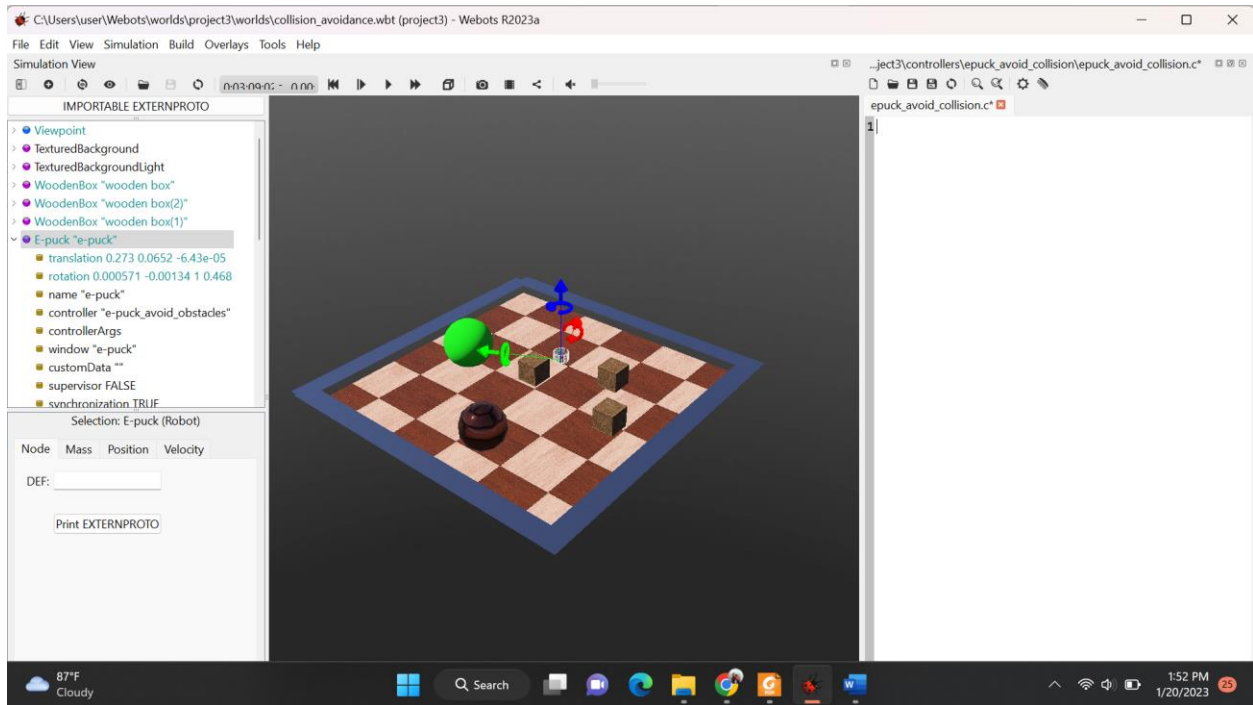


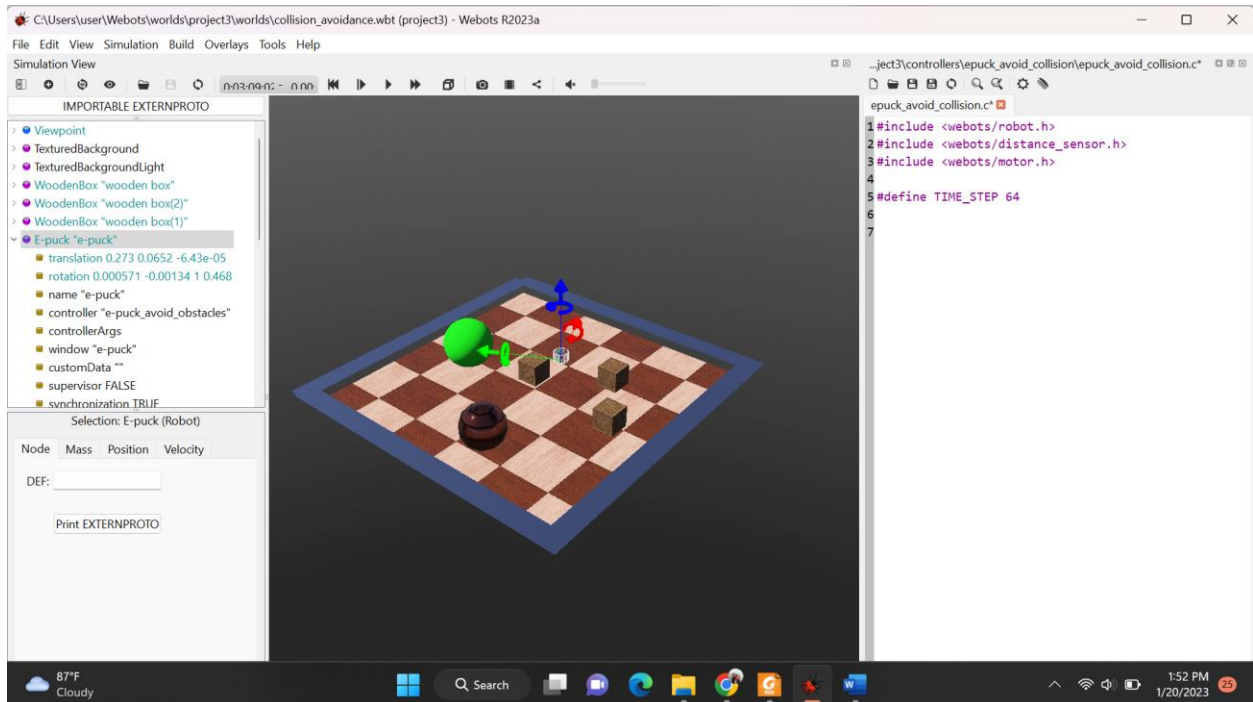
Tutorial 4: More about Controllers

Brilliant Friezka Aina (1103194186)

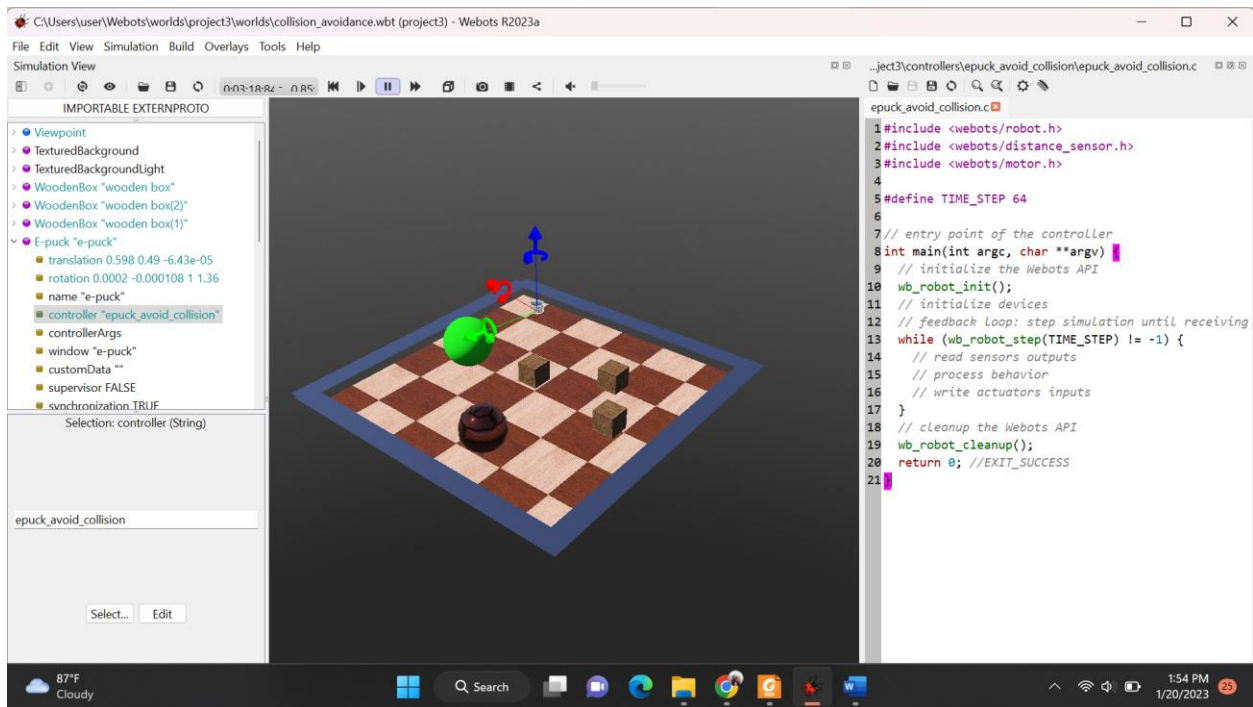
Hands-on#1



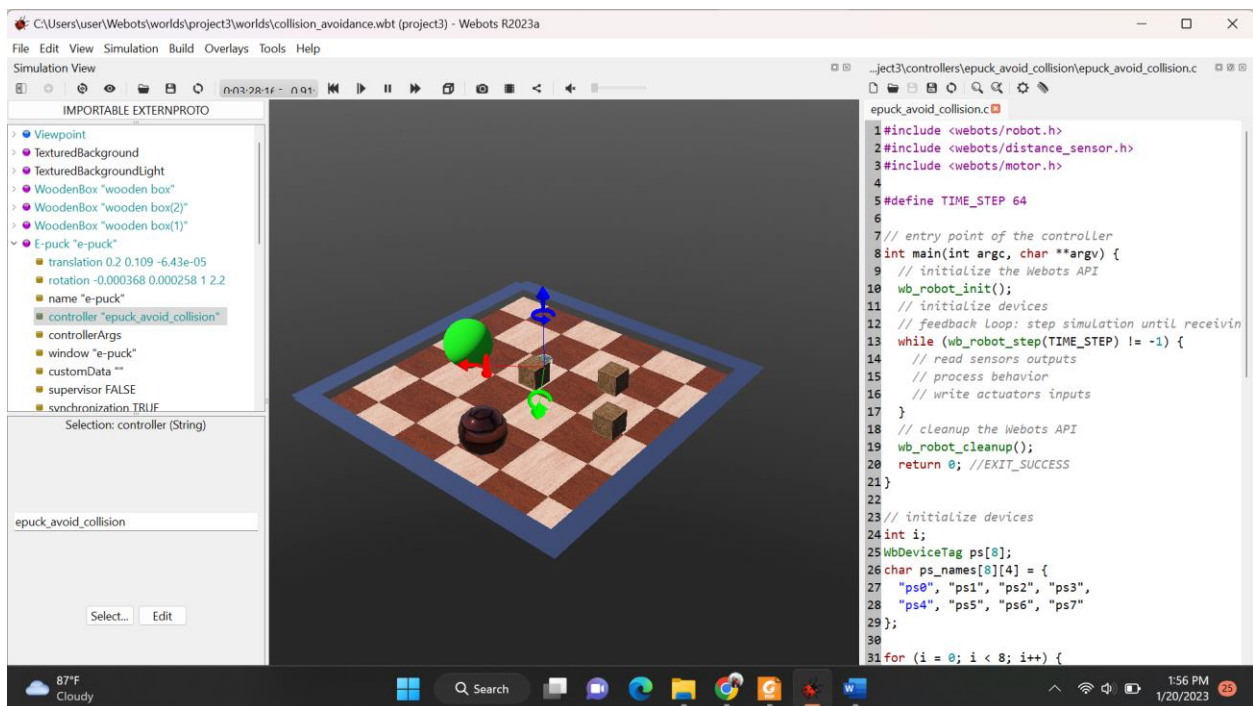
Hands-on#2



Hands-on#3



Hands-on#4



C:\Users\user\Webots\worlds\project3\worlds\collision_avoidance.wbt (project3) - Webots R2023a

File Edit View Simulation Build Overlays Tools Help

Simulation View

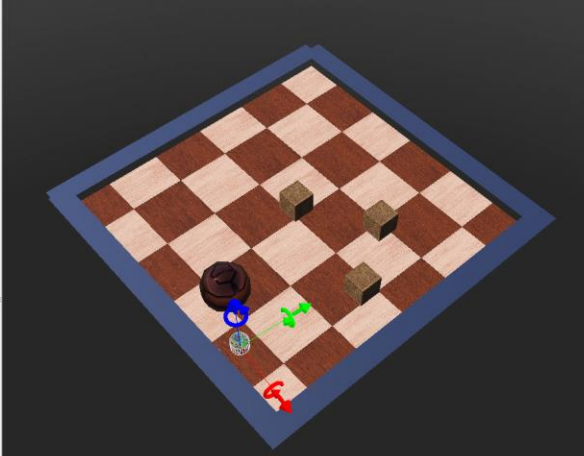
IMPORTABLE EXTERNPROTO

- > Viewpoint
- > TexturedBackground
- > TexturedBackgroundLight
- > WoodenBox "wooden box"
- > WoodenBox "wooden box(2)"
- > WoodenBox "wooden box(1)"
- > E-puck "e-puck"
 - translation -0.578 -0.336 -6.43e-05
 - rotation -0.000431 -0.000292 1 -1.66
 - name "e-puck"
 - controller "epuck_avoid_collision"
 - controllerArgs
 - window "e-puck"
 - customData ""
 - supervisor FALSE
 - synchronization TRUE

Selection: E-puck (Robot)

Node	Mass	Position	Velocity
DEF:			

Print EXTERNPROTO



epuck_avoid_collision.c

```
1#include <webots/robot.h>
2#include <webots/distance_sensor.h>
3#include <webots/motor.h>
4
5// time in [ms] of a simulation step
6#define TIME_STEP 64
7
8#define MAX_SPEED 6.28
9
10// entry point of the controller
11int main(int argc, char **argv) {
12    // initialize the Webots API
13    wb_robot_init();
14
15    // internal variables
16    int i;
17    WbDeviceTag ps[8];
18    char ps_names[8][4] = {
19        "ps0", "ps1", "ps2", "ps3",
20        "ps4", "ps5", "ps6", "ps7"
21    };
22
23    // initialize devices
24    for (i = 0; i < 8; i++) {
25        ps[i] = wb_robot_get_device(ps_names[i]);
26        wb_distance_sensor_enable(ps[i], TIME_STEP);
27    }
28
29    WbDeviceTag left_motor = wb_robot_get_device("l");
30    WbDeviceTag right_motor = wb_robot_get_device("r");
31    wb_motor_set_position(left_motor, INFINITY);
```