Robotics and mechatronics workshop

3 days workshop for NFU students at UTSA

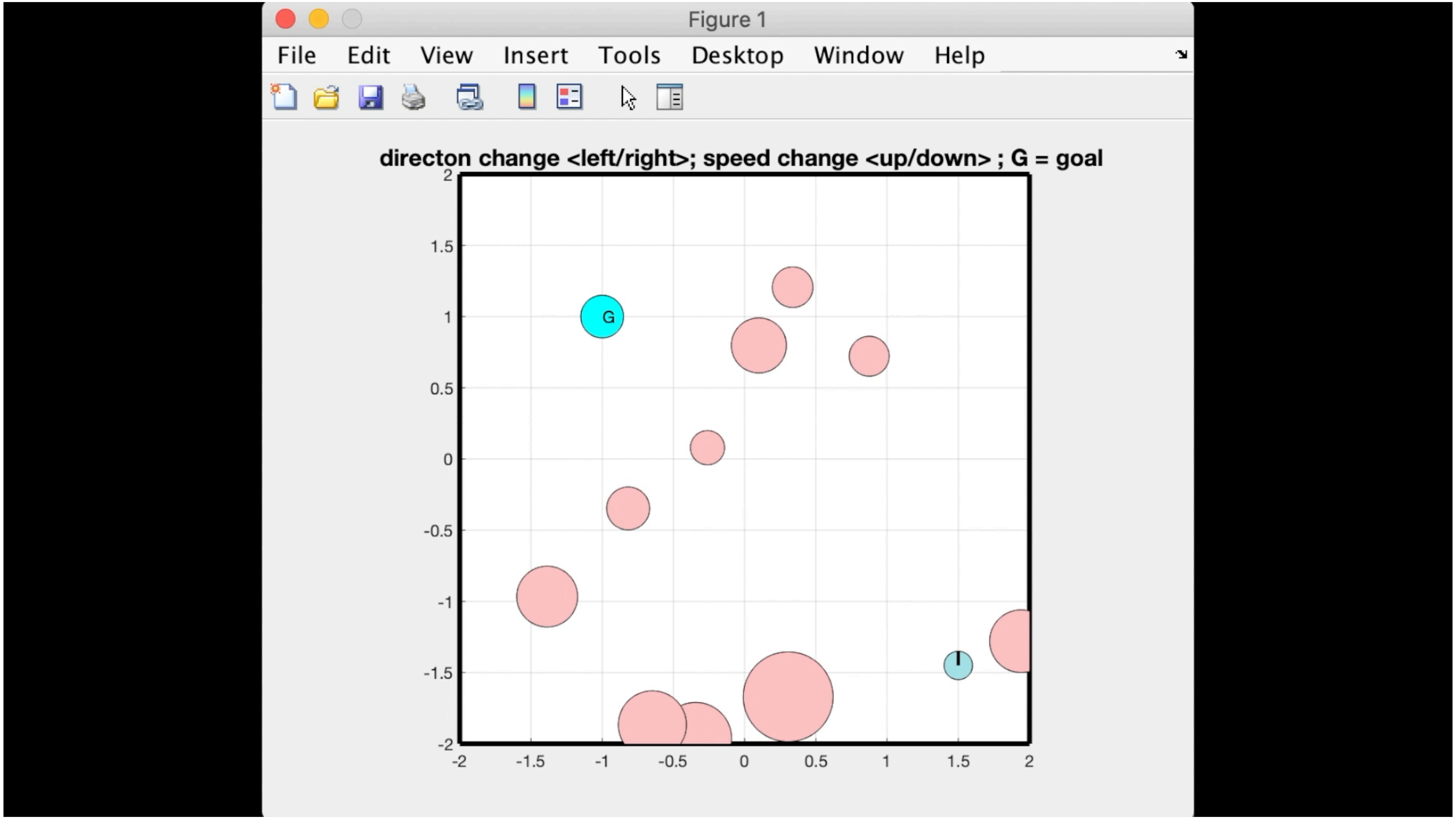
Pranav A. Bhounsule
website: https://pab47.github.io/
email: pranav@uic.edu

Robotics Overview

course outcomes

- Learn MATLAB (as a tool to simulation and animation)
- Modeling a manipulator
- Modeling a differential drive car
- Final project (create a video game)

Final robotics project



Robotics course material content

(I emailed you a folder)

- robotics_workshop.pdf
  - main document with overall plan;
  - 4 parts: each part has an exercise to be done after workshop ends
- MATLAB-basics.pdf, MATLAB-scripts.pdf, MATLAB-animations.pdf
  - pdf files of MATLAB reference material (Part 1)
- robotics_notes.pdf
  - some theory for manipulators and car modeling
- Folder “matlab” that contains all programs needed
- This presentation (will be provided at the end)

Rough schedule

Monday

- 9:30 - 10:30 Part 1: MATLAB intro and basics
- 10:45 - 12:00 Part 1: MATLAB matlab scripts
- 12 - 1 Lunch break
- 1 - 2:15 Part 1: MATLAB animation
- 2:30 - 3:30 Part 2: Manipulator
- 3:45 - 5:00 Part 3: Differential drive car

Rough schedule

Tuesday

- 9:30 - 10:30 Part 4: Video game
- 10:45 - 12:00 Part 1: Arduino basics
- 12 - 1 Lunch break
- 1 - 2:15 Part 1: Arduino basics (contd)
- 2:30 - 3:30 Part 2: Servo and sensor
- 3:45 - 5:00 Part 3: Motor

Rough schedule

Wednesday

- 9: 30 - 10:30 Part 3: Motors (contd.)
- 10:45 - 12:00 Part 4: Car consruction
- 12 - 1 Lunch break
- 1 - 2:15 Part 4: Car construction
- 2:30 - 3:30 Part 4: Car programming
- 3:45 - 5:00 Part 4: Car programming

Part 1: MATLAB

Lets work together

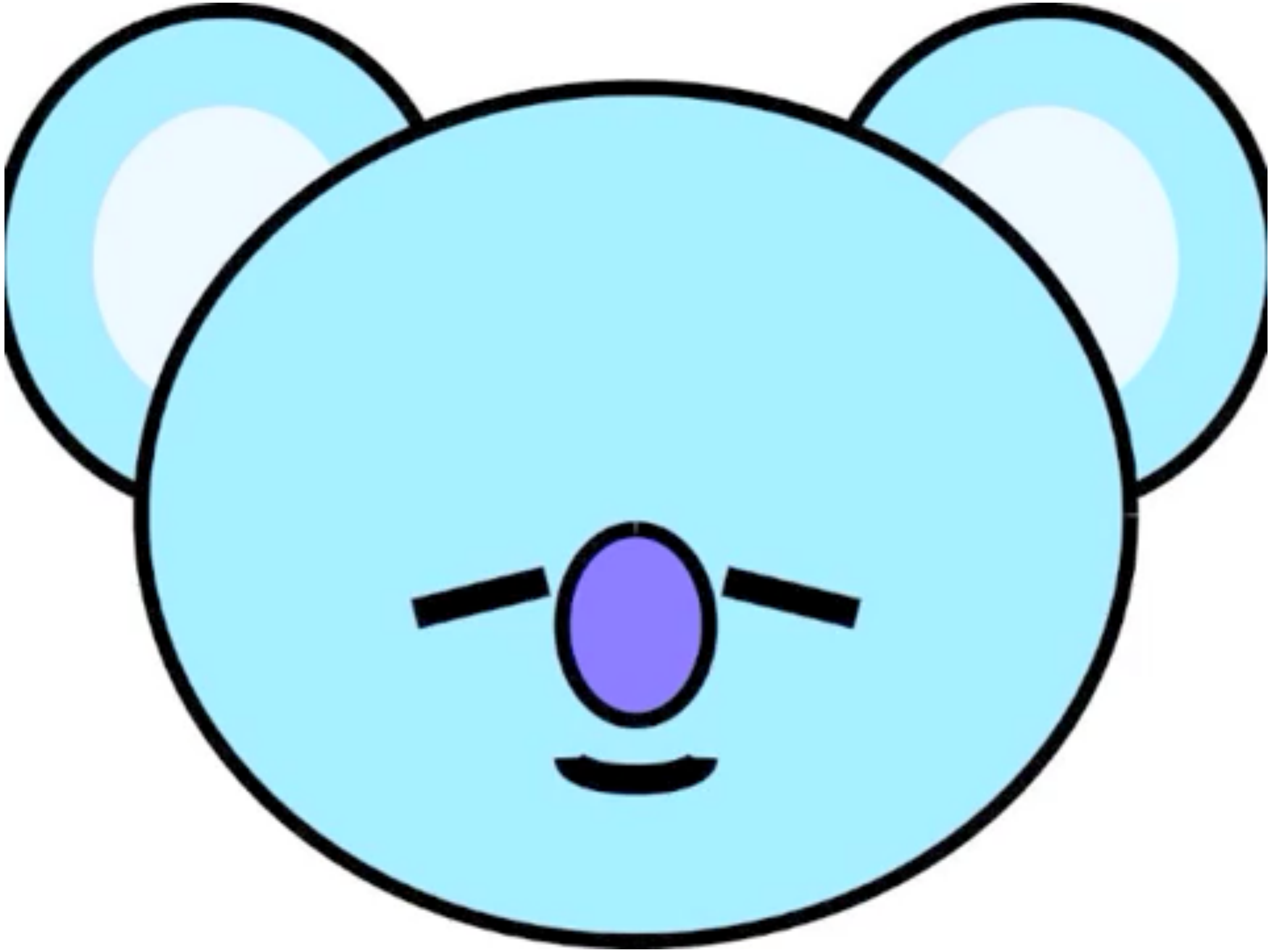
1. Basic usage of MATLAB

2. Scripts in MATLAB

3. Making animation (this is fun)

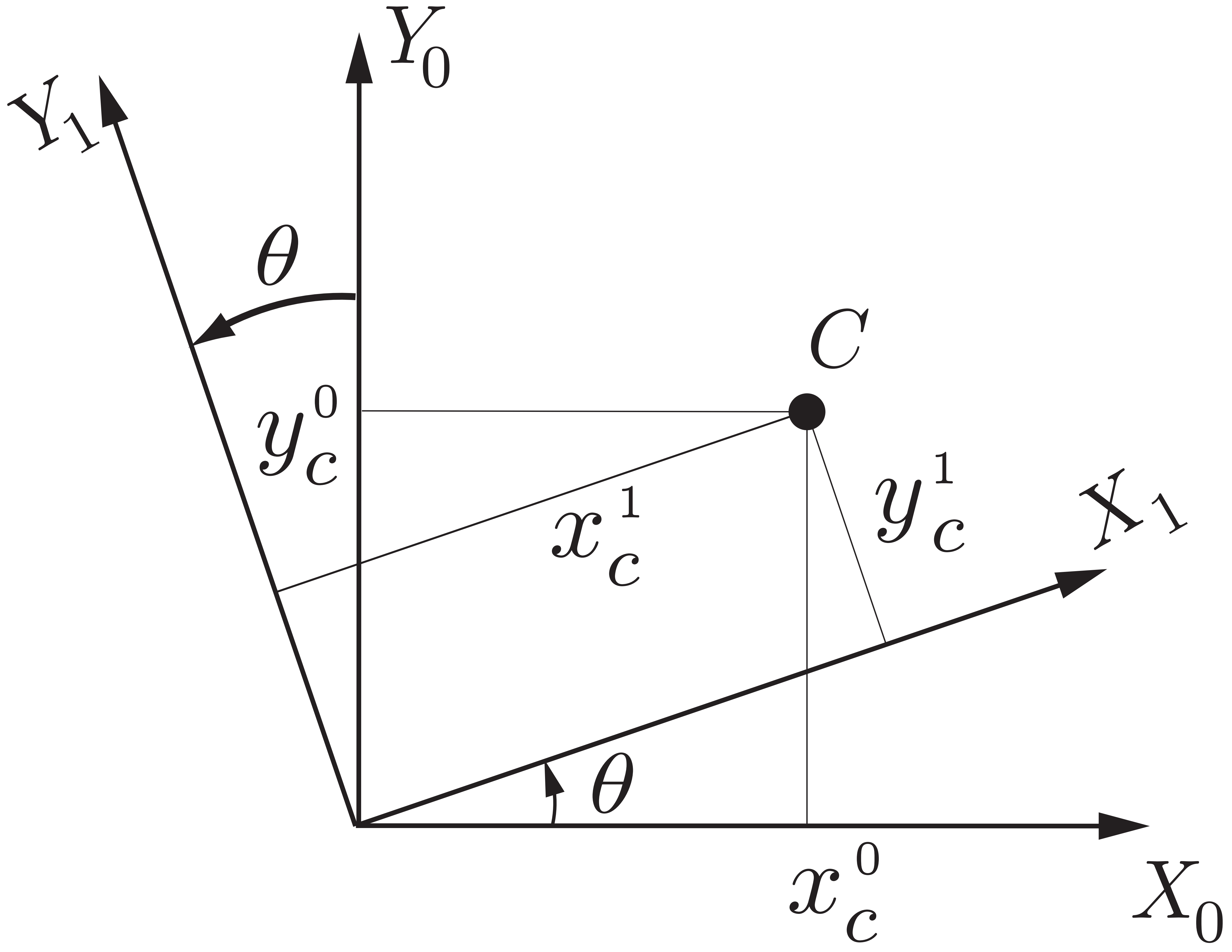
Part 1

Exercise: Animate a face



Part 2: Manipulator

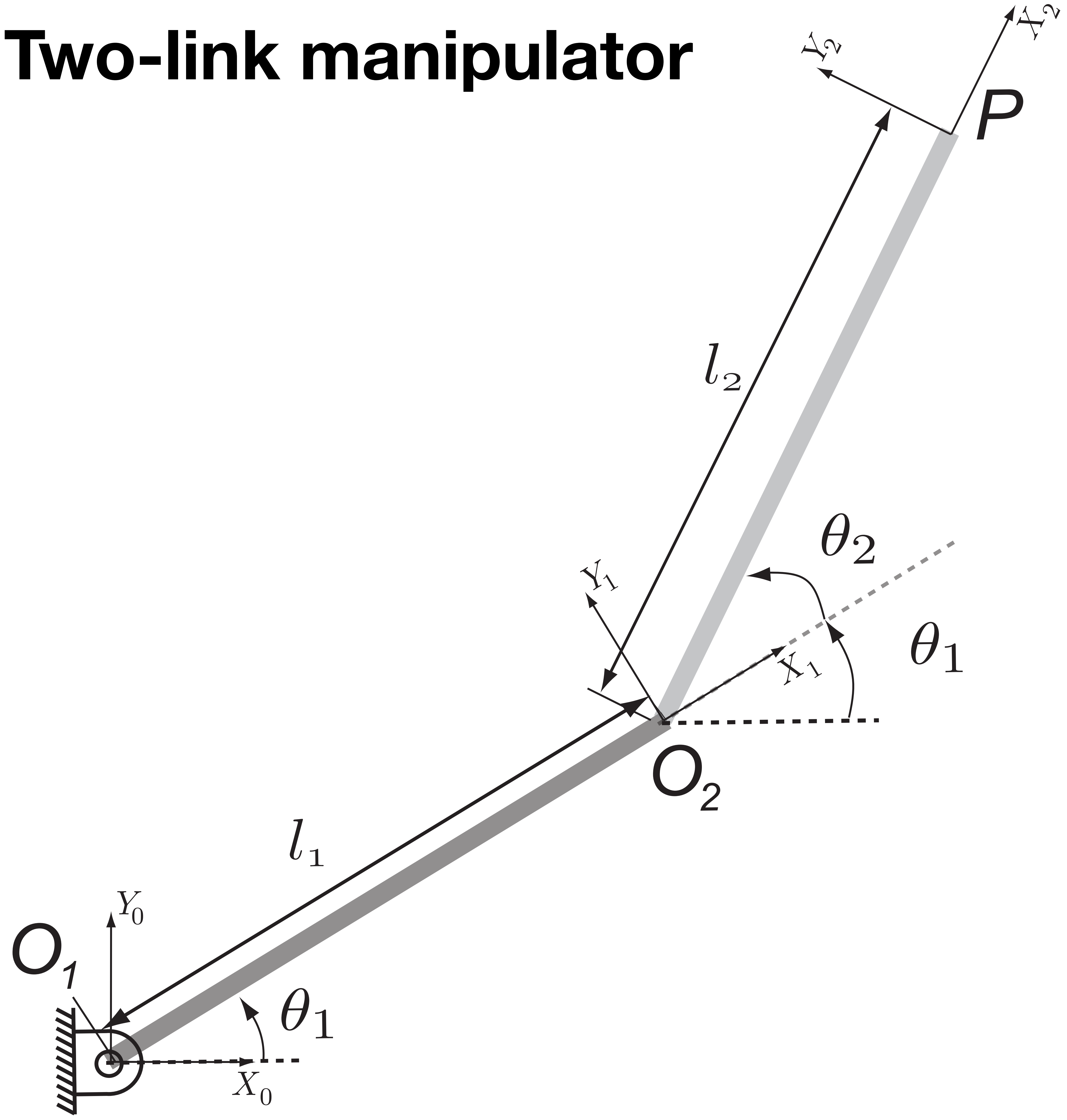
Coordinate frames



$$\begin{bmatrix} x_c^0 \\ y_c^0 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_c^1 \\ y_c^1 \end{bmatrix}$$

Part 2: Manipulator

Two-link manipulator



Location of elbow O_2

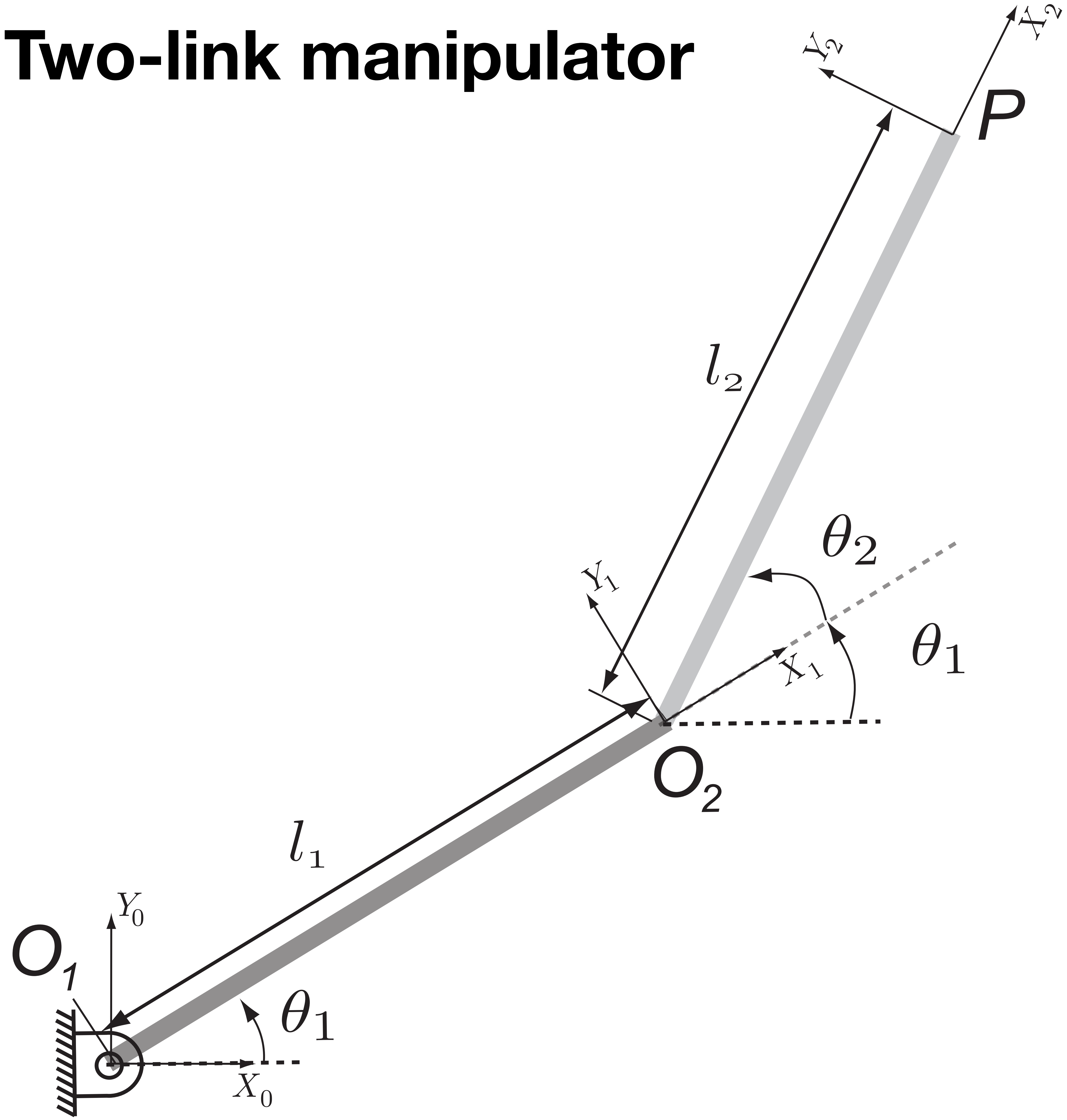
$$\begin{bmatrix} x^0_{O_2} \\ y^0_{O_2} \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 \\ \sin\theta_1 & \cos\theta_1 \end{bmatrix} \begin{bmatrix} \ell_1 \\ 0 \end{bmatrix} = \begin{bmatrix} \ell_1 \cos\theta_1 \\ \ell_1 \sin\theta_1 \end{bmatrix}$$

Location of tip P

$$\begin{bmatrix} x^0_{P} \\ y^0_{P} \end{bmatrix} = \begin{bmatrix} x^0_{O_2} \\ y^0_{O_2} \end{bmatrix} + \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 \\ \sin\theta_2 & \cos\theta_2 \end{bmatrix} \begin{bmatrix} \ell_2 \\ 0 \end{bmatrix} = \begin{bmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 \end{bmatrix}$$

Part 2: Manipulator

Two-link manipulator



Forward kinematics

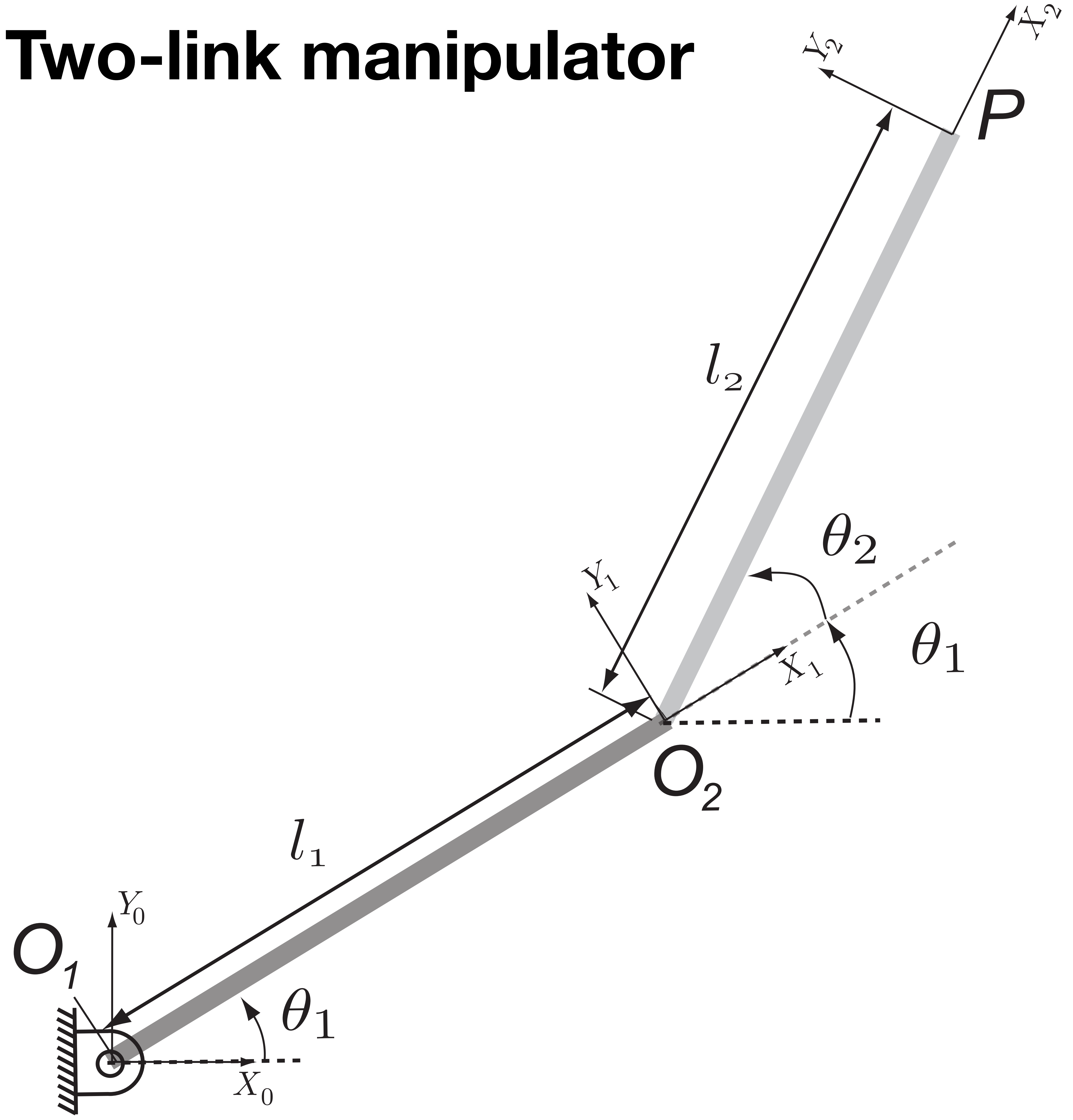
Given theta_1 and theta_2 find:
x_p and y_p

$$\begin{bmatrix} x_P^0 \\ y_P^0 \end{bmatrix} = \begin{bmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 \end{bmatrix}$$

see manipulator_forward.m

Part 2: Manipulator

Two-link manipulator



Inverse kinematics

Given x_p and y_p find:
theta_1 and theta_2

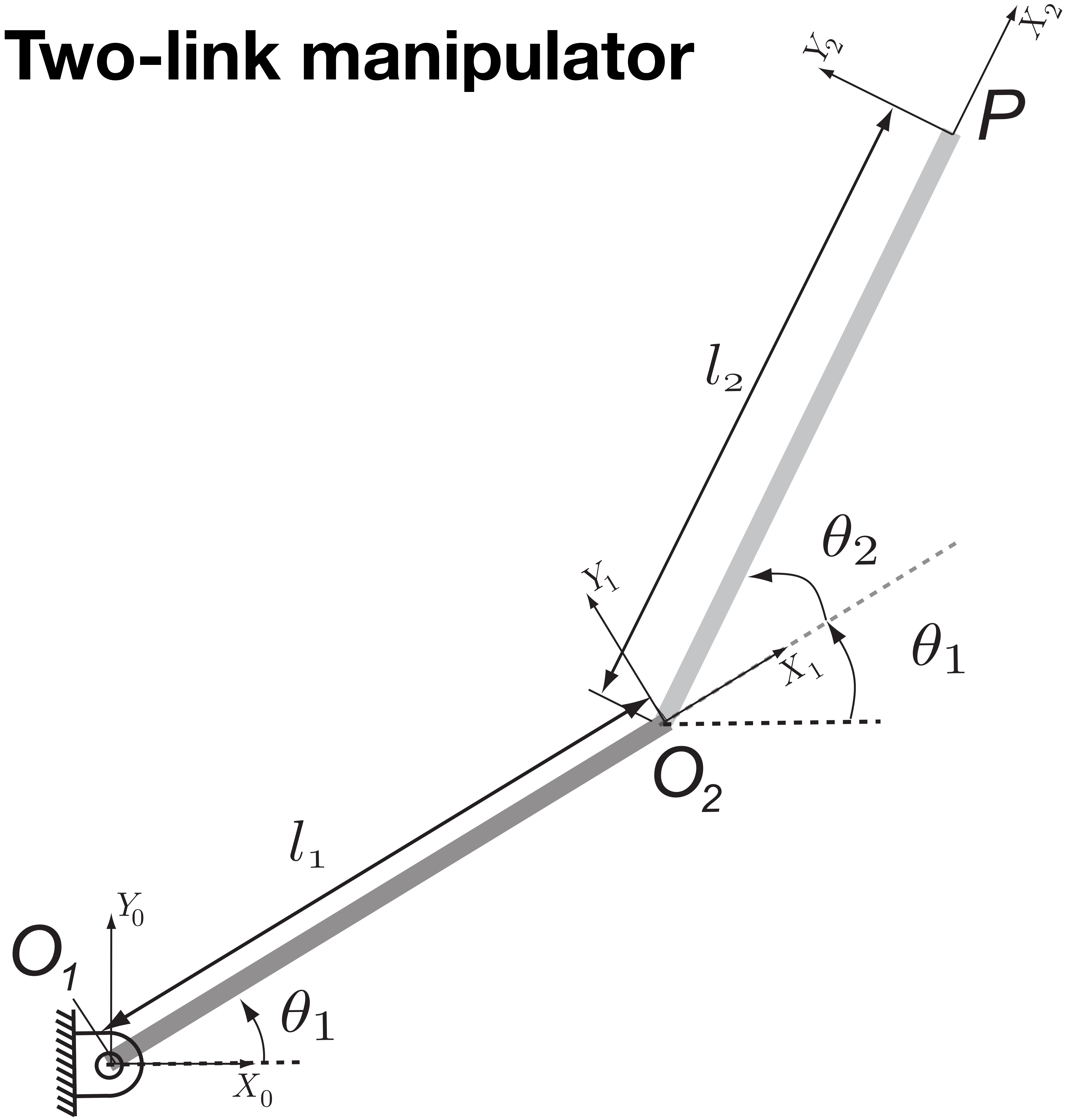
$$\begin{bmatrix} x_P^0 \\ y_P^0 \end{bmatrix} = \begin{bmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 \end{bmatrix}$$

- much harder
- many/no solutions

see manipulator_inverse.m

Part 2: Manipulator

Two-link manipulator



Get the manipulator to draw a circle

Again Inverse kinematics

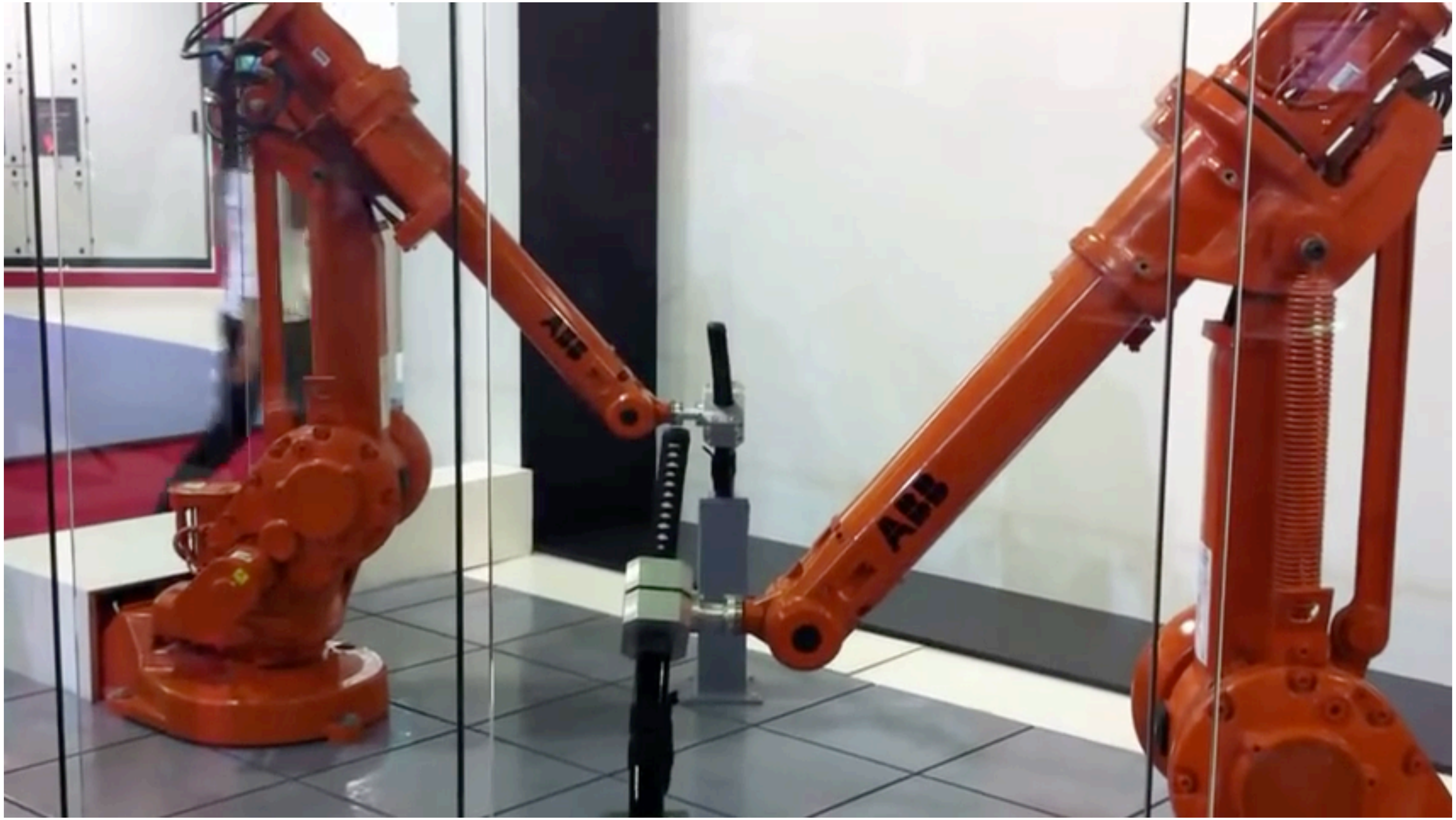
Given x_p(i) and y_p(i) points circumference, find theta_1(i) and theta_2(i)

$$\begin{bmatrix} x_P^0 \\ y_P^0 \end{bmatrix} = \begin{bmatrix} \ell_1 \cos\theta_1 + \ell_2 \cos\theta_2 \\ \ell_1 \sin\theta_1 + \ell_2 \sin\theta_2 \end{bmatrix}$$

see manipulator_inverse_circle.m

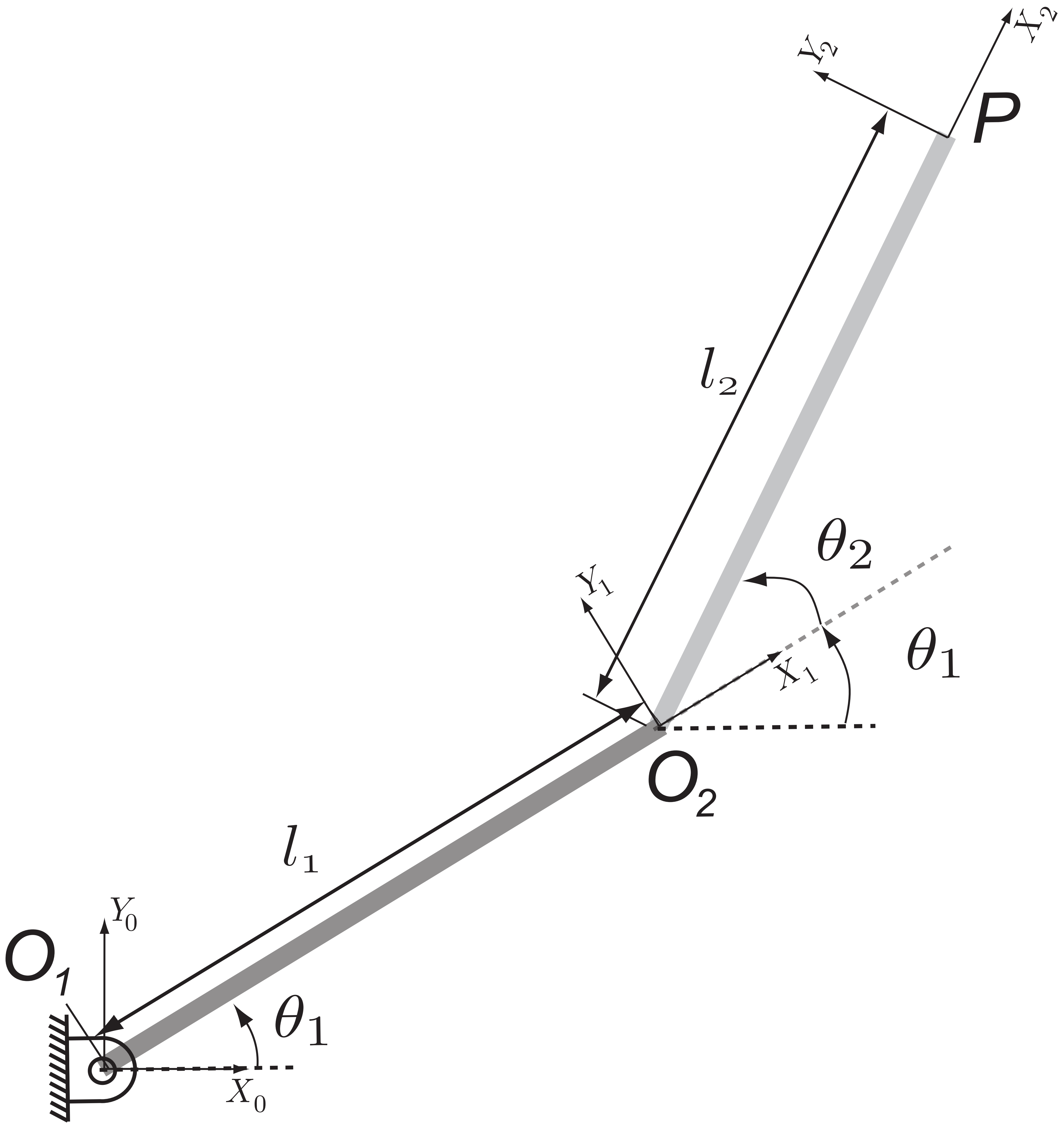
Part 2: Inverse kinematics example

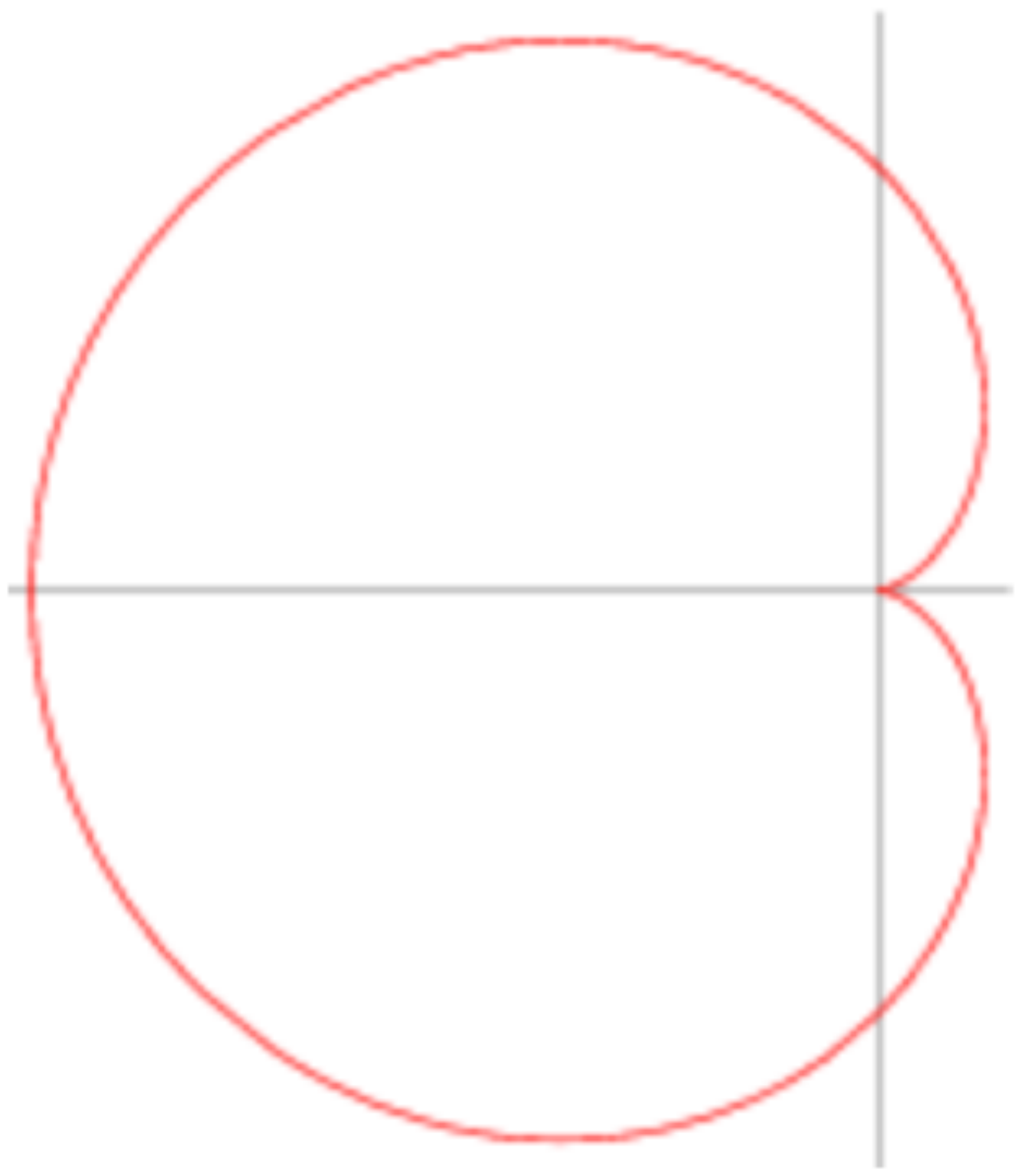
Part 2: Inverse kinematics example



Part 2: Manipulator

Exercise: Draw the cardoid (see eqn in notes)

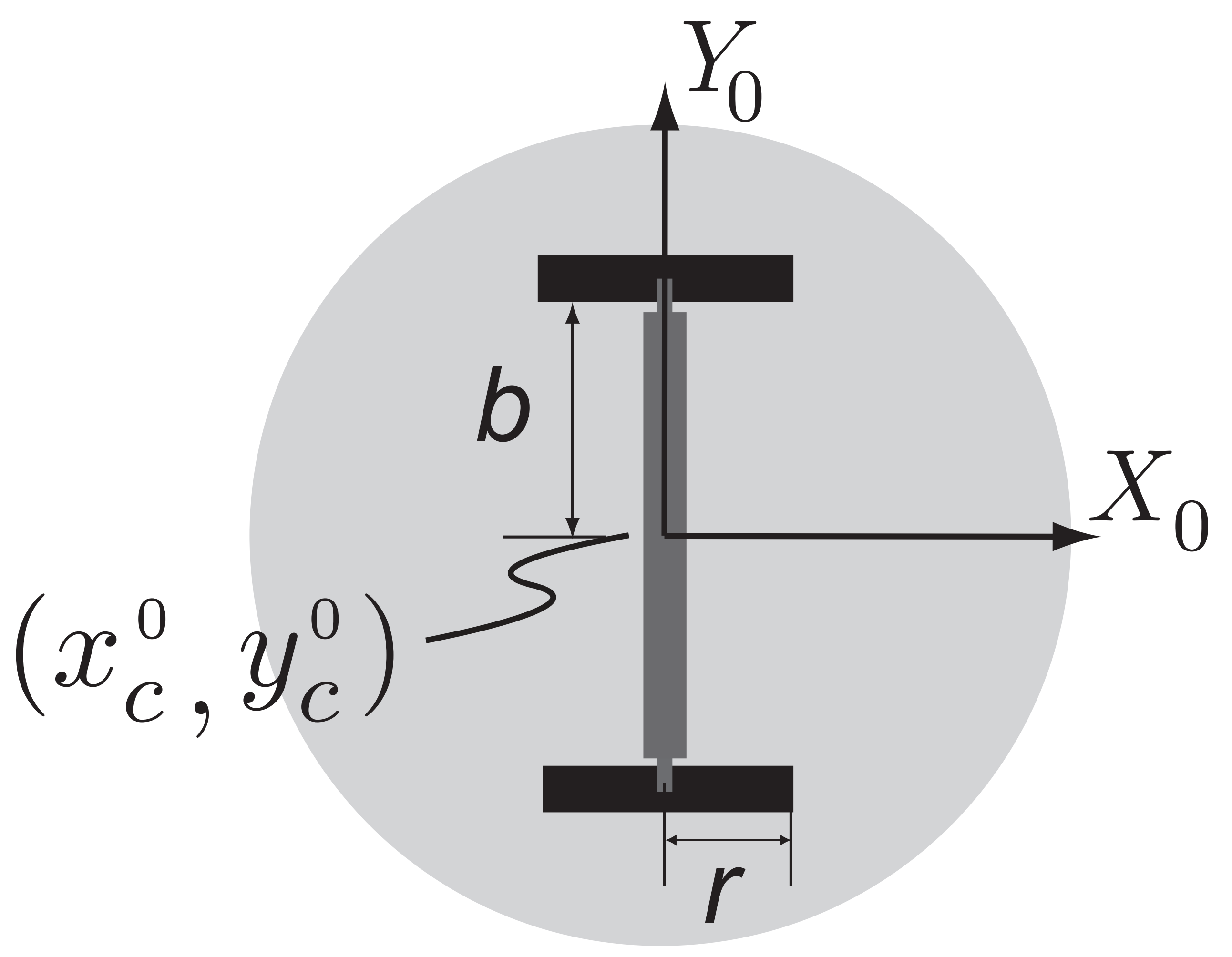




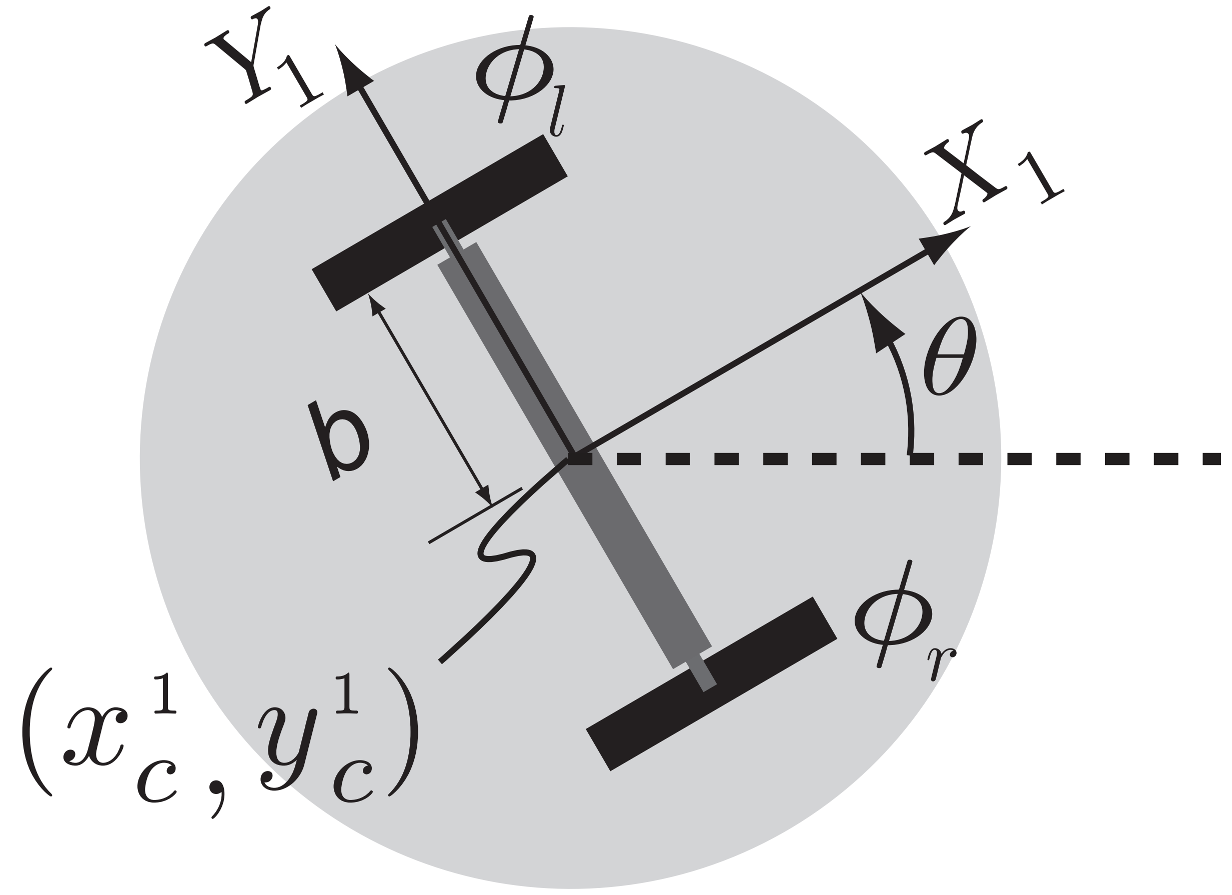
Part 3: Mobile robot

Differential drive car

(a) position at start



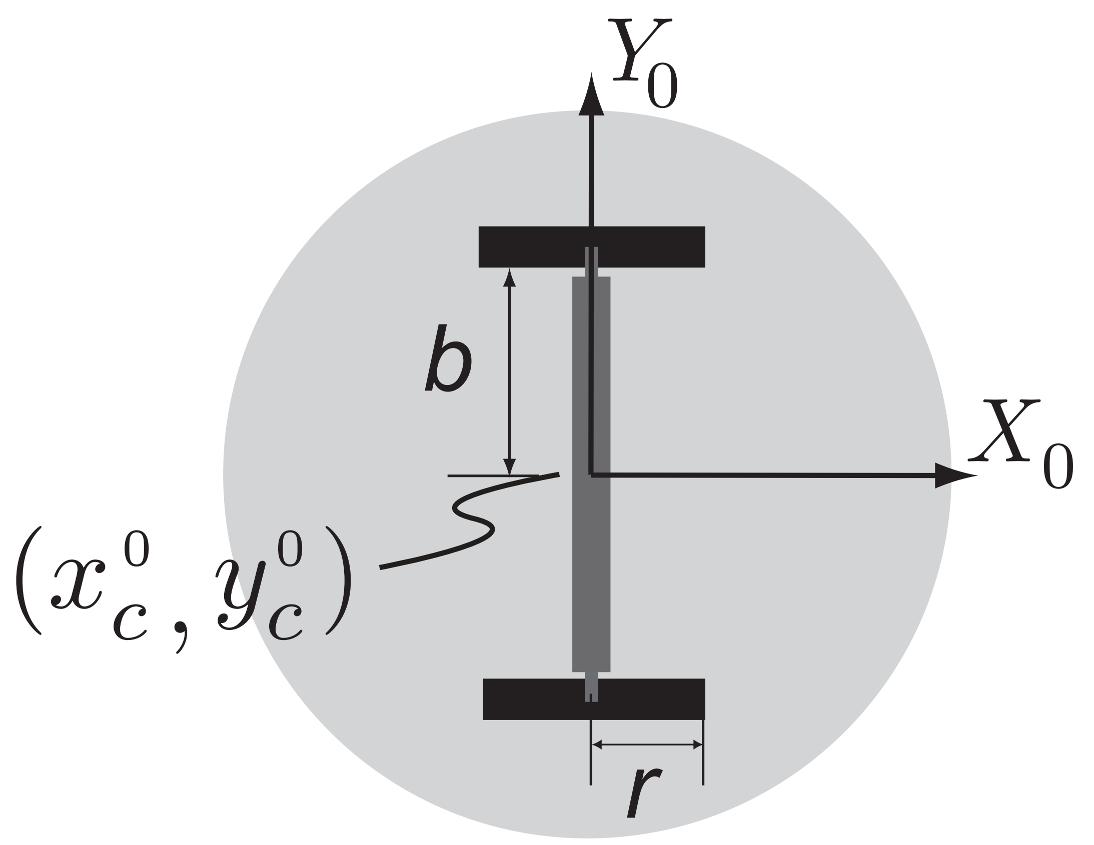
(b) position at some point of time



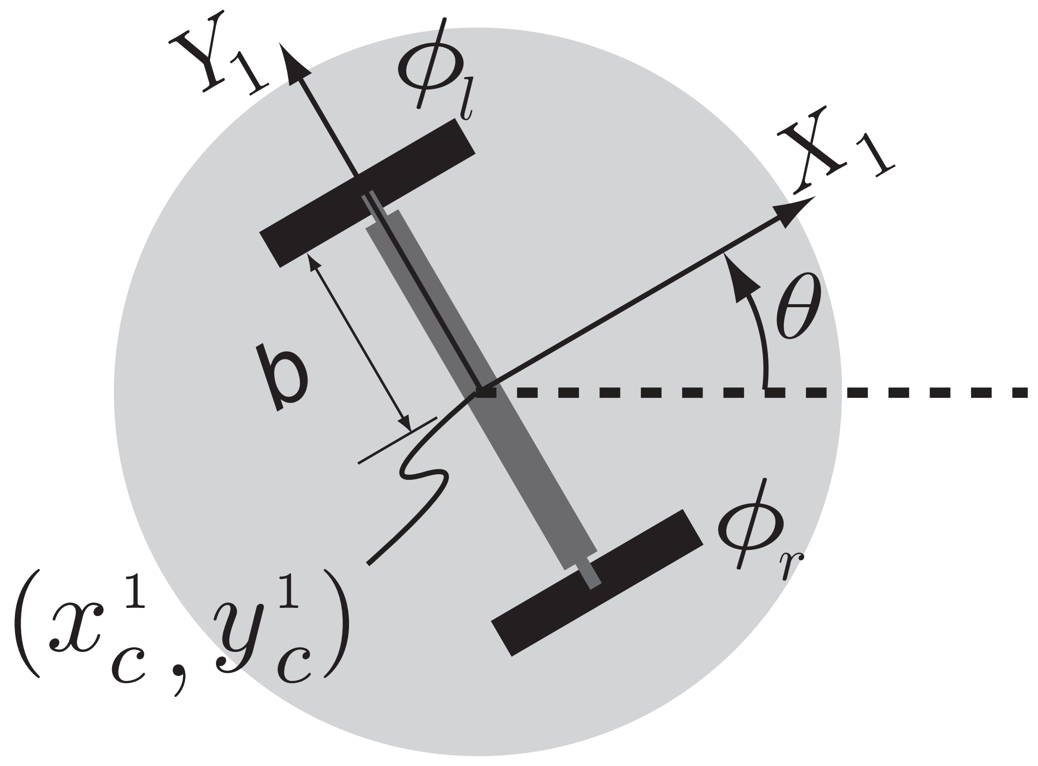
Part 3: Mobile robot

Differential drive car

(a) position at start



(b) position at some point of time



kinematics of x_c and y_c

$$\dot{x}_c^1 = 0.5r(\dot{\phi}_r + \dot{\phi}_l)$$
$$\dot{y}_c^1 = 0$$

$$\begin{bmatrix}\dot{x}_c^0 \\ \dot{y}_c^0\end{bmatrix} = \begin{bmatrix}\cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta)\end{bmatrix}\begin{bmatrix}\dot{x}_c^1 \\ \dot{y}_c^1\end{bmatrix}$$
$$= \begin{bmatrix}\cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta)\end{bmatrix}\begin{bmatrix}0.5r(\dot{\phi}_r + \dot{\phi}_l) \\ 0\end{bmatrix}$$

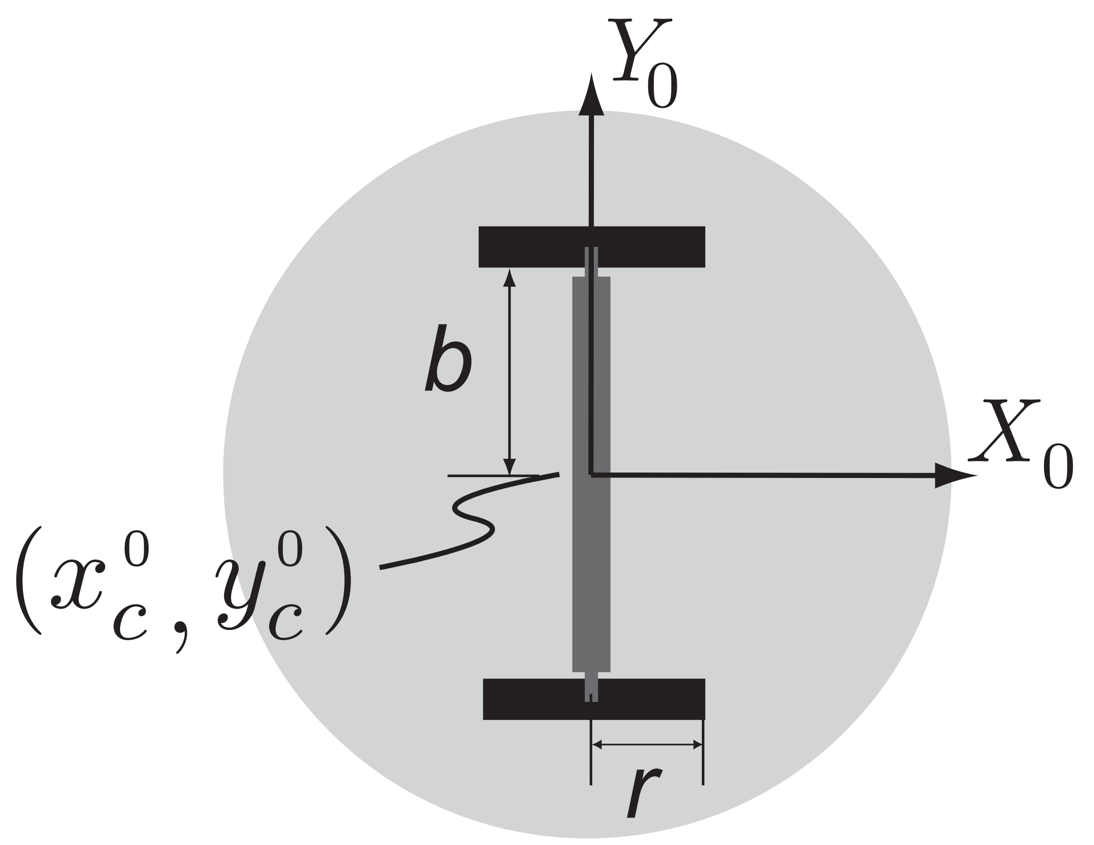
$$\dot{x}_c^0 = 0.5r(\dot{\phi}_r + \dot{\phi}_l)\cos(\theta)$$
$$\dot{y}_c^0 = 0.5r(\dot{\phi}_r + \dot{\phi}_l)\sin(\theta)$$

Part 3: Mobile robot

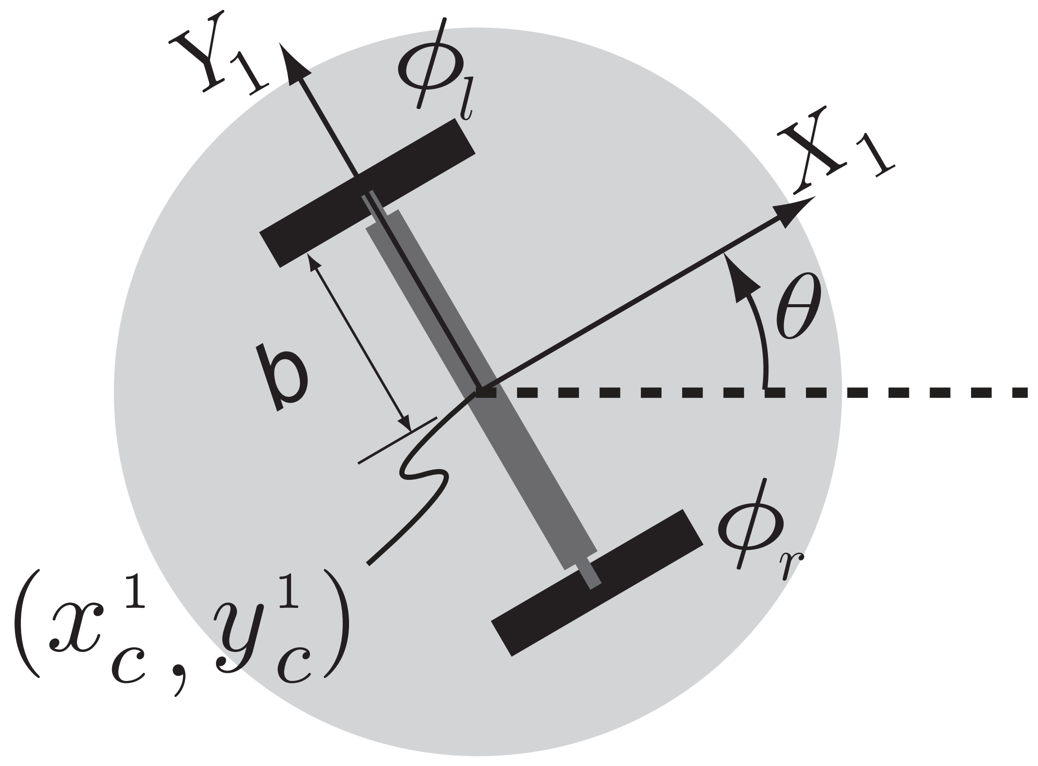
kinematics of rotation theta

Differential drive car

(a) position at start



(b) position at some point of time



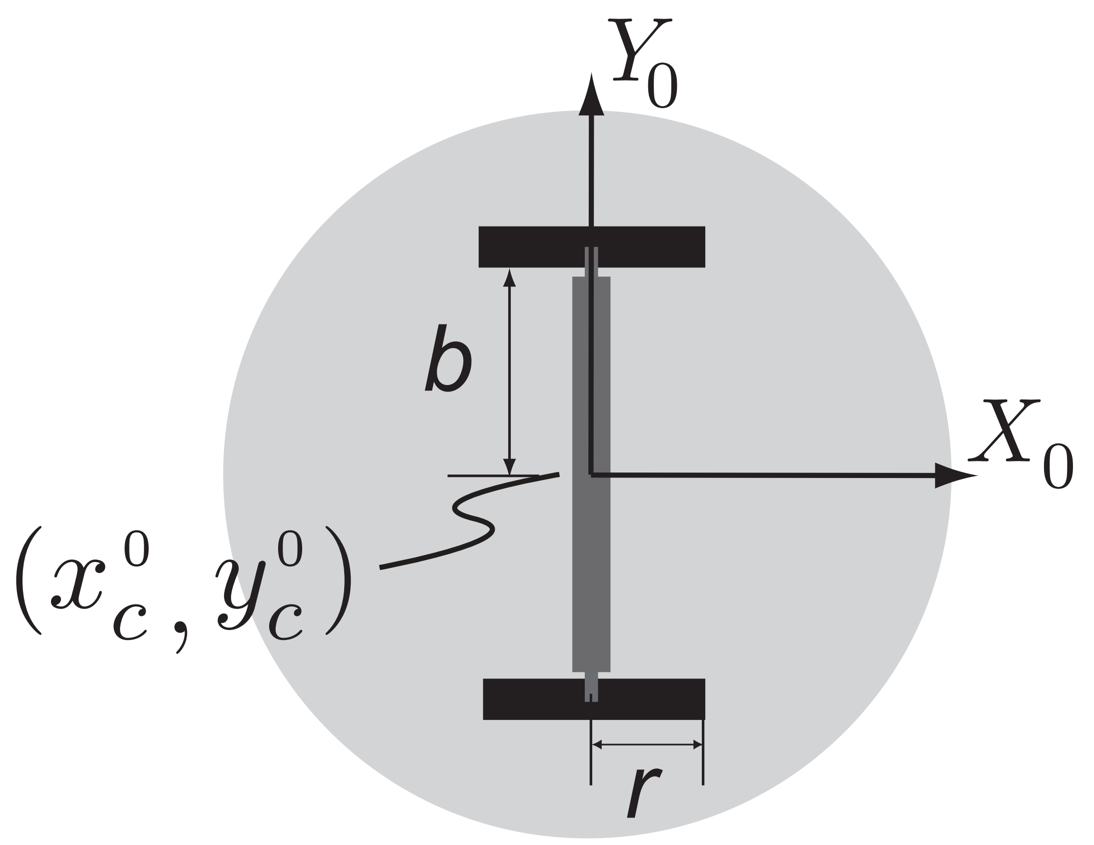
$$\dot{\theta} = 0.5\frac{r}{b}(\dot{\phi}_r - \dot{\phi}_l)$$

$$\dot{x}_c^0 = 0.5r(\dot{\phi}_r + \dot{\phi}_l)\cos(\theta)$$
$$\dot{y}_c^0 = 0.5r(\dot{\phi}_r + \dot{\phi}_l)\sin(\theta)$$

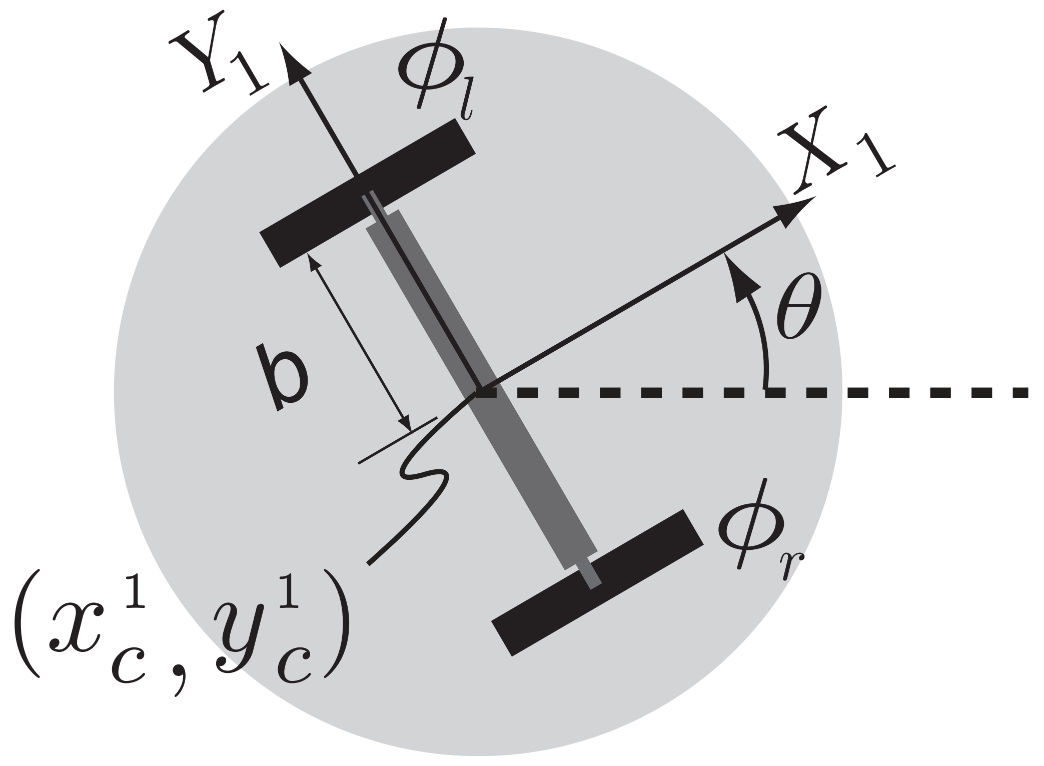
Part 3: Mobile robot

Differential drive car

(a) position at start



(b) position at some point of time



All equations

$$\dot{x}_c^0 = 0.5r(\dot{\phi}_r + \dot{\phi}_l)\cos(\theta)$$
$$\dot{y}_c^0 = 0.5r(\dot{\phi}_r + \dot{\phi}_l)\sin(\theta)$$
$$\dot{\theta} = 0.5\frac{r}{b}(\dot{\phi}_r - \dot{\phi}_l)$$

Simplified

$$\dot{x}_c^0 = 0.5r\omega\cos(\theta)$$
$$\dot{y}_c^0 = 0.5r\omega\sin(\theta)$$
$$\dot{\theta} = 0.5\frac{r}{b}\Omega$$

controlled variables

Part 3: Mobile robot

Integration

$$x_c^0(t_{i+1}) = x_c^0(t_i) + 0.5r\omega(t_i)\cos\theta(t_i)h$$

$$y_c^0(t_{i+1}) = y_c^0(t_i) + 0.5r\omega(t_i)\sin\theta(t_i)h$$

$$\theta(t_{i+1}) = \theta(t_i) + 0.5\frac{r}{b}\Omega(t_i)h$$

controlled variables

see diff_drive_main.m
see euler_integration.m

Part 3: Mobile robot

Exercise: Can you control the robot to write your initials?

Part 4: Project

see car_game.m

Main loop

```
function car_game
%Modified the pong code by David Buckingham
%https://www.mathworks.com/matlabcentral/fileexchange/31177-dave-s-matlab-pong

%%%%%% main part of the code %%%
global game_over

close all
initData  %first function, initialize the data variables
initFigure %second function, initialize the figure
while ~game_over %runs till game_over = 1
    moveCar; %second function, compute car movement including collision detection
    refreshPlot; %fourth function, refresh plot based on moveCar
end
```

Rough schedule

Tuesday

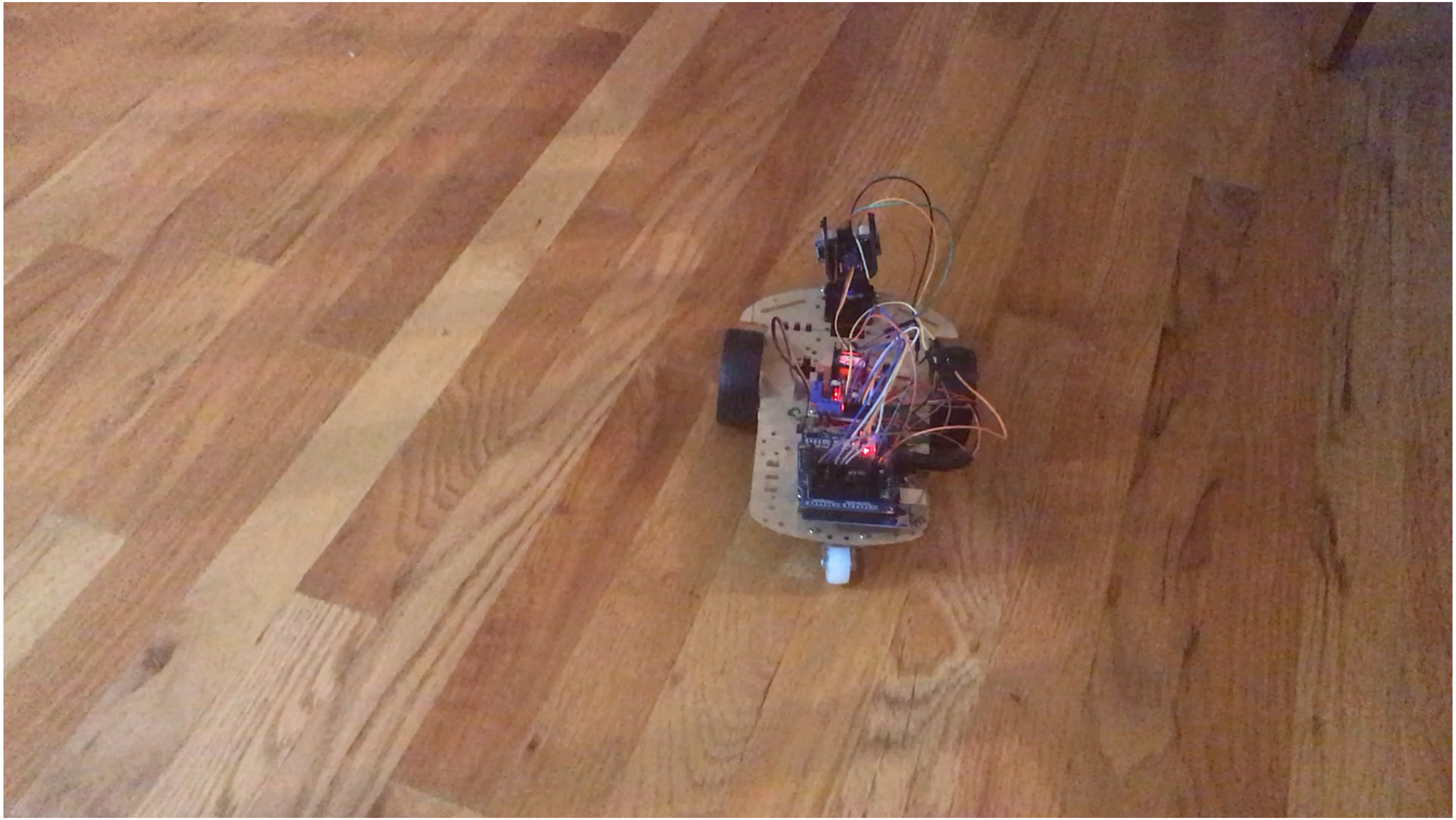
- 9: 30 - 10:30 Part 4: Video game
- 10:45 - 12:00 Part 1: Arduino basics
- 12 - 1 Lunch break
- 1 - 2:15 Part 1: Arduino basics (contd)
- 2:30 - 3:30 Part 2: Servo and sensor
- 3:45 - 5:00 Part 3: Motor

Mechatronics Overview

course outcomes

- Learn Arduino (as a tool to create a mechatronics system)
- Basic electronics: resistor, breadboard, push-button, light-emitting diode.
- Basics C programming: variables, functions, loops, conditionals
- Using Analog in/out and digital in/out
- Sensors: Ultrasonic sensor
- Actuators: DC motors and servos
- Final project (create a different drive car)

Final mechatronics project

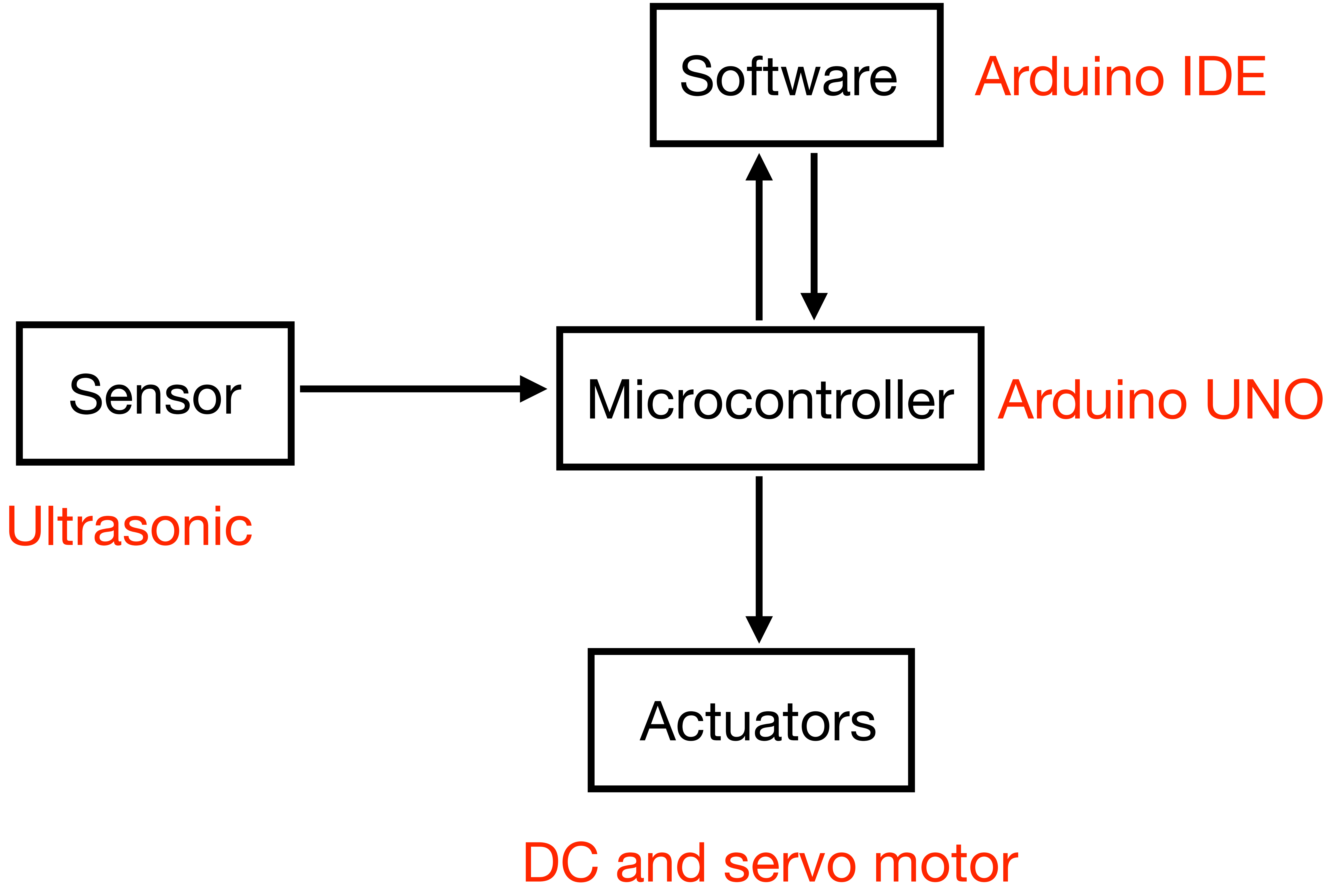


Mechatronics course material content

(I emailed you a folder)

- mechatronics_workshop.pdf
    - main document with overall plan;
    - 4 parts: each part has an exercise to be done after workshop ends
- Arduino basics, servo-sensor, motor, car-project pdf files
- Folder “arduino” that contains all programs needed
- This presentation (will be provided at the end)

Mechatronics system

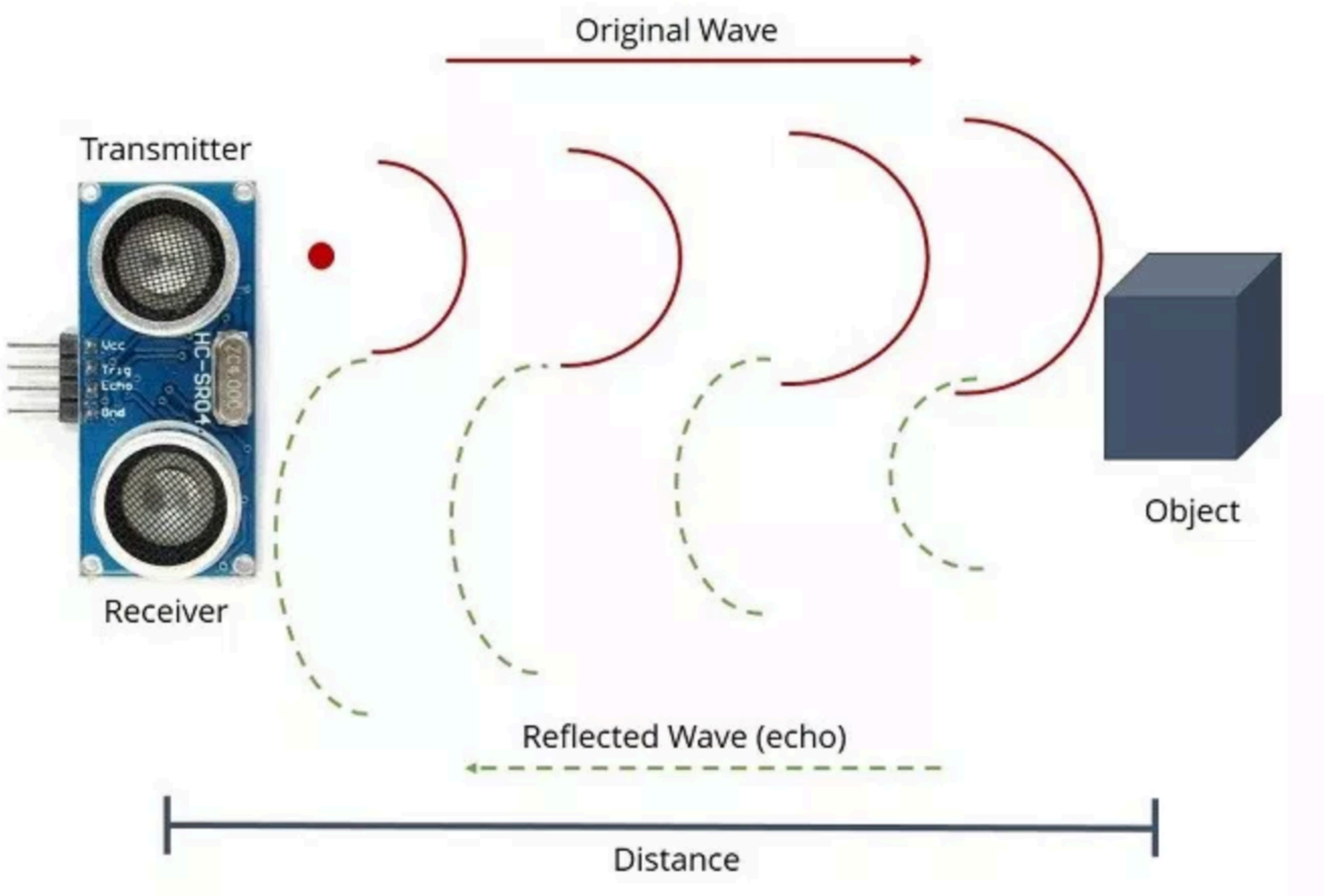


Part 1: Arduino basics

Check 1. Arduino-basics.pdf

Part 2: Servo and sensor

Ultrasonic sensor

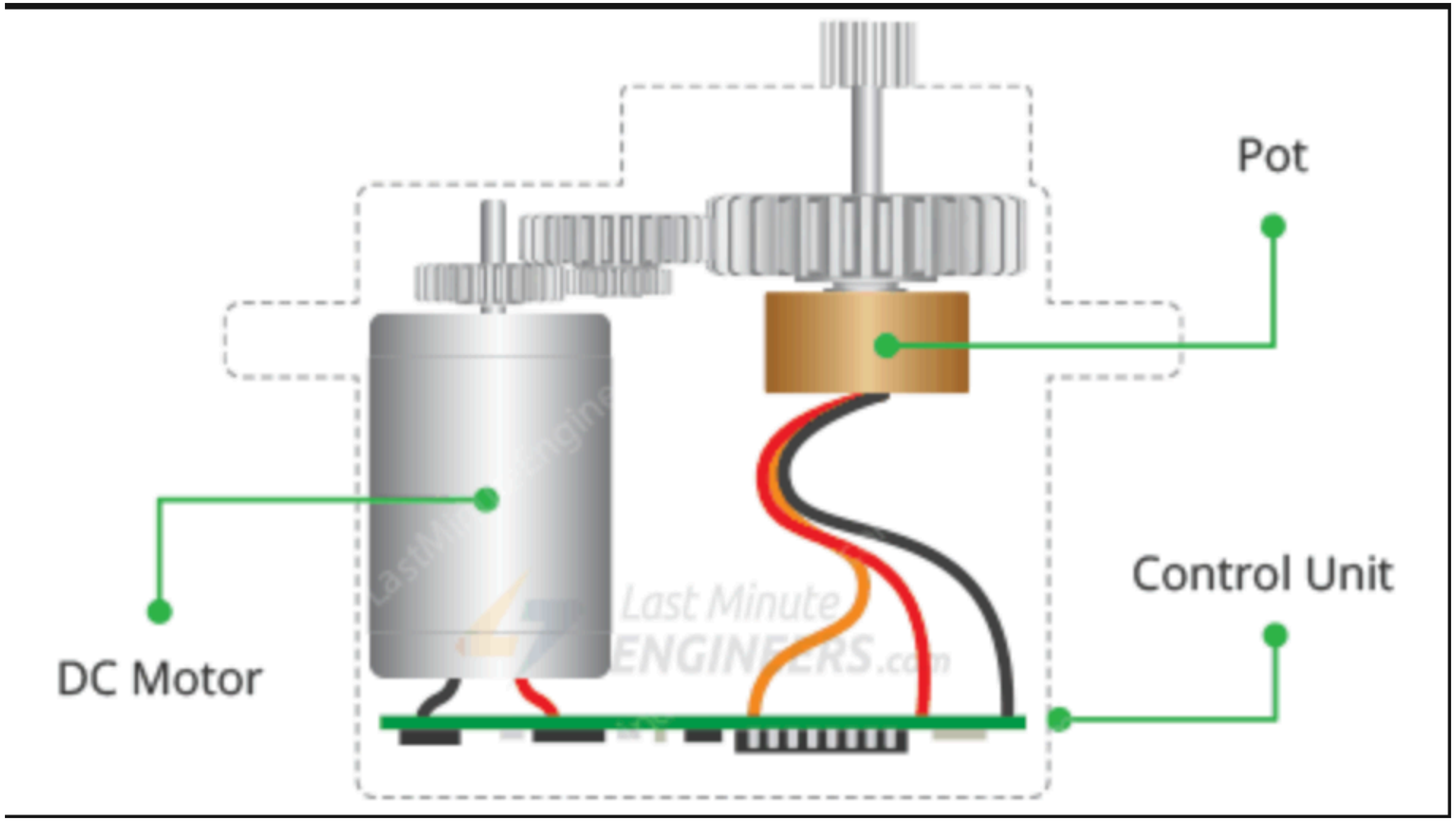


https://randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/

Part 2: Servo and sensor

Servo motor: DC motor + potentiometer (position sensor)

more details see DC motor notes



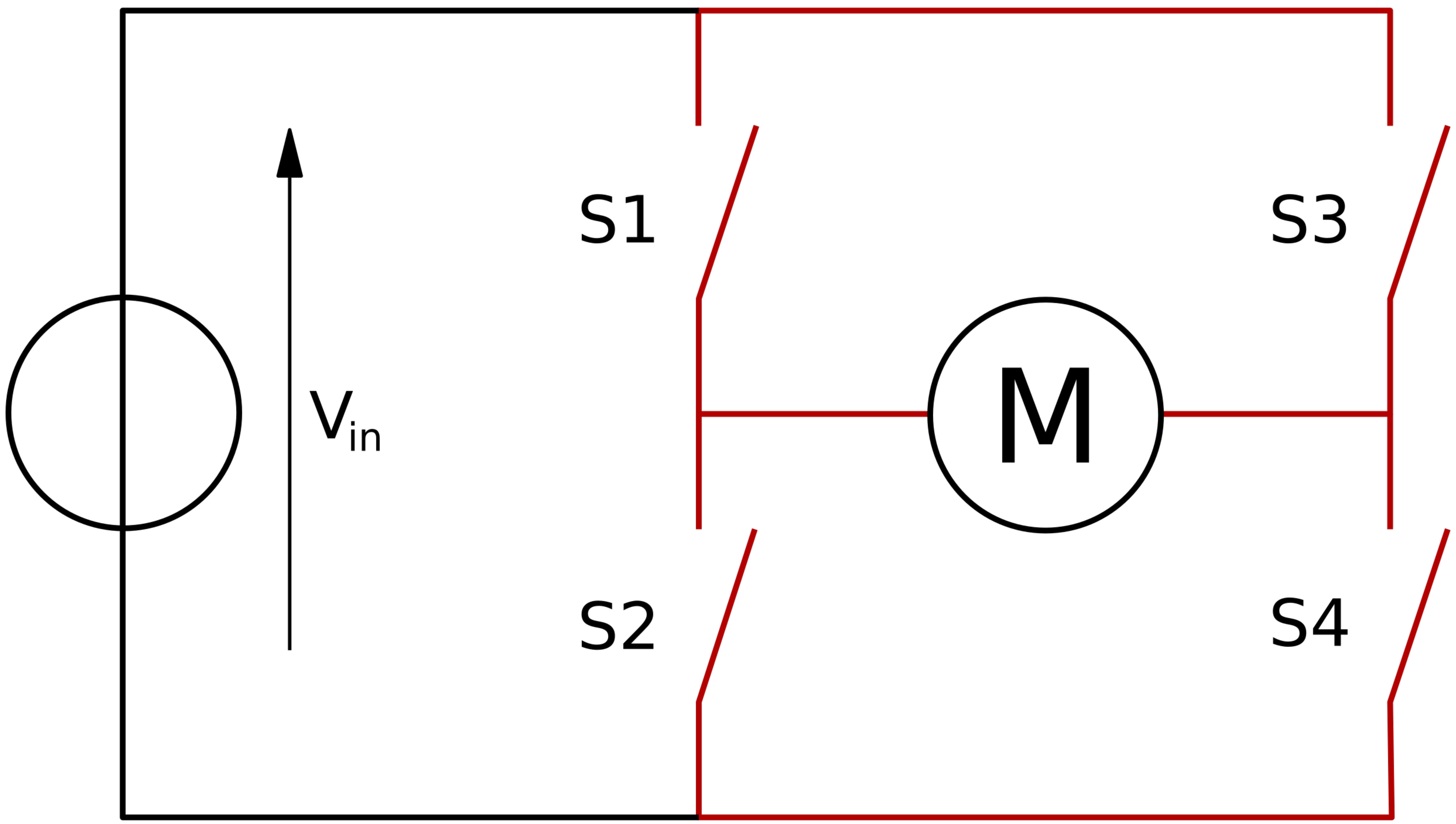
https://lastminuteengineers.com/servo-motor-arduino-tutorial/

Part 2: Servo and sensor

Check 2.Arduino-servo-sensor

Part 3: DC motor

H-bridge or motor controller

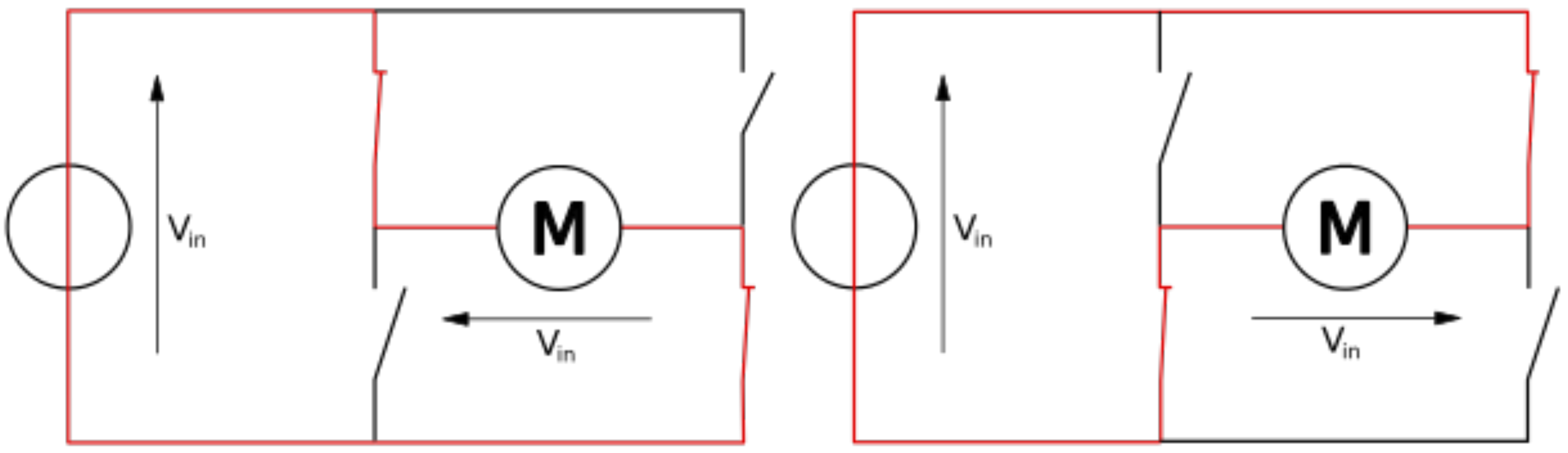


DC motor

Part 3: DC motor

Direction control

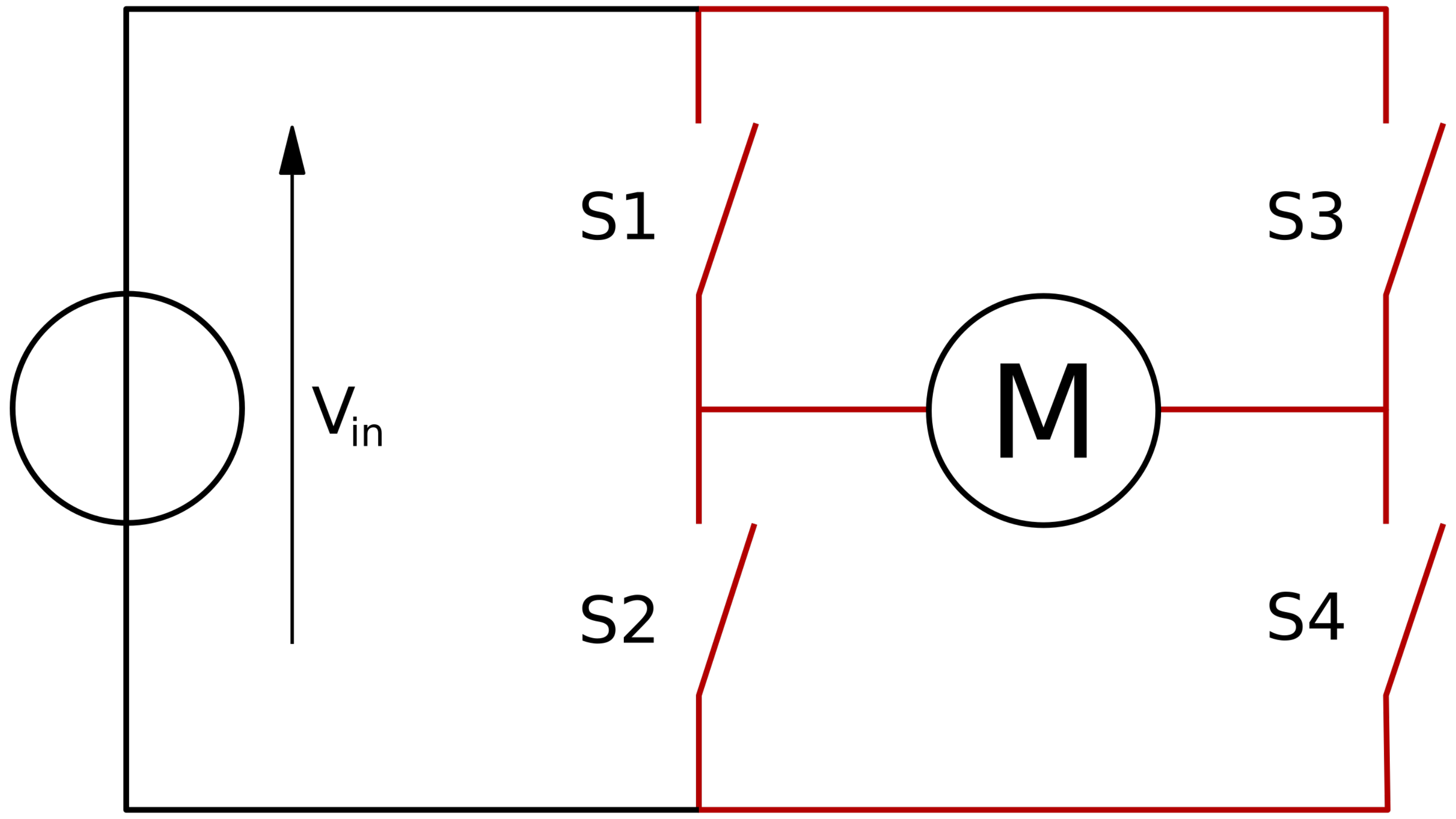
Motor spin direction controller by closing appropriate switches



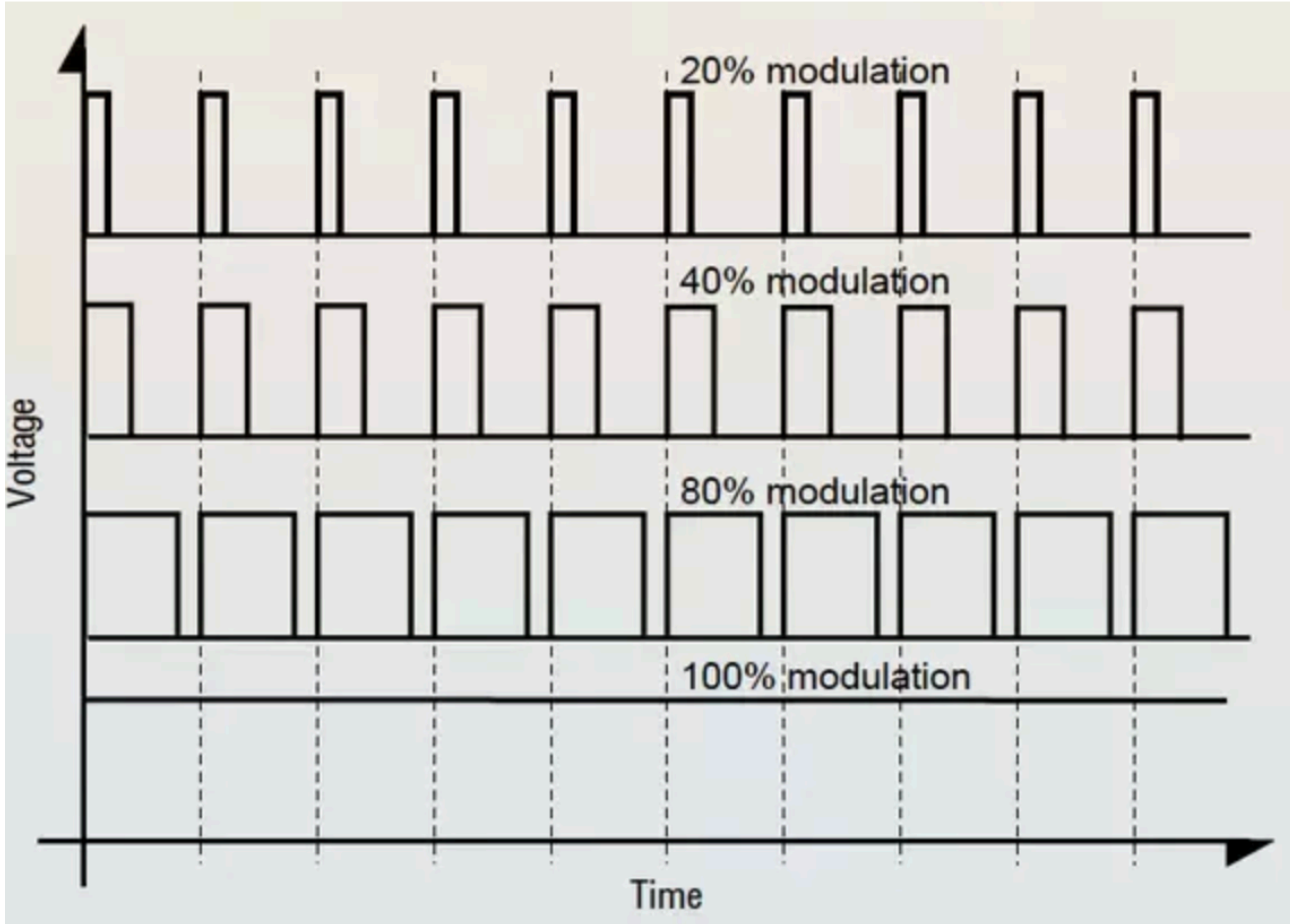
Part 3: DC motor

Speed control

Motor speed controlled by time the switches are closed



V_in



0.2 V_in

0.4 V_in

0.8 V_in

V_in

Part 3: DC motor

Check 3.Arduino-motor

Rough schedule

Wednesday

- 9:30 - 10:30 Part 3: Motors (contd.)
- 10:45 - 12:00 Part 4: Car construction
- 12 - 1 Lunch break
- 1 - 2:15 Part 4: Car construction
- 2:30 - 3:30 Part 4: Car programming
- 3:45 - 5:00 Part 4: Car programming

Part 4: Car construction and programming

Check 4.Arduino-car-project