

Boston University
Electrical & Computer Engineering
EC463 Senior Design Project

First Prototype Testing Plan



By

Team 3
EchoView.AI

Team Members

Riya Deokar riyaa@bu.edu
Nicholas Hardy nckhrdy@bu.edu
Marybel Boujaoude marybelb@bu.edu
Hassan Hijazi hhijazi@bu.edu
Jazmyn Walker jwalker@bu.edu

Required Materials

Hardware:

- Teensy 4.1 Microcontroller
 - Power Supply: 3.3V to 5V
- ESP32-C3 Module
 - Power Supply: 3.0V to 3.6V
- Microphone
- OLED Display

Software:

- Xcode
 - Swift Programming Language
- Espressif IoT Development Framework (ESP-IDF)
 - Wifi, UDP, and BLE libraries
- Python 3
- Arduino IDE
 - Teensyduino

Set Up

Our testing setup involves a structured plan focusing on both hardware integration and software development. The hardware module consists of several components: the Teensy 4.1 microcontroller, which will receive and process audio signals from the connected microphone, the ESP32-C3, which is necessary for its Wifi and Bluetooth capabilities, and an OLED display, which will show the converted speech-to-text. Our microphone will be connected to the Teensy 4.1 and use code to capture audio data from the microphone using the Teensy's ADC (Analog to Digital Converter). Next, we will establish a serial connection using UART protocol and attempt to send a message from the Teensy to the ESP32-C3. After that, we will set up Wifi on the ESP32-C3 by using the protocols/socket/udp_sever library available on the Espressif IoT Development Framework Guide. UDP will be best for our design because of its low-latency which is important for real-time audio data transmission. There is similarly a bluetooth/bluedroid library available to test. We will also wire the OLED display to the Teensy 4.1 which will receive text data from the ESP32-C3 and display it on the screen. Regarding the software aspect, we will be using Xcode and the Swift programming language to develop an iOS mobile application, tailored for user interaction and data visualization. After initial testing, we will focus on combining both aspects of the hardware and software pieces to seamlessly work together and create a cohesive system.

Pre-testing Setup Procedure:

ESP32-C3 side:

1. Install ESP-IDF and Python 3 using HomeBrew
2. Set up the tools used by ESP-IDF (compiler, debugger, Python packages, etc.)
3. Set up the environment variables
 - a. Run in terminal: `. $HOME/esp/esp-idf/export.sh`

Teensy 4.1 side:

1. Download Arduino IDE
 - a. Add on the Teensyduino library

Xcode side:

1. Download the Xcode app

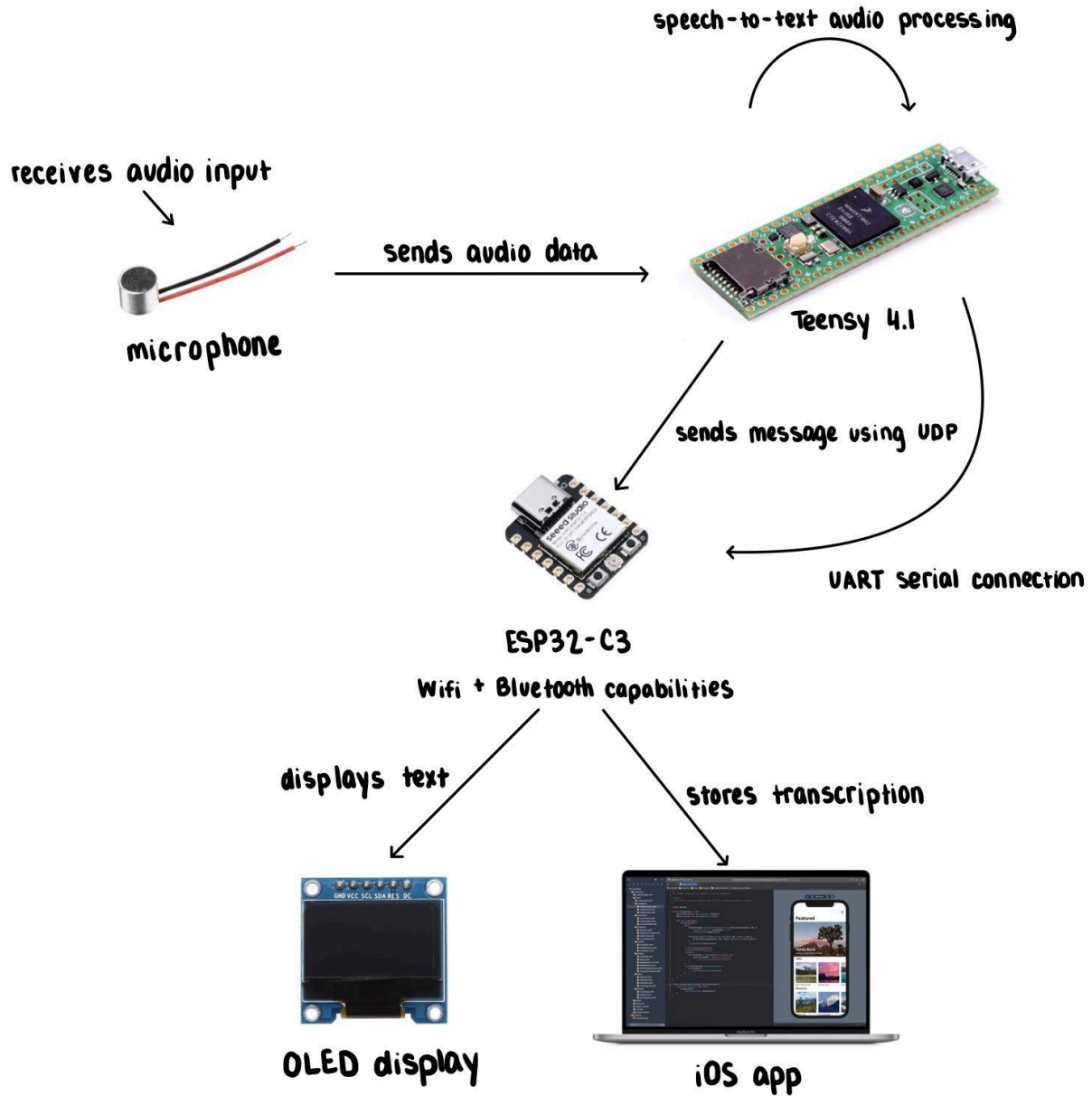


Figure 1: Illustration of Setup and Process Flow

Testing Procedure:

1. Test each hardware component separately to ensure they are functioning correctly (ex: microphone input, OLED display output)
2. Verify the serial communication between Teensy 4.1 and ESP32-C3
3. Check the audio capture and processing capabilities of the Teensy 4.1
4. Test the iOS app for functionality

Measurable Criteria

The criteria for successful running and output is as follows:

- I. The OLED should display text that is easily readable
- II. The audio captured by the microphone is accurate and clear
- III. There is minimal delay from when the audio is captured to when it is received by the ESP32-C3
- IV. The Teensy 4.1 is able to send messages to the ESP32-C3 through UDP and display it on the console