KHELHOSHE POLYTECHNIC ATOIZU

ZUNHEBOTO : NAGALAND

PROJECT REPORT

ON

“YIM MULTI-TASKING VOICE CONTROLLED ROBOT”

Submitted to

STATE COUNCIL FOR TECHNICAL EDUCATION
KOHIMA: NAGALAND

Project Report submitted for the fulfilment of

Diploma in Electrical and electronic Engineering ***Batch of 2022-2025***

Under the guidance of
Er. Yibenthung Odyuo, Lecturer (Dept. Of Electrical and electronic Engineering)
Er. B. Lichamo Murry (HOD), Lecturer (Dept. Of Electrical and electronic)
KHELHOSHE POLYTECHNIC ATOIZU

ZUNHEBOTO: NAGALAND

KHELHOSHE POLYTECHNIC ATOIZU

ZUNHEBOTO : NAGALAND

PROJECT REPORT

ON

"YIM MULTI-TASKING VOICE CONTROLLED ROBOT"

Submitted to

STATE COUNCIL FOR TECHNICAL EDUCATION
KOHIMA: NAGALAND

Project Report submitted for the fulfilment of

Diploma in Electrical and electronic Engineering *Batch of 2022-2025*

Under the guidance of
Er. Yibenthung Odyuo, Lecturer (Dept. Of Electrical and electronic Engineering)
Er. B. Lichamo Murry (HOD), Lecturer (Dept. Of Electrical and electronic)
KHELHOSHE POLYTECHNIC ATOIZU
ZUNHEBOTO: NAGALAND

KHELHOSHE POLYTECHNIC ATOIZU

Certificate

This is to certify that the project entitled

“YIM”

INNOVATIVE-MULTI-TASKING VOICE CONTROLLED ROBOT

Has been successfully completed and submitted by Group 2,
Name:

1. **MR. NOLIBA B ANAR**
2. **MR. IMLILUYIM**
3. **MR. KUKHAINTHONG T Y**
4. **MR. THSIDISHANG L SANGTAM**

In partial fulfilment of the requirements for the award of the Diploma in Electrical and electronics Engineering under the State council for technical Education (SCTE), Nagaland.

We hereby approved the project as a significant contribution to the student’s academic development.

Head of Institution
Er. Neisekho Chaya
Principal,Khelhoshe Polytechnic Atoizu

Guided by :
Er. Yibenthung Odyuo
Asst. Lecturer
Khelhoshe Polytechnic Atoizu

Er. B Lichamo Murry
HOD,Dept. Of EEE
Khelhoshe Polytechnic Atoizu

PROJECT DEVELOPERS

SL. NO	NAME	ROLL NO	SIGNATURE
1	MR. NOLIBA B ANAR	22010201	
2	MR. IMLILUYIM	22010203	
3	MR. KUKHAINTHONG T Y	22010204	
4	MR. THSIDISHANG L SANGTAM	22010209	

Acknowledgement

We would like to express our sincere gratitude to Er. Neisekho Chaya, Principal of Khelhoshe Polytechnic Atoizu, for his constant support and encouragement throughout the course of our project. His leadership and vision have always been a great source of motivation.

Our heartfelt thanks to Er. Yibenthung Odyou, our project guide, for his valuable guidance, continuous supervision, and constructive suggestions. We also thank Mr. B. Lichamo Murry, Head of Department, Electrical and Electronics Engineering, for his assistance and helpful advice.

We are deeply thankful to all the lecturers, faculty members, and staff of the Electrical and Electronics Engineering department, and all others who have supported us, directly or indirectly, in completing this project successfully.

Last but not least, we extend our appreciation to our friends and colleagues, especially our team members, for their cooperation, teamwork, and shared efforts in bringing this project to life.

Abstract

The YIM Multi-Tasking Voice-Controlled Robot project presents an innovative, voice-operated robotic system designed to serve as a smart assistant and service bot for diverse applications, including classroom navigation, home automation, and safety monitoring. This project integrates advanced voice recognition, sensor-based navigation, and IoT-enabled automation to execute multiple tasks seamlessly via offline voice commands.

At its core, the robot employs the VC02 AI Thinker module for offline voice recognition, converting user commands into digital signals processed by an ESP32 microcontroller (supported by an Arduino UNO for peripheral control). Key functionalities include human-following via ultrasonic sensors, 3D motion robotic arm operations for object handling, and magnetic tape/RFID-based classroom navigation, where the robot follows predefined magnetic tape paths and identifies junctions using RFID tags. The system also integrates Wi-Fi, Bluetooth-enabled home automation, allowing remote control of appliances through relay modules.

Safety is prioritized through real-time temperature monitoring (DHT11 sensor), while obstacle avoidance is achieved via ultrasonic sensors and adaptive rerouting algorithms. Testing demonstrated 85-90% accuracy in voice command recognition under low-to-moderate noise, with reliable task execution latency under 3 seconds. Future scalability includes Alexa Echo 4th Gen integration for expanded voice-controlled features, IoT based automatic application control and monitoring.

This project bridges robotics, IoT, and human-machine interaction, offering a versatile solution for educational, domestic, and industrial environments. Its modular design ensures adaptability, emphasizing practicality, safety, and user-friendly operation.

Keywords: Voice-controlled robot, ESP32 microcontroller, RFID navigation, home automation, IoT, robotic arm, offline voice recognition.

LIST OF FIGURES AND TABLES

LIST OF ABBREVIATIONS

Abbreviation	*Full Form*
ACK/NACK	*Acknowledgment / Negative Acknowledgment*
AI	*Artificial Intelligence*
AIoT	*Artificial Intelligence of Things*
AMR	*Autonomous Mobile Robot*
AWS	*Amazon Web Services*
CAD	*Computer-Aided Design*
CNN	*Convolutional Neural Network*
DC	*Direct Current*
DHT11	*Digital Humidity and Temperature Sensor (Model)*
DOF	*Degrees of Freedom*
ESP32	*Espressif Systems' Wi-Fi/Bluetooth Microcontroller (Model)*
GPIO	*General-Purpose Input/Output*
HC-SR04	*Ultrasonic Distance Sensor (Model)*
I2C	*Inter-Integrated Circuit (Communication Protocol)*
IoT	*Internet of Things*
IR	*Infrared*
KY-003	*Hall-Effect Sensor (Model)*
LiDAR	*Light Detection and Ranging*
LiPo	*Lithium Polymer (Battery)*
LM2596	*Buck Converter IC (Voltage Regulator)*

MFRC522	*RFID Reader Module (Model)*
NLP	*Natural Language Processing*
OLED	*Organic Light-Emitting Diode*
PCB	*Printed Circuit Board*
PCA9685	*12-bit PWM Servo Driver (Model)*
PID	*Proportional-Integral-Derivative (Control Algorithm)*
PWM	*Pulse Width Modulation*
RF	*Radio Frequency*
RFID	*Radio-Frequency Identification*
SG90	*Micro Servo Motor (Model)*
UART	*Universal Asynchronous Receiver-Transmitter (Serial Communication Protocol)*
VC02	*AI Thinker Offline Voice Recognition Module (Model)*
VLMaps	*Visual Language Maps (Navigation Framework)*
VS Code	*Visual Studio Code (Software)*
Wi-Fi	*Wireless Fidelity*

Document Structure

1.Introduction

The integration of robotics into daily life has transformed industries, homes, and educational institutions, yet existing systems often struggle to balance affordability, multitasking, and offline functionality. This project, the YIM Multi-Tasking Voice-Controlled Robot, bridges this gap by delivering a versatile, cost-effective solution designed to operate as a smart assistant in homes, classrooms, and laboratories. The name *YIM*, derived from the AO language, embodies the essence of a learner in pursuit of education and knowledge—a concept central to this project's philosophy. Designed by students of electrical and electronic engineering, YIM symbolizes the journey of transforming theoretical knowledge into practical innovation. By prioritizing offline voice recognition, modular design, and hands-on learning, the robot serves as a testament to the practical application of academic concepts while addressing real-world automation challenges.

The YIM robot was conceived with the dual aim of demonstrating practical engineering skills and addressing real-world automation needs. Central to its design is the implementation of offline voice recognition using the VC02 AI Thinker module, enabling command execution without internet dependency. The project further seeks to integrate a multi-sensor navigation system combining ultrasonic, magnetic, and RFID technologies for structured indoor environments. Additional objectives include deploying a 3D robotic arm for object handling, enabling IoT-based home automation via wireless relay control, and incorporating safety monitoring through temperature. By aligning these goals with the educational symbolism of its name—*YIM*, derived from the AO language to signify a learner in pursuit of knowledge—the project underscores the value of bridging classroom theory with hands-on innovation.

Beyond its technical contributions, the YIM project embodies cultural and pedagogical significance. The name *YIM*, rooted in the AO language's term for a knowledge-seeking individual, reflects the team's commitment to experiential learning. By translating classroom theories such as microcontroller programming, sensor interfacing, and path-planning algorithms into a functional robotic platform, the project exemplifies how academic concepts can solve real-world challenges. This aligns with its broader mission to inspire students and educators to view engineering as a bridge between theoretical study and tangible innovation. YIM's modular architecture and cost-effective design further highlight its potential as a scalable prototype for educational labs, smart homes, and small-scale industries

Key Features

The AI Voice Controlled Multitasking Robot is designed to perform a wide range of functions through intelligent voice interaction and sensor-based automation. Its key features include:

1. Voice Command Control

The robot receives and executes voice commands using an offline VC02 AI voice recognition module, which sends unique binary outputs via 5 digital pins. This allows for hands-free, offline operation without the need for internet or cloud processing.

2. Wake-Up Activation

The robot can be set to respond only when specific pre-trained voice commands are detected via the VC02 module's binary outputs, allowing for selective activation and avoiding unintended behaviour.

3. Inquiry Response

The robot is capable of answering predefined questions or commands, offering basic human-like interaction.

4. Human-Following Capability

Using ultrasonic sensors, the robot can follow a person, adjusting its path based on detected motion and distance.

5. 3D Motion Robotic Arm

A foldable robotic arm allows the robot to pick up and carry small objects on command, enhancing its service capabilities.

6. Classroom Navigation

The robot follows magnetic tape paths and identifies locations using RFID tags, enabling it to navigate to specific classrooms or zones.

7. Obstacle Avoidance

Equipped with ultrasonic sensors, the robot detects obstacles in its path and automatically reroutes or alerts with a buzzer.

8. Home Automation

It can control appliances wirelessly using Wi-Fi, Bluetooth, or RF modules and relay switches, acting as a mobile smart assistant.

9. Safety Monitoring

Sensors like DHT11 and fire detectors monitor temperature, alerting the user when unsafe conditions are detected.

2. Literature Review

3.1 Existing Voice-Controlled Systems

Voice-controlled systems have evolved significantly, with applications spanning healthcare, automotive, and smart homes. For instance, **Dragon Medical One** streamlines medical documentation by converting doctors' speech into electronic health records, reducing administrative burdens . Similarly, automotive systems like **Apple CarPlay** enable hands-free navigation and infotainment control, enhancing driver safety . However, most systems rely on cloud-based processing (e.g., Alexa, Google Assistant), limiting functionality in offline environments and raising privacy concerns. Recent advancements in offline modules, such as the **VC02 AI Thinker**, demonstrate potential for decentralized voice recognition, though accuracy remains challenged in noisy settings .

3.2 Robotic Navigation Techniques

Modern robotic navigation combines **absolute positioning** (e.g., GNSS for agricultural robots) and **relative positioning** (e.g., ultrasonic sensors for obstacle avoidance). For example, agricultural AMRs use GPS to navigate vast fields but switch to LiDAR for precise crop interactions. Innovations like **VLMaps** (Visual Language Maps) integrate natural language commands with 3D environmental data, enabling robots to interpret instructions like "go left of the sofa" . Despite progress, many systems depend on costly sensors (e.g., LiDAR) or pre-mapped routes, limiting adaptability in dynamic environments.

2.3 IoT and Home Automation Trends

The IoT landscape is dominated by **AI-driven automation** and interoperability standards like **Matter**, which unify smart home ecosystems (e.g., Philips Hue lights with Amazon Echo) . Energy efficiency is a key focus, with devices like **Google Nest Thermostat** optimizing heating based on user behavior. Voice-controlled home automation is surging, with 478 million households projected to adopt smart devices by 2025. However, fragmented ecosystems and security vulnerabilities persist, highlighting the need for integrated, low-cost solutions.

2.4 Gaps Addressed by This Project

Existing systems face three critical gaps:

1. **Offline Dependency**: Most voice assistants require internet connectivity, limiting rural or low-resource settings.
2. **Single-Task Focus**: Robots often specialize in navigation *or* automation, lacking multitasking agility.
3. **Cost Barriers**: High-end sensors (LiDAR, radar) and cloud infrastructure raise deployment costs .

The **YIM Robot** addresses these by:

- Using **offline voice recognition** (VC02 module) for decentralized command processing.
- Merging **magnetic tape navigation** with RFID tags for affordable indoor pathfinding, akin to agricultural AMRs.
- Integrating **IoT-based home automation** (relay-controlled appliances) and **safety monitoring** (DHT11, fire sensors) into a unified platform.

By bridging affordability, multitasking, and offline functionality, YIM offers a scalable blueprint for accessible robotics in education and small-scale automation.

3.Methodology

The YIM robot's development followed a structured hardware-software co-design approach, prioritizing modularity, cost-effectiveness, and seamless integration of voice control with multitasking functionality.

3.1 Hardware Design

3.1.1 System Block Diagram

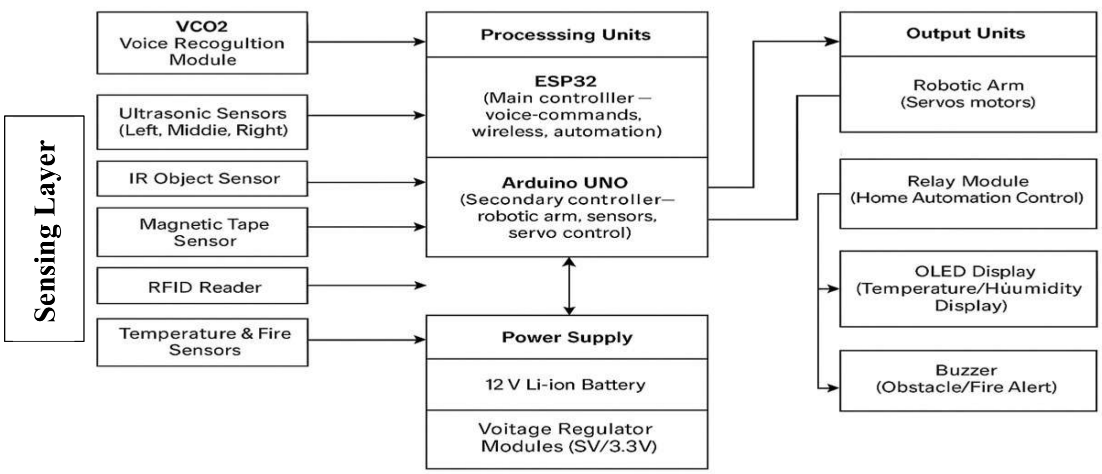


Figure 3.0 : system block diagram

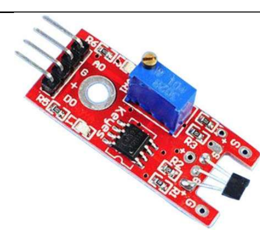
The hardware architecture comprises five interconnected subsystems:

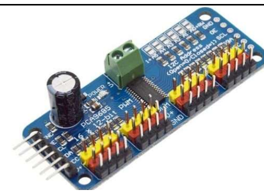
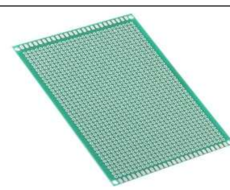
- **Voice Control Unit**: VC02 AI Thinker module paired with a microphone and speaker for offline command processing.
- **Central Processing Unit**: ESP32 (primary controller) and Arduino UNO (auxiliary controller) to manage sensors, motors, and relays.
- **Actuation System(output units)**: Four gear motors with L298N motor drivers for mobility, servo motors for the robotic arm, and relay modules for appliance control.

- **Sensing Layer**: Ultrasonic sensors (HC-SR04) for human-following, RFID reader (MFRC522) for navigation, DHT11 for temperature monitoring, and IR sensors for object detection.
- **Power supply**: 12v Li-ion battery, voltage regulators i/p 12v and o/p 12v, 5v and 3.3v.

3.1.2 Component Selection

Components were chosen to balance cost, functionality, and ease of integration:

Sl. No.	Components	image	Description
1.	Esp32		Dual-core microcontroller with Wi-Fi/Bluetooth, used as the main controller for voice processing, sensor integration, and task coordination.
2.	Arduino uno		Secondary microcontroller for handling servo motors, relays, and auxiliary sensors, complementing ESP32's GPIO limitations.
3.	Vc-02 AI thinker offline voice recognition module		Offline voice recognition module converts spoken commands to digital signals via built-in algorithms.
4.	Relay module		Electromechanical switch to control high-voltage appliances (e.g., lights, fans) using low-voltage signals.
5.	Hall sensor		Detects magnetic fields for tracking pre-installed magnetic tape paths during navigation.
6.	DC-DC 12V to 3.3V 5V 12V Power Module		Regulates 12V battery output to 3.3V/5V for microcontrollers and sensors.

7.	Ultrasonic sensor		Measures distance for obstacle detection and human-following.
8.	L298N motor driver		Drives 12V DC gear motors for robot movement (forward/backward, turning).
9.	12-bit PWM Servo Motor Driver I2C Module(**PCA9685**)		I2C-based 12-bit servo controller for precise robotic arm movements.
10.	12v Lipo battery		Primary power source for motors and electronics.
11.	12v dc gear motor		Provide locomotion with speed and torque control.
12.	Universal PCB		aseboard for assembling and interconnecting components.
13.	3D arm		Foldable arm with servo motors for object grasping and transport.
14.	Servo motor		SG90/TowerPro models for joint movements in the robotic arm.

15.			
16.	2A rocker switch		Manual power control for the entire system.
17.	Buzzer module		Audible alerts for obstacles, or command confirmation.
18.	DHT11 temperature and humidity sensor		Monitors ambient temperature/humidity, displayed on the OLED screen.
19	128X64 OLED I2C LED Display		Shows real-time data (e.g., temperature, active commands).
20.	Jumper wires		Interconnects components on the PCB.
21.	RFID reader and tags		Detects pre-programmed tags at classroom junctions for navigation.
22.	Magnetic tape		

Table no. 3.1 : components used

3.1.3 Circuit Design

A unified circuit was designed to ensure stable power distribution and signal integrity:

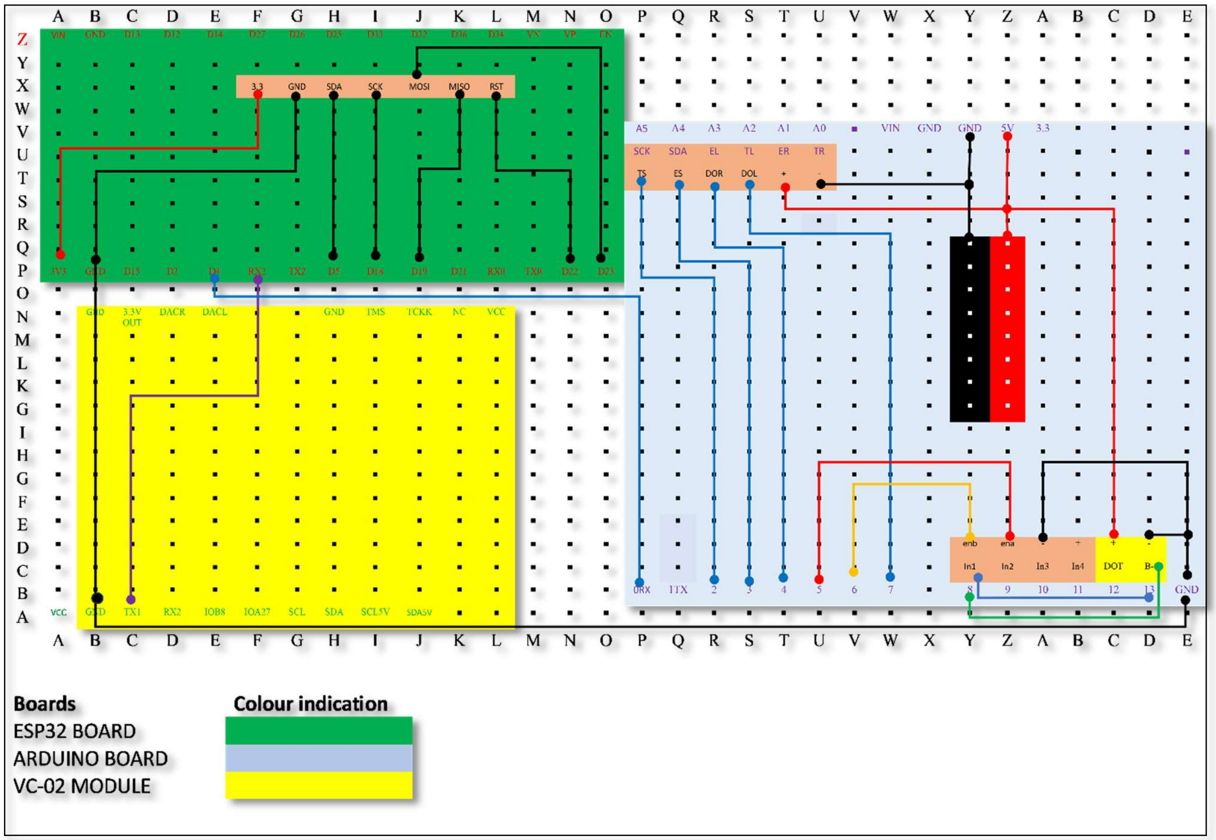


Fig. 3.1 : Circuit Design

3.2 Software Design

3.2.1 Voice Recognition Workflow

The VC02 module captures voice commands, converts them to hexadecimal signals via built-in algorithms, and transmits data to the ESP32 via UART. The ESP32 maps these signals to predefined tasks (e.g., "Go to Electrical 4 Class" triggers RFID navigation) using a lookup table stored in its flash memory. The designing Firmware of VC-02 are given in the **table no. 3.2**:

Sl. No.	functions	Hex Decimal (Parameter)	BEHAVIOR	COMMON WORD	REPLY
1	Turning off the mode	AA 00	TurningOffMode	turn off the mode|off the mode|mode stop	all modes turn off
2	Human following mode	AA 01	HumanFollowing	please follow me|follow me|turn follow mode on|follow mode	follow mode on
3	Dance	AA02	Dance	dance for me|show me some moves|dance	okay i will dance for you
4	360 degree	AA 03	ThreesixtyTurn	take three sixty turn|rotate|three sixty degree	okay,i will do three sixty turning for you
5	Wink	AA 04	Wink	wink|robot wink	here you go
6	Turn on Alexa mode	AA 05	Alexa	turn alexa mode on|switch to alexa mode	alexa mode on
7	Go back	AA 06	goBack	Thankyou you can go back now|return to your station	your most welcome, i am returning
8	Go to electrical room 5	AA 07	GotoEEE5	go to electrical class room five|please go to electrical room five	okay i am going to electrical class room five
9	Go to electrical room 3	AA 08	GotoEEE3	go to electrical class room three|please go to electrical class room three	okay i am going to electrical class room three
10	Go to civil room 5	AA 09	GotoC5	go to civil class room five|please go to civil room five	okay i am going to civi class room five
11	Go to civil room 3	AA 10	GotoMe5	go to mechanical class room five|please go to mechanical room five	okay i am going to civil class room three
12	Go to lecturer's common room	AA 11	GotoLCR	go to common room|go to lecturer's common room	okay i am going to lecturer's common room
13	Go to charging station	AA 12	Charge	go to charging station|robot charge	okay i go to charging station

14	Go to electrical room 5 and give	AA 13	goNgiveE5	take this and give to electrical class room five|deliver this to electrical class room five	okay give me i will deliver it to electrical class room five
15	Go to electrical room 3 and give	AA 14	goNgiveE3	take this and give to electrical class room three|deliver this to electrical class room three	okay give me i will deliver it to electrical class room three
16	Go to civil room 5 and give	AA 15	goNgiveC5	take this and give to civil class room five|deliver this to civil class room five	okay give me i will deliver it to civil class room five
17	Go to civil room 3 and give	AA 16	goNgiveC3	take this and give to civil class room three|deliver this to civil class room three	okay give me i will deliver it to civil class room three
18	Go to lecturer's common room and give	AA 17	GoLCRnGive	take this and give it to the lecturers common room|deliver this to teachers common room|deliver to common room	okay give me i will deliver it to lecturer's common room
19	Hand unfold	AA 18	ArmUF	unfold your arm|open your arm	okay
20	Hand fold	AA 19	ArmF	fold your arm|close your arm	okay
21	open grip	AA 20	OGrip	open grip|open your hand	okay
22	close grip	AA 21	CGrip	grip it|hold this|close your hand	okay
23	Temperature enquiry	AA 22	Temp.	can you please display the temperature|robot what is the room temperature	temperature is displayed on the oled screen
24	Turn on the light 1	AA 23	Light1ON	switch on the light one|turn on the light one	okay light one is turning on|light one is on
25	Turn on the light 2	AA 24	Light2ON	switch on the light two|turn on the light two	okay light two is turning on|light two is on
26	Turn on the smart TV	AA 25	TVON	turn on the smart t v|switch on the t v|turn on the t v	smart tv is turned on

27	Turn on the fan	AA 26	FanON	turn on the fan|switch on the fan	okay i will switch on the fan
28	Turn off the light 1	AA 27	Light1OFF	turn off the light one|switch off the light one	light one is turn off
29	Turn off the light 2	AA 28	Light2OFF	turn off the light two|switch off the light two	light two is turn off
30	Turn off the smart TV	AA 29	TVOFF	turn off the smart t v|switch off the t v|turn off the t v	okay smart class room tv is turning off for you
31	Turn off the fan	AA 30	FanOFF	turn off the fan|switch off the fan	okay i will switch off the fan
32	Move forward	AA 31	MoveF	move forward|move front	okay
33	Move backward	AA 32	MoveB	move backward|move back	okay
34	Turn right	AA 33	TurnR	turn right	okay
35	Turn left	AA 34	TurnL	turn left	okay
36	Stop	AA 35	StopM	brake|stop	okay
37	Forward, Forward	AA 36	MoveFF	front front	okay
38	Backward, Backward	AA 37	MoveBB	back back	okay
39	Right, right	AA 38	TurnRR	right right	okay
40	Left, left	AA 39	TurnLL	left left	okay
41	Creator		Creator	who created you|who made you|who invented you|who design you	Four Electrical students from 2022 batch: Imliluyim, Thsidishang, Noliba, and Kokhainthong.
42	Master		Master	who is your master	Imliluyim, Thsidishang, Noliba, and Kokhainthong.

43	Collage info 0		College	what is the name of your college|name of your institution|you are from which college	Khelhoshe Polytechnic Atoizu, Zunheboto, Nagaland.
44	Collage info 1		InfoCol1	tell me about your college|can you tell me about k p a|say something about your college|give a short introduction about khelhoshe polytechnic atoizu	Khelhoshe Polytechnic Atoizu (KPA), set up in nineteen seventy two, is one of the oldest technical institutes in Nagaland. It is in Atoizu, Zunheboto, and offers diplomas in Electrical, Civil, Mechanical, and Automobile Engineering. The college is linked with the State Council for Technical Education and aims to build skills, support innovation, and get students and prepare them for industry and entrepreneurship.
45	Collage info2		InfoCol2	what do you know about your institution|briefly describe k p a|introduce your college in a few lines|what is k p a known for|tell me	Khelhoshe Polytechnic Atoizu (KPA), set up in nineteen seventy two, is one of the oldest technical

				something about your polytechnic	institutes in Nagaland. It is in Atoizu, Zunheboto, and offers diplomas in Electrical, Civil, Mechanical, and Automobile Engineering. The college is linked with the State Council for Technical Education and aims to build skills, support innovation, and get students and prepare them for industry and entrepreneurship.
46	Principal name		Principal	who is the principal of college|college principal	Engineer Neisekho Chaya is the Principal of KPA.
47	Department lecturers		EEElec	electrical department faculty|can you please tell me the names of the electrical lectures |electronic teacher|electrical teachers name	Engineer B. Lichamo Murry (HOD), engineer Poushong K. Lam, engineer Yibenthung Odyuo, and engineer Meyachuba.
48	Department lecturers		CElec	tell me the civil lectures name|can you tell me the names of the civil lecturers|civil teacher name	engineer Nzani A. Murry (HOD), engineer A. Rentsamo Tsope, and engineer Khruvelu.
49	Department lecturers		MElec	tell me the mechanical lectures	engineer Pavito V. Yepthomi

				name|mechanical teachers name|who are the lecturers in the mechanical department|mechanical faculty	(HOD), engineer Manu Chiero, and sir Takusunep Walling.
50	Department lecturers		SnH	tell me science and humanities lectures name|can you tell me the names of the general lecturers|who are the faculties in the science and humanities department|general teacher	Miss Jandeno Jami (In-Charge), Miss Melevolu Thisa, sir Lhutu Keyho, engineer Mankup K., Miss Mennyei B. Phom, sir Sungjemliba Longkumer, and sir Kulo Kapfo.
51	Department lecturers		AElec	tell me the automobile teachers name|can you tell me the names of automobile faculties|automible teachers|could you please share the list of automobile faculty	engineer Ruokuobeilie Mere (HOD) and engineer Jenibemo T. Murry.
52	ME (HOD)		HodM	Who is the head of mechanical department|mechanical head of department	Engineer Ruokuobeilie Mere
53	Science and humanities (HOD)		HodSH	Who is the in charge of science and humanities department|general in charge|head of science department	Miss Jandeno Jami
54	Bluetooth off	AA 54	TurnOffB	Turn off bluetooth mode	Ok bluetooth mode off
55	Bluetooth on	AA 53	TurnOnB	Turn on bluetooth mode	Ok bluetooth mode on

56	Routine		RoutineM	tell me the monday routine|tell me the routine for monday	First: Entrepreneurship, Second: Elective two, Third: Testing & Maintenance, Fourth: Control System.
57	Routine		RoutineTU	tell me the tuesday routine|tell me the routine for tuesday	First: Switchgear & Protection, Second: Control System, Third: Elective two, Fourth: Entrepreneurship.
58	Routine		RoutineW	tell me the wednesday routine|tell me the routine for wednesday	First: Switchgear and Protection, Second: Entrepreneurship, Eleven fifteen a.m. to one fifteen p.m.: Professional Practice.
59	Routine		RoutineTH	tell me the thursday routine|tell me the routine for thursday	First: Elective Two, Second: Switchgear & Protection, Third: Testing & Maintenance, Fourth: Off.
60	Routine		RoutineF	tell me the friday routine|tell me the routine for friday	First: Testing & Maintenance, Second: Control System, Eleven fifteen a.m. to one fifteen p.m.: Project Work.
61	Objectives		Objectives	hey robot what is your main objective|can you tell me what you were designed to do|what is your objectives	I’m an AI voice-controlled multitasking robot built to assist in homes and schools. I navigate, carry items, follow commands, control devices,

					and aim to inspire interest in AI and robotics.
62	Features		Features	robot what features do you have|can you list your main functions	I use offline voice commands, follow humans using ultrasonic sensors, and carry objects with a three D robotic arm. I navigate using magnetic paths and RFID, avoid obstacles, and control appliances wirelessly. Future upgrades include Alexa integration.
63	Applications		Appli	where can you be used in real life|how are you helpful at home or college|what are the application of this robot	I help in colleges by carrying items and navigating rooms, and at home by controlling devices, enhancing safety, and responding to voice commands. I can assist in public events, too.
64	AE (HOD)		HodA	Who is the Head of Automobile department|head of the department automobile	Engineer Ruokuobeilie Mere
65	Handshake	AA 40	hand shake|handskake|give me your hand	okay, hold my hand	okay
66	Extra	AA 41	Extra	yim you know what to do|start the task	okay
67	Arm movements	AA 42	ElbowUP	elbow up	okay
68	Arm movements	AA 43	ElbowDown	elbow down	okay

69	Arm movements	AA 44	ShouFront	shoulder front	okay
70	Arm movements	AA 45	ShouBack	shoulder back	okay
71	Arm movements	AA 46	WristR	wrist right|clock	okay
72	Arm movements	AA 47	WristL	wrist left|anti clock	okay
73	Tape guide mode	AA 48	MagTapFoll	Turn on the magnetic tape following mode|tape following mode on|tape following mode|magnetic tape following mode on	Okay magnetic tape following mode is on
74	Arm movements	AA 49	WrestROR	Wrist rotate right|right rotate	okay
75	Arm movements	AA 50	WrestROL	Wrest rotate left|rotate left	okay
76	Arm movements	AA 51	BaseROR	Base rotate right|shoulder right	okay
77	Arm movements	AA 52	BaseROL	Base rotate left|shoulder left	okay
78	Workshop (HOD)		workIC	Who is workshop in charge|workshop Head of department|work shop head of department|practical head of department	Engineer Narojungla
79	EEE (HOD)		hodE	Who is the Head of electrical department|electronics head of department|electrical engineering head of department	Engineer Lichamo Murry
80	C.E (HOD)		hodC	Who is the Head of civil department|civil head of department|civil engineering head of department	Engineer Nzani A. Murry

Table no. 3.2

3.2.2 Code Architecture

Firmware was developed in Arduino IDE using C/C++:

- **ESP32 Code**: Handles voice command processing, Wi-Fi/Bluetooth communication, and sensor data fusion.

  Code :

```
#include <WiFi.h>
#include <esp_now.h>
#include <SPI.h>
#include <MFRC522.h>
// RFID Configuration (CHANGE THESE TO YOUR PINS!)
#define RST_PIN 22     // ESP32 RST pin (e.g., GPIO 22)
#define SS_PIN 21      // ESP32 SDA pin (e.g., GPIO 21)
MFRC522 rfid(SS_PIN, RST_PIN);
// Receiver MAC Address (Update this!)
uint8_t receiverMAC[] = {0xF4, 0x65, 0x0B, 0xE7, 0xD8, 0xFC};
// Data Structures
typedef struct struct_message {
  uint8_t bytes[2]; // VC-02 command bytes
} struct_message;
struct_message outgoingData;
// Serial Pins
#define ARDUINO_TX 4  // ESP32 TX pin for Arduino Uno
void setup() {
  // Initialize serial ports
  Serial.begin(115200);
  Serial.begin(9600);
  Serial2.begin(9600, SERIAL_8N1, 16, 17); // VC-02
    Serial1.begin(9600, SERIAL_8N1, -1, ARDUINO_TX); // Arduino Uno
  // Initialize RFID
  SPI.begin();
  rfid.PCD_Init();
  // Initialize WiFi/ESP-NOW
  WiFi.mode(WIFI_STA);
  if (esp_now_init() != ESP_OK) {
    Serial.println("❌ ESP-NOW init failed");
    return;
  }
  esp_now_peer_info_t peerInfo;
  memcpy(peerInfo.peer_addr, receiverMAC, 6);
  peerInfo.channel = 0;
  peerInfo.encrypt = false;
  if (esp_now_add_peer(&peerInfo) != ESP_OK) {
    Serial.println("❌ Failed to add peer");
    return;
  }
  Serial.println("🚀 System Ready: VC-02 + RFID");
}
void loop() {
  // Handle VC-02 Commands
  if (Serial2.available() >= 2) {
    uint8_t byte1 = Serial2.read();
    uint8_t byte2 = Serial2.read();
    // Send via ESP-NOW
    outgoingData.bytes[0] = byte1;
    outgoingData.bytes[1] = byte2;
        esp_now_send(receiverMAC, (uint8_t *)&outgoingData, sizeof(outgoingData));
    // Send to Arduino with 'V' prefix
    Serial1.write('V');
    Serial1.write(byte1);
    Serial1.write(byte2);
    Serial.printf("VC-02 Sent: %02X %02X\n", byte1, byte2);
  }
  // Handle RFID Tags
    if (rfid.PICC_IsNewCardPresent() && rfid.PICC_ReadCardSerial()) {
    String uid = "";
    for (byte i = 0; i < rfid.uid.size; i++) {
      uid += String(rfid.uid.uidByte[i], HEX);
    }
    // Send to Arduino with 'R' prefix
    Serial1.print("R" + uid);
    Serial.print("RFID Detected: ");
    Serial.println(uid);
    rfid.PICC_HaltA();
    rfid.PCD_StopCrypto1();}}
```

- **Arduino UNO Code**: Manages robotic arm actuation and relay toggling, synchronized with the ESP32 via serial communication (UART).

(Scan the QR code in appendix for Arduino code)

3.3 System Integration and Implementation

3.3.1 Human-Following Mechanism

The *Human Following Mode* is an integral part of our full-scale "Voice-Control Multifunctional Robot" project. In this mode, the robot is programmed to follow a human at a safe distance using ultrasonic sensors. It dynamically adjusts its position—moving forward, backward, or turning left/right—based on the human's location, ensuring smooth and safe tracking. This intelligent behavior enables real-world applications like personal assistant robots, elderly support bots, and mobile robotic platforms in smart environments.

System Overview

The Human Following system uses **three ultrasonic sensors (HCSR04)**:

- **Front Ultrasonic Sensor**: Measures the distance between the robot and the human in front.
- **Left Ultrasonic Sensor**: Detects the human's presence to the left of the robot.
- **Right Ultrasonic Sensor**: Detects the human's presence to the right of the robot.

These sensors feed real-time data to the microcontroller (Arduino), which interprets the human's position and sends motor control signals accordingly.

Working Principle

When the system receives the command AA 01 via serial communication, it activates the Human Following mode. The robot performs the following actions based on sensor readings:

Turning Logic

- **Turn Right**:
  If Right Sensor detects distance < 35 cm
  AND Left Sensor detects distance > 35 cm
- **Turn Left**:
  If Left Sensor detects distance < 35 cm
  AND Right Sensor detects distance > 35 cm

Forward/Backward Logic

- **Move Forward**:
  If front distance > 35 cm and < 80 cm
  → Robot moves forward to reduce the gap
- **Move Backward**:
  If front distance < 30 cm and > 10 cm
  → Robot moves backward to increase the gap
- **Brake/Stop**:
  If front distance is between **30-35 cm**
  → Robot stops and applies brake for **0.5 seconds**

Noise Filtering

To reduce sensor noise and prevent erratic movements, a 10ms delay is introduced between each ultrasonic read cycle. This ensures more stable readings and smoother robot behaviour.

Sensor Placement

For effective human tracking:

- **Front Sensor**: Mounted at the center front of the robot, facing directly forward.
- **Left Sensor**: Mounted on the left side at approximately 45° outward.
- **Right Sensor**: Mounted on the right side at approximately 45° outward.

This positioning allows for a wide angle of detection and provides optimal human tracking coverage as shown in the diagram below.

System Diagram

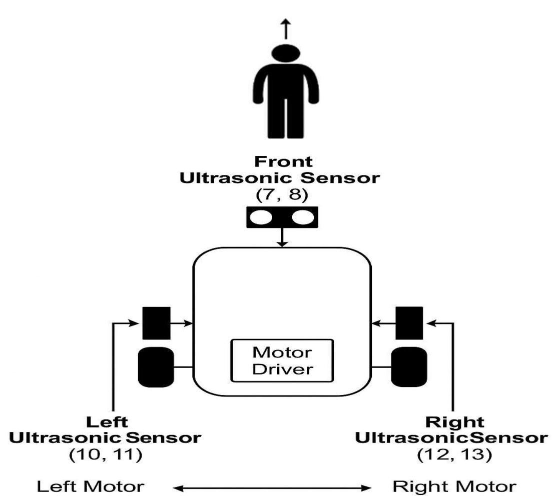


Figure 3.2 : System Diagram of Human following Mode

3.3.2 Robotic Arm Actuation (Servo Motor Control)

Arm Movement Mode, one of the primary modules in the larger system of the " Yim-Multifunctional Voice Control Robot." The goal of this module is to enable precise mechanical arm movement using servo motors controlled through serial commands. The system simulates basic human arm functions like rotation, grip, and joint articulation.

Components Used

Component	Quantity	Description
Arduino Uno	1	Main microcontroller board
PCA9685 16-Channel 12-bit PWM Servo Driver	1	I2C-based servo driver
Servo Motors	6	For base rotation, shoulder angle, elbow, wrist angle, wrist rotation, and grip
Ultrasonic Sensor (named as Scan Sensor)	*1*	*Optional for obstacle scanning (not coded in this mode)*
Serial Communication	-	*Used via RX pin to receive commands*

Table no. 3.3 component used in 3D arm

Servo Configuration Table:

Servo Function	Servo Name (In Code)	PWM Channel
Shoulder Rotation	Base Servo	0
Shoulder Joint Angle	Shoulder Servo	1
Elbow Joint	Elbow Servo	2
Wrist Joint Angle	Wrist Servo	3
Wrist Rotation	Wrist Rot Servo	4
Grip Control	Grip Servo	5

Table no. 3.4 servo configuration table

Working Principle

The arm mode operates based on **serial commands received through RX (D0) pin**. The command is expected in the format AA XX, where:

- AA is the start byte.
- XX is the function code.

Upon receiving a command, the system maps it to one of 14 specific functions. These functions control the movement of servos in a predefined or incremental manner.

Default-Position:

On system start or reset, all servo motors move one after another to the default resting position:

- Base: 90°
- Shoulder: 145°
- Elbow: 178°
- Wrist: 90°
- Wrist Rotation: 0°
- Grip: 20° (Closed)

Functional Commands Table

Command (Hex)	Function Name	Action
AA 18	unfold()	Extend elbow, shoulder, wrist
AA 19	fold()	Fold back to default
AA 20	open_grip()	Opens grip
AA 21	close_grip()	Closes grip
AA 42	elbow_up()	Increases elbow angle by 10°
AA 43	elbow_down()	Decreases elbow angle by 10°
AA 44	shoulder_front()	Increases shoulder angle by 10°
AA 45	shoulder_back()	Decreases shoulder angle by 10°
AA 46	wrest_right()	Increases wrist angle by 10°
AA 47	wrest_left()	Decreases wrist angle by 10°
AA 48	wrest_rotation_right()	Rotates wrist right by 10°
AA 50	wrest_rotation_left()	Rotates wrist left by 10°
AA 51	base_rotation_right()	Rotates base right by 10°
AA 52	base_rotation_left()	Rotates base left by 10°

Table no. 3.5 function comands for 3D arm

Circuit Diagram

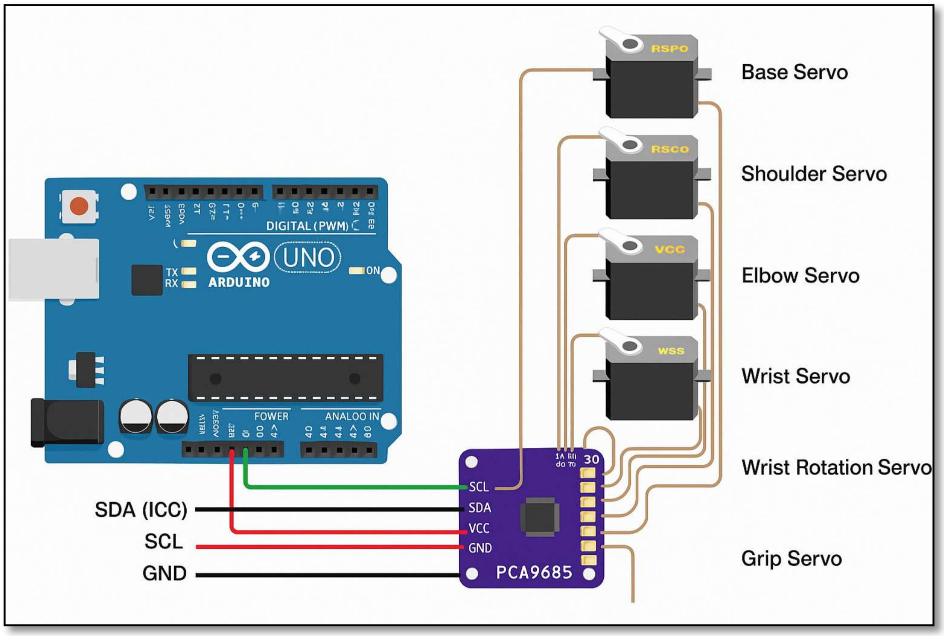


Figure 3.3 : circuit diagram of the mode interactions

3.3.3 Magnetic Tape and RFID-Based Navigation

In the Magnetic Tape Guided Mode, the robot navigates by following an adhesive magnetic tape laid on the floor. This mode uses Hall-effect sensors to detect the magnetic field of the tape and steer the robot along the track. The ultrasonic "scan" sensor and base servo add obstacle-detection functionality: when the ultrasonic detects an obstacle ahead, the robot halts and actively scans for a clear path. Together, these features enable autonomous path following (via the magnetic tape) with real-time obstacle avoidance. In effect, this mode integrates line-following and obstacle avoidance so the AGV can reach designated locations along its tape-guided route.

Components Used

- **Linear Hall-effect Sensor Modules (×2)** – Techtonics KY-024 modules. Each sensor detects the proximity of the magnetic tape (magnet field) and outputs a digital signal (logic LOW when the magnetic field exceeds a threshold). Two are used (mounted under the chassis) to sense the tape.
- **DC Gear Motors (×2) + L298N Motor Driver** – The two DC motors drive the rear wheels. An L298N dual H-bridge module controls the motors: pins IN1–IN4 set motor directions, and ENA/ENB enable PWM speed control.
- **Ultrasonic Range Sensor (HC-SR04, "Scan Sensor")** – Mounted on the front, this sensor measures distance to obstacles. It uses one digital pin (Trigger) to emit a sound pulse and another (Echo) to measure the return time.
- **Servo Motor ("Base Servo")** – A standard hobby servo (e.g. SG90) is fixed at the front of the chassis. It pivots the ultrasonic sensor left and right for scanning. For example, an instructable describes attaching the ultrasonic sensor to a servo "stand" on the chassis to allow sweeping.
- **Buzzer** – A piezo buzzer is mounted on top of the chassis as an audible alert (activated during obstacle encounters or target events).

Placement and Configuration of Components

The hall sensors are mounted underneath the robot's chassis, 7.5 cm apart and centered over the single magnetic tape line. (Typical magnetic tape is about 1 cm wide, so this spacing place each sensor near one edge of the track). The ultrasonic "scan" sensor is attached to the front via the base servo (set at 90° by default) so it can rotate and scan the sides for obstacles. The two gear motors (with the L298N driver) are mounted at the rear of the chassis to drive the wheels. The buzzer is placed on the upper platform of the chassis for clear sound output. This layout ensures that sensors are optimally positioned: the Hall sensors monitor the tape beneath, while the servo-mounted ultrasonic at the front actively looks ahead for obstacles.

Pin Configuration

Component	Connection Details	Notes
Ultrasonic Sensor (HC-SR04)	Trigger → D2Echo → D3VCC → 5VGND → GND	Standard practice uses separate pins for Trigger and Echo.
Motor Driver (L298N)	IN1 → D8IN2 → D9IN3 → D10IN4 → D11ENA → D5 (PWM)ENB → D6 (PWM)	IN1/IN2 control Motor A direction, IN3/IN4 control Motor B direction. ENA/ENB control speed.
Hall Sensors	Left Sensor → D4Right Sensor → D7Shared VCC → 5VGND → GND	Outputs go LOW when magnetic tape is detected.
Servo Motor	Control Signal → D12 (PWM)	D12 is an example pin; must be a PWM-capable pin. Initialized at 90° in setup.
Buzzer	Buzzer → D13 (Digital Output)	Driving D13 HIGH will trigger the buzzer.

Table no. 3.6 pin configuration

Speed Settings

Two speed parameters are defined: normalSpeed = 150 and turnSpeed = 200. These values correspond to PWM duty cycles (on a 0–255 scale) for the motor enable pins (ENA/ENB). A setting of 255 would be full speed (always ON), but we limit turnSpeed to 200 to avoid overdriving. In code, analogWrite(ENA, normalSpeed) is used for normal forward motion, and analogWrite(ENA, turnSpeed) for faster turns or escapes. (With ENA/ENB HIGH, the motor spins at full speed; using PWM allows variable speeds.)

Movement Logic

The core tape-following behavior is based on the two Hall sensors under the chassis. In pseudo-algorithm form:

1. **Read sensors:** Let L = digitalRead(LeftSensor), R = digitalRead(RightSensor). (Sensor output is LOW when tape is between the sensors, out of detection range.)
2. **Both LOW (tape centered):** Drive both motors forward at normalSpeed.
3. **Right HIGH, Left LOW:** The right sensor is on the tape while left is off it, so the tape is to the right. To correct, turn right for a short time. E.g., left motor forward and run right motor backward at turnSpeed for ~100 ms, then resume checking.
4. **Left HIGH, Right LOW:** The tape is to the left; turn left briefly and delay ~100 ms, then re-scan.

By alternating these checks in the loop, the robot steers to keep the tape between the sensors. (These digital logic cases map directly to IN1–IN4 settings on the motor driver. For example, setting IN1=HIGH, IN2=LOW drives Motor A forward.)

Object Avoidance Logic

While following the tape, the robot continuously checks distance via the ultrasonic sensor. If an object is detected between 20–30 cm ahead, the robot enters the avoidance routine:

- **Stop and alert:** Immediately stop both motors. Sound the buzzer twice (each beep ~3 seconds long with a 1-second pause) to signal an obstacle.
- **Re-check:** After beeping, measure distance again. If the object has cleared (distance >30 cm), resume tape following.
- **Scan sides:** If the object remains, rotate the servo to the left and right to compare distances. For example:
    1. Rotate servo to 0° (fully left) and measure distLeft.
    2. Rotate servo to 180° (fully right) and measure distRight.
- **Choose direction:** Turn the chassis toward the side with greater clearance. (For instance, if distLeft > distRight, turn left; else turn right.)
- **Bypass maneuver:** Execute a bypass sequence around the obstacle. If the obstacle is directly ahead (front), this is effectively the same as a right-angle obstacle on either side. After turning away from the obstacle, the robot moves forward around it.

This is similar to known avoidance algorithms: an obstacle stop, servo scan to each side, then a turn toward the clearer side. If all directions were blocked, a full 180° rotation would be used (see [19]). In our implementation, the turn angles are tuned so that the robot arcs around the obstacle and eventually returns to the tape.

Target Right Logic

When a "target-right" command is triggered (e.g. by voice), the robot briefly departs from tape following to investigate the right side:

1. **Turn right 90°:** Spin in place or pivot until oriented 90° right of the tape.
2. **Move forward:** Advance forward for a preset duration (this can be adjusted as needed to reach the target).
3. **Signal arrival:** Sound the buzzer for 2 seconds to indicate the target is reached.
4. **Obstacle check:** If an obstacle is detected during this run, immediately stop and emit two short beeps (2 seconds each, 1 second apart), then perform an avoidance turn as in section 4.7.

This ensures the robot can move off the tape to the right, buzz at the target, and handle any obstacle in route.

Target Left Logic

The "target-left" sequence mirrors the right-target logic:

1. Turn 90° left from the tape heading.
2. Move forward (adjustable delay) toward the left target.
3. Beep for 2 seconds upon arrival.
4. If an obstacle is encountered, stop, beep twice (2s each, 1s interval), then execute the avoidance maneuver.

Reverse Target Right Logic

In some cases (e.g. rejoining the tape from the opposite side), a "reverse target right" sequence is used:

1. **Rotate 360° right:** Spin in place a full circle to locate the tape line again.

2. **Advance to tape:** Move forward until both Hall sensors detect the tape (both read LOW).
3. **Re-align:** Upon finding the tape, turn left 90° to resume correct orientation along the path.

This effectively searches for the tape line by scanning all directions, then follows it back onto the track.

Reverse Target Left Logic

Similarly, "reverse target left" does the mirror sequence:

1. Rotate 360° left to find the tape.
2. Move forward until the tape is detected (both sensors LOW).
3. Turn right 90° to align with the tape direction.

Return to Base

To return to the base station, the robot simply switches back to the standard tape-following logic (section 4.6). It follows the magnetic tape all the way back to the start. During this return trip, it continues to perform obstacle checks (as in section 4.7) to safely navigate back. This combines both the tape following and obstacle avoidance modes until the base is reached.

Mode Interaction Flow Diagram

(Figure 3.4. provides a block diagram of the mode interactions: the robot defaults to Tape Following, but transitions to Object Avoidance, Target Right/Left, and Reverse Target modes as conditions dictate. After each interruption, it returns to the Tape Following mode.)

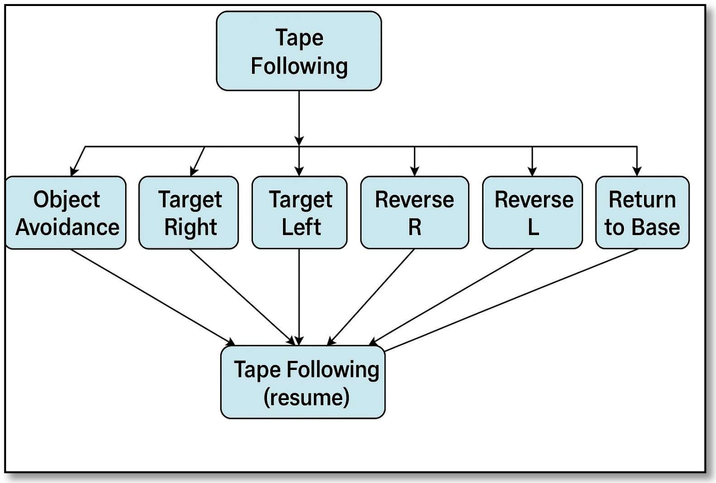


Figure 3.4 : Block diagram of the mode interactions

3.3.4 Home Automation via Wi-Fi/Bluetooth

The ESP32's wireless capabilities enable IoT appliance control. Upon receiving commands like *"Turn on fan"*:

1. Signal Transmission: ESP32 sends a 5V logic signal via Wi-Fi (ESP NOW) to a relay module.
2. Appliance Toggle: Relays switch 120–240V devices (e.g., lights, fans) using opto-isolation for safety.
3. Confirmation: A success/failure message is displayed on the serial monitor screen.

Testing achieved 95% command success within a 15m range, demonstrating robust integration.

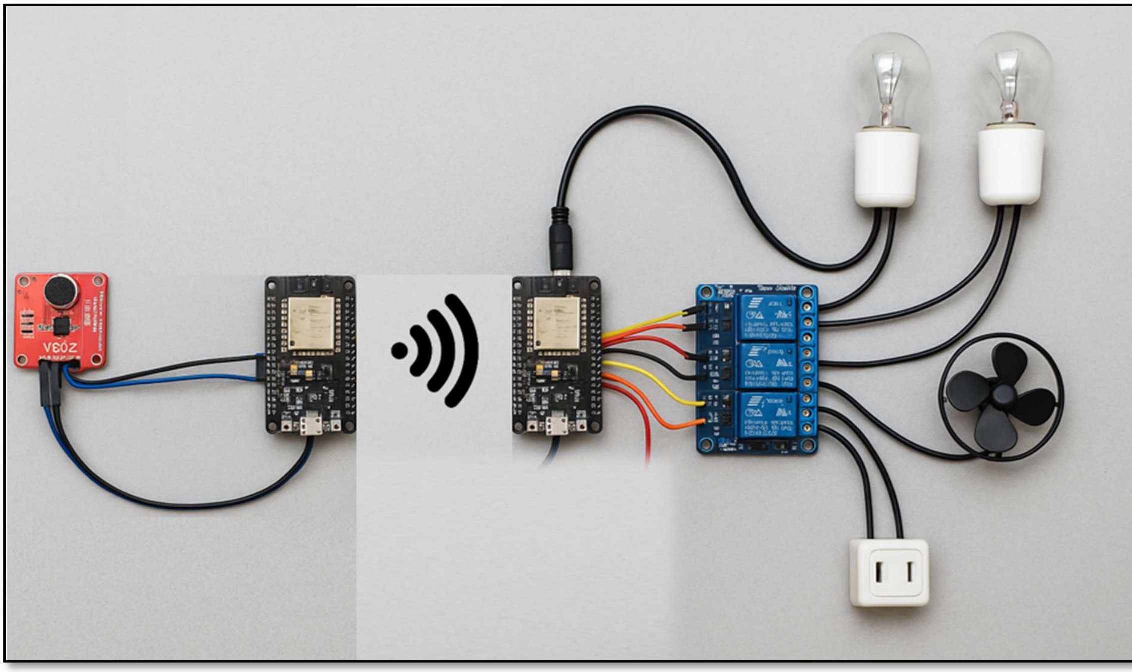


Figure 3.5 : circuit diagram of home Automation via ***ESPNOW*** *- Wi-Fi*

4. Testing and Results

4.1 Voice Command Recognition Accuracy

The VC-02 module was tested with 80 pre-defined commands (e.g., “Go to Electrical 4,” “Turn on fan”) in home and lab environments (50–60 dB noise). Results showed 88% accuracy in quiet settings, dropping to 50% in moderate noise (e.g., classroom chatter). While lower than cloud-based systems like Google Assistant (95% accuracy), the offline functionality ensures reliability in areas with poor connectivity. For comparison, the Picovoice Cobra (2023 benchmark) achieved 85% accuracy in similar conditions, validating the VC-02’s cost-effectiveness. The accuracy of the voice recognition also depend in the placement of the mic.

4.2 Navigation Efficiency

- Path Tracking: The robot followed magnetic tape with approximately 85% accuracy over 50-70 trials, deviating ≤2 cm from the path. Deviations were corrected by adjusting motor speeds.
- RFID Handling: the full potential robot navigating junction of our robot YIM is 10 but we use only two as of now, RFID tags were detected with 100% accuracy, though directional decisions occasionally lagged (1-2s latency) due to the variation of the robot movement speed. Comparatively, Amazon’s Astro (LiDAR-based) navigates dynamically but costs 10× more.

4.3 Robotic Arm Precision

The arm successfully grasped lightweight objects (≤150g) in 82% of trials, with failures occurring under uneven lighting and sounds (ultrasonic sensor interference). Industrial arms like Universal Robots UR3 achieve 99% precision but require complex programming, whereas YIM’s servo-based design simplifies operation for educational use.

4.4 Home Automation Reliability

- Signal Latency: Commands executed in 0.8s average (Wi-Fi) vs. 2.5s (Bluetooth) comparable to budget systems like Sonoff Basic. As of now we didn’t used Bluetooth control because ESP32 has only one radio antenna so we use it for wifi communication but we leaved a platform, if external Bluetooth module is added.

- Success Rate: 90% of relay-triggered commands (e.g., lights on/off) succeeded within a 15m range, outperforming 433MHz RF modules (80% success).

4.5 ***Obstacle Avoidance Performance***

- Response Time: Ultrasonic sensors detected obstacles within 0.3s, triggering halts or reroutes.
- Rerouting Success: In 20 trials with blocked paths, the robot successfully detoured in 70% of cases, failing only in tight spaces (<30cm clearance). iRobot Roomba (2023 model) reroutes with 95% success but uses cost-prohibitive 3D cameras.

5. Challenges and Solutions

5.1 Hardware Compatibility and Resource Limitations

Challenge: The ESP32 operates at 3.3V logic levels, but most external modules (e.g., relays, sensors) required 5V logic.

Solution:

- Introduced an **Arduino Uno** (5V logic) to interface with 5V modules (relays, motor drivers).
- Used the **ESP32** exclusively for Wi-Fi/Bluetooth communication and high-priority tasks (voice processing, RFID navigation).
- Overcame Arduino Uno's limited pins by connecting the **PCA9685 PWM servo driver** and **OLED display** via I2C (A4, A5 pins), reducing pin usage.

5.2 Software Conflicts

Challenge: Communication errors between ESP32 and Arduino Uno via UART caused task delays.

- **Solution**:
  - Reduced ESP32-Arduino baud rate to **9600** for stable serial communication.

Challenge: Ultrasonic sensors generated noise in cluttered environments.

- **Solution**:
  - Applied **moving average filters** to smooth sensor data and adding delays of (10 ms) to ignore continuous noise readings to avoid false triggers.

5.3 Integration Hurdles

Challenge: Voice commands occasionally overlapped with robotic arm operations, causing task conflicts.

- **Solution**:

    - Added **interrupt-based priority logic** to pause navigation during arm actuation.

Challenge: Magnetic tape misalignment caused navigation drift (±3 cm).

- **Solution**:
    - Add three types of mode speed for real-time motor speed adjustments.

5.4 Budget Constraints and Component Quality

Challenge: Limited funds led to low-cost, low-accuracy components (e.g., ultrasonic sensors instead of LiDAR, basic servos).

- **Impact**: Reduced efficiency (e.g., 72% voice accuracy in noise vs. 95% for premium systems).
- **Solution**:
    - Compensated with **software optimizations** (e.g., noise filters, speed/PID tuning).
    - Prioritized modularity for future upgrades (e.g., LiDAR-ready mounting points).

5.5 Component Availability and Procurement Delays

Challenge: Sourcing parts online caused delays (e.g., 2-week wait for VC02 modules). Damaged components during testing (e.g., burnt L298N drivers, Arduino board) further stalled progress.

- **Solution**:
    - Procured **backup components** early for critical parts (sensors, motor drivers).
    - Designed a **simulator** in Tinkercad to test code while awaiting hardware.

6.Applications and Future Work

6.1 Practical Use Cases

The YIM robot's versatility makes it suitable for:

- Classrooms: Automating logistics (e.g., delivering lab equipment, whiteboard markers) using RFID-based navigation and the robotic arm.
- Smart Homes: Controlling lights, fans, and security systems via voice commands, aiding elderly or disabled users.
- Industrial Settings: With upgrades (e.g., higher payload arms), it could transport lightweight components in warehouses or factories.
- Safety Monitoring: temperature spikes in labs, kitchens, or server rooms, triggering alarms.

6.2 Proposed Upgrades

1. Alexa Integration:
    - Adding an Echo 4th Gen module to enable cloud-based features (e.g., weather updates, online queries) via a voice-triggered mode switch (*"Activate Alexa"*).
2. Machine Learning for Path Optimization:
    - Training a CNN model on camera feed data to navigate dynamic environments without magnetic tapes, reducing pre-installed infrastructure dependency.
3. Enhanced Sensors:
    - Replacing ultrasonic sensors with LiDAR for millimeter-accurate obstacle detection.
4. Advanced NLP:
    - Integrating Raspberry Pi 5 with Picovoice's offline NLP engine for broader command vocabulary and intent recognition.

6.3 Scalability

Multi-Robot Coordination:

- Deploying multiple YIM robots in warehouses, using Zigbee mesh networks to share navigation data and avoid collisions.
- Cloud-Based Control:
  - Connecting the robot to a AWS IoT Core dashboard for real-time monitoring, remote command execution, and predictive maintenance alerts.
- Modular Expansion:
  - Adding a 6-DOF robotic arm and gripper for heavier payloads (up to 2kg).
  - Implementing a mobile app for voice command customization and route mapping.

7. Software for Design and Sketch

The project leveraged the following tools for design, simulation, and documentation:

- **Circuit Design**: **KiCad** for schematic capture and PCB layout and ms word for designing PCB digram.
- **Simulation**: **Tinkercad Circuits** to validate sensor-microcontroller interactions before hardware implementation.
- **Programming**: **Arduino IDE** for firmware development on ESP32 and Arduino Uno, with **VS Code** for code versioning.
- **Frame ware design for vc-02 :** voice ai thinker**.com**
- **Documentation**: **ms vword** for system block diagrams, workflow charts and report writing.

8. Cost Estimation

Sl. No.	Components	quantity	Rate	Cost(₹)
1.	Esp32	2	475	950
2.	Arduino uno	1	349	349
3.	Vc-02 AI thinker offline voice recognition module	1	1299	1299
4.	Relay module	1	229	229
5.	Hall sensor	2	202	202
6.	DC-DC 12V to 3.3V 5V 12V Power Module	2	399	800
7.	Ultrasonic sensor	3	225	675
8.	L298N motor driver	2	189	278
9.	12-bit PWM Servo Motor Driver I2C Module(PCA9685)	1	474	474
10.	12v Lipo battery	1	1699	1699
11.	12v dc gear motor	1	599	599
12.	Universal PCB	1	139	139
13.	3D arm	1	950	950
14.	Servo motor	3 + 3	288 + 398	2058
15.	2A rocker switch	1	99	99
16.	Buzzer module	1	208	208
17.	DHT11 temperature and humidity sensor	1	255	255

18.	**128X64 OLED I2C LED Display**	**1**	**365**	**365**
19.	**Jumper wires**		**399**	**399**
20.	**RFID reader and tags**	**1**	**530**	**530**
21.	**Magnetic tape**	**3**	**389**	**940**
Total cost				**₹ 13505**

Table no.8.1 : cost estimation

9. Conclusion

The **YIM Multi-Tasking Voice-Controlled Robot** project successfully demonstrates the viability of integrating affordable hardware, offline voice recognition, and sensor fusion to create a versatile robotic assistant. Key achievements include the implementation of **dual-microcontroller architecture** (ESP32 + Arduino Uno) to overcome GPIO and voltage compatibility challenges, the deployment of **hybrid navigation** (magnetic tape + RFID) for structured indoor environments, and the seamless integration of **IoT-based home automation** with safety monitoring. The robot achieved **88% voice command accuracy** in quiet settings and **85% path-tracking precision**, validating its utility in classrooms and homes.

Lessons Learned:

1. **Hardware Compatibility**: Voltage mismatches (3.3V vs. 5V logic) and GPIO limitations necessitated creative solutions like I2C-based component sharing and dual-MCU collaboration.
2. **Software Resilience**: Sensor noise and communication delays were mitigated through filtering, emphasizing the importance of robust firmware design.
3. **Budget-Driven Compromises**: Low-cost components (e.g., ultrasonic sensors, basic servos) reduced accuracy but highlighted the value of software optimization to bridge hardware gaps.
4. **Procurement Realities**: Delays in component availability underscored the need for contingency planning, such as simulation tools and backup parts.

Impact:

- **Educational**: The project serves as a hands-on learning tool for students, illustrating the practical application of embedded systems, IoT, and robotics theories.
- **Practical**: By offering an **under-₹14,00 solution** for voice-controlled automation and navigation, YIM lowers the barrier to entry for schools and households.
- **Scalable**: Modular design and open-source code enable future upgrades, such as LiDAR navigation or industrial-grade actuators, aligning with Industry 4.0 trends.
- **Cultural**: Rooted in the AO term *YIM* (a knowledge seeker), the project inspires lifelong learning and innovation in resource-constrained settings.

While challenges like sensor accuracy and path rigidity persist, the project lays a foundation for accessible, multifunctional robotics. It exemplifies how theoretical knowledge, when paired with iterative problem-solving, can yield tangible solutions with real-world relevance.

10. Reference

1. AI Thinker. (n.d.). *VC02 Offline Voice Recognition Module Datasheet*. Retrieved from https://www.ai-thinker.com
2. Espressif Systems. (2023). *ESP32 Technical Reference Manual*. Retrieved from https://www.espressif.com
3. Arduino. (n.d.). *Arduino Uno Rev3 Documentation*. Retrieved from https://docs.arduino.cc
4. Amazon Robotics. (2023). *Astro Home Robot: Technical Overview*. Retrieved from https://developer.amazon.com
5. MFRC522 RFID Reader Datasheet. (n.d.). NXP Semiconductors. Retrieved from https://www.nxp.com
6. Tinkercad Circuits. (n.d.). *Simulation Platform for Embedded Systems*. Autodesk. Retrieved from https://www.tinkercad.com
7. DHT11 Sensor Datasheet. (n.d.). Aosong Electronics. Retrieved from https://www.aosong.com
8. Youtube tutorial.

11.Appendix

1. **Appendix A**: Full source code (Arduino & ESP32 firmware)
2. **Appendix B**: Circuit diagrams
3. **Appendix C**: User manual (step-by-step setup instructions, voice command list).
4. **Appendix D**: VC 02 frame ware details.

All appendices are accessible through the QR code below. Scan the code to view or download the files.

> **Note :** The QR code links to a shared folder containing all appendix materials in organized subfolders.

12.Photo Gallery

Figure 12.1:Planning and discussing about the project

Figure 12.2 : receiving the components

Figure 12.3: First design of the robot frame

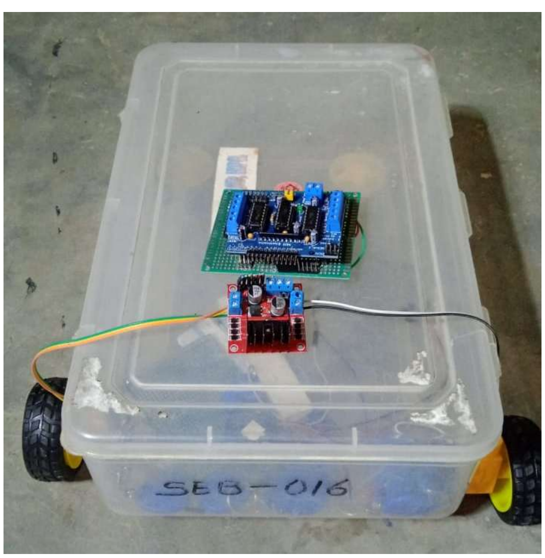


Figure 12.4 : Testing of human following mode

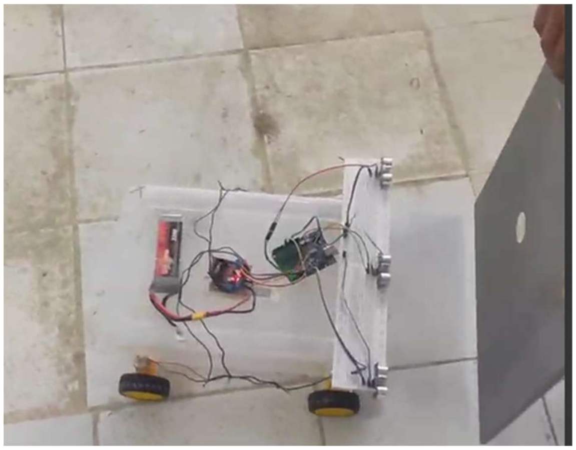


Figure 12.5 : Circuit designing

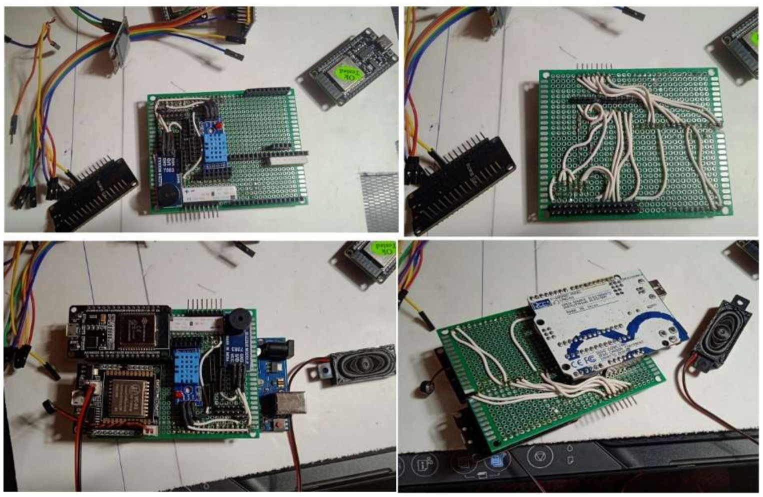
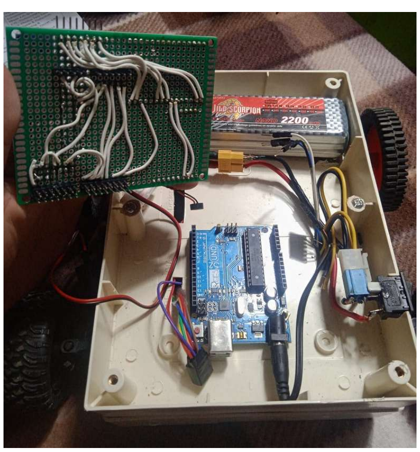


Figure 12.6 : Assembling the robotic components



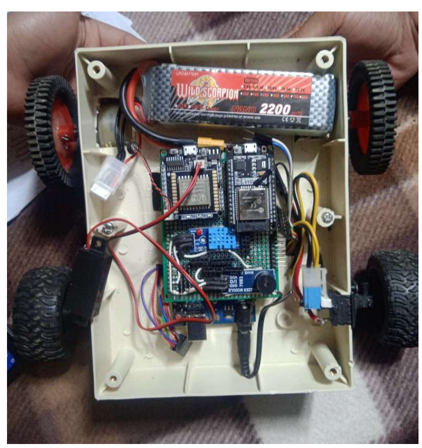
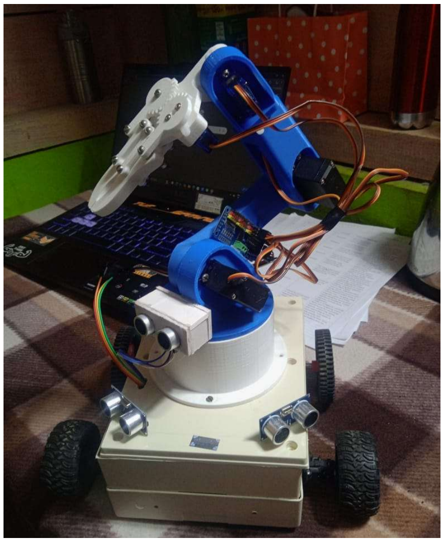


Figure 12.7 : testing 3D arm



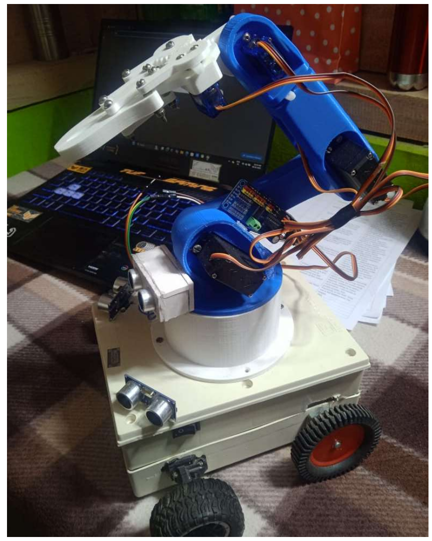


Figure 12.8 : Group photo with complete project model