International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 12 Issue: 11 | Nov 2025 www.irjet.net p-ISSN: 2395-0072

HOTPIN — WEARABLE AI ASSISTANT

Keshav Patil[1], Vighnesh Nilajakar[2], Sanket Jatrate[3], Vinayak Patil[4], Darshan Kalkuppi[5]

[1]*Assistant Professor, Maratha Mandal's Engineering College, Belagavi, Karnataka, India*
[2,3,4,5] *Student, Maratha Mandal's Engineering College, Belagavi, Karnataka, India*

Abstract – *This paper presents the development of HOTPIN, a compact wearable assistant designed to offer real-time voice interaction and optional visual analysis using a lightweight hardware platform. The system uses an ESP32-based controller connected to an I²S microphone and digital audio amplifier to process speech requests, while a low-power camera module can be activated only when visual input is needed. User audio is captured, converted to text, and transmitted over Wi-Fi to an edge server that communicates with a locally hosted large-language-model (LLM). The backend processes both text and image inputs and returns a contextual response, which is then converted to audio on the wearable device. This on-demand multimodal approach ensures reduced power consumption while maintaining responsiveness. Experimental evaluation shows efficient performance, with a median latency of around 300ms for voice-only requests and approximately 800ms for combined voice-and-image interactions. Power measurements indicate average current draws of 280mA and 450mA, enabling several hours of operation from a 5V, 2000mAh Li-ion battery. These results highlight the feasibility of integrating local AI processing with a wearable form factor, providing an energy-aware and flexible platform for personal ambient intelligence.*

Key Words: **Wearable AI, ESP32-CAM, Multimodal Assistant, Edge AI, LLM, I²S Audio.**

1.INTRODUCTION

Wearable AI devices are evolving rapidly as users increasingly demand hands-free interaction, personalization, and real-time assistance without relying on cloud connectivity. To address these requirements, the HOTPIN wearable assistant combines speech recognition, contextual processing, and selective image capture through a compact embedded platform and a local LLM-powered backend. The objective of the system is to provide an efficient multimodal assistant capable of performing tasks such as question answering, environmental understanding, and interactive guidance, while keeping power consumption and latency low.

The ESP32 microcontroller serves as the central controller, handling digital audio through an I²S microphone and speaker amplifier. A camera module remains disabled during idle operation and is activated only when the user triggers a visual query, ensuring minimal standby consumption. Speech data is sent to a local REST API that passes the request to an LLM, enabling fast and private inference. The system's architecture reflects the growing trend of shifting computation from cloud-based services to nearby edge devices, increasing responsiveness and privacy. The prototype demonstrates the practicality of integrating voice and vision processing within a small, battery-powered wearable assistant.

1.1 Technology Assessment

The HOTPIN prototype demonstrates emerging trends in embedded AI, where multimodal processing is enabled using a combination of microcontrollers and edge-hosted language models. The use of I²S audio modules allows precise digital capture and playback, while the selective activation of the camera provides visual context only when necessary.

However, the limitations of the ESP32 platform must be considered, including constrained memory, limited parallel peripheral usage, and occasional conflicts between camera and audio drivers. These challenges indicate that while ESP32-based designs are suitable for early-stage wearables, achieving seamless multimodal performance may require more advanced microcontrollers or specialized AI-focused hardware in future iterations.

1.2 Cost–Benefit Analysis

The overall hardware cost of the HOTPIN wearable remains relatively low because it is built using widely available microcontroller components. Despite its modest cost, the system delivers strong functional benefits: real-time voice assistance, optional image-based interaction, reduced user effort, and improved energy efficiency. The on-demand activation of the camera avoids unnecessary power consumption, extending battery life. Although additional work is required to refine monetization and production strategies, the system demonstrates a high value-to-cost ratio for personal and experimental use cases.

2. SYSTEM ARCHITECTURE

The HOTPIN assistant follows a layered design consisting of the Wearable Device Layer, Edge/API Layer, and LLM Backend Layer.

At the wearable layer, an ESP32 module interfaces with an INMP441 I²S microphone for audio capture and a MAX98357A I²S amplifier for audio output. A camera module remains powered off during idle periods and is enabled only

when the user requests an image-based interaction. This selective activation significantly reduces idle power usage.

The wearable connects to the API layer through Wi-Fi or a mobile hotspot. The API server receives JSON packets containing the transcribed audio and, when applicable, an image file. The server maintains conversation context and forwards the request to the LLM backend, which processes multimodal input and returns structured responses. These responses are relayed back to the ESP32 for audio playback.

To reduce hardware conflicts and memory strain, I^2S and camera drivers are initialized dynamically based on operating mode. Additional measures include pruning session history to prevent oversized payloads and optimizing power states such as Wi-Fi sleep modes. This architecture effectively balances performance, modularity, and energy efficiency.

2.1 Operational Research

Performance evaluation was conducted under two primary use cases:

1. Voice-only mode, and
2. Voice-plus-image mode.

During testing, event timestamps such as microphone activation, HTTP transmission, response reception, and playback start were recorded. Current draw was monitored using inline power measurement tools.

Results show a median latency of around 300ms for voice-only interactions, which increases to approximately 800ms when the camera is activated due to additional initialization and upload overhead. Power consumption averaged 280 mA in voice-only mode and 450mA during image-assisted queries. On a 2000mAh battery, the system provided around 5 hours of voice-only operation and 3–4 hours under mixed usage.

A failure rate of nearly 10% was observed in camera initialization during rapid switching between audio and image modes, indicating peripheral limitations of the ESP32 platform. A small group of test users found the system intuitive and responsive but suggested improvements to audio volume and long-term conversation memory.

2.2 Firmware

The firmware coordinates event-driven operation for voice requests, camera capture, Wi-Fi communication, and response playback. When the user presses the input button, the microphone begins recording a short voice clip, which is encoded and forwarded to the API server. For image-enabled requests, the firmware activates the camera, captures a frame, and uploads it before sending the corresponding query.

The firmware also logs timestamps and current measurements to support performance evaluation. It manages memory allocation carefully to avoid buffer overflows and alternates between audio and camera modes by dynamically initializing and de-initializing drivers. The system demonstrates consistent performance under typical loads, although occasional camera initialization failures highlight the need for more robust peripheral handling in future designs.

3. METHODOLOGY

The methodology used in this project covers the hardware setup, firmware development, API communication, and testing procedures. Hardware integration involved connecting the ESP32 to the I^2S microphone, I^2S amplifier, and camera module. The firmware was programmed to handle button-activated events, audio capture sequences, image capture routines, and JSON data transmission.

At the software backend, a REST API was implemented to receive requests, maintain conversation state, and forward user inputs to the LLM. The wearable receives text responses, converts them to audio, and plays them through the I^2S amplifier. Testing involved measuring latency, analyzing power usage under different modes, and gathering user feedback to assess system reliability and functionality.

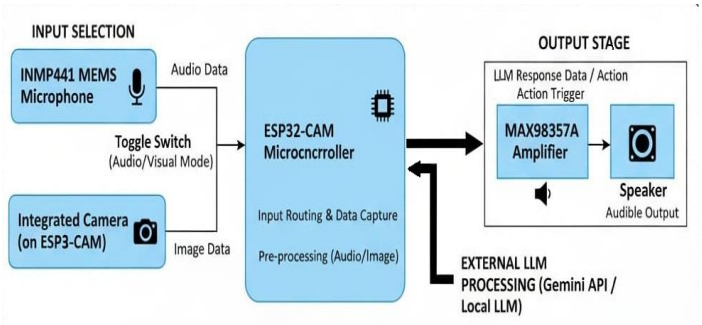

Fig - 3.1: Block Diagram

3.2 Circuit Diagram

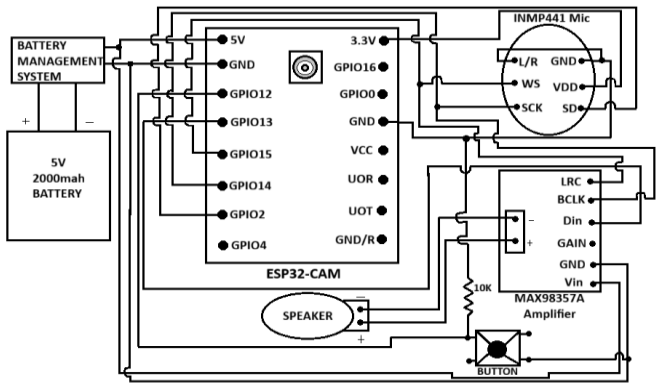

Fig - 3.2: Circuit Diagram

4. RESULTS AND DISCUSSION

The HOTPIN system displayed strong performance in real-world testing scenarios. In voice-only mode, the average response time was approximately 310ms, providing a near-instantaneous user experience. When the camera was activated, overall latency increased to around 840ms, primarily because of camera initialization and image transfer.

Power consumption trends aligned with expected behavior, with 285 mA drawn during voice-only usage and 460 mA during combined operations. With a 2000mAh battery, the estimated runtime ranged from 3 to 5 hours, depending on the balance between voice and camera usage.

A 9% failure rate was observed during rapid switching between camera and audio modes, underscoring the ESP32's peripheral limitations. User feedback from a small pilot group indicated satisfaction with the system's responsiveness and ease of use, though improvements such as louder audio output and deeper memory retention were suggested.

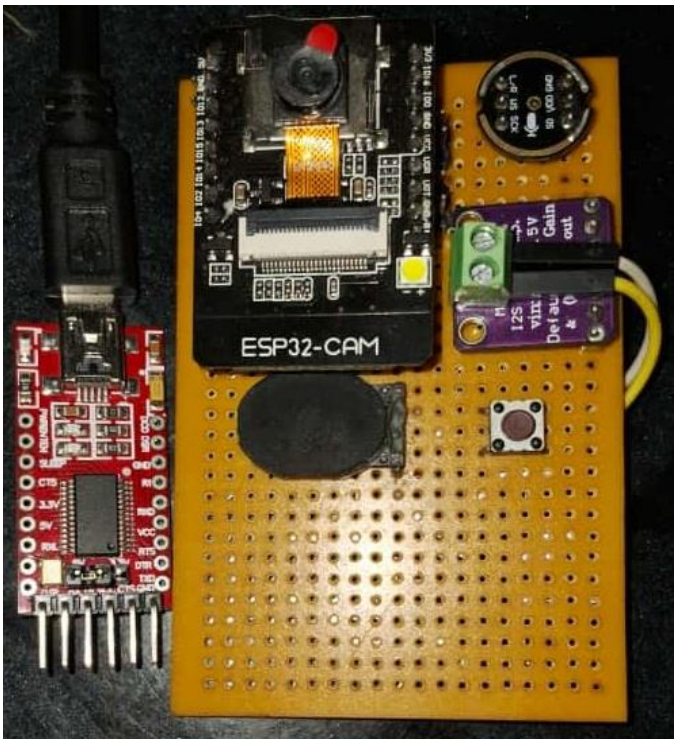

Fig - 4.1: Final Model

5. CONCLUSIONS

This project demonstrates that a wearable assistant combining voice processing and optional image capture can be effectively implemented using low-power microcontroller hardware and an LLM-based backend. By activating high-power peripherals such as the camera only when required, the system maintains efficient energy usage while still supporting multimodal interaction. Experimental results show competitive latency and reasonable operating time on a compact Li-ion battery.

Some hardware limitations—including occasional peripheral initialization failures—highlight opportunities for improvement, such as migrating to an upgraded ESP32 variant or employing a dual-MCU design. Overall, the HOTPIN wearable provides a practical blueprint for future edge-AI devices designed for personal assistance and context-aware interaction.

REFERENCES

[1] Mathew, A., "Humane Ai Pin - InnoGlove AI Embrace," International Journal for Multidisciplinary Research (IJFMR), Vol. 5, Issue 6, pp. 1-5, 2023.

[2] Imran Hussain S,Mehul M S,Mohana Sankaran S., "Design and Implementation of a Smart Speech Interface using OCR and TTS APIs on ESP32-CAM," In Proceedings of the International Conference on Trends in Material Science and Inventive Materials (ICTMIM2025), pp. 319-324, IEEE, 2025.

[3] Bibhakar Das,Kalyan Kumar Halder., "Face Recognition Using ESP32-Cam for Real-Time Tracking and Monitoring," In 2024 International Conference on Advances in Computing, Communication, Electrical, and Smart Systems (iCACCESS), IEEE, 2024.

[4] Umberto Cappellazzo, Minsu Kim, Honglie Chen, Pingchuan Ma, Stavros Petridis, Daniele Falavigna, Alessio Brutti, and Maja Pantic, "Large Language Models Are Strong Audio Visual Speech Recognition Learners," arXiv preprint arXiv:2409.12319, 2024.