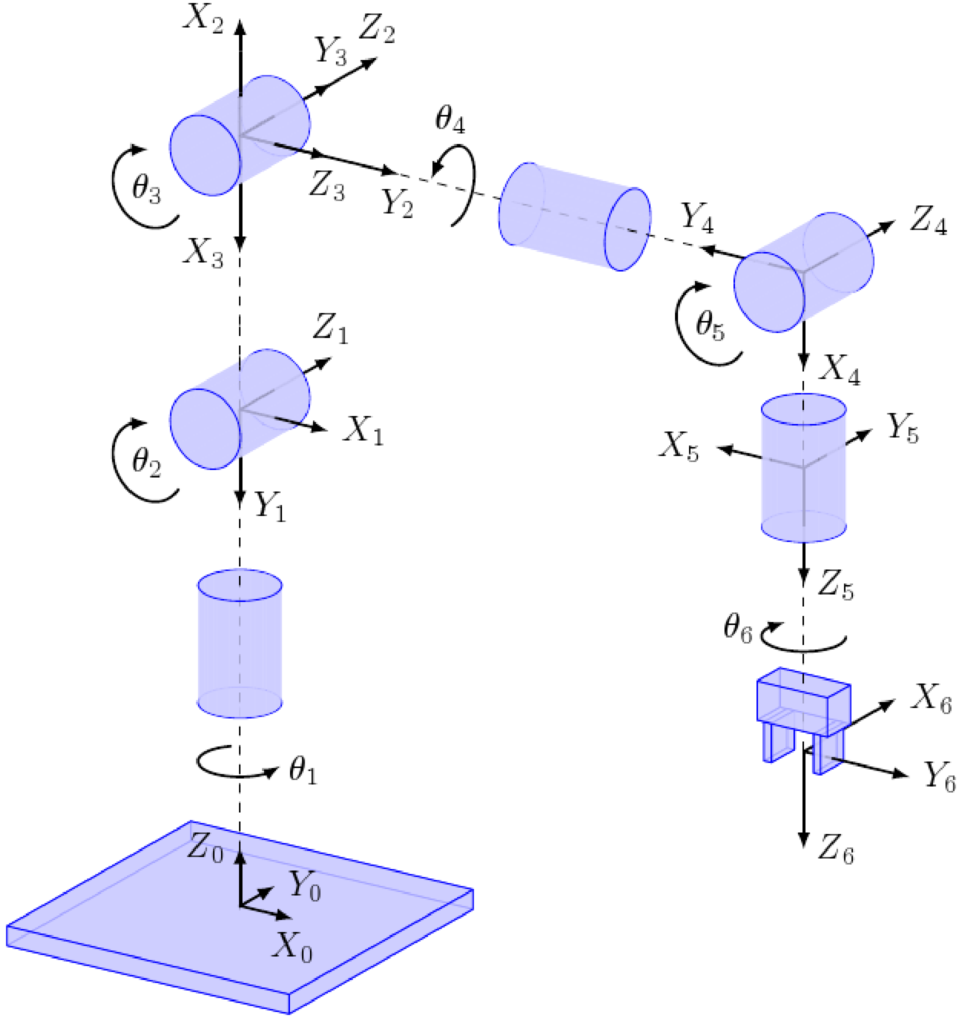
Kinematics and Dynamics of Robots

Module 6

Forward Kinematics



Introduction

- The problem of manipulator kinematics is to describe the motion of a manipulator without consideration of the forces and torques causing the motion.
- ***Forward kinematics***: to determine the position and orientation of the end effector given the values for the joint variables of the robot. This problem is easily solved by attaching coordinate frames to each link of the robot and expressing the relationships among these frames as homogeneous transformations.

Kinematics Chains

A robot manipulator is composed of a set of links connected together by joints.

The joints can either be very simple, such as a revolute joint or a prismatic joint, or they can be more complex, such as a ball and socket joint

Revolute joint is like a hinge that allows a relative rotation about a single axis, and a prismatic joint permits a linear motion along a single axis, namely an extension or retraction.

The difference between the two situations is that in the first instance the joint has only a single degree-of-freedom of motion: the angle of rotation in the case of a revolute joint, and the amount of linear displacement in the case of a prismatic joint. In contrast, a ball and socket joint has two degrees of freedom.

We assume that all joints have only a ***single*** degree of freedom. This assumption does not involve any real loss of generality, since joints such as a ball and socket joint (two degrees of freedom) or a spherical wrist (three degrees of freedom) can always be thought of as a succession of single degree-of-freedom joints with links of length zero in between.

With the assumption that each joint has a single degree-of-freedom, the action of each joint can be described by a single real number: the ***angle of rotation*** in the case of a revolute joint or the ***displacement*** in the case of a prismatic joint.

A robot manipulator with ***n joints*** will have ***n + 1 links***, since each joint connects two links. We number the ***joints*** from ***1 to n***, and we number the links from ***0 to n***, starting from the base.

By this convention, ***joint i*** connects ***link i − 1*** to ***link i***. We will consider ***the location of joint i to be fixed with respect to link i − 1***. When joint i is actuated, link i moves. Therefore, link 0 (the first link or base) is fixed, and does not move when the joints are actuated.

Of course, the robot manipulator could itself be mobile (e.g., it could be mounted on a mobile platform or on an autonomous vehicle), but this case can be handled easily by slightly extending the techniques presented here.

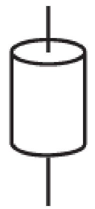
With the ith joint, we associate a ***joint variable***, denoted by q_i. In the case of a revolute joint, q_i is the ***angle of rotation***, and in the case of a ***prismatic joint***, q_i is the ***joint displacement***:

$$q_i = \begin{cases} \theta_i & \text{if joint } i \text{ is revolute} \\ d_i & \text{if joint } i \text{ is prismatic} \end{cases}$$

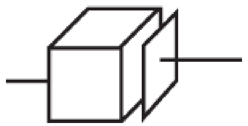
To perform the kinematic analysis, we attach a coordinate frame rigidly to each link. In particular, we attach $o_i x_i y_i z_i$ to link i. This means that, whatever motion the robot executes, the coordinates of each point on link i are constant when expressed in the ith coordinate frame. Furthermore, when joint i is actuated, link i and its attached frame, $o_i x_i y_i z_i$, experience a resulting motion. The frame $o_0 x_0 y_0 z_0$, which is attached to the robot base, is referred to as the ***base frame***, ***inertial frame*** or ***world frame***.

Making Universal Rules for Robotics Engineers

- Now it is time to make some rules so when we draw a diagram any robotic engineer around the world gets our idea.
- We are showing each joint a degree of freedom (***DOF***), a 6 DOF robot has 6 joints.
- Each joint can only be revolute (rotational movement) or prismatic (translational movement), therefore if a joint can do a combination of these movements, it will be considered multiple joints. We show a revolute joint with a cylinder and a prismatic joint with a cube.



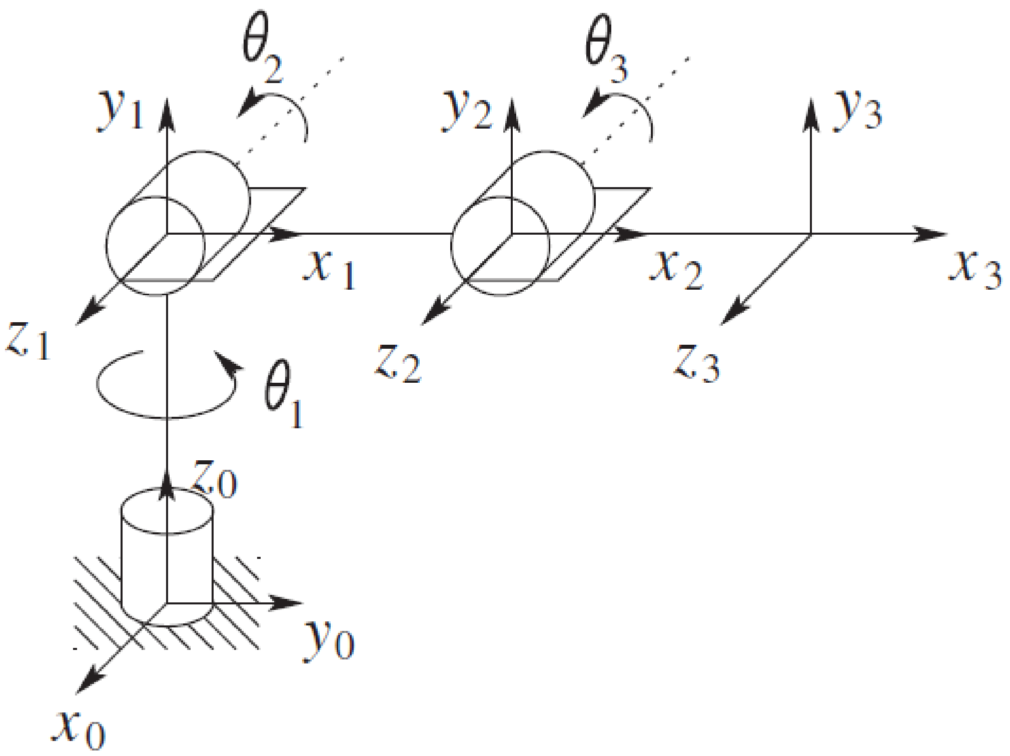
Revolute joint



Prismatic joint

HUMBER

- To perform the kinematic analysis, we attach a coordinate frame rigidly to each link. In particular, we attach $o_i x_i y_i z_i$ to link i. This means that, whatever motion the robot executes, the coordinates of each point on link i are constant when expressed in the i^{th} coordinate frame. Furthermore, when joint i is actuated, link i and its attached frame, $o_i x_i y_i z_i$, experience a resulting motion. The frame $o_0 x_0 y_0 z_0$, which is attached to the robot base, is referred to as the **base frame**, **inertial frame** or **world frame.**
- We call the parameter that changes when a joint moves ***Joint Variable (θ and d)***.



Now, suppose A_i is the homogeneous transformation matrix that gives the position and orientation of $o_ix_iy_iz_i$ with respect to $o_{i-1}x_{i-1}y_{i-1}z_{i-1}$. The matrix A_i is not constant, but varies as the configuration of the robot is changed. However, the assumption that all joints are either revolute or prismatic means that A_i is a function of only a single joint variable, namely q_i. In other words,

$$A_i = A_i(q_i)$$

The homogeneous transformation matrix that expresses the position and orientation of $o_jx_jy_jz_j$ with respect to $o_ix_iy_iz_i$ is called a ***transformation matrix***, and is denoted by T_j^i. We have

$$T_j^i = \begin{cases} A_{i+1}A_{i+2}\cdots A_{j-1}A_j & \text{if } i < j \\ I & \text{if } i = j \\ (T_i^j)^{-1} & \text{if } j > i \end{cases}$$

By the manner in which we have rigidly attached the various frames to the corresponding links, it follows that the position of any point on the end effector when expressed in the last frame *n* is a constant independent of the configuration of the robot. We denote the position and orientation of the end effector with respect to the inertial or base frame by a three-vector o_n^0 (which gives the coordinates of the origin of the end-effector frame with respect to the base frame) and the 3 × 3 rotation matrix R_n^0, and define the homogeneous transformation matrix

$$H = \begin{bmatrix} R_n^0 & o_n^0 \\ 0 & 1 \end{bmatrix}$$

Then the position and orientation of the end effector in the inertial frame are given by the product

$$H = T_n^0 = A_1(q_1) \cdots A_n(q_n)$$

Each homogeneous transformation A_i is of the form

$$A_i = \begin{bmatrix} R_i^{i-1} & o_i^{i-1} \\ 0 & 1 \end{bmatrix}$$

Hence, for $i < j$

$$T_j^i = A_{i+1} \cdots A_j = \begin{bmatrix} R_j^i & o_j^i \\ 0 & 1 \end{bmatrix}$$

The matrix R_j^i expresses the orientation of $o_j x_j y_j z_j$ relative to $o_i x_i y_i z_i$ and is given by the rotational parts of the *A* matrices as

$$R_j^i = R_{i+1}^i \cdots R_j^{j-1}$$

The coordinate vectors o_j^i are given recursively by the formula

$$o_j^i = o_{j-1}^i + R_{j-1}^i o_j^{j-1}$$

These expressions will be useful when we study Jacobian matrices.

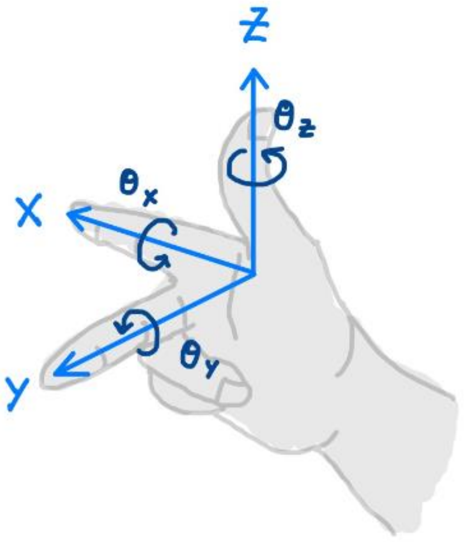
In principle, that is all there is to forward kinematics; determine the functions $A_i(q_i)$, and multiply them together as needed. However, it is possible to achieve a considerable amount of streamlining and simplification by introducing further conventions, such as the ***Denavit-Hartenberg*** representation of a joint, and this is the objective of the next section

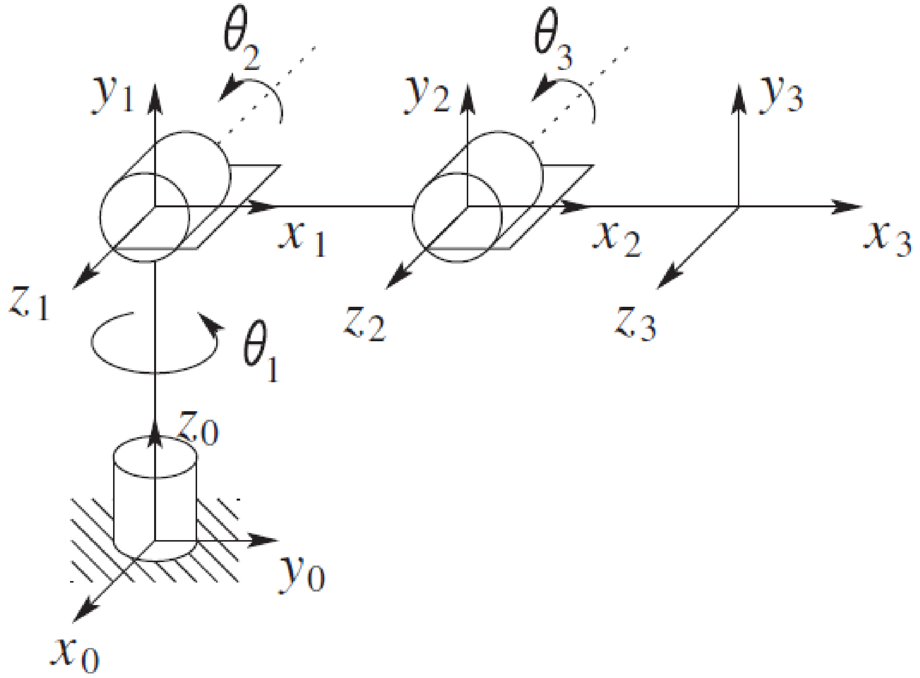
The Denavit-Hartenberg Convention

- Denavit-Hartenberg showed that the kinematics chain can be simplified significantly by a convention that they proposed.
- We do the DH convention in two steps:
  - Rules for assigning the coordinate systems according to DH conventions:
  - Rules for finding DH parameters

- We always follow DH rules for assigning the coordinate systems.

- DH rules for assigning the coordinate systems
  1. The z axis must be the axis of rotation for a revolute joint or the direction of translation for a prismatic joint. For end-effector (tool frame) since we do not have a joint, we can choose z axis as parallel the previous z.
  2. The x axis must be perpendicular both to its own z axis, and the z axis of the frame before it (except x_0).
  3. All frames must follow the right-hand tool.
  4. Each x axis (except x_0) must intersect Z axis of the previous frame.





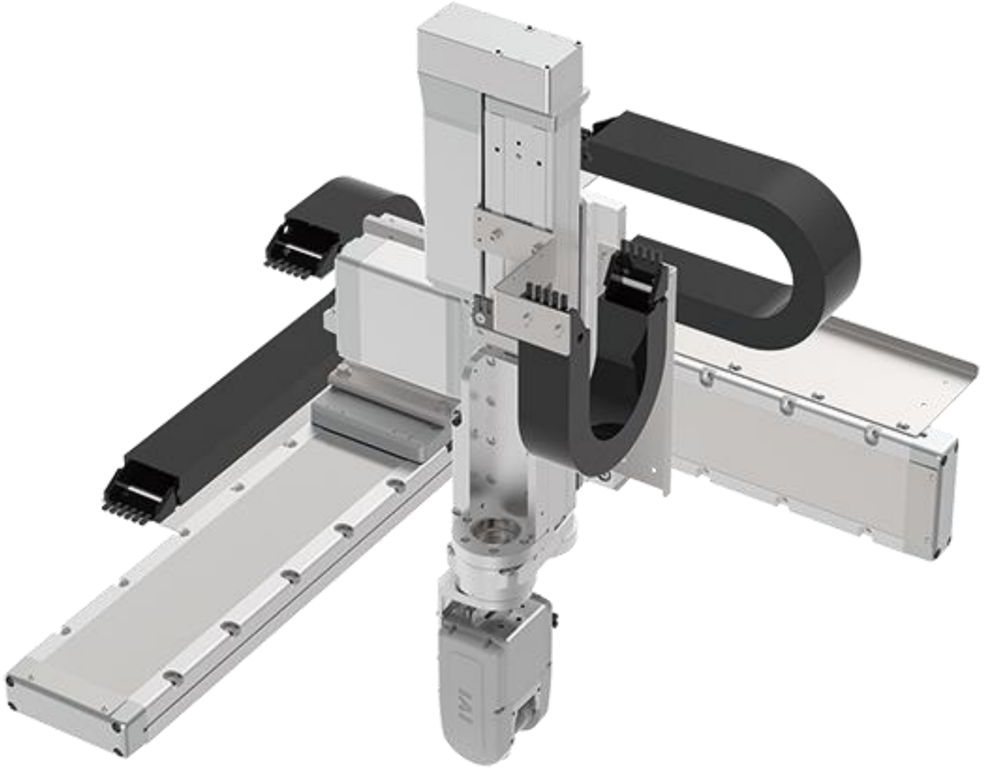
Examples: 6 Standard Types of 3-DoF Manipulators

- 5 standard types of 3-DoF manipulators are:

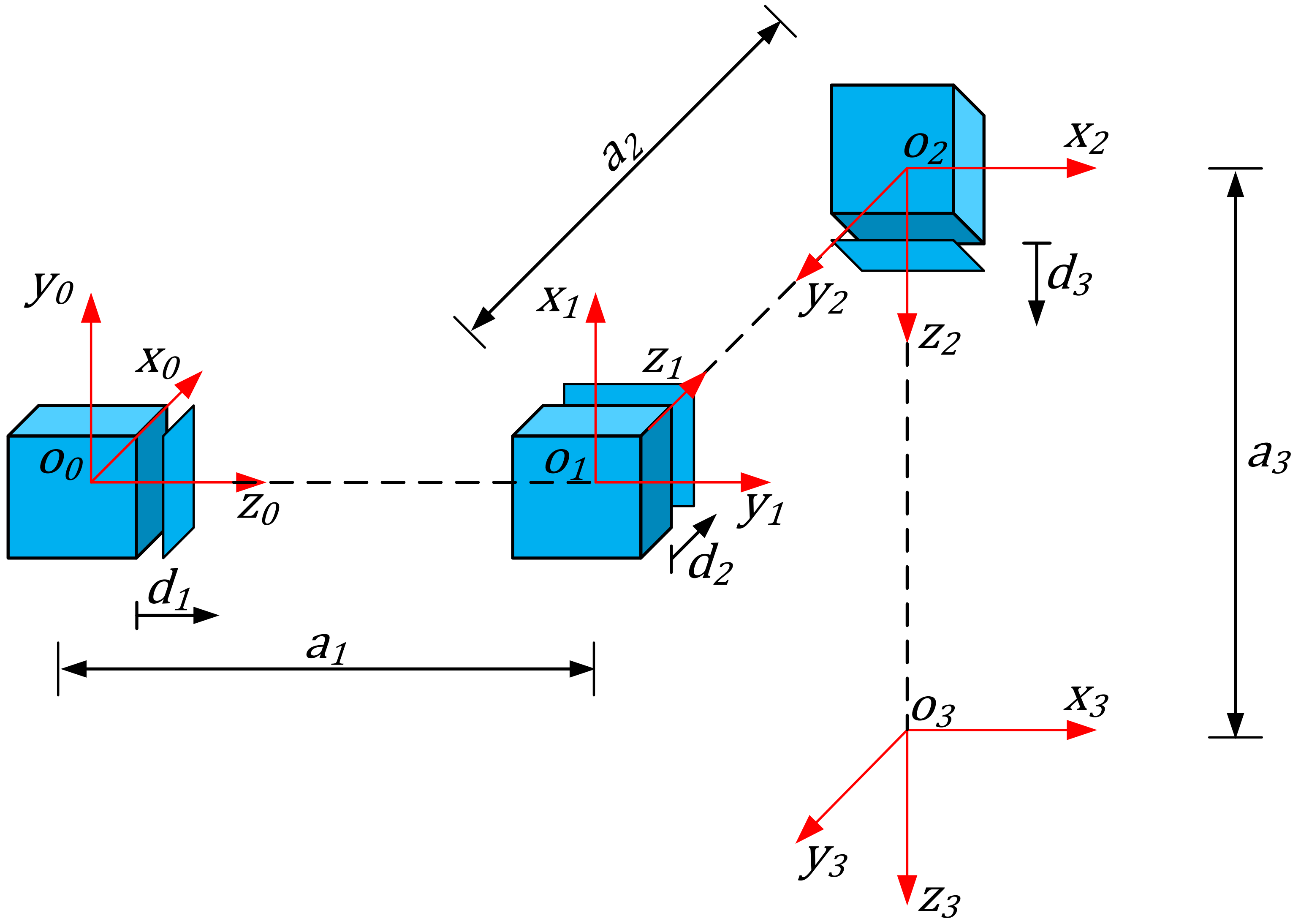
    1. Cartesian:
    2. Articulated 3 DoF
    3. Scara
    4. Spherical
    5. Cylindrical
    6. Spherical Wrist

Examples: Cartesian

- Cartesian

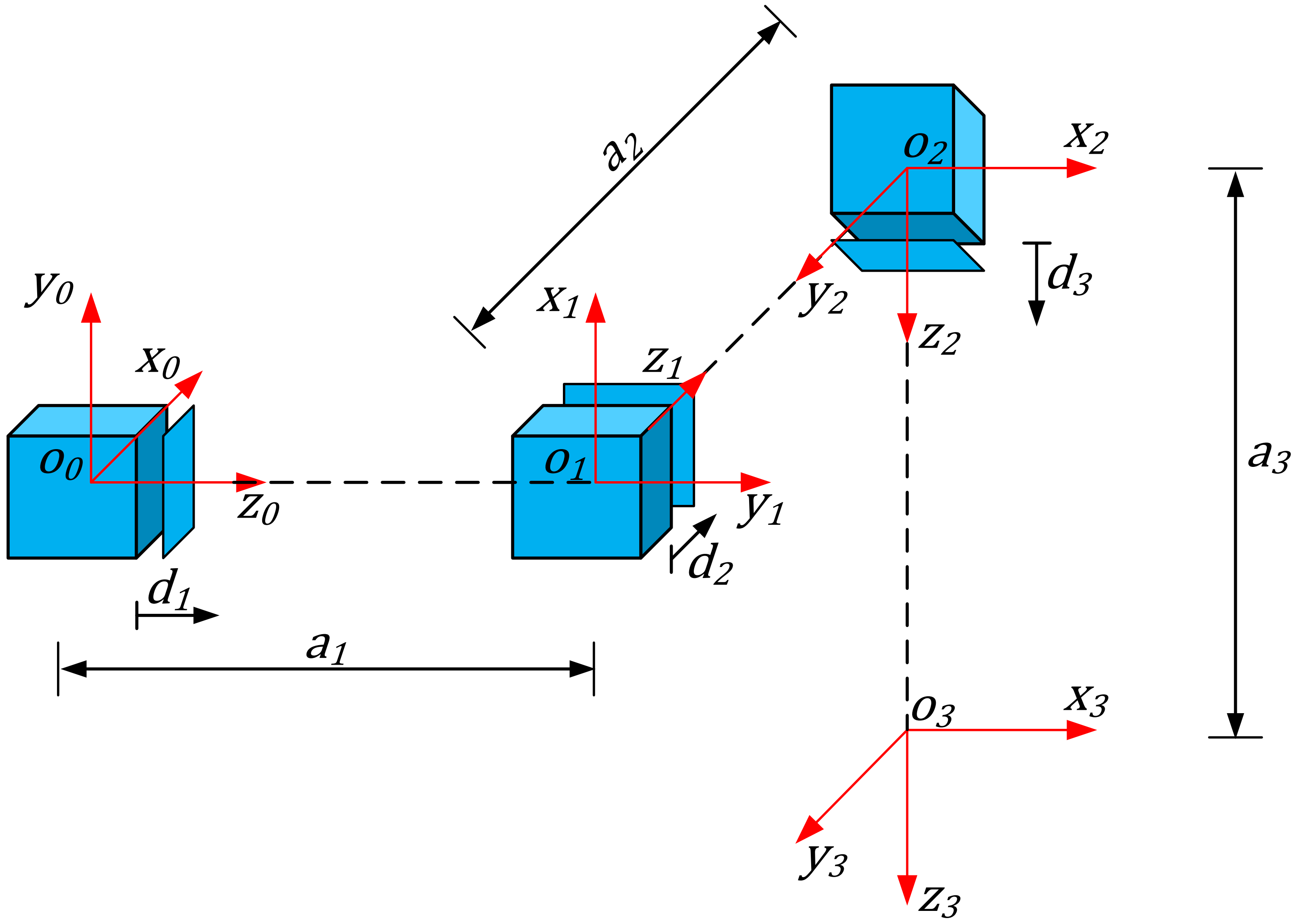


- Cartesian

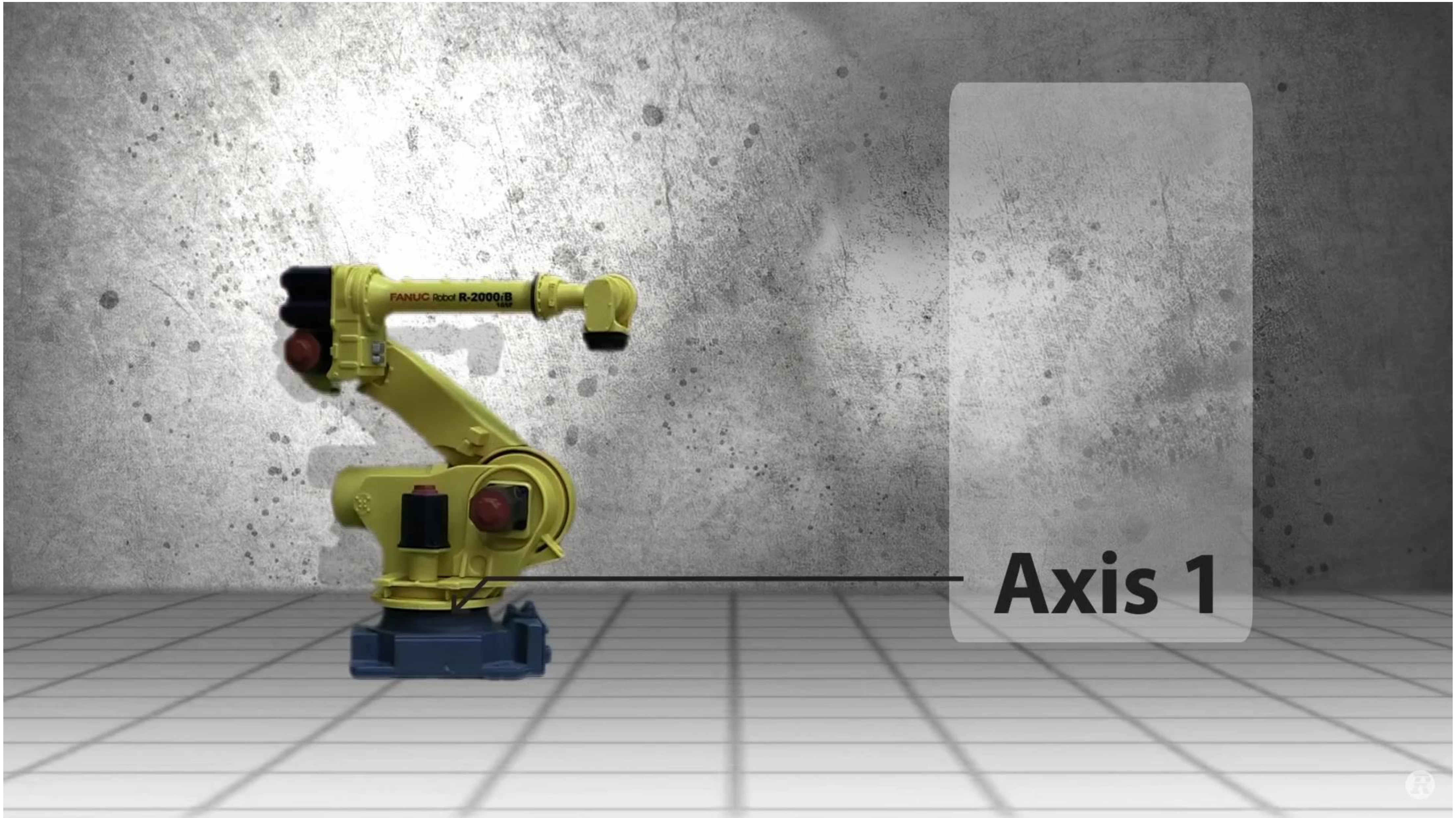


HUMBER

- Cartesian



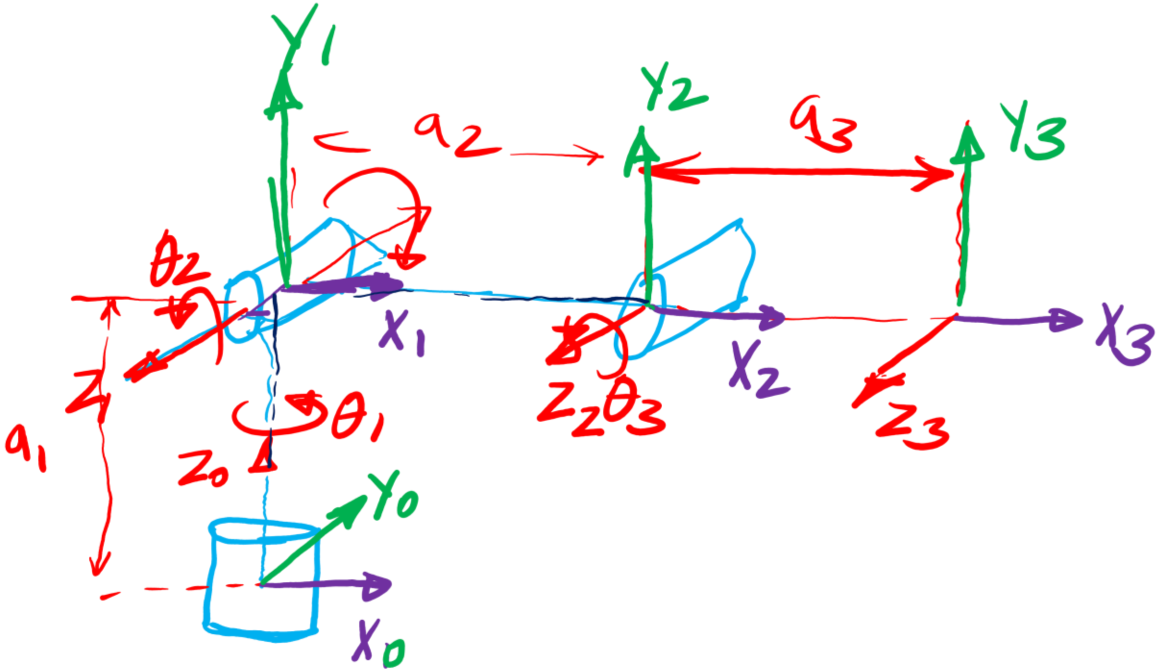
Examples: Articulate 3 DoF



HUMBER

- Articulate 3 DoF: $(\theta_1, \theta_2, \theta_3)$

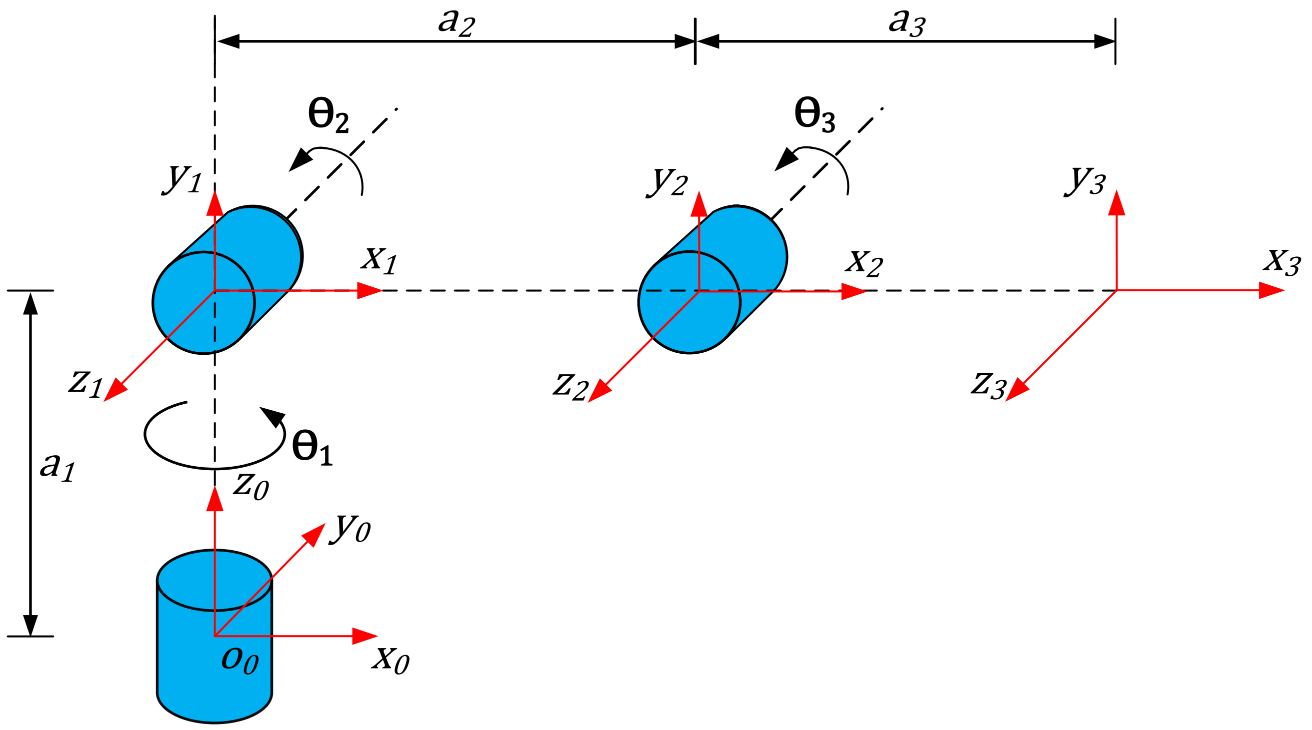
link: 4: (0 - 3)



✓

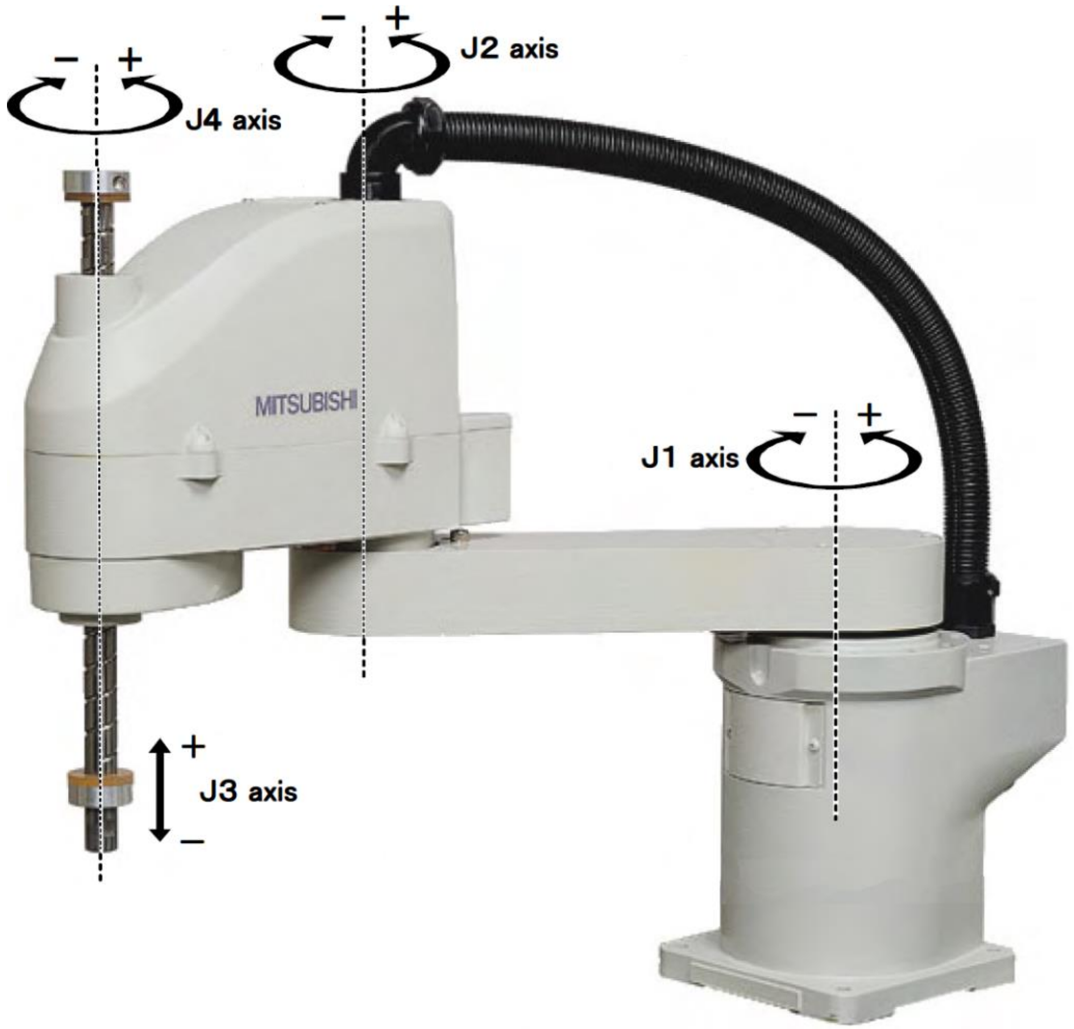
Examples: Articulate 3 DoF

- Articulate 3 DoF:



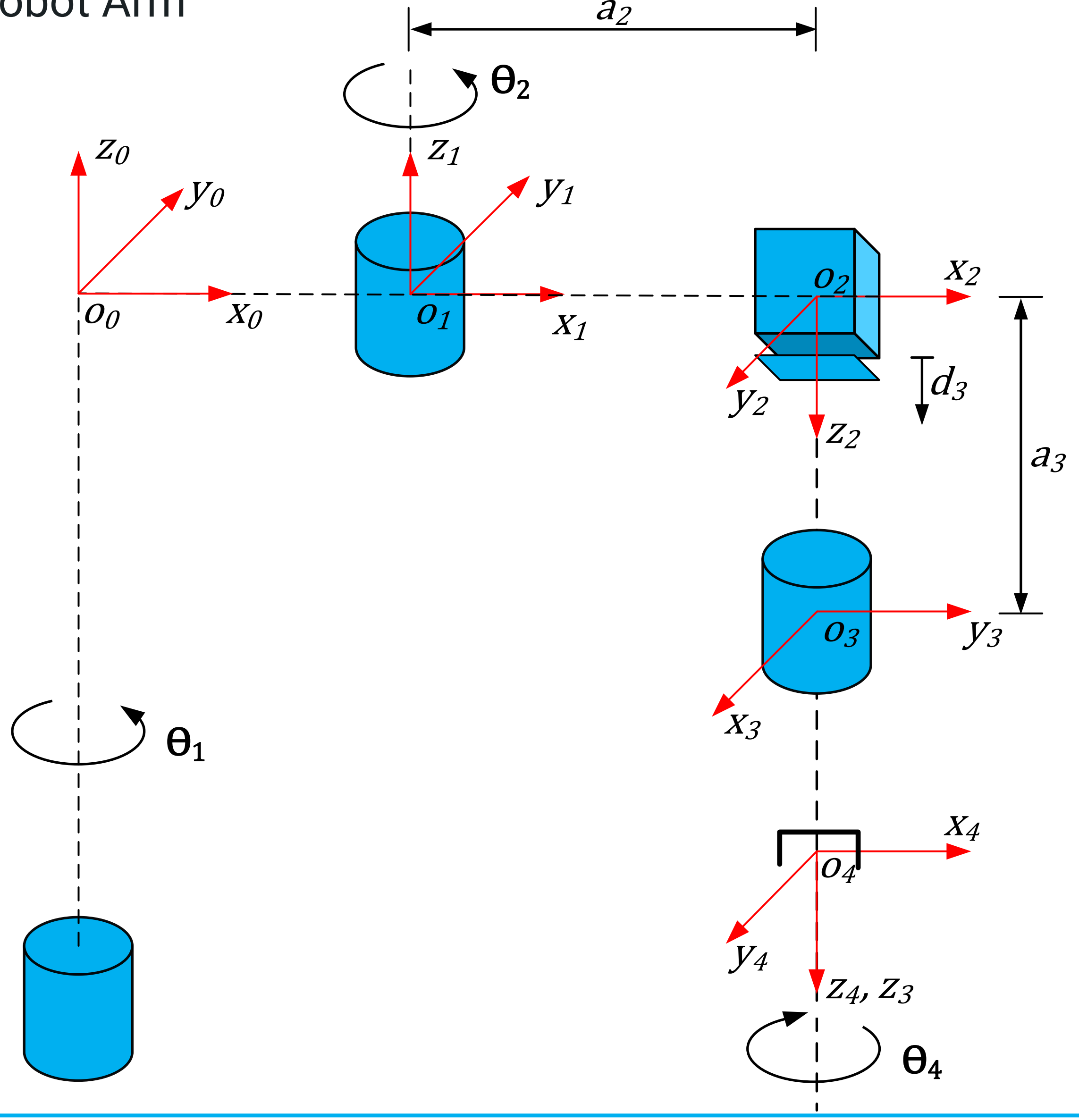
Examples: SCARA

- SCARA: Selective Compliance Assembly Robot Arm
- Generally faster than comparable Cartesian robot systems.
- Advantageous for many types of assembly operations and for transferring parts from one cell to another or for loading or unloading process stations that are enclosed.

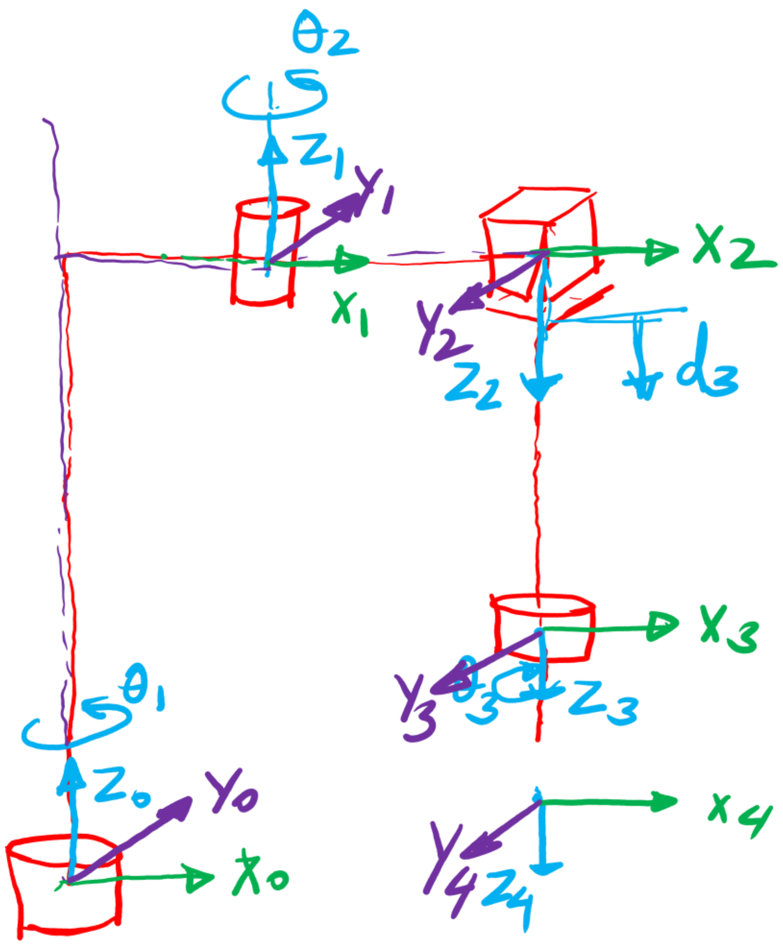


HUMBER

- SCARA: Selective Compliance Assembly Robot Arm



- SCARA: Selective Compliance Assembly Robot Arm



HUMBER

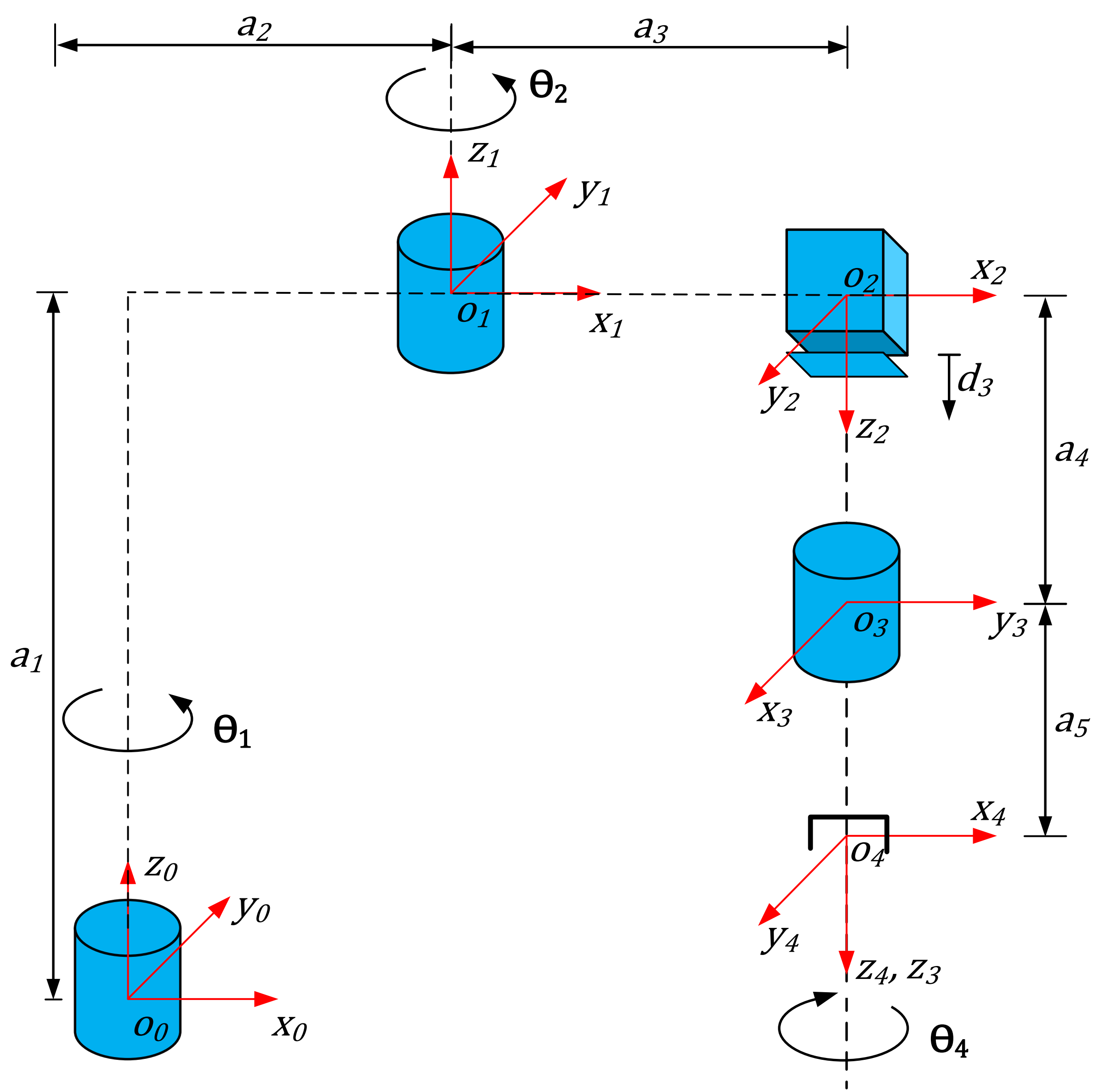
- SCARA: Selective Compliance Assembly Robot Arm

input $(\theta_1, \theta_2, d_3, \theta_4) + P^4$
output P^0

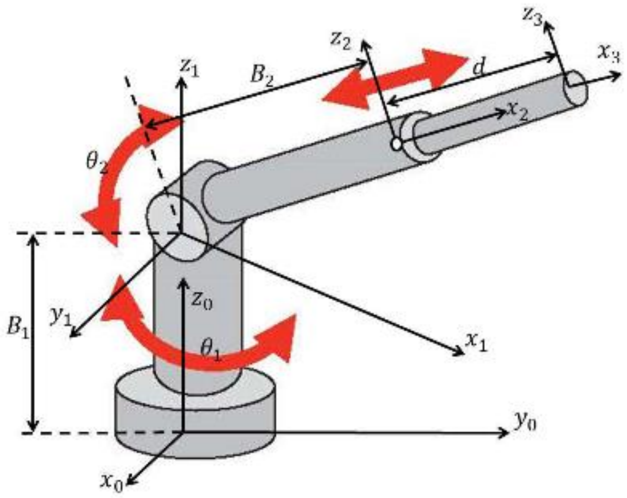
The program is asking for
a_1, a_2, a_3, a_4, a_5

The program takes these as input
$(\theta_1, \theta_2, d_3, \theta_4) + P^4$

The program calculates P^0

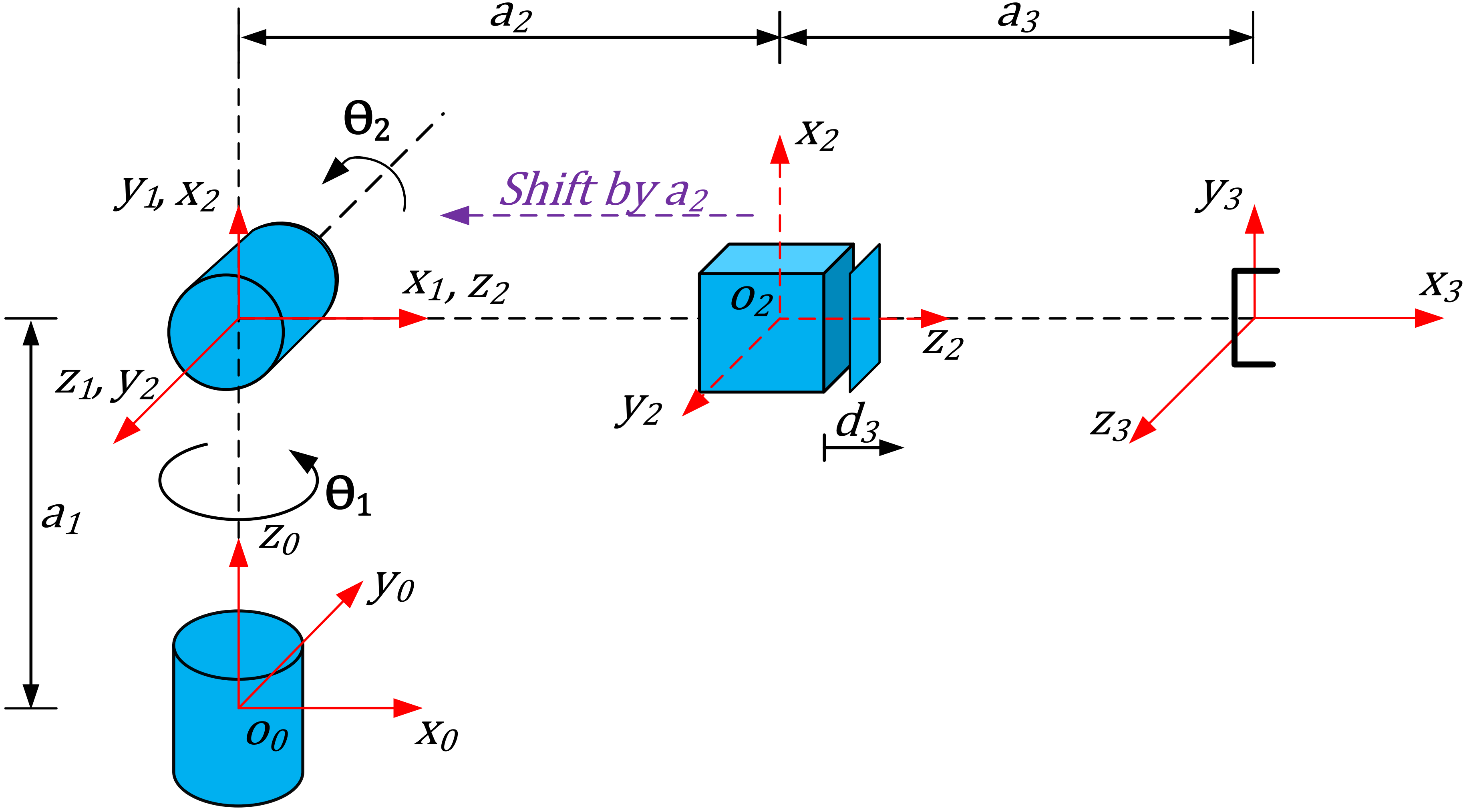


- Spherical

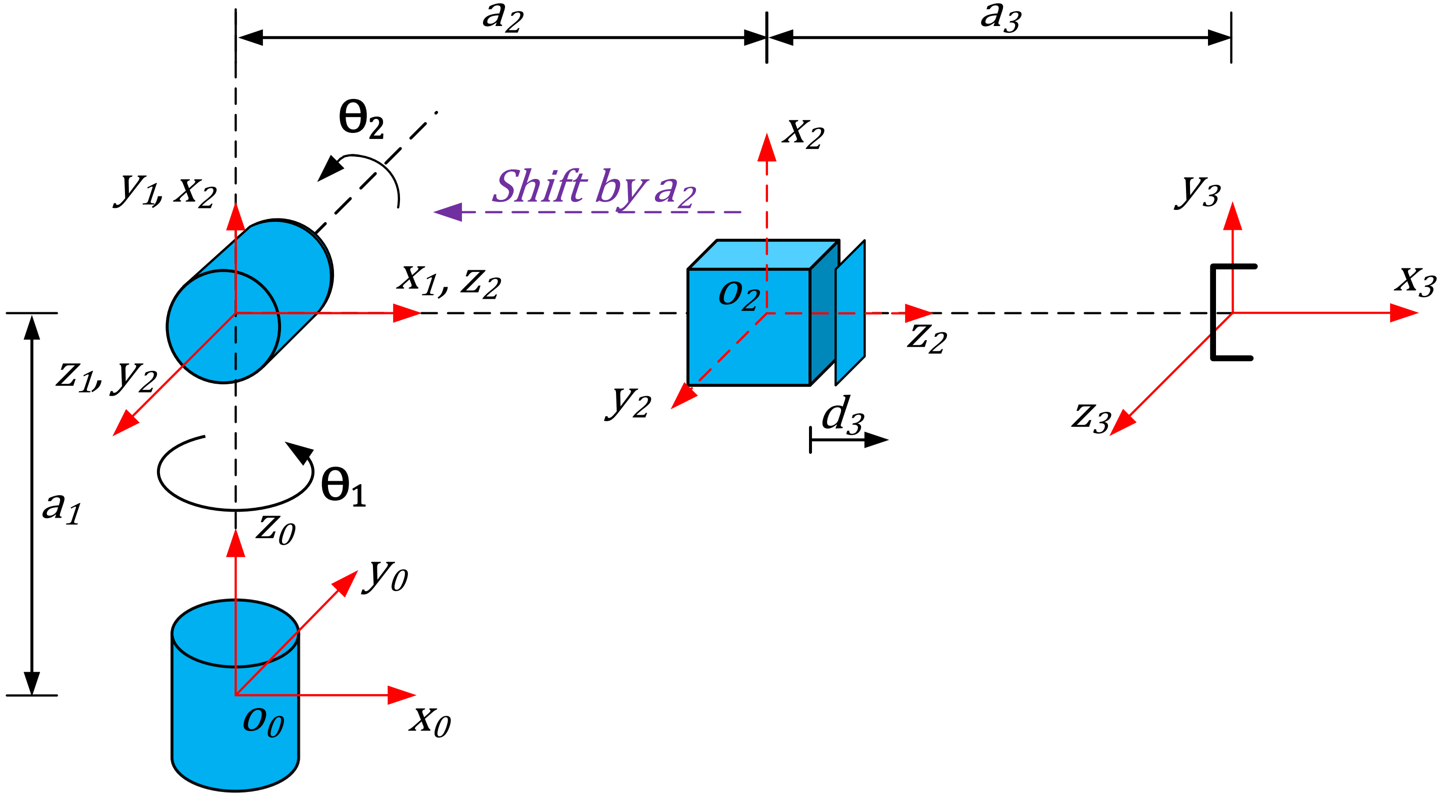


HUMBER

- Spherical:

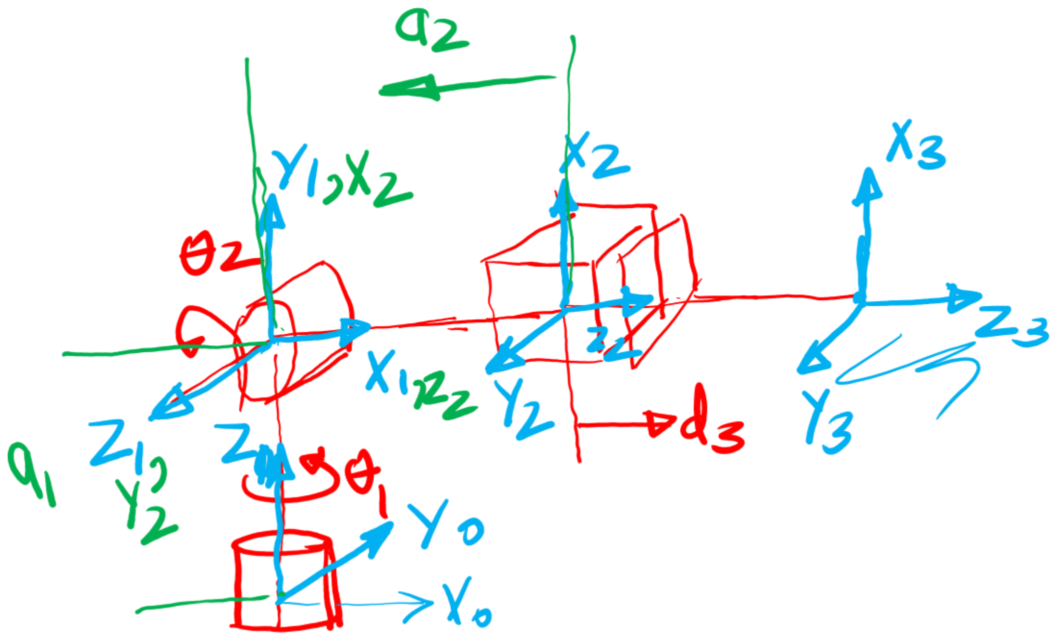


- Spherical:

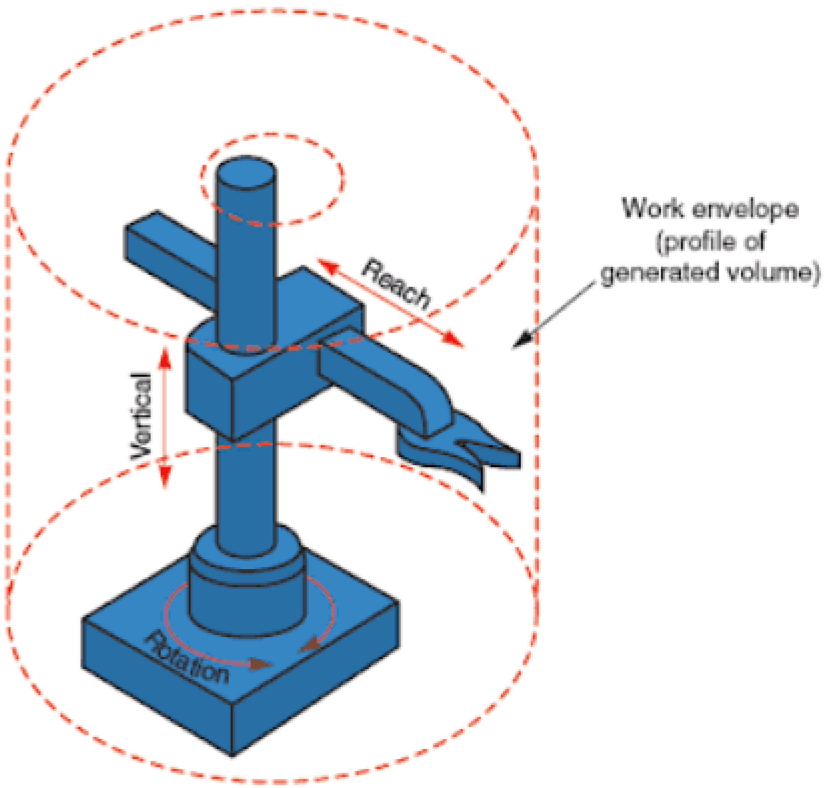

a2
a3
θ2
x2
Shift by a2
y1, x2
y3
x1, z2
o2
x3
z2
z1, y2
y2
d3
z3
a1
θ1
z0
y0
o0
x0

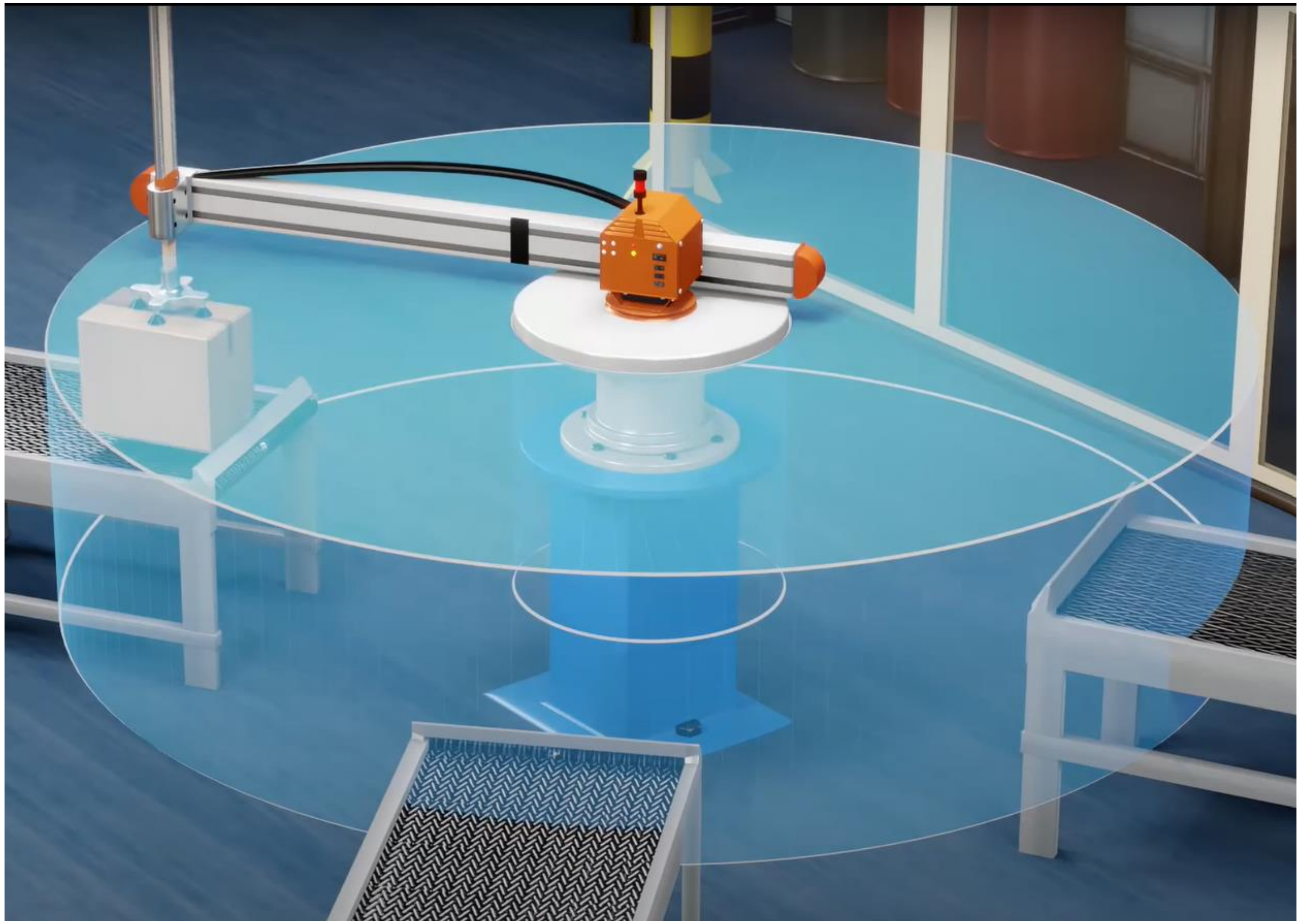
Examples: Spherical

- Spherical:



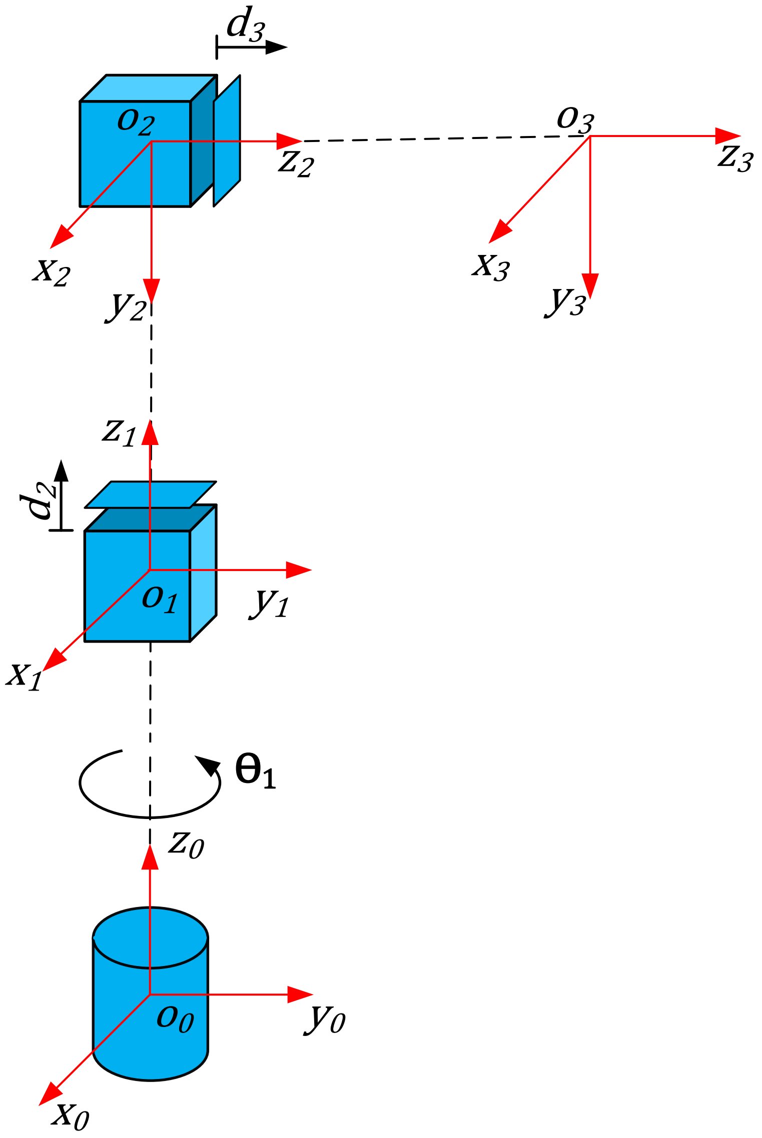
- Cylindrical



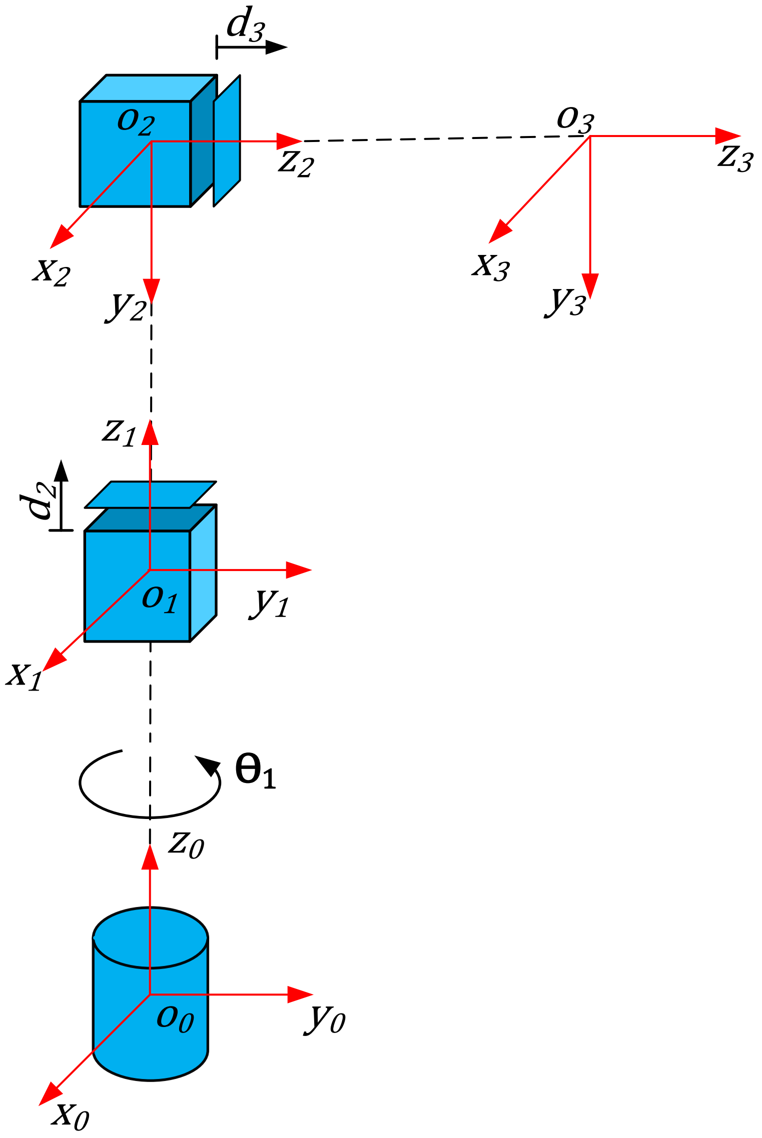


Examples: Three-Link Cylindrical

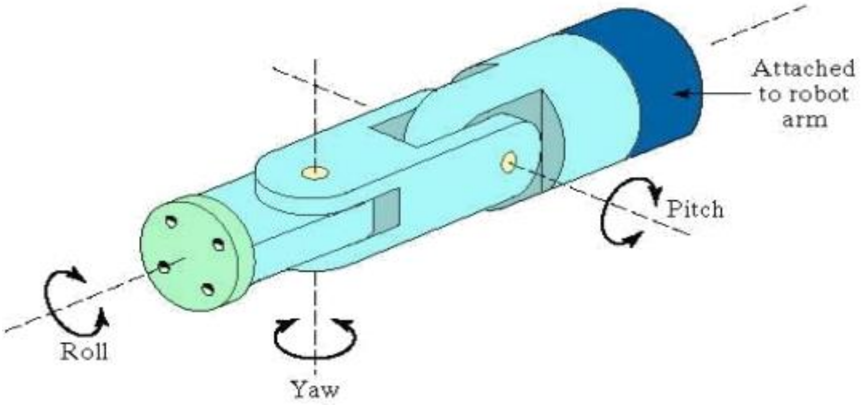
- Cylindrical

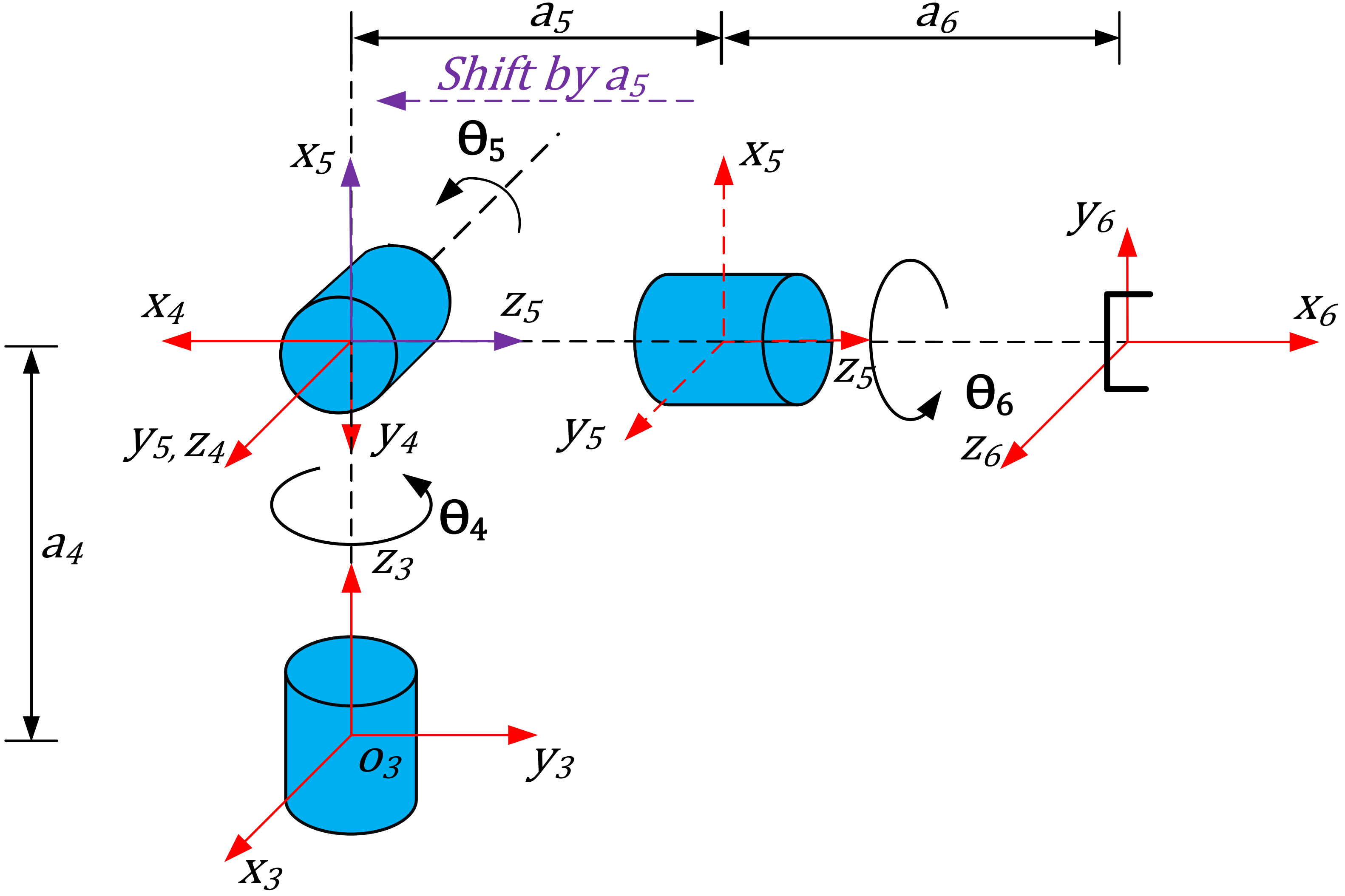


- Cylindrical



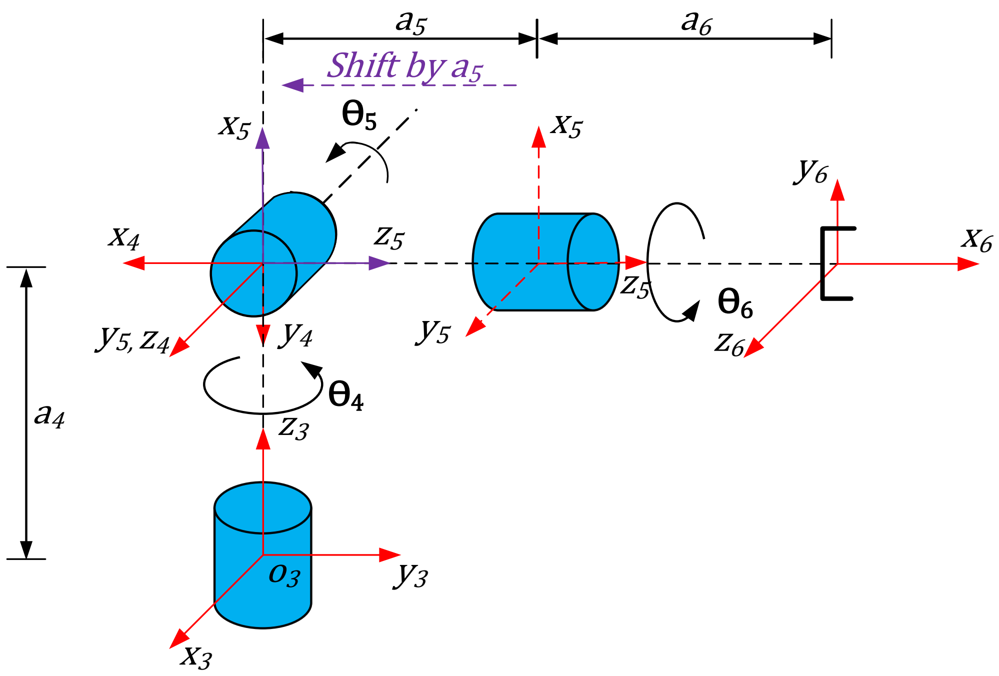
Examples: The Spherical Wrist

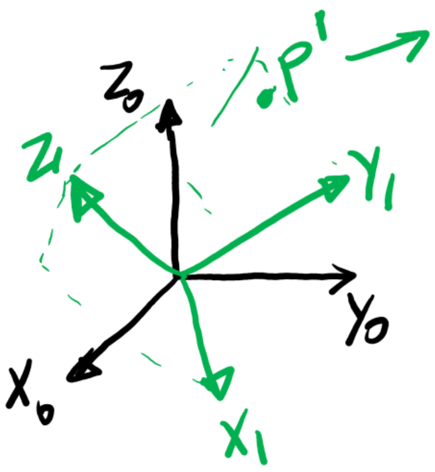




a5
a6
Shift by a5
θ5
x5
x5
y6
x4
z5
x6
z5
θ6
y5, z4
y4
y5
z6
θ4
a4
z3
o3
y3
x3

Examples: The Spherical Wrist



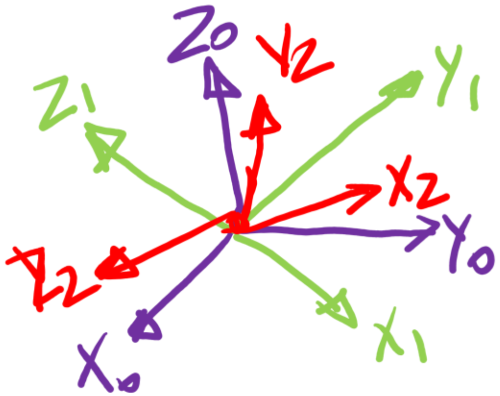


coordinate of P wrt 1 coordinate

$$P^0 = R_1^0 P^1$$

wrt Ø

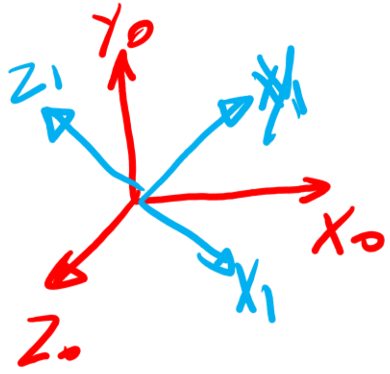
$$R_1^0 = \begin{array}{c|ccc} & X_1 & Y_1 & Z_1 \\ \hline X_0 & \square & \square & \square \\ Y_0 & \square & \square & \square \\ Z_0 & \square & \square & \square \end{array} = \begin{bmatrix} X_1 \cdot X_0 & Y_1 \cdot X_0 & Z_1 \cdot X_0 \\ X_1 \cdot Y_0 & Y_1 \cdot Y_0 & Z_1 \cdot Y_0 \\ X_1 \cdot Z_0 & Y_1 \cdot Z_0 & Z_1 \cdot Z_0 \end{bmatrix}$$

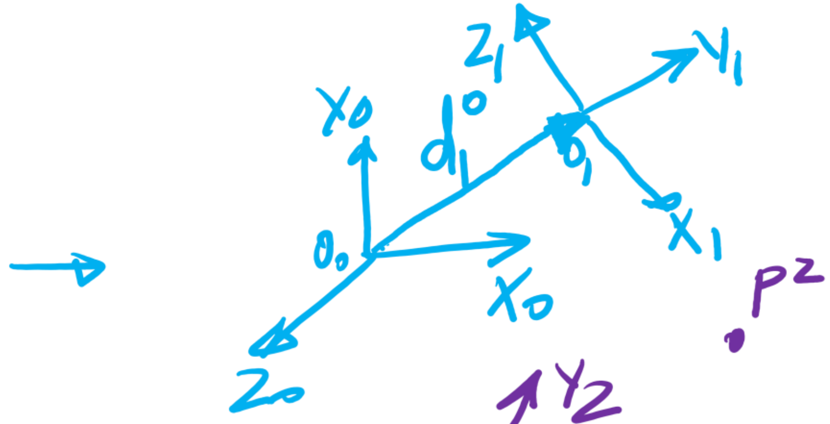


$$P^2$$

$$R_2^0 = R_1^0 R_2^1$$

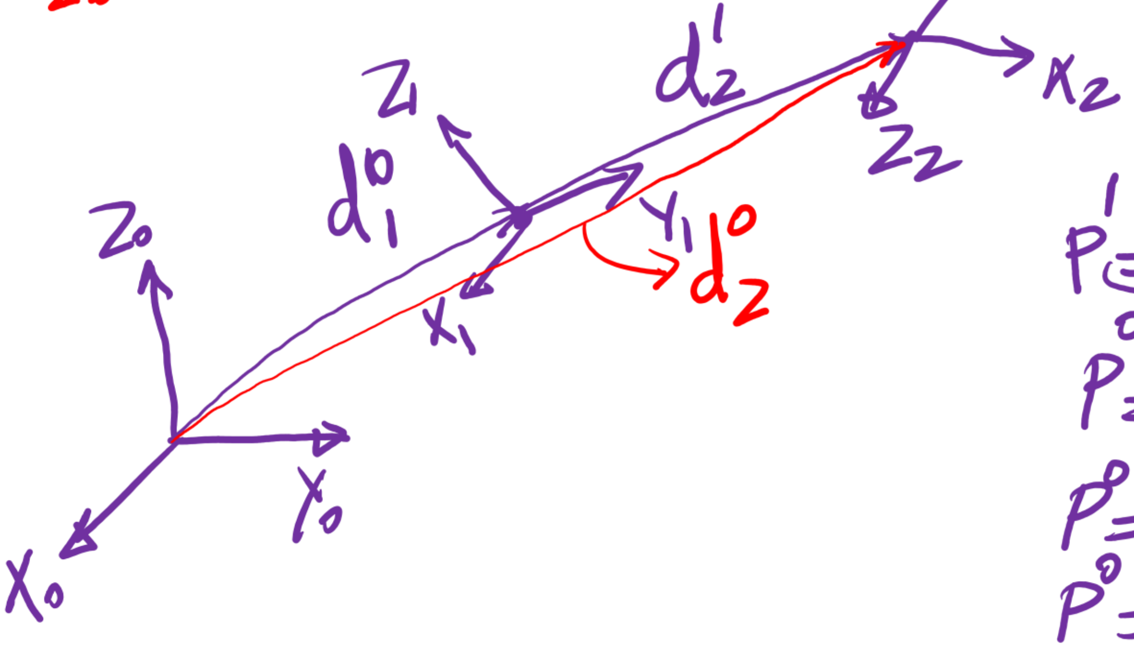
$$P^0 = R_2^0 P^2 = R_1^0 R_2^1 P^2$$





$$P^0 = \underbrace{R_1^0}\, P^1 + \underbrace{d_1^0}$$

$$P^0 = \underbrace{\begin{bmatrix} \square & \square & \square \\ \square & \square & \square \\ \square & \square & \square \end{bmatrix}}_{3\times 3} \begin{bmatrix} P_{x_1} \\ P_{y_1} \\ P_{z_1} \end{bmatrix} + \begin{bmatrix} O_{1,x} \\ O_{1,y} \\ O_{1,z} \end{bmatrix}$$



$$P^1 = R_2^1 P^2 + d_2^1$$

$$P^0 = R_1^0 P^1 + d_1^0$$

$$P^0 = R_1^0 (R_2^1 P^2 + d_2^1) + d_1^0$$

$$P^0 = \underbrace{R_1^0 R_2^1}_{R_2^0} P^2 + \underbrace{R_1^0 d_2^1 + d_1^0}_{d_2^0}$$

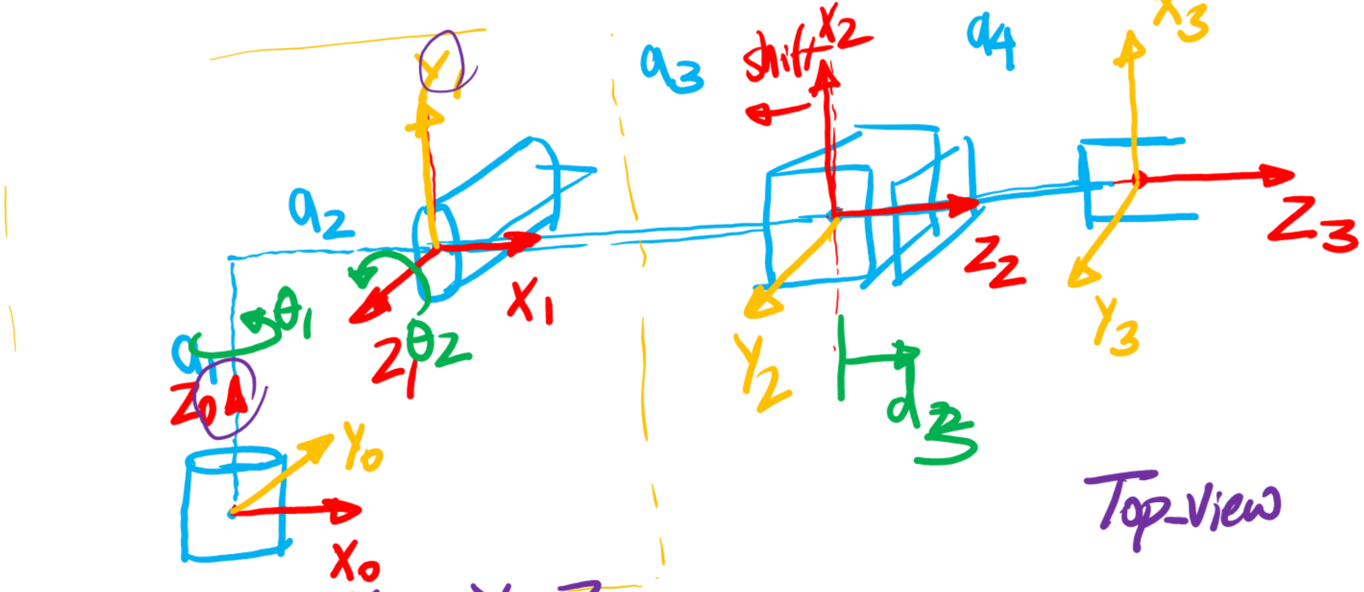
$$\begin{cases} R_2^0 = R_1^0 R_2^1 \\ d_2^0 = R_1^0 d_2^1 + d_1^0 \end{cases}$$

$$H_2 = \begin{bmatrix} R & d \\ 0 & 1 \end{bmatrix}$$

$$H_2^0 = H_1^0 H_2^1 = \begin{bmatrix} R_1^0 & d_1^0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_2^1 & d_2^1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \overbrace{R_1^0 R_2^1}^{R_2^0} & \overbrace{R_1^0 d_2^1 + d_0^1}^{d_2^0} \\ 0 & 1 \end{bmatrix}$$

$$H_2^0 = \begin{bmatrix} R_2^0 & d_2^0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_1^0 R_2^1 & \underline{R_1^0 d_2^1 + d_0^1} \\ 0 & 1 \end{bmatrix}$$

$$P = \begin{bmatrix} P_x \\ P_y \\ P_z \\ 1 \end{bmatrix}$$

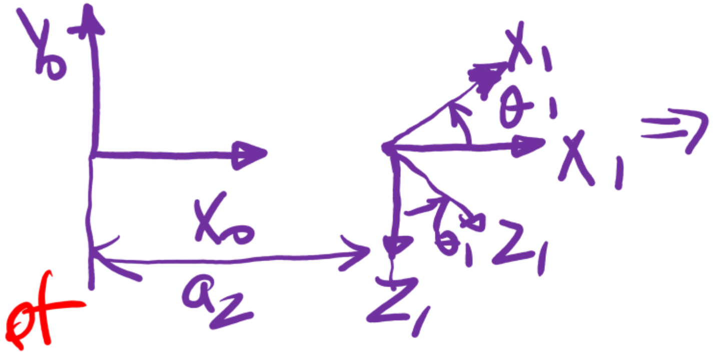


$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix}$$

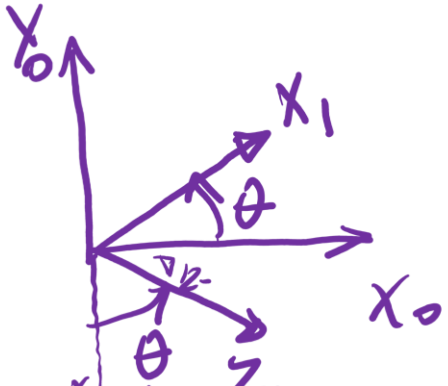
$$= \begin{bmatrix} C\theta & 0 & S\theta \\ S\theta & 0 & -C\theta \\ 0 & 1 & 0 \end{bmatrix} \times$$

$$R_1^0 = \underbrace{\begin{array}{c|ccc} & X_1 & Y_1 & Z_1 \\ \hline X_0 & 1 & 0 & 0 \\ Y_0 & 0 & 0 & -1 \\ Z_0 & 0 & 1 & 0 \end{array}}_{\text{Rotation because of frame movement}} \quad \underbrace{R_{y_1,\theta_1}}_{\text{Rotation because of joint variable}}$$

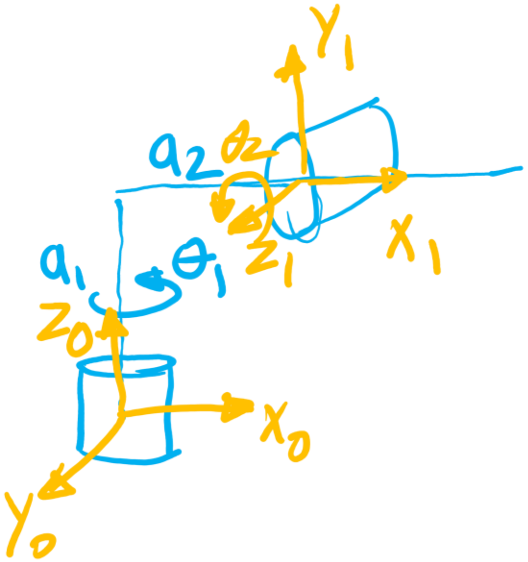
Top-view



$\Rightarrow$



$$\begin{array}{c|ccc} & X_1 & Y_1 & Z_1 \\ \hline X_0 & \cos\theta & 0 & S\theta \\ Y_0 & S\theta & 0 & -C\theta \\ Z_0 & 0 & 1 & 0 \end{array} \times$$



$R_1^0 =$

First: Find R because of frame movement

$$\begin{array}{c|ccc} & X_1 & Y_1 & Z_1 \\ \hline X_0 & 1 & 0 & 0 \\ Y_0 & 0 & 0 & -1 \\ Z_0 & 0 & 1 & 0 \end{array}$$

Second: Find R because of θ_1

θ_1: is Rotation of about Z_0 (fixed frame) Previous frame

θ_1: is " " " Y_1 (current frame) New frame

$$R_1^0 = R_{z_0,\theta_1} \begin{bmatrix} & & \\ & & \end{bmatrix}$$

$$R_1^0 = \begin{bmatrix} & & \\ & & \end{bmatrix} R_{y_1,\theta_1}$$

$$H_1^0 = \begin{bmatrix} R_1^0 & d_1^0 \\ 0 & 1 \end{bmatrix} \Rightarrow H_1^0 = \begin{bmatrix} R_1^0 & \begin{matrix} a_2 C\theta_1 \\ a_2 S\theta_1 \\ a_1 \end{matrix} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

$$R_1^0 = \begin{array}{c|c} & X_1 \quad Y_1 \quad Z_1 \\ \begin{matrix} X_0 \\ Y_0 \\ Z_0 \end{matrix} & \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix} \end{array}$$

$$d_1^0 = \begin{bmatrix} a_2 \\ 0 \\ a_1 \end{bmatrix}$$

$$R_{Z_0,\theta_1} d_1^0 = \begin{bmatrix} C\theta_1 & -S\theta_1 & 0 \\ S\theta_1 & C\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_2 \\ 0 \\ a_1 \end{bmatrix}$$

$$R_0^1 = R_{Z_0,\theta_1} \begin{bmatrix} \\ \\ \end{bmatrix}$$

$$d_1^0 = R_{Z_0,\theta_1} d_1^0 = \begin{bmatrix} a_2 C\theta_1 \\ a_2 S\theta_1 \\ a_1 \end{bmatrix}$$

$$R_0^1 = \begin{bmatrix} \\ \\ \end{bmatrix} R_{Y_1,\theta_1}$$

$$H^0_{n=4} = H^0_1 \, H^1_2 \, H^2_3 \, H^3_4$$

$$P^0 = H^0_4 \, P^4$$

$$H^0_1 = \begin{bmatrix} R^0_1 & d^0_1 \\ 0 & 1 \end{bmatrix}$$

$$R^0_1 = R_{fixed,\theta_1} \begin{array}{c c} & \begin{array}{ccc} x_1 & y_1 & z_1 \end{array} \\ \begin{array}{c} x_0 \\ y_0 \\ z_0 \end{array} & \begin{bmatrix} \; & \; & \; \\ \; & \; & \; \\ \; & \; & \; \end{bmatrix} \end{array}$$

$$d^0_1 = R_{fixed,\theta_1} \, d \qquad d = \begin{bmatrix} \vdots \\ \vdots \end{bmatrix}$$