JargonJolt

Design Document - Spring 2024

Team 28

Daniel Chamoun, Luke Hartmann, Nan Kang

6th February 2024

TA: Angquan Yu

ECE 445: Senior Design Project

Table of Contents:

1. Introduction: **2**
1.1 Problem and Solution: 2
1.2 Visual Aid: 4
1.3 High-Level Requirements: 4
2. Design: **5**
2.1 Block Diagram: 5
2.2 Physical Design: 6
2.3 Subsystems Descriptions: 7
2.3.1 Control/Internet Subsystem: 7
2.3.2: Power 11
2.3.3: User Interface 13
2.3.4: Memory 14
2.4 Tolerance Analysis: 17
3. Cost and Schedule: **18**
3.1 Cost Analysis: 18
3.1.1 Labor: 18
3.1.2 Parts 18
3.1.3 Miscellaneous Costs 19
3.1.4 Grand (UPDATE REQUIRED) 19
3.2 Schedule: 20
4. Ethics and Safety: **21**
5. Citations **23**

1. Introduction:

1.1 Problem and Solution:

When learning a new language, amassing and retaining vocabulary is often one of the most challenging parts of the learning process and can be a choke point for advancing into conversational fluency. It is very easy for people to fall off track when learning a new language/new content, especially in the later stages which can prove detrimental to spaced repetition algorithms. According to an American 2021 study by preply.com, 71% of those surveyed who have given up on learning second languages regret letting their language skills slip. Furthermore, 43% of those people stopped studying due to either a lack of opportunity to practice, boredom, or a perceived high level of difficulty. Our project aims to assist those people to continue their endeavors to learn language.

Flashcard applications that already exist do so primarily as mobile or desktop applications. Desktop applications such as Anki have high functionality, but are not portable and could cause the user to miss days if they do not have access to their PC. Mobile applications require that the user has a smartphone, which is not ideal for certain audiences such as children or the elderly. Battery life is also a concern for longer practice sessions and portability.

Our solution is the *JargonJolt*, a digital pet and portable flashcard device that makes consistently practicing your language skills convenient and fun! The *JargonJolt* will take advantage of the "Tamagotchi effect". Named after the popular toy by Bandai, the Tamagotchi effect is the phenomenon of humans becoming emotionally attached to machines, robots, or otherwise inanimate entities. We plan to harness this aspect of human psychology to encourage people to keep up with their daily language review and practice. Nurturing/playing with a digital pet who gets happier as you do better in your flashcard reviews will keep flashcard users more engaged during their reviews as well as more consistent.

Users of the *JargonJolt* will be able to download Anki flashcard sets, where we will make use of spaced repetition algorithms to show users flashcards in optimal order for memory and knowledge retention. The *JargonJolt* will feature a low-power digital ink screen for displaying both flashcards and the digital pet and several buttons for selecting options for responding to flashcards. Applications of similar functionality may exist as smartphone apps, but the *JargonJolt* has unique advantages that give it cause to exist as a product. The simplicity and toy-like nature of the *JargonJolt* makes it ideal for children who are not ready for a smartphone or tablet. A rechargeable battery will also allow users to take their *JargonJolt* on the go without worrying about the battery life of their mobile devices or the cell reception in any given area.

1.2 Visual Aid:

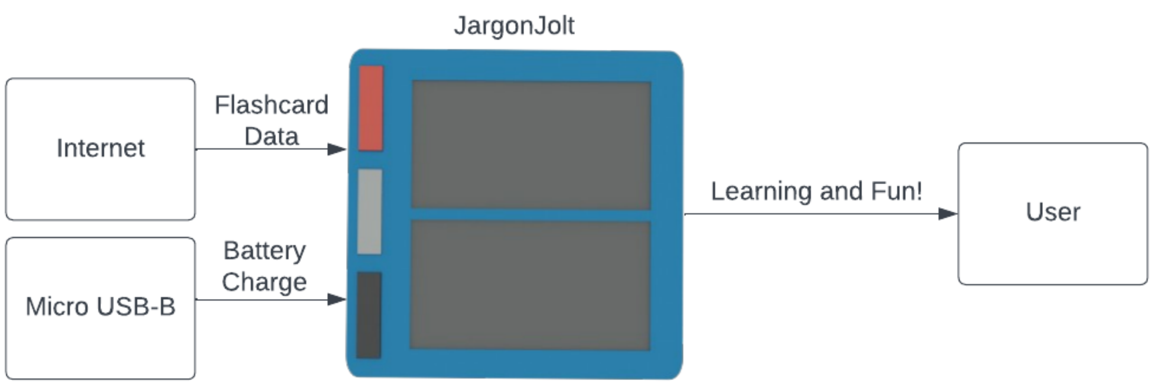

Figure 1: Visual aid showing high level connections (physical representation not fully up to date)

1.3 High-Level Requirements:

In order to create a successful mobile solution, at least the following must be completed:

1. The device enables users to view flashcards, see answers, select their results, and monitor the status of a digital pet. Flipping and switching between flashcards must be completed within 1 second, and the digital pet should respond to any state changes within 1 second.
2. The device must have the capacity to store and recall 'question and answer' data for up to 500 flashcards, in addition to retaining user interaction history with the flashcard set.

Furthermore, it should be capable of downloading flashcard sets from the internet in under 5 minutes.

3. The device should be portable, with dimensions not exceeding 160mm x 120mm, and designed for long-term use. It must feature a rechargeable battery with a lifespan of at least 2 hours on a single charge.

2. Design:

2.1 Block Diagram:

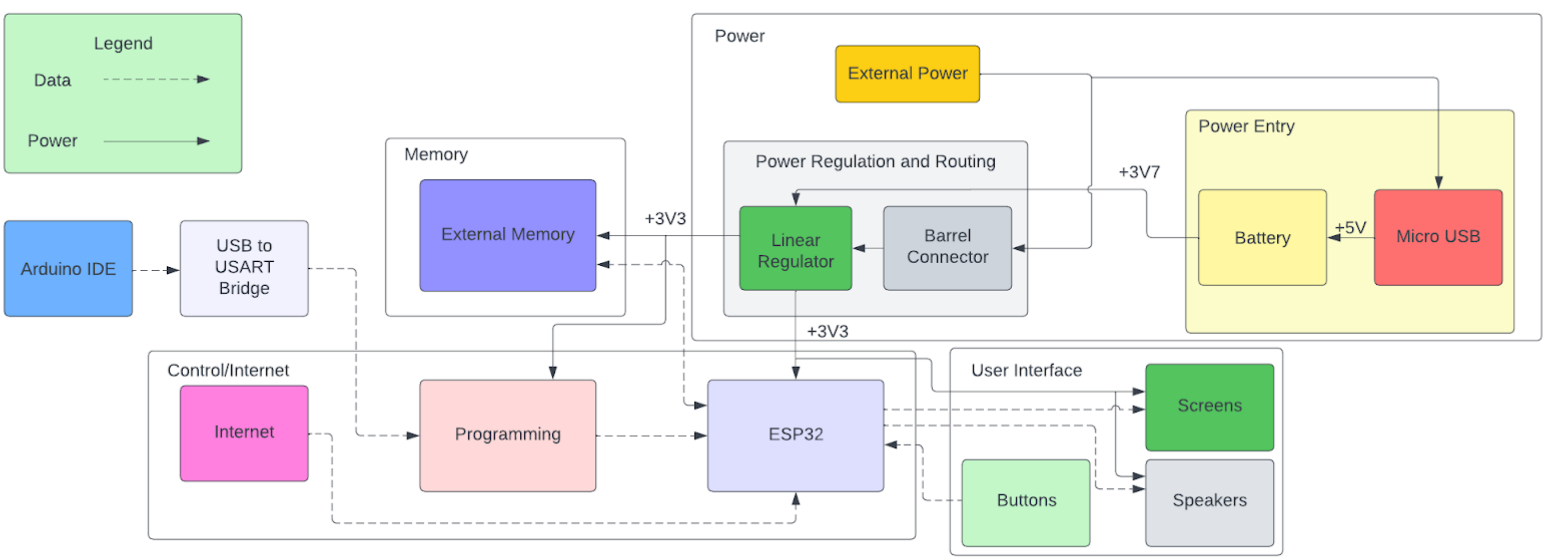

Figure 2: High level block diagram for *JargonJolt*

2.2 Physical Design:

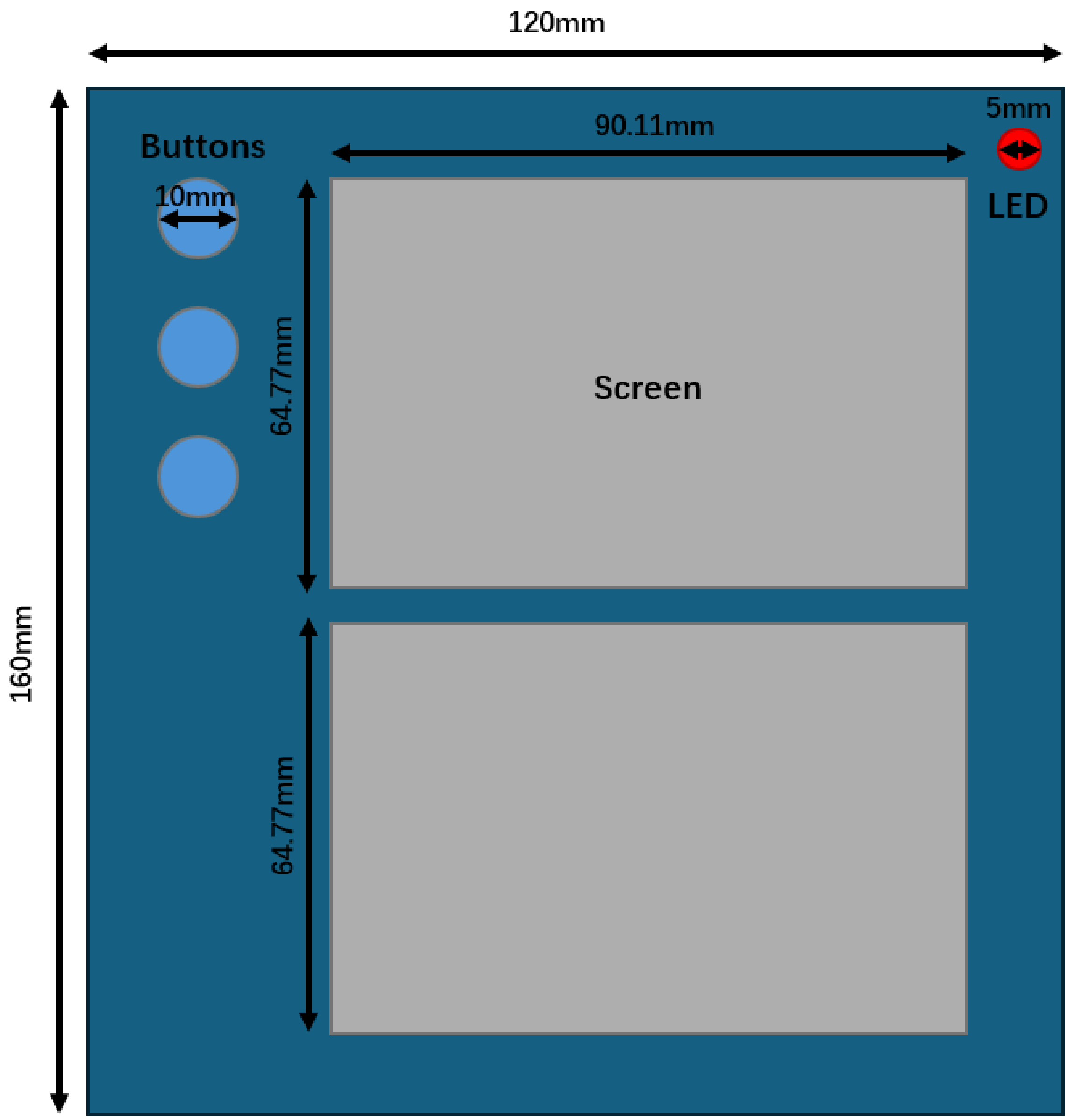

Figure 3: Physical dimensions of the *JargonJolt*

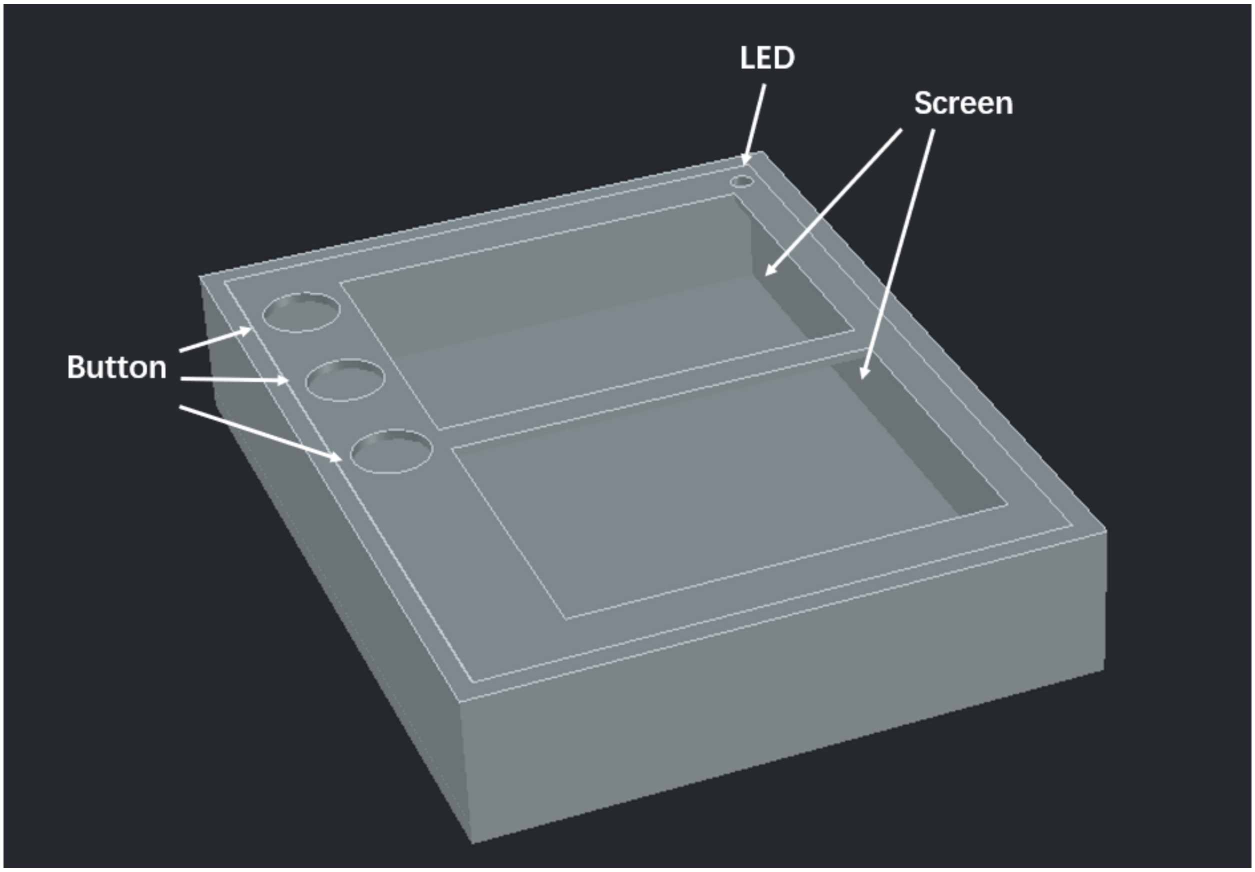

Figure 4: CAD representation of the *JargonJolt* physical design

2.3 Subsystems Descriptions:

2.3.1 Control/Internet Subsystem:

The control/internet subsystem is the most complicated and most important subsystem of this project, as most of the other subsystems depend on the control system to function properly or function at all. The control\internet subsystem consists of the ESP32-S3-WROOM-1 module and all of its direct supporting circuitry, including programing and strapping circuitry. The tasks performed by this subsystem are downloading flashcard sets from the internet, storing and recalling flashcard data to/from external sram (memory subsystem), tracking the user's flashcard progress and the state of the digital pet, and sending/receiving control signals from the user interface submodule.

The control subsystem needs to perform its job quickly and consistently to provide a smooth and comfortable experience for the user. The ESP32-S3-WROOM-1 provides the user with 2 functionally identical general purpose SPI channels, known as HSPI and VSPI. One of these channels will be used to communicate with the memory module, while the other will be used to communicate with the two digital-ink screens in the user interface module. The SPI channels can support running in half-duplex mode at up to 80MHz, which is a much faster data rate than we will require for communication that will feel instant to a human user. In order to send audio to the user interface module, an I^2S line will be used. The buttons in the user interface module will have pull up resistors and will feed into GPIO pins of the ESP32-S3-WROOM-1. +3.3 V will be supplied to the subsystem by the linear voltage regulator in the power subsystem. This submodule is expected to have the highest energy consumption of all submodules.

Figure 5: High level MCU algorithm flowchart

Requirements	Verification
• Communicate with a web server to retrieve flashcard set information	• Ensure the MCU is in an empty state with no flashcards loaded with a constant voltage of 3.3V at the Vin pin. • Then, start the protocol to retrieve text and audio data from the web server. • Then, use a multimeter to ensure the VSPI_MOSI and VSPI_MISO data pins are transferring data when the corresponding chip select is on. • Then check the HSPI_MOSI and HSPI_MISO data pins to ensure a voltage of 3.3 V +/- 0.1 V during data transfer. • Then, check the contents of the external memory by downloading the data on the JargonJolt back to the web server to ensure that data transfer protocols are functioning as intended.
• Read user button inputs	• Ensure all buttons send interrupts to the microcontroller. • Then, based on the button pressed, ensure that the corresponding status updates on the microcontroller. • BUTTON1, BUTTON2, and BUTTON3 pins on the MCU should all show a voltage of 3.3 V +/- 0.1 V upon being pressed in agreement with the MCU's max voltage tolerance.
• Facilitate communication between the memory and user interface modules including audio, display, and flashcard data	• Ensure the MCU can first receive data from the web server. • Then, after updating a flashcard by clicking a button, check that the MISO\MOSI output properly from the VSLI and HSLI lines on the ESP32 which can clearly be seen via the digital screen. • Then, ensure that the audio is functioning by following I2S protocol and probing the data pins.

Table 1: Control/Internet Subsystem - Requirements & Verification

2.3.2: Power

The power subsystem is responsible for regulating and delivering power to the rest of the project. The power subsystem centers around a 3.7 V rechargeable lithium ion battery with 1000 mAh of battery life. The battery is capable of pushing a total of 3600 coulombs, which at 3.7 V is a total energy storage of 13.32 kilowatts. The battery will be rechargeable through a 5 V micro-usb type B port, a common type of port used for phones and other small electronic devices. The battery output will be regulated down to 3.3 V by a linear voltage regulator. The part we have selected is the ADP3339, which is capable of outputting up to 1.5A continuously.

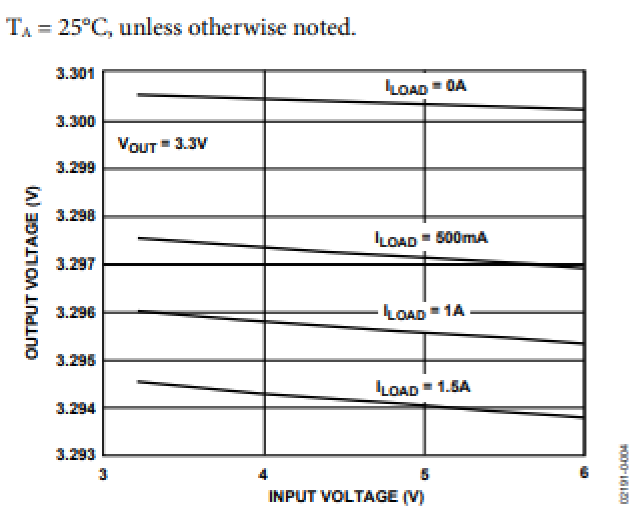

Figure 6: ADP3339 Voltage Input/Output curves with current load lines

According to Figure 6, with an input voltage of 3.7 V the ADP3339 can still output at above 3.294 V even when loaded to the maximum rating of 1.5 A. This is well within the requirements that we have listed in the requirements and verification section. The power subsystem will also consist of a low power LED that signifies to the user when the device is powered on, which is significant as the digital ink screen will not be automatically wiped upon power off. Lastly, the power subsystem includes a switch that will power off the ESP32-S3-WROOM-1 and all of the powered user interface devices. The only device that will be powered at all times is the memory module which contains volatile SRAM.

Requirements	Verification
● When the device is idling, the Power Subsystem must be able to supply at least 500 mA continuously at 3.3 V +/- 0.1 V.	● Ensure the device displays the pet and the flashcard ● Then, use a multimeter to measure the resistance across the power supply by connecting the probes to the input and the output of the Linear Voltage Regulator and selecting the 'Resistance' option. ● Then, obtain the DC voltage out from the power supply using the multimeter using the same probes as before and selecting the 'DC Voltage' option to ensure a voltage of 3.3 V +/- 0.1 V. ● Then, use Ohm's Law to verify that the current out of the linear voltage regulator is at least 500 mA.
● When the device undergoes data transfers or other short-term actions like screen updates, be able to supply at least 1 A for short periods of time to the rest of the system at 3.3 V +/- 0.1 V.	● Ensure the device displays the pet and the flashcard ● Then, after causing a change from the idle mode, use a multimeter to measure the resistance across the power supply output by connecting the probes to the input and the output of the Linear Voltage Regulator and selecting the 'Resistance' option. ● Then, obtain the DC voltage out from the power supply using the multimeter using the same probes as before and selecting the 'DC Voltage' option to ensure a voltage of 3.3 V +/- 0.1 V. ● Then, use Ohm's Law to verify that the current of the linear voltage regulator is at least 1 A.
● Must allow recharging through the use of a USB charging cable with a battery life of at least 2 hours.	● Ensure that the led on the charging board turns on when charging. ● Then, when fully charged, ensure that the status LED turns off. ● Then, while the device is in idle mode, ensure that the screen stays on for at least 2 hours.

Table 2: Power Subsystem - Requirements & Verification

2.3.3: User Interface

The user interface subsystem consists mainly of the pieces of the project that the user directly interacts with. This includes the two digital ink screens, three face buttons, and speaker. It also includes an I^2S amplifier that converts I^2S signals directly from the control unit into an audio signal that is able to be fed directly into a speaker. The digital ink screens will be the Waveshare 4.37inch e-Paper Module (G), which will communicate with the control/internet subsystem via an SPI interface. The digital ink has a slow refresh rate, but the ultra low power consumption in standby makes it ideal for a product that needs to operate on very low power.

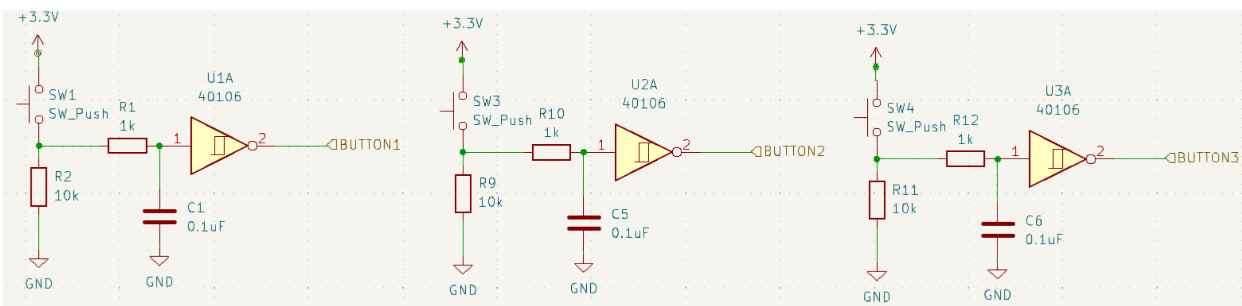

Figure 7: Schematic of face buttons

The face buttons will make use of pull down resistors, such that when the button is not pressed the button pin connected to a GPIO port of the ESP32-S3-WROOM-1 is grounded and no power flows. When the button is pressed, the pin is shorted to 3.3 V. A very large pull down resistor is used so that when the button is pressed, a very small amount of current flows through the resistor and is wasted. To prevent any other unwanted signals from being input, a schmitt trigger will be utilized to ensure no unwanted additional button presses occur.

Requirements	Verification
● Must allow the user to interact with the flashcards	● Ensure that words are properly demonstrated on the screen ● Then, that the user can give positive/negative feedback on the current words by buttons ● Then, that the user can choose/exit different word units ● Then, ensure that the audio from sent from the MCU outputs properly
● Must be able to show the digital pet's status on the screen	● Ensure that the ● digital pet can demonstrate different statuses according to the flashcard learning progress ● Then
● Must have accurate debounced buttons	● BUTTON1, BUTTON2, and BUTTON3 pins should be debounced without any improper inputs

Table 3: User Interface Subsystem - Requirements & Verification

2.3.4: Memory

The memory subsystem is the least complex subsystem included in this project. However, it is still vital to the overall success of the project. The memory module consists of a single chip, the IS66WVS4M8ALL. This chip is a 32 MB external SRAM that can be read from and written to through an SPI interface, and it will interact with the control/internet subsystem to store flashcard data streaming in from the internet and send information back when a request is made. In order to store 500 flashcards on 32 MB of external memory, each flashcard will be allocated 64 KB of data. This chip stores memory in 8 bit words with an addressability of 4 million. Therefore each flashcard will be allocated 8000 addresses worth of memory. This should be plenty for storing text and a short audio clip. Unicode characters are 16 bits long ,so each character takes up 2 addresses. Questions will be capped at 50 characters and answers will be capped at 150 characters, which should be plenty long. This means that together, they will occupy 400 addresses in memory. An additional 2 addresses will be set aside for each flashcard to store metadata such as if the card has been shown to the user before, how well the user has learned the card, and the number of days remaining until the card will be shown to the user again. This leaves 7698 addresses for the audio clip per flashcard. Audio samples will also be given 1 address each, which while lower than the standard audio resolution should be appropriate for these purposes. If stored with a sampling rate of 8000 Hz, this gives about a second of audio, which should be enough for speaking a word or two in another language.

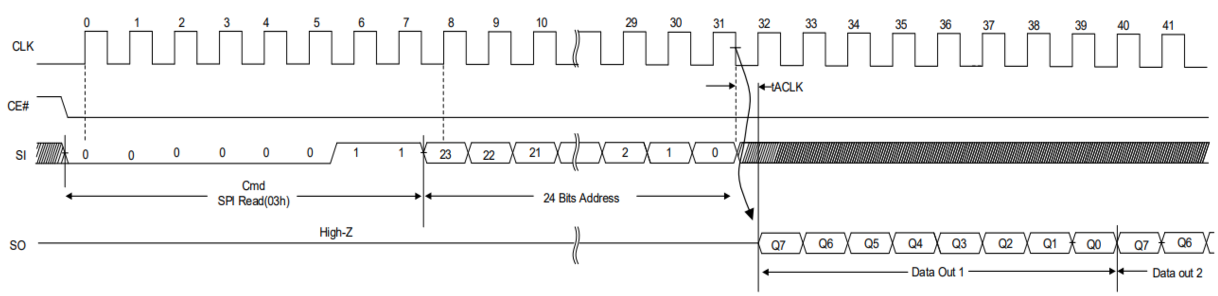

Figure 8: Timing diagram for reading from IS66WVS4M8ALL

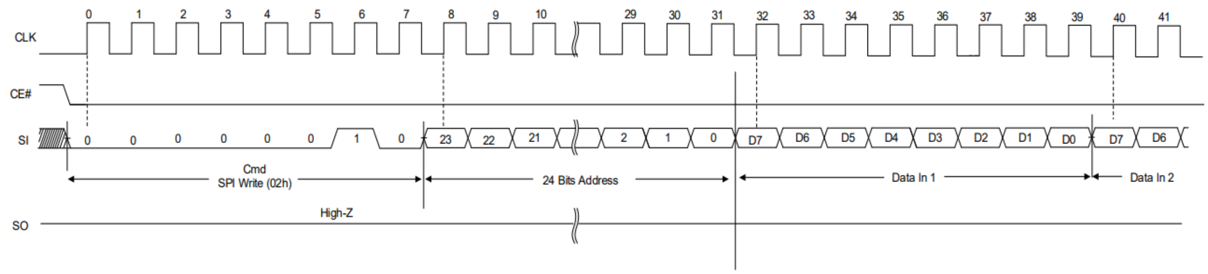

Figure 9: Timing diagram for writing to IS66WVS4M8ALL

Figures 8 and 9 show the timing for both reading and writing to the IS66WVS4M8ALL in 4 line half-duplex communication. Each message will consist of a command followed by a 24 bit memory address and finally data. Once the command has been issued, data will be continuously written/read as long as the serial clock continues to be pulsed, allowing us to write or read data from many locations without having to reissue function commands or addresses.

Requirements	Verification
● Be able to store data for 500 flashcards in 5 minutes or less.	● Create a set of 500 flashcards complete with questions, answers, and audio and ensure download can complete. ● Time the download to make sure it completes within the allotted time.
● Send flashcard data back to the control module upon request.	● Create a custom flashcard set and ensure that data can be recalled. ● Create requests for specific flashcards in the set and ensure the correct data is recalled.

Table 4: Memory Subsystem - Requirements & Verification

2.4 Tolerance Analysis:

One potential cause for concern in this project is the life of the battery. One of the biggest selling points of the JargonJolt is portability, the ability to work completely offline, and being unbound from a mobile device. The JargonJolt is planned to be equipped with a 3.7 V battery that holds 1000 mAh of charge. This means that the battery is capable of supplying 1 amp at 3.7 V for one hour. Using P = IV, this corresponds to a power output of 3.7 watts for 1 hour, or a total stored power of 13.3 kilojoules.

Device	Worst case current draw (mA)	Worst case energy cost (mW)
Digital Ink Screen	18.7	61.7
Speaker	333	1000
ESP32	500	1650
Memory	8	26.4
Total	859.7	2837.01

Table 5: Tolerance Analysis

In the worst case scenario, which would consist of the speakers firing, the screens performing a full refresh, and the memory unit being written to all consecutively, we would still get a battery life of around 1.3 hours. This is not too far off from the goal of a 2-hour long battery life, and this worst-case scenario would be impossible in real operation. Furthermore, the battery could be bumped up to a larger but similar battery that can supply 3.7 volts for 2000 mAh, doubling the total stored energy and pushing us well into spec for battery life. Furthermore, instantaneous current draw is not a concern, as the battery we selected can supply up to 1 A without seeing a dip in output voltage. Linear voltage regulators can approach 95% to 99% efficiency when regulating down from a voltage not far above the output voltage. This is the case in our design, so losing significant power there is not a large concern.

3. Cost and Schedule:

3.1 Cost Analysis:

3.1.1 Labor:

We assume that a graduate from ECE at Illinois makes $35 per hour. We plan to spend about 10 hours per week for the next 10 weeks on this project. Each member will work 100 hours on this project, and since we have 3 members in the team, the total cost of the labor force is $10,500.

3.1.2 Parts

Part Name	Part Number	Manufacturer	Quantity	Cost($)	Purchase Link
ESP32-S3-WROOM-1	ESP32-S3-WROOM-1-N16	Espressif	1	16.53	link
Micro USB-B Connector	10118194-0001LF	Amphenol ICC (FCI)	1	2.95	link
4.37 inch digital-ink screens	4.37inch e-Paper (G)	Waveshare	2	47.98	link
Barrel Jack Connector	694108301002	Würth Elektronik	1	1.02	link
Speaker	SP-1605	Soberton Inc.	1	1.95	link
3.7V 1000mAh Lithium Battery	ASR00012	TinyCircuits	1	9.95	link
Battery Charger	ASL2112	TinyCircuits	1	6.95	link
Linear Voltage Regulator	ADP160AUJZ-3.3-R7	Analog Devices Inc.	1	1.53	link

32Mb, SerialRAM, 2.7V-3.6V	IS66WVS4M8BLL-104NLI	ISSI, Integrated Silicon Solution Inc	1	3.36	link
I2S Amplifier	MAX98357	DFRobot	1	5.04	link
Miscellaneous*	N/A	N/A	N/A	30.00	N/A
				Total: 127.26	

Table 6: Parts Names/Manufacturers - Part List Information

*Miscellaneous SMD and THT components such as resistors, buttons, LED, capacitors, line drivers, wire headers, and CMOS logic will be necessary, but with an unfinalized design this number is not yet locked in. Passive components can be bought for a few cents each, and the other components will cost a dollar or two each. An affordance of $30.00 for these components is a reasonable estimate.

3.1.3 Miscellaneous Costs

Other than parts with manufacturer part numbers that are directly purchasable from sites such as Digikey and Mouser, there are other costs associated with this project. The approximate cost of the PCB material of dimension 160mm x 100mm is about $12. For 3D printing the physical shell, assuming the total amount of print material used is approximately 300g, the estimated cost for this material will be about $35.

3.1.4 Grand

The total cost is estimated to be: $10,500 + $127.26 + $12 + 35 = $10,674.26

3.2 Schedule:

	Daniel	Luke	Nancy
Week of 2/19	● Design review with TA ● Submit order for parts ● Pin layout for ESP32-S3-WROOM-1	● Begin designing PCB schematic ● Work on Design Document Subsystems	● Work on Design Document cost and analysis ● Design 3D model of circuit and screen housing
Week of 2/26	● Prepare for design review with professor ● Research internet connectivity for downloading flashcards	● Prepare for design review with professor ● Finish first draft of PCB design for first PCB submission deadline	● Prepare for design review with professor ● Research SPI communication with screens and memory
Week of 3/4	● Teamwork evaluations ● Begin coding internet connectivity	● Teamwork evaluations ● Order parts pertaining to power subsystem	● Teamwork evaluations ● Begin writing SPI communication related code
Week of 3/11	● Spring break	● Spring break	● Spring break
Week of 3/18	● Firmware coding in full swing; memory organization, internet, display, flashcard algorithm	● Preliminary PCB hardware testing and power testing ● Begin PCB redesign if necessary	● 3D modeled version of device complete, gain access to 3D printer
Week of 3/25	● Individual reports ● Continue coding firmware ● Firmware reached functionality such that testing with hardware can begin	● Individual reports ● Redesign PCB if necessary ● Begin drafting final paper	● Individual reports ● 3D print physical shell of the system ● Assist with coding and firmware testing if applicable

Week of 4/1	• Firmware reaches full functionality • Assist drafting final paper	• PCB hardware testing if redesigned or rework if more applicable • Continue drafting final paper	• Work with Daniel on completing firmware writing and testing • Assist drafting final paper • Integrate electronics with physical aspects of the user interface
Week of 4/8	• Full integration testing and verification • Final paper drafting	• Full integration testing and verification • Final paper drafting	• Full integration testing and verification • Final paper drafting
Week of 4/15	• Mock demo with TA • Last minute corrections and tweaks to firmware	• Mock demo with TA • Last minute corrections and tweaks to PCB via rework	• Mock demo with TA • Last minute corrections and tweaks to firmware and user interface
Week of 4/22	• Final demo with TA • Finalize final paper • Work on final presentation	• Final demo with TA • Finalize final paper • Work on final presentation	• Final demo with TA • Finalize final paper • Work on final presentation
Week of 4/29	• Final Presentation	• Final Presentation	• Final Presentation

Table 7: Schedule

4. Ethics and Safety:

When developing JargonJolt at the University of Illinois, the IEEE Code of Ethics and Safety will be upheld including improving capabilities of emerging technologies (I.2), seeking criticisms of our technological developments (I.5), and crediting those who have contributed to any of our own developments (I.5). In addition to the IEEE Code of Ethics, the ACM Code of

Ethics will also be upheld. This project focuses on limiting the damages that electronics often have on the environment. By using a wide range of power-reducing techniques, JargonJolt will aim to limit the amount of energy needed in accordance with ACM's environmental sustainability principle (1.1). Should any issues arise, we will disclose any problems that our developments may cause in a transparent manner (1.3), all while respecting the privacy of our users' data. To mitigate any high risk factors in this project, all safety precautions will be carefully followed in accordance with safety manuals given by the manufacturers of determined high risk parts such as power supplies or any high voltage devices. Should any issues with parts arise to this regard, a safe procedure will be followed to ensure the defective or unsafe part is replaced, further assuring that the developed device is safe for any user to use.

5. Citations

"Arduino TFT LCD Touch Screen Tutorial." YouTube, YouTube, 7 Dec. 2015, www.youtube.com/watch?app=desktop&v=9Ms59ofSJIY.

Code of Ethics - Association for Computing Machinery, www.acm.org/code-of-ethics. Accessed 8 Feb. 2024.

"Display Examples - Which One to Use?" YouTube, YouTube, 2 Dec. 2018, www.youtube.com/watch?v=H4OuxIwLgHw.

"IEEE Code of Ethics." IEEE, www.ieee.org/about/corporate/governance/p7-8.html. Accessed 7 Feb. 2024.

"Lipos." TinyCircuits, tinycircuits.com/collections/lipos. Accessed 7 Feb. 2024.

"University Code of Conduct." University Code of Conduct - University of Illinois Ethics ..., www.ethics.uillinois.edu/compliance/university_code_of_conduct. Accessed 7 Feb. 2024.

Wienclaw, Laveréna. "Charging TinyCircuits Lithium Ion Polymer Batteries." TinyCircuits, TinyCircuits, 24 Jan. 2022, tinycircuits.com/blogs/learn/charging-tinycircuits-lithium-batteries.

Mykhalevych, Nadiia. "Foreign Language Education Statistics: Second Languages in the US." Language Learning with Preply Blog, 13 Oct. 2023, preply.com/en/blog/americans-foreign-language-survey/.