**JEREMY GERALD PRAWIRA** 001202000003
**ANGELINA YULISA WANEY** 001202000076
**MULYA FAJAR NINGSIH** 001202000101
Robotics
Mid-Semester Report

## I. RULES

- Human Arms are capable of a wide variety of functions, including gross and fine motor movements. Gross motor movements allow us to pick up large objects or perform heavy labor. Fine motor movements enable us to perform delicate tasks, such as holding small objects or performing detailed work.
- Robot Arms also have the same functions as Human arms. **Use the available components and Build a Robot Arms** that have as many functions as possible, you need to mention all the detailed Arm functions presented in the Brief Description part of the Report.
- **Write a Report that has the following Format:**
  1) Brief Project Description (Mention all Robot Arm functions presented).
  2) Project Requirements (Hardware and Software)
  3) Photo images showing the robotic arm in its entirety and additional photos showing details of the function of the arm when needed.
  4) Circuit Diagram using Fritzing (Fritzing.org)
  5) Sketch (List of Code) with additional comments for important functions.

## II. BRIEF PROJECT DESCRIPTION

The project that we are doing is about how to create a functional arm robot that is from Arduino. There will be two functions existed on this arm robot. The first one is an automated function, where the robot itself will detect an object around it’s environment, and grab the object. Later, the robot will take it to another place. While the second function is the manual function, where in order to take the object around it’s environment and place it on another type of place, the robot should be controlled by buttons that is existed.

## III. PROJECT REQUIREMENT

Software: Fritzing, Arduino IDE

Hardware: Arduino Uno R3, Small breadboard, Push buttons (5pcs), Jumper wires, Ultrasonic sensor, Robotic arm, Motor servos (4pcs)

## IV. ATTACHMENT

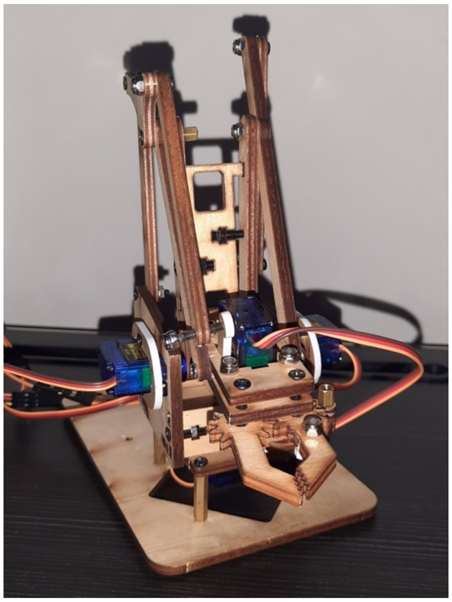


*Figure 1 – All Servo*

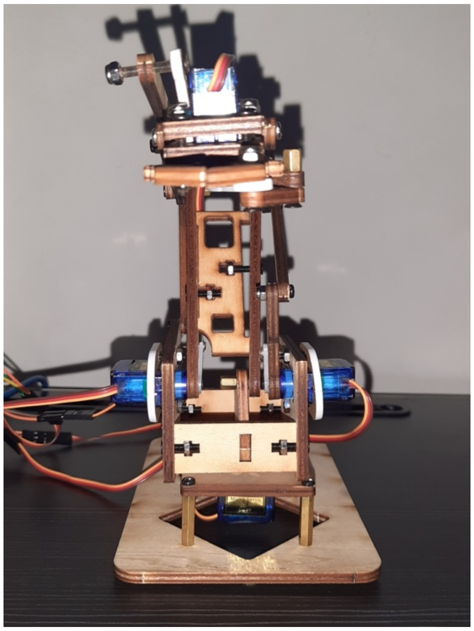


*Figure 2 – Bottom Servo*

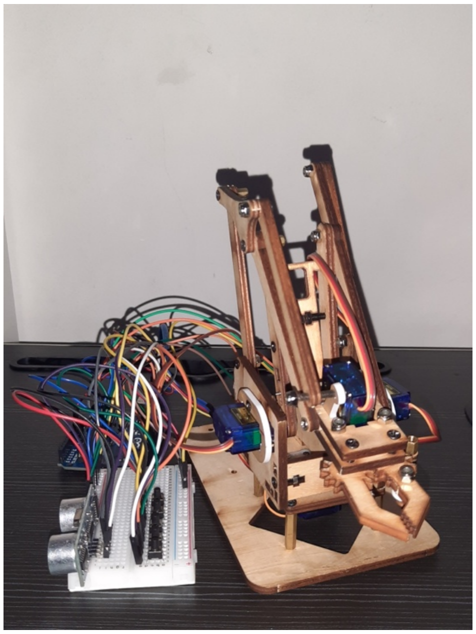


*Figure 3 – Circuit*

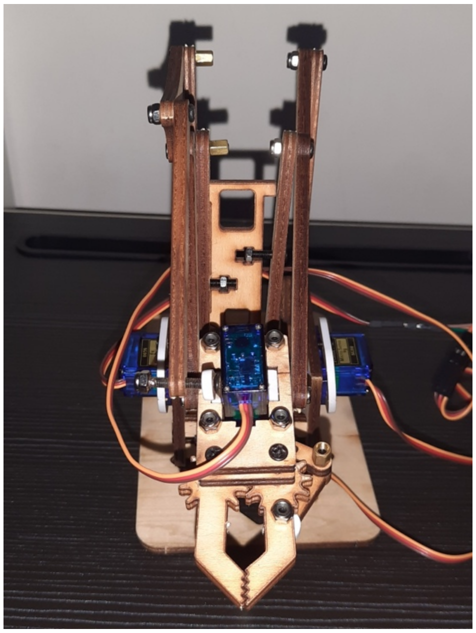


*Figure 4 – Claw Servo*

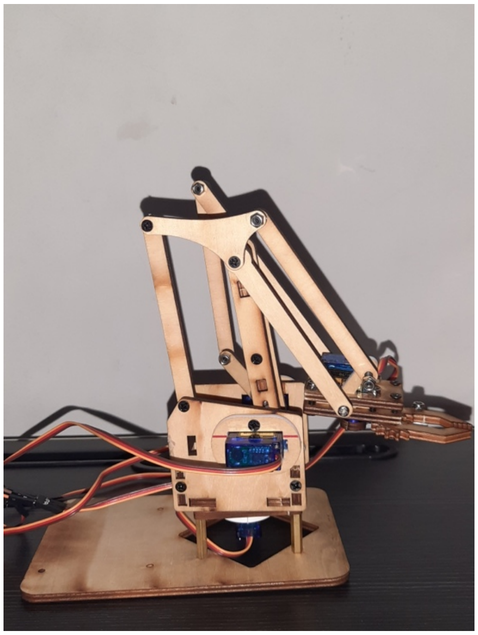


*Figure 5 – Left Servo*

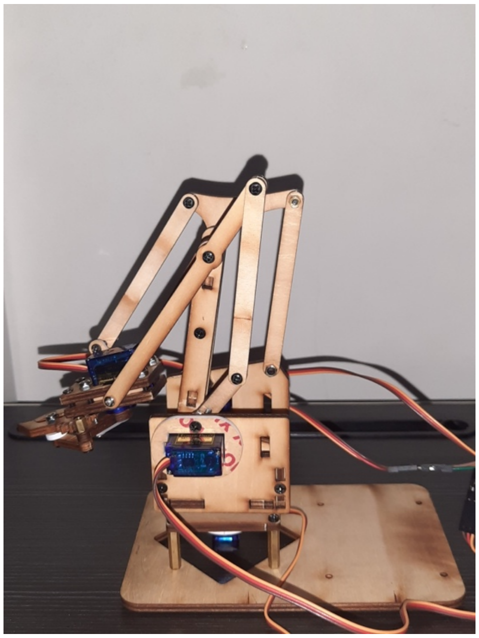
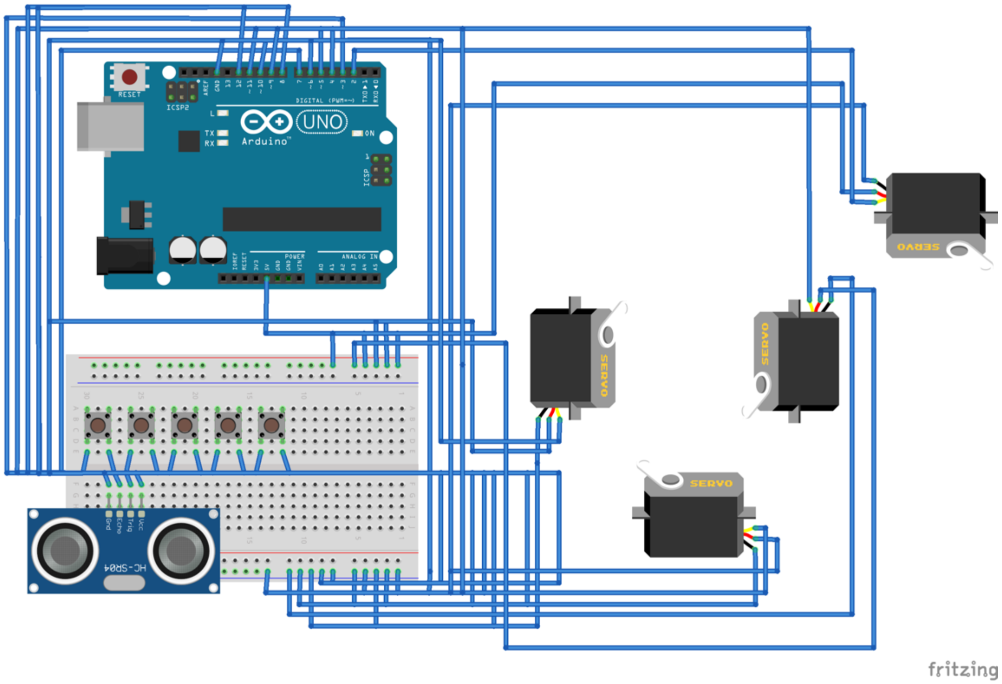
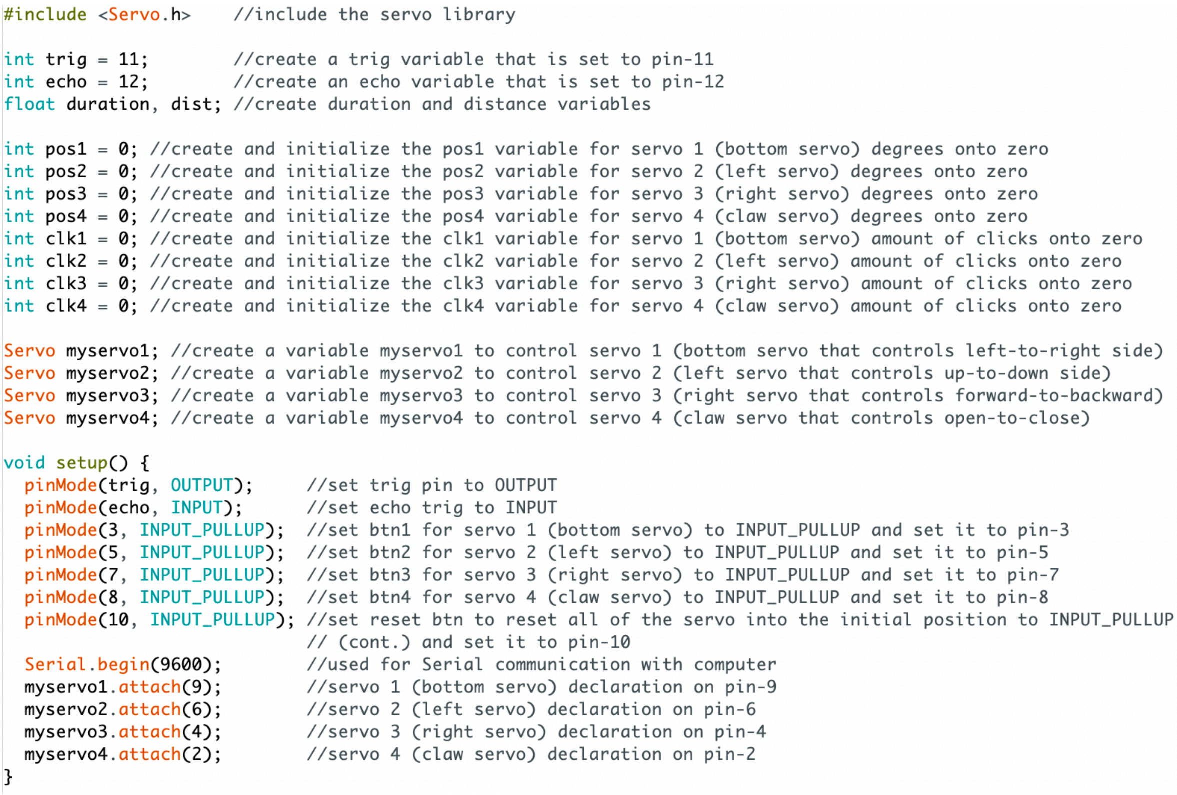


*Figure 6 – Right Servo*

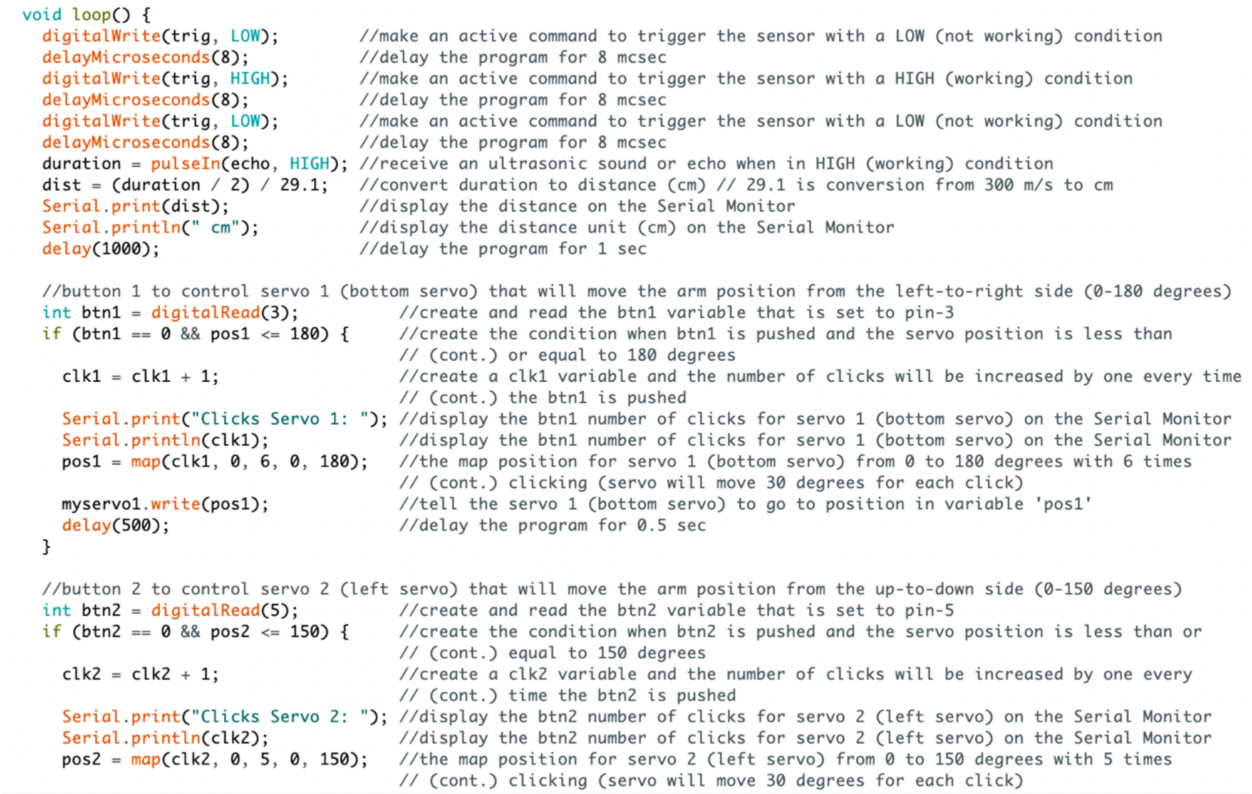
## V. CIRCUIT DIAGRAM



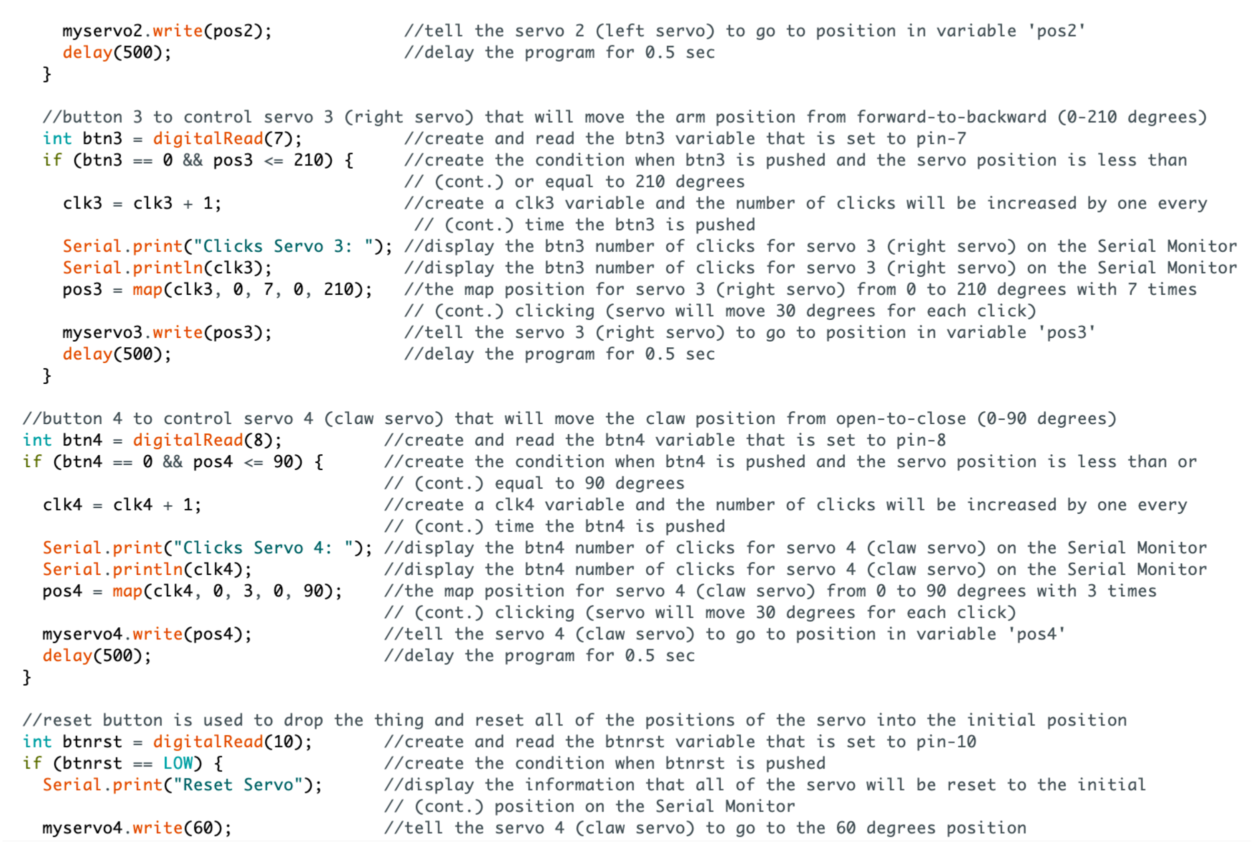
## VI. SKETCH



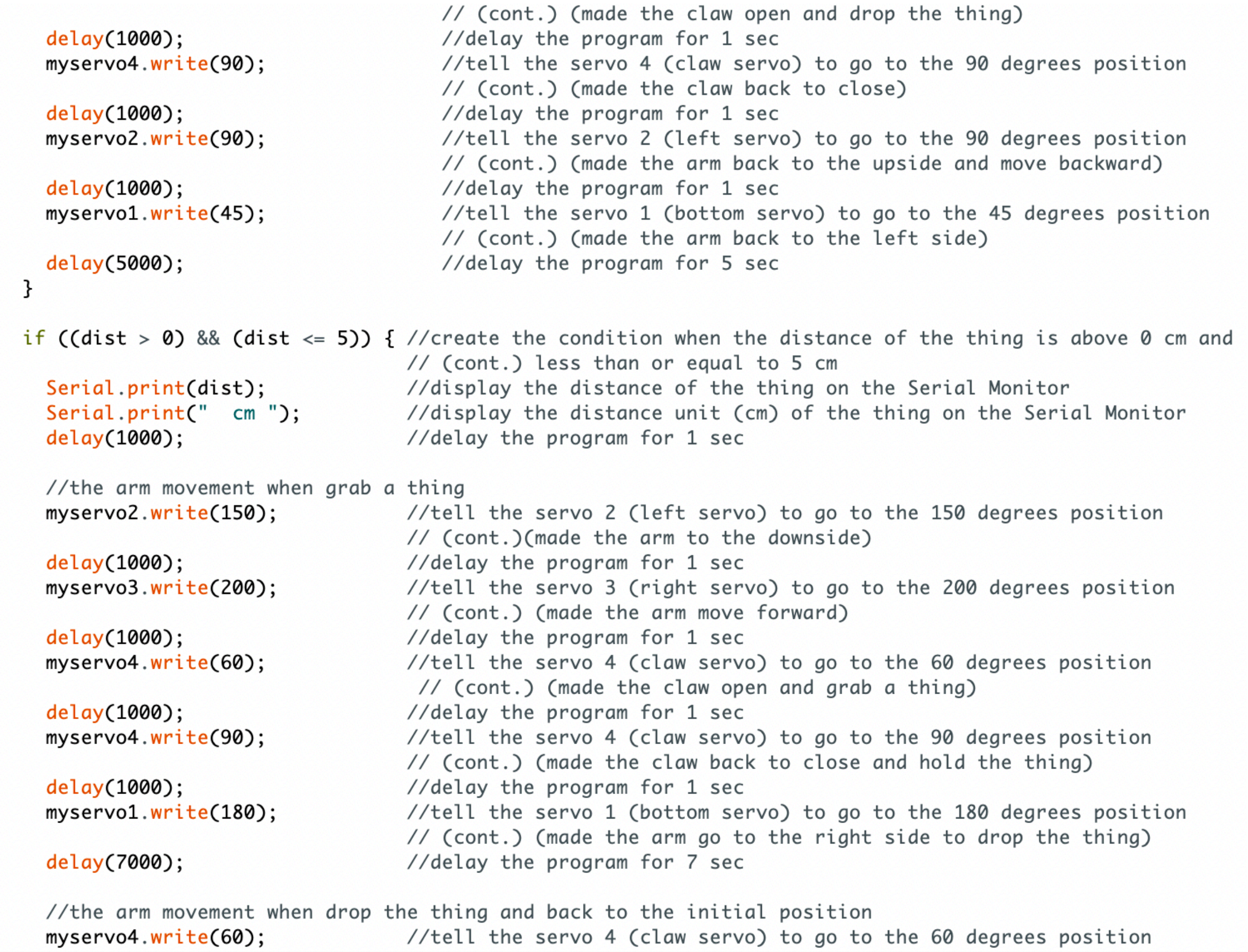
```
#include <Servo.h>    //include the servo library

int trig = 11;        //create a trig variable that is set to pin-11
int echo = 12;        //create an echo variable that is set to pin-12
float duration, dist; //create duration and distance variables

int pos1 = 0; //create and initialize the pos1 variable for servo 1 (bottom servo) degrees onto zero
int pos2 = 0; //create and initialize the pos2 variable for servo 2 (left servo) degrees onto zero
int pos3 = 0; //create and initialize the pos3 variable for servo 3 (right servo) degrees onto zero
int pos4 = 0; //create and initialize the pos4 variable for servo 4 (claw servo) degrees onto zero
int clk1 = 0; //create and initialize the clk1 variable for servo 1 (bottom servo) amount of clicks onto zero
int clk2 = 0; //create and initialize the clk2 variable for servo 2 (left servo) amount of clicks onto zero
int clk3 = 0; //create and initialize the clk3 variable for servo 3 (right servo) amount of clicks onto zero
int clk4 = 0; //create and initialize the clk4 variable for servo 4 (claw servo) amount of clicks onto zero

Servo myservo1; //create a variable myservo1 to control servo 1 (bottom servo that controls left-to-right side)
Servo myservo2; //create a variable myservo2 to control servo 2 (left servo that controls up-to-down side)
Servo myservo3; //create a variable myservo3 to control servo 3 (right servo that controls forward-to-backward)
Servo myservo4; //create a variable myservo4 to control servo 4 (claw servo that controls open-to-close)

void setup() {
  pinMode(trig, OUTPUT);     //set trig pin to OUTPUT
  pinMode(echo, INPUT);      //set echo trig to INPUT
  pinMode(3, INPUT_PULLUP);  //set btn1 for servo 1 (bottom servo) to INPUT_PULLUP and set it to pin-3
  pinMode(5, INPUT_PULLUP);  //set btn2 for servo 2 (left servo) to INPUT_PULLUP and set it to pin-5
  pinMode(7, INPUT_PULLUP);  //set btn3 for servo 3 (right servo) to INPUT_PULLUP and set it to pin-7
  pinMode(8, INPUT_PULLUP);  //set btn4 for servo 4 (claw servo) to INPUT_PULLUP and set it to pin-8
  pinMode(10, INPUT_PULLUP); //set reset btn to reset all of the servo into the initial position to INPUT_PULLUP
                             // (cont.) and set it to pin-10
  Serial.begin(9600);        //used for Serial communication with computer
  myservo1.attach(9);        //servo 1 (bottom servo) declaration on pin-9
  myservo2.attach(6);        //servo 2 (left servo) declaration on pin-6
  myservo3.attach(4);        //servo 3 (right servo) declaration on pin-4
  myservo4.attach(2);        //servo 4 (claw servo) declaration on pin-2
}
```



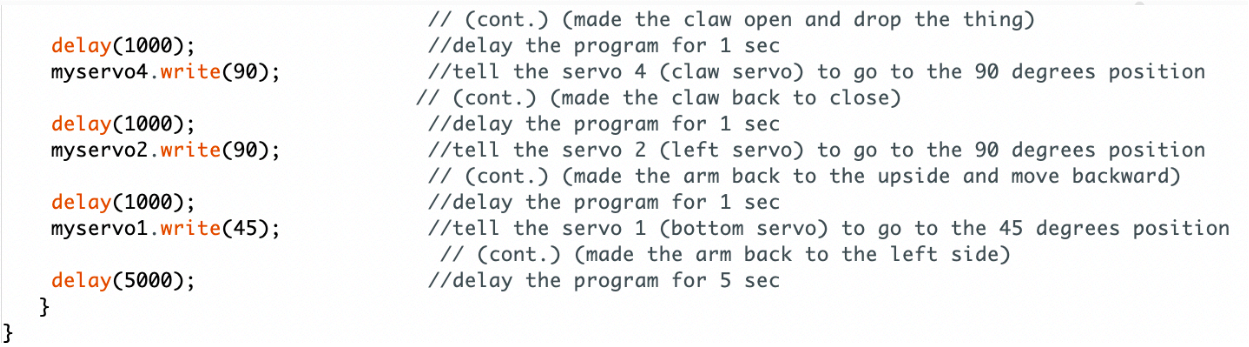
```
void loop() {
  digitalWrite(trig, LOW);        //make an active command to trigger the sensor with a LOW (not working) condition
  delayMicroseconds(8);           //delay the program for 8 mcsec
  digitalWrite(trig, HIGH);       //make an active command to trigger the sensor with a HIGH (working) condition
  delayMicroseconds(8);           //delay the program for 8 mcsec
  digitalWrite(trig, LOW);        //make an active command to trigger the sensor with a LOW (not working) condition
  delayMicroseconds(8);           //delay the program for 8 mcsec
  duration = pulseIn(echo, HIGH); //receive an ultrasonic sound or echo when in HIGH (working) condition
  dist = (duration / 2) / 29.1;   //convert duration to distance (cm) // 29.1 is conversion from 300 m/s to cm
  Serial.print(dist);             //display the distance on the Serial Monitor
  Serial.println(" cm");          //display the distance unit (cm) on the Serial Monitor
  delay(1000);                    //delay the program for 1 sec

  //button 1 to control servo 1 (bottom servo) that will move the arm position from the left-to-right side (0-180 degrees)
  int btn1 = digitalRead(3);          //create and read the btn1 variable that is set to pin-3
  if (btn1 == 0 && pos1 <= 180) {     //create the condition when btn1 is pushed and the servo position is less than
                                      // (cont.) or equal to 180 degrees
    clk1 = clk1 + 1;                  //create a clk1 variable and the number of clicks will be increased by one every time
                                      // (cont.) the btn1 is pushed
    Serial.print("Clicks Servo 1: "); //display the btn1 number of clicks for servo 1 (bottom servo) on the Serial Monitor
    Serial.println(clk1);             //display the btn1 number of clicks for servo 1 (bottom servo) on the Serial Monitor
    pos1 = map(clk1, 0, 6, 0, 180);   //the map position for servo 1 (bottom servo) from 0 to 180 degrees with 6 times
                                      // (cont.) clicking (servo will move 30 degrees for each click)
    myservo1.write(pos1);             //tell the servo 1 (bottom servo) to go to position in variable 'pos1'
    delay(500);                       //delay the program for 0.5 sec
  }

  //button 2 to control servo 2 (left servo) that will move the arm position from the up-to-down side (0-150 degrees)
  int btn2 = digitalRead(5);          //create and read the btn2 variable that is set to pin-5
  if (btn2 == 0 && pos2 <= 150) {     //create the condition when btn2 is pushed and the servo position is less than or
                                      // (cont.) equal to 150 degrees
    clk2 = clk2 + 1;                  //create a clk2 variable and the number of clicks will be increased by one every
                                      // (cont.) time the btn2 is pushed
    Serial.print("Clicks Servo 2: "); //display the btn2 number of clicks for servo 2 (left servo) on the Serial Monitor
    Serial.println(clk2);             //display the btn2 number of clicks for servo 2 (left servo) on the Serial Monitor
    pos2 = map(clk2, 0, 5, 0, 150);   //the map position for servo 2 (left servo) from 0 to 150 degrees with 5 times
                                      // (cont.) clicking (servo will move 30 degrees for each click)
```



```
    myservo2.write(pos2);              //tell the servo 2 (left servo) to go to position in variable 'pos2'
    delay(500);                        //delay the program for 0.5 sec
  }

  //button 3 to control servo 3 (right servo) that will move the arm position from forward-to-backward (0-210 degrees)
  int btn3 = digitalRead(7);           //create and read the btn3 variable that is set to pin-7
  if (btn3 == 0 && pos3 <= 210) {      //create the condition when btn3 is pushed and the servo position is less than
                                       // (cont.) or equal to 210 degrees
    clk3 = clk3 + 1;                   //create a clk3 variable and the number of clicks will be increased by one every
                                        // (cont.) time the btn3 is pushed
    Serial.print("Clicks Servo 3: ");  //display the btn3 number of clicks for servo 3 (right servo) on the Serial Monitor
    Serial.println(clk3);              //display the btn3 number of clicks for servo 3 (right servo) on the Serial Monitor
    pos3 = map(clk3, 0, 7, 0, 210);    //the map position for servo 3 (right servo) from 0 to 210 degrees with 7 times
                                       // (cont.) clicking (servo will move 30 degrees for each click)
    myservo3.write(pos3);              //tell the servo 3 (right servo) to go to position in variable 'pos3'
    delay(500);                        //delay the program for 0.5 sec
  }

//button 4 to control servo 4 (claw servo) that will move the claw position from open-to-close (0-90 degrees)
int btn4 = digitalRead(8);          //create and read the btn4 variable that is set to pin-8
if (btn4 == 0 && pos4 <= 90) {      //create the condition when btn4 is pushed and the servo position is less than or
                                    // (cont.) equal to 90 degrees
  clk4 = clk4 + 1;                  //create a clk4 variable and the number of clicks will be increased by one every
                                    // (cont.) time the btn4 is pushed
  Serial.print("Clicks Servo 4: "); //display the btn4 number of clicks for servo 4 (claw servo) on the Serial Monitor
  Serial.println(clk4);             //display the btn4 number of clicks for servo 4 (claw servo) on the Serial Monitor
  pos4 = map(clk4, 0, 3, 0, 90);    //the map position for servo 4 (claw servo) from 0 to 90 degrees with 3 times
                                    // (cont.) clicking (servo will move 30 degrees for each click)
  myservo4.write(pos4);             //tell the servo 4 (claw servo) to go to position in variable 'pos4'
  delay(500);                       //delay the program for 0.5 sec
}

//reset button is used to drop the thing and reset all of the positions of the servo into the initial position
int btnrst = digitalRead(10);       //create and read the btnrst variable that is set to pin-10
if (btnrst == LOW) {                //create the condition when btnrst is pushed
  Serial.print("Reset Servo");      //display the information that all of the servo will be reset to the initial
                                    // (cont.) position on the Serial Monitor
  myservo4.write(60);               //tell the servo 4 (claw servo) to go to the 60 degrees position
```



```
                                    // (cont.) (made the claw open and drop the thing)
    delay(1000);                    //delay the program for 1 sec
    myservo4.write(90);             //tell the servo 4 (claw servo) to go to the 90 degrees position
                                    // (cont.) (made the claw back to close)
    delay(1000);                    //delay the program for 1 sec
    myservo2.write(90);             //tell the servo 2 (left servo) to go to the 90 degrees position
                                    // (cont.) (made the arm back to the upside and move backward)
    delay(1000);                    //delay the program for 1 sec
    myservo1.write(45);             //tell the servo 1 (bottom servo) to go to the 45 degrees position
                                    // (cont.) (made the arm back to the left side)
    delay(5000);                    //delay the program for 5 sec
  }

  if ((dist > 0) && (dist <= 5)) { //create the condition when the distance of the thing is above 0 cm and
                                 // (cont.) less than or equal to 5 cm
    Serial.print(dist);          //display the distance of the thing on the Serial Monitor
    Serial.print("  cm ");       //display the distance unit (cm) of the thing on the Serial Monitor
    delay(1000);                 //delay the program for 1 sec

    //the arm movement when grab a thing
    myservo2.write(150);         //tell the servo 2 (left servo) to go to the 150 degrees position
                                 // (cont.)(made the arm to the downside)
    delay(1000);                 //delay the program for 1 sec
    myservo3.write(200);         //tell the servo 3 (right servo) to go to the 200 degrees position
                                 // (cont.) (made the arm move forward)
    delay(1000);                 //delay the program for 1 sec
    myservo4.write(60);          //tell the servo 4 (claw servo) to go to the 60 degrees position
                                  // (cont.) (made the claw open and grab a thing)
    delay(1000);                 //delay the program for 1 sec
    myservo4.write(90);          //tell the servo 4 (claw servo) to go to the 90 degrees position
                                 // (cont.) (made the claw back to close and hold the thing)
    delay(1000);                 //delay the program for 1 sec
    myservo1.write(180);         //tell the servo 1 (bottom servo) to go to the 180 degrees position
                                 // (cont.) (made the arm go to the right side to drop the thing)
    delay(7000);                 //delay the program for 7 sec

    //the arm movement when drop the thing and back to the initial position
    myservo4.write(60);          //tell the servo 4 (claw servo) to go to the 60 degrees position
```



```
                                 // (cont.) (made the claw open and drop the thing)
    delay(1000);                 //delay the program for 1 sec
    myservo4.write(90);          //tell the servo 4 (claw servo) to go to the 90 degrees position
                                // (cont.) (made the claw back to close)
    delay(1000);                 //delay the program for 1 sec
    myservo2.write(90);          //tell the servo 2 (left servo) to go to the 90 degrees position
                                 // (cont.) (made the arm back to the upside and move backward)
    delay(1000);                 //delay the program for 1 sec
    myservo1.write(45);          //tell the servo 1 (bottom servo) to go to the 45 degrees position
                                  // (cont.) (made the arm back to the left side)
    delay(5000);                 //delay the program for 5 sec
  }
}
```

## VII. VIDEO

Link:
https://drive.google.com/drive/folders/172G4znyTgcc3gimoMMqFVoQ3yvF2MzUM?usp=sharing