Robotic Arm:

Controlled Wirelessly

Department of Electrical and Electronic Engineering

Submitted To:

- Dr. Md. Farhad Hossain

  Professor

  Department of Electrical & Electronic Engineering

- Ramit Datta

  Lecturer

  Department of Electrical & Electronic Engineering

Submitted By: Group 08

Fatin Ishrak Rakin	1806183
Omar Faruk Siyam	1806159
Khondokar Zahin	1808161
Md. Jawadul Karim	1806162

Table of Contents

INTRODUCTION

Robotic hands that appear and act like human hands are constructed in a way that makes them very similar to real thing. This simple human like hand uses multiple motors with long tendon roped through the fingers to close and relax the hand and move the fingers independently. Robots are used in science and industry to replace or entertain humans and although they do not have to look like us nor have to perform task in human manner, many of them do. In fact, some of the most popular in the industry are robot manipulators which resemble the human arm.

In this project our main objectives are:

✓ To construct a wirelessly controlled robotic arm

✓To understand wirelessly controlled system

✓To do some fixed movements repeatedly

✓To assess whether this idea can be a possible future for industrial developing countries

Robotics arms are designed to manipulate and transport parts, tools or special manufacturing elements through reprogrammable movements to perform a variety of tasks. This is why they are commonly used in industrial environments.

Robotic arm is a mechanical arm like structured robot that is capable of lifting an object and keeping it in desired place within the limit of arms. As manual labor is being reduced at big scale industries and factories to increase efficiency and gain profit by installing robots that can do repetitive works.

A. METHODOLOGIES

Define problem & Identify objective:

Industry 4.0, also known as Connected Industry, whereby people and robots can work together safely and exchange knowledge to refine their operations, thereby improving decision-making. Mobile robots and mobile manipulators provide the modular and automated industrial automation required to build Smart Factories. Robotic arms can be used to automate the process of placing goods or products onto pallets. By automating the process, palletizing becomes more accurate, cost-effective, and predictable. The use of robotic arms also frees human workers from performing tasks that present a risk of bodily injury. Therefore, we select robotic arm as our project and we demonstrated this through microcontroller wirelessly.

Research & Brainstorm

First of all, we select appropriate components like servo motors, Arduino board, Bluetooth module etc. for construction. A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a

suitable motor coupled to a sensor for position feedback. That's why we select servo motor for our project purpose. When Bluetooth-enabled devices are close to each other, they automatically detect each other. So, we use HC 05 Bluetooth module to control the arm wirelessly. We use Arduino uno as a processing unit. We estimated that our project requires 2A DC supply. For this we use a mobile charger (5v-2) as an external power supply.

Build a Prototype:

A prototype is an early sample, model, or release of a product built to test a concept or process. It is a term used in a variety of contexts, including semantics, design, electronics, and software programming. We took the following steps to design our prototype:

- We connected the Bluetooth module with the Arduino using serial port
- We connected the servo motors' control pin with the Arduino using digital I/O pins
- We supplied external power to the motor through breadboard
- We wrote an Arduino code for testing
- We designed an app using MIT App Inventor. This app building interface sets connections between our mobile device with Bluetooth module

Build Robotic Arm:

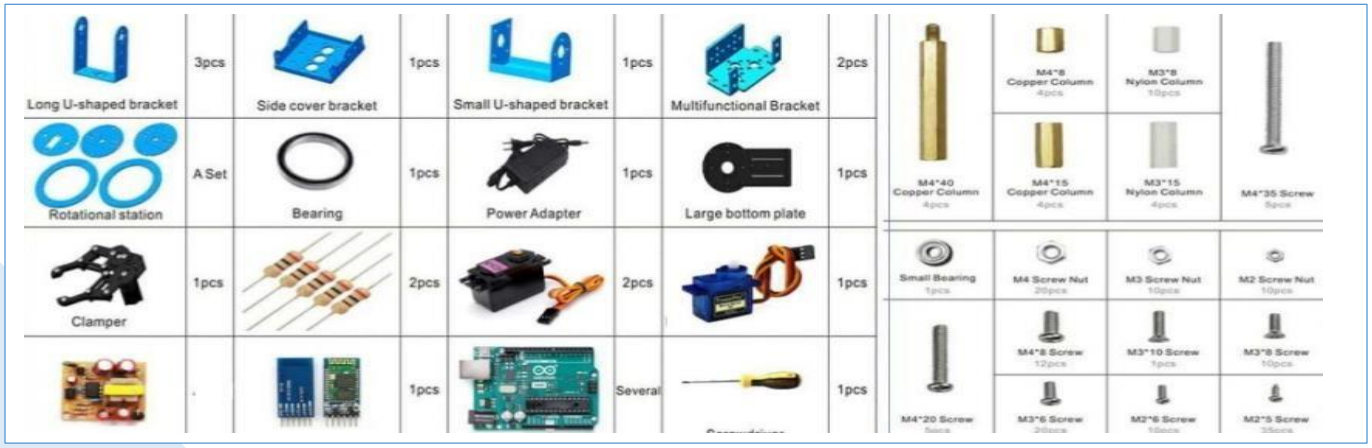
A robotic arm is a type of mechanical arm, usually programmable, with similar functions to a human arm; the arm may be the sum total of the mechanism or may be part of a more complex robot.

- Using a 3D printer, we printed the frame of our robotic arm parts by parts
- We assembled the base, wrist, claw mechanism separately. Then we put together all the separate parts &motors into a unified frame.

B. PARTS OF THE PROJECT

Hardware Equipment

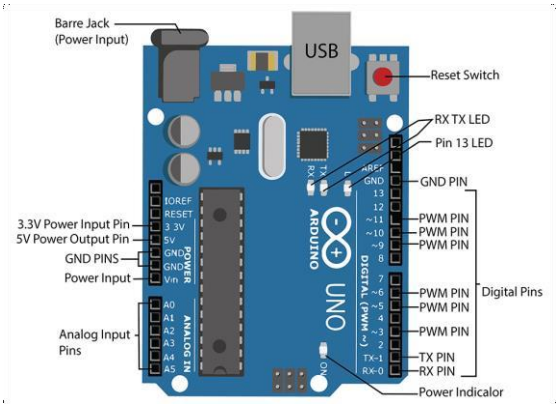
The following equipment we used:



Hardware components:

Arduino Uno:

Arduino Uno is a microcontroller board based on the ATmega328P It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.



Bluetooth Module:

HC-05 Bluetooth Module is easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Its communication is via serial communication which makes an easy way to interface with controller.

Servo Motor:

SG90 Micro Servo Motor:

Micro Servo Motor SG90 is a tiny and lightweight server motor with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. We can use any servo code, hardware or library to control these servos. The SG90 Micro is a servo motor with a maximum stall torque of 2.5kg/cm.

MG996R Servo Motor:

The MG996R is a metal gear servo motor with a maximum stall torque of 11 kg/cm. Like other RC servos the motor rotates from 0 to 180 degree based on the duty cycle of the PWM wave supplied to its signal pin.

Power Supplier:

Charger:

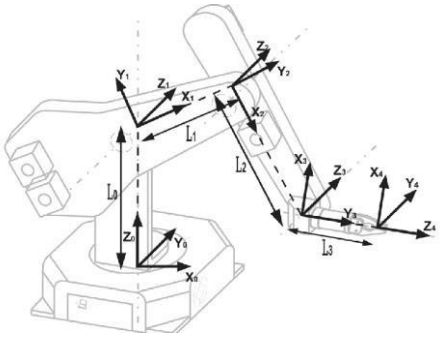
Product Description. DC 5V/2A, means that the input voltage, to recharge the battery is 5 volts of direct current. 2A Is the amount of amperage that is suggested to recharge the battery.

Battery:

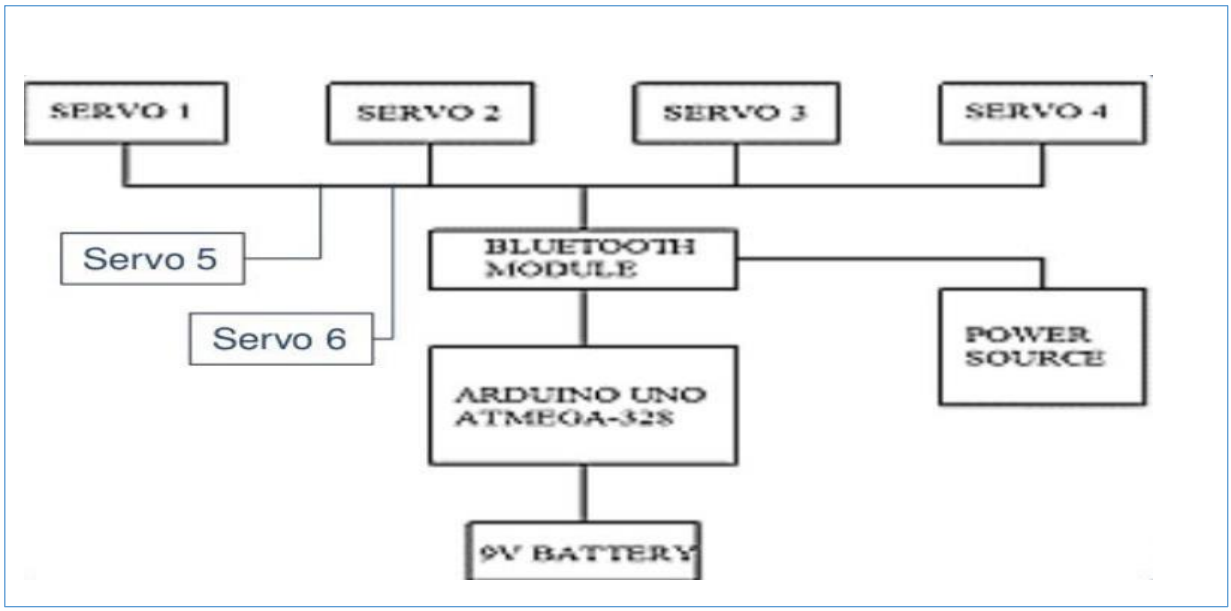
A nine-volt battery, either disposable or rechargeable, is usually used in smoke alarms, smoke detectors, walkie-talkies, transistor radios, test and instrumentation devices, medical batteries, LCD displays, and other small portable appliances.

Frame:

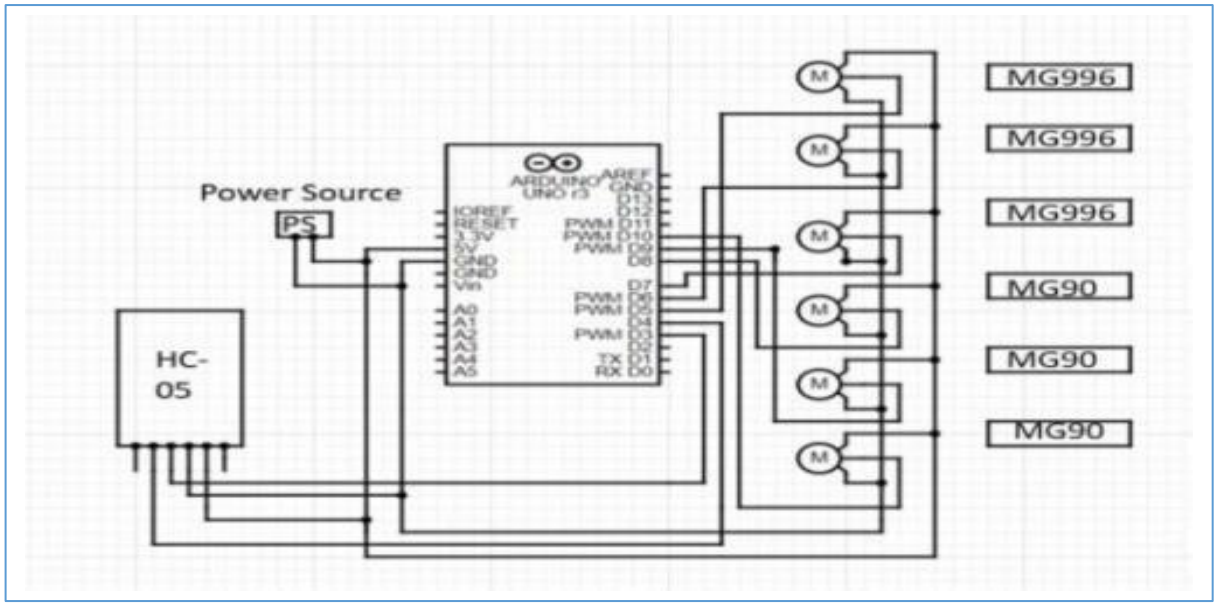
We assembled the base ,wrist ,claw mechanism separately .Then we put together all the separate parts &motors into a unified frame.



Block Diagram

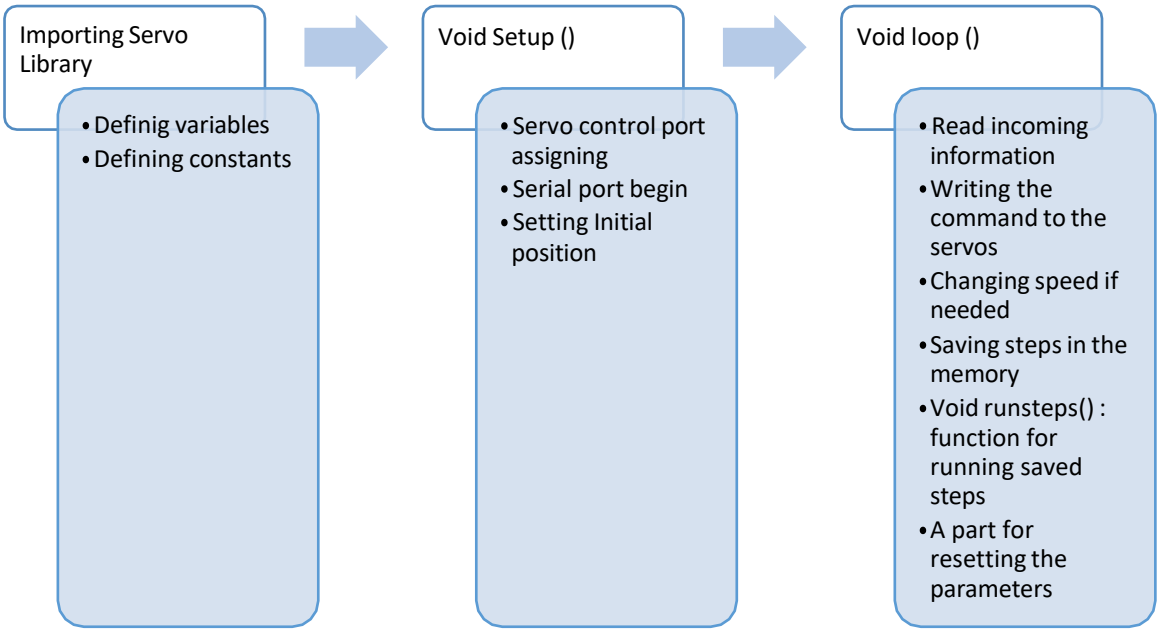


Schematic Diagram



Software:

Arduino code flowchart:



Description of functions used in code:

1. Void Setup:

```
servo01.attach(5);
servo02.attach(6);
servo03.attach(7);
servo04.attach(8);
servo05.attach(9);
servo06.attach(10);
```

In this part we assigned digital port 5, 6, 7, 8, 9 and 10 to the control pins of servo 1, 2, 3, 4, 5 and 6 respectively.

We selected bod rate for our Bluetooth serial 9600. Also, we declared initial positions for the different motors.

```
servo1PPos = 90;
servo01.write(servo1PPos);
servo2PPos = 100;
servo02.write(servo2PPos);
servo3PPos = 120;
servo03.write(servo3PPos);
servo4PPos = 95;
servo04.write(servo4PPos);
servo5PPos = 60;
servo05.write(servo5PPos);
servo6PPos = 110;
servo06.write(servo6PPos);
```

Void loop:

This is the main function of the entire code. This part loops infinitely as long as power is supplied. Here at the very beginning we read the Bluetooth data and according to it we sometimes change the position of the motor, such as

```
if (dataIn == 16) {
    m = 16;
  }
while (m == 16) {
    if (Serial.available() > 0) {
      m = Serial.read();
    }
    servo01.write(servo1PPos);
    servo1PPos++;
    delay(speedDelay);
  }
```

Sometimes save the step, such as

```
if (m == 12) {



    servo01SP[index] = servo1PPos;
    servo02SP[index] = servo2PPos;
    servo03SP[index] = servo3PPos;
    servo04SP[index] = servo4PPos;
    servo05SP[index] = servo5PPos;
    servo06SP[index] = servo6PPos;
    index++;
    m = 0;
  }
```

Sometimes speeding up the motor, such as

```
if (dataIn > 101 & dataIn < 250) {
    speedDelay = dataIn / 10;
  }
```

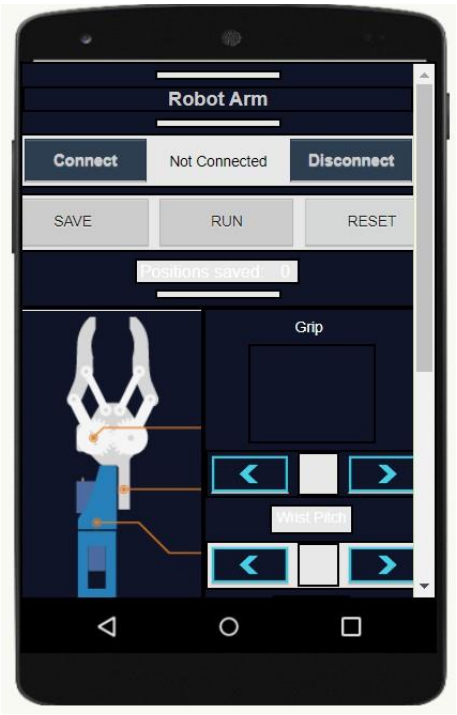
Sometimes run those saved steps, we created a function called runSteps() below we show a part of it -

```
void runSteps() {
 while (dataIn != 13) {  // Run the steps over and over again until "RESET" button is pressed
  for (int i = 0; i <= index - 2; i++) {  // Run through all steps(index)
   if (Serial.available() > 0) {      // Check for incomding data
    dataIn = Serial.read();
    if ( dataIn == 15) {          // If button "PAUSE" is pressed
     while (dataIn != 14) {        // Wait until "RUN" is pressed again
      if (Serial.available() > 0) {
       dataIn = Serial.read();
       if ( dataIn == 13) {
        break;
       }
      }
     }
    }
    // If speed slider is changed
    if (dataIn > 100 & dataIn < 150) {
     speedDelay = dataIn / 10;
    }

   if (servo01SP[i] > servo01SP[i + 1]) {
    for ( int j = servo01SP[i]; j >= servo01SP[i + 1]; j--) {
     servo01.write(j);
     delay(speedDelay);
    }
```

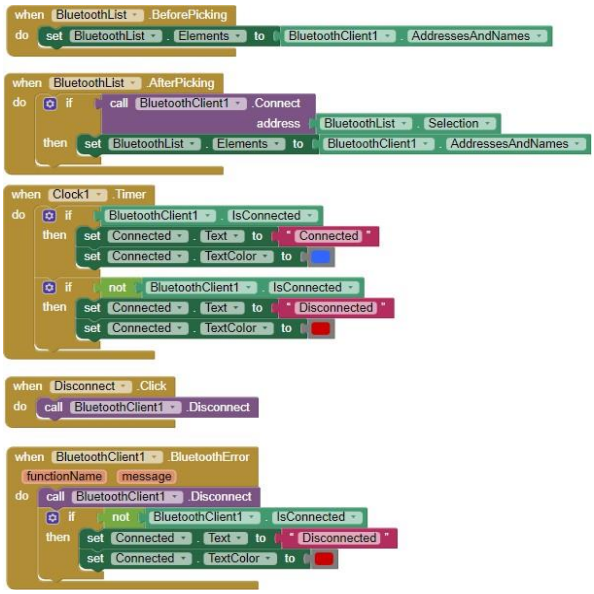
APP DESIGN

We used MIT App Inventor to make the app because it is a quite friendly and feature rich tool for basic app building using Bluetooth and there are many resources-

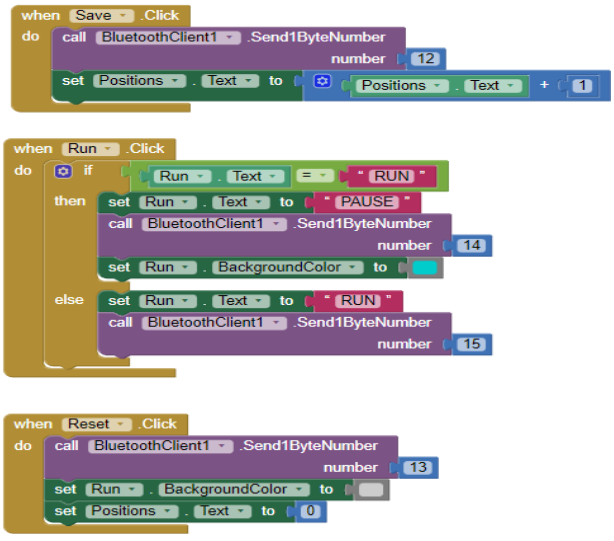


Our app interface looked like this –

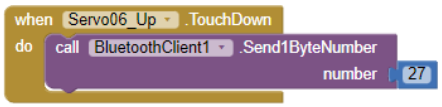
●Adding Bluetooth picking block to the app. Making 2 button for connecting and disconnecting the Bluetooth named connect and disconnect. And one text box showing the status. This 'Connect' button when clicked takes user to another internal screen which shows available networks and can select them.

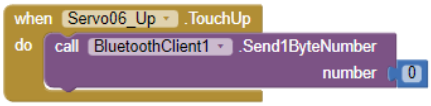


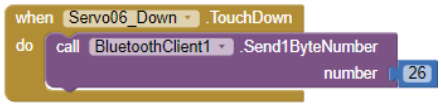
●Below we added three more buttons named SAVE, RUN, RESET respectively. As the name suggest these buttons are coded so that they will send respective integers for the actions of saving, running, resetting.

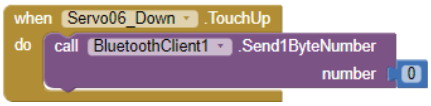


- Below this comes the part where we gave options for operating different motors at different positions. Here we used ‘Touch Up’ and ‘Touch Down’ method. Where as long as we keep in touch with the button or press it down it will send one value and when we release it, it will send another value.

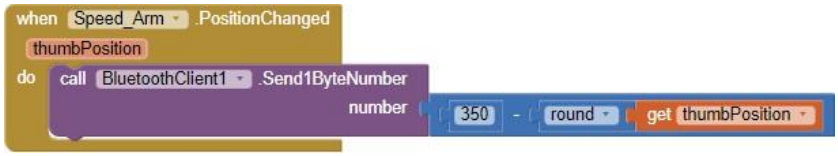








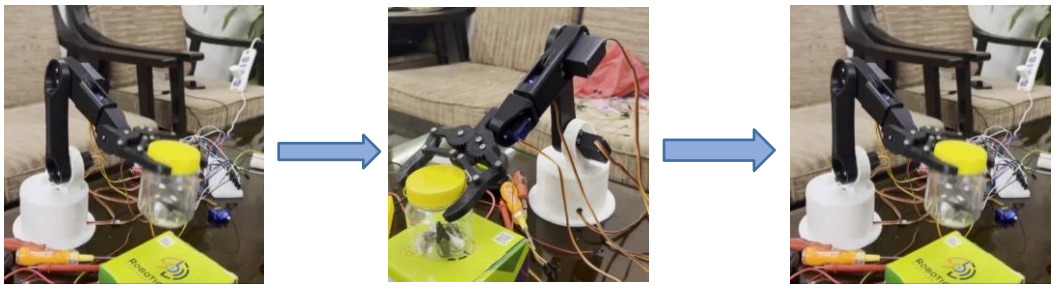
- Also we added a slider to control the speed of the sevos.



- We also added a picture of our model robotic arm and indicated which motors are controlled by which controller.

Operation Output:

We are lifting a load using the robotic arm. And doing this work repeatedly.



Problem And Debugging

Arduino single serial port problem:

A Bluetooth module requires a serial port in the Arduino to communicate. The Arduino uno we selected has one serial port. But what we did not know that this serial port is used to upload the code from computer. As long as it is connected to the computer, we cannot use its serial port to communicate.

To solve this issue, we found out that we can install software serial library of Arduino and make one of the digital pins a software serial port. But as we found out after using it, that software serial is not as reliable as hardware serial. So, we decided to use the serial to upload the code first the power the Arduino using a 9V external power then use the TX and RX serial ports as receiver for the Bluetooth module.

External source problem:

Our 6 motors in total needed approximately 2A current from their specification. But the Arduino 5 V supply cannot handle that much current. So, we had to use external source. As we needed 5V 2A rated constant voltage supply we selected on a simple phone charger. Which we think must be more reliable than building our own.

Building claw mechanism:

This part of the project was quite difficult for us. Firstly, the nut and bolts needed to be smooth and strong at the same time. Smooth because if we screw it too stiffly the mechanism which consists of gears will not move and strong because if it is given loose, any fast movement could take it apart. We had to find the screw with the exact width and ended up somewhat widening the screw holes.

Data receiving from Bluetooth:

We passed information from the app to the Arduino through Bluetooth. Firstly, we used word like up, down, on, off etc. as messages or signals to pass between devices using Bluetooth but as we found out Bluetooth is not good with long words. Then we changed it to integers. Which worked much better than strings.

PERFORMANCE ANALYSIS:

- ✓ Radius coverage almost 42 cm
- ✓ Claw can hold an object with width of 6.5 cm
- ✓ All the 6 servos are receiving signal from app properly
- ✓ Claw mechanism working smoothly

- We were not able to quantify maximum lifting capacity
- Due to a mechanical difficulty waist motor gets jammed sometimes
- As there are several mechanical joints frictional loss is high so movement becomes less smooth

COST ANALYSIS:

Arduino Uno	1	1070
MG996R 10 KG Servo(3)	3	1470
Mini servo SG90(3)	3	585
Bluetooth Module	1	345
Breadboard	1	165
Jumper Wire	80	190
Resistor	5 (1000 ohm)	5
Resistor	5(2000 ohm)	5
Jumper Wire (10.5)	12	9
Arm body	1	1500
Battery	2	180
Charger	1	250
	Total	5774

FUTURE IMPROVEMENT:

For industrial purpose of work, we can use metal like Aluminium frame instead of plastic frame. Here we used Bluetooth module to control our system which can easily interrupt after some distance. To minimize this kind of interruption, we can use WIFI module. Here our servo motors perform well only for light load. To do the same performance for heavier load, we need to use higher rated motors. Similarly, we need to improve our external supply as we are dealing with higher power required whenever we use metallic frame instead of plastic. Finally, to perform fast, accurate &reliable we can improve the control loops and the controller type.

DISCUSSION:

In this project, our system control is a key competence for us and a lot of development is made to increase its' performance, reduce cost and introduce new functionalities. Examples of development areas that get big attention are multi robot control, safe control, force control, 3D vision, remote supervision and wireless communication. The application benefits from these developments are discussed as well as the technical challenges that we meet. Model-based control is now a key technology for the control of industrial robots and models and control schemes are continuously refined to meet the requirements on higher performance even when the cost pressure leads to the design of robot mechanics that is more difficult to control. Driving forces for the future development of robots can be found in, for example, new robot applications in the automotive industry, especially for the final assembly, in small and medium size enterprises, in foundries, in food industry and in the processing and assembly of large structures. Some scenarios on future control development are proposed. Such a development could result in modular robots and in control schemes using sensors in the robot arm structure, sensors that could also be used for the implementation of redundant safe control. Introducing highly modular robots will increase the need of robot installation support, making Plug and Play functionality even more important. One possibility to obtain a highly modular robot program could be to use a recently developed new type of parallel kinematic robot structure with large work space in relation to the robot foot print. For further efficient use of robots, the scenario of adaptive robot performance is introduced. This means that the robot control is optimized with respect to the thermal and fatigue load on the robot for the specific program that the robot performs. The main conclusion of this discussion is that industrial development is far away from its limits and that a lot of research and development is needed to obtain a more widely use of this in industry.

Reference:

1. https://roboindia.com/tutorials/sending-receiving-with-hc05-mit-app-inventor/?fbclid=IwAR2_GvfeKQ1Q6RDkTZmEDIMdty7kHlF6NzT_CNHbk7NFCbHNqu6c5b2gbi4

2. https://grabcad.com/library/5-dof-robotic-arm-6?fbclid=IwAR3NsXwy_UVh7trBJ3aejqXq-qpbpd9AQlQElVJuvGsuv6EJrbzsVHerJgE

3. https://electronics.stackexchange.com/questions/280500/why-do-you-have-to-use-a-voltage-divider-with-hc-05-bluetooth-module-arduino?fbclid=IwAR1EdaP1uA1pvjahdLNYuLnpRESfK1c8dXBznEXz91r_xdsbwARjpEM1B-k