ECE 445

Spring 2025

Senior Design Final Report

AI-based Meeting Transcription Device

Team 45

Ziyang Huang, Gao Gao, Chang Liu

TA: Jiankun Yang

Abstract

After the pandemic, Zoom meetings became crucial to our daily study and work. The impressive functionality of live transcription caught our attention: no matter how noisy the environment is, the accurate transcribed subtitle can sufficiently help us to capture key points from the presenter. However, in an in-person lecture or meeting, there is no such kind of individual device that can do live transcription for us, or we can use some software that provides similar features but requires internet connection and deployment on electronic devices. To tackle this problem, we developed an individual, offline, easy-to-use, and portable AI-based meeting transcription device, which allows us to obtain live transcription anywhere.

Contents

1. Introduction

1.1 Problem and solution

1.1.1 Problem

In many academic, professional, and social settings, clear and accurate communication is essential. However, individuals who rely on speech-to-text technology—such as students taking lecture notes, professionals transcribing meetings, and those with hearing impairments—often face challenges when real-time transcription is not readily available. Current solutions for live transcription, such as Zoom's automatic captions or mobile speech-to-text applications, are largely platform-dependent and require a stable internet connection. This limits their effectiveness in offline environments, such as in-person meetings, classrooms, and research discussions, where accurate note-taking is crucial.

For individuals with hearing impairments, the lack of reliable transcription tools creates significant accessibility barriers. Many venues, including classrooms and workplaces, do not provide real-time captioning, leaving individuals dependent on third-party applications that may not be readily available. Furthermore, for professionals and students who need detailed, distraction-free transcripts, mobile-based solutions can be unreliable due to background noise, latency, or limited processing power.

While commercial AI-powered transcription services exist, they often require cloud connectivity and recurring subscription costs, making them unsuitable for offline or private use cases. Additionally, reliance on cloud services raises data privacy concerns, as sensitive conversations and confidential meetings may be processed externally. Given these challenges, there is a need for an independent, portable transcription device that can function reliably without an internet connection while offering real-time, high-accuracy transcription in a variety of settings.

1.1.2 Solution

To address these challenges, we propose the AI-based Meeting Transcription Device, a standalone, portable system that captures, transcribes, and displays spoken text in real time without requiring an internet connection. Unlike existing solutions that rely on cloud-based APIs, our device will leverage on-device AI processing to ensure low-latency transcription, enhanced privacy, and greater accessibility.

Our system consists of the following key components:

- A microphone module to capture audio input from the speaker.
- A speech processing unit (Raspberry Pi 5) running the VOSK speech-to-text model, which converts spoken words into text.
- An STM32 microcontroller, which serves as the central controller, handling user interactions, formatting text output, and managing storage operations.
- An LCD screen to display real-time transcriptions, ensuring the user can follow the conversation seamlessly.
- A power system (battery with efficient power management) to provide a portable and reliable power source for extended operation.

By combining real-time speech-to-text processing with a standalone embedded system, our device offers a versatile, privacy-conscious, and highly accessible solution for professionals, students, and individuals with hearing impairments. This system will provide a reliable alternative to cloud-based transcription services, allowing users to operate in any environment without connectivity constraints while maintaining full control over their data

1.2 Visual Aid

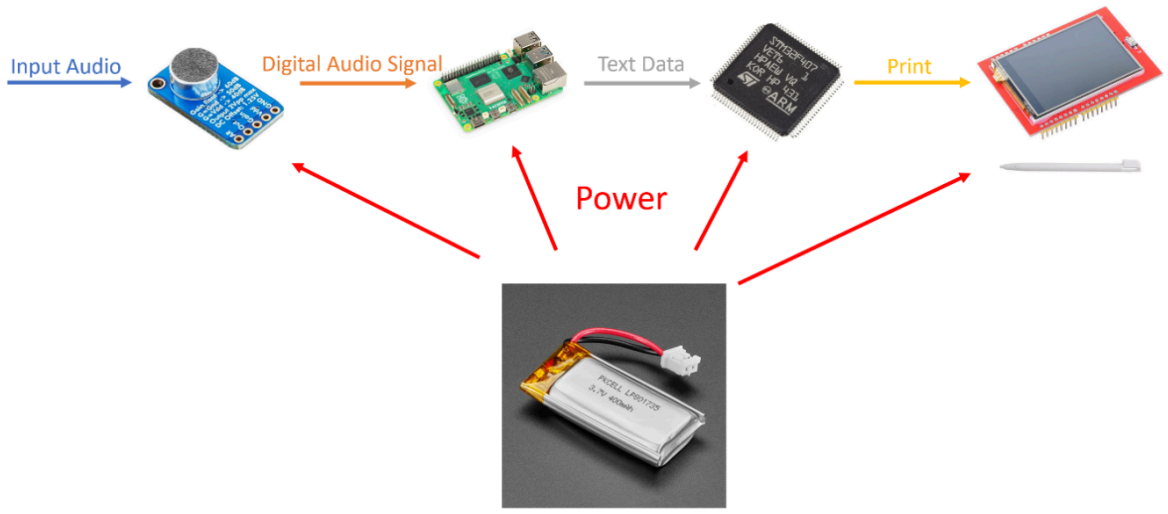

Figure 1: Visual aid for the AI-based Meeting Transcription Device.

1.3 High-level Requirements List

- The device must operate for at least 3 hours on a full charge under normal usage conditions, ensuring uninterrupted transcription throughout an average-length meeting or lecture.
- The speech processing unit must transcribe spoken sentences with a maximum delay of 2 seconds, ensuring that the device processes and displays each sentence before the next is spoken.
- The transcribed text must be displayed clearly and accurately on the LCD screen, with no more than 5% error rate in normal speaking conditions.
- The system must store at least 50 transcribed sentences in local memory (SD card), ensuring users can review and retrieve past transcriptions when needed.

2. Design

2.1 Block Diagram

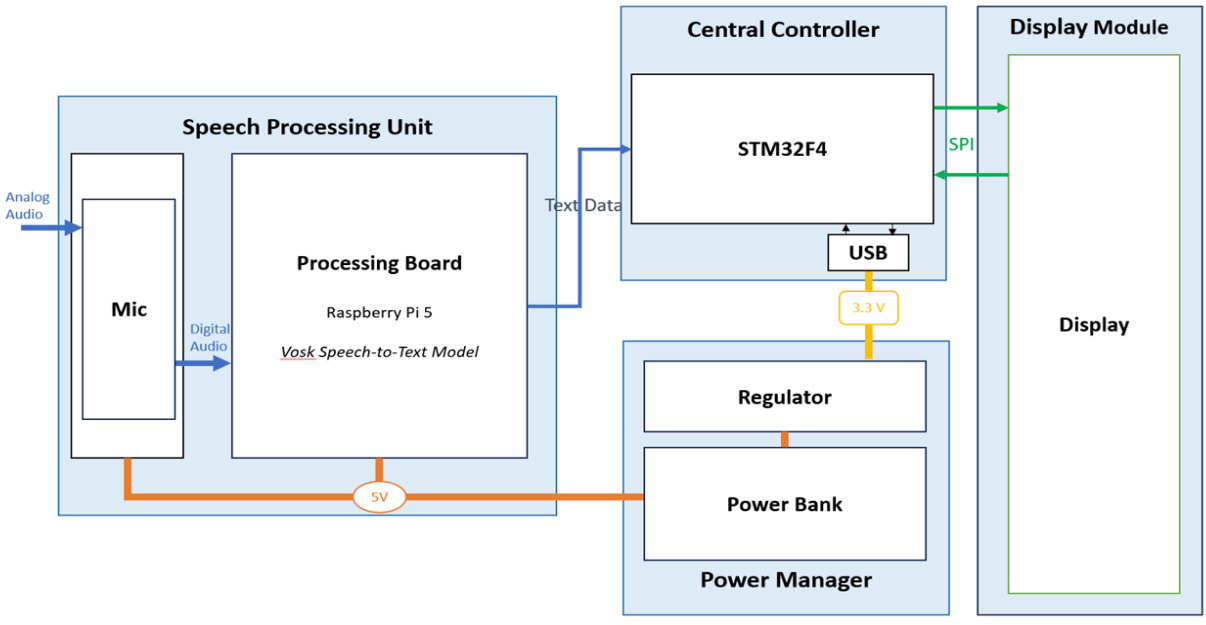

Figure 2: Block Diagram of AI-based Meeting Transcription Device.

2.2 Subsystem Overview

2.2.1 Speech Processing Unit

The speech processing unit is responsible for the compute-intensive workload of the transcription system. It consists of a Raspberry Pi 5, a microphone module, and an SD card for storage. The Raspberry Pi 5 is used to run the VOSK speech-to-text Model from Alpha Cephei, which processes the audio input and converts it into text.

Function in the System: This unit receives real-time audio from the microphone, applies noise reduction processing, and runs the VOSK model to produce textual output. The processed text is then sent to the Central Controller via UART communication.

Justification: The Raspberry Pi 5 was chosen due to its quad-core processor and sufficient RAM to efficiently run the AI model while keeping power consumption reasonable.

Interfaces:

Microphone Module → Raspberry Pi 5 (Audio Input)

Raspberry Pi 5 → STM32 (UART Communication for Text Data Transmission)

2.2.2 Central Controller

The Central Controller acts as the main processing unit responsible for managing data flow, user interactions, and communication between other components. It consists of the STM32F4 microcontroller and flash memory.

Function in the System: The STM32F4 receives transcribed text data from the speech processing unit, formats it, and sends it to the Display Module for visualization. It also manages storage operations for saving recent transcriptions in flash memory.

Justification: The STM32F4 was selected due to its high-speed SPI communication, low power consumption, and adequate flash storage options.

Interfaces:

STM32 ↔ Speech Processing Unit (UART for receiving transcribed text)

STM32 ↔ Display Module (SPI for text rendering)

2.2.3 Display Module

The Display Module is responsible for showing real-time transcriptions to the user. It consists of a 2.8-inch TFT LCD screen (ILI9341), which communicates with the STM32 microcontroller via SPI.

Function in the System: Displays real-time text output from the central controller.

Justification: The ILI9341 LCD was chosen for its high refresh rate, SPI compatibility, and low power consumption.

Interfaces:

STM32 ↔ LCD (SPI Communication for text rendering)

2.2.4 Power Manager

The Power Manager ensures a stable power supply for all components. It consists of a power bank, a 3.3V regulator, and a USB-C port.

Function in the System: Provides regulated power to all subsystems.

Justification: The power bank offers high energy density and rechargeability, making it ideal for portable applications.

Interfaces:

Power bank → Voltage Regulator → STM32 (3.3V)

Power bank → Speech Processing Unit (5V)

2.3 Subsystem Requirements & Verification

Subsystem	Requirements	Verification
Speech Processing Unit	Each sentence must be displayed within 3 seconds after finishing speaking.	Use a stopwatch to record the time gap between speaking and displaying.
	The transcribed accuracy has to be at least 80%.	Speak poems to the device. Calculate the error rate.
Central Controller	The SPI communication speed should be fast enough to avoid noticeable delay.	Compare the sentence displayed time on the screen to the time in the Linux terminal.
Display Module	The displayed text must be readable.	Read a sentence to the device and check if it displays all characters reasonably.
Power Manager	The battery must support ≥3 hours of operation on a full charge.	Fully charge the device, run continuous transcription and measure uptime.

Table 1: Requirements and verifications.

2.4 Tolerance Analysis

2.4.1 Speech Processing Unit

- According to Alpha Cephei, the VOSK can be run on the Raspberry Pi, so we need not worry too much about the board's performance.
- Assume the lecturer speaks 4 words per second (the normal speaking speed should be 2 – 3 words per second). Each letter consumes 7 bits of resource during transmission, and we assume each word has 8 letters on average (which should normally be 6). Hence, the transmission speed requirement is:

$$Speed = 4\ \frac{word}{s} \cdot 8\ \frac{letter}{word} \cdot 7\ \frac{bit}{letter} = 224\ b/s$$

The speed is relatively low compared to the peak transmission capacity of any ports on the board. Therefore, we need not worry too much about it.

- Based on the official data sheet, the typical bare-board active current consumption of Raspberry Pi 5 is 800mA. Hence, the energy capacity of a battery that can support the device to work for 3 hours:

$$E = It = 800mA \cdot 3hr = 2400mAh$$

2.4.2 Central Controller

- According to the official datasheet of STM32F4, the run mode current is about 45 mA. Therefore, the battery capacity required to let the chip run for 3 hours is:

$$E = It = 45mA \cdot 3hr = 135mAh$$

- Since there is no further change of text data in the central controller, the estimated communication rate required for a regular meeting is also 224 b/s, significantly lower than the max data rate of SPI of 4 Mb/s. Therefore, there should be no issue during the SPI communication between the controller and the display module.
- Assume a lecturer delivers 50 sentences in 3 minutes, and each sentence has 10 words. Based on the mentioned assumption in the previous subsystem, we can estimate the storage capacity required for the flash memory as follows:

$$Storage = 50\ sentence \cdot 10\ \frac{word}{sentence} \cdot 8 \frac{letter}{word} \cdot 7\ \frac{bit}{letter} = 2800\ bits$$

A standard memory capacity of MCU is 512kb; therefore, the time required to fulfill the flash memory is as follows:

$$Time = 512\ kb \div 2800\ b \cdot 3 = 561\ min$$

The result is large enough to save the entire meeting's transcription in the flash memory. Therefore, we should not worry that the device cannot save the latest 50 sentences in the actual situation.

2.4.3 Display Module

- Assume the average current consumption of the screen is about 5 mA (standard current consumption should be around 3 mA). Therefore, the battery capacity required to support it to run 3 hours is:

$$E = It = 3mA \cdot 3hr = 9mAh$$

2.4.4 Power Manager

- According to previous calculations, the estimated battery capacity required to power the entire system for 3 hours is:

$$E = 9 + 135 + 2400 = 2544mAh$$

  To ensure everything runs safely, the estimated battery capacity should be at least 3000 mAh.
- The Raspberry Pi 5 requires a 5A current to ensure its peak performance. The normal boost converter can not bear such a high current. Though there are some QFN packed options, considering its soldering difficulty, excessive battery requirement, and expensive cost, we finally choose a power bank solution. The power bank provides 5V and up to 4.5A to the entire system. By using a regulator to reduce input voltage to 3.3V, the power bank solution can reliably power our systems.

2.5 Cost Analysis

Component	Quantity	Unit Cost ($)	Total Cost ($)
Raspberry Pi 5	1	120	120
STM32F411RET6	1	5.35	5.35
Microphone Module	1	0.67	0.67
LCD	1	16.9	16.9
SD Card	1	21.99	21.99
Power Bank	1	67.99	67.99
LDV1117-33	1	0.64	0.64
INA128UA/2K5	1	8.17	8.17
ECS-80-20-4	1	0.71	0.71
ECS-.327-12.5-34B-TR	1	0.41	0.41
PCM1808PWR	1	1.47	1.47
1727-3825-1-ND	8	0.186	1.488
497-11882-1-ND	1	0.298	0.298
CAP 4.4UF	4	0.012	0.048
CAP 10UF	30	0.078	2.34
CAP 0.1UF	30	0.006	0.18
CAP 1UF	30	0.011	0.33
HEADER 2.54MM	30	0.124	3.72
RES 2.2K	30	0.011	0.33
RES 10K	30	0.025	0.75
RES 1K	30	0.012	0.36
RES 5.1K	2	0.008	0.016
CT3088CT-ND	2	0.181	0.362
USB-C	1	0.78	0.78
Larbor	3	5000 (40*50*2.5)	15000
Total			15256

Table 2: Cost summary.

3. Ethics and Safety

3.1 Ethics

Our AI-based Meeting Transcription Device raises several ethical considerations, especially related to the collection and use of audio data. In keeping with the principles outlined by the IEEE Code of Ethics and the ACM Code of Ethics, we identify key issues and approaches to preventing unethical outcomes:

3.1.1 Privacy and Consent (ACM 1.6 & IEEE I.1)

Because the device is designed to capture and transcribe spoken communications, there is a risk that conversations could be recorded without the knowledge or permission of participants. To address this, we will require user interaction to inform meeting participants that transcription is active.

3.1.2 Data Security and Confidentiality (ACM 1.6 & 1.7)

The device could be misused for unauthorized surveillance or eavesdropping. As highlighted by professional codes of ethics, engineers have a responsibility to anticipate how a product might be misused and take steps to limit harmful outcomes. We will use design features (e.g., audible alert whenever the recording is active) to discourage covert use.

3.2 Safety

Safety concerns for our device primarily revolve around two areas: (1) electrical and mechanical safety of the hardware (notably the lithium-ion battery and its charging circuitry), and (2) compliance with relevant regulations governing electronic devices.

3.2.1 Battery and Electrical Safety

The device relies on a lithium-ion battery and a boost converter to power the Raspberry Pi 5 and other components. Lithium-ion batteries can pose fire or explosion risks if they are damaged, improperly charged, or short-circuited.

3.2.2 Safe Handling and Disposal

Users will be instructed on safe handling of the device, including proper storage, transport, and disposal of the lithium-ion battery according to environmental regulations.

3.2.3 Regulatory Compliance

Even though our device primarily functions as a stand-alone transcription, we must still review relevant federal regulations (e.g., FCC Part 15 in the United States) for any emissions from the microcontroller or other components.

4. Verification and Result

4.1 Transcription Speed:

In the verification of the processing speed of our speech processing unit, we used a stopwatch to record the response time of our device after reading multiple sentences. Then, we calculate the average response time and compare it to the requirement. The result is around 1.67 seconds, which is lower than our original expectation which is 3 seconds.

# Trial	Delta Time (s)
1	1.61
2	1.58
3	1.66
4	1.61
5	1.90
Average	1.672

Table 3: Delay verification.

4.2 Accuracy and Readability:

This part of the test tested both the accuracy (speech processing unit) and the readability (screen). In this verification part, we read a famous poem Dreams, and see how much it correctly captures the result. We checked whether the screen driver correctly displays all the characters in a word. As a result, in the picture below, it correctly displayed all the English Characters.

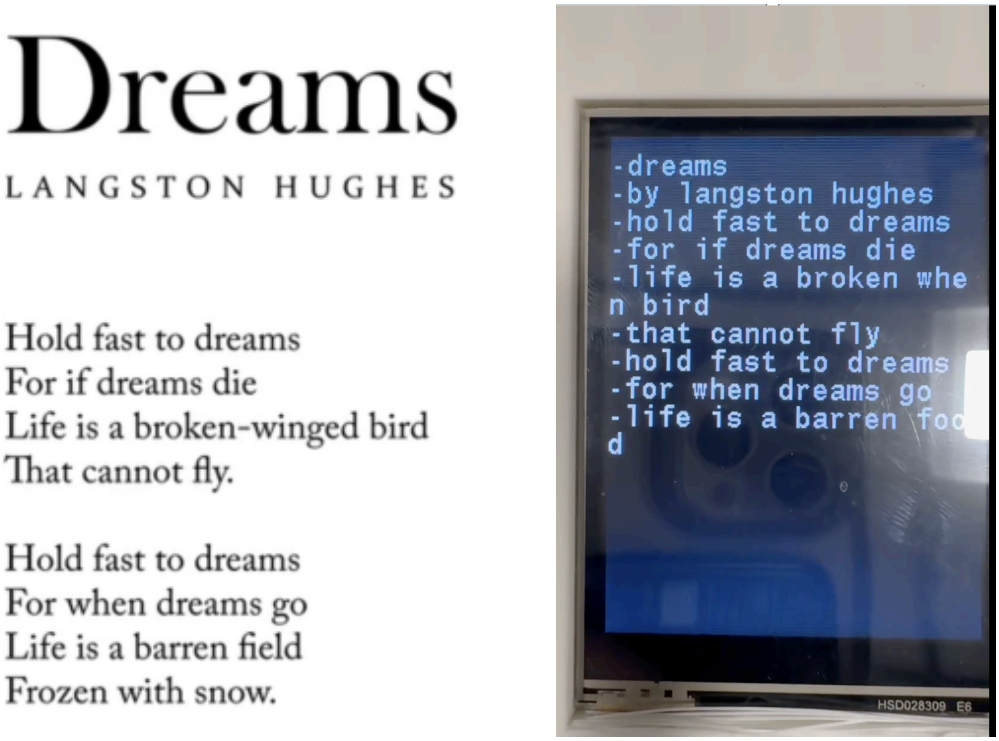

Figure 3: “Dreams” and result.

From the result above, you can see that there are still some words that are falsely transcribed. Then, we record another poem, Fire and Ice, to calculate the accuracy.

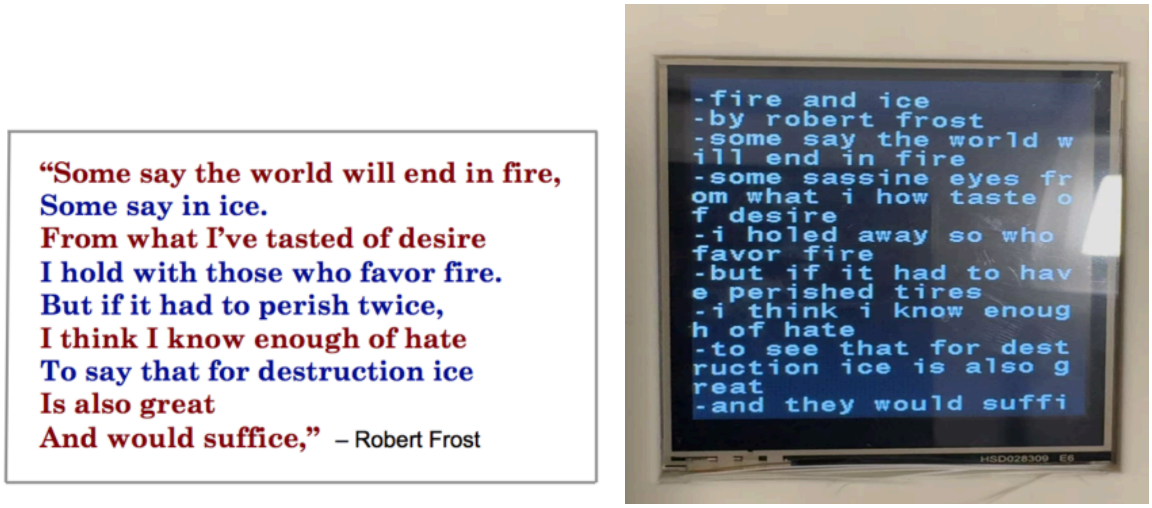

Figure 4: "Fire and Ice" and result.

Here is the result of the other attempts we made. These poems have 93 characters in total and 12 falsely transcribed characters. From the above result, we calculated the error rate with the following function:

$$Error\ Rate\ =\ 12\ /\ 93{\cdot}100\%\ =\ 13\%$$

The result is 13%, which is lower than our requirement of 20% (80% accuracy).

5. Conclusion and Further Resources

5.1 Success & Challenges

During implementation, we encountered four main challenges. First, the Raspberry Pi's 5 A power requirement proved difficult to meet; although not ideal, we ultimately used a portable power bank to satisfy this demand. Second, our initial UART setup suffered from data loss—low communication rates and long wait times caused character loss—so we switched to an interrupt driven UART scheme to achieve lossless communication. Third, setting up the Raspberry Pi environment and launching the application was a problem. In the beginning, the Raspberry Pi had to be connected with a terminal in the computer, so we wrote a shell script to automate environment loading and program startup with a single command. Finally, overlapping characters in the display layout impaired readability; by modifying and optimizing the display driver, we eliminated overlap and restored clear, properly formatted text.

5.2 Conclusion

In this project, we set out to build a fully standalone, battery powered speech to text device by combining a Raspberry Pi 5–based inference engine, an STM32 microcontroller, an ILI9341 display and custom power management PCB—and in doing so we not only proved the feasibility of on device AI transcription but also gained hands on mastery of several key disciplines: deploying neural networks at the edge, designing robust UART and I²S based communication protocols, laying out high efficiency power delivery circuit in Altium, and writing clean, real time software in both Python and C. Along the way we tackled challenges in delivering a stable 5 A rail to the Pi, optimizing our LCD update routines to minimize latency, and verifying end to end audio capture and display synchronization under varying battery loads. Looking forward, our next steps are to integrate the power management components directly onto our own PCB footprint (eliminating bulky breakout modules), to evaluate alternative compute platforms (for example NVIDIA Jetson Nano or an FPGA soft core) for improved performance and lower power draw, to collect and self train a small speech dataset for a bespoke, on device model tuned to classroom acoustics, and to pursue further miniaturization—ultimately creating an even more compact, reliable, and portable device that can deliver high quality live captions in any setting, without reliance on cloud services.

6. References

6.1 Safety & Ethics Documentation

1. IEEE Code of Ethics, IEEE, [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html.

2. ACM Code of Ethics and Professional Conduct, ACM, [Online]. Available: https://www.acm.org/code-of-ethics.

6.2 Hardware & Software Documentation

3. Raspberry Pi Documentation, Raspberry Pi Foundation, [Online]. Available: https://github.com/raspberrypi/documentation/blob/develop/documentation/asciidoc/computers/raspberry-pi/power-supplies.adoc.

4. STM32F4 Power Modes & Efficiency, STMicroelectronics, Application Note AN4365, [Online]. Available: https://www.st.com/resource/en/application_note/an4365-using-stm32f4-mcu-power-modes-with-best-dynamic-efficiency-stmicroelectronics.pdf.

5. ILI9341 Display Controller Datasheet, Adafruit, [Online]. Available: https://cdn-shop.adafruit.com/datasheets/ILI9341.pdf.

6. VOSK Speech Recognition Toolkit, AlphaCephei, [Online]. Available: https://alphacephei.com/vosk/.

6.3 Literary References

7. *Dreams*, Langston Hughes, Poetry Foundation, [Online]. Available: https://www.poetryfoundation.org/poems/150995/dreams-5d767850da976.

8. *Fire and Ice*, Robert Frost, Poetry Foundation, [Online]. Available: https://www.poetryfoundation.org/poems/44263/fire-and-ice.

7. Appendix

A. Schematic Version 1.0

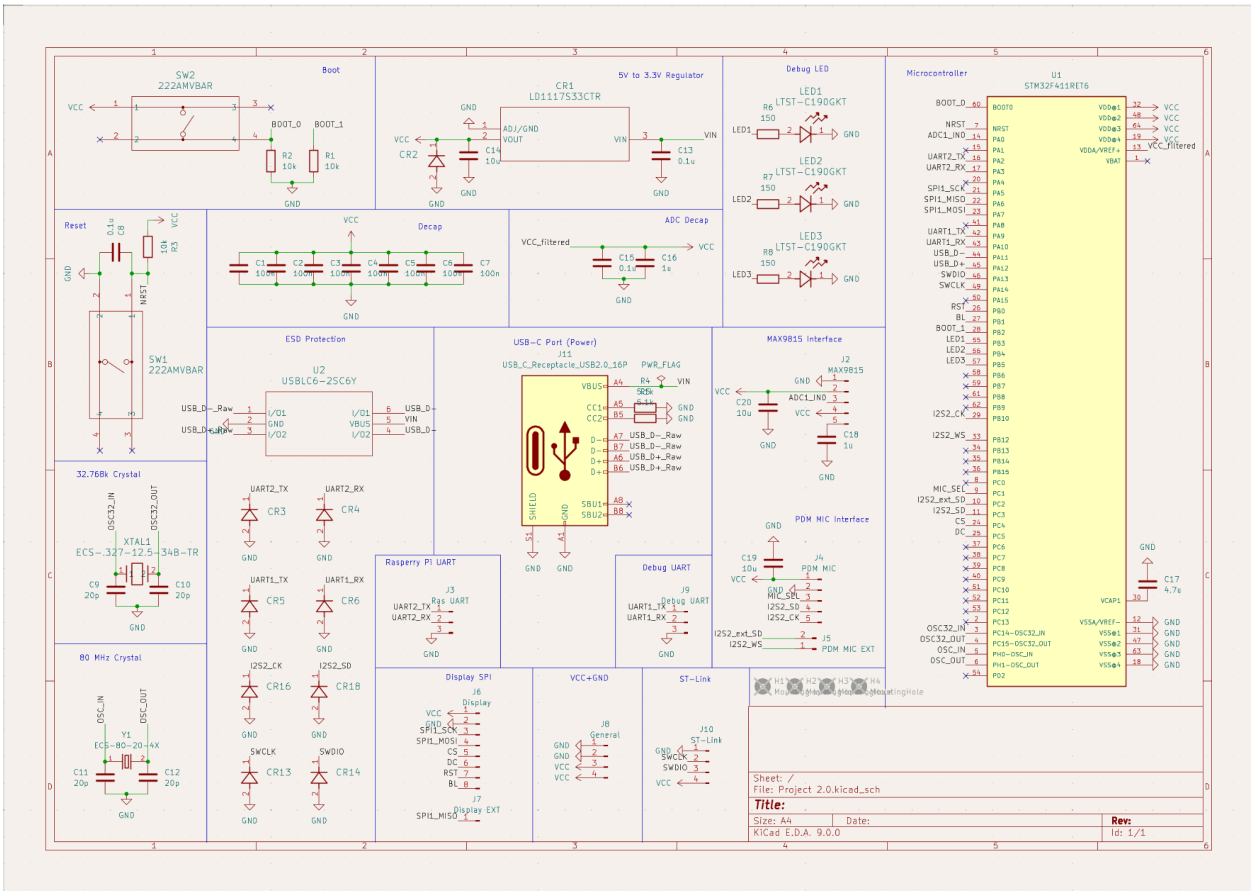

Figure 5: Schematic 1.0.

B. PCB Layout Version 1.0

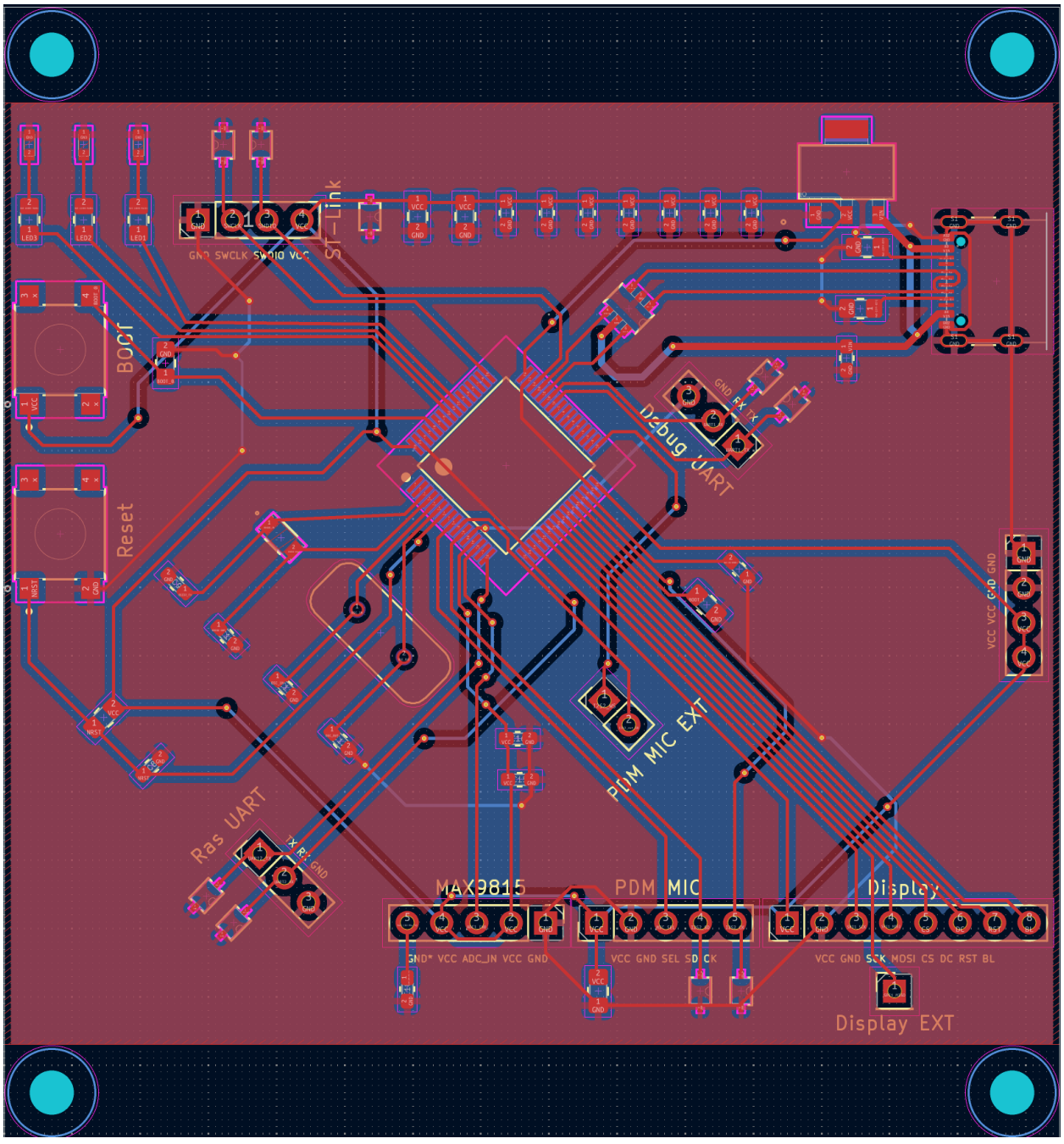

Figure 6: PCB layout 1.0.

C. Schematic Version 2.0

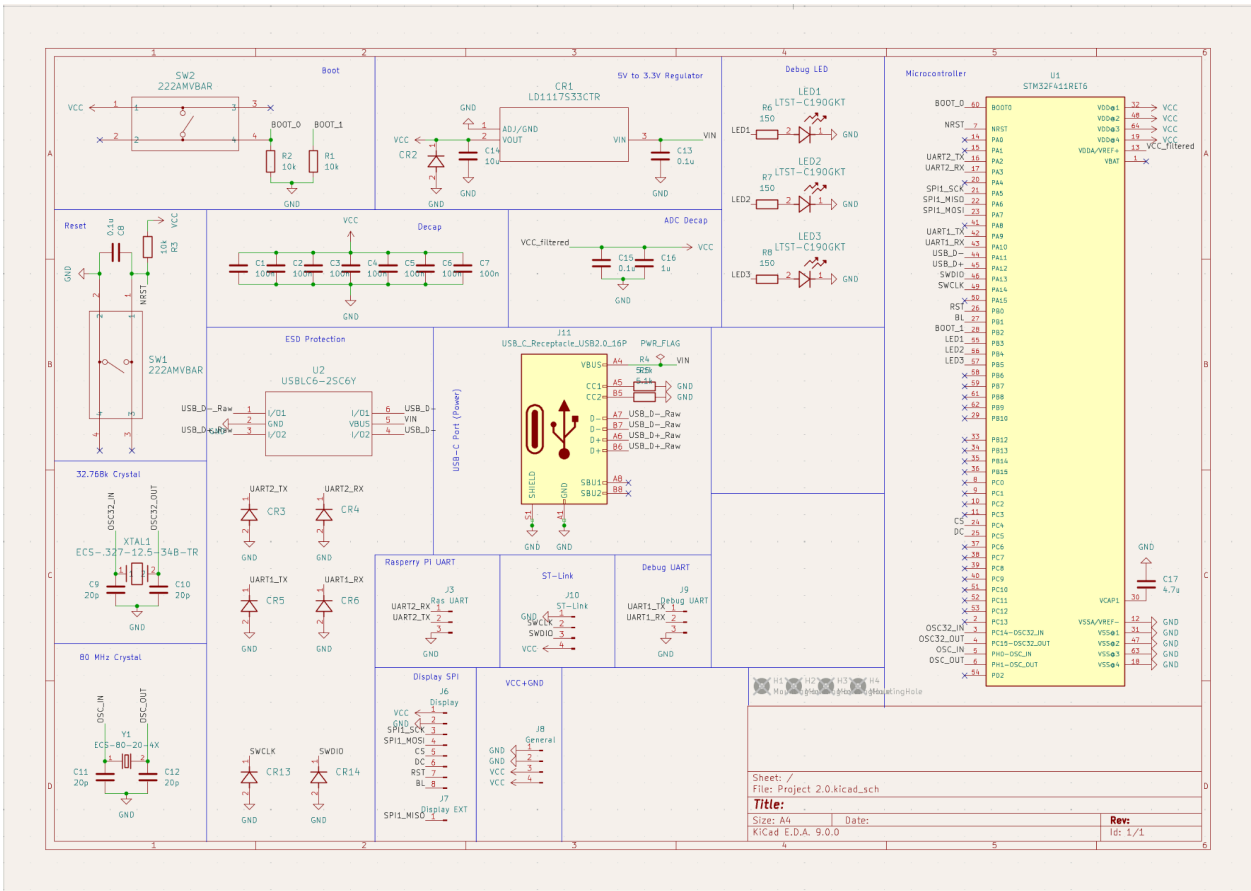

Figure 7: Schematic 2.0.

(Unnecessary I/Os are removed after on board test)

D. PCB Layout 2.0

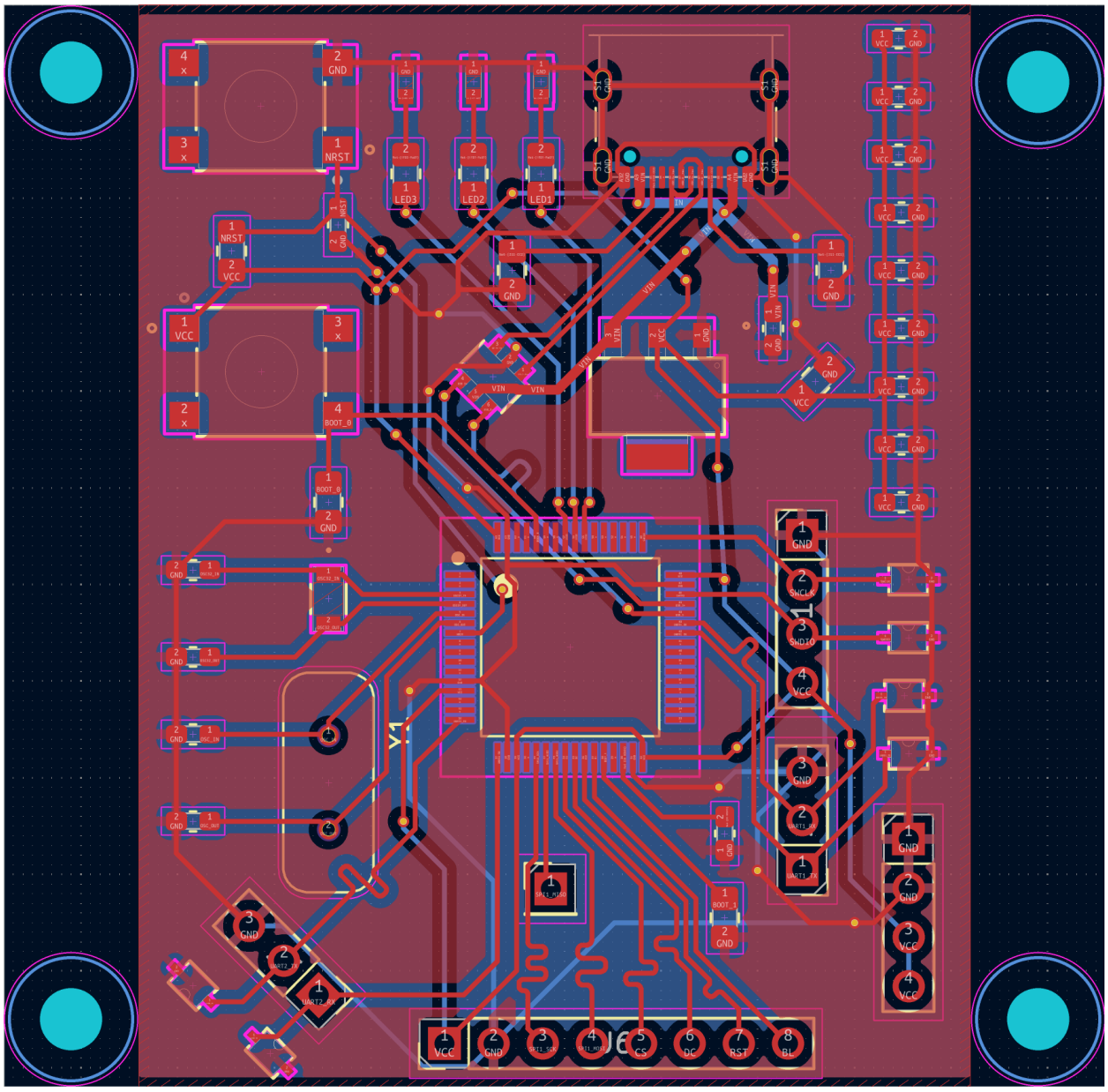

Figure 8: PCB layout 2.0.

(Board size is shrinked by 50% & Signal integrity is confirmed)