SMART ROBOTIC ARM

MINIPROJECT REPORT

Submitted by

KISHODHARANI S	**717821L323**
LOGESH M	**717821L326**
SUDARSUN S	**717821L354**

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

KARPAGAM
COLLEGE OF ENGINEERING
Rediscover | Refine | Redefine
(Autonomous)

ANNA UNIVERSITY,CHENNAI

NOVEMBER 2023

KARPAGAM COLLEGE OF ENGINEERING
Rediscover | Refine | Redefine
(Autonomous)

CERTIFICATE

Certified that this project report titled **"SMART ROBOTIC ARM"** is the bonafide work of **SUDARSUN S, LOGESH M, KISHODHARANI S**, who carried out the project under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report. I understand the policy on plagiarism and declare that the project and publications are my own work, except where specifically acknowledged and has not been copied from other sources or been previously submitted for award or assessment.

Faculty guide
P.K. GHIBITHA BEBIN M.E,

Head of the Department
Dr R. SARANKUMAR M.E., Ph.D.,

Certified that the candidate was examined in the viva -voce examination held on __________

Internal Examiner

ACKNOWLEDGEMENT

We would like to show our gratitude to the Managing Trustee of Karpagam College of Engineering, **Dr. R. VASANTHAKUMAR, B.E (Hons)., D.Sc.,** Chairman and Managing Trustee, Karpagam Educational Institutions, for providing us with all sorts of supports in the completion of this project

It is the moment of immense pride for us to reveal our profound thanks to our respected principal, **Dr. V. KUMARCHINNAIYAN, M.E., Ph.D.,** who happens to be striving force inall our endeavors.

We express our sincere thanks to the Head of the Department of Electronics and Communication Engineering **Dr. R. SARANKUMAR, M.E., Ph.D.,** for providing an opportunity to work on this project. Her valuable suggestions helped us a lot to do this project.

We are grateful to our Project Coordinator, **Mr. N.M. JAYADEVAN, M.E., Ph.D.,** Associate Professor, Department of Electronics and Communication Engineering, for his valuable suggestions and guidance throughout the course of this project.

A word of thanks would not be sufficient for the work of our mini project guide **P.K.GHIBITHA BEBIN M.E.,** Department of Electronics and Communication Engineering whose efforts and inspiration lead us through every trying circumstance.

We deeply express our gratitude to all the members of the faculty of the Department of electronics and communication engineering for the encouragement, which we received throughout the semester.

ABSTRACT

The system is to design and develop a functional robotic arm that canperform a variety of tasks, such as pick-and-place operations, object manipulation, and basic gestures. Robotic arms have gained significant importance in various industries, from manufacturing to healthcare and even in everyday life. The system aims to provide a fundamental understanding of the key principles and components involved in creating a roboticarm. The project involves the design of a simple robotic arm prototype, driven by servo motorsand controlled by a microcontroller. In addition to the hardware development, a user-friendly program has been given to control the robotic arm. A 4 Degree of Freedom (DOF) robotic armhas been developed. It is controlled by an Arduino Uno microcontroller which accepts input signals from a user by means of a glove unit. The arm is made up of three rotary joints and an end effector, where rotary motion is provided by a servomotor. Each link and the outer structureis created by 3d printing. The servo-motors and links thus produced assembled with fasteners produced the final shape of the arm. The Arduino has been programmed to provide rotation toeach servo motor corresponding to the amount of rotation of movement of the hand glove. Theglove has a gyroscope and accelerometer which detects the movement of the hand in the 3- dimensional space. The interface allows users to manipulate the arm's movements. This aspectof the project introduces concepts related to human-machine interaction and programming.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF ABBERIVATIONS

UART	Universal Asynchronous Receiver-Transmitter
USB	Universal Serial Bus
LCD	Liquid Crystal Display
LED	Light Emitting Diode
SPI	Serial Peripheral Interfaces
PWM	Pulse Width Modulation

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Robotic arms have revolutionized automation in a wide range of industries, from manufacturing and healthcare to aerospace and research. They are versatile tools capable of precise and complex movements. Integrating gyroscopic sensors and controllers into a robotic arm project adds a new dimension of control and functionality, enabling the arm to respond dynamically to changes in orientation and enhance its accuracy and adaptability. The integration of gyroscopic sensors and controllers allows the robotic arm to perceive its own position and orientation in real time. Gyroscopic sensors, such as gyroscopes and accelerometers, can detect changes in angular velocity and acceleration, enabling the arm to adjust its movements accordingly. These sensors provide vital feedback to the control system, allowing the robotic arm to maintain stability, make precise movements, and adapt to changing environments. This project harnesses the power of gyroscopic sensors and controllers to develop a sophisticated robotic arm with enhanced capabilities. The project involves the mechanical structure of the robotic arm, integrating gyroscopic sensors, and developing a control system that interprets sensor data and adjusts the arm's movements using servo motor in real time. This interdisciplinary project combines aspects of mechanical engineering, electronics, software development, and control theory to create a dynamic and adaptable robotic arm.

1.2 APPLICATION

The robotic arm project with gyroscopic sensors and controllers has a wide range of potential applications across various industries and fields, they can enhance precision and speed in manufacturing and assembly lines. They can be used for tasks like picking and placing components, soldering, welding, and quality control. Robotic surgical arms with gyroscopic sensors can provide greater dexterity and stability for minimally invasive surgeries. Surgeons can perform procedures with enhanced precision and reduced risk, and also valuable for sorting and handling packages in warehouses and distribution centers, improving efficiency and reducing the risk of damage during transportation. The versatility and adaptability of robotic arms with gyroscopic sensors make them suitable for a wide range of applications, contributing to increased efficiency, safety, and precision across various industries. As technology continues to advance, the potential for new and innovative applications in robotics is virtually limitless.

CHAPTER 2

METHODOLOGY

2.1 EXISTING METHODOLOGY

In the existing system we have the robotic arm with 4 degree of rotation. Also the control of the Robotic arm is by the remote control or infrared signal. The Robotic Arm we have now can pick and place the Objects but it is restricted due to the lack of access to all surrounding place to it. Because there is only 4 degree of rotation and it can't drive from one place to another to pick or place an object too far from the robotic Arm.

2.2 PROPOSED METHODOLOGY

In the Proposed methodology the main part of the project is the robotic arm, which can pick and place things from one place to another. To control this action, we can move the robotic arm by giving moving the hands which is connected to the glove unit. The robotic arm is equipped with servo motors. These motors help to move the arm in desired direction. The motors are controlled with the help of a microcontroller (ARDUINO). The control is given via the glove unit which contains the accelerometer. This acts as the transmitter and gives the signal to the microcontroller regarding the movement of the hand. The signal which is given to the robotic arm is actually is verified by the controller and hence we can access the robotic arm from a desired distance.

2.3 BLOCK DIAGRAM

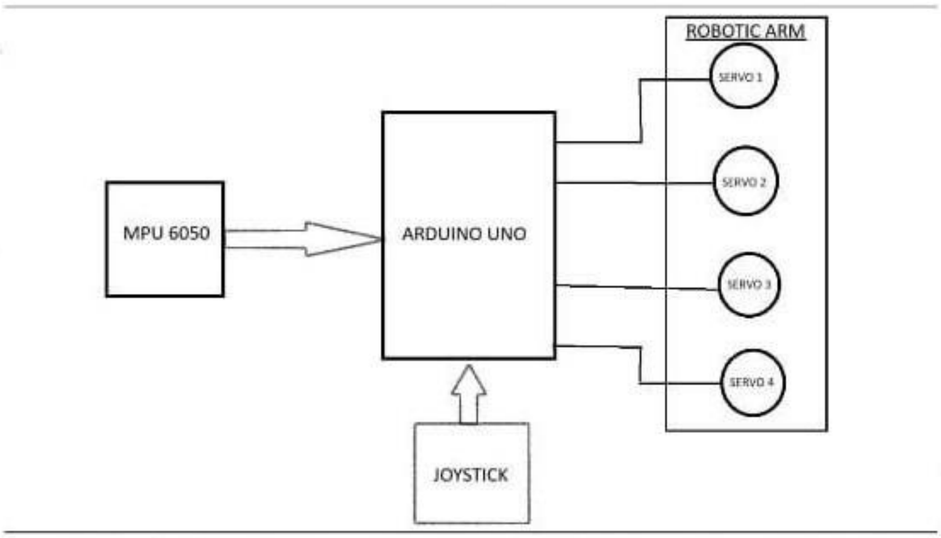

FIG 2.1 BLOCK DIAGRAM

2.4 CIRCUIT DIAGRAM

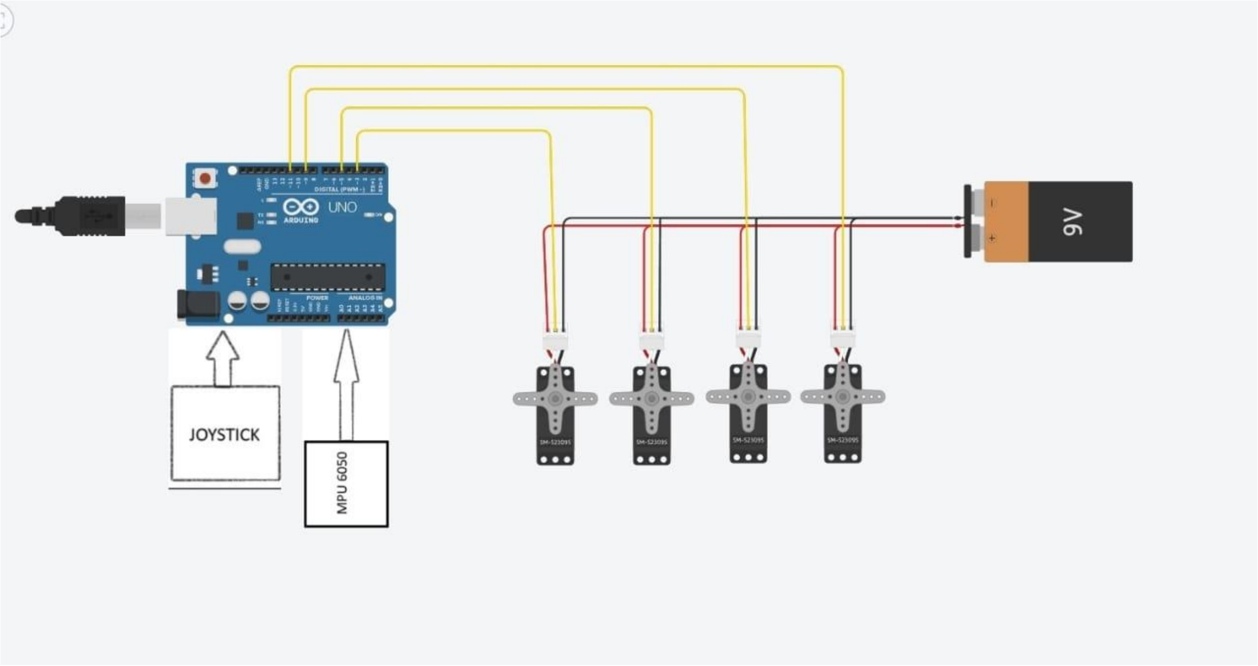

FIG 2.2 CIRCUIT DIAGRAM OF SMART ROBOTIC ARM

CHAPTER 3

HARDWARE AND SOFTWARE REQUIREMENTS

3.1 HARDWARE REQUIREMENTS

3.1.1 ARDUINO UNO

The Arduino Uno is a versatile and widely-used microcontroller board that forms the cornerstone of the Arduino platform. Designed with accessibility and user-friendliness in mind, it empowers electronics enthusiasts, hobbyists, and professionals to create interactive projects and prototypes. This compact and affordable board is built around the ATmega328P microcontroller, offering an array of digital and analog input/output pins, making it suitable for a broad spectrum of applications. The Arduino Uno provides an integrated development environment (IDE) for programming and a simplified variantof the Embedded C programming language. This facilitates the rapid development of custom software to control various electronic components, such as sensors, displays, motors, and communication modules. Its open-source nature encourages a thriving community of users, allowing for easy sharing of projects and code. Key features of the Arduino Uno include its ease of use, flexibility, and a vast library ofpre-written code and libraries that simplify project development. The board connects to a computer via USB, enabling users to upload their code and interact with their creations in real-time. Its compatibility with a wide range of expansion shields and accessories further extends its capabilities. Therefore the Arduino Uno is an accessibleand versatile microcontroller board, fostering creativity and innovation in the field of electronics. It has become an indispensable tool for both beginners and experienced developers, serving as a platform for experimentation, learning, and the realization of innovative projects.

FIG 3.1 ARDUINO UNO

3.1.2 FEATURES OF ARDUINO UNO

The Arduino Uno board is a popular microcontroller board with a variety of features that make it well-suited for a wide range of electronic projects.

1. Microcontroller: The Arduino Uno is based on the ATmega328P microcontroller, which has 32KB of flash memory for storing your code, 2KB of SRAM for variables, and 1KB of EEPROM for data storage.

2. USB Interface: The Arduino Uno can be easily connected to a computer via USB, making it easy to upload code and communicate with your projects.

3. LED Indicator: There's a built-in LED on pin 13 that can be used for simple debugging and as a status indicator.

4. Reset Button: A reset button allows you to restart your code or reprogram the board without unplugging it.

5. ICSP Header: The In-Circuit Serial Programming (ICSP) header provides a way to reprogram the microcontroller using an external programmer.

6. Compatibility: Arduino Uno is compatible with a wide range of expansion shields

and modules, allowing you to add functionalities such as Wi-Fi, Bluetooth, motor control, and more.

7. Open-Source Software and Community: Arduino Uno is supported by the Arduino IDE (Integrated Development Environment), an open-source software tool that simplifies coding and programming. The Arduino community provides extensive resources, libraries, and example code to help users get started with their projects.

8. Cross-Platform: The Arduino IDE is available for Windows, macOS, and Linux, making it accessible to a wide range of users.

9. Low Cost: The Arduino Uno is relatively inexpensive, making it an affordable option for both beginners and experienced developers.

10. Large User Base: Arduino Uno has a vast and active user community, which means you can find plenty of online resources, tutorials, and forums for support and inspiration.

3.1.3 SERVO MOTOR:

A servo motor, at its core, is an actuator designed to precisely control angular or rotational motion. It achieves this remarkable precision through a closed-loop control system, wherein a feedback device, such as an encoder or potentiometer, continuously informs the controller about the motor's actual position, speed, and direction. This real-time feedback allows the controller to make instantaneous adjustments, minimizing any discrepancies between the desired and actual positions. Servo motors offer versatility that extends to applications requiring exact positioning, speed control, and torque regulation. They can maintain consistent loads and are adept at tasks ranging from intricate robotics movements to demanding industrial machinery operations. This versatility is partly due to the compact size-to-torque ratio that servo motors boast, making them particularly well-suited for space-constrained environments. In operation, servo motors receive control signals, typically in the form of PWM (Pulse Width Modulation) signals. By varying the width of the pulse, users can command specific positions, speeds, or torques, allowing for fine-grained control over the motor's movements. Servo motors come in various categories, including hobby servo motors for recreational applications and industrial servo motors for heavy-duty industrial settings. Their precision and efficiency have made them indispensable in fields such as robotics, CNC machining, automation systems, camera gimbals, and much more. In conclusion, servo motors represent the pinnacle of motion control technology, enabling the precise manipulation of angles and rotational motion. Their incorporation of closed-loop control, compact design, and versatility has revolutionized numerous industries, ensuring that movements are not only precise but also adaptable to a wide array of applications, ultimately contributing to advancement of automation and precision engineering.

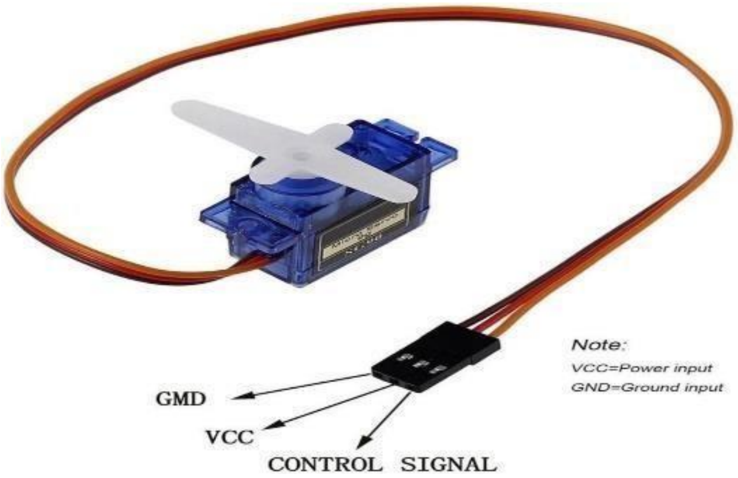

FIG 3.2 SERVO MOTOR

3.1.4 FEATURES OF SERVO MOTOR:

Servo motors are essential components in various applications where precise control of position, speed, and torque is required. These motors come with a range of features that make them suitable for a wide array of industrial, robotic, and automation applications. The key features of servo motors:

1. Precision Control: Servo motors offer extremely accurate and precise control over position, speed, and torque. Their closed-loop control system, which includes feedback devices like encoders, ensures that the motor maintains the desired position with minimal error.

2. Feedback Mechanism: Servo motors incorporate feedback devices, such as encoders or resolvers, which continuously provide information about the motor's actual position, speed, and direction. This feedback allows for real-time adjustments to ensure the motor performs as expected.

3. High Torque-to-Inertia Ratio: Servo motors are designed to provide high torque relative to their rotor inertia. This characteristic enables them to accelerate quickly and handle dynamic changes in load or speed effectively.

4. Fast Response Time: Servo motors have a fast response time, which means they can react quickly to changes in the control signal, making them suitable for applications that require rapid adjustments.

5. Variable Speed Control: They can maintain and control a wide range of speeds, making them versatile for applications that require both low-speed precision and high-speed operation.

6. High Efficiency: Servo motors are known for their efficiency. They can convert a high percentage of electrical input power into mechanical output power.

7. Low Inertia: Servo motors typically have low rotor inertia, which enables them tostart and stop quickly and handle rapid changes in direction and speed.

8. Dynamic Braking: Some servo motors are equipped with dynamic braking systems that can dissipate energy when the motor is slowing down, preventing overhauling conditions and enhancing safety.

9. Multiple Control Options: Servo motors can be controlled via various methods, including analog voltage, PWM signals, step and direction signals, and digital communication protocols like Modbus or CAN.

10. Robust and Durable: Servo motors are designed to be reliable and have a long operational life, even in demanding industrial environments.

11. Wide Range of Sizes and Types: Servo motors come in various sizes, power ratings, and types, allowing for flexibility in selecting the right motor for a specific application.

12. Compatibility with Controllers: Servo motors are typically used in conjunctionwith specialized servo motor controllers or drives that provide the necessary controland power amplification.

10. Overload Protection: Some servo motor systems have built-in overload protection to prevent damages in cases of excessive load or mechanical obstructions.

3.1.5 ESP-32

The ESP32 is a highly versatile and feature-rich microcontroller and wireless module developed by Es press if Systems. With its dual-core processor, it offers superior processing capabilities and is equipped with both Wi-Fi and Bluetooth connectivity options. Its low power consumption features, extensive GPIO pins, integrated sensors, and support for various communication interfaces make it ideal for a wide range of applications. It also provides significant flash memory for program storage and RAM for data handling. Security features like secure boot and flash encryption enhance device protection, while support for OTA updates enables remote firmware upgrades. The ESP32's adaptability, coupled with its compatibility with the Arduino IDE and Free RTOS, makes it a top choice for developers in the IoT and embedded systems realms. Its affordability and the active community surrounding it have led to its rapid adoption in both hobbyist and professional projects, cementing its position as a go-to solution for IoT and connected devices.

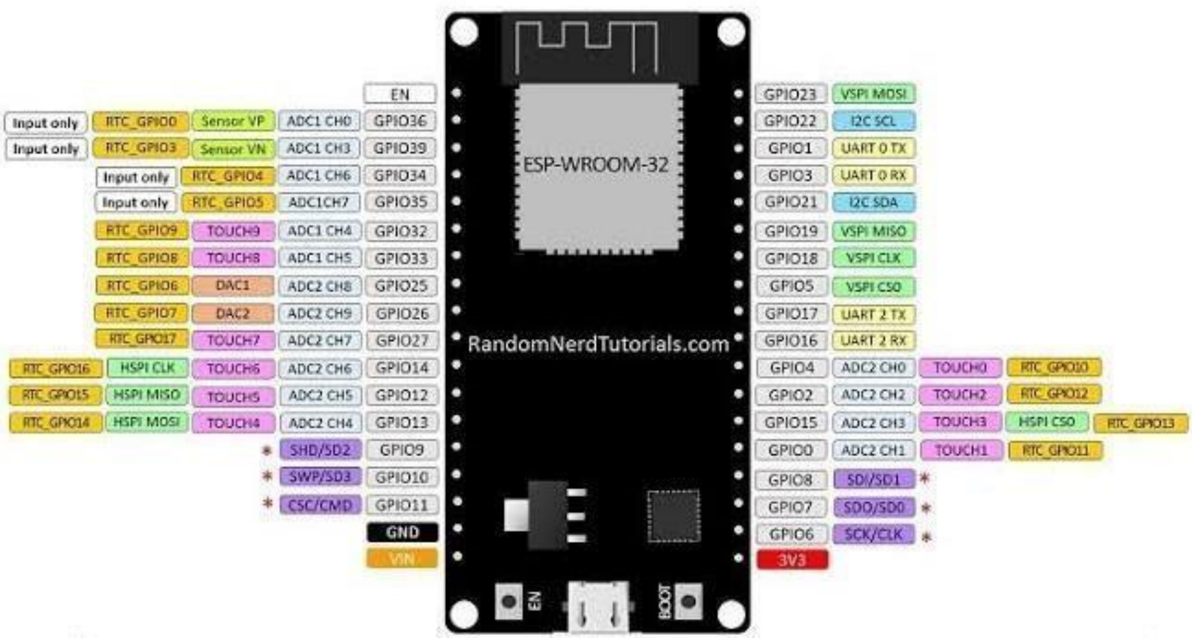

FIG 3.3 ESP32

3.1.6 SPECIFICATIONS OF USP-32

- Integrated Crystal− 40 MHz
- Module Interfaces− UART, SPI, I2C, PWM, ADC, DAC, GPIO, pulse counter, capacitive touch sensor
- Integrated SPI flash− 4 MB
- ROM− 448 KB (for booting and core functions)
- SRAM− 520 KB
- Integrated Connectivity Protocols− WiFi , Bluetooth, BLE
- On−chip sensor− Hall sensor
- Operating temperature range− −40 − 85 degrees Celsius
- Operating Voltage− 3.3V
- Operating Current− 80 mA (average)

3.1.7 BLUETOOTH MODULE

A Bluetooth module is a compact and versatile electronic component that has revolutionized wireless connectivity. It enables seamless data exchange, audio streaming, and control signals between electronic devices, eliminating the need for physical connections. These modules, adhering to Bluetooth standards such as Bluetooth Classic and Bluetooth Low Energy (BLE), are equipped with serial communication interfaces like UART for easy integration with microcontrollers. They come in various sizes, making them adaptable to a wide range of applications. The low power consumption of BLE modules has made them ideal for battery-operated devices, while Bluetooth Classic modules serve data- intensive applications. Secure pairing and data encryption are also among their features, ensuring the privacy and integrity of transmitted data. Bluetooth modules are the backbone of wireless audio devices, IoT applications, and countless other technologies, playing a pivotal role in the modern world of seamless and wireless communication.

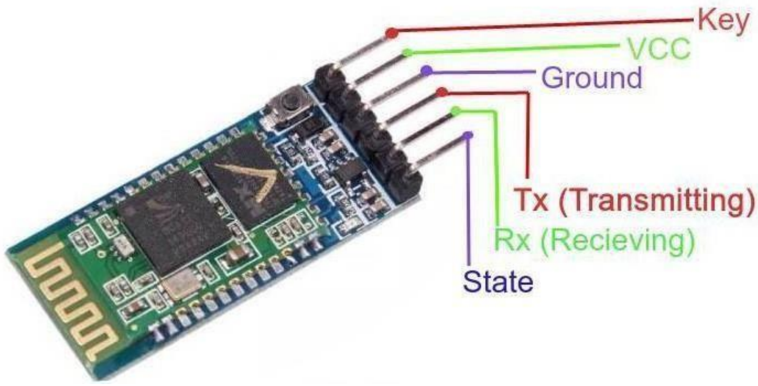

FIG 3.4 BLUETOOTH MODULE

3.1.8 MPU 6050:

The MPU-6050 is a versatile motion-tracking device renowned for its compact size and powerful capabilities. Combining a 3-axis gyroscope and a 3-axis accelerometer in a single package, it is a valuable tool in applications requiring motion sensing and orientation tracking. The MPU-6050's Digital Motion Processing (DMP) capabilities relieve the burden of complex sensor fusion computations, providing reliable motion data and quaternion outputs for orientation. Its sensitivity ranges from subtle, slow movements to rapid changes in orientation, accommodating a wide array of motion dynamics. Additionally, this module features an integrated temperature sensor, enhancing its versatility. It communicates via I2C, facilitating easy integration with popular development platforms, and is often embedded in development boards, making it accessible to both hobbyists and professionals. While calibration may be necessary for precise measurements in some cases, the MPU-6050's compatibility with sensor fusion techniques, such as magnetometers, enables the attainment of accurate 9-DoF data, propelling its application in robotics, drones, gaming, and other motion- driven projects.

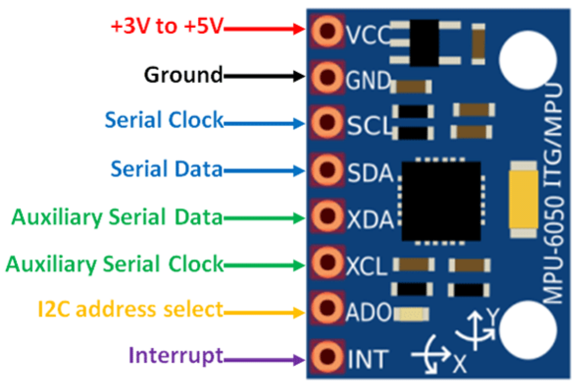

FIG 3.5 MPU 6050

3.1.9 SPECIFICATIONS OF MPU 6050:

- Module chip. MPU6050.
- Voltage Supply. 3.3 - 5 V DC.
- Logic level. 3.3 V.
- Degrees of Freedom. 6 x.
- Interface. I²C.
- Built-in Chip. 16-bit AD converter.
- Pins. 8 x.
- Pin spacing. 2.54 mm.

3.1.10 ROBOTIC ARM 3D PRINED STRUCTURE

A 3D-printed structure for a robotic arm represents a remarkable fusion of cutting-edge technology and practical engineering. 3D printing technology allows for the creation of intricate, customized, and lightweight components with high precision. When applied to the construction of robotic arms, it offers several advantages. The flexibility in design allows engineers to optimize the structure for strength, weight, and functionality. Additionally, 3D printing allows for the rapid production of prototypes and iterations, significantly reducing development time and costs. The use of lightweight materials can enhance the overall performance of the robotic arm, enabling it to be more agile and energy-efficient.Further more, the ability to tailor the structure to specific tasks and end-effectors makes 3D-printed robotic arms versatile tools, finding applications in industries such as manufacturing, healthcare, and research. The fusion of 3D printing and robotics exemplifies the innovation and potential of additive manufacturing in pushing the boundaries of automation and technological advancement.

FIG 3.6 ROBOTIC ARM STRUCTURE

3.2 SOFTWARE REQUIREMENTS

3.2.1 ARDUINO IDE

The development of electronics is now easier thanks to Arduino software (IDE), and Arduino boards (hardware). This set help to build digital and interactive deviceswith the help of other components. In Previous article we talk about Arduino boards. In this article we will recognize what is Arduino software (IDE), and how use it. The Arduino software (IDE) is an open source software, which is used to program the Arduino boards, and is an integrated development environment, developed by arduino.cc. Allow to write and upload code to Arduino boards. And it consist of many libraries and a set of examples of mini projects. The Arduino software is easy to use for beginners, or advanced users. It uses to get started with electronics programming and robotics, and build interactive proto types. So Arduino software is a tool to develop new things. and create new electronic projects, by Anyone (children, hobbyists, engineers, programmers, … etc). The Arduino IDE is an open-source software, which is used to write and upload code tothe Arduino boards. The IDE application is suitable for different operating systems such as **Windows, Mac OS X, and Linux**. It supports the programming languagesC and C++. Here, IDE stands for **Integrated Development Environment**'s program or code written in the Arduino IDE is often called as sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino.' TheArduino IDE will appear as:

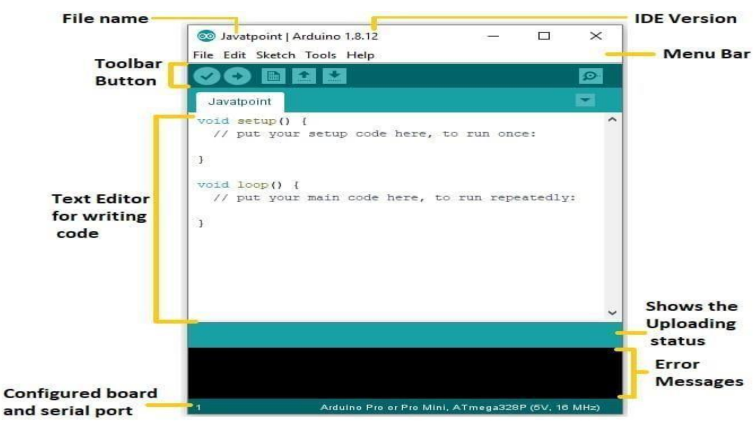

FIG 3.7 ARDUINO IDE

CHAPTER 4

RESULT AND CONCLUSION

4.1 RESULT

FIG 4.1 FINAL ARM MOUNTED WITH SERVO MOTORS

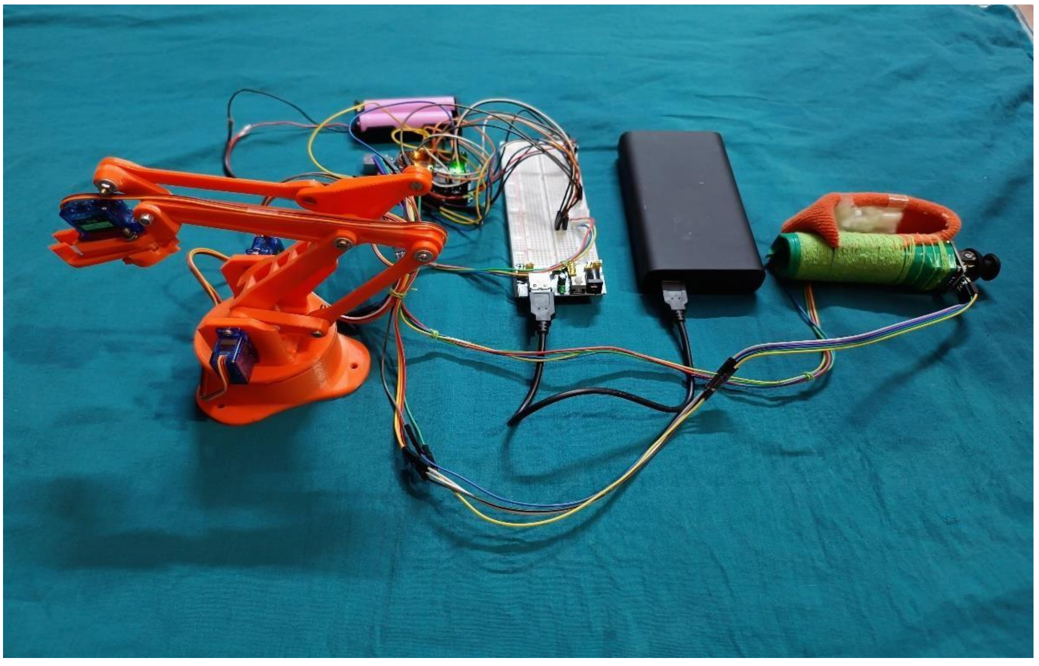

FIG 4.2 The glove unit holds the mpu 6050 sensor and gives the signal to arm structure, with help of this glove holded in the hand it can be tilted and the arm automatically receives the signal through Arduino and moves the arm.

4.2 CONCLUSION

In conclusion, the robotic arm project has demonstrated the successful integration of mechanical, electrical, and software components to create a functional and versatile robotic arm. Throughout the project, we have achieved several key milestones, including precise motion control, object manipulation, and even automation through programming. This project not only highlights the potential of robotics in various industries but also serves as a valuable learning experience in engineering, automation, and problem-solving. As we look ahead, further enhancements and applications can be explored, paving the way for more advanced and specialized robotic arm systems in the future.

FUTURE SCOPE

Robotic Arms have a wide scope of development. Shortly the arms will be able to perform every task as humans and in a much better way. Brain Computer Interface can be used to acquire signals from the human brain and control the arm. The system can work in the same way as a human arm. Robotic arms have a wide scope of development. They can perform every task as humans and in a much better way. For example, robotic arms can use brain computer interface to acquire signals from the human brain and control the arm grip wafers used to manufacture, computer processors detect where an object is move things from one place to another place or pick objects that are away from the user. Robots will increase economic growth and productivity and create new career opportunities for many people worldwide. However, there are still warnings about massive job losses, forecasting losses of 20 million manufacturing jobs by 2030, or how 30% of all jobs could be automated by 2030.

REFERENCE

1.Hu, Zheng; McCauley, Raymond; Schaeffer, Steve; Deng, Xinyan (May 2009). "Aerodynamics of dragonfly flight and robotic design". 2009 IEEE International Conference on Robotics and Automation. pp. 3061–3066. doi:10.1109/ROBOT.2009.5152760. ISBN 978-1-4244-2788-8. S2CID 12291429.

2.Billing, Rius; Fleischner, Richard (2011). "Mars Science Laboratory Robotic Arm" (PDF). 15th European Space Mechanisms and Tribology Symposium 2011. Retrieved 2012-08-21.

3.Pollock, Emily (7 June 2018). "Construction Robotics Industry Set to Double by 2023". engineering.com. Archived from the original on 2020-08-07. Retrieved 2018-12-03.

4.Staff (Sandia National Labs) (August 16, 2012), "Life-like, cost-effective robotic hand can disable IEDs", *R&D Magazine*, rdmag.com, retrieved September 13, 2012.

5."History of Industrial Robots" (PDF). Archived from the original (PDF) on 2012-12-24. Retrieved 2012-10-27.

6."Robots then and now". *BBC*. 22 July 2004. Archived from the original on 20 December 2010.

7."Paper on Space Robotics, pg 9" (PDF). Archived from the original (PDF) on 2017-11-16. Retrieved 2007-04-09.

8.To Get Some Knowledge About Previous Robotic Arms. https://ieeexplore.ieee.org/document/7479938.https://www.learnrobotics.org/blog/simple-smart-robotic-arm-using-arduino/

9.To Learn About Motors and Their Uses and How It Works

https://www.modmypi. com/blog/whats-the-difference-betweendc-servo- stepper-motors

10. "MeArm Open Source Robot Arm (source files)". Retrieved 21 June 2016

APPENDICES

```
#include<Servo.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <Wire.h>
Adafruit_MPU6050 mpu;
int servoPin1 = 11;
int servoPin2 = 9;
int servoPin3 = 6;
int servoPin4 = 3;
int xPin = A1;
int  xValue;
int yPin = A0;
int yValue;
//int potPin = A0;
int servoPosB = 160;
int servoPosFr = 130;
int servoPosG = 60;
int servoPosA = 100;
//float potValue;
Servo myServo1;
Servo myServo2;
Servo myServo3;
Servo myServo4;

void setup() {
  // put your setup code here, to run once:
pinMode(xPin,INPUT);
pinMode(yPin,INPUT);
Serial.begin(115200);
  // Try to initialize!
```

```
  if (!mpu.begin()) {
    Serial.println("Failed to find MPU6050 chip");
    while (1) {
      delay(10);
    }
  }
  Serial.println("MPU6050 Found!");
  // set accelerometer range to +-8G
  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  // set gyro range to +- 500 deg/s
  mpu.setGyroRange(MPU6050_RANGE_500_DEG);
  // set filter bandwidth to 21 Hz
  mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);
  delay(100);
myServo1.attach(servoPin1);
myServo2.attach(servoPin2);
myServo3.attach(servoPin3);
myServo4.attach(servoPin4);
//pinMode(potPin,INPUT);
}
 void loop() {
  // put your main code here, to run repeatedly:
  sensors_event_t a, g, temp;
  mpu.getEvent(&a, &g, &temp);
  xValue=analogRead(xPin);
  /* Print out the readings */
  Serial.print("Acceleration X: ");
  Serial.print(a.acceleration.x);
  Serial.print(", Y: ");
  Serial.print(a.acceleration.y);
```

```
Serial.print(", Z: ");
Serial.print(a.acceleration.z);
Serial.println(" m/s^2");

if(a.acceleration.x > 3.25){
  servoPosFr=servoPosFr+10;
}
if(a.acceleration.x < -3.25){
  servoPosFr=servoPosFr-10;
}
if(a.acceleration.y > 3.25){
  servoPosB=servoPosB+10;
}
if(a.acceleration.y < -3.25){
  servoPosB=servoPosB-10;
}
if(servoPosFr > 160){
  servoPosFr=160;
}
if(servoPosFr < 10){
  servoPosFr=10;
}
if(servoPosB > 180){
  servoPosB=180;
}
if(servoPosB < 0){
  servoPosB=0;
}
if(xValue > 650){
servoPosA=servoPosA+10;
}
```

```
  if(xValue < 350){
  servoPosA=servoPosA-10;
  }
  if(servoPosA > 180){
    servoPosA = 180;
  }
  if(servoPosA < 70){
    servoPosA = 70;
  }
  //Serial.print(" Rotation X: ");
  //Serial.print(g.gyro.x);
  //Serial.print(", Y: ");
  //Serial.print(g.gyro.y);
  //Serial.print(", Z: ");
  //Serial.print(g.gyro.z);
  //Serial.println(" rad/s");
  //Serial.print("Temperature: ");
  //Serial.print(temp.temperature);
  //Serial.println(" degC");
  //Serial.println("");
  delay(100);
//potValue=analogRead(A0);
//servoPos=int((.175/.1023)*(potValue/10));
//Serial.print("potValue = ");
//Serial.print(potValue);
//Serial.print("    ");
//Serial.print("servo position = ");
//Serial.println(servoPosB);
myServo1.write(servoPosB);
myServo2.write(servoPosFr);
myServo3.write(servoPosG);
```

```
mySservo4.write(servoPosA);
//delay(200);
}
```