Real-Time Sign Language Translator

ECE 445 Design Document - Fall 2021

Project #27

Gene Lee(genel2)

Kaelan To(kto3)

Professor: Wei Hu

TA: Bonhyun Ku

Table of Contents

Introduction

Problem and Solution Overview

Technology is improving rapidly, and with that, it serves the purpose of making our lives easier. We want to leverage the technology available to us and further integrate those with disabilities more into our society; specifically deaf individuals in an academic setting. When we imagine students with hearing impairments working with others, we think of the other students having to wait for the student to type out their thoughts. However, brainstorming/bouncing off ideas requires rapid discussion in order to spark good ideas. Sometimes typing may not be as fast (which might hinder the group) or even may not be accessible to students (especially in K-12) in classroom settings.

We propose a portable real-time sign language translator to solve this problem. We would utilize computer vision to differentiate the hand signs and feed the visual input to a microcontroller and give audio feedback (sound translation of the hand sign). Given enough time, we do also want to add a feature of speech to text in order to help the deaf student understand their peers too. All in all, this portable system would assist them in communicating with teachers, but most definitely help deaf students work in a team with other students efficiently.

Visual Aid

Figure 1: Application of device in classroom setting

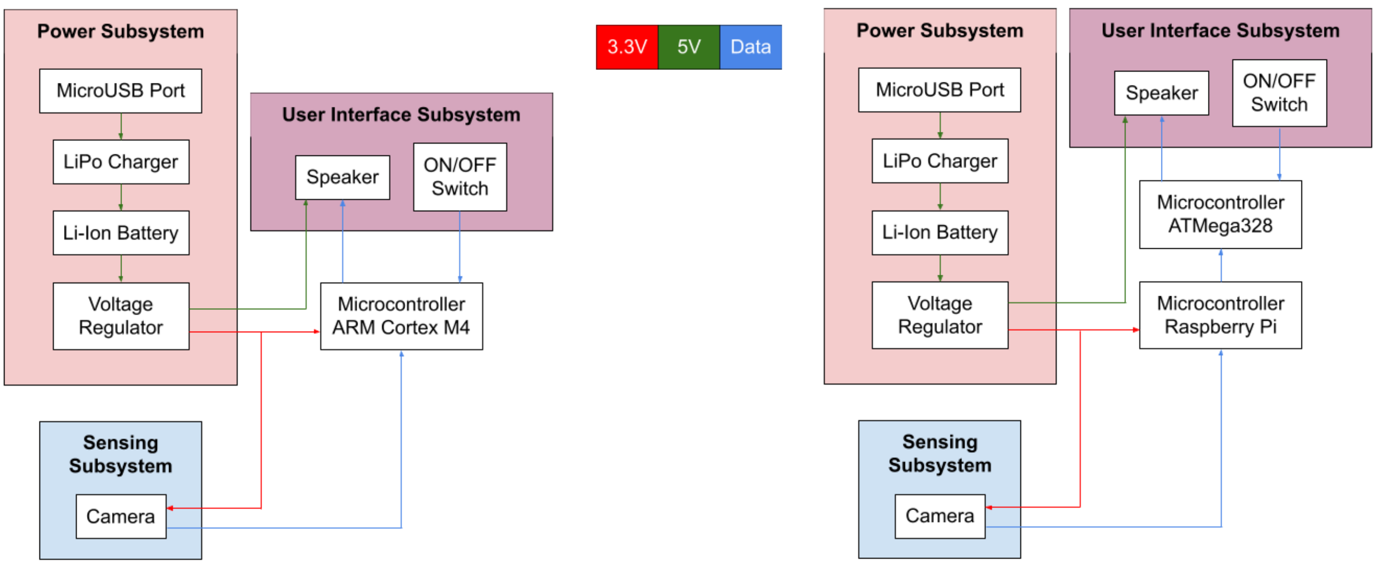
High-level Requirements

For the device to accomplish the minimum goal for helping deaf students in K-12 classrooms communicate with their peers, the system must reach the following requirements:

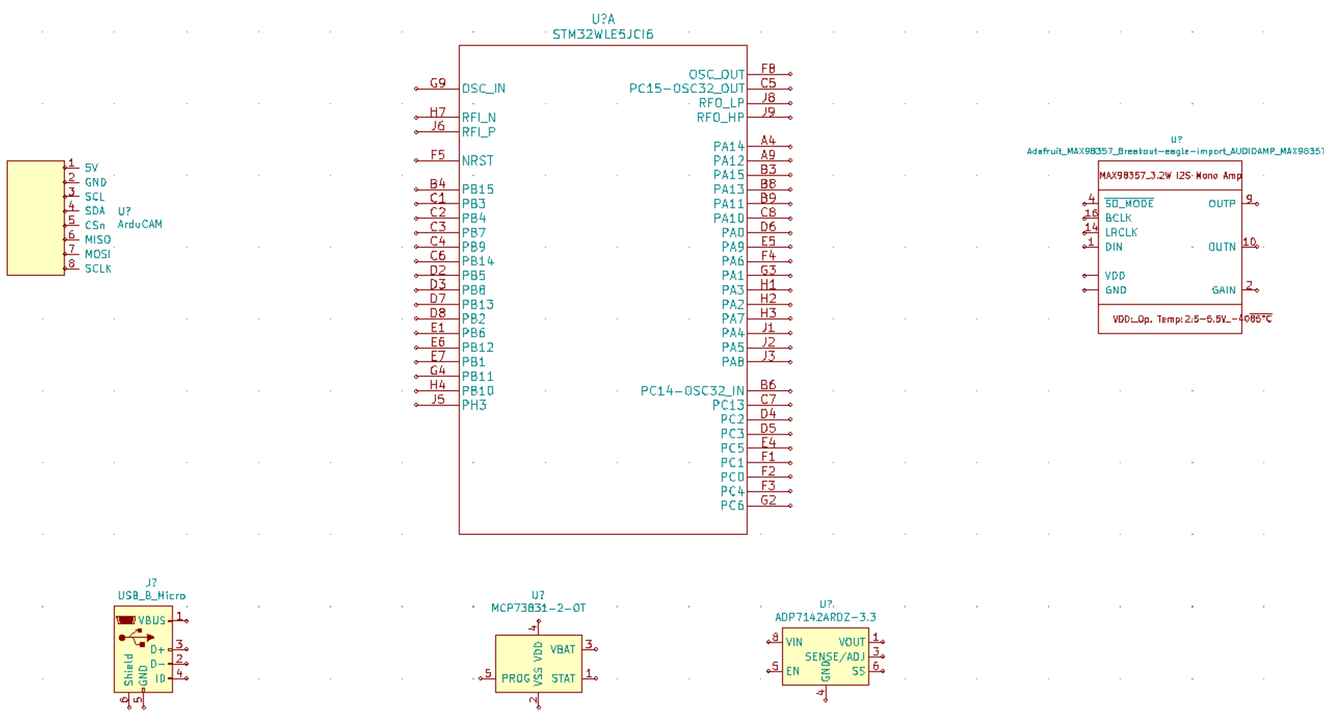
- Ability to identify ASL and translate into English in real-time (threshold set to be within 1.5 seconds)
- System is lightweight/portable (desired size set to be a volume of 15x15x15cm)
- Battery lifetime of at least one school day (8 hours - also subsystem requirement)

Design

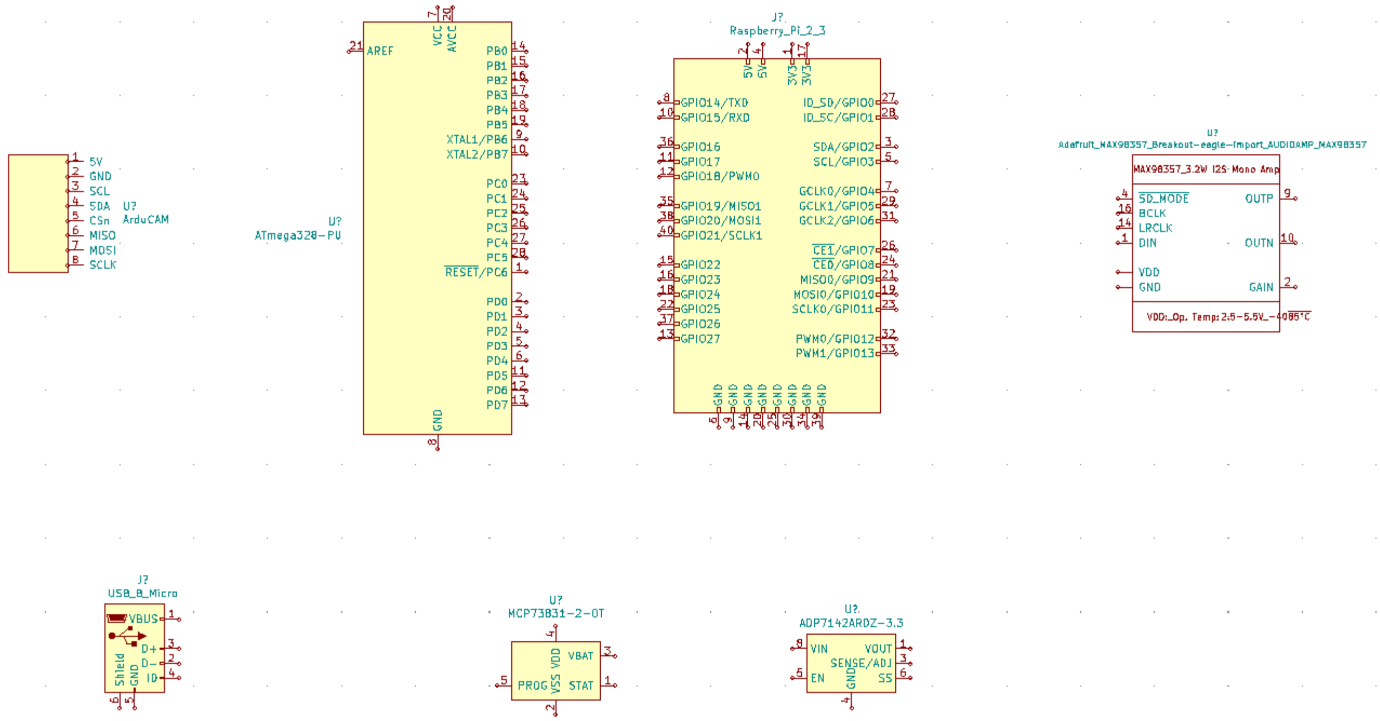
Block Diagram



Schematics and PCB Design



Schematic with ARM Cortex M4



Schematic with Raspberry Pi and ATMega

Physical Design

This entire device will be just one portable box. This device is meant to be used in a classroom setting for school children so it should be easily stored within a classroom or backpack. As such, it would just be a 15x15x15cm box with a camera on the front, a simple on/off switch, and a built-in speaker. It would be placed upon classroom tables in front of the deaf school children such that our visual sensor, the camera, would be able to take in the signed words and send it to our system for processing and translation into English.

Subsystems

Visual Input

Overview: The camera will be the main sensor used for our device. The camera will be used to read sign language inputs from the user and send the information to the central processor.

Requirements: Due to the speed that users will be signing at, and the nature of the pose estimation model, the frames per second and even the resolution needed do not need to be that high.

Requirement	Verification
1. Must be able to capture video at a frame rate when signing at a moderate “conversational” speed	1. Camera is able to properly send images to central processor 80% of the time

Central Processor

Overview: In charge of putting together all the subsystems and processing the inputs to generate outputs. It will take in the visual input and process it using a pose estimation model to generate the necessary audio output to the speakers.

Requirements: Needs enough computing power to run a pose estimation model. Needs to be power efficient enough so that the battery is able to provide enough power to reach our minimum requirement.

Remarks: We will attempt to use a microcontroller with the ARM Cortex M4 as the core (since it has enough computing power) first and try to build our own custom board that will be able to handle all subsystems centrally AND run the pose estimation model. If this fails, we plan on falling back to using a Raspberry Pi to deal with SOLELY the pose estimation functionality of the device. The PCB we design will still be in charge of integrating all the other subsystems together.

Requirement	Verification
1. Able to properly process data input from camera and output audio to the speaker	1. Device is able to detect the correct hand sign and output the corresponding audio

based off it 2. Must be able to process data and write output within 1.5 seconds	correctly 80% of the time 2. Will use a stopwatch to check detection to output time with an error of ±1 second

Audio

Overview: Responsible for producing audio that it receives from the central processor and outputs it through the speakers.

Requirements: It must be loud enough to be heard by multiple people nearby the device.

Requirement	Verification
1. Loud enough to be heard by several people sitting nearby	1. Will test device translate from 1, 2, 3 ,4, and 5 feet to see if the intended audience can hear the output clearly a. Loud background music/noise can be played to further test volume

Power

Overview: Responsible for powering all the components in the device.

Requirements: Must be able to power the device for the duration of a typical school day, which would be 8 hours.

Requirement	Verification
1. Has enough battery to operate the device for at least 8 hours 2. Able to supply 3.3±0.5V and 5V to different components in device 3. Is rechargeable overnight	1. Operate device periodically for 8 hours as well as on standby, if device lasts longer than 8 hours, also passes requirement 2. A multimeter as well as just normal use will be used to check proper voltage values and operation of the device 3. Starting from dead battery, able to charge to full within 8-12 hours

Part Specifications

Power Requirements (only microcontroller)

Device	Voltage/Current Requirements
STM32WLE5JCI6TR	Voltage: 1.8V - 3.6V Current: 4.82 mA (active mode)
ArduCAM-2MP-Plus	Voltage: 1.7V - 3.3V Current: 43 mA (max)
MAX98357A	Voltage: 2.5V - 5.5V Current: 2.4 mA
S-1313D33H-A4T2U3	Voltage:3.3V - 5.55V Current:1.35μA (max)
ICR18650	Voltage: 3.7 V Power Rating: 4400 mAh

Calculation: Battery Power Rating (mAh)/ Component Current Requirement (mA) = 4400/50.22 = 87.6 hrs

Power Requirements (microcontroller + Raspberry Pi)

Device	Voltage/Current Requirements
ArduCAM-2MP-Plus	Voltage: 1.7V - 3.3V Current: 43 mA (max)
MAX98357A	Voltage: 2.5V - 5.5V Current: 2.4 mA
Raspberry Pi Model B+	Voltage: 5 V Current: 400 mA
ATMEGA328-PU	Voltage: 2.7V - 5.5 V Current: 1.5 mA (at 3V)
S-1313D33H-A4T2U3	Voltage:3.3V - 5.55V Current:1.35μA (max)
ICR18650	Voltage: 3.7 V Power Rating: 4400 mAh

Calculation: Battery Power Rating (mAh)/ Component Current Requirement (mA) = 4400/446.95 = 9.85 hrs

Encasing

Overview: Responsible for holding all components together while providing the user with a simple way to interact with the device and achieve our goal.

Requirements: Needs to be small and lightweight (15x15x15 cm) in order to be easily carried around. Needs to be able to fit all of the components and have openings for the camera, speaker, and battery. Needs to be able to let the user charge/change the battery easily.

Requirement	Verification
1. Encasing can house all components	1. Measured dimensions of the encasing have a volume equal to or less than 15x15x15cm or $3375cm^3$

Software Design

The entire project is based on using computer vision to process and translate ASL to English. This task is split into two parts. We plan on using computer vision to help us recognize the individual alphabet characters signed by the user. The next step is to take this entire string of characters and parse them into a coherent sentence.

1. Identifying Characters

We initially planned on using an open source pose estimation library called OpenPose to take in the input and train a model ourselves with our own dataset. However, upon further research, we have found multiple open source models where this task is already accomplished for us. Thus, we can focus on the second part of the task, a component where existing implementations/models have been unable to accomplish.

2. Parsing

The quickest and easiest approach to this is to input our own dictionary of words that our algorithm to “look” for. We plan on slowly building up the library, starting with basic conversational words and hoping to build up to common words used within classrooms. This could be further tailored towards individual classrooms by letting teachers/students input their own words to the library.

Tolerance Analysis

Visual Input

One issue may be the camera frame rate, as it may not be high enough to actually interface well with the microcontroller. The solution to this problem would be to find a stronger, more powerful camera, but that may affect our power subsystem and likely will require the use of a Raspberry Pi.

Power

Given the two designs of our schematic (one microcontroller vs. microcontroller + Raspberry Pi), the power consumption of our system varies drastically. The Raspberry Pi consumes 400 mA alone, and in order to meet our high level requirement of having the battery sustain for at least a school day of 8 hours, we decide to order one battery that would work for both systems. This results in the single microcontroller having a usage of 87 hours on one charge, but the other only being able to last around 9.85 hours on a single charge. This does prove to us that a more specific use schematic tailored towards our needs would be more optimized and would lower the cost at least in terms of the battery.

Central Processor

The main issue that we may run into is that the microcontroller may not be powerful enough to take the camera input and process the data or would purely be too hard to optimize and utilize to our needs. To remedy this issue, we have considered a design that involves a Raspberry Pi as well, which has shown to have camera and audio capabilities.

Software

A potential issue we foresee with the software lies in the second part of the solution: the runtime of parsing the string of characters. If we have a large dictionary, it would take a longer time for the processor to finish parsing the string into a coherent sentence. For example, the most direct algorithm to parse the string would be to iterate through the entire dictionary and use strncmp on the entire string. Purely in terms of Big-Oh notation runtime, this would result in an average runtime of $O(n^2)$. However, this could be easily resolved through various algorithm optimization techniques, for example taking advantage of multithreading, or checking for words more likely to show up first. The latter solution would not change average runtime, but would significantly optimize practical runtime. But of course, we can just stick to basic words as a proof of concept to keep our dictionary short.

Cost and Schedule

Cost Analysis

Over a course of 15 weeks, and with 2 group members who would each work 15 hours per week, we estimate that given a wage of 50 dollars per hour, the total development cost would be around 22,500 dollars.

Part	**Part Number**	**Unit Price**	**Quantity**	**Total**
Camera Module	ArduCAM-2MP-Plus	25.99	1	25.99
Microcontroller	STM32WLE5JCI6TR	9.82	1	9.82
Microcontroller	Raspberry Pi Model B+	29.95	1	29.95
Microcontroller	ATMEGA328-PU	2.65	1	2.65
Audio	MAX98357A	2.44	1	2.44
Li-Ion Battery	ICR18650 4400mAh 3.7V	19.95	1	19.95
Li-po Charger	MCP73831	6.95	1	6.95

Voltage Regulator	S-1313D33H-A4T2U3	1.19	4	4.76
TOTAL COST				102.51

Schedule

Week	Gene Lee	Kaelan To
8/23	Brainstorm Project/Submit RFA	Brainstorm Project/Submit RFA
8/30	Brainstorm Project/Submit RFA	Brainstorm Project/Submit RFA
9/6	Submit RFA	Submit RFA
9/13	Project Proposal	Project Proposal
9/20	Design Document (Parts)	Design Document (Parts)
9/27	Design Document (Software)	Design Document (Circuit)
10/4	Design Review/Finalize 1st PCB	Design Review/Finalize 1st PCB
10/11	Integrate Microcontroller	Populate PCB
10/18	Confirm feasible w/o Raspberry Pi/2nd PCB	Populate PCB/2nd PCB
10/25	Software Development	Encasing Design
11/1	Software Development	Encasing Design
11/8	Debugging/Start Final Report	Debugging/Start Final Report
11/15	Mock Demo	Mock Demo
11/22	Debugging/Start Final Report	Debugging/Start Final Report
11/29	Prepare for Demo	Prepare for Demo
12/6	Presentation/Final Paper	Presentation/Final Paper

Ethics and Safety

The main safety concern when working with electronics is with the batteries. Batteries can be potentially hazardous as they can leak and even explode in the wrong conditions. To comply with IEEE(I.1), we will do our best to not push the batteries past the point of danger. Another possible safety concern is with the microcontroller, as if pushed too far, it may overheat and become a fire hazard, similarly to the batteries. To avoid this issue, we will stay within spec of the devices that we use.

An additional safety concern would be potential sharp edges that our encasing has. As this is directed towards school children, we would want to make sure that they are not accidentally hurt while using this system in the classroom. To this end, we would design our encasing such that the edges are rounded and durable.

A final concern would be privacy. In today's age, many people are concerned that their personal information would be stolen without their knowing. In this case, the camera might pose as a concern to many. To mediate this, we plan on incorporating a sliding mechanism on our encasing to cover the lenses when not in use.

Citations

- https://learn.adafruit.com/how-to-choose-a-microcontroller/the-microcontrollers-in-adafruit-products
- https://www.mouser.com/Semiconductors/Embedded-Processors-Controllers/Microcontrollers-MCU/ARM-Microcontrollers-MCU/ARM-Cortex-M4-Core/_/N-a85pc?P=1z0wa45Z1yztkji&pop=48561&gclid=Cj0KCQjws4aKBhDPARIsAIWH0JUJ3LbzQF8fnkfyH-dGgkeWBP_PC5kJ1A5PXVQglBveFrGWRbh3HUsaAiNnEALw_wcB
- https://www.analyticsvidhya.com/blog/2021/06/sign-language-recognition-for-computer-vision-enthusiasts/