ARDUINO code

```
#if defined(ARDUINO) && ARDUINO >= 100
  #include "Arduino.h"
#else
  #include <WProgram.h>
#endif

#include <Servo.h>
#include <ros.h>
#include <std_msgs/UInt16.h>
#include <sensor_msgs/JointState.h>

ros::NodeHandle  nh;
Servo gripper;
Servo wrist;
Servo elbow;
Servo shoulder;
Servo base;

double base_angle=90;
double shoulder_angle=90;
double elbow_angle=90;
double wrist_angle=90;

double prev_base = 0;
double prev_shoulder = 0;
double prev_elbow = 0;
double prev_wrist = 0;
```

```
int gripperState = 0;
int positionChanged = 0;

void servo_cb(const sensor_msgs::JointState& cmd_msg){
  base_angle=radiansToDegrees(cmd_msg.position[0]);
  shoulder_angle=radiansToDegrees(cmd_msg.position[1]);
  elbow_angle=radiansToDegrees(cmd_msg.position[2]);
  wrist_angle=radiansToDegrees(cmd_msg.position[3]);

  base.write(base_angle);
  shoulder.write(shoulder_angle);
  elbow.write(elbow_angle);
  wrist.write(wrist_angle);

  if (prev_base==base_angle && prev_shoulder==shoulder_angle && prev_elbow==elbow_angle && prev_wrist==wrist_angle && positionChanged==0)
  {
    if (gripperState==0)
    {
      gripper.write(60);
      gripperState = 1;
    }
    else if (gripperState==1)
    {
      gripper.write(0);
      gripperState = 0;
    }
    positionChanged = 1;
  }
  else if ((prev_base!=base_angle || prev_shoulder!=shoulder_angle || prev_elbow!=elbow_angle ||
prev_wrist!=wrist_angle) && positionChanged==1)
  {
    positionChanged = 0;
  }

  prev_base = base_angle;
  prev_shoulder = shoulder_angle;
  prev_elbow = elbow_angle;
  prev_wrist = wrist_angle;
}


ros::Subscriber<sensor_msgs::JointState> sub("joint_states", servo_cb);

void setup(){
  nh.getHardware()->setBaud(115200);
  nh.initNode();
  nh.subscribe(sub);

  base.attach(8);
  shoulder.attach(9);
  elbow.attach(10);
  wrist.attach(11);
  gripper.attach(12);

  delay(1);
  base.write(90);
  shoulder.write(90);
  elbow.write(90);

  wrist.write(90);

  gripper.write(0);

}


void loop(){

  nh.spinOnce();

}


double radiansToDegrees(float position_radians)

{


  position_radians = position_radians + 1.6;


  return position_radians * 55.2313;


}
```