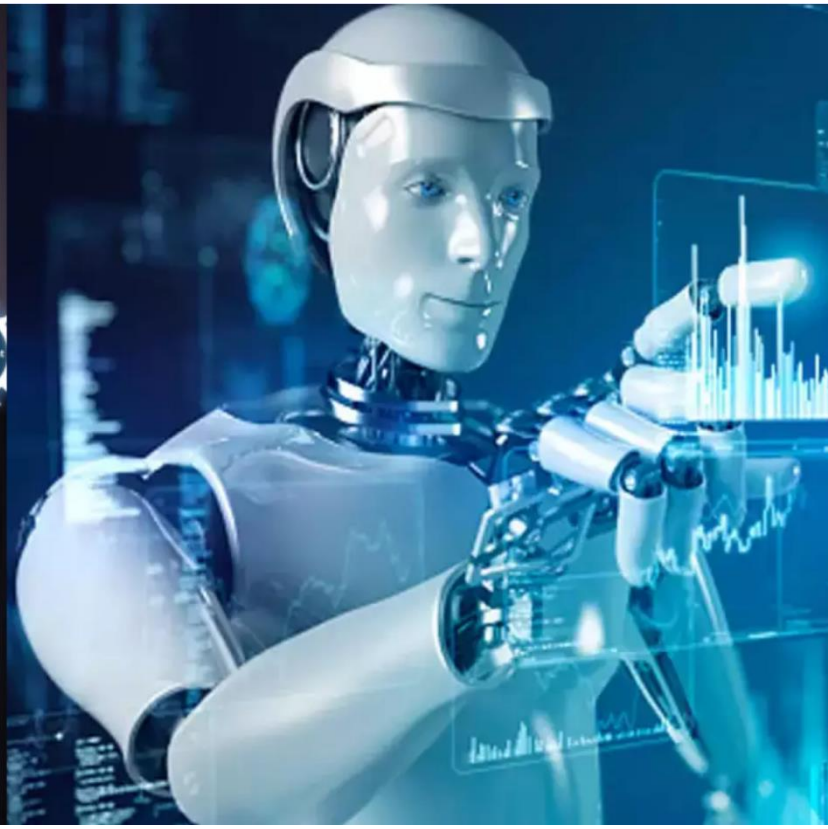
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 6, June 2025

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406074|

Design and Implementation of a Privacy-Focused AI Voice Assistant using ChatGPT/Gemini on ESP32

Dr R R Itkarkar, , Prajwal Karande, Omkar Vartak, Anisha Kandhare, Dr D S Bormane

Dept. of ENTC, AISSMS College of Engineering, Pune, Maharashtra, India

ABSTRACT: AI assistants have become an essential part of modern life, enabling hands-free interaction with technology. Devices like Amazon Alexa and Google Assistant have set the standard for voice-controlled systems. Such devices offer multiple functionalities for smart homes. However, these voice assistants often raise concerns about privacy due to their continuous listening features. This research aims to solve concerns about unauthorized data collection related to voice assistants and provide a cost-effective, privacy-focused AI voice assistant using Generative AI models like ChatGPT/Gemini integrated into an ESP32 microcontroller.

KEYWORDS: AI Voice Assistant, Speech-to-Text, Text-to-Speech, ChatGPT, Gemini, API.

I. INTRODUCTION

Voice assistants, like Amazon Alexa, Google Assistant, and Siri, have changed the way people interact with technology. These systems made day-to-day tasks easier by allowing hands-free interaction. However, these voice assistants also raise important privacy and security concerns. Voice assistants are continuously listening to detect wake words, such as 'Alexa' or 'OK Google'. This word helps to identify the question of the user. This means they are constantly monitoring the environment. This can lead to unauthorized data collection and security risks, as private conversations of users and personal data can be recorded and stored on servers without user's permission.

This research aims to address these concerns and create an alternative voice assistant that focuses on user privacy, customization, and cost-effectiveness. The system uses the ESP32 microcontroller with a large language model such as ChatGPT/Gemini. The System works by capturing the user's question in audio format through a mic, which is then converted into text using Speech-to-Text API. The question in text format is provided to ChatGPT/Gemini for generating responses. The responses are then transformed back into audio through a Text-to-Speech (TTS) service and played through a connected audio amplifier and speaker. This paper presents the implementation and analysis of project originally developed by techiesms. The system was recreated based on the published design for academic learning and evaluation purposes. The system is designed to operate efficiently in various environments, supporting applications such as virtual assistants, smart home devices, or accessibility tools. It ensures low-latency processing and can integrate with additional hardware or software components, such as cloud-based storage for conversation history or IoT devices for executing commands, thereby providing a scalable and versatile solution for voice-driven interactions.

II. LITERATURE REVIEW

[1] The paper presents the design and implementation of personal AI Companion and Voice Assistant using ESP32 microcontroller. The author addressed the privacy and security concerns in current voice assistants and proposed a system which uses voice recognition & NLP for processing user query and providing audio output.[2] The paper discusses the integration of ChatGPT functionality with voice assistants in a game named Farcana to enhance user's gaming experience, interaction and automate processes. [3] The authors demonstrate use of the OpenAI API library to use ChatGPT, OpenAI's large language model into Python programs, making chatbots more realistic and interactive for users. [4] The paper talks about a multilingual voice assistant who runs on Raspberry Pi having a mic and speaker connected. The system utilizes Natural Language Processing (NLP), and various APIs for speech-to-text and text-to-speech conversion enabling hands free interaction of the users.[5] Paper identifies six main types of vulnerabilities of voice assistants, including weak authentication, replay attacks, and cloud infrastructure weaknesses. Also, various

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406074|

technical countermeasures are discussed to protect against these vulnerabilities, such as disabling microphones and implementing voice authentication.[6] The article examines security and privacy issues in voice assistant applications and focusses mainly on the Automatic Speech Recognition (ASR) and Speaker Identification (SI) models. Authors of the paper have highlighted types of attack which target machine learning models and hardware components in voice assistants, such as backdoor attacks, adversarial attacks, and hidden command attacks.

The reviewed literature highlights several key advancements and ongoing challenges in the development of AI-powered voice assistants. Many papers discuss the design and implementation of voice assistants on microcontroller platforms, such as the ESP32 and Raspberry Pi. Integrating them with NLP algorithms or using ChatGPT for generation of response for the user. Some Papers focuses the vulnerabilities and security threats in current voice assistants, including weak authentication, replay attacks, and privacy breaches and proposes defensive methods, the need for more secure, privacy-focused voice assistants remains a priority. In contrast to existing solutions, the project leverages the low-cost ESP32 microcontroller to build a customizable and privacy-focused AI voice assistant. By addressing the identified gaps such as insufficient privacy measures. This project supports custom use cases and offers flexibility in integration with smart home systems, making it a practical alternative for users seeking greater control over their data and more functionality. Thus, this project aligns with the evolving market demand for secure, private, and affordable voice assistants

III. METHODOLOGY

This research aims to develop a voice assistant system with ESP32. The main task is connecting ESP32 with mic, audio amplifier, and SD card reader module. The audio input (user question) will be recorded inside the SD card and then processed using a Speech-to-Text API. Users' questions in text will be provided to AI-based APIs like ChatGPT/Gemini. The system will then convert the responses into audio using the Text-to-Speech API. The final audio is output through the speaker

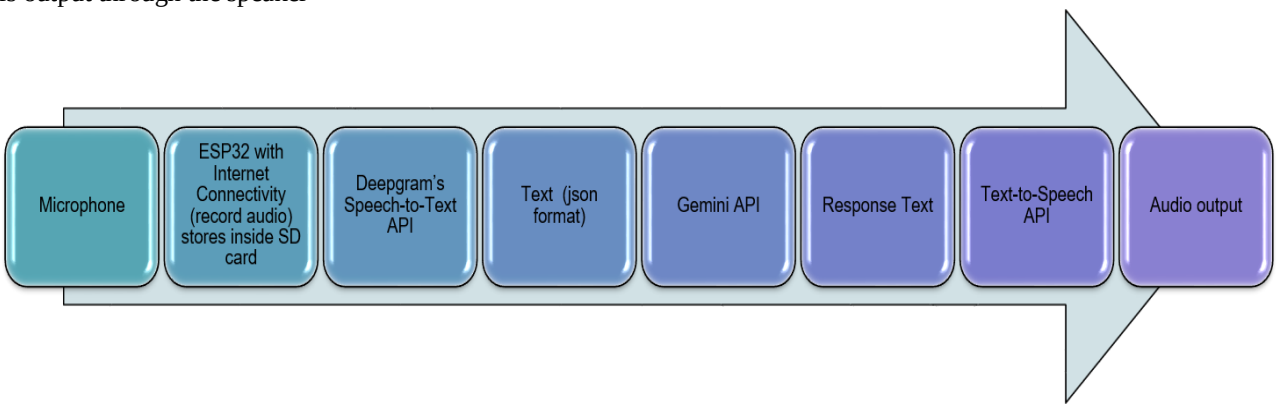


Fig. 1: Methodology

The figure1 illustrates the workflow of the proposed AI voice assistant system. The process begins with the user speaking a command into a microphone connected to the ESP32 microcontroller. This audio input is recorded and stored on an SD card using an SD card module interfaced with the ESP32. The recorded audio is then sent to a cloud-based Speech-to-Text (STT) service, such as Deepgram, which processes the audio and converts it into corresponding text. The ESP32 receives this text and forwards it to an AI model—either ChatGPT or Gemini—via an API call. The AI model analyzes the input and generates a contextually appropriate response. This response, in text form, is then passed to a Text-to-Speech (TTS) API, which synthesizes the text into spoken audio. Finally, the converted speech is played back to the user through a speaker connected to the ESP32, completing the interactive voice assistant cycle.

ESP32 Microcontroller

At the core of the system, its ESP32 microcontroller. First, ESP32 captures voice commands through a microphone, processes the audio data, and sends them to AI services to generate responses. The response is in textual format and is provided to ESP32. Then the AI-generated text responses are converted into speech and provided to the user. This is done using Google's TTS service.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406074|

Microphone and Speaker Integration

For high-quality audio capture, the ESP32 is paired with an INMP441 microphone module, which uses its I2S interface for audio input. For output in audio format, a MAX98357 audio amplifier and an 8-ohm speaker are used.

Fig 2: INMP441mic and MAX98357 Audio Amp.

SD Card & Card Reader Module

The internal memory of ESP32 is limited, hence large audio file containing user's query cannot be stored inside ESP32. To overcome this limitation, we can use external memory such as an SD card. The audio file which will be generated for storing the user's question will be saved inside SD card. This can be done using Micro SD card reader module like HW-125. SD card reader module utilizes the Serial Peripheral Interface (SPI) for communication with the microcontroller.

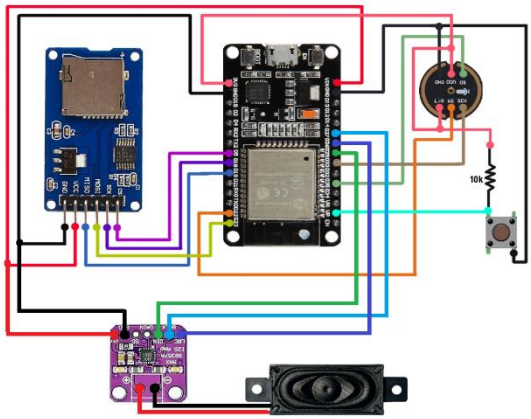


Fig 3: Circuit design

Programming Environment

ESP32 is programmed with the Arduino Integrated Development Environment (IDE). The IDE provides open-source libraries and tools for programming ESP32. Such tools made programming easier and efficient. The programming code is written in C++.

AI Integration

After recording and transcribing the user's question, the text is forwarded to AI service such as OpenAI's ChatGPT or Google's Gemini through an API service. The AI generates response based on the context of question asked. The responses are then sent back to the ESP32. Both Gemini[9] and ChatGPT[7] are advanced AI models, but they have their own strengths that affect their response generation. ChatGPT has shown very strong abilities in complex reasoning and code generation. However, Gemini is designed with Google data, and therefore has more contextual awareness. This makes Gemini a better choice as a voice Assistant.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406074|

Speech-to-Text Processing

The ESP32 captures audio input from the microphone and processes it with an audio library that works with the INMP441 module. Once processed, the audio data is sent to a cloud-based Speech-to-Text (STT) service, which converts spoken commands into text format. Various platforms provide STT service. Ex. Google, Deepgram. However, Deepgram's STT services is more suitable for real-time Speech-to-text conversion. To use the service API key is required, which provides authentication. Key can be obtained from the official website of service provider. API keys can be tested applications like postman.

Text-to-Speech Conversion

The ESP32 takes the text response received from the AI model and converts it into speech using a compatible audio library for the MAX98357 module. This audio library[10] is developed by the GitHub user schreibfaul1. It is open-source and publicly available. The generated audio is played through the connected speaker.

IV. RESULTS

API Acquisition:

1. **Deepgram Speech-to-Text API:**
   - This API was used to convert spoken language into text. It was tested in the Postman application to ensure accurate transcription and smooth communication with the API.
   - The test confirmed that the API can reliably process audio input and return a transcribed text output, which is essential for converting user voice commands into text for further processing in the system.
2. **Gemini API Integration:**
   - The Gemini API was tested in the Postman application to verify its functionality and how it could interact with the ESP32.
   - After successful testing, the Gemini API was integrated with the ESP32 microcontroller. The integration allowed the ESP32 to send and receive requests, with the responses from the API being generated and processed in real time.
   - The API responses were displayed on the ESP32's **serial output**, confirming that the integration works as expected and that the ESP32 can communicate effectively with external services to generate intelligent responses.

Glipms:

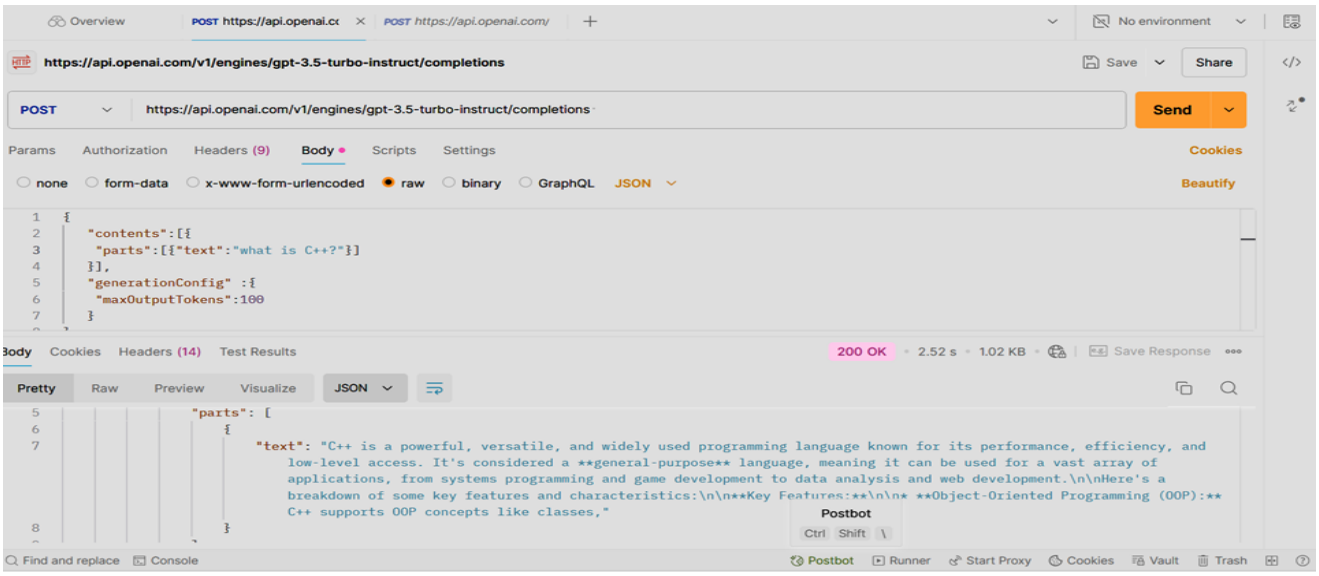


Fig 4 : Deepgram API in Postman

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406074|

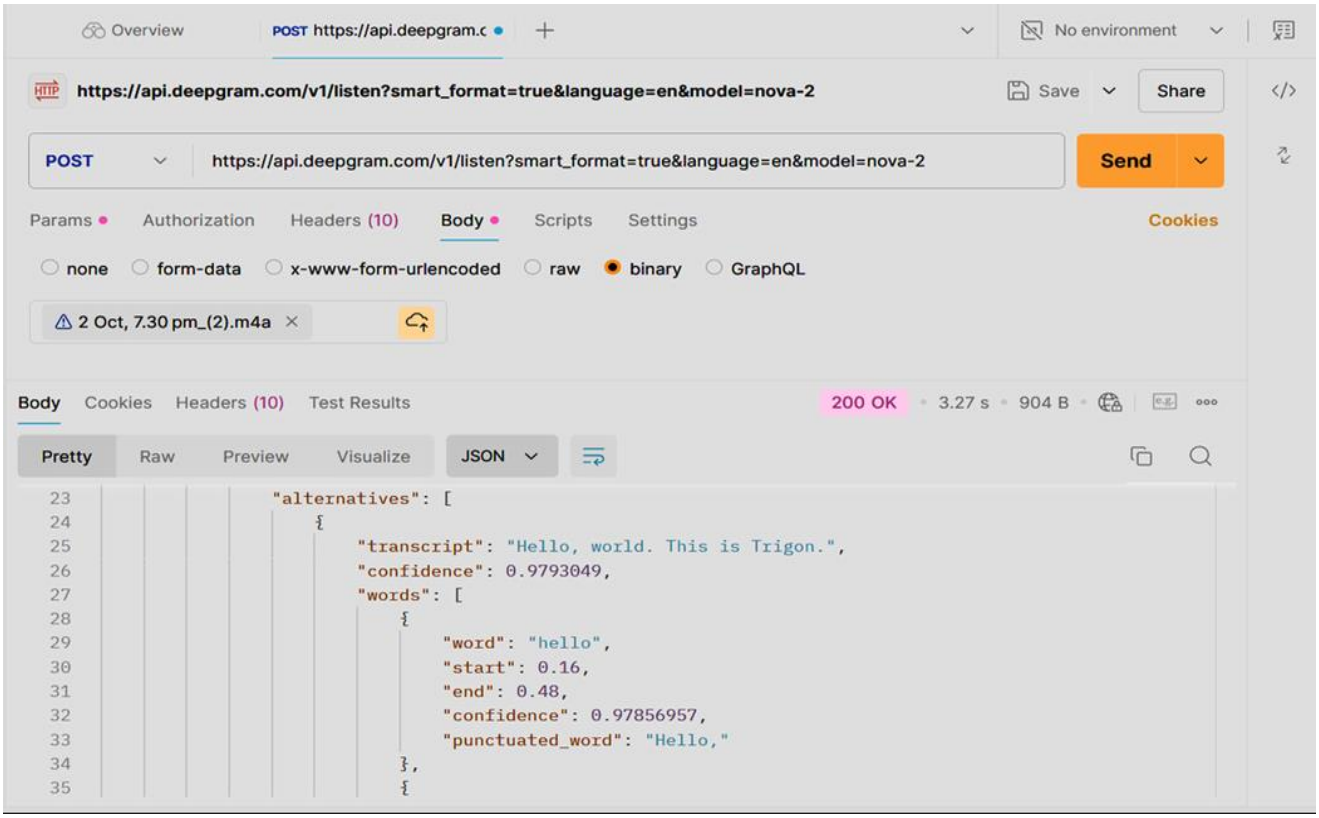


Fig 5 : Gemini API in Postman

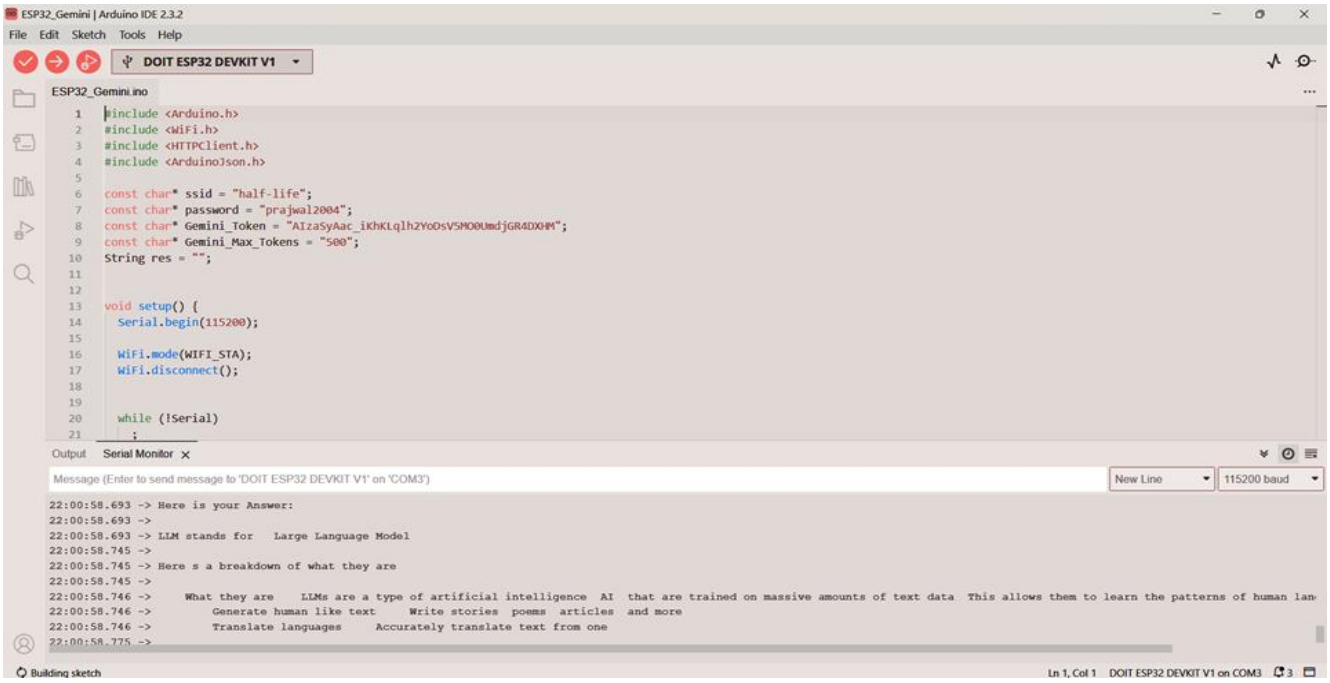


Fig 6: Gemini on ESP32

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406074|

Custom PCB for AI Voice Assistant System

The final system for the AI voice assistant utilizes a custom-designed Printed Circuit Board (PCB) developed in collaboration with a professional PCB design and assembly company. This custom PCB integrates the ESP32-WROOM-DA microcontroller, HW-125 microSD card reader module, INMP441 I2S digital microphone, MAX98357A I2S audio amplifier for Text-to-Speech output, BMS System with Lithium ion battery and supporting components into a compact and efficient layout. The PCB design optimizes signal integrity, power distribution, and thermal management to ensure reliable performance for real-time voice processing and AI integration. The company provided comprehensive services such as PCB layout, and full assembly.

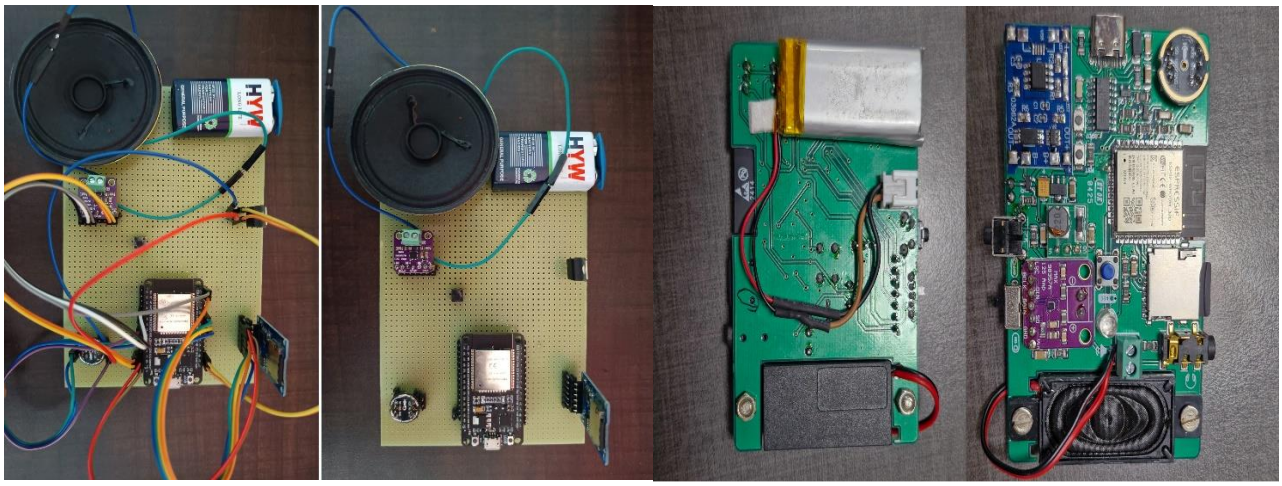


Fig 4: System Implementation

VII. CONCLUSION & FUTURE SCOPE

This research demonstrates a functional, affordable, and privacy-conscious AI voice assistant using the ESP32 microcontroller and generative AI services. By limiting continuous cloud-based data transmission and storing audio locally, the system reduces potential privacy violations common in commercial devices. Additionally, the modular architecture supports flexibility in selecting AI models and APIs, making it highly customizable. The implementation proved successful in enabling two-way voice interaction using advanced AI tools, all while being mindful of hardware limitations and user privacy. It validates the feasibility of using microcontrollers for AI-based voice assistants, offering a valuable alternative to commercial voice assistants.

The AI assistant developed using the ESP32 can act as a framework and has vast future enhancements and applications. It can act as a backbone of various IoT system. Some key areas for future development are:

- Integration with Additional AI Models: Integrating the system with other AI models for applications like image recognition, object detection.
- Multilingual Support: Adding support for languages other than English, making it more user-friendly and suitable.
- Cloud and IoT Ecosystem: Integrating the system with IoT devices and platforms such as MQTT, AWS IoT, Google Home for applications like smart home.
- Offline Processing: Deploying lightweight AI models on the ESP32 to enable offline responses.

REFERENCES

[1] K. Gowtham, V. Deepak, N. Harshitha, S. Aditya, H. D. N. Rao, and V. Venkatesh, “Personal AI Companion Voice Assistant,” Indonesian Journal of Computer Science, vol. 14, no. 2, Apr. 2024.

[2] I. Shazhaev, A. Tularov, D. Mikhaylov, I. Shazhaev, A. Shafeeg, “Voice Assistant Integrated with ChatGPT” Indonesian Journal of Computer Science, vol. 12, no. 1, Feb. 2023.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 6, June 2025

|DOI: 10.15680/IJIRSET.2025.1406074|

[3] Y. M. Mohialden, K. Raisian, P. Singh, K. Joshi, "Enhancement of ChatGPT using API Wrappers Techniques" Al-Mustansiriyah Journal of Science, vol. 34, no. 2, Jun. 2023.
[4] Rajakumar P., K. Suresh, Boobalan. M., M. Gokul, G. D. Kumar, Archana R., "IoT Based Voice Assistant using Raspberry Pi and Natural Language Processing", International Conference on Power, Energy, Control and Transmission Systems, Dec.2022
[5] Khairunisa Sharif, Bastian Tenbergen, "Smart Home Voice Assistants: Survey of User Privacy and Security Vulnerabilities" Complex Systems Informatics and Modeling Quarterly, Article 139, Issue 24, Oct. 2020.
[6] Jingjin Li, A. C. Chen, A. L. Pan, A. M. R. Azghadi, A. H. Ghodosi, "Security and Privacy Problems in Voice Assistant Applications: A Survey" CSCR, vol. 134, Nov. 2023.
[7] OpenAI. (2023, June 10). GPT-4: The future of AI language models. OpenAI. https://openai.com/gpt-4
[8] Techiesms "Portable AI Assistant" YouTube. http://youtube.com/watch?v=zvR9DTfMwPE
[9] Google. (2023, November 1). Introducing Gemini: The next evolution of AI. Google. https://blog.google/technology/ai/introducing-gemini-next-evolution-ai/
[10] schreibfaul1, ESP32-audioI2S: A library for MAX98357 and other I2S DACs, GitHub. [Online]. Available: https://github.com/schreibfaul1/ESP32-audioI2S.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details