International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Fabrication of Wireless Haptic Robotic Arm

Shaik Aslam[1], Chirivella Venkata Saravan Kumar[2], Kunam Venkata Krishna Reddy[3], Gudluru Sai Teja[4], Shaik Zakir Hussain[5], S.M Shafee[6]

[1,2,3,4]B.Tech., Department of Mechanical Engineering, Sree venkateswara college of engineering, Nellore, Andhra Pradesh, India
[5]Assistant professor, Department of Mechanical Engineering, Sree venkateswara college of engineering, Nellore, Andhra Pradesh, India
[6]Associate Professor, Department of Mechanical Engineering, Sree venkateswara college of engineering, Nellore, Andhra Pradesh, India

Abstract
The current generation robot has been used in fields isolated from the human society. Their application is limited because of they cannot manipulation and interaction with humans. In order to represent the robotic technology with human-machine interaction and wireless communication allows interactivity in real-time with virtual objects. The main objective of the work is to design and develop a Controller that is used to move the robotic arm using wireless system by recognizing hand motion that is controlled by flex sensor & tilt sensor for virtual environment & human-machine interaction

Keywords: Robotic Hand, Sensors

1.1 INTRODUCTION

A robotic arm consists of several sections connected together by linkages that help the arm to travel specifically in a designed pattern, with sensors ensuring that all movements are exactly of the similar pattern. They are endowed with several degrees-of-freedom, giving them the flexibility to move in many directions through multiple angles with utmost ease and agility.

. 1.2 SOFTWARE

Embedded C is a set of language extensions for the C Programming the C standards committee to address commonality issues that exist between C extensions for different embedded systems. Historically, embedded C programming requires nonstandard extensions to the C language in order to support exotic features such as fixed-point arithmetic, multiple distinct memory banks and basic input output operations. In 2008, the C Standards Committee extended the C language to address these issues by providing a common standard for all implementations to adhere to.

1.2.1 INTRODUCTION TO EMBEDDED C

Looking around, we find ourselves to be surrounded by various types of embedded systems. Be it a digital camera or a mobile phone or a washing machine, all of them has some kind of processor functioning inside it. Associated with each processor is the embedded software. If hardware forms the body of an embedded system, embedded processor acts as the brain, and embedded software forms its soul. There used to be no mechanism to find what the program was doing. LEDs, switches, etc. were used to check correct execution of the program.

1.2.2EMBEDDEDSYSTEMS PROGRAMMING

Embedded systems programming is different from developing applications on a desktop computer. Key characteristics of an embedded system, when compared to PCs, are as follows:
Embedded systems are programmed using different type of languages:

- Machine Code
- Low level language, i.e., assembly
- High level language like C, C++, Java, Ada, etc.
- Application-level language like Visual Basic, Access, etc.

Assembly language maps mnemonic words with the binary machine codes that the processor uses to code the instructions. Assembly language seems to be an obvious choice for programming embedded devices. However, use of assembly language is restricted to developing efficient codes in terms of size and speed. Also, assembly codes lead to higher software development costs and code portability is not there.

1.2.3 USE OF C IN EMBEDDED SYSTEMS

- It is small and reasonably simpler to learn, understand, program and debug.
- It is fairly efficient
- It supports access to I/O and provides ease of management of large embedded projects.

2.1 LITERATURE SURVEY

BACKGROUND The word robot was derived from Czech word Robota which means "a forced labor". Thus, the robot technology is advancing rapidly. Robotic hand is basically kinematics chain of rigid links interconnected by movable joints. The hand is also called end effector. Much of it refers to the categorization and study of six grasps: cylindrical, fingertip, hook, palmar, spherical and lateral, leading to associating the kind of human grasps with the shapes of the objects to be manipulated.

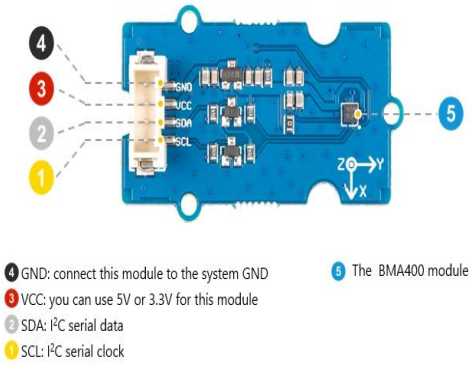
- The research work performed by **Ankit Purohit and MakarandKakatkar** describes Tele-operation system using haptic technology, an operator controls the movements of a robot which is located at some distance by using different types of force sensors, angel sensors and gyro sensors.
- Robotic grippers are employed in robotic manipulators that perform repetitive tasks. These grippers can only execute limited and specific manipulation tasks **[penisi et al., 2003].** They are limited to objects that are very similar in terms of shapes, weight and manipulation requirements, **[chen,1982].** The use of grippers is also limited to the grasping of objects with regular geometry.

COMPONENTS

- ARDUINO
- ACCERELOMETER SENSOR
- TRANSCEIVER MODULE
- 12V BATTERY
- DC MOTOR DRIVE MODULE
- GRIPPER

3.1 ACCELEROMETER SENSOR

An accelerometer is a tool that measures proper acceleration. . For example, an accelerometer at rest on the surface of the Earth will measure an acceleration due to Earth's gravity, straight upwards (by definition) of $g \approx 9.81\ m/s^2$. By contrast, accelerometers in free fall (falling toward the center of the Earth at a rate of about $9.81\ m/s^2$) will measure zero.



Accelerometers have many uses in industry and science. Highly sensitive accelerometers are used in inertial navigation systems for aircraft and missiles. Vibration in rotating machines is monitored by accelerometers. They are used in tablet computers and digital cameras so that images on screens are always displayed upright. In unmanned aerial vehicles, accelerometers help to stabilise flight.

3.2 TRANSCEIVER MODULE

In radio communication, a transceiver is an electronic device which is a combination of a radio transmitter and a receiver, hence the name. It can both transmit and receive radio waves using an antenna, for communication purposes. These two related functions are often combined in a single device to reduce manufacturing costs. The term is also used for other devices which can both transmit and receive through a communications channel, such as *optical transceivers* which transmit and receive light in optical fibre systems, and *bus transceivers* which transmit and receive digital data in computer data buses.

3.3 12V BATTERY

The twelve-volt battery, or 12-volt battery, is an electric battery that supplies a nominal voltage of 12 volts. Actual voltage measures 12 to 20 volts, depending on battery chemistry. Batteries of various sizes and capacities are manufactured; a very common size is known as PP3, introduced for early transistor radios. The PP3 has a rectangular prism shape with rounded edges and two polarized snap connectors on the top. This type is commonly used for many applications including household uses such as smoke and gas detectors, clocks, and toys.

3.4 DC MOTOR DRIVE MODULE

Motor drive means a system that includes a motor. An adjustable speed motor drive means a system that includes a motor that has multiple operating speeds. A variable speed motor drive is a system that includes a motor and is continuously variable in speed. If the motor is generating electrical energy rather than using it – this could be called a generator drive but is often still referred to as a motor drive.

If the output speed can be changed without steps over a range, the drive is usually referred to as *variable speed*.

3.5 GRIPPER

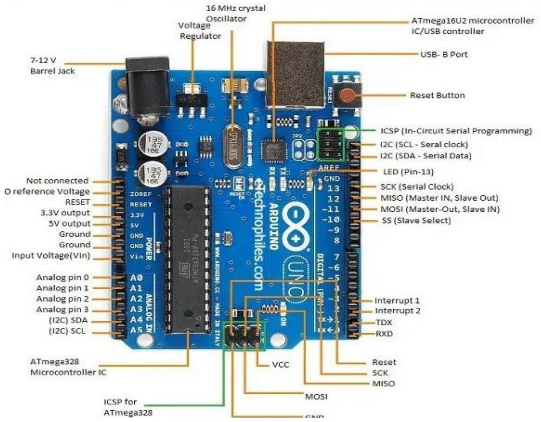
Grippers, sometimes called hand grippers, are primarily used for testing and increasing the strength of the hands; this specific form of grip strength has been called crushing grip, which has been defined as meaning the prime movers are the four fingers, rather than the thumb.

There are differences from brand to brand, but the common features of standard grippers are that they use a torsion spring fitted with two handles. The exact dimensions of these elements vary, as well as the materials used to make them; the springs are made from various types of steel, and the handles are generally made from wood, plastic, steel or aluminium.

3.6 ARDUINO

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to arious expansion boards (shields) and other circuits.[1] The board has 14 digital I/O pins (six capable of PWM output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable. It can be powered by the USB cable or by an external 12-volt battery, though it accepts voltages between 12 and 20 volts.

3.6.1 PIN DIAGRAM



GENERAL PIN FUNCTIONS

LED: There is a built-in LED driven by digital pin 13. When the pin is high value, the LED is on, when the pin is low, it is off.

VIN: The input voltage to the Arduino/Genuino board when it is using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

5V: This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V).

4.1 EMBEDDED SYSTEM

Embedded systems are controllers with on chip control. They consist of microcontrollers, input and output devices, memories etc., on chip and they can be used for a specific application. A small computer designed in a single chip is called a single chip microcomputer. A single chip microcomputer typically includes a microprocessor RAM, ROM, timer, interrupt and peripheral controller in a single chip. This single chip microcomputer is also called as microcontroller. Embedded systems are used for real time applications with high reliability, accuracy and precision, Embedded systems are operated with Real Time Operating systems like WinCE, RT Linux, VxWorks, PSOS, etc...,

4.3 APPLICATIONS OF EMBEDDED SYSTEM

- Robotics
- Aviation
- Telecommunication and Broadcasting

4.4 CODING

```
#include <SoftwareSerial.h>
SoftwareSerial HC12(11, 12); // HC-12 TX Pin, HC-12 RX Pin
int x_axis_1, y_axis_1, x_axis_2, y_axis_2;
void setup()
{
 Serial.begin(9600);            // Serial port to computer
 HC12.begin(9600);              // Serial port to HC12
}
void loop()
{
  x_axis_1 = analogRead(A0);
  y_axis_1 = analogRead(A1);
  x_axis_2 = analogRead(A2);
  y_axis_2 = analogRead(A3);
  HC12.write('%');
  HC12.write(x_axis_1/100%10+48);
  HC12.write(x_axis_1/10%10+48);
  HC12.write(x_axis_1/1%10+48);
  HC12.write(',');
  HC12.write(y_axis_1/100%10+48);
  HC12.write(y_axis_1/10%10+48);
  HC12.write(y_axis_1/1%10+48);
  HC12.write(',');
  HC12.write(x_axis_2/100%10+48);
  HC12.write(x_axis_2/10%10+48);
  HC12.write(x_axis_2/1%10+48);
  HC12.write(',');
  HC12.write(y_axis_2/100%10+48);
  HC12.write(y_axis_2/10%10+48);
  HC12.write(y_axis_2/1%10+48);
  HC12.print("#\r\n");
    delay(100);
}
#include <SoftwareSerial.h>
```

```
SoftwareSerial HC12(11, 12); // HC-12 TX Pin, HC-12 RX Pin
const int IN1 = 9;
const int IN2 = 8;
const int IN3 = 6;
const int IN4 = 7;
const int IN5 = 5;
const int IN6 = 4;
const int IN7 = 3;
const int IN8 = 2;
int x_axis_1, y_axis_1, x_axis_2, y_axis_2;
unsigned char inByte;
unsigned char arr[20], ii, start_flag = 0, over_flag = 0;
void setup()
{
 Serial.begin(9600);            // Serial port to computer
 HC12.begin(9600);              // Serial port to HC12
 pinMode(IN1, OUTPUT); digitalWrite(IN1, LOW);
 pinMode(IN2, OUTPUT); digitalWrite(IN2, LOW);
 pinMode(IN3, OUTPUT); digitalWrite(IN3, LOW);
 pinMode(IN4, OUTPUT); digitalWrite(IN4, LOW);
 pinMode(IN5, OUTPUT); digitalWrite(IN5, LOW);
 pinMode(IN6, OUTPUT); digitalWrite(IN6, LOW);
 pinMode(IN7, OUTPUT); digitalWrite(IN7, LOW);
 pinMode(IN8, OUTPUT); digitalWrite(IN8, LOW);
}
void loop()
{
  while(HC12.available())  // If HC-12 has data
  {
   inByte = HC12.read();
   //Serial.write(inByte);
   if(start_flag == 0)
   {
    if(inByte == '%')
    {
     ii = 0;
     start_flag = 1;
    }
   }
   else if(start_flag == 1)
```

```
 {
  if(inByte == '#')
  {
   over_flag = 1;
   start_flag = 0;
  }
  else
  {
   arr[ii++] = inByte;
  }
 }
}
if(over_flag == 1)
{
 x_axis_1 = (((int)arr[ 0] - 48) * 100) + (((int)arr[ 1] - 48) * 10) + (((int)arr[ 2] - 48) * 1);
 y_axis_1 = (((int)arr[ 4] - 48) * 100) + (((int)arr[ 5] - 48) * 10) + (((int)arr[ 6] - 48) * 1);
 x_axis_2 = (((int)arr[ 8] - 48) * 100) + (((int)arr[ 9] - 48) * 10) + (((int)arr[10] - 48) * 1);
 y_axis_2 = (((int)arr[12] - 48) * 100) + (((int)arr[13] - 48) * 10) + (((int)arr[14] - 48) * 1);
/*
Serial.write(x_axis_1/100%10+48);
Serial.write(x_axis_1/10%10+48);
Serial.write(x_axis_1/1%10+48);
Serial.write(',');
Serial.write(y_axis_1/100%10+48);
Serial.write(y_axis_1/10%10+48);
Serial.write(y_axis_1/1%10+48);
Serial.write(',');
Serial.write(x_axis_2/100%10+48);
Serial.write(x_axis_2/10%10+48);
Serial.write(x_axis_2/1%10+48);
Serial.write(',');
Serial.write(y_axis_2/100%10+48);
Serial.write(y_axis_2/10%10+48);
Serial.write(y_axis_2/1%10+48);
Serial.print("\r\n");
*/
 if(  (x_axis_1 > 290 && x_axis_1 < 370) && (y_axis_1 > 290 && y_axis_1 < 370) )
 {
  //Serial.write('S');
  digitalWrite(IN1,  LOW);
```

```
digitalWrite(IN2,  LOW);
 digitalWrite(IN3,  LOW);
 digitalWrite(IN4,  LOW);
}
else if( !(x_axis_1 > 290 && x_axis_1 < 370) && (y_axis_1 > 290 && y_axis_1 < 370) )
{
 if(x_axis_1 <= 290)
 {
  //Serial.write('R');
  digitalWrite(IN1,  LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4,  LOW);
 }
 else if(x_axis_1 >= 370)
 {
  //Serial.write('L');
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2,  LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
 }
}
else if(  (x_axis_1 > 290 && x_axis_1 < 370) && !(y_axis_1 > 290 && y_axis_1 < 370) )
{
 if(y_axis_1 <= 290)
 {
  //Serial.write('F');
  digitalWrite(IN1,  LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3,  LOW);
  digitalWrite(IN4, HIGH);
 }
 else if(y_axis_1 >= 370)
 {
  //Serial.write('B');
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2,  LOW);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4,  LOW);
```

```
 }
}
if(  (x_axis_2 > 290 && x_axis_2 < 370) && (y_axis_2 > 290 && y_axis_2 < 370) )
{
 //Serial.write('E');
 digitalWrite(IN5,  LOW);
 digitalWrite(IN6,  LOW);  //gripper
 digitalWrite(IN7,  LOW);
 digitalWrite(IN8,  LOW);  //shoulder
}
else if( !(x_axis_2 > 290 && x_axis_2 < 370) && (y_axis_2 > 290 && y_axis_2 < 370) )
{
 if(x_axis_2 <= 290)
 {
  //Serial.write('A');
  digitalWrite(IN5,  LOW);
  digitalWrite(IN6, HIGH);  //gripper
 }
 else if(x_axis_2 >= 370)
 {
  //Serial.write('B');
  digitalWrite(IN5, HIGH);
  digitalWrite(IN6,  LOW);  //gripper
 }
}
else if(  (x_axis_2 > 290 && x_axis_2 < 370) && !(y_axis_2 > 290 && y_axis_2 < 370) )
{
 if(y_axis_2 <= 290)
 {
  //Serial.write('C');
 digitalWrite(IN7,  LOW);
 digitalWrite(IN8, HIGH);  //shoulder
}
 else if(y_axis_2 >= 370)
 {
  //Serial.write('D');
  digitalWrite(IN7, HIGH);
  digitalWrite(IN8,  LOW);  //shoulder
 }
}
```

```
  over_flag = 0;
 }
```

4.3 PROJECT IMAGES

CONCLUSION

The overall system performs reasonably well. The user is able to carry out comfortable and precise functions of the robotic Hand through the use of a sensor-based control glove. Furthermore, the robotic Hand is capable to carry normal routine function as human hand does. The microcontroller accepts inputs from the sensor and generates the proper control signals based on those inputs. The usable lifetime of the flex sensors seems to be limited. The sensors themselves are very fragile and easily wear out from overuse. Careful maintenance and protection of the flex sensors is crucial to successful operation of the system. The robotic hand was able to imitate the hand motions of the controller glove and grip objects through wireless communication. The robotic hand was able to grip objects.

FUTURE SCOPE

We will work on this project in future and make it an essential component of all factories involving movement of hazardous substances and chemicals. Further robotic hand will be made for more advanced technologies when it will be wireless and can be used from far places without hindrance. Now, sensors are used for moving fingers afterwards we will make it auto for all those works which need not variable instructions.

Another form of this hand will be access from software wirelessly without sensors which will reduce cost as well.

REFERENCES

1. M. Nore and C. Westerberg, Robotic arm controlled by arm movements, URN: urn: nbn:se: kth: diva-264509, 2019.
2. F. Hardell and J. Tjomsland, Motorized tensioner system for prosthetic hands, URN: urn: nbn:se: kth: diva-230144, 2018.
3. R. Nilsson and A. Winquist de Val, Hand gesture-controlled wheelchair, URN: urn: nbn:se: kth: diva-264512, 2019.
4. R. J. Schwarz and C. Taylor, "The anatomy and mechanics of the human hand", Artificial limbs, vol. 2, no. 2, pp. 22–35, 1955.