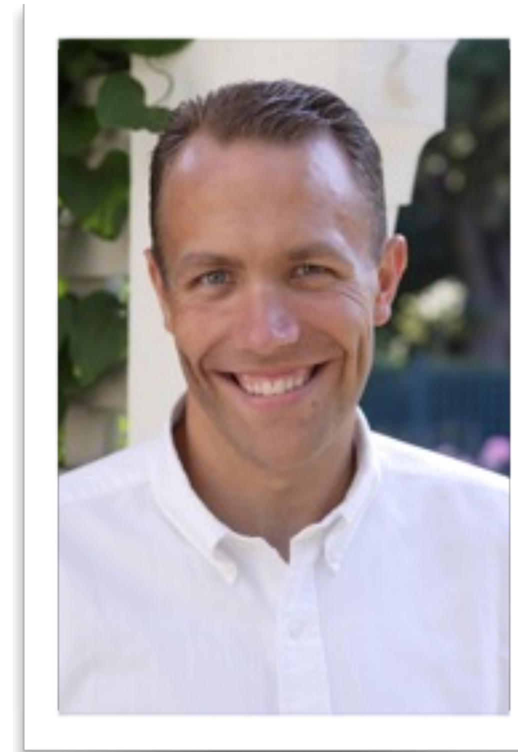
MEAM 520

More Denavit-Hartenberg (DH)

Katherine J. Kuchenbecker, Ph.D.
General Robotics, Automation, Sensing, and Perception Lab (GRASP)
MEAM Department, SEAS, University of Pennsylvania

Lecture 6: September 25, 2012



Slides created by
Jonathan Fiene

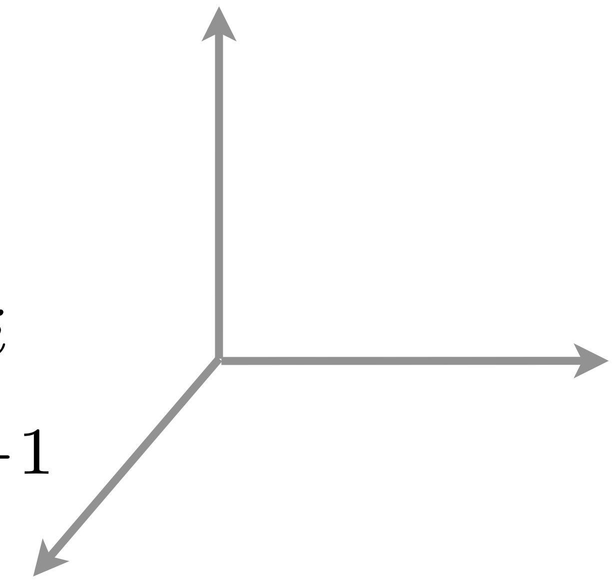
Denavit-Hartenberg Parameters

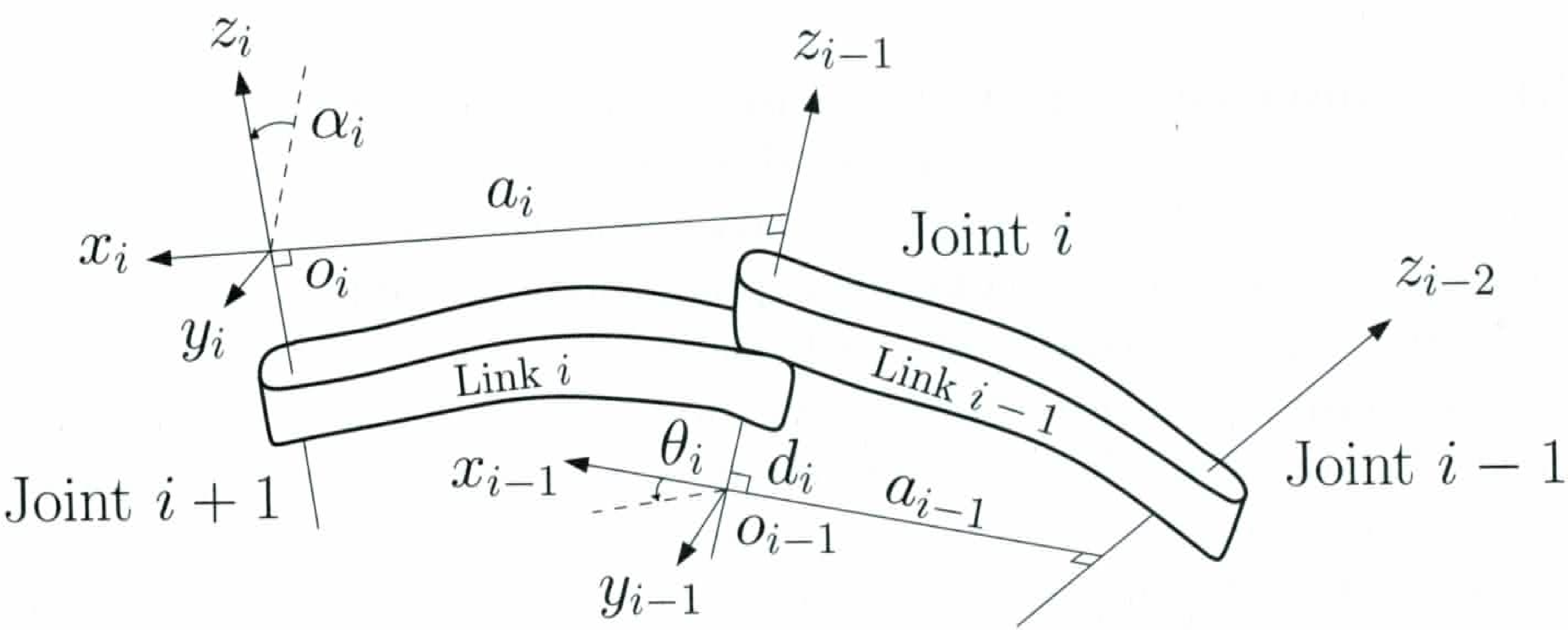
The **Denavit-Hartenberg convention** defines four parameters and some rules to help characterize arbitrary kinematic chains

start by attaching a frame to each link:

the joint variable for joint i+1 acts along/around z_i

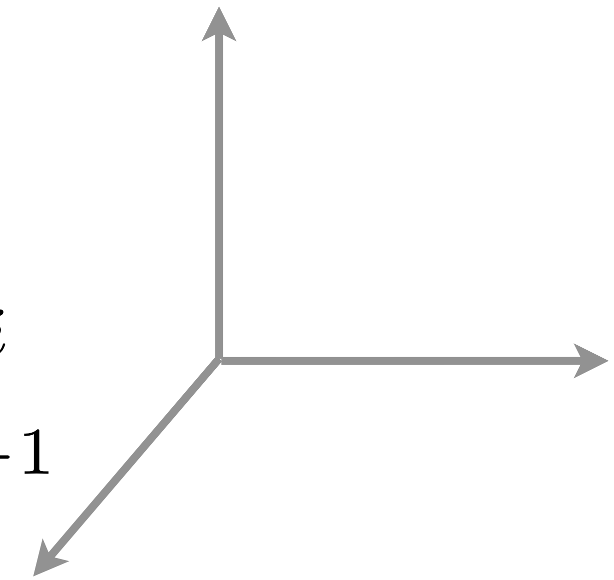
the axis x_i is perpendicular to, and intersects, z_{i-1}





Denavit & Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," ASME Journal of Applied Mechanics, June 1955

The **Denavit-Hartenberg convention** defines four parameters and some rules to help characterize arbitrary kinematic chains



start by attaching a frame to each link:

the joint variable for joint i+1 acts along/around z_i

the axis x_i is perpendicular to, and intersects, z_{i-1}

the following conventions make this process easier (p. 82 in SHV):

if z_{i-1} is parallel to z_i	orient x_i away from z_{i-1}
if z_{i-1} intersects z_i	orient x_i normal to the plane formed by z_{i-1} and z_i
if z_{i-1} is not coplanar with z_i	orient x_i along normal with z_{i-1}

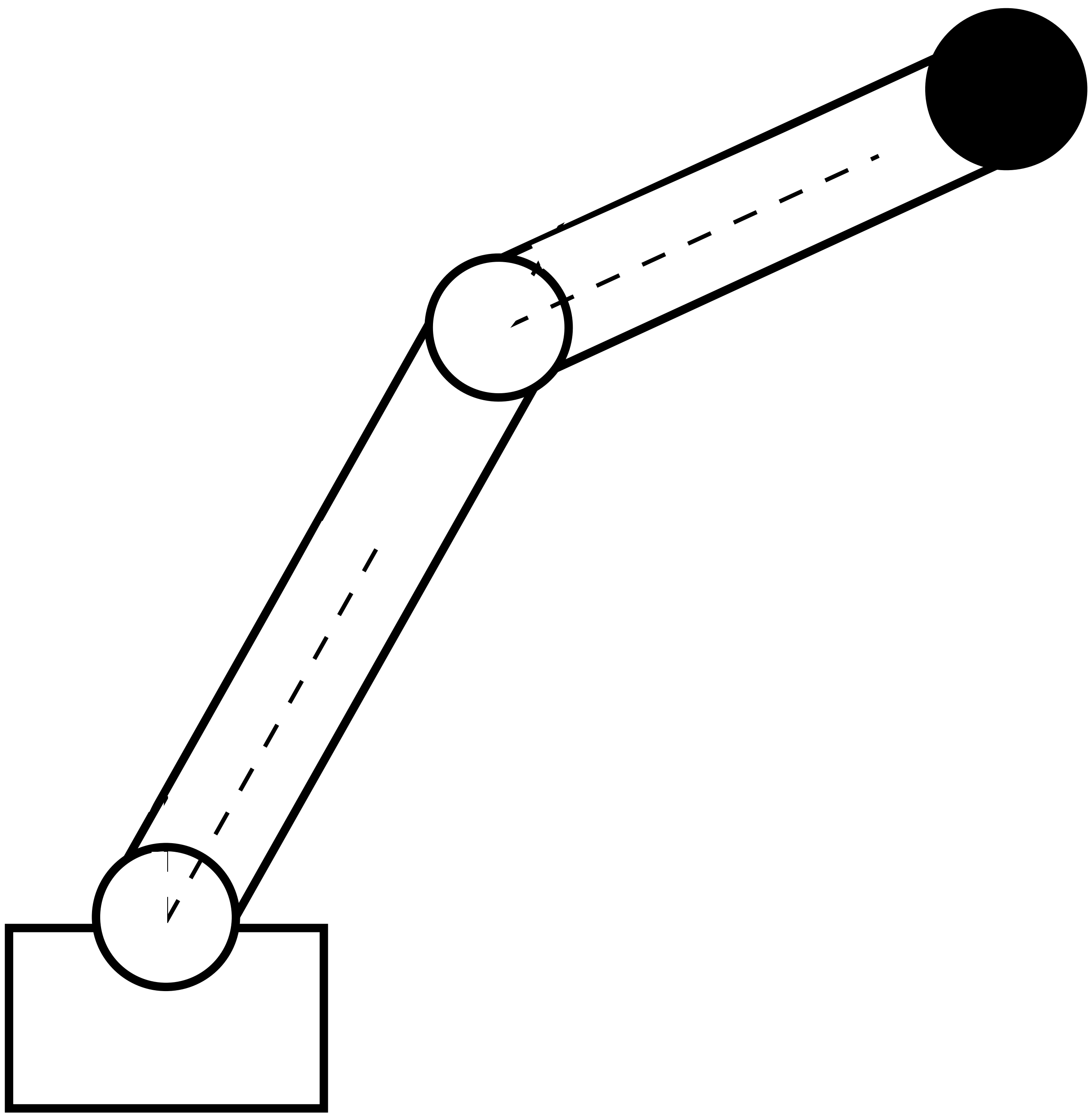
Denavit & Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," ASME Journal of Applied Mechanics, June 1955

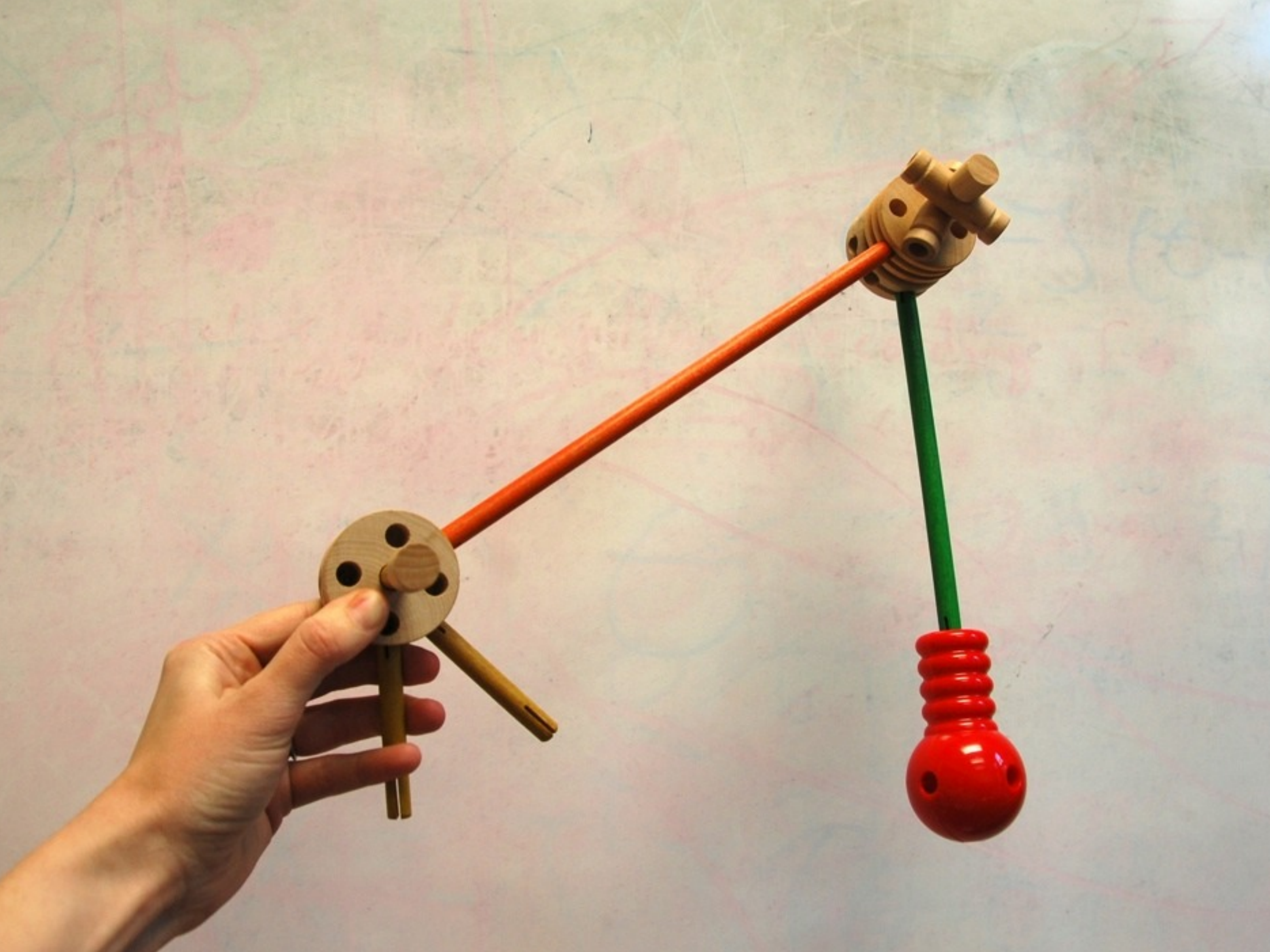
The **Denavit-Hartenberg convention** defines four parameters and some rules to help characterize arbitrary kinematic chains

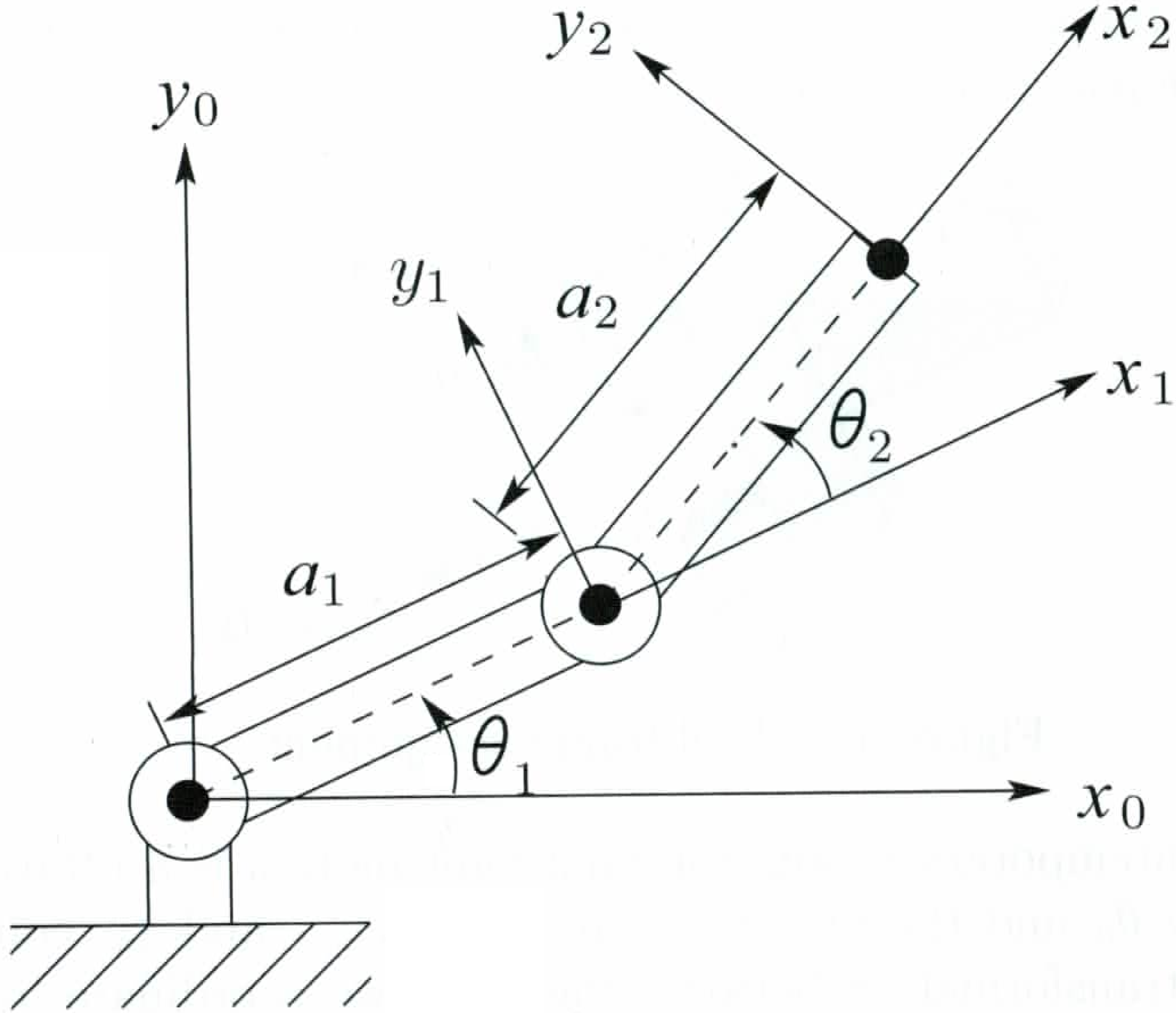
a_i **Link Length**	the perpendicular distance between z_i and z_{i-1}, measured along x_i
α_i **Link Twist**	the angle between z_{i-1} and z_i, measured in the plane normal to x_i (right-hand rule around x_i)
d_i **Link Offset**	the distance along z_{i-1} from o_{i-1} to the intersection with x_i
θ_i **Joint Angle**	the angle between x_{i-1} and x_i, measured in the plane normal to z_{i-1} (right-hand rule around z_{i-1})

Denavit & Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," ASME Journal of Applied Mechanics, June 1955

Example 1: Planar RR Robot







y_2
x_2
y_0
y_1
a_2
x_1
θ_2
a_1
θ_1
x_0

The **Denavit-Hartenberg transform** results from successive rotations and translations via the four DH parameters

The transform from i-1 to i:

$$A_i = \mathrm{Rot}_{z,\theta_i}\ \mathrm{Trans}_{z,d_i}\ \mathrm{Trans}_{x,a_i}\ \mathrm{Rot}_{x,\alpha_i}$$

$$= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$
\begin{aligned}
A_i &= \mathrm{Rot}_{z,\theta_i}\mathrm{Trans}_{z,d_i}\mathrm{Trans}_{x,a_i}\mathrm{Rot}_{x,\alpha_i} \\
&= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&\quad \times \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1^*
2	a_2	0	0	θ_2^*

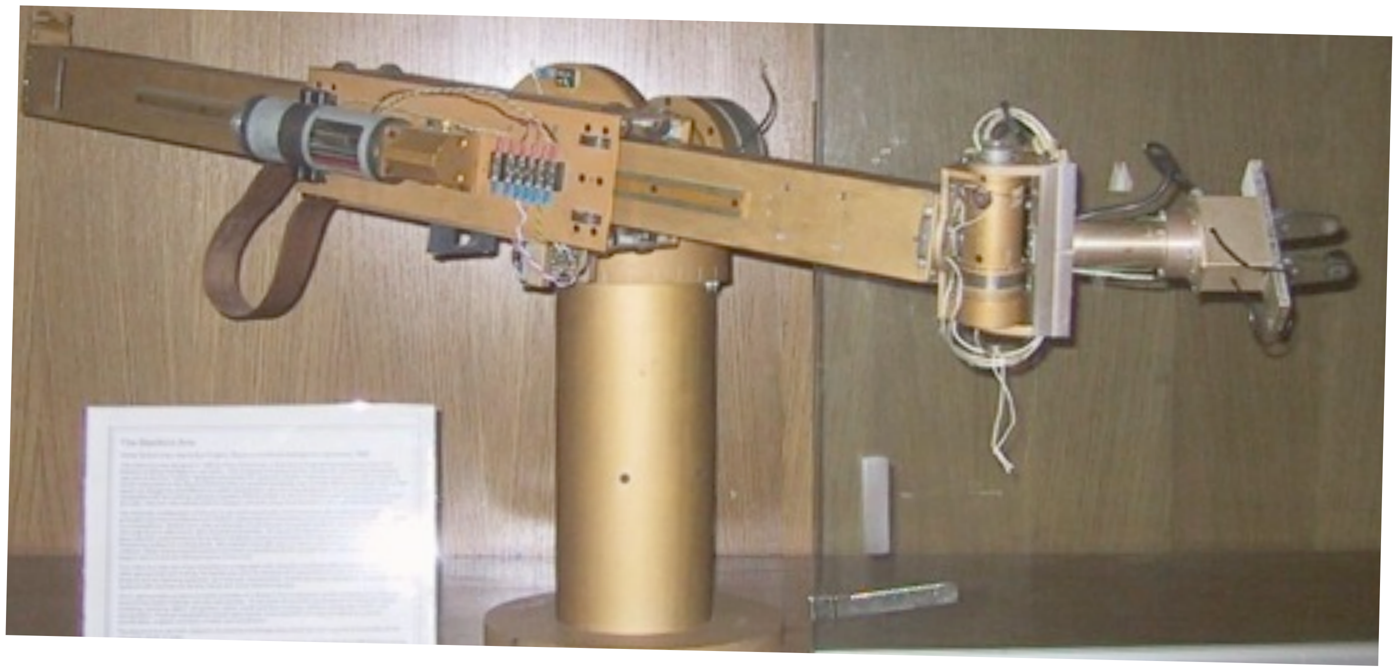
$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1c_1 \\ s_1 & c_1 & 0 & a_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

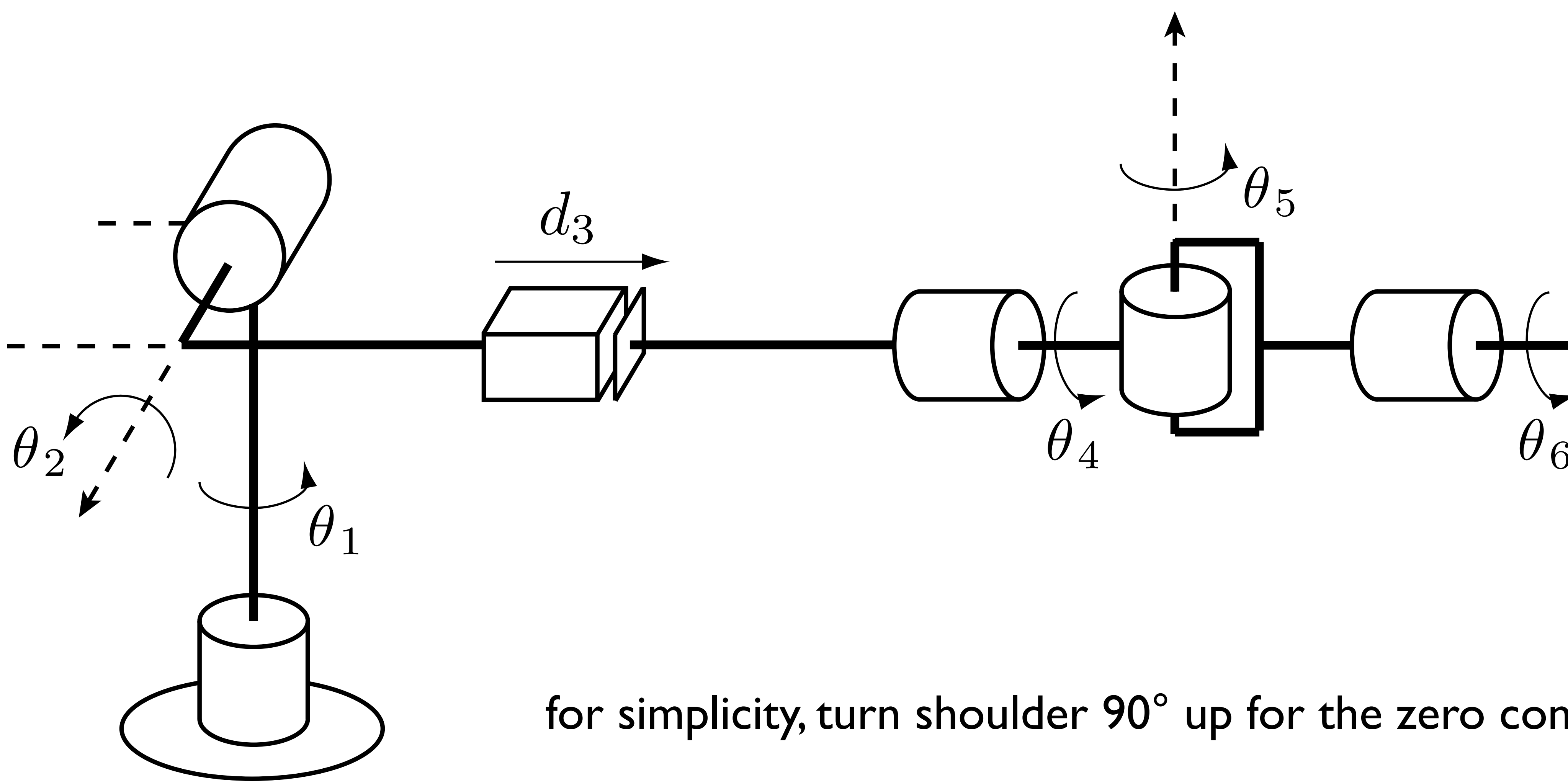
$$T_1^0 = A_1$$

$$T_2^0 = A_1A_2 = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1c_1 + a_2c_{12} \\ s_{12} & c_{12} & 0 & a_1s_1 + a_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Questions ❓

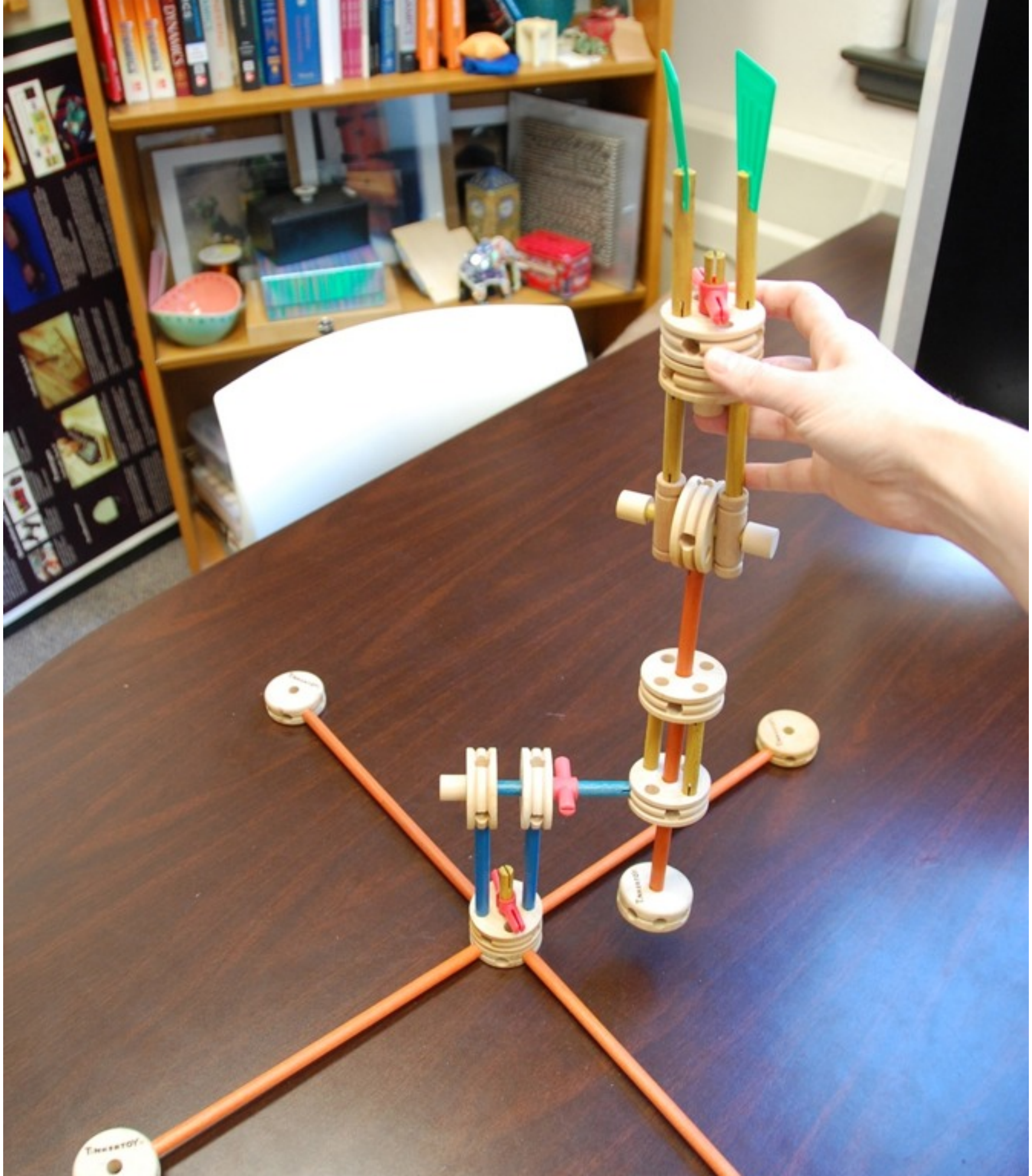
Example 2: The Stanford Manipulator (RRPRRR)

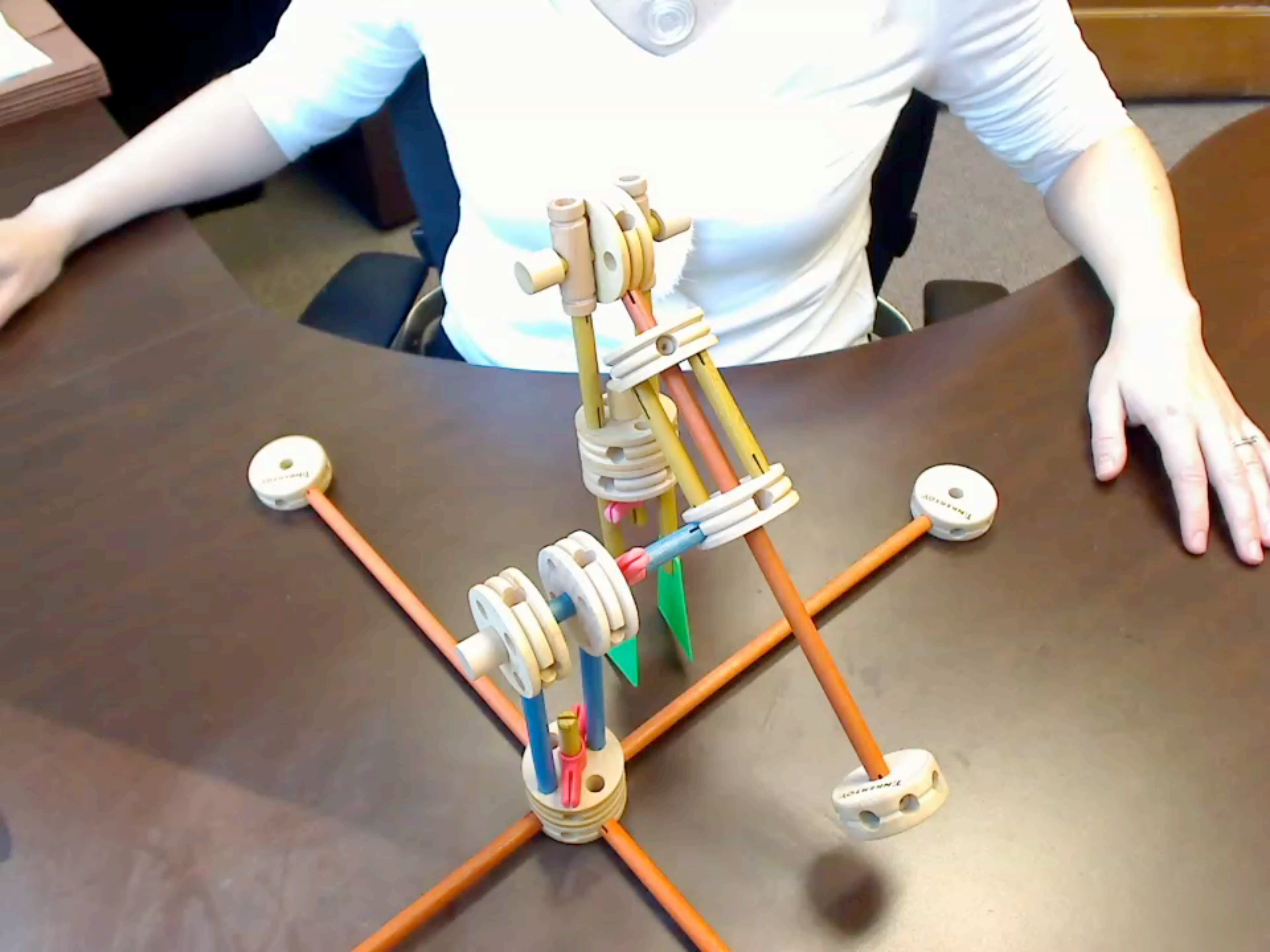


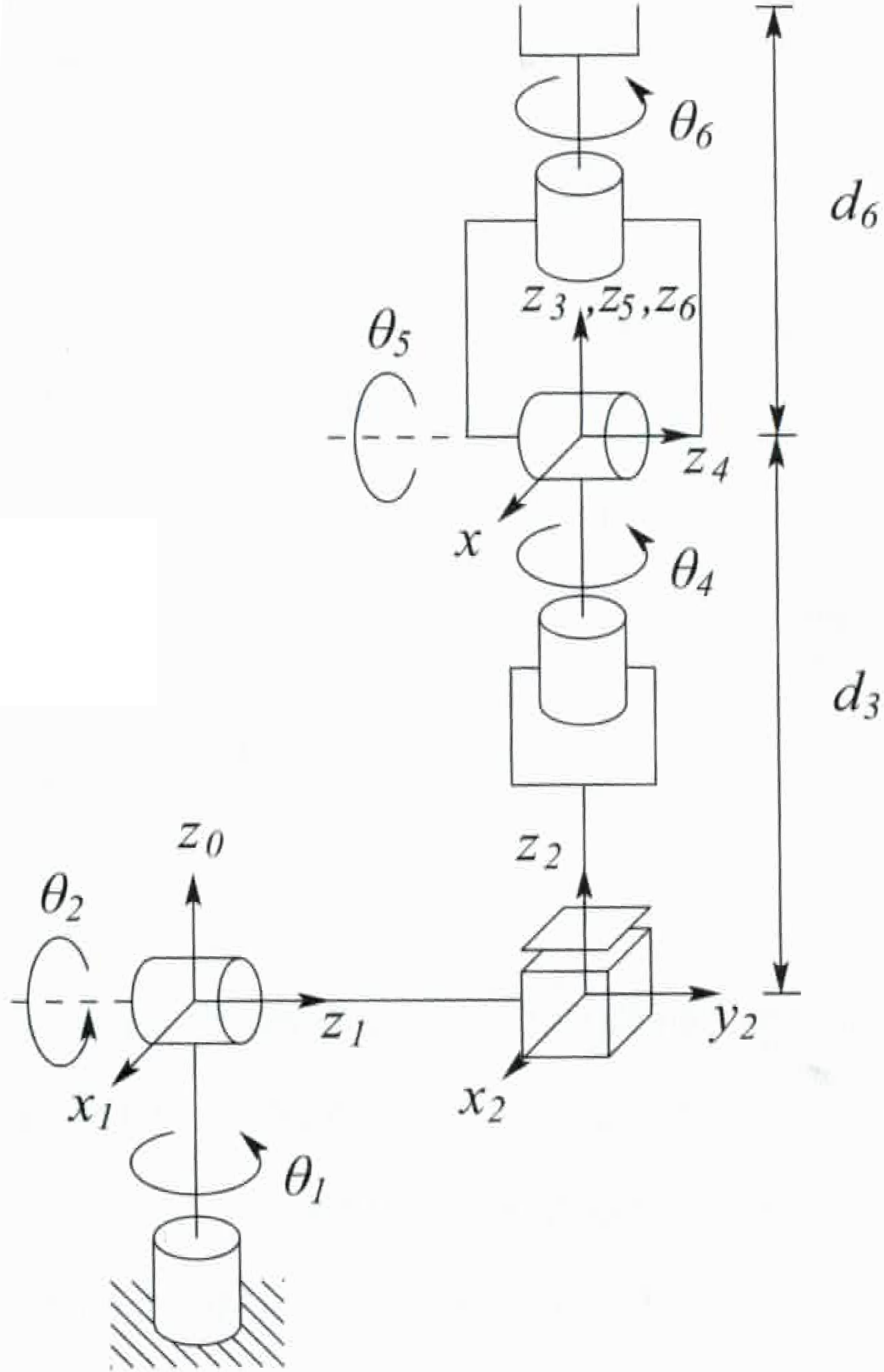


for simplicity, turn shoulder 90° up for the zero configuration


Instant Insanity – Computer Vision & Robotics – YouTube
http://www.youtube.com/watch?v=O1oJzUSITeY
Google
YouTube
Browse
Movies
Upload
kathjulk
Instant Insanity - Computer Vision & Robotics
ComputerHistory
Subscribe
176 videos
0:49 / 6:26
Like
Add to
Share
8,216
Uploaded by ComputerHistory on Apr 28, 2009
[Recorded: 1971]
Over the last decades, computer vision systems have become increasingly capable of controlling robotic movement. One example of early research and development of computer vision and robotic systems was recorded at the
49 likes, 3 dislikes
Robot Violinist
by diagonaluk
6,529,070 views
FEATURED
1:40
How Computers Work - Journey Into The Walk-
by ComputerHistory
126,480 views
26:31
UNIVAC - Then and Now
by ComputerHistory
44,455 views
13:17
Cray-1 Supercomputer 30th Anniversary
by ComputerHistory
21,880 views
1:42:22
Secret History of Silicon Valley
by ComputerHistory
134,089 views
1:02:45
Computer Pioneers - Pioneer Computers
by ComputerHistory
79,207 views
53:26
Computer Pioneers - Pioneer Computers
Go to "http://www.youtube.com/watch?v=EzjkBwZtxp4&feature=fvwrel"


DYNAMICS




θ6
d6
z3 ,z5 ,z6
θ5
z4
x
θ4
d3
z0
z2
θ2
z1
y2
x1
x2
θ1

Link	d_i	a_i	α_i	θ_i
1	0	0	-90	$\theta_1^\star$
2	d_2	0	$+90$	$\theta_2^\star$
3	$d_3^\star$	0	0	0
4	0	0	-90	$\theta_4^\star$
5	0	0	$+90$	$\theta_5^\star$
6	d_6	0	0	$\theta_6^\star$

$$T_6^0 = A_1 \cdots A_6 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

page 91 in SHV

$$A_1 = \begin{bmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_2 = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_4 = \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

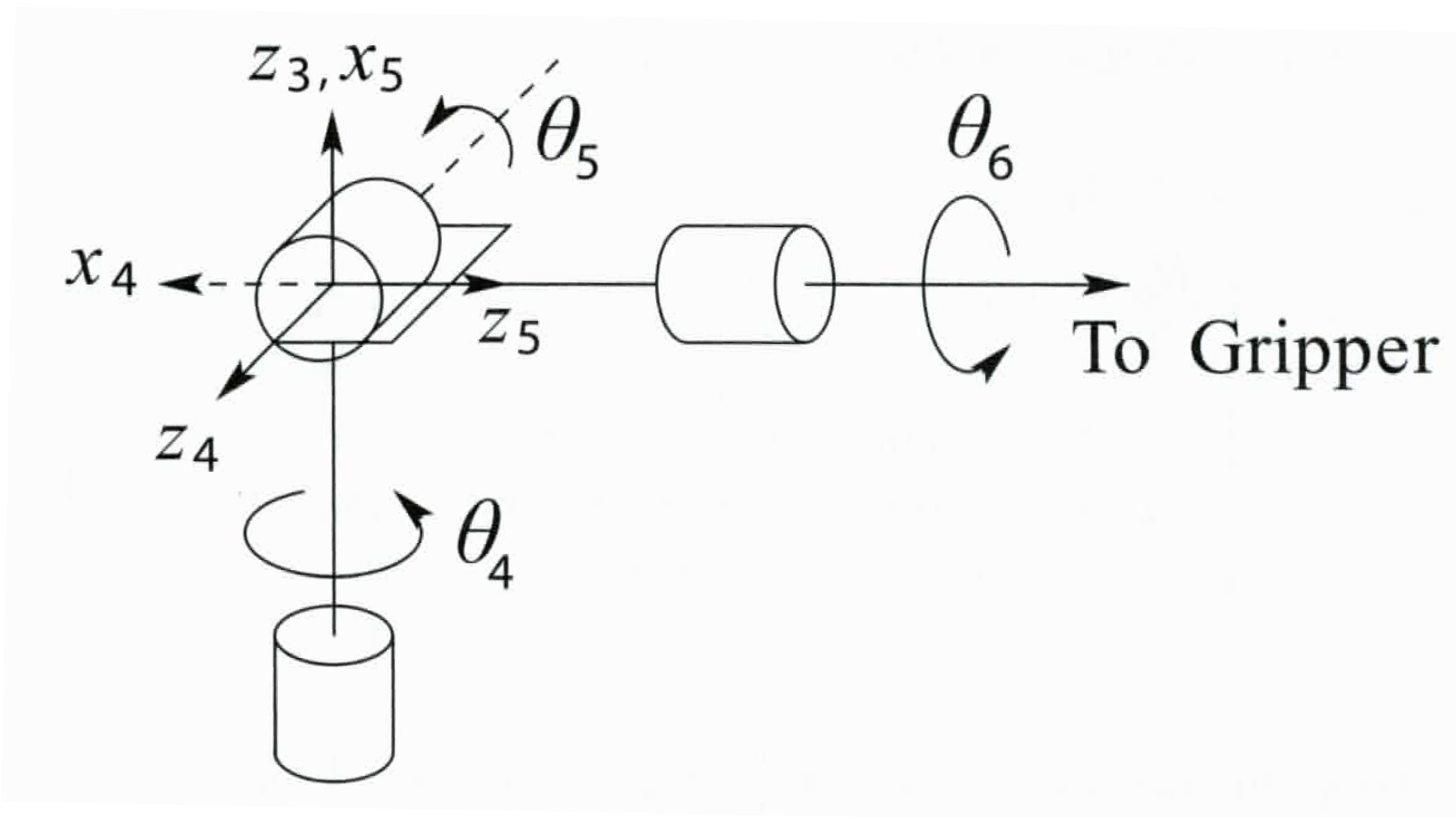
$$A_5 = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

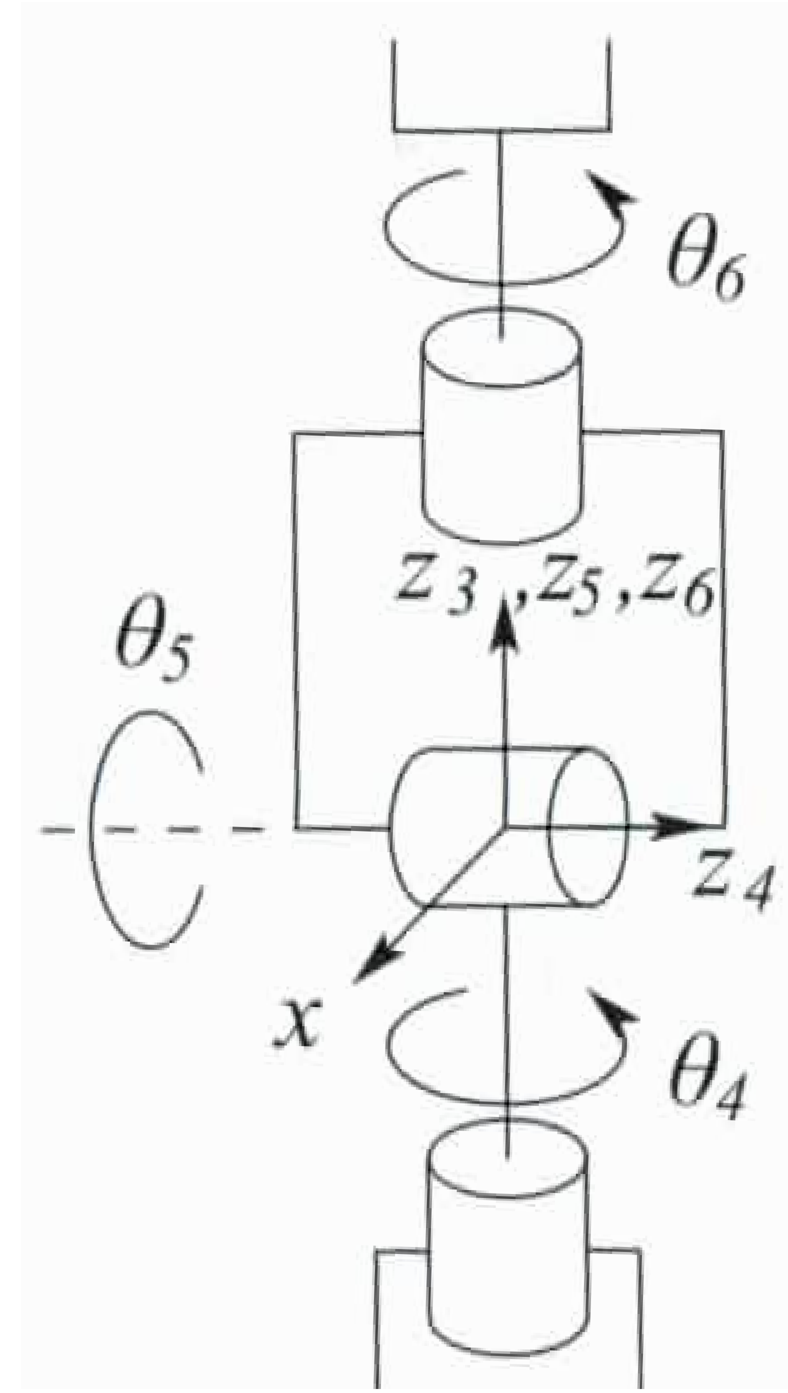
More examples in the book:

Three-Link Cylindrical Robot

Spherical Wrist

SCARA Manipulator





Note: Spherical wrist in Figure 3.8 is drawn with θ_5 = -90°

Questions ?

Homework 2 due Thursday by 5:00 p.m.

Homework 2: Manipulator Kinematics and DH Parameters

MEAM 520, University of Pennsylvania
Katherine J. Kuchenbecker, Ph.D.

September 18, 2012

This assignment is due on **Thursday, September 27 (updated)**, by 5:00 p.m. sharp. You should aim to turn the paper part in during class that day. If you don't finish until later in the day, you can turn it in to Professor Kuchenbecker's office, Towne 224. The code must be emailed according to the instructions at the end of this document. Late submissions of either or both parts will be accepted until 5:00 p.m. on Friday, but they will be penalized by 25%. After that deadline, no further assignments may be submitted.

You may talk with other students about this assignment, ask the teaching team questions, use a calculator and other tools, and consult outside sources such as the Internet. To help you actually learn the material, what you write down should be your own work, not copied from a peer or a solution manual.

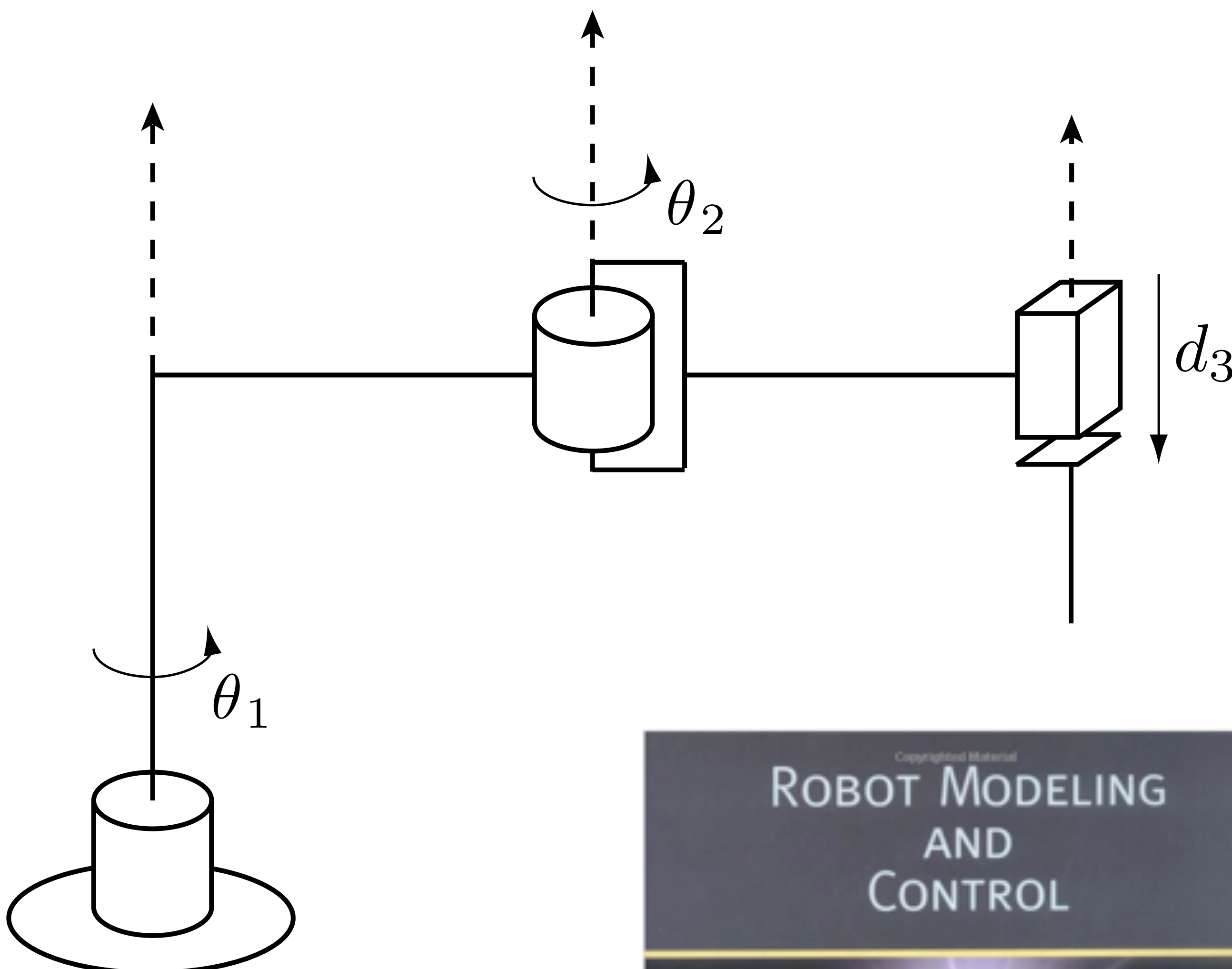
Written Problems *(30 points)*

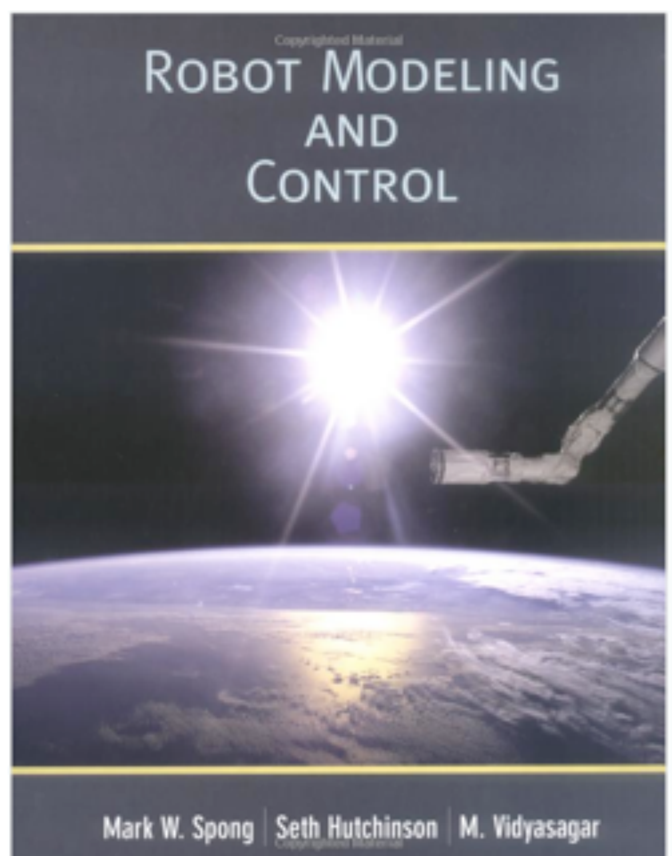
The first set of problems are written, including two from the textbook, *Robot Modeling and Control* by Spong, Hutchinson, and Vidyasagar (SHV). Please follow the extra clarifications and instructions when provided. Write in pencil, show your work clearly, box your answers, and staple your pages together.

1. Custom problem – Kinematics of Baxter *(5 points)*
   Rethink Robotics recently released a new robot named Baxter. Watch YouTube videos of Baxter (e.g., http://www.youtube.com/watch?v=rjPFqkFyrOY) to learn about its kinematics. Draw a schematic of the serial kinematic chain of Baxter's left arm (the one the woman is touching in the picture above.) Use the book's conventions for how to draw revolute and prismatic joints in 3D.

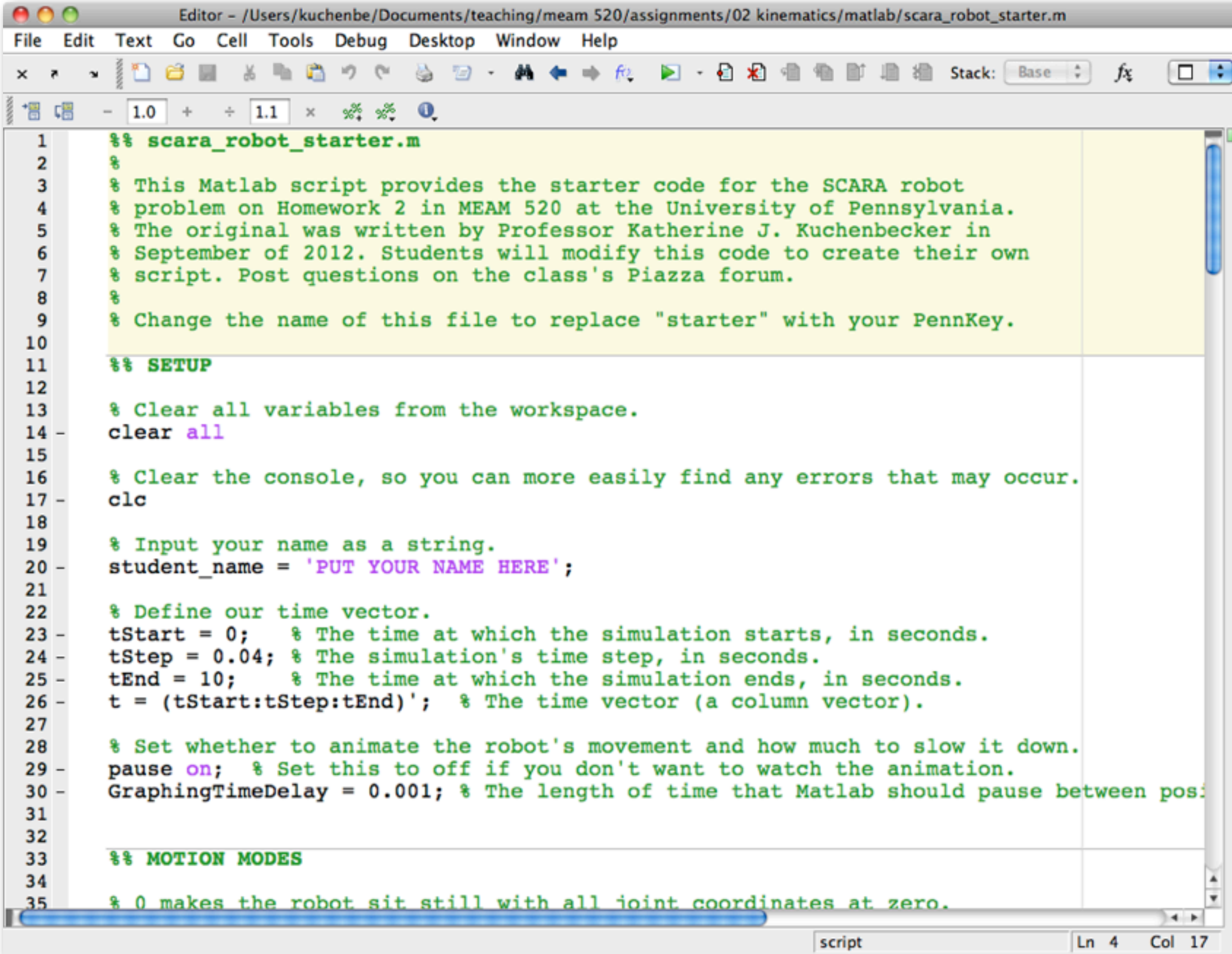
2. SHV 3-7, page 113 – Three-link Cartesian Robot *(10 points)*
   Your solution should include a schematic of the manipulator with appropriately placed coordinate frames, a table of the DH parameters, and the final transformation matrix. Then answer the following question: What are the x, y, and z coordinates of the tip of the robot's end-effector in the base frame (as a function of the robot parameters and the joint coordinates)?

DH Parameters for SCARA Manipulator

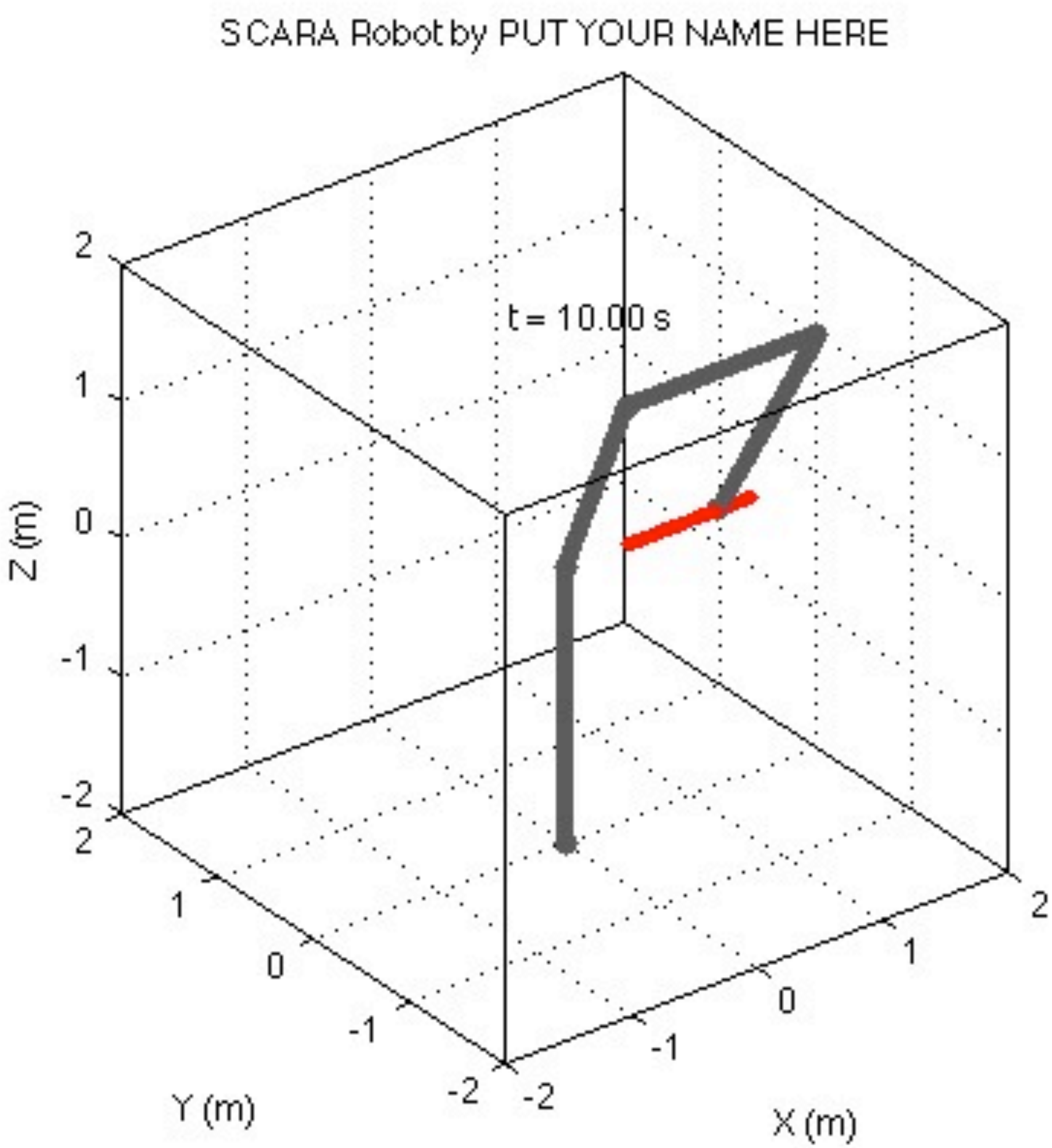




pages 91-93




```matlab
%% scara_robot_starter.m
%
% This Matlab script provides the starter code for the SCARA robot
% problem on Homework 2 in MEAM 520 at the University of Pennsylvania.
% The original was written by Professor Katherine J. Kuchenbecker in
% September of 2012. Students will modify this code to create their own
% script. Post questions on the class's Piazza forum.
%
% Change the name of this file to replace "starter" with your PennKey.

%% SETUP

% Clear all variables from the workspace.
clear all

% Clear the console, so you can more easily find any errors that may occur.
clc

% Input your name as a string.
student_name = 'PUT YOUR NAME HERE';

% Define our time vector.
tStart = 0;   % The time at which the simulation starts, in seconds.
tStep = 0.04; % The simulation's time step, in seconds.
tEnd = 10;    % The time at which the simulation ends, in seconds.
t = (tStart:tStep:tEnd)';  % The time vector (a column vector).

% Set whether to animate the robot's movement and how much to slow it down.
pause on;  % Set this to off if you don't want to watch the animation.
GraphingTimeDelay = 0.001; % The length of time that Matlab should pause between pos


%% MOTION MODES

% 0 makes the robot sit still with all joint coordinates at zero.
```



SCARA Robot by PUT YOUR NAME HERE
t = 10.00 s
2
1
0
-1
-2
Z (m)
2
1
0
-1
-2
Y (m)
-2
-1
0
1
2
X (m)