Medical Assistant Robot ARM for COVID-19 Patients Treatment – A Raspberry Pi Project

Prakash Kanade[1], Sunay Kanade[2]

[1]*Hobby Researcher in Robotics, Artificial Intelligence, IOT*
[2]*Student, Bangalore, Karnataka, India*

---***---

Abstract – *In this pandemic, the most important people are the health care workers. Without proper PPE, many of our health care workers are losing their lives to Covid-19. Let's go into the life of a health care worker. Imagine you are a health care worker treating a COVID patient. You are scared to treat them for fear of catching the virus, even with PPE. Now with a medical robot arm, you can control all the actions to treat patients from a safe distance. Giving food, water, medicine and taking different readings from patients has never been easier. This robot arms can be controlled with the health care worker's Smartphone from a safe distance.*

Key Words: COVID-19, Robotics, Raspberry PI, Bluetooth, PPE, medical robots, LeenaBOT.

1. INTRODUCTION

In this COVID-19 pandemic most important people are the medical staff. With a lack of PPE, we are losing many of our medical health care workers to the virus. For safety of health care workers, we are proposing a medical assistant robot arm that can be controlled from safe distance to take care of COVID patients. In this article we will cover how to control a robot arm using a Bluetooth connection between a phone and raspberry pi. Using a Bluetooth app installed on the phone we can connect to the python code running in raspberry pi interfaced to the robot arm. By sending signals from phone to raspberry pi we will be able move the robot arm, thus helping many health care workers whose lives are on the line.

Table -1: Bill of Materials

Components Name	Quantity	Description
6DOF Robot Arm	1	Already assembled robot arm considered for this project
MG996R servo Motor	5	Recommended servo motor for Arm
5V out Power bank	1	Two USB Out: 5V , 2.1Amp 1Amp
Breadboard	1	Small size
Breadboard power supply board	1	
Android Smart Phone	1	Smart phone to install Bluetooth App

Scope of this article is limited to controlling robot arm using Bluetooth device. For robot arm assembly please refer [1]

2. SYSTEM INTERFACE

In our proposed setup, robot arm control configuration is not tied to one Smartphone. In this COVID situation, each health care worker is not assigned to one patient, but are constantly moving from one patient to another. Availability of the health care workers is also constantly changing. In our proposed system, the robot arm is configured to connect to any nearby health care worker's Bluetooth Smartphone which comes within 30 meters range. Interface diagram of complete project. [3] [4]

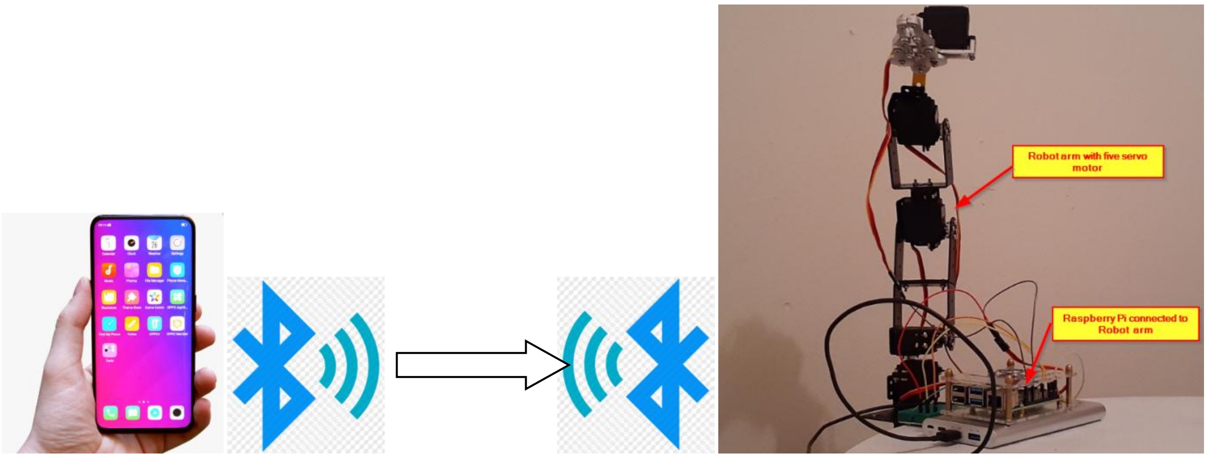

Robot arm with five servo motor
Raspberry Pi connected to Robot arm

Fig -1: System Interface diagram

3. HARDWARE ASSEMBLY

MG996R servo Motor has 3wires. Connection from servo motor to power supply board and to raspberry pi GPIO pin is as given in this Fig 2.

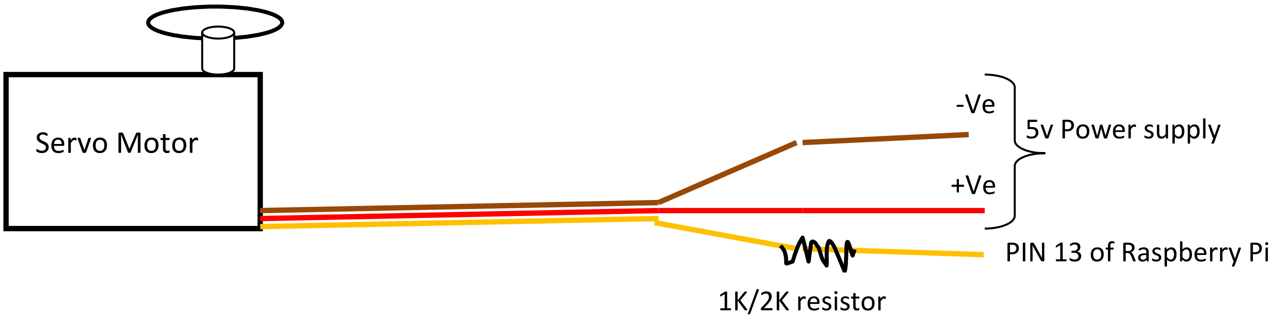


Fig -2: Servo motor to Raspberry Pi connection

Robot arm has five servo motors connected as per Table 2.

The servo motor data wire is connected to GPIO pin on raspberry pi as per circuit diagram given in Fig 2. Make the connection to each servo motor in robot hand. In our circuit we are using the raspberry pi physical pin numbers as per Table 2.

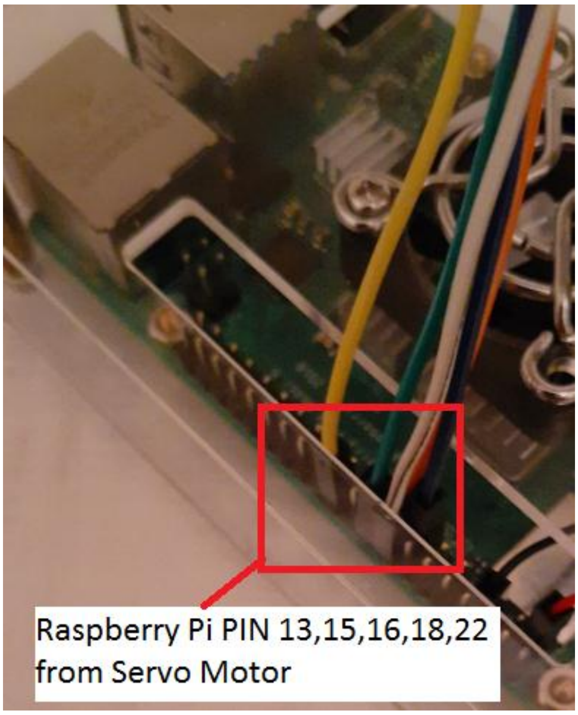


Fig -3: Raspberry Pi PIN connection

4. SOFTWARE

In this project we have three software modules

- Software to control servo motors
- Server side python code to receive data from phone
- Bluetooth client side app in android phone to send data

4.1 Software driver for servo Motor

We are using open source code servo blaster created by Richard Hirst [1] for servo motor driver.

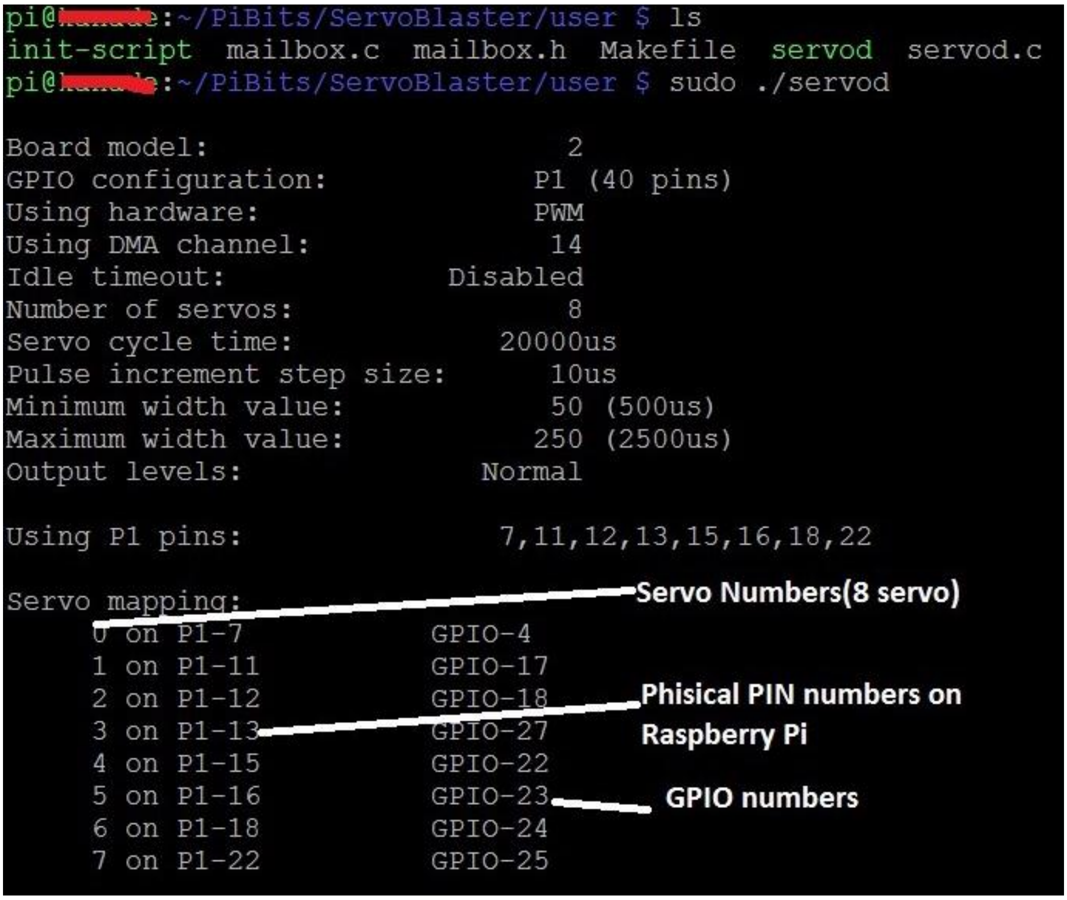
Servo blaster installation instruction
$sudo apt-get install git

Clone github servoblaster page

$sudo git clone https://github.com/srcshelton/servoblaster.git

Run Make and Make Install commands to compile and install servo blaster code

To test servo blaster installation run below command and you will get servo mapping for raspberry pi. First run simple echo command to test one servo motor, Once one servo motor is working good. Same test to be done for all the joints.



```
pi@[redacted]:~/PiBits/ServoBlaster/user $ ls
init-script  mailbox.c  mailbox.h  Makefile  servod  servod.c
pi@[redacted]:~/PiBits/ServoBlaster/user $ sudo ./servod

Board model:                     2
GPIO configuration:            P1 (40 pins)
Using hardware:                PWM
Using DMA channel:              14
Idle timeout:             Disabled
Number of servos:                8
Servo cycle time:          20000us
Pulse increment step size:    10us
Minimum width value:            50 (500us)
Maximum width value:           250 (2500us)
Output levels:              Normal

Using P1 pins:               7,11,12,13,15,16,18,22

Servo mapping:
     0 on P1-7           GPIO-4
     1 on P1-11          GPIO-17
     2 on P1-12          GPIO-18
     3 on P1-13          GPIO-27
     4 on P1-15          GPIO-22
     5 on P1-16          GPIO-23
     6 on P1-18          GPIO-24
     7 on P1-22          GPIO-25
```

Servo Numbers(8 servo)

Phisical PIN numbers on Raspberry Pi

GPIO numbers

Fig -4: Servo blaster mapping

Execute the command in this format. Here servo motor is connected to servo pin 3(Raspberry Pi PIN 13). Circuit is as per Fig 2. Once you run this command servo has to rotate for 100 degree angle. [2]

```
$ sudo echo 3=100 >/dev/servoblaster
```

We are using five GPIO PINS for five servo motors in Robot arm. Mapping and angle of rotation is as given in below table

Table -2: Robot Arm servo mapping

Servo Motor of arm	ServoBlaster PIN	Raspberry Pi PIN	Angle of rotation
Base servo Motor	3	PIN 13	60-240
Shoulder Two servo	6,4	PIN 18, PIN15	60-100
Elbow Servo	7	PIN 22	65-115
Gripper Servo	5	PIN 16	60-130

4.2 Server side Program

Server side python code is a Bluetooth socket program to connect between two devices. Bluetooth socket program to make connection with Bluetooth app running on smart phone and Bluetooth server side python code running on Raspberry Pi. In Raspberry Pi execute Bluetooth Server side python code and it will be in Listening mode to accept connection. [2]

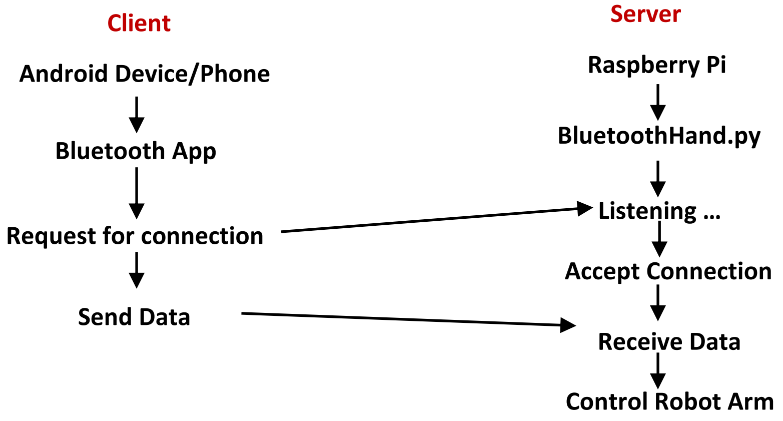


Fig -5: Sequence diagram

4.3 Bluetooth app Installed on Android Phone

There are multiple Bluetooth app in Google store of android smart phone. You can use any of these apps to send data. Depending on app you installed, you need to modify Bluetooth server side socket program to receive data from phone. For our testing we installed MeArm Bluetooth App [5] and configured data code in MeArm setting as per Table 2.

Now run the BlueHand.py program in raspberry pi and it will go into "Listening.." mode for incoming connection.

From Bluetooth app go to setting and select raspberry pi to make Bluetooth connection to BlueHand.py program in Listening mode.[2]

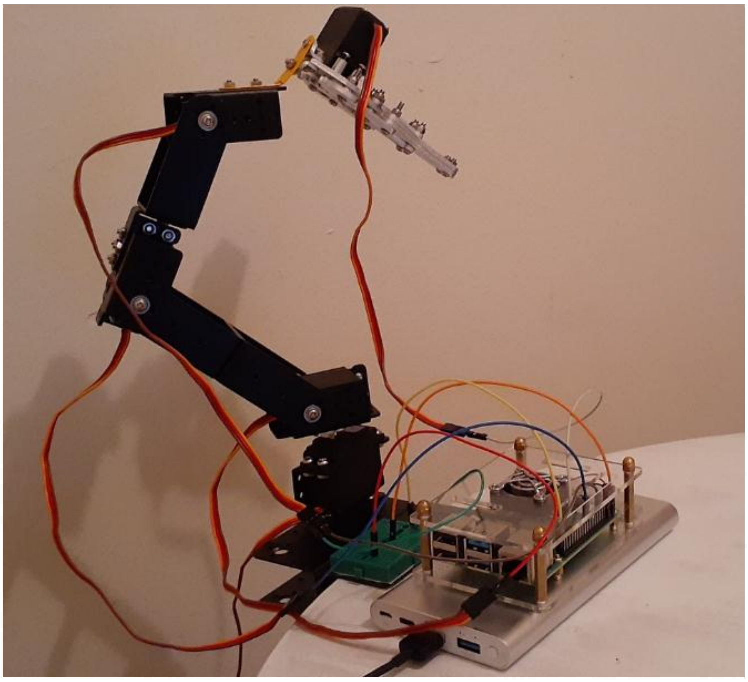
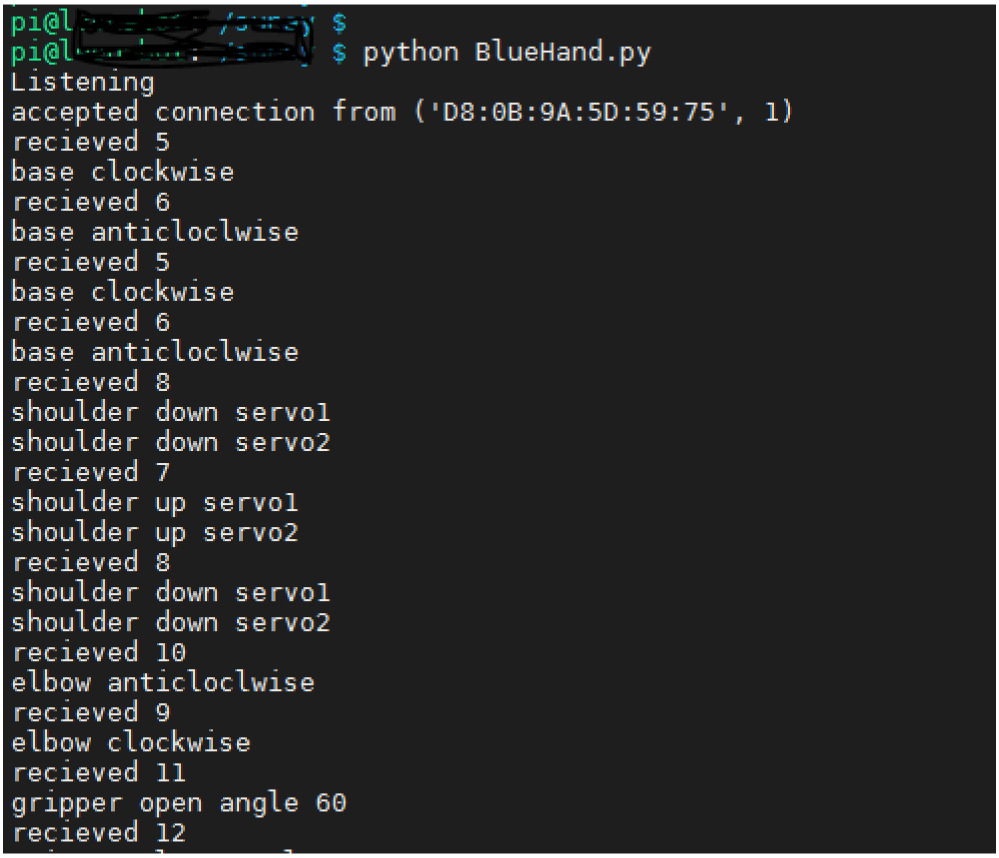


Fig -5: Robot arm complete setup with raspberry pi

5. Testing and Result

Any Bluetooth enabled Smartphone with Bluetooth robot control app installed can establish connection once in acceptable range of 30 meters from Raspberry Pi attached robot arm. So In our scenario Healthcare worker will have Bluetooth robot control app installed and will be able to establish connection once they come within range. Once the connection is established Healthcare workers can easily control or move robot arm with click of buttons on phone app screen.

In Bluetooth App select Raspberry and click connect. Connection will be established with BlueHand.py program running in Raspberry pi. Now send control values from Bluetooth app from Phone by click on buttons on the app. Based on received data robot arms servo motors will make move. Result is as displayed in Fig 6.



```
pi@l[redacted] $
pi@l[redacted] $ python BlueHand.py
Listening
accepted connection from ('D8:0B:9A:5D:59:75', 1)
recieved 5
base clockwise
recieved 6
base anticloclwise
recieved 5
base clockwise
recieved 6
base anticloclwise
recieved 8
shoulder down servo1
shoulder down servo2
recieved 7
shoulder up servo1
shoulder up servo2
recieved 8
shoulder down servo1
shoulder down servo2
recieved 10
elbow anticloclwise
recieved 9
elbow clockwise
recieved 11
gripper open angle 60
recieved 12
```

Fig -6: Test results

5. Conclusion

In this paper, we are proposing a smart phone Bluetooth controlled Robot arm to assist health care workers treating highly contagious disease like COVID-19. With existing situation of health care workers availability we proposed solution where any healthcare worker in acceptable distance from robot arm can easily connect to it and treat patients or provide water, food, medicine to patients. In our test we were able to test the setup with 6DOF robot arm and Bluetooth app installed on phone. Controller board we used is very cost-effective raspberry pi. We believe if this solution implemented in field, can help save life of hundreds of healthcare workers those who are fighting this invisible enemy of Human race.

REFERENCES

[1] Richard Hirst Servo Blaster https://github.com/richardghirst/PiBits/tree/master/ServoBlaster

[2] https://www.leenabot.com/Raspberry-Pi-Controlling-Robot-using-Bluetooth/

[3] Rahul Kumar1, Ushapreethi P2 “Android Phone controlled Bluetooth Robot” International Research Journal of Engineering and Technology (IRJET)

[4] Arpit Sharma1, Reetesh Varma2, Saurabh Gupta3 and Sukhdeep Bhatia4, “Android Phone Controlled Robot Using Bluetooth” IJEEE ISSN 0974-2174, Volume 7, Number 5 (2014, pp. 443-448)

[5] https://play.google.com/store/apps/details?id=appinventor.ai_Muhandi_prasetyo.MeArm&hl=en_US