

Microcontrollers and Robotic

Week 10: Simulator II



Fenerbahçe University



Professor & TAs

Prof: Dr. Vecdi Emre Levent

Office: 311

Email: emre.levent@fbu.edu.tr

TA: Arş. Gör. Uğur Özbalkan

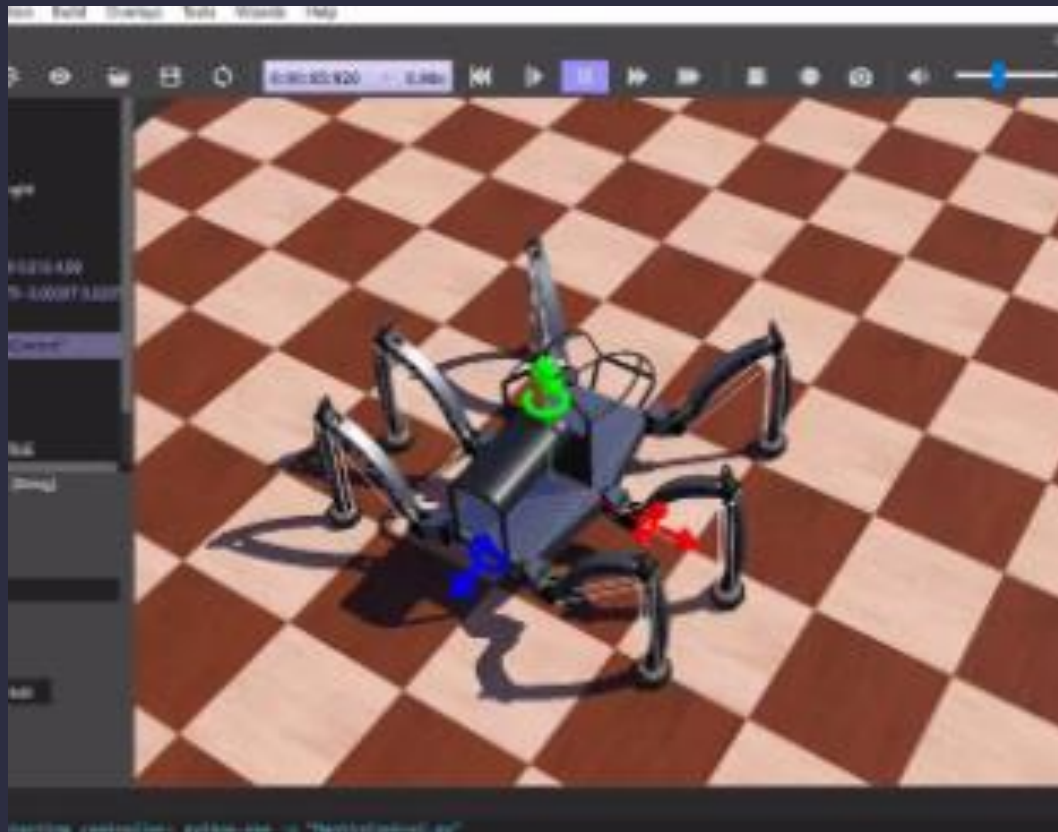
Office: 311

Email: ugur.ozbalkan@fbu.edu.tr

Simulator II

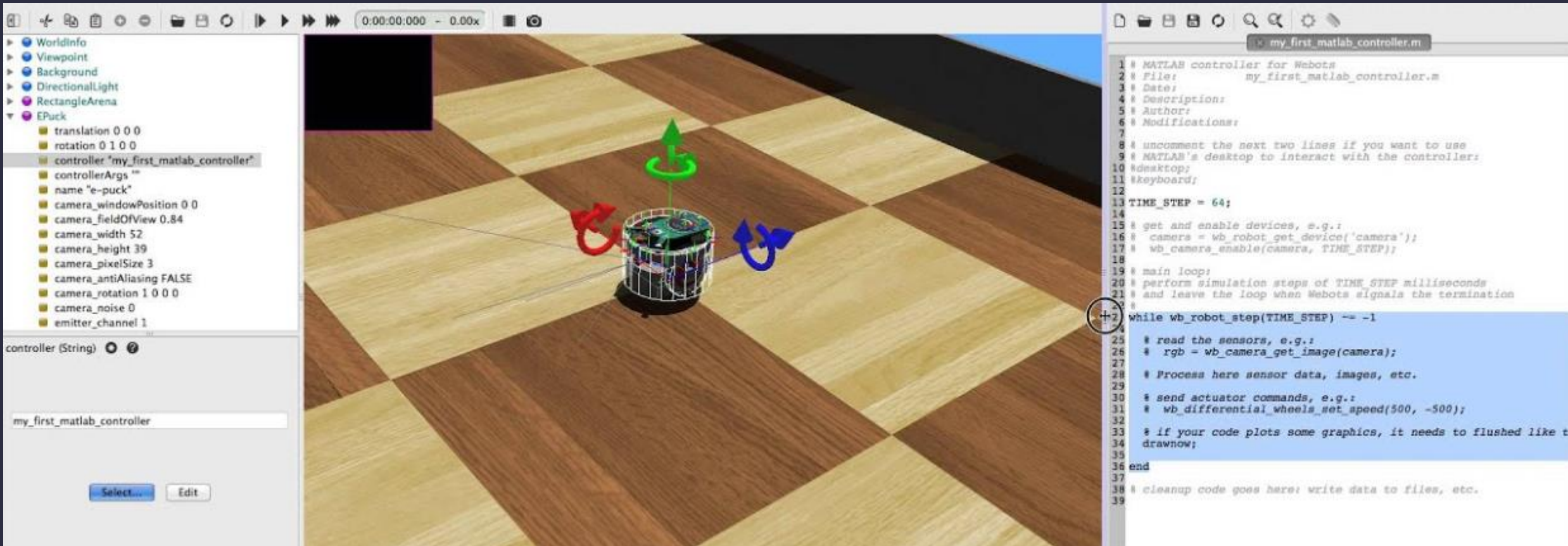
3D Simulators

WeBots



Simulator II

3D Simulators

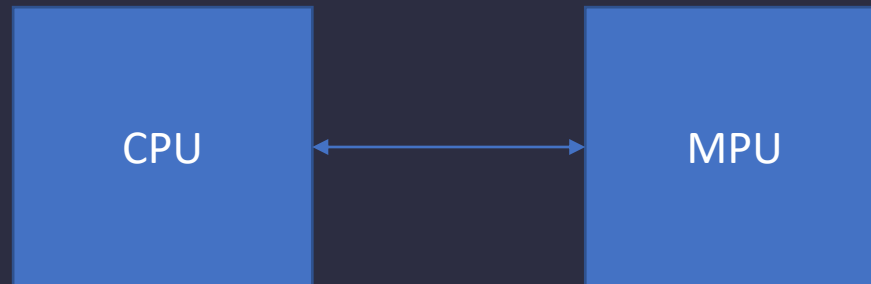


The image displays a 3D simulation environment. On the left, a control panel lists various parameters for the robot, including translation, rotation, camera settings, and emitter channel. The central 3D view shows a robot on a checkered floor with three colored arrows (red, green, blue) indicating movement directions. On the right, a MATLAB script titled 'my_first_matlab_controller.m' is shown, containing comments and code for controlling the robot's movement.

```
1 # MATLAB controller for Webots
2 # File: my_first_matlab_controller.m
3 # Date:
4 # Description:
5 # Author:
6 # Modifications:
7
8 # uncomment the next two lines if you want to use
9 # MATLAB's desktop to interact with the controllers:
10 #desktop;
11 #keyboard;
12
13 TIME_STEP = 64;
14
15 # get and enable devices, e.g.:
16 # camera = wb_robot_get_device('camera');
17 # wb_camera_enable(camera, TIME_STEP);
18
19 # main loop:
20 # perform simulation steps of TIME_STEP milliseconds
21 # and leave the loop when Webots signals the termination
22 #
23 while wb_robot_step(TIME_STEP) != -1
24
25 # read the sensors, e.g.:
26 # rgb = wb_camera_get_image(camera);
27
28 # Process here sensor data, images, etc.
29
30 # send actuator commands, e.g.:
31 # wb_differential_wheels_set_speed(500, -500);
32
33 # if your code plots some graphics, it needs to be flushed like this:
34 drawnow;
35
36 end
37
38 # cleanup code goes here: write data to files, etc.
39
```

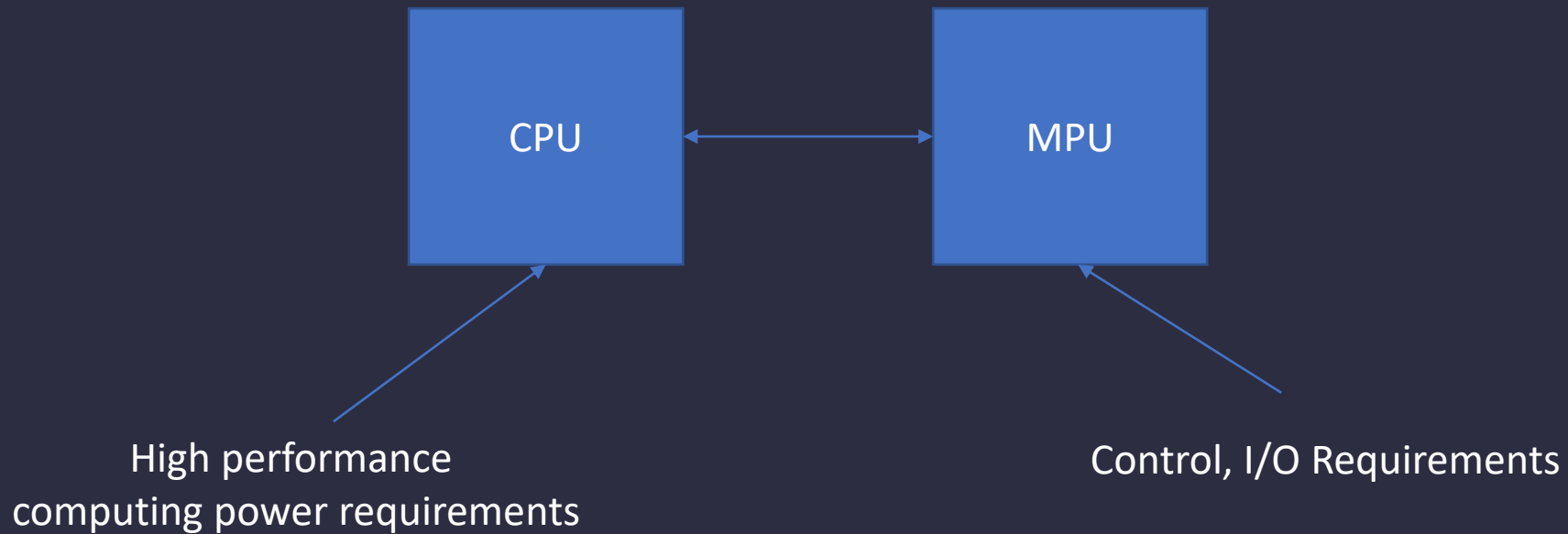
Simulator II

CPU/MPU Codesign Methodology



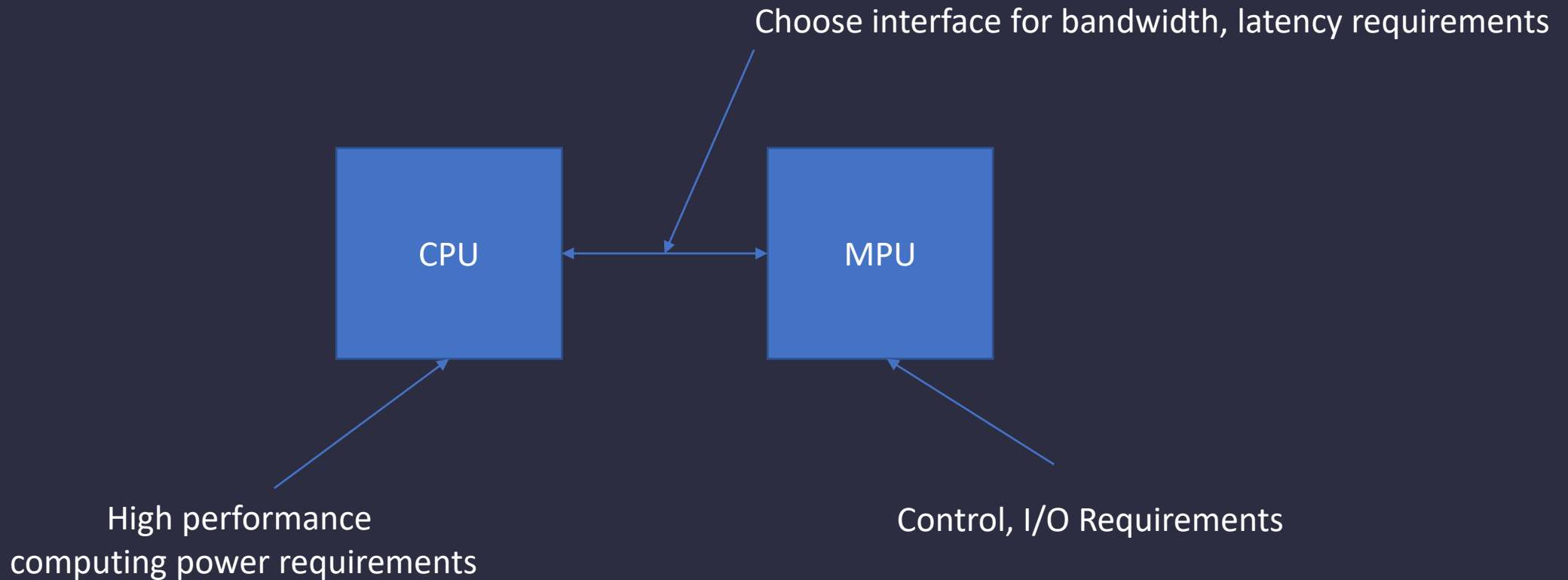
Simulator II

CPU/MPU Codesign Methodology



Simulator II

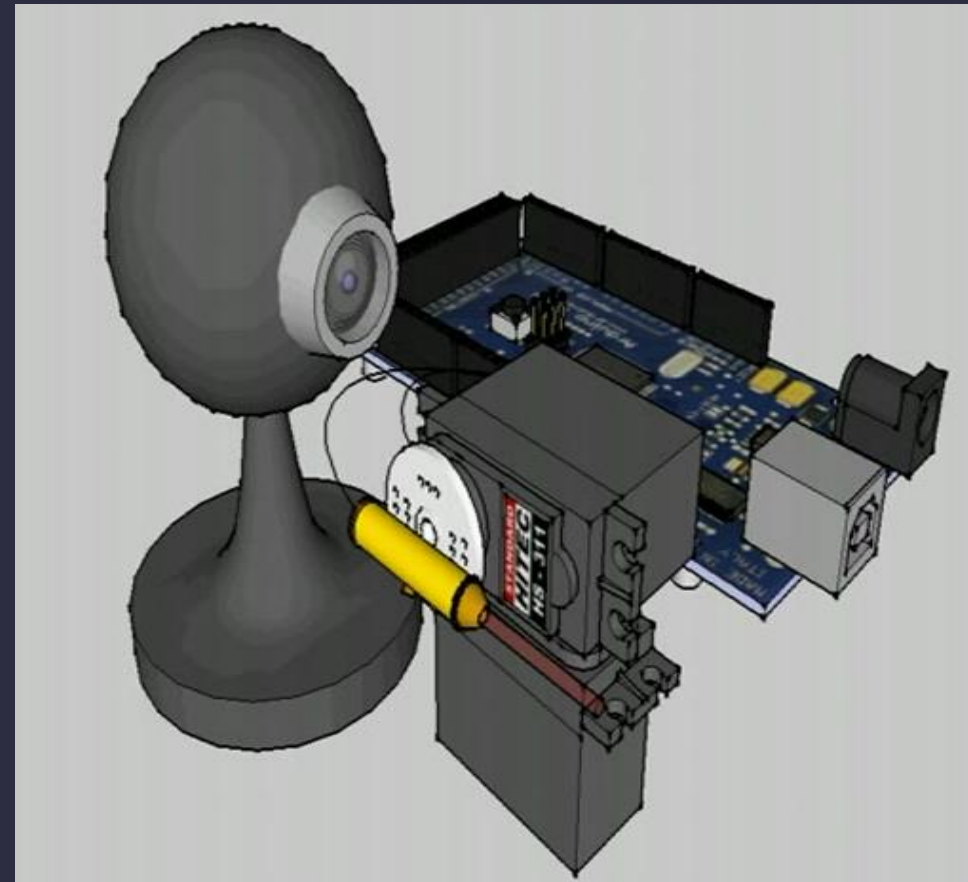
CPU/MPU Codesign Methodology



Simulator II

CPU/MPU Codesign Methodology

A use case, image processing and motor control



Simulator II

CPU/MPU Codesign Methodology

A use case, image processing and motor control

