames to the mechanism and give the Denavit–Hartenberg parameters.

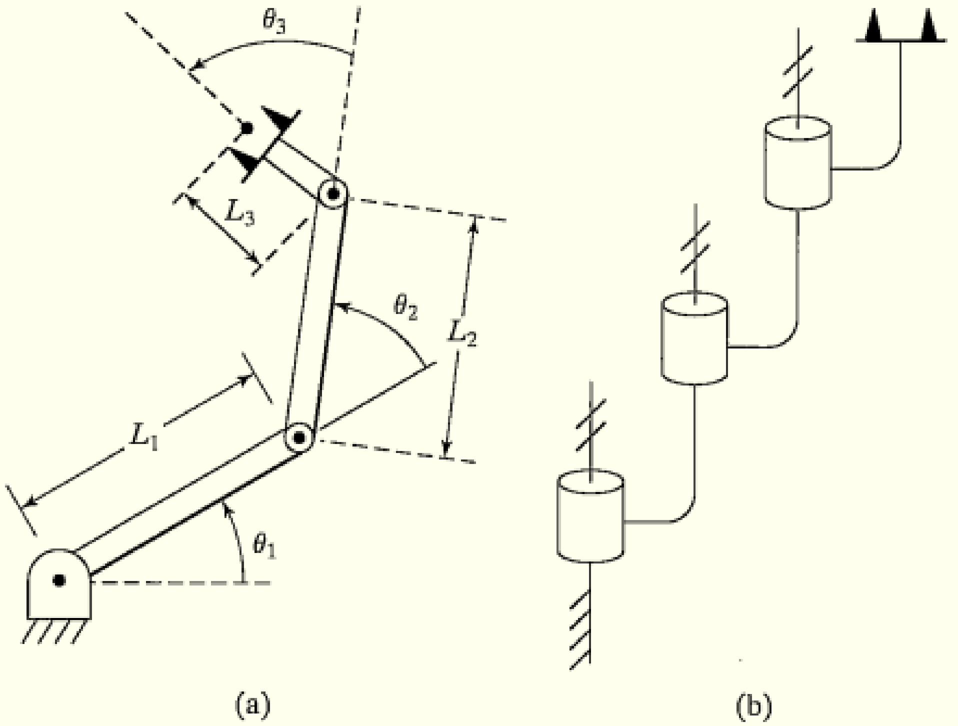


FIGURE 3.6: A three-link planar arm. On the right, we show the same manipulator by means of a simple schematic notation. Hash marks on the axes indicate that they are mutually parallel.

We start by defining the reference frame, frame {0}. It is fixed to the base and aligns with frame {1} when the first joint variable (θ_1) is zero. Therefore, we position frame {0} as shown in Fig. 3.7 with $\hat{Z}_0$ aligned with the joint-1 axis. For this arm, all joint axes are oriented perpendicular to the plane of the arm. Because the arm lies in a plane with all $\hat{Z}$ axes parallel, there are no link offsets—all d_i are zero. All joints are rotational, so when they are at zero degrees, all $\hat{X}$ axes must align.

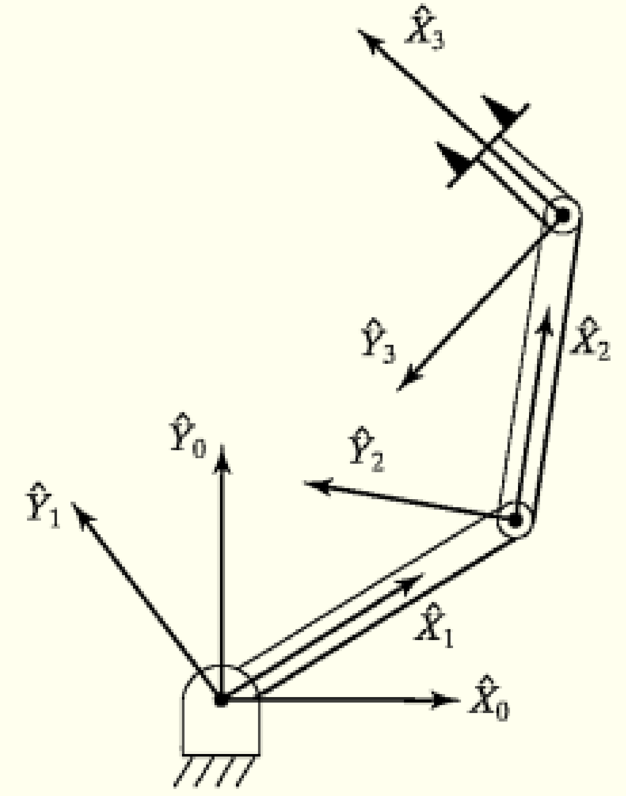


FIGURE 3.7: Link-frame assignments.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	0	L_1	0	θ_2
3	0	L_2	0	θ_3

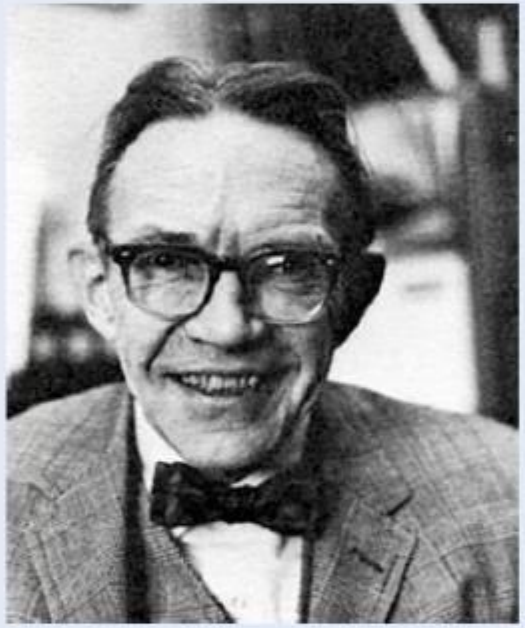
FIGURE 3.8: Link parameters of the three-link planar manipulator.

Excursse 7.4: Denavit and Hartenberg

Jacques Denavit and Richard Hartenberg introduced many of the key concepts of kinematics for serial-link manipulators in a 1955 paper (Denavit and Hartenberg 1955) and their later classic text *Kinematic Synthesis of Linkages* (Hartenberg and Denavit 1964).

Jacques Denavit (1930–2012) was born in Paris where he studied for his Bachelor degree before pursuing his masters and doctoral degrees in mechanical engineering at Northwestern University, Illinois. In 1958 he joined the Department of Mechanical Engineering and Astronautical Science at Northwestern where the collaboration with Hartenberg was formed. In addition to his interest in dynamics and kinematics Denavit was also interested in plasma physics and kinetics. After the publication of the book he moved to Lawrence Livermore National Lab, Livermore, California, where he undertook research on computer analysis of plasma physics problems.

Richard Hartenberg (1907–1997) was born in Chicago and studied for his degrees at the University of Wisconsin, Madison. He served in the merchant marine and studied aeronautics for two years at the University of Göttingen with space-flight pioneer Theodore von Kármán. He was Professor of mechanical engineering at Northwestern University where he taught for 56 years. His research in kinematics led to a revival of interest in this field in the 1960s, and his efforts helped put kinematics on a scientific basis for use in computer applications in the analysis and design of complex mechanisms. He also wrote extensively on the history of mechanical engineering.



Modified Denavit Hartenberg Notation

หนังสือบางเล่ม เช่น Introduction to Robotics: Mechanics and Control (ฉบับที่ 3) ใช้พารามิเตอร์ DH ที่ดัดแปลง (ใกล้เคียง) ความแตกต่างระหว่างพารามิเตอร์ DH แบบคลาสสิก (ไกล) และพารามิเตอร์ DH ที่ดัดแปลงคือตำแหน่งของการยึดระบบพิกัดกับข้อต่อและลำดับของการแปลงที่ดำเนินการ

$$^{i-1}T_i = R_X(\alpha_{i-1})D_X(a_{i-1})R_Z(\theta_i)D_Z(d_i)$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_{i-1} & -\sin\alpha_{i-1} & 0 \\ 0 & \sin\alpha_{i-1} & \cos\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i\cos\alpha_{i-1} & \cos\theta_i\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1}d_i \\ \sin\theta_i\sin\alpha_{i-1} & \cos\theta_i\sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{eq.(3.2)}$$

EXAMPLE 3.3

Figure 3.6(a) shows a three-link planar arm. Because all three joints are revolute, this manipulator is sometimes called an ***RRR*** (or **3*R***) **mechanism**. Fig. 3.6(b) is a schematic representation of the same manipulator. Note the double hash marks

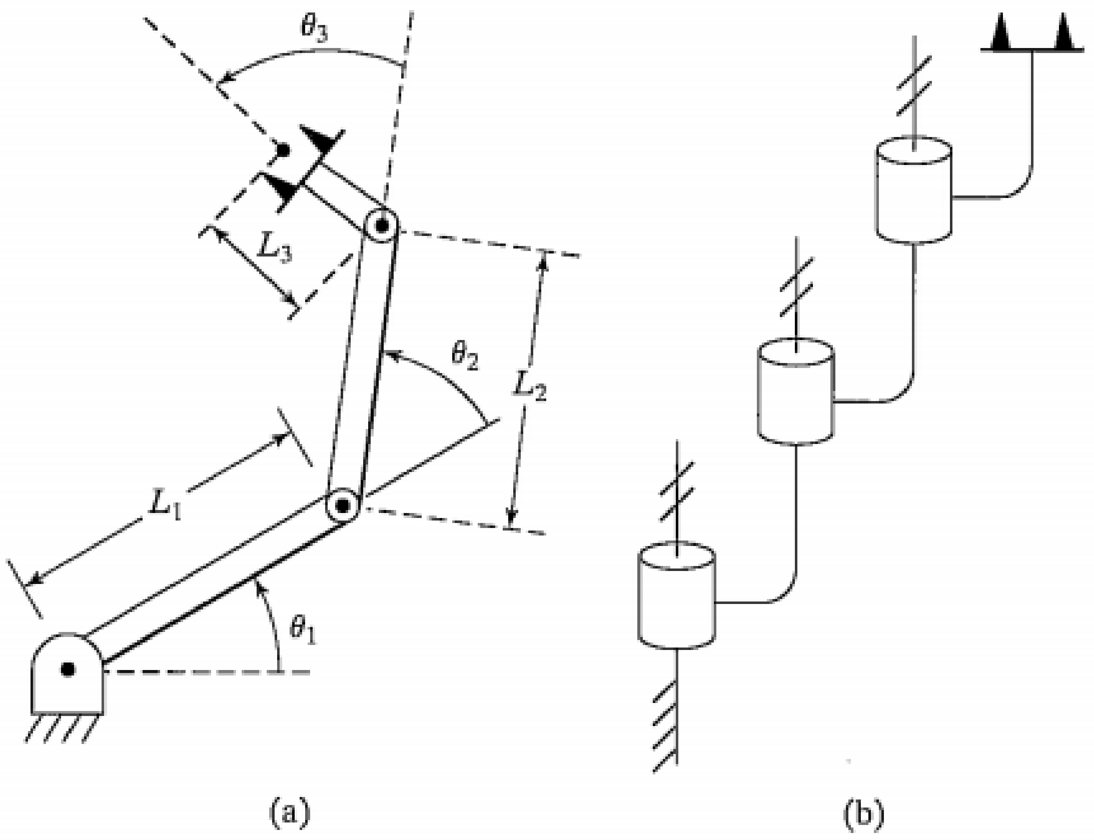


FIGURE 3.6: A three-link planar arm. On the right, we show the same manipulator by means of a simple schematic notation. Hash marks on the axes indicate that they are mutually parallel.

Denavit-Hartenberg Parameters

Denavit-Hartenberg notation.mp4

```
mdl_puma560
p560 ; p560.plot(qz) ;  p560.plot(qr) ; p560.teach
p560.fkine([pi/6 pi/6 pi/6 0 0 0])
```

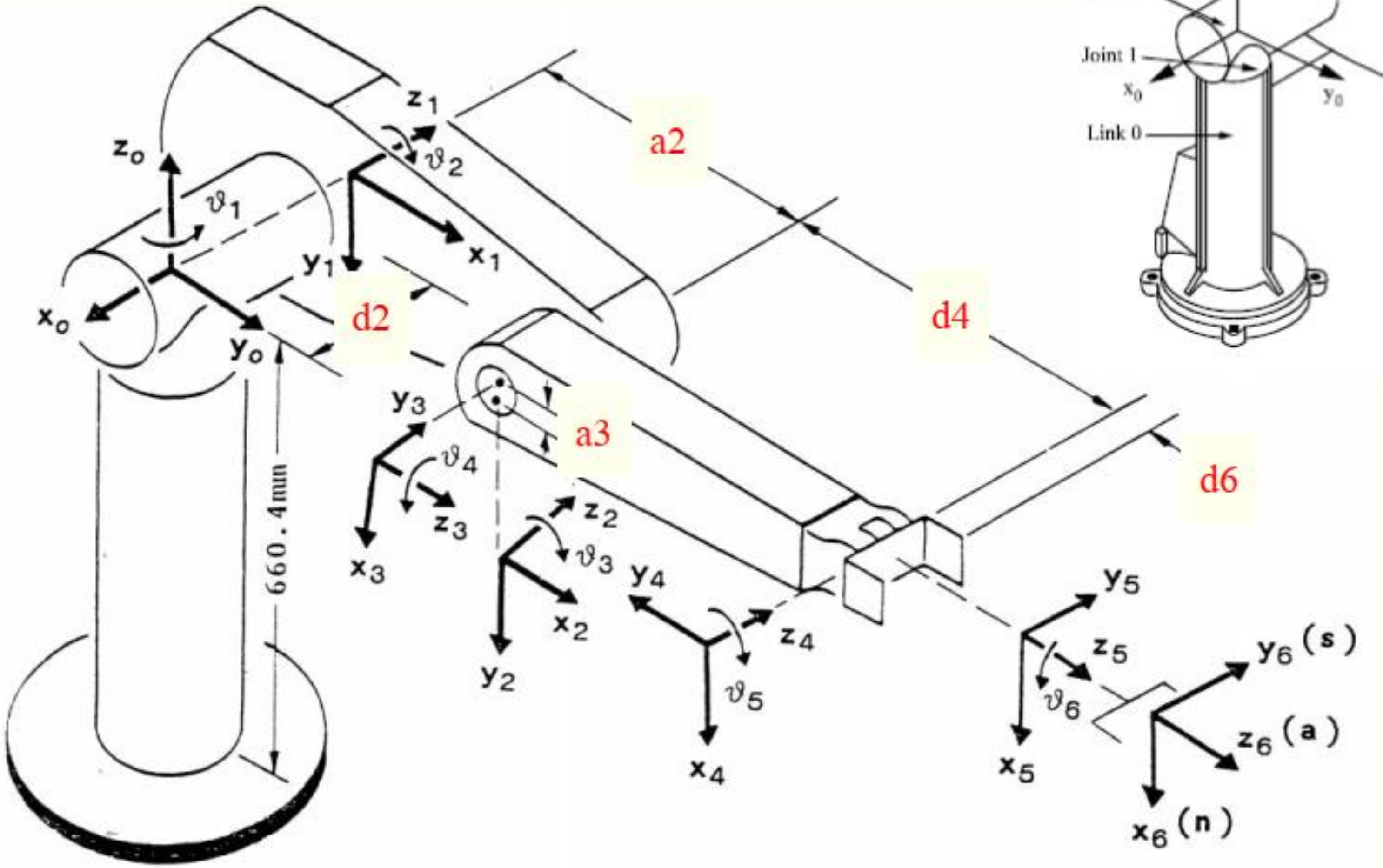
Denavit-Hartenberg parameters for **Puma**

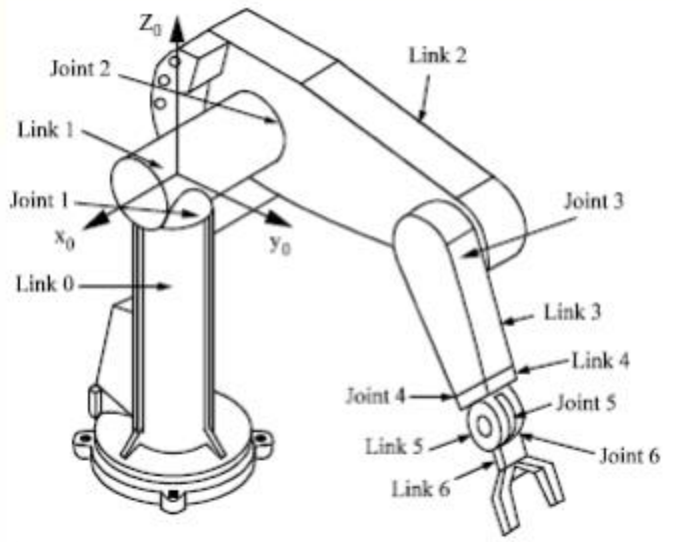
Unimate Puma 500
Oussama Khatib | Used with permission

θ_j	d_j	a_j	α_j
q1	0.0000	0.0000	$\pi/2$
q2	0.0000	0.4318	0
q3	0.1500	0.0203	$-\pi/2$
q4	0.4318	0.0000	$\pi/2$
q5	0.0000	0.0000	$-\pi/2$
q6	0.0000	0.0000	0

https://robotacademy.net.au/

Denavit-Hartenberg Parameters

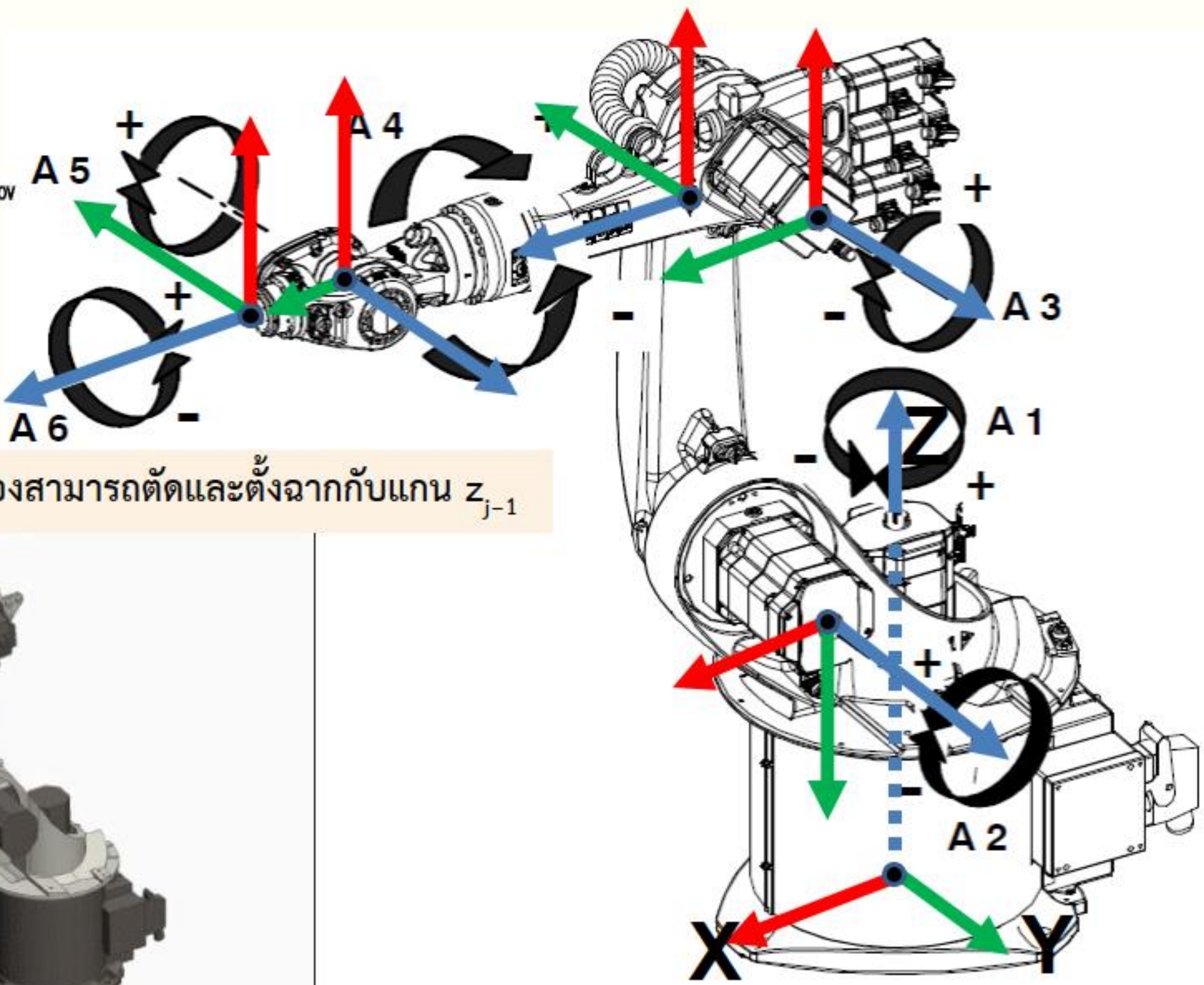




Denavit-Hartenberg Parameters

DH.mp4

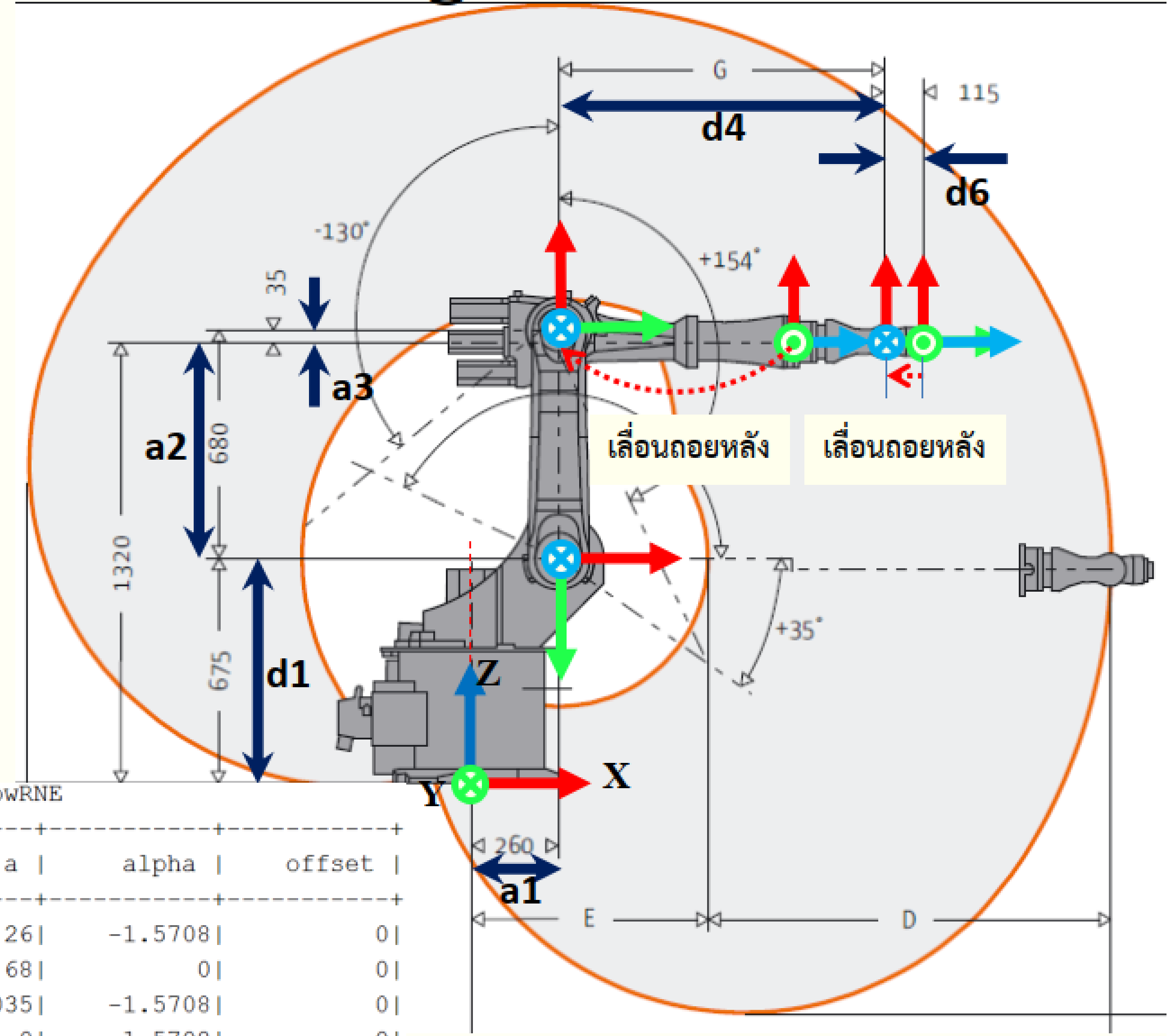
Inside Axis 4,5&6 of KUKA KR5 Robot_trim0.mov



<u>ข้อบังคับ</u> : แนวแกน x_j ต้องสามารถตัดและตั้งฉากกับแกน z_{j-1}

Denavit-Hartenberg Parameters

```
mdl_KR16_L6

KR16 ; KR16.plot(qz);
KR16.teach;

KR16.fkine([0 0 -pi/2 0 0 0])
```



Kuka KR16:: 6 axis, RRRRRR, stdDH, slowRNE

j	theta	d	a	alpha	offset
1	q1	0.675	0.26	-1.5708	0
2	q2	0	0.68	0	0
3	q3	0	-0.035	-1.5708	0
4	q4	0.97	0	1.5708	0
5	q5	0	0	-1.5708	0
6	q6	0.115	0	0	0

Pose in 3D:Homogeneous Transformation Matrix of KUKA KR16

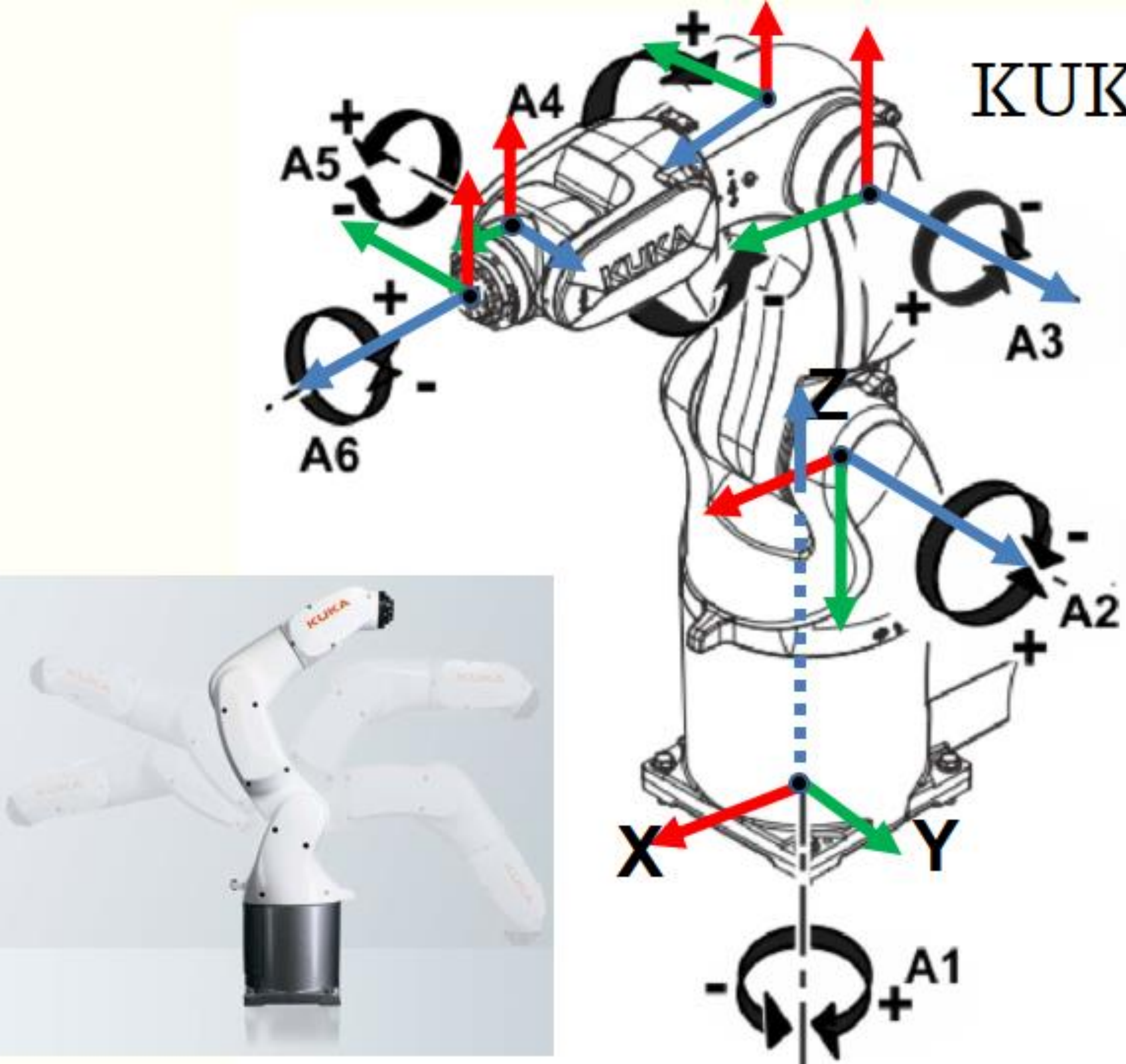
```
clear
A1=0; A2=0; A3=0; A4=0; A5=0; A6=0;
T1 = rotz(-A1,'deg’);  T1(4,4)=1;
T2 = rotx(-90,'deg')*rotz(A2+90,'deg'); T2(4,4)=1; T2 = transl(0.26,0,0.675)*T2;
T3 = rotz(-90,'deg')*rotz(A3-90,'deg'); T3(4,4)=1;  T3 = transl(0,-0.68,0)*T3;
T4 = rotx(-90,'deg')*rotz(-A4,'deg'); T4(4,4)=1;     T4 = transl(-0.035,0,0)*T4;
T5 = rotx(90,'deg')*rotz(A5,'deg'); T5(4,4)=1;       T5 = transl(0,0,0.97)*T5;
T6 = rotx(-90,'deg')*rotz(-(180+A6),'deg');T6(4,4)=1; T6 = transl(0,0.115,0)*T6;
T = T1*T2*T3*T4*T5*T6
```

DH Parameters for KUKA KR16_L6

```
>> KR16
KR16 =
Kuka KR16:: 6 axis, RRRRRR, stdDH, slowRNE
```

j	theta	d	a	alpha	offset
1	q1	0.675	0.26	-1.5708	0
2	q2	0	0.68	0	0
3	q3	0	-0.035	-1.5708	0
4	q4	0.97	0	1.5708	0
5	q5	0	0	-1.5708	0
6	q6	0.115	0	0	0

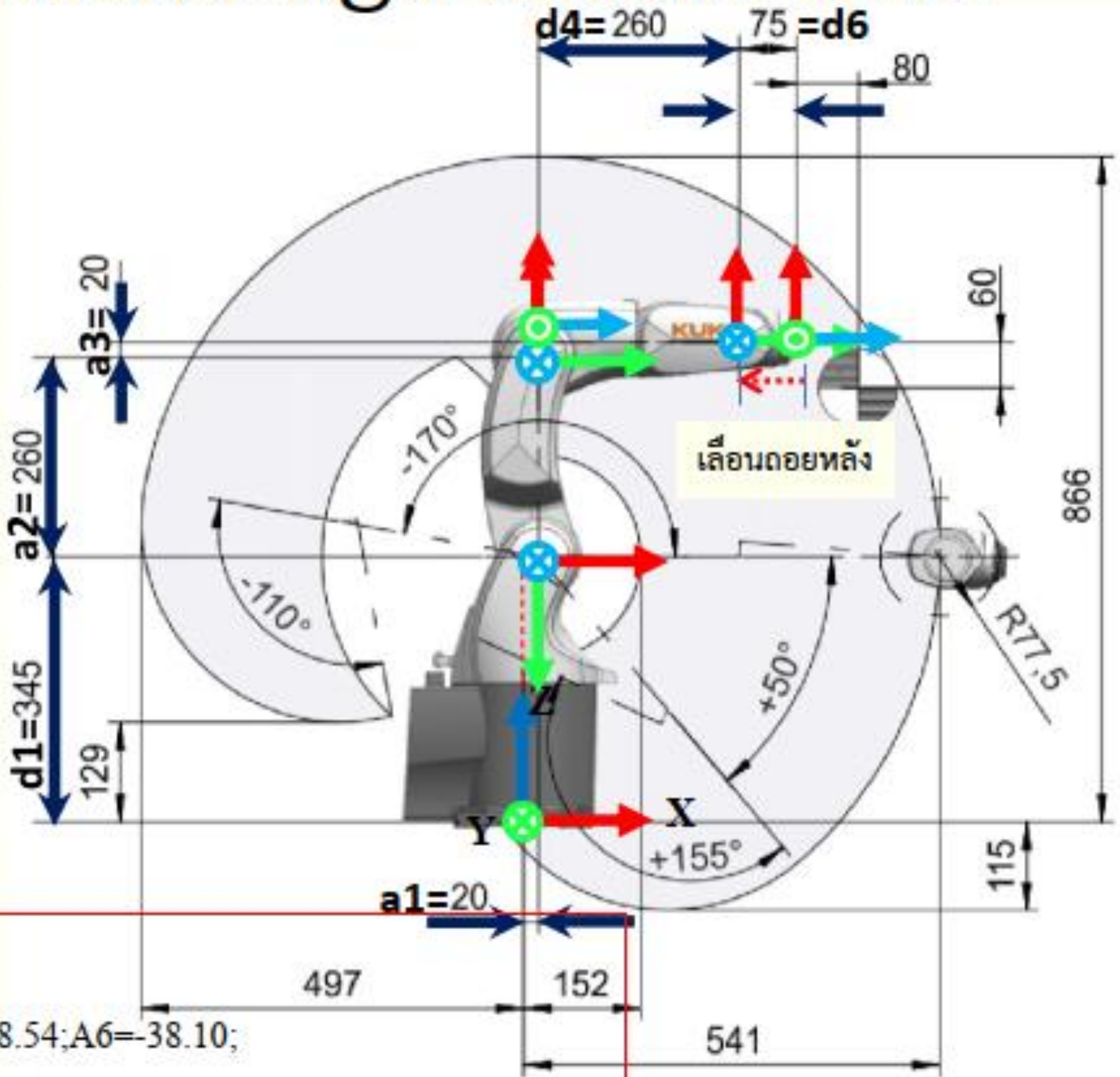
KUKA KR3



Denavit-Hartenberg Parameters

Kuka KR3:: 6 axis, RRRRRR, stdDH,

j	theta	d	a	alpha
1	q1	0.345	0.02	-1.5708
2	q2	0	0.26	0
3	q3	0	0.02	-1.5708
4	q4	0.26	0	1.5708
5	q5	0	0	-1.5708
6	q6	0.075	0	0




```
clear
mdl_KR3; KR3 ; KR3.plot(qz); KR3.teach;
A1=-12.09;A2=-1.71;A3=31.85;A4=21.42;A5=18.54;A6=-38.10;
A=168.11, B=42.06, C=170.40
deg = pi/180
ZYX=rotz(A*deg)*roty(B*deg)*rotx(C*deg)
KR3.fkine([(-A1)*deg (A2)*deg (A3-90)*deg (-A4)*deg (A5 )*deg -(180+A6))*deg])
```

KUKA KR3

```
clear

mdl_KR3

q1 = [0 0 pi/2 0 0 -pi]

TE = KR3.fkine(q1)

KR3.tool =  SE3(0, 0, 0.2);

TE2 = KR3.fkine(q1)

 KR3.base =  SE3(0, 0, 30*0.0254);

KR3.fkine(q1)
```

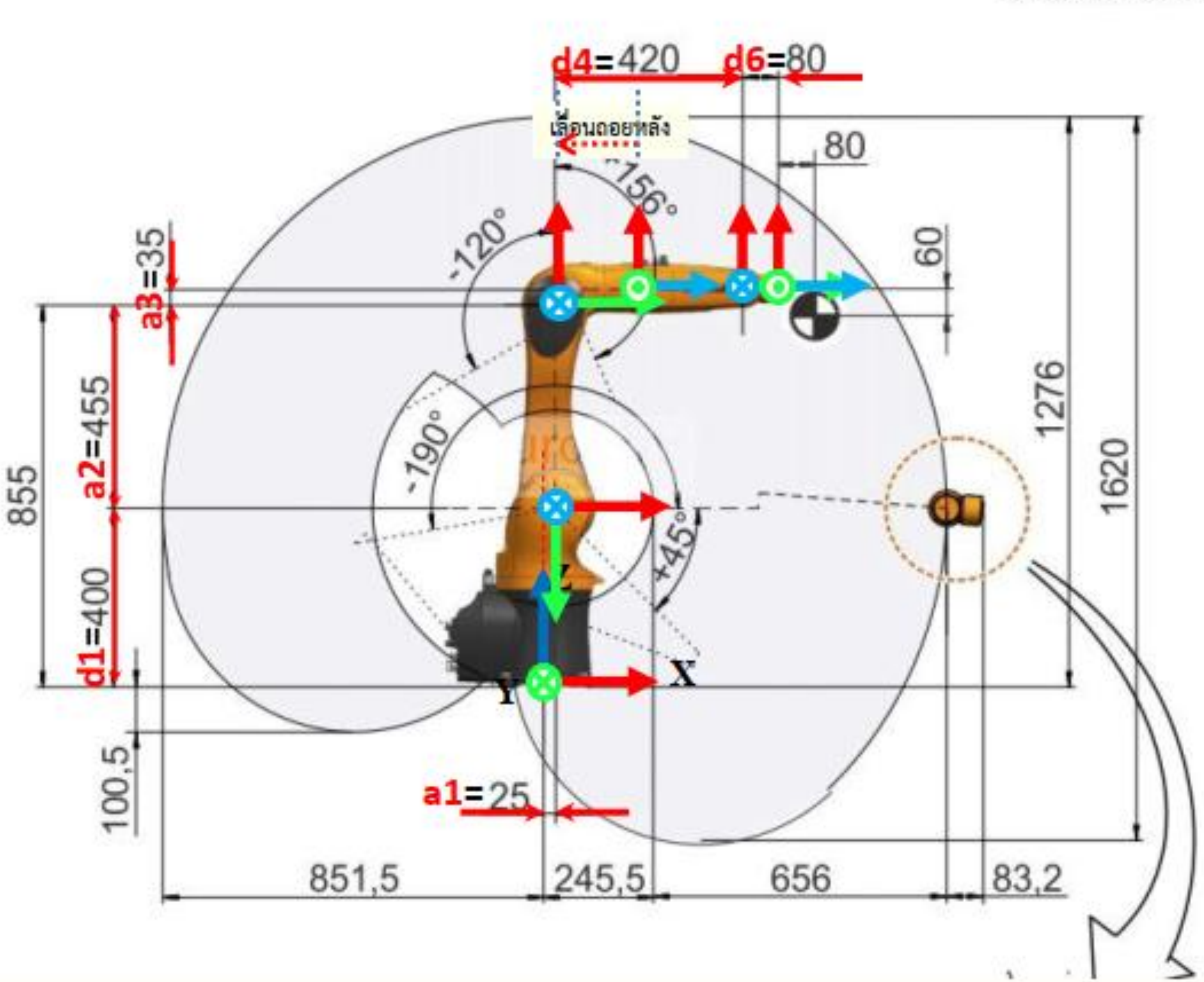
Numerical Solution of KUKA KR3

```
clear
mdl_KR3
qn = [0 0 -pi/2 -pi/4 0  0]
T1 = KR3.fkine(qn)
qi = KR3.ikine(T1)
T2 = KR3.fkine(qi)
KR3.plot(qi)
figure()
KR3.plot(qn)
```

Pose in 3D:Homogeneous Transformation Matrix of KUKA KR3

```
clear

A1=-12.09;A2=-1.71;A3=31.85;A4=21.42;A5=18.54;A6=-38.10;

A=168.11; B=42.06; C=170.40;

T1 = rotz(-A1,'deg'); T1(4,4)=1;  T1 = transl(0,0,0)*T1;

T2 = rotx(-90,'deg')*rotz(A2+90,'deg'); T2(4,4)=1;T2 =transl(0.020,0,0.345)*T2;

T3 = rotz(-90,'deg')*rotz(A3-90,'deg'); T3(4,4)=1 ;T3 = transl(0,-0.260,0)*T3;

T4 = rotx(-90,'deg')*rotz(-A4,'deg'); T4(4,4)=1 ;T4 = transl(0.020,0,0)*T4;

T5 = rotx(90,'deg')*rotz(A5,'deg');  T5(4,4)=1;T5 = transl(0,0,0.260)*T5;

T6 = rotx(-90,'deg')*rotz(-(180+A6),'deg'); T6(4,4)=1;T6 = transl(0,0.075,0)*T6;

T = T1*T2*T3*T4*T5*T6 %Homogeneous Transformation at tool frame
```

KUKA KR6



Dimensions: mm
d4=420
d6=80
เลื่อนถอยหลัง
80
-156°
-120°
60
a3=35
855
a2=455
d1=400
-190°
1276
1620
+45°
Z
X
Y
100,5
a1=25
851,5
245,5
656
83,2

DH Parameters for KUKA KR6

Kuka KR6:: 6 axis, RRRRRR, stdDH, slowRNE

j	theta	d	a	alpha	offset
1	q1	0.4	0.025	-1.5708	0
2	q2	0	0.455	0	0
3	q3	0	0.035	-1.5708	0
4	q4	0.42	0	1.5708	0
5	q5	0	0	-1.5708	0
6	q6	0.08	0	0	0

Pose in 3D:Homogeneous Transformation Matrix of KUKA KR6

```
clear
A1=0;A2=-90;A3=90;A4=0;A5=0;A6=0;
A=-116.09; B=90; C=-116.09;
T1 = rotz(-A1,'deg'); T1(4,4)=1;  T1 = transl(0,0,0)*T1;
T2 = rotx(-90,'deg')*rotz(A2+90,'deg'); T2(4,4)=1;T2 =transl(0.025,0,0.400)*T2;
T3 = rotz(-90,'deg')*rotz(A3-90,'deg'); T3(4,4)=1 ;T3 = transl(0,-0.455,0)*T3;
T4 = rotx(-90,'deg')*rotz(-A4,'deg'); T4(4,4)=1 ;T4 = transl(0.035,0,0)*T4;
T5 = rotx(90,'deg')*rotz(A5,'deg');  T5(4,4)=1;T5 = transl(0,0,0.420)*T5;
T6 = rotx(-90,'deg')*rotz(-(180+A6),'deg'); T6(4,4)=1;T6 = transl(0,0.080,0)*T6;
T = T1*T2*T3*T4*T5*T6 %Homogeneous Transformation at tool frame
```

Pose in 3D:Homogeneous Transformation Matrix of KUKA KR6

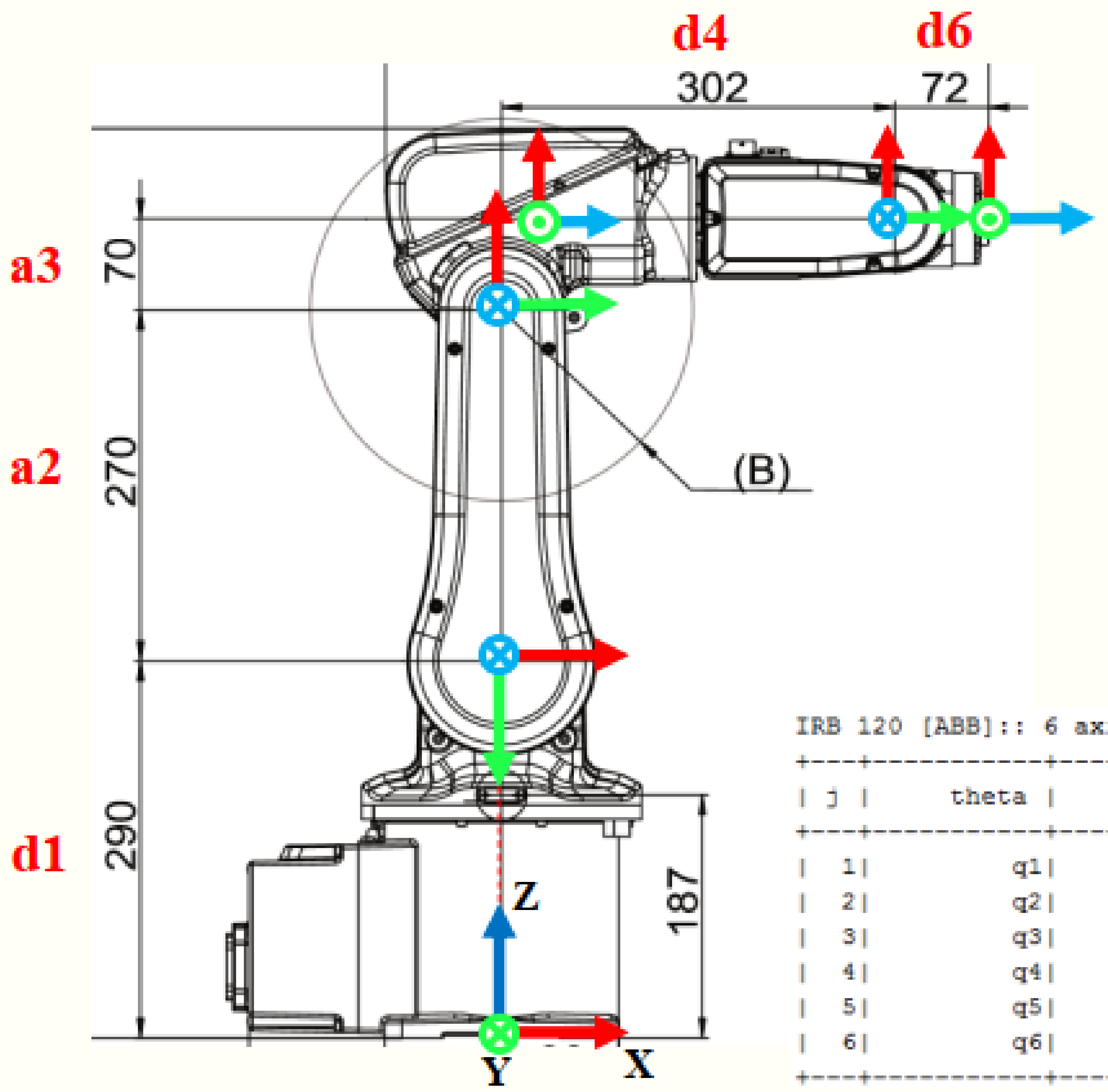
%ZYX=$R_z(\theta_y)\ R_y(\theta_p)\ R_x(\theta_r)$

```
A=-116.09;  % yaw

B=90;       % pitch

C=-116.09;  % roll

RPY1 = rpy2tr(C, B, A ,'deg') %

RPY2 = rotz(A ,'deg') * roty(B ,'deg') * rotx(C ,'deg')
```

KUKA KR6

```
clear

mdl_KR6; KR6 ; KR6.plot(qz); KR6.teach;

A1=0;A2=-90;A3=90;A4=0;A5=0;A6=0;

A=-116.09; B=90; C=-116.09;

deg = pi/180

ZYX=rotz(A*deg)*roty(B*deg)*rotx(C*deg)

KR6.fkine([(-A1)*deg (A2)*deg (A3-90)*deg (-A4)*deg (A5
)*deg -(180+A6)*deg])
```

ABB IRB120



IRB 120 [ABB]:: 6 axis, RRRRRR, stdDH, slowRNE

j	theta	d	a	alpha	offset
1	q1	0.29	0	-1.5708	0
2	q2	0	0.27	0	0
3	q3	0	0.07	-1.5708	0
4	q4	0.302	0	1.5708	0
5	q5	0	0	-1.5708	0
6	q6	0.072	0	0	0

DH Parameters for ABB IRB120

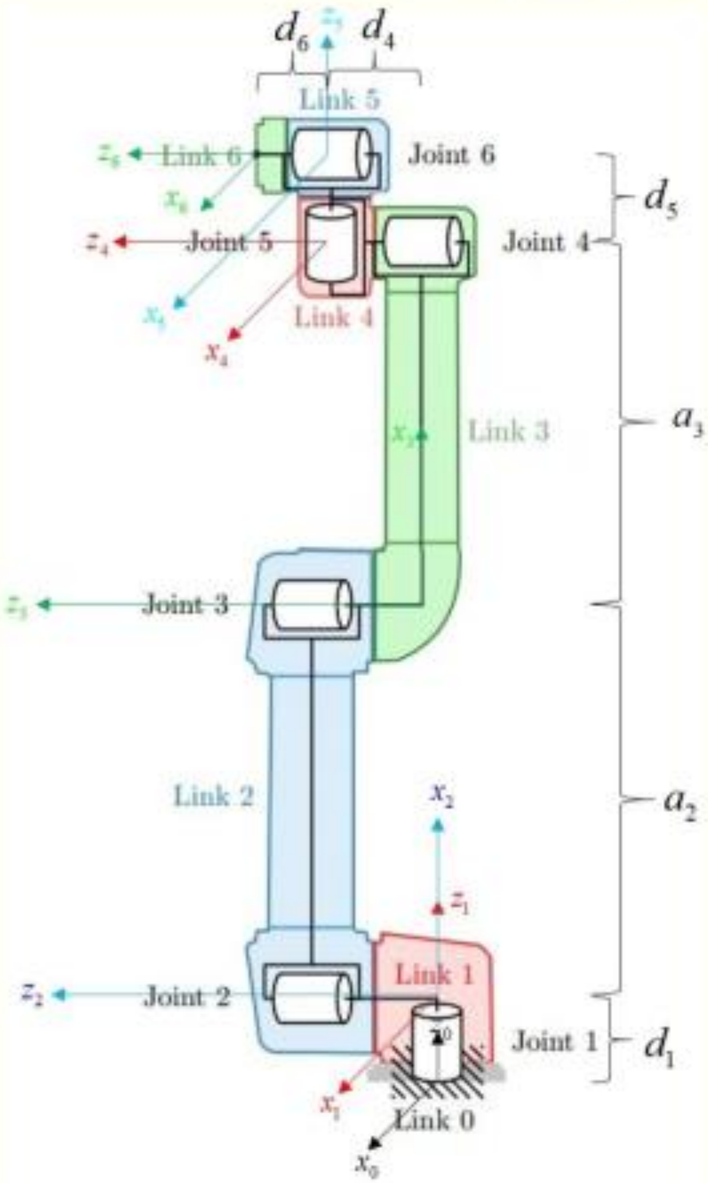
IRB 120 [ABB]:: 6 axis, RRRRRR, stdDH, slowRNE

j	theta	d	a	alpha	offset
1	q1	0.29	0	-1.5708	0
2	q2	0	0.27	0	0
3	q3	0	0.07	-1.5708	0
4	q4	0.302	0	1.5708	0
5	q5	0	0	-1.5708	0
6	q6	0.072	0	0	0

Pose in 3D:Homogeneous Transformation Matrix of ABB IRB120

```
clear

A1=-2.0; A2=-47.3; A3=25.0; A4=2.6; A5=80.7; A6=88.7;

T1 = rotz(A1,'deg'); T1(4,4)=1;T1 = transl(0,0,0)*T1;

T2 = rotx(-90,'deg')*rotz(A2,'deg'); T2(4,4)=1;T2 = transl(0,0,0.29)*T2;

T3 = rotz(-90,'deg')*rotz(A3,'deg'); T3(4,4)=1;T3 = transl(0,-0.27,0)*T3;

T4 = rotx(-90,'deg')*rotz(A4,'deg'); T4(4,4)=1;T4 = transl(0.07,0,0)*T4;

T5 = rotx(90,'deg')*rotz(A5,'deg');  T5(4,4)=1;T5 = transl(0,0,0.302)*T5;

T6 = rotx(-90,'deg')*rotz(180+A6,'deg');T6(4,4)=1;T6 = transl(0,0.072,0)*T6;

T = T1*T2*T3*T4*T5*T6 %Homogeneous Transformation at tool frame
```

Inverse Kinematics: UR5

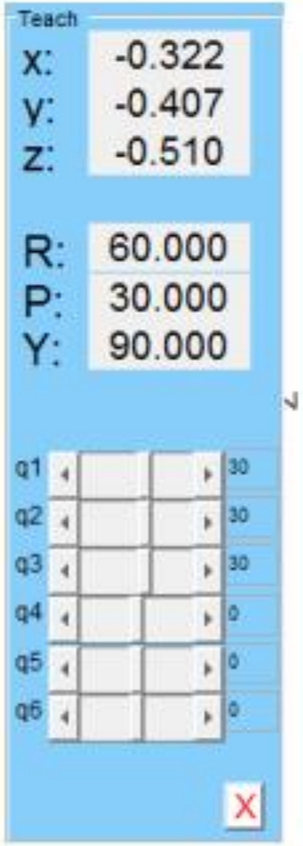


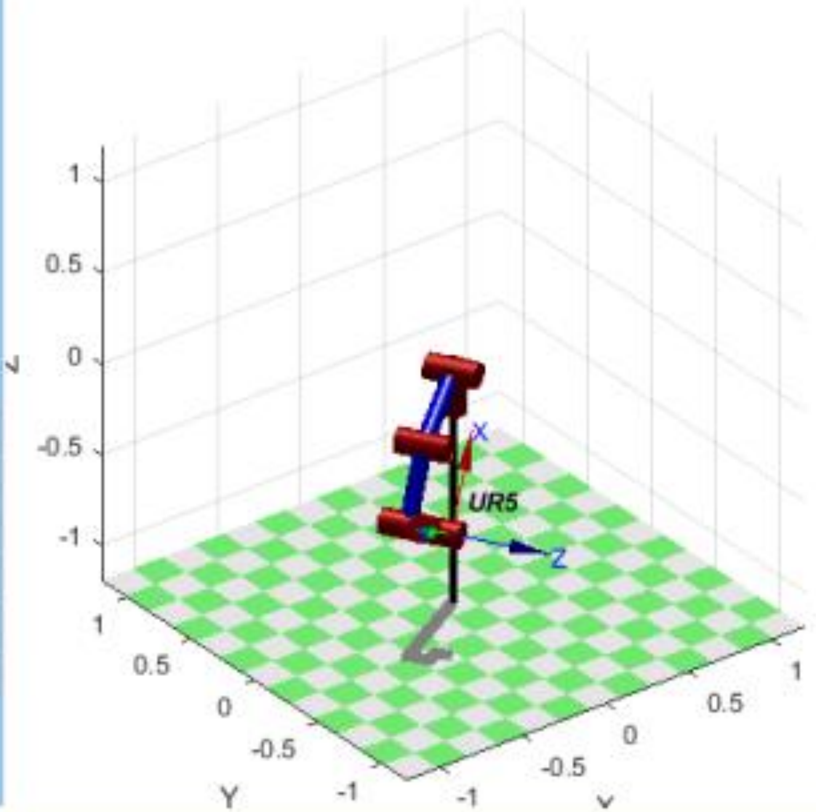
mdl_ur5

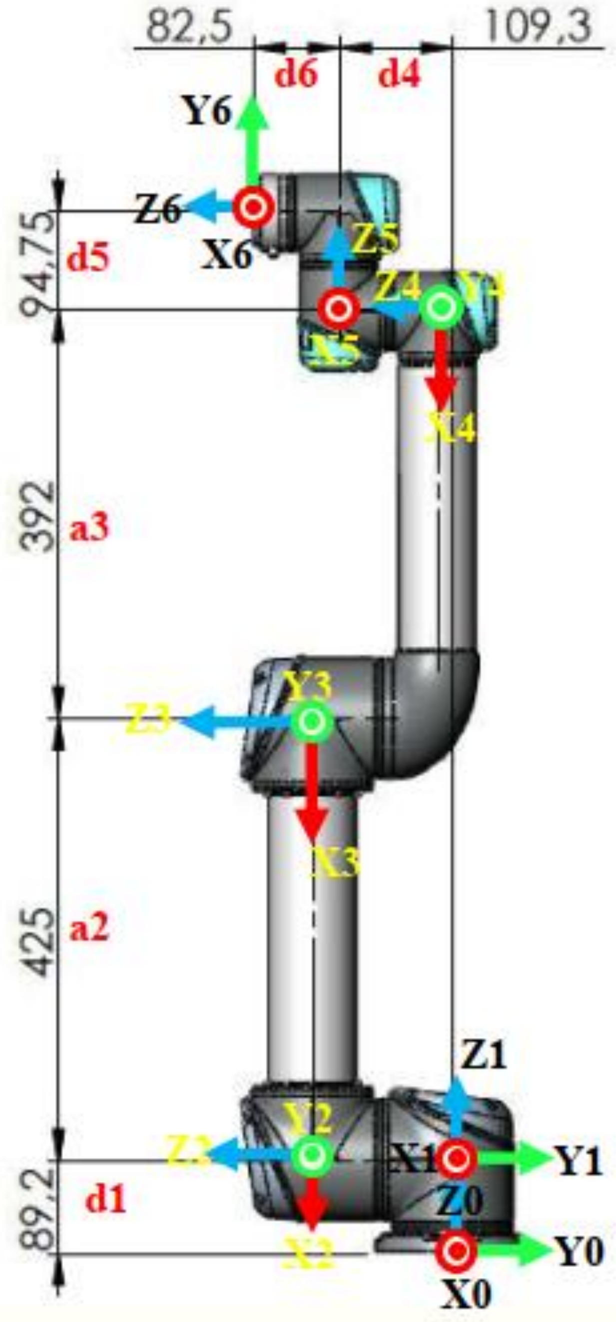
ur5

ur5.teach

ur5.fkine([pi/6 pi/6 pi/6 0 0 0])



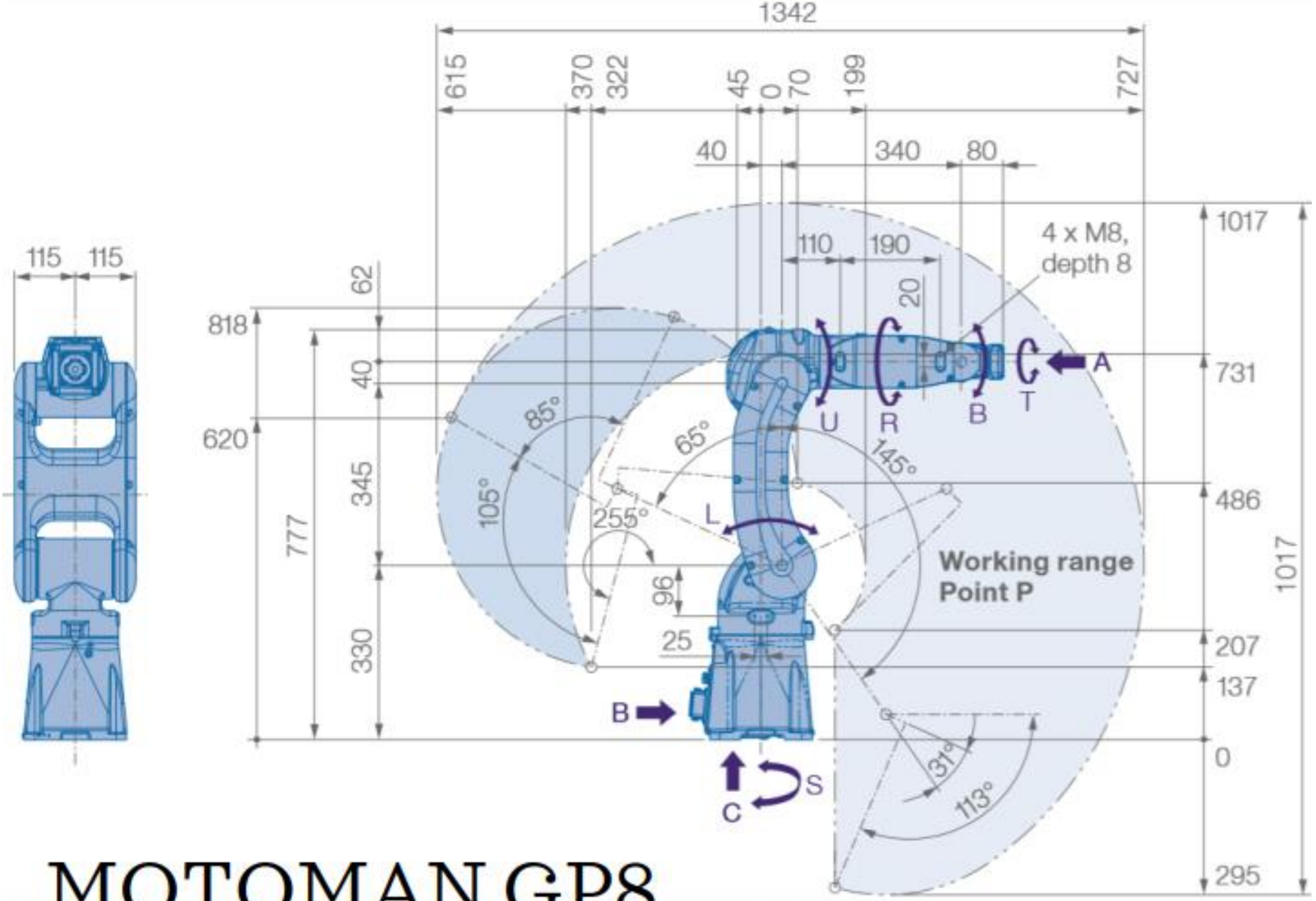



82,5
109,3
d6
d4
Y6
Z6
X6
Z5
Z4
Y4
X5
X4
94,75
d5
392
a3
Y3
Z3
X3
425
a2
Z1
Y2
Z2
X1
Y1
Z0
89,2
d1
X2
Y0
X0

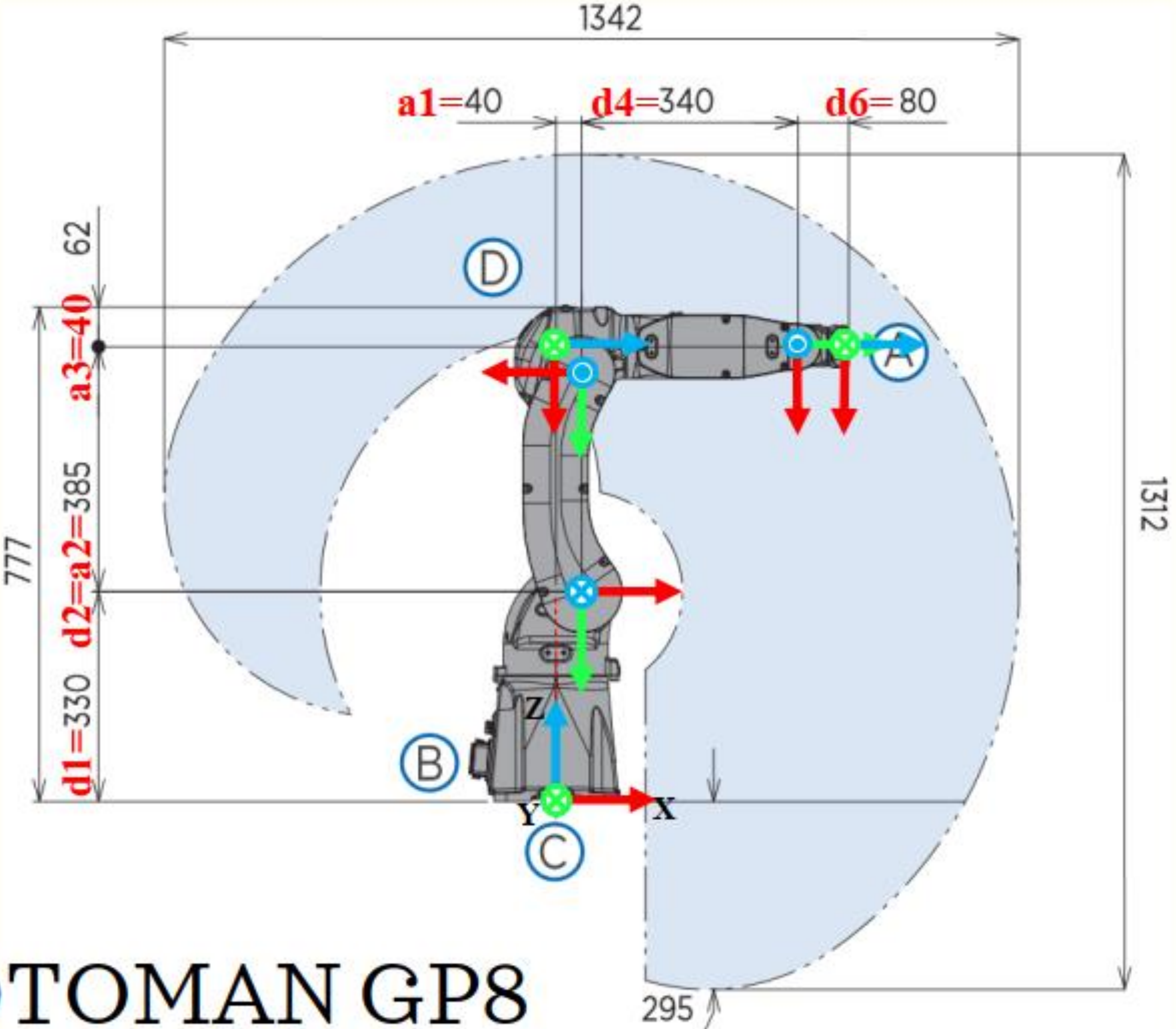
DH Parameters for UR5

UR5 [Universal Robotics]:: 6 axis, RRRRRR, stdDH, slowRNE

j	θ_j	d_j	a_{j-1}	$alpha_{j-1}$	offset
1	q1	0.089459	0	1.5708	0
2	q2	0	-0.425	0	0
3	q3	0	-0.39225	0	0
4	q4	0.10915	0	1.5708	0
5	q5	0.09465	0	-1.5708	0
6	q6	0.0823	0	0	0


1342
615
370
322
45
0
70
199
727
40
340
80
110
190
20
4 x M8, depth 8
115
115
62
818
40
620
345
777
330
96
25
85°
105°
65°
255°
145°
31°
113°
U
R
B
T
A
L
B
C
S
Working range Point P
1017
731
486
207
137
0
295
1017

MOTOMAN GP8

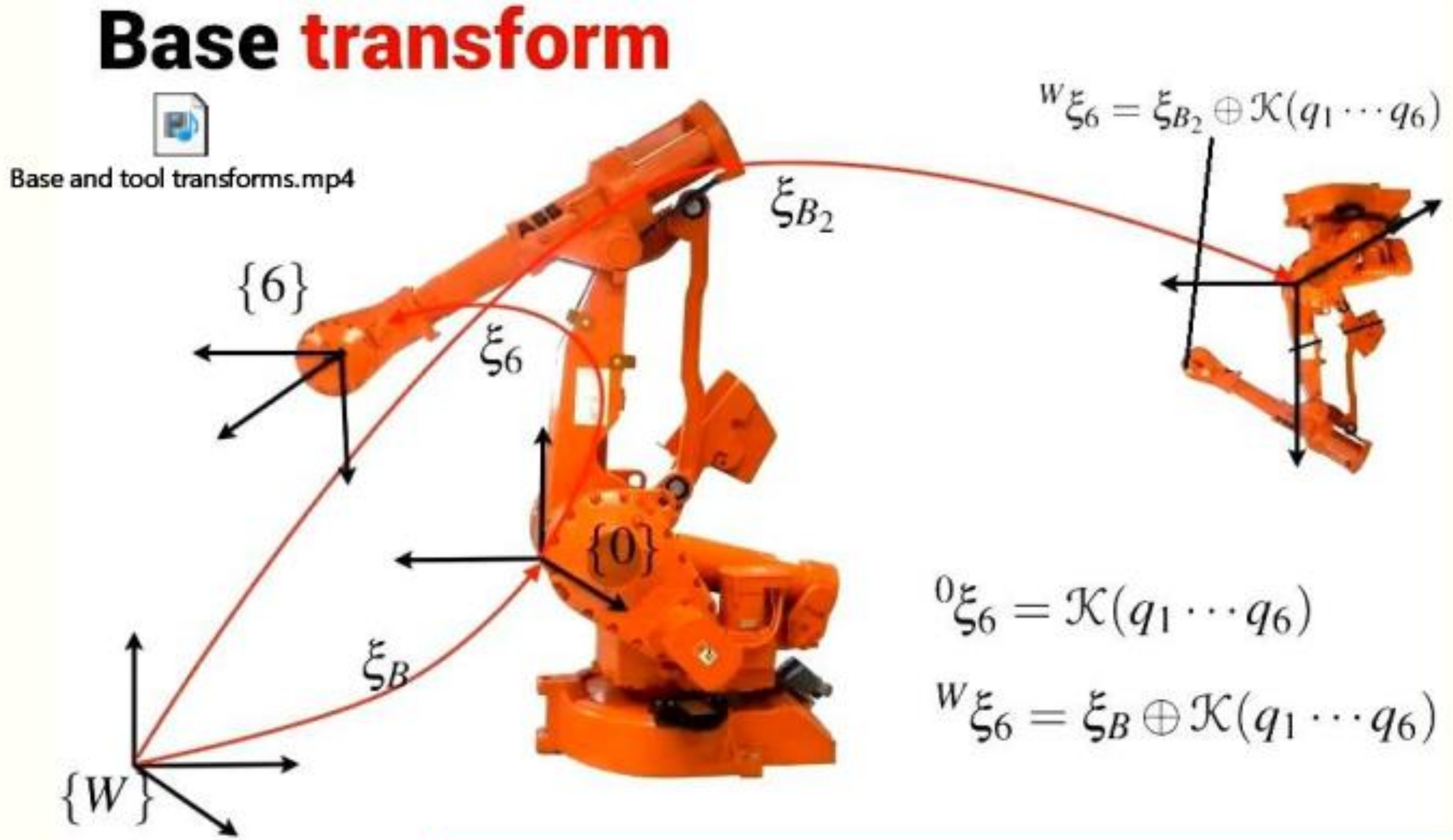

1342
a1=40
d4=340
d6= 80
62
a3=40
777
d2=a2=385
d1=330
1312
295
D
A
B
C
Z
X
Y

MOTOMAN GP8

Pose in 3D:Homogeneous Transformation Matrix of MOTOMAN GP8

```
clear

A1=46.5087; A2=-15.3869; A3=-32.6821; A4=61.8562; A5=-42.0350; A6=-121.5410;

Rx=-133.2223; Ry=-40.2914; Rz=-56.0222;X=390.422; Y=187.121; Z=180.826;

T1 = rotz(A1,'deg'); T1(4,4)=1;T1 = transl(0,0,0)*T1;

T2 = rotx(-90,'deg')*rotz(A2-90,'deg'); T2(4,4)=1;T2 = transl(0.04,0,0.33)*T2;

T3 = roty(-180,'deg')*rotz(A3,'deg'); T3(4,4)=1;T3 = transl(0,-0.345,0)*T3;

T4 = rotx(-90,'deg')*roty(-90,'deg')*rotz(A4,'deg'); T4(4,4)=1;T4 = transl(0.04,-0.04,0)*T4;

T5 = rotx(90,'deg')*rotz(A5,'deg');  T5(4,4)=1;T5 = transl(0,0,0.34)*T5;

T6 = rotx(-90,'deg')*rotz(180+A6,'deg');T6(4,4)=1;T6 = transl(0,0.08,0)*T6;

T = T1*T2*T3*T4*T5*T6 %Homogeneous Transformation at tool frame
```

Base & Tool Transform




```
mdl_puma560; p560 ; p560.fkine([0.1 0.2 0.3 0 0 0])
p560.base = transl(10, 15, 2) ; p560.fkine([0.1 0.2 0.3 0 0 0])
```

https://robotacademy.net.au/