Sushila Devi Bansal College of Technology Indore

Minor Project Report

on

“…….Worship Robot…..……”

Submitted by

Gourav Kushwah(0829EC201009)

Owes Niyazi(0829EC201016)

Shrishti Tayde(0829EC201023)

Submitted to

Department of Electronics and Communication Engineering

MAY 2023

‘OGASHI – A Worship Robot’

Session 2023

A project report submitted to ***Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (M.P.)*** towards the partial fulfilment of the degree of ***Bachelor of Engineering*** in **Electronics and Communication Engineering**

By

Gourav Kushwah (0829EC201009)
Owes Niyazi (0829EC201016)
Shrishti Tayde(0829EC201023)

Under the guidance of
Dr. Neelam Sharma

Sushila Devi Bansal College of Technology Indore

Department of Electronics and Communication Engineering

Sushila Devi Bansal College of Technology Indore

Session 2023-24

Recommendation

The Project report entitled **"OGASHI- a worshiping robot"** submitted by **'Gourav Kushwah, Owes Niyazi, and Shrishti Tayde'** towards the partial fulfilment of degree of **Bachelor of Engineering in Electronics and Communication Engineering** of Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal is satisfactory account of the progress made by them in project work.

Guided By

Dr. Neelam Sharma

Head of Department

Dr. Neelam Sharma

Sushila Devi Bansal College of Technology Indore

Session 2023-24

Certificate

It is certified that this Project report entitled **"OGASHI- a worshiping robot"** submitted by **'Gourav Kushwah, Owes Niyazi, and Shrishti Tayde'** towards the partial fulfilment of degree of **Bachelor of Engineering in Electronics and Communication Engineering** of Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal is satisfactory account of his work based on syllabus and is approved for the award.

Internal Examiner

Date

External Examiner

Date

ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to **DR. Neelam Sharma,** our project head, for her constant guidance and support throughout the project. Her invaluable insights and expert advice have been instrumental in bringing this project to fruition. We would also like to extend our sincere thanks to **Prof. Divya Dubey**, our project coordinator, for her tireless efforts in coordinating and supervising the project. Her hands-on approach and meticulous attention to detail have been critical in ensuring the project's success. We would further like to acknowledge the invaluable contribution made by **Mr. Yogesh Gupta**, our minor lab technician, who has supported the project in numerous ways by providing timely assistance and expert technical support. Finally, we would also like to thank all the individuals who have contributed to this project's success, including our friends and family members who have supported us during the project's duration. Thank you all for being an essential part of this project's success

CONTENT

Chapter **Page No.**

Abstract:

It is a type of humanoid robot basically designed to perform the act of pooja or aarti. It rotates the worship plate with one hand and rings the bell with the other in the initial stage. It rotates its hand with the help of servo motors and the base of the robot move with help of DC motors. The body and the base of the robot is made up of a thin sheet of aluminum and the head of the robot is of plastic it also has two LEDs in its eyes and a servo motor which helps in the movement of the head too. This is the initial tasks which this robot can perform but in future this robot will eventually work as a AI based Humanoid Robot.

List OF Figures

List Of Tables

CHAPTER: 1
INTRODUCTION

Introduction :

Introducing the newest addition to the world of technology, a worship robot that transcends beyond the mundane machines! Our worship robot is designed to cater to the spiritual needs of individuals who seek a deeper connection with a higher power. With its advanced AI system programmed to provide insight into various beliefs and practices, this robot is your perfect companion for soulful guidance.

The worship robot has been built to emulate human emotions, with its expressive features ensuring a comforting and relatable experience for users. Whether you're seeking solace in prayer, looking for rituals to perform, or simply in need of a word of inspiration, our worship robot is equipped with a comprehensive database of religious practices from across the globe.

With its ability to adapt to any language, culture, and belief, the worship robot is truly unique. The perfect solution for individuals seeking a personal spiritual journey, it is a testament to the power of technology in bringing us closer to the divine. We invite you to join us in welcoming this innovative creation into your homes and hearts.

CHAPTER: 2
SYSTEM MODEL

Simulation Circuit

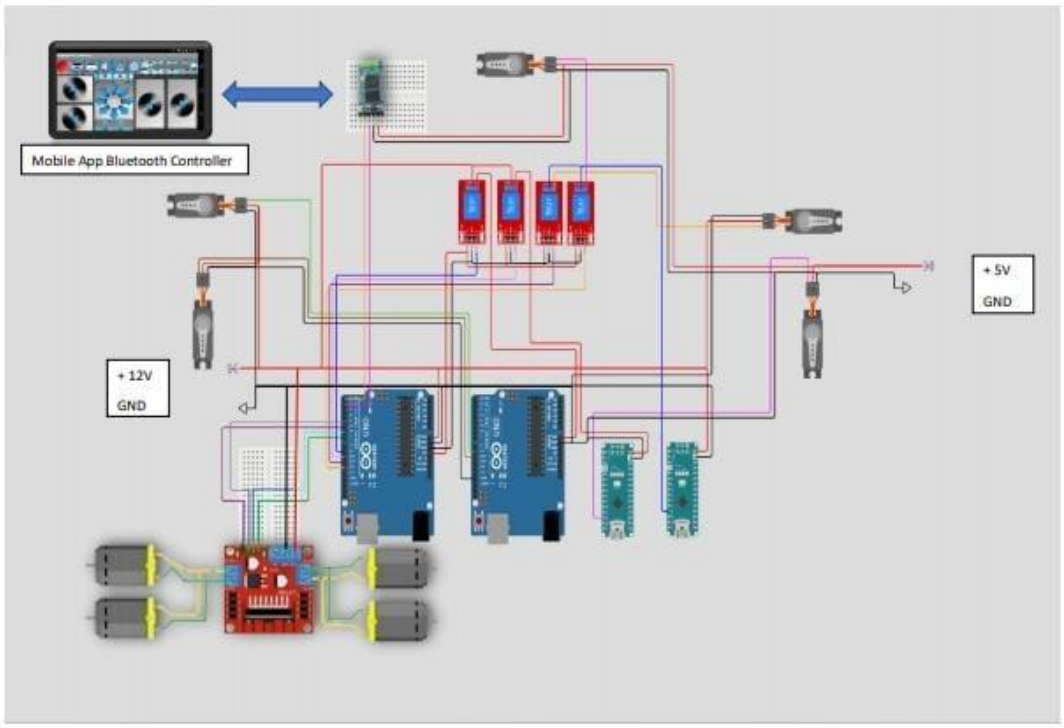


FIG.1.

Prototype Model

FIG. 2.

Final Model

FIG. 3.

CHAPTER: 3 DISCRIPTION AND WORKING OF THE CIRCUIT

Description Of Ogashi

The worship robot is designed to cater to the spiritual needs of individuals in a personalized manner. Its advanced AI system is programmed to respond to voice commands and provide users with the information they seek on spirituality, beliefs, and religious practices.

The robot has multiple features that work together to provide a holistic spiritual experience. These include a comprehensive database of religious texts, prayers, and rituals from various belief systems, an expressive voice system that mimics human emotions, and the ability to adapt to any language and culture.

To use the worship robot, users can simply initiate a conversation and give voice commands on the specific religious topic or practice they wish to learn or perform. The robot provides step-by-step guidance and can even lead users in prayer.

Through its advanced technology, the worship robot provides a unique and personalized experience for individuals seeking a deeper connection with a higher power. Its relatable and comforting interface, combined with its powerful database, makes it the perfect spiritual companion for people of all faiths.

Working Of Ogashi

(Working Model of OGASHI)

FIG.4.

The working of a worship robot involves a combination of hardware and software components that come together to provide users with personalized spiritual guidance. Here's a general overview of how a worship robot works:

1. User interaction: The worship robot interacts with users through its various sensors such as microphones, cameras, and touch sensors. Users can interact with the robot through voice commands, gestures, or touch inputs.

2. Data Collection: Based on the user interaction, the robot's software collects data on the user's beliefs, preferences, and requirements to provide personalized guidance. It uses advanced algorithms and machine learning techniques to analyze this data and create an understanding of the user's spiritual journey.

3. Synthesis of Information: The worship robot uses a vast database of information on different religions, belief systems, and practices to provide users with personalized feedback. It synthesizes information from various sources and provides users with customized guidance relevant to their specific requirements.

4. Real-time Feedback: The robot provides users with real-time feedback on their spiritual practices, helping them to achieve their goals and deepen their understanding of their faith. It may provide audio, video, or text-based feedback to the user.

5. Maintenance and Upgrades: The worship robot requires regular maintenance and upgrades to keep it running smoothly. The manufacturer or service provider may provide maintenance and upgrade services to ensure optimal performance.

In summary, the worship robot works by using sensors, software, and a vast database of spiritual information to provide personalized guidance to users on their spiritual journey. Its intuitive AI technology and machine learning capabilities make it a powerful tool for facilitating spiritual growth and deepening the connection with one's faith.

FLOW CHARTS

- **Flow chart for working process of the**

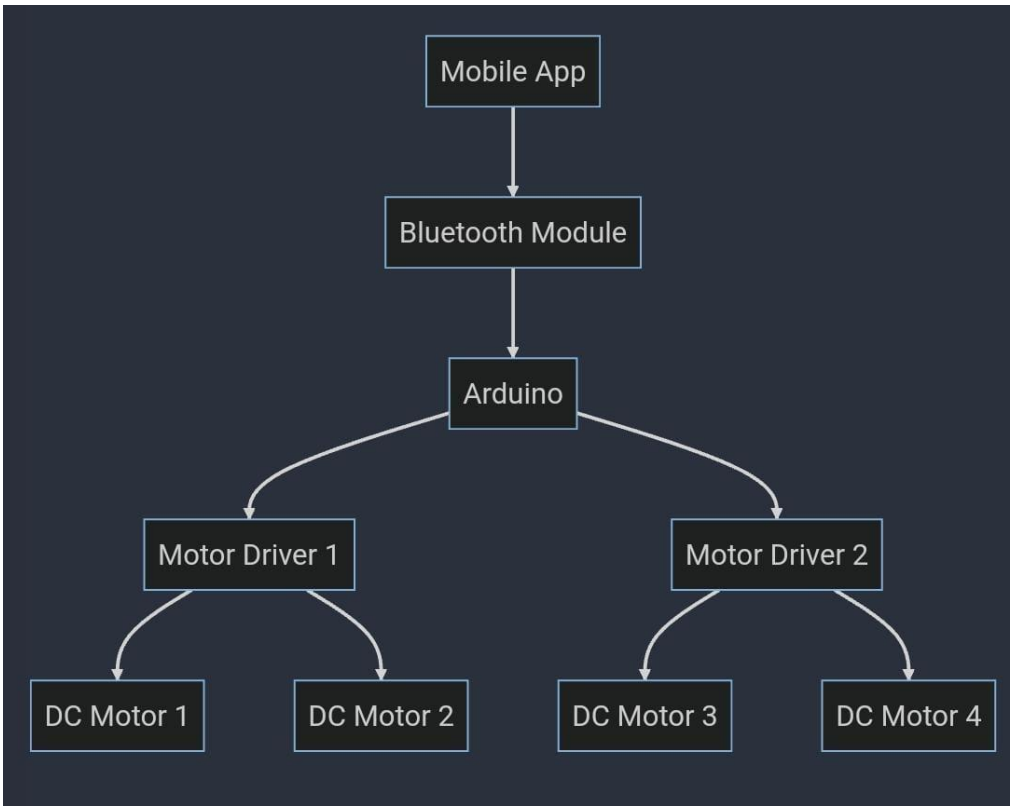


FIG.5.

- **Flow chart of working of hands**

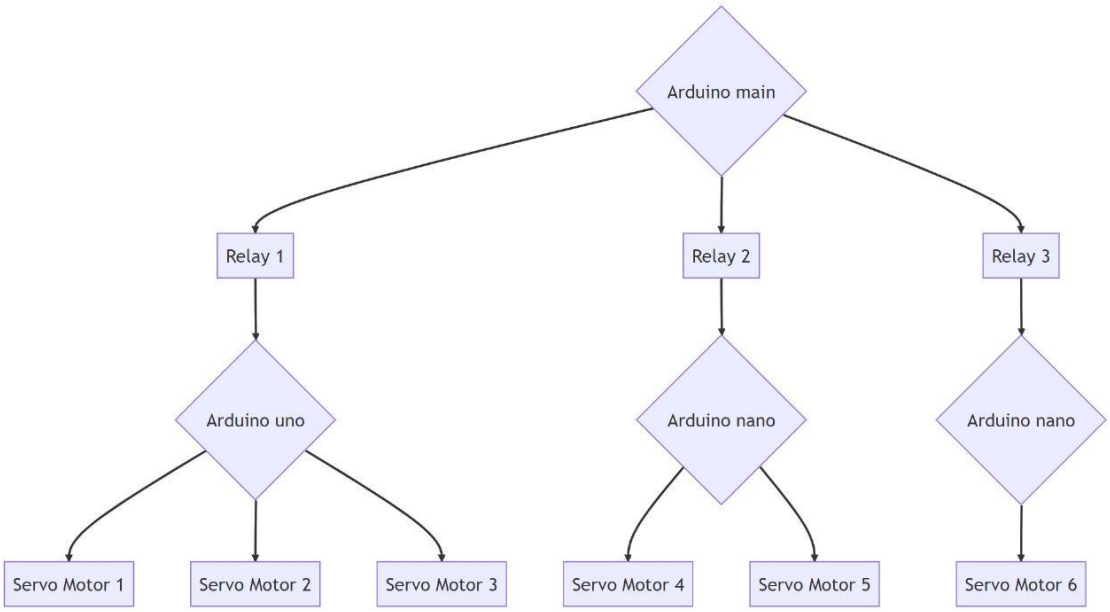


FIG.6.

BLOCK DIAGRAM:

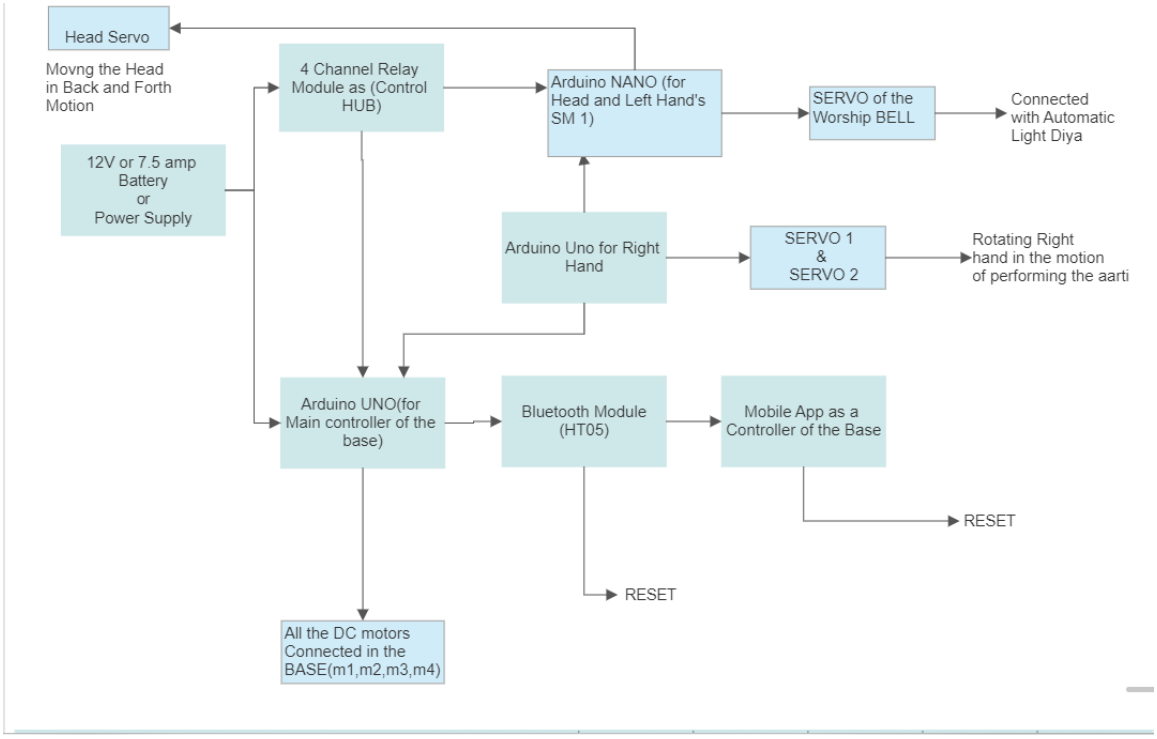


FIG. 7.

CHAPTER: 4
HARDWARE DISCRIPTION

Metal Bars and Sheets:

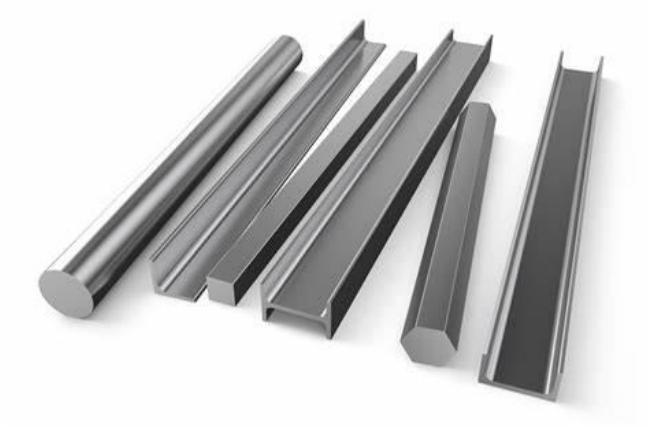


FIG. 8.

FIG.9.

For this robot's body, we are using aluminum bars and sheet for support and structure as the metal.

#Benefits of using aluminum over other materials:

To robotics hobbyists, aluminum is one of the most important materials you will use. It's strong, lightweight, resistant to corrosion, affordable, and most importantly, easy to manipulate.

Strength: While aluminum isn't as strong as steel, it has a much higher strength-to-weight ratio. What this means is, for a mass of aluminum and an equal mass of steel, aluminum would be stronger, additionally, because of that ratio, you would need less aluminum than steel to achieve the same strength levels.

Corrosion Resistant: One of the most interesting parts of aluminum is that it has an inherent resistance to rust. When exposed to the elements it forms an oxidized layer that acts as a protective coating against any further rust. However, there are some occasions where rust can still occur and potentially present you with a problem. Physical wear over time will wipe off that protective rust layer and cause oxidation to slowly eat away at the metal. There are several ways to permanently prevent oxidation which can be easily found online.

High Heat Tolerance: Almost all heat sinks created are made out of aluminum. This is because aluminum conducts heat extremely well. The only other metal that's as easily available that has a higher thermal conductivity than aluminum is copper.

Servo Motor

FIG.10.

This servomotors are used in very varied applications where precision is required. This type of servomotor is a very common one, stronger than the S3003, and you can use it to build your 3D printer or CNC or in teletelehandlerhines to move the front wheels into curves.
These actuators consist of a DC motor, a motor-driven potentiometer measuring the angle at which it rotates, a circuit that compares the signal from the potentiometer to the user input, and a gear mechanism that reduces the speed engine, but increases its torque.
Using the information in the datasheet, you can calculate why the signal is needed to rotate the engine to a certain number of degrees.
An advantage to the stepper motors is that when they are idle, the servo motors do not consume current, but they can not keep the shaft locked.

Specification

- Operating voltage: 4.8 V to 7.2 V;
- Current: 4.8V to 8.8mA, 6V to 9.1mA (empty);
- Stall current: 4.8V to 350 mA, 6V to 450 mA;
- Speed: 0.2 s / 60º (4.8V), 0.16 s / 60º (6V);
- Couple: 8.5 kgf · cm (4.8V), 10 kgf · cm (6V).
- Dimensions: 40.7 x 19.7 x 42.9 mm.

DC Motor

FIG. 11.

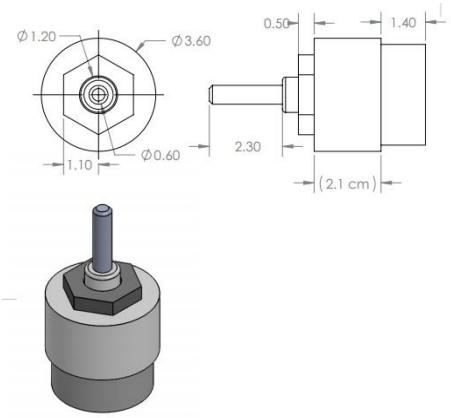
This DC Motor with Metal Gear Head is generally used in various roboticsapplications , It hasthe following electrical and mechanical specifications.

Specifications:

Motor Type	**:**	DC with Gear Box, Metal Gears
Base Motor	**:**	DC 3000 RPM
Shaft Type	**:**	Circular 6mm Diam with Internal Hole for coupling, 23 mm shaft Length
MaximumTorque	**:**	~ 3.5 Kg-cm at 12V-60RPM
RPM	**:**	60 RPM at 12V
Weight	**:**	145 Gms
Max Load Current	**:**	~350mA at 12V-60RPM

TABLE. 1

Dimensional Drawing :



Wheels

FIG. 12.

These Robot Wheel 7cm x 4cm Width for DC Motors can be easily installed on any type of geared DC motor. This is Robot Wheel with 7a cm Diameter and 4cm Width. These Wheelsare mostly used for the small roare bot, easy to mount,, durable and cheap. These wheels have a 6mm hole for a sat with the screw for fitting making it very easy to mount on motors.

Features :

- Smooth surface
- Durable
- Lightweight
- Screw for fastening on motor shaft
- Made from virgin plastic.

Specifications:

Shaft Diameter (mm)	6
Wheel Diameter(mm)	70
Wheel Width(mm)	40
Weight (gm)	45

TABLE. 2

❑ Arduino UNO

FIG. 13.

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller.

TECHNICAL SPECIFICATION:

- Operating Voltage: 5 Volts
- Input Voltage: 7 to 20 Volts
- Digital I/O Pins: 14
- PWM Pins: 6 (Pin # 3, 5, 6, 9, 10, and 11)
- UART: 1, I2C: 1, SPI: 1
- Analog Input Pins: 6
- DC Current per I/O Pin: 20 mA, DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB is used the by bootloader
- SRAM: 2 KB, EEPROM: 1 KB
- Clock Speed: 16 MHz
- Length: 68.6 mm
- Width: 53.4 mm
- Weight: 25 g
- ICSP Header: Yes
- Power Sources: DC Power Jack, USB Port, and the VIN pin (+5 volt only)

Arduino NANO

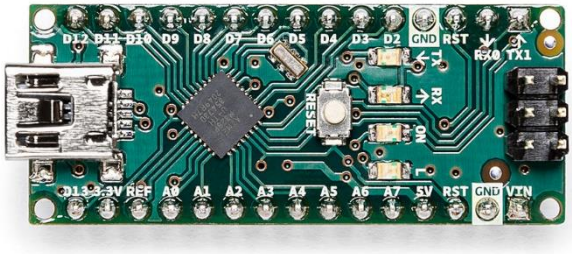


FIG. 14.

- The Arduino Nano is another popular Arduino development board very much similar to the Arduino UNO. They use the same Processor (Atmega328p) and hence they both can share the same program. This board is used so that the communication between both the boards can be handled accurately and by that, the precision and accuracy of the robot can also be controlled.

Board	Name	Arduino Nano
	SKU	A000005
Microcontroller	ATmega328	
USB connector	Mini-B USB	
Pins	Built-in LED Pin	13
	Digital I/O Pins	14
	Analog input pins	8
	PWM pins	6
Communication	UART	RX/TX
	I2C	A4 (SDA), A5 (SCL)

	SPI	D11 (COPI), D12 (CIPO), D13 (SCK). Use any GPIO for Chip Select (CS).
Power	I/O Voltage	5V
	Input voltage (nominal)	7-12V
	DC Current per I/O Pin	20 mA
Clock speed	Processor	ATmega328 16 MHz
Memory	ATmega328P	2KB SRAM, 32KB flash 1KB EEPROM
Dimensions	Weight	5gr
	Width	18 mm
	Length	45 mm

TABLE. 3

Bluetooth Module

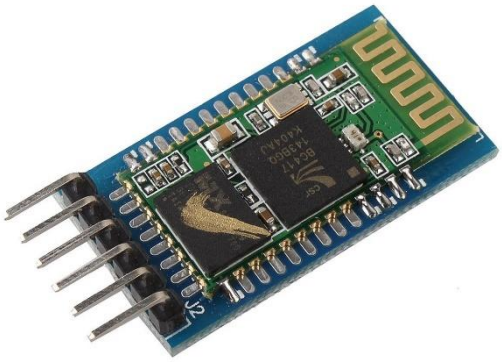


FIG. 15.

The main job of the HC-05 is adding two-way (full-duplex) wireless functionality to our project. It can be used to communicate between two microcontrollers with serial capabilities (like two Arduinos), but it can also be used to control any Bluetooth device with a microcontroller or vice versa.

The HC-05 gets controlled via the TX and RX pins and supports the use of standard AT commands. For that purpose, users must enter a special command mode when the device powers up. This is done by pulling the key pin low when turning the module on. Otherwise, the device boots into data mode, which allows it to communicate with other devices wirelessly.

SPECIFICATIONS

- Length: 28 mm (1 in)
- Width: 15 mm (â…” in)
- Height: 2.35 mm (0.1 in)
- Supply voltage: 3.3V to 6.0V
- Operating voltages: 3.3V (all other pins, except VCC)
- Working current: 30mA
- Operating range: max. 10m (33 ft)
- Default password: 0000 or 1234 (depends on model/manufacturer)
- Supported baud rate: 9600,19200,38400,57600,115200,230400,460800
- Follows IEEE 802.15.1

Relay Module

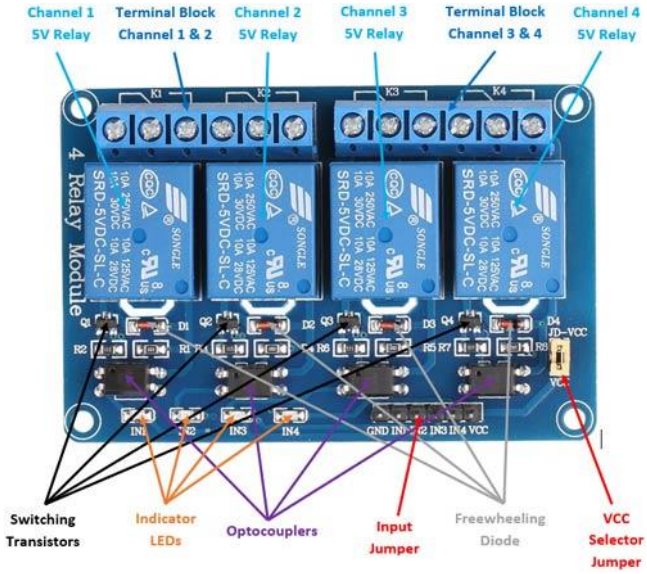


FIG. 16.

The **four-channel relay module** contains four 5V relays and the associated switching and isolating components, which makes interfacing with a microcontroller or sensor easy with minimum components and connections. The contacts on each relay are specified for 250VAC and 30VDC and 10A in each case, as marked on the body of the relays.

Four-Channel Relay Module Specifications

- Supply voltage – 3.75V to 6V
- Trigger current – 5mA
- Current when the relay is active - ~70mA (single),
  ~300mA (all four)
- Relay maximum contact voltage – 250VAC, 30VDC
- Relay maximum current – 10A

L298 Motor Driver Module

FIG. 17.

This **L298N Motor Driver Module** is a high power motor driver module for driving DC and Stepper Motors. This module consists of an L298 motor driver IC and a 78M05 5V regulator. **L298N Module** can control up to 4 DC motors, or 2 DC motors with directional and speed control.

L298N Module Pinout Configuration

Pin Name	Description
IN1 & IN2	Motor A input pins. Used to control the spinning direction of Motor A
IN3 & IN4	Motor B input pins. Used to control the spinning direction of Motor B
ENA	Enables PWM signal for Motor A
ENB	Enables PWM signal for Motor B
OUT1 & OUT2	Output pins of Motor A
OUT3 & OUT4	Output pins of Motor B

12V	12V input from DC power Source
5V	Supplies power for the switching logic circuitry inside L298N IC
GND	Ground pin

TABLE. 4

Features & Specifications

- Driver Model: L298N 2A
- Driver Chip: Double H Bridge L298N
- Motor Supply Voltage (Maximum): 46V
- Motor Supply Current (Maximum): 2A
- Logic Voltage: 5V
- Driver Voltage: 5-35V
- Driver Current:2A
- Logical Current:0-36mA
- Maximum Power (W): 25W
- Current Sense for each motor
- Heatsink for better performance
- Power-On LED indicator

Power Supply

FIG. 18.

In this project we are using Chloride Sealed Lead Acid Battery , which is more beneficial than any other power supply as the many motors used in this project recquires a high amount of power supply simultaneously.

Specifications:

Output Voltage: 12V

Initial Current : 1.4 amp

Current: 7amp

Constant Voltage regulation : 27 degree

Worship Plate

FIG. 19.

Worship Bell

FIG. 20.

Battery Diya

FIG. 21.

Connecting Wires

FIG. 22.

Specification Of Ogashi

Robotic Part	Weight	Dimension
Right hand(total) (from Shoulder to Elbow) (from Elbow to Wrist)	197gm 98gm 99gm	34.5cm 18cm 14cm
Left hand(total) (from Shoulder to Elbow) (from Elbow to Wrist)	154gm 76gm 78gm	33cm 18cm 14cm
Neck	72gm	13.5cm
Base		38cm X 32cm X 3mm
Wheel	100gm (each)	7X4X3 cm
DC Motor	125gm(each)	20X15X4 cm
Servo Motor	57gm(each)	
Supporting Pillars		45cm
Shoulder		38cm
Head		
Battery	2006gm	
Worship Thali, Worship bell	42gm, 48gm	14.5cm(diameter),10cm(length)

TABLE. 5

CHAPTER: 5

SOFTWARE DISCRIPTION

Software Description

The Program written for different Parts of the Robot are as follows:

//Code of Head & Left hand of Robot

```
#include <Servo.h>

Servo myservo;  // create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0;    // variable to store the servo position

void setup() {
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}

void loop() {
  for (pos = 5; pos <= 100; pos += 1) { // goes from 0 degrees to 180 degrees
    // in steps of 1 degree
    myservo.write(pos);              // tell servo to go to position in variable 'pos'
    delay(40);                       // waits 15 ms for the servo to reach the position
  }
  for (pos = 100; pos >= 5; pos -= 1) { // goes from 180 degrees to 0 degrees
    myservo.write(pos);              // tell servo to go to position in variable 'pos'
    delay(40);                       // waits 15 ms for the servo to reach the position
  }
}
```

// Robot Base Code with Relay Module Controller code.

```
#define light_FR   9    //Front Light   Digital pin 9
#define light_RE   10   //Rear Light   Digital pin 10
#define horn_BUZZ  11   //Horn    Digital pin 11
#define light_EX   12   //Extra Light    Digital pin 12

//L293 Connection
  const int motorAspeed  = 3;
  const int motorA1      = 4;
  const int motorA2      = 5;
  const int motorB1      = 6;
  const int motorB2      = 7;
  const int motorBspeed  = 8;

  int command;
  int vSpeed=200;     // Default speed, from 0 to 255

  boolean lightFront = false;
```

```
  boolean lightRear  = false;
  boolean hornBUZZER   = false;
  boolean lightExtra = false;

void setup() {
    // Set pins as outputs:
    pinMode(light_FR, OUTPUT);
    pinMode(light_RE, OUTPUT);
    pinMode(horn_BUZZ, OUTPUT);
    pinMode(light_EX, OUTPUT);

    pinMode(motorA1, OUTPUT);
    pinMode(motorA2, OUTPUT);
    pinMode(motorB1, OUTPUT);
    pinMode(motorB2, OUTPUT);

    Serial.begin(9600);

  }

/********Forward*********/
  //If state is equal with letter 'F', car will go forward!
void goAhead(){
  digitalWrite (motorA1,LOW);
  delay(1);
  digitalWrite(motorA2,HIGH);
  delay(1);

  digitalWrite (motorB1,HIGH);
  delay(1);
  digitalWrite(motorB2,LOW);

  analogWrite (motorAspeed, vSpeed);
  analogWrite (motorBspeed, vSpeed);
  }
  /*******Forward Left*********/
  //If state is equal with letter 'I', car will go forward left
void goAheadLeft(){
  digitalWrite (motorA1,LOW);
  delay(1);
  digitalWrite(motorA2,HIGH);
  delay(1);

  digitalWrite (motorB1,LOW);
  delay(1);
  digitalWrite(motorB2,LOW);

  analogWrite (motorAspeed, vSpeed);
  analogWrite (motorBspeed, 0);
    }
```

```
 /*******Forward Right*********/
  //If state is equal with letter 'G', car will go forward right
void goAheadRight(){
  digitalWrite (motorA1,LOW);
  delay(1);
  digitalWrite(motorA2,LOW);
  delay(1);

  digitalWrite (motorB1,HIGH);
  delay(1);
  digitalWrite(motorB2,LOW);

  analogWrite (motorAspeed, 0);
  analogWrite (motorBspeed, vSpeed);
    }
  /********Backward*********/
  //If state is equal with letter 'B', car will go backward
void goBack(){
  digitalWrite (motorA1,HIGH);
  delay(1);
  digitalWrite(motorA2,LOW);
  delay(1);

  digitalWrite (motorB1,LOW);
  delay(1);
  digitalWrite(motorB2,HIGH);

  analogWrite (motorAspeed, vSpeed);
  analogWrite (motorBspeed, vSpeed);
    }
  /*******Backward Left*********/
  //If state is equal with letter 'J', car will go backward left
void goBackLeft(){
  digitalWrite (motorA1,LOW);
  delay(1);
  digitalWrite(motorA2,LOW);
  delay(1);

  digitalWrite (motorB1,LOW);
  delay(1);
  digitalWrite(motorB2,HIGH);

  analogWrite (motorAspeed, 0);
  analogWrite (motorBspeed, vSpeed);
    }
  /*******Backward Right*********/
  //If state is equal with letter 'H', car will go backward right
void goBackRight(){
  digitalWrite (motorA1,HIGH);
```

```
  delay(1);
  digitalWrite(motorA2,LOW);
  delay(1);

  digitalWrite (motorB1,LOW);
  delay(1);
  digitalWrite(motorB2,LOW);

  analogWrite (motorAspeed, vSpeed);
  analogWrite (motorBspeed, 0);
    }
  /**********Left**********/
  //If state is equal with letter 'L', wheels will turn left
void goLeft(){
  digitalWrite (motorA2,HIGH);
  delay(1);
  digitalWrite(motorA1,LOW);
  delay(1);

  digitalWrite (motorB2,HIGH);
  delay(1);
  digitalWrite(motorB1,LOW);

  analogWrite (motorAspeed, vSpeed);
  analogWrite (motorBspeed, vSpeed);
    }
  /**********Right**********/
  //If state is equal with letter 'R', wheels will turn right
void goRight(){
  digitalWrite (motorA2,LOW);
  delay(1);
  digitalWrite(motorA1,HIGH);
  delay(1);

  digitalWrite (motorB2,LOW);
  delay(1);
  digitalWrite(motorB1,HIGH);

  analogWrite (motorAspeed, vSpeed);
  analogWrite (motorBspeed, vSpeed);
    }

     /**********Stop**********/
  //If state is equal with letter 'S', stop the car
void stopRobot(){
   analogWrite(motorA1, 0);          analogWrite(motorA2, 0);
   analogWrite(motorB1, 0);  analogWrite(motorB2, 0);
    }
```

```
void loop(){

if (Serial.available() > 0) {
  command = Serial.read();
  stopRobot();      //Initialize with motors stopped.

if (lightFront)  {digitalWrite(light_FR, HIGH);}
if (!lightFront) {digitalWrite(light_FR, LOW);}
if (lightRear)   {digitalWrite(light_RE, HIGH);}
if (!lightRear)  {digitalWrite(light_RE, LOW);}
if (hornBUZZER)  {digitalWrite(horn_BUZZ, HIGH);}
if (!hornBUZZER) {digitalWrite(horn_BUZZ, LOW);}
if (lightExtra)  {digitalWrite(light_EX, HIGH);}
if (!lightExtra) {digitalWrite(light_EX, LOW);}

switch (command) {
case 'F':goAhead();break;
case 'B':goBack();break;
case 'L':goLeft();break;
case 'R':goRight();break;
case 'I':goAheadRight();break;
case 'G':goAheadLeft();break;
case 'J':goBackRight();break;
case 'H':goBackLeft();break;
case 'D':stopRobot();break;
case '0':vSpeed = 0;break;
case '1':vSpeed = 25.5;break;
case '2':vSpeed = 51;break;
case '3':vSpeed = 76.5;break;
case '4':vSpeed = 102;break;
case '5':vSpeed = 127.5;break;
case '6':vSpeed = 153;break;
case '7':vSpeed = 178.5;break;
case '8':vSpeed = 204;break;
case '9':vSpeed = 229.5;break;
case 'q':vSpeed = 255;break;
case 'W':lightFront = true;break;
case 'w':lightFront = false;break;
case 'U':lightRear  = true;break;
case 'u':lightRear  = false;break;
case 'V':hornBUZZER = true;break;
case 'v':hornBUZZER = false;break;
case 'X':lightExtra = true;break;
case 'x':lightExtra = false;break;
}
}
}
```

// Right Hand Code

```
#include <Servo.h>

Servo servo1;
Servo servo2;

const int SERVO1_PIN = 9;
const int SERVO2_PIN = 10;

void setup() {
  Serial.begin(9600); // Initialize serial communication
  servo1.attach(SERVO1_PIN);
  servo2.attach(SERVO2_PIN);
}

void loop() {
  for (int angle = 0; angle <= 360; angle += 5) {
    float radian = angle * (3.14159 / 180.0);
    int servo1_pos = (int)(90 + 60 * sin(radian));
    int servo2_pos = (int)(90 + 60 * cos(radian));

    servo1.write(servo1_pos);
    servo2.write(servo2_pos);

    Serial.print("Servo 1: ");
    Serial.print(servo1_pos);
    Serial.print("  Servo 2: ");
    Serial.println(servo2_pos);

    delay(51);
  }

 }
```

CHAPTER:6

ADVANTAGES

Advantages Of Ogashi

Having a worship robot has many advantages, such as:

1. Personalized Spiritual Guidance: The worship robot provides personalized spiritual guidance based on the user's beliefs, needs, and preferences. It can cater to users of different religions and cultures, offering comprehensive information about religious practices, rituals, and belief systems.

2. Convenience: The worship robot is available 24/7, which makes it very convenient for users to get their queries and concerns addressed at any time. Users can interact with it at their convenience and in the comfort of their homes.

3. Consistency: Since the worship robot provides consistent guidance, users can rely on it to provide accurate and relevant information. It eliminates the inconsistencies that can arise with human spiritual leaders and provides reliable guidance.

4. Increased Access: The worship robot can reach people who may not have access to spiritual leaders or resources due to their location, ethnicity, or socioeconomic status. It provides an opportunity for people to connect with their spirituality and get guidance from a trusted source.

5. Enhanced Learning: The worship robot can provide users with an immersive and interactive learning experience. It can use various modes such as audio, video, and text to provide users with a comprehensive understanding of spiritual practices and beliefs.

Overall, the worship robot offers a unique and innovative way to connect people with their spirituality by providing personalized, consistent, and convenient guidance that is accessible to everyone. It is a powerful tool that helps individuals deepen their spiritual beliefs and foster a sense of community in the ever-evolving digital world.

CHAPTER: 7

APPLICATIONS

Application Of Ogashi

Besides performing worshiping this robot has multiple different uses as this project is basically to build a humanoid robot it has all the related applications a humanoid robot can perform a few of them are:

- Research and space exploration
- Personal assistance and caregiving
- Education and Entertainment
- Search and rescue
- Manufacturing and maintenance etc**.**

Future Enhancement

- As we already know currently this robot is only performing the worshiping task which is assigned to it. In the future, we are planning to make this robot completely AI-based which will allow the robot to perform the task assigned to it immediately and with more accuracy as being an AI tool it will compare all the good outcomes at achieve the best results with that.
- After being a fully AI-based Robot the applications of this robot will also enhance as it can be used in space research, for marine purposes, in special forces, and many more.
- Real-time working capability will ensure that this robot will perform every task with accuracy and precision.

Conclusion

- Humanoid robots have a great future. Currently, humanoids have constraints. They can function according to the algorithm that's added to their programs. Although some humanoid robots learn and work in real-time according to certain situations due to recurrent AI development, they still have limits. It's still a long way to go to bridge the gap. And we can achieve the same by working on this project.
- Humanoids are efficient, but they don't have a conscience or a complex thinking capacity like humans, which is still debatable globally. There are many talks about whether humanoids will take over the world or be a friend to humans. It's difficult to determine, but we can safely conclude that humanoids will change the market world. Whether it's the entertainment industry or the medical industry, the humanoid robots will play a key instrument in businesses' growth graphs without fail.

REFRENCES:

https://www.kjell.com/globalassets/mediaassets/701905_87902_datasheet_en.pdf

https://www.vox.com/future-perfect/2019/9/9/20851753/ai-religion-robot-priest-mindar-buddhism-christianity

https://www.bbc.com/news/av/technology-58983047

https://www.theguardian.com/technology/2017/may/30/robot-priest-blessu-2-germany-reformation-exhibition

https://www.usatoday.com/story/news/world/2017/10/11/could-robots-replace-pastors-one-just-gives-blessings/754999001/

https://www.dailystar.co.uk/news/world-news/worlds-religions-embracing-ai-god-22406074

https://medium.com/in-our-times/meet-your-maker-the-robot-priests-taking-the-world-by-storm-32b2e398383