UCF Senior Design 1

Enhanced Ball Milling Machine

Department of Electrical Engineering and Computer Science
University of Central Florida
Dr. Chan
Initial Project Document and Group Identification
Divide and Conquer

Sponsored by the Blair Research Group

Group 30

Aaron Dahl	Electrical Engineer
Chase Szafranski	Electrical Engineer
Flavio Ortiz	Electrical Engineer
Korey Menefee	Electrical Engineer

Reviewers

Dr. Blair	Physics
Dr. Abichar	ECE
Dr. Borowczak	ECE

Table of Contents

1 Executive Summary

2 Project Description

2.1 Motivation and background

The ball milling machine is a simplistic device that grinds substances into a fine powder by vigorously shaking its contents into a more fine material or mixing them over an extended amount of time, similar to a paint mixer. Deep in the basement of the University of Central Florida's Math Science Building, the Blair Research Group - led by Dr. Blair - has proposed upgrading their current ball milling machine to be more effective towards their research efforts. Currently, the machinery is only capable of its basic and fundamental purpose - grinding substances down. Though the ball milling machine still accomplishes this endeavor, to better assist Dr. Blair and quantify the expenditures - both the cost and maintenance - of his research, our Senior Design group is carrying out Dr. Blair 's request to improve the university ball milling machine and provide it with substantial upgrades. The nature of these improvements will be to measure and detect multiple key parameters and provide a significant increase to user quality of life.

The purpose of these upgrades added to the ball milling machine are intended to provide diagnostic and maintenance metrics to track pertinent information necessary to measure the cost of research and the lifespan of the ball milling machine while also making the ball milling machine more convenient to use. The additional improvements proposed and requested include: a means to measure amperage drawn to track costs of power consumption; time operated to track machine life (similar to tracking mileage on a car to better estimate life cycle and life cycle expectancy); the capability for the ball milling machine to power back on (after, say, a power outage) with memory of where the ball milling left off for more accurate milling times and user quality of life; and automated scheduling, turning the ball milling machine on and off again to keep its motor from overheating to increase and elongate the ball milling machine life cycle. These additions to the ball milling machine will be significant, relevant, and necessary upgrades, providing key metric measurements and documentation while also giving substantial user convenience to be able to automate the motor's on and offs on a schedule to avoid waiting on the ball milling machine to do so by hand, all from the comfort of their home.

Many universities have old but still viable and well running ball milling machines. Sitting aged and rusted in the basement of the Math Science Building on campus exists such a model. Buying a whole new ball milling machine that will cost thousands of dollars is needless for what's relatively could be a cheap addition. The marketability of this project - though granted will be humbly shy of a 'million dollar idea' - will still be exceedingly attractive and greatly appreciated by research groups all across the United States and the rest of the world - from the University of Central Florida to Cambridge. In addition to helping our university and the Blair Research Group this product has the potential to be copied and swapped into another ball milling machine at a fair and reasonable price, giving it a small but noticeable amount of potential sales and marketability; its major selling points being both to save money from buying an entire new model of ball milling machine, and ease of switching, being able to be easily set into the panel of a standard ball milling machine.

Moreover, our motivation beyond helping the university or trying to market a product, this project serves as a culminating challenge of our academic careers to test ourselves and our skills as engineers to locate and fixate on a problem or the potential to improve something, and design a solution and then implement it with the skills we've developed and the knowledge we've culminated. This is an opportunity to prove ourselves capable and worthy of our degree and to join many of our professors and mentors as engineers. This project will be our crucible to prove our mettle and earn our caps and gowns.

2.2 Goals and objectives

Our goal is to enhance fully functional, but aging ball mill/mixers utilized in many research labs today. By achieving the following goals, we hope to modernize the user experience, increase productivity, and implement data-driven decision-making features.

Main Goals

- Enable remote control of mill operations from smart devices, reducing the need for physical presence.
- Upgrade memory storage for seamless recovery after power outages.
- Integrate smart home technology to provide advanced features and control options.
- Introduce environmental sensors to monitor temperature and humidity, safeguarding sensitive materials.

Stretch Goals

- Monitor power draw to identify component issues and facilitate maintenance planning.
- Implement accelerometer to monitor critical component failure such as belt failures.
- Set up remote based scheduling to automate mill operation for specific intervals.
- Set up alerts on the cellphone application that can alert the user within a reason.

Objectives

Many ball mill/mixers utilized in various research labs around the world today only offer rudimentary features such as a manual on/off and a timer. The fact that these machines can only be controlled via manual operation limits their current functionality. The need for a physical person to be present to operate the machine can be burdensome in a research environment where resources, especially time become invaluable. The main goal of our project is to establish remote operations allowing the user to control the mill from their smart device. This will eliminate the need for a person to be present to physically operate the mill, reducing the overall amount of man hours needed to run it.

We want to provide smart features to aging mills that otherwise work perfectly fine. Using smart home technology, we can increase the range of operating modes and features. This leads us to our second goal in which we will utilize the open-source home automation software called home assistant as the base for the user interface. It will be used to control many of the features we wish to implement. The software will allow us to integrate the system into a raspberry-pi4 controlled smart-home. The home assistant software allows users to create custom cards that can be used to display various information which will be used to monitor the mill's status and desired running statistics. The use of open-source smart home software will provide the user with the ability to continuously expand the functionality if they wish as well as for seamless integration within an existing system.

Ball mixer/mills have a wide range of uses and applications involving various materials. Due to the unique properties of certain materials, they can be sensitive to extreme temperatures and require extremely long run times. This can pose a problem as the motor generates heat the longer it runs that can negatively affect both the sample in the mill, as well as the health and lifespan of the motor itself. To help mitigate this problem while simultaneously being able to implement long runtimes our third goal will provide the ability to create run schedules. These will be used to start and stop the machine for specified intervals without the need for constant user interaction. For example, a sample that needs to be run for 30 hours total can be placed in the mill over a long weekend running for intervals of 100 mins on and 100 mins off for a total of 30 hours running.

As previously mentioned, the sensitivity of the materials that are processed in the mills requires them to be maintained in certain environmental conditions. Heat and moisture can ruin samples costing labs precious research dollars. Older facilities tend to suffer more from power outages or problems with their HVAC systems. In a place like Florida where we consistently have high humidity levels if the HVAC system fails the humidity levels inside a facility can increase to sample ruining levels. If no one is physically in the lab operating the machine there would be no way of knowing if a sudden increase in temp or humidity is present. We would like to introduce Various temperature and humidity sensors to measure the humidity and air temperature inside the mill. By monitoring the air temp/humidity inside the machine we can alert the user to any anomalies that may arise while they are away from the facility. By alerting the user via the app they may have time to get to the facility and save the samples before any harm is done. Additionally, by monitoring the motor operating temperature you can ensure that you don't continuously run the motor at threshold temperatures for extended periods of time degrading the lifetime of the motor.

Power outages tend to be another issue that older facilities tend to deal with, lasting only a few minutes to a few hours in duration. When such an event occurs, there is currently no recovery feature to resume to process where it left off or to keep track of this leads us to the upgraded memory storage and recovery feature. We would like to be able to keep track of where the process is so that in the event of a power failure the mill can immediately resume upon the restoration of power. We will accomplish this by keeping a log of the run time and tracking its place in the schedule, updating every minute or so. It will need to be stored in a memory that is noise tolerant and must be able to handle many write cycles.

Certain milling processes require the mill to run beyond the 5-20 minute recommended intervals such as with mechanical alloying. The prolonged use for such an application can potentially cause components to wear down and require extra maintenance. Our final goal is to closely monitor the power draw of the mill. By doing so we can identify unusual load increases due to failing or poorly lubricated components. Establishing a proper maintenance schedule is important to prolonging the life of just about any machine especially those operating at relatively high RPMs. However, given the wide array of uses for mixer/mills there is no one size fits all maintenance schedule. By offering this alternative we wish to allow the operator to determine for themselves based on recorded data when is the best time to perform additional maintenance on their mills.

2.3 Existing product

We found a previous UCF senior design project from 2020, where the team worked with Helicon to automate a chemical process [1]. Setting up the specific run process involves a schedule and controlled rate of flow of liquid through the machines.

Projected Product Similarities and Features

2.3.1 Control loop

Their process involved a feedback loop that measured the rate of flow of liquid through pipes and fed the measured data into a controller which would use the data to control a valve. Our systems main task is to schedule a milling machine and to do this, we will also use a feedback loop. Our control system will ping sensors at a certain frequency and based on these measurements, will determine if the machine should shut down or continue. If these measurements warrant a shut down, the controller will shut down the process, store data and send an error message to the user. The main difference between our system and theirs is their systems output is analog while ours is binary.

2.3.2 Emergency Shutdown

For the Helicon project, in the code of the shutdown routine, the group set a global Boolean flag that if set to true, triggers an ISR (interrupt service routine) that takes highest precedence. After the routine is finished, the program is returned to the main function while the global Boolean flag remains true. Because of this, the main function can go into a function that decides what to do next. The main function always checks this flag. In their system, a second type of shutdown signal can be initiated. This also uses a global Boolean flag. This shut down signal is lower priority and will trigger due to out-of-range sensor readings. The code will determine based on historical logs if this error constitutes a shutdown of the system. The emergency shut down signal and the other shutdown signal are isolated in the code.

Our system is projected to have a similar approach. Our system will have an emergency shutdown, but this will be caused by a sensor reading that is out of range rather than a user input. A user input in our system takes lower precedence as it indicates a remote input. A sensor reading out of range will warrant an emergency because this indicates an emergency. Examples include, the machine overheating, or a canister has been dislodged. We may also do an emergency stop button that is physical or remote as well. I imagine this will trigger the same emergency routine as the out of range sensor emergency shutdown routine.

2.3.3 LCD

In their project, they decided to use I2C to communicate with their LCD display because they were using an Arduino and had to depend on libraries. We are planning to use the ESP32 which is also library based. Their initial strategy was to use an HMI product from Kinco. HMI stands for Human Machine Interface. This is a product that has buttons packaged with an LCD screen. The pros are that it comes ready to use with an Arduino. The downside is that since it comes with a microcontroller, so it is not great for a minimal design-oriented project with the main functionalities on a PCB. For this reason, they decided to not go this route. However, it may be worth considering for our project. But because they used a stand alone LCD screen, they needed to use a Hitachi HD44780 LCD controller chip so that it could be programmed with an Arduino. They detail how this chip is programmed in pages 90-91 which is a good reference in case we decide to go this route.

2.4 Requirements and Specifications

Table 2.4.1 shows our engineering requirements and specifications. We intend to demonstrate the three requirements highlighted below in green. We believe that these features are the most significant in providing a good user experience and are the most visual and straightforward to demonstrate to prove our project. The system needs to quickly boot, quickly respond to user input, and be able to recover from power outages without user interaction.

Two of the most important specifications are the response time to user input and the accurate recovery of the timer state after power outages. If the system takes to long to respond to commands sent from the Home Assistant cell phone application, it creates a poor user experience. Since creating a better experience for the user is one of the main motivations behind this project, it is critical that we do not make that mistake. Accurate recovery after power outages is also a major requirement from our sponsor. The lab that the mixer mill is located in often experiences temporary power outages, and being able to automatically recover from them will be a significant improvement over the existing mill. The time to reboot after power outages is a bit less important for the user experience, but we want to minimize the amount of time that is wasted after these occur. Timer drift is an extremely important specification.

We want our new controller to ensure that the mixer mill is operating for the correct amount of time. The only reason that we currently are not planning to use this specification as part of our demonstration is that it is more difficult to measure during a reasonable runtime for the demonstration. Our timer's error can be measured with more accuracy during runs that last hours.

The last three specifications are required for their respective features to work correctly. To monitor motor temperature, we must be able to measure the full range of temperatures that the motor can reasonable reach. The same reasoning applies for the humidity. There have been cases in the past where the lab air conditioning has failed, and the humidity has approached 95%. We want the upgraded mixer mill to be able to detect these conditions. Finally, the size limit for the primary PCB will allow it to be a drop-in replacement for the existing controller.

Specifications	
Response time to remote user input	Less than 5 seconds
Time to reboot after power outages	Less than 10 seconds
Accuracy of timer state recovery after power outages	Error of less than 60 seconds
Timer drift during continuous operation	Timer error less than 0.5%
Environmental humidity recording	Measure humidity between 5% and 95%

Motor temperature monitoring	Measure temperatures from 15℃ to 175℃
Size of the primary PCB	162 x 85.45 mm

Table 2.4.1: Engineering Specifications

2.5 Block Diagrams

The system's hardware block diagram is shown in Figure 1. The different hardware blocks are color coded according to the team member who is primarily responsible for the block. The central part of the system is the microcontroller unit. It drives the mill's relays, communicates with the Home Assistant server, and reads data from the sensors. The Raspberry Pi will run a modified version of the Home Assistant server software to send commands to, and read data from, the MCU in addition to communicating with the modified Home Assistant app on the user's smartphone.

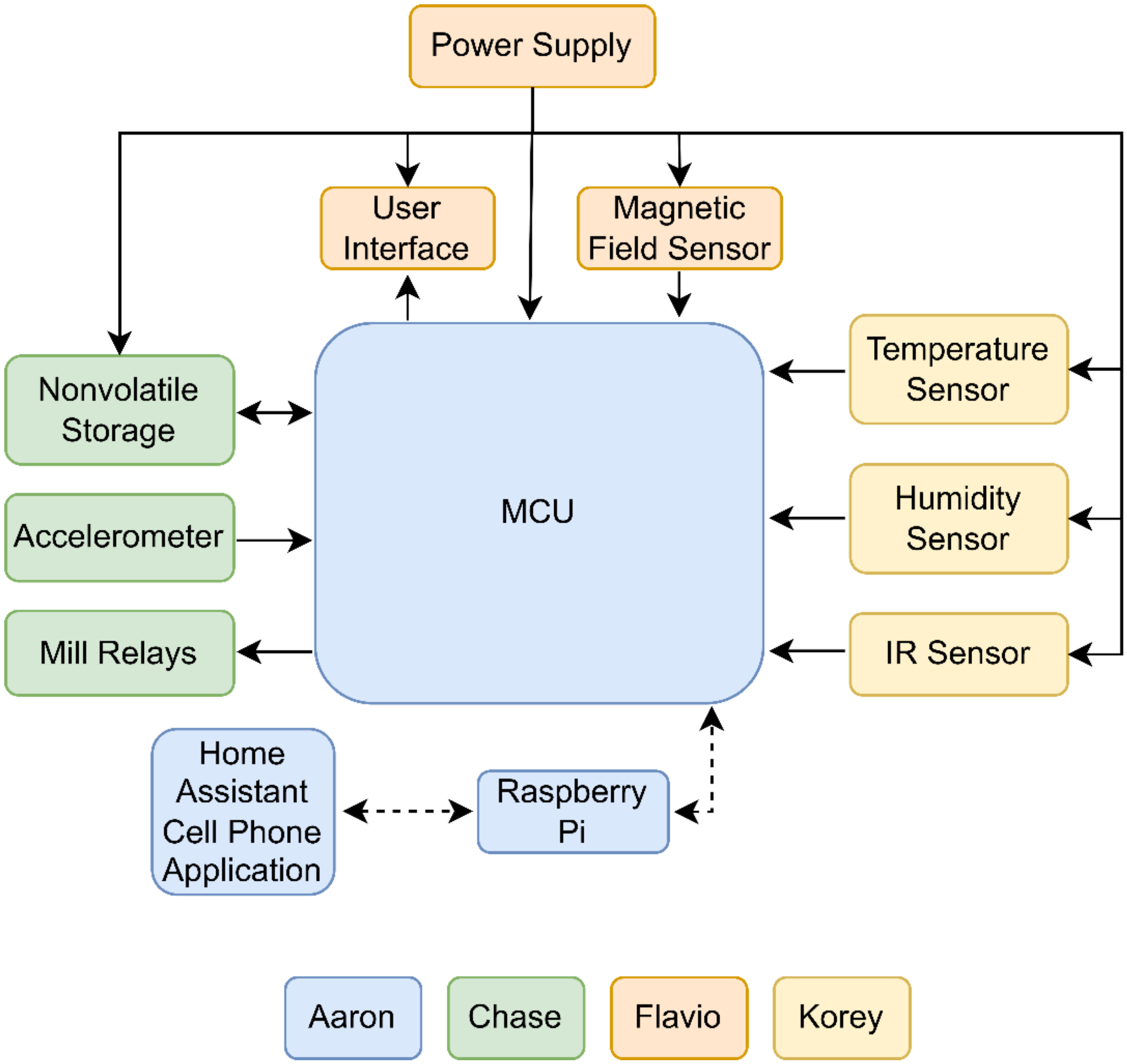


Figure 1: Hardware Block Diagram

The network block diagram is shown in Figure 2. It shows the IoT devices in purple on the left, the LAN in blue in the middle, and the home assistant app in green on the right. The LAN is hosted on a Raspberry Pi 4. MQTT (MQ Telemetry Transport) protocol is a standard networking protocol designed for IoT devices and is what we will use as our LAN. MQTT uses a subscribe model where IoT devices and users can subscribe to topics and/or publish to topics. In this way, only one channel on the WiFi and only one address is needed, making this model light weight. This is good for our use case since we will only be requiring communication between the mill and our app infrequently. The topics in Table 2 are Mill/Sensor_Readings, Mill/Error_Msgs, Mill/Status, and Mill/HomeAssistant_Cmds. The backslash / is used to define scope. For example, if we needed to use each sensor reading individually, we could create topics on top of the Mill/Sensor_Readings as: Mill/Sensor_Readings/Temp, Mill/Sensor_Readings/IR, etc.

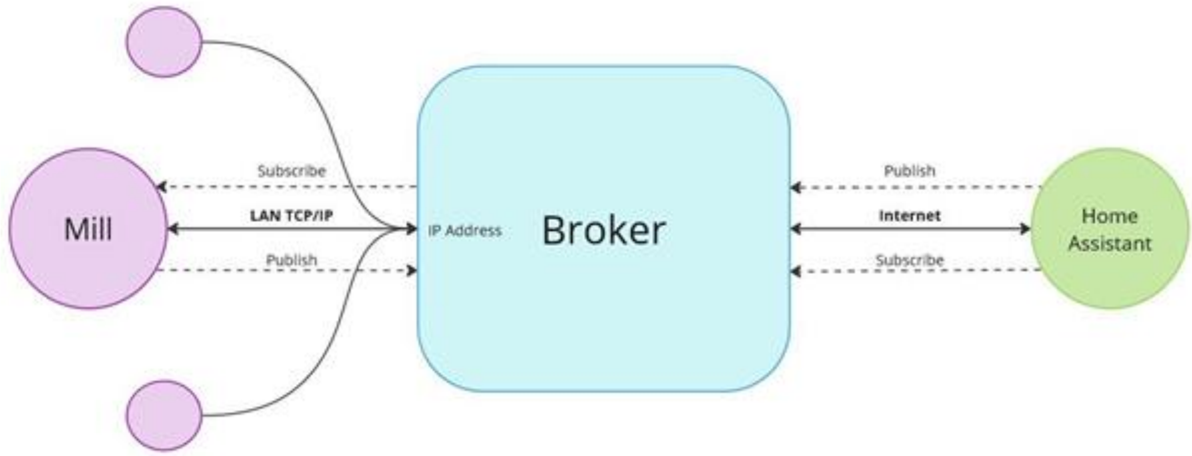


Figure 2: Network Diagram

Mill	Home Assistant
Publish Sensor Readings	Subscribe Mill/Sensor_Readings
Publish Error Messages	Subscribe Mill/Error_Msgs
Publish Status Msg	Subscribe Mill/Status
Subscribe Mill/HomeAssistant_Cmds	Publish Command Msgs

Table 2: MQTT Topics

2.6 House of Quality

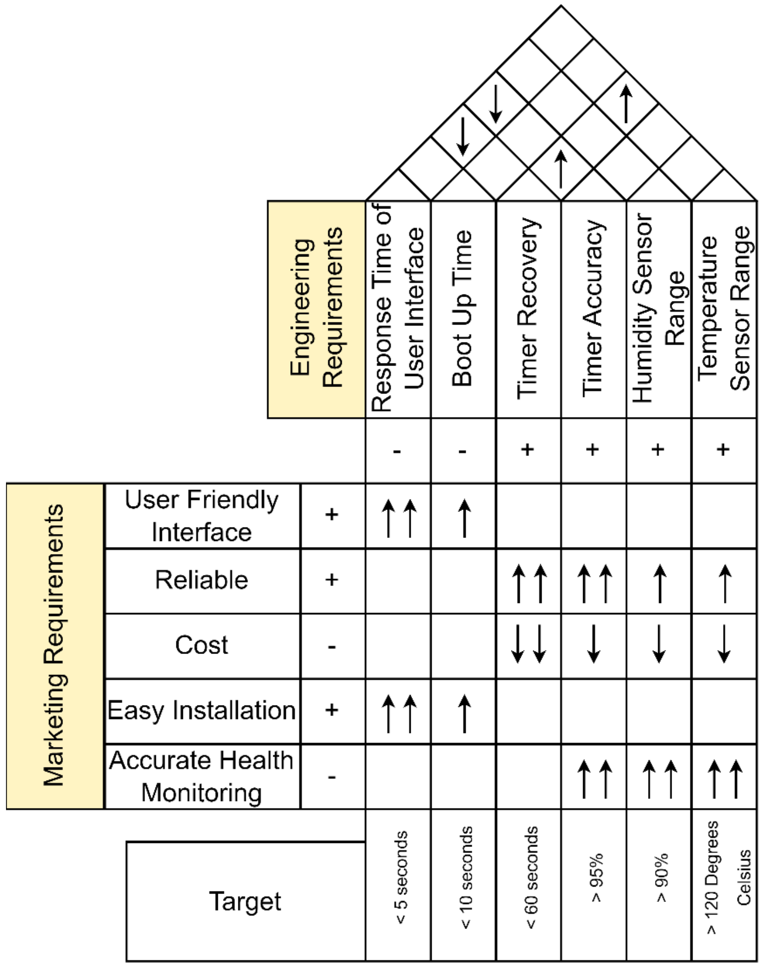


Figure 2.6.1: House of Quality

2.7 House of Quality Analysis

As shown in the project's House of Quality graphic above, the improvements our Senior Design Group is making to the Blair Research Group ball milling machine will also have its costs. For all of the measurement readings from sensors will incur an immediate cost in the form of both purchasing the sensors and the increased power costs to power and maintain them. Thankfully, many of the other upgrades to the Blair Research Group ball milling machine will not incur any cost or downgrade save the time and effort it takes for our Senior Design Group to design and implement them such as the improved user interface and streamlined installation process.

3 Research and Part Selection

3.1 Single Board Computer Comparison

One of the main objectives of our project is to communicate with the micro controller so that health monitoring can be displayed remotely, and functions can be triggered remotely. There are a few ways to achieve this. We could connect the controller to a local network or to the internet. After this, we could rent a server, host our own server, or make an application that connects directly to the MCU. The option with the most optimal bandwidth is to have a local network hosted by a server running an IoT application called Home Assistant. This option is good because other devices in the lab will need the same network capabilities and the same process can be reused for every device added to the network. This method also allows output data from each device to be processed locally first before being sent over the internet. From here, we need to decide what device should host the local network. Every technology compared in this section is in the class of single chip computers.

	HomeAssistant Green	**Raspberry Pi 4**	**ODROID C4**
Driving Frequency	1.8 GHz	1.8 GHz	2 GHz
Storage	32 GB	0	8 – 64 GB
RAM	4 GB	4 GB	4 GB
CPU Bus	64 bit	64 bit	32 bit
Cores	4	2	4
Ethernet	1 Gbps	1 Gbps	10/100/1000 Mbps
Size	4.41 x 4.41 x 1.26 in^3	2.2 x 3.35 in^2	3.35 x 2.2 x 0.4 in^3
Price	$100	$55	$75

Table ___: Single Board Computer Specs Comparison

Temperature

When the power goes out, so will the single board computer go out, so in this case, it doesn't matter how hot the device can operate in. But, in cases where the AC goes out but the power remains on, the temperature in the lab can get into 90 degree Fahrenheit temperatures and greater. Having good cooling will make the device last longer since the CPUs will not burn out due to overheating. The Home Assistant Green has the best performance in terms of operating conditions, being able to operate in the range of 32 to 104 degrees Fahrenheit. This is because it has the largest area and the largest heat sink. The other devices do not have specs on operating conditions. The ODROID C4 comes with a small passive heat sink directly over the CPU. The Raspberry Pi 4 does not come with any cooling features. Even though the C4 and Pi 4 do not come packaged with robust cooling systems, they can be added on since they do not come prepackaged in an enclosure. For the C4, there are not any options for pre-built cooling systems because ODROID is not as popular of a brand, so a cooling system will need to be built manually from standalone heat sinks and fans. The Pi 4, however, has many great options for cooling that will fit right on top of the board. Some options online come with 1 or 2 fans and heat sinks. The price for one of these is $10 to $15.

Other Features

The Home Assistant Green has a couple of features that are nice. The main draw to the Home Assistant Green single board computer is its all-in-one package and plug and play ability. It comes packaged with an enclosure, cooling system, and cables. Because it comes pre-installed with the Home Assistant operating system, no initialization is necessary and so once it is plugged in, it can be used. Another feature is its status LEDs. It has a white LED to indicate if it is receiving power, a green LED to indicate network/transmission activity, and a yellow LED to indicate health or issues. It also comes with an SD slot for backup memory in case onboard memory becomes corrupt. Another pro to the Home Assistant Green, is that it is made by the makers of Home Assistant, so it is built for the sole purpose of hosting Home Assistant. This means the computer does not need to perform well for many applications like any other SBC and so performance should be better suited for Home Assistant. This also means features and documentation will be geared towards Home Assistant applications.

The Raspberry Pi 4 and ODROID C4 are Linux based single board computers and so they will need to be flashed with the Linux distribution or other operating system that it supports. Raspberry Pi supports Home Assistant through their OS installer, so when the SD card with the installer is being booted from the Pi, it gives an option to install Home Assistant as an alternate OS. The C4 version of ODROID does not have documentation for installing Home Assistant. There is documentation for the C2 and N2, but installation for the C4 is considered “experimental” but should have “few surprises” [33]. We could use the N2, but it costs twice as much as the C4, but this might be ok if we decide later a more powerful single board computer warrants the price. Even with the N2, Home Assistant will need to be manually installed and this will be much more difficult. The benefits to using the C4 of Pi 4 is that Home Assistant can be installed as a “container”. This means Home Assistant can run on an environment inside an operating system. This allows for parallel tasks to run alongside Home Assistant. This feature may come in handy later if the circumstances or scope of our project change in the development process.

Pros	**Cons**
HomeAssistant Green • Plug and play • Most robust operating conditions (large heat sink, wide area, holes in enclosure) • Built by the makers of HomeAssistant • Relevant status LEDs	HomeAssistant Green • Most expensive option • Not customizable (software or hardware) • No add-ons available Raspberry Pi 4 • No onboard storage

• Supports eMMC (has SD slot) Raspberry Pi 4 • Supports HomeAssistant • Has many cooling add-ons available online • Has plug and play eMMC module • Has a lot of documentation • Parallel tasking available • Is the cheapest option ODROID C4 • Parallel tasking available • Second cheapest option • Supports eMMC (has SD slot) • General status LEDs	• Least number of cores ODROID C4 • Manual installation of HomeAssistant required • No HomeAssistant documentation for the C4 • The most limited documentation • More expensive than the Raspberry Pi • Sparse cooling packages for the C4 available online • 32 bit data bus

Table ___: Single Board Computer Pros and Cons

3.2 Microcontroller Comparison

3.2.1 Microcontroller Family Selection

For our project we need a microcontroller that is library based. This is because we need to use APIs to communicate with Home Assistant. This rules out the MSP430 board that we used in our embedded systems course. The purpose of this section is to narrow down our options as well as give a good idea of what to expect from each of these options. To start, some of the most popular types of microcontrollers include Arduino MCUs, STM32s, and ESP32s. The Arduino is good because it is user friendly and the MCU comes packaged on a PCB with all the basics. The ESP32 is popular because there is good documentation, and the chip comes with onboard Wi-Fi and Bluetooth. Finally, The STM32 is popular because it has the ARM ISA and has good documentation and available code resources. Because there are so many variations of each of these, some initial research needed to be done to find good benchmarks for comparison's sake. For the ESP32, the original board is the D0WD v3.

The S3 is a newer version that has more pins and generally better specs but has a bigger foot print and is slightly more expensive. It is also widely available in a bagged form and has many different variations whereas the D0WD v3 is no longer sold. I decided to use these two as benchmarks for the ESP32 because if we decide to go with an ESP32, we will likely choose a variation of the S3 or a different product that is in between these two in terms of specs. Note in table 3, the s3 has more GPIO but is bigger and costs more. For Arduino, there are 4 "families" or product lines. These include the Nano, MKR, Classic and Mega. All these product lines have big footprints besides the Nano. For this reason, I chose the Nano product line. Note that in table 3, the BLE has the faster clock rate and better memory. Products to use as benchmarks, I chose to go with the Nano 33 IoT and the Nano 33 BLE. I chose these two because they have good performance specs compared to the other Nanos and have good Wi-Fi and Bluetooth capabilities. One of the main features of our project is networking as an internet of things device, so these two options works well with our application. For the STM32, I chose the STM32G0 family. This is because this option is mainstream and is available to purchase in a single chip form.

	ESP32	**Arduino Nano**	**STM32**
Clock	240MHz	48MHz – 64MHz	48MHz
Core	2	1	1
Flash	0	256KB – 1MB	512KB
SRAM	512KB	32KB – 256KB	144KB
GPIO	34 – 45	22	36
I2C	2	1	3
I2S	2	0	2
SPI	4	1	3
UART	3	1	2
Size (mm)	5x5 - 7x7	45x18	7x7
Price ($)	2.45 – 6.68	27.00	6.35

Table 3: MCU Product Line Comparison

Footprint

Arduino is the worst in this area of the three MCUs. Arduino is unique in that the product is not the microcontroller but the whole package. This means hardware and software together. Arduinos come with various modules and components that are pre-configured to work with libraries that are built into the Arduino IDE. The IDE also makes installing other libraries straight forward. Another advantage to Arduino is the BLE and IoT come with a Wi-Fi module on board the PCB. The tradeoff however the footprint. Arduinos do not come packaged in an MCU IC so the 45mm x 18mm PCB is the smallest form from Arduino that can be soldered onto our PCB. STM32s come packaged in an MCU IC in the sizes of 5x5, 7x7 and 8x8 (mm).

The one I used for comparison is a surface mount and comes in 7x7. There are also options available for 5x5 and 8x8. The downside to the STM32s is that they do not have built in Wi-Fi. To get Wi-Fi capabilities, another IC module will be needed increasing the footprint by about a factor of 2. There is an STM32 that has built in Bluetooth. These STM32s are apart of the STM32WB family. This is a valid option for working with Home Assistant as Home Assistant can connect using Bluetooth, but the downside to Bluetooth is it is not as efficient of a technology as compared to Wi-Fi for our application. Finally, the ESP32 has the smallest foot print of all 3. This is because it has Wi-Fi built into the MCU chip making it smaller than the STM32s. They come in 5x5, 6x6, and 7x7 (mm).

Documentation

	ESP32	**Arduino**	**STM32**
IDE Set Up	Manual	Automatic	Automatic
MQTT Support	Yes	Yes	Limited
Mosquito API Support	Yes	No	No

Table 4: *Documentation Comparison*

Pros	**Cons**
ESP32 • Built in Wi-Fi and Bluetooth • Smallest footprint • Many variations available to buy • Every variation has 2 cores • Very fast processor • The most of SRAM • 2nd most I2C/I2S pins (total = 4) • Most GPIO pins (total = 45) • Good MQTT documentation STM32 • Built in LoRa and Bluetooth • 2nd best footprint • Many variations available to buy • Many variations have 2 cores • The most total RAM • 2nd most GPIO pins (total = 36) • Most I2C/I2S pins (total = 5) • Some what easy to set up IDE Arduino Nano • Easy to set up IDE • Easy to program • Good documentation • Comes packaged with Wi-Fi • Comes packaged with power regulation • Comes packaged with Antenna	ESP32 • No Flash memory • Difficult to set up code environment STM32 • No built in Wi-Fi • Limited SRAM • Limited MQTT documentation Arduino Nano • Only 1 pin for I2C/SPI/UART • No variations available to buy • Least amount of GPIO pins (total =22) • Useless components on board the PCB • Bad footprint • Very expensive

Table 5: MCU Product Line Pros and Cons

3.2.2 Microcontroller Part Selection

After deciding that the ESP32 product line has the best feature set for our project, we still need to decide on a specific model of ESP32 microcontroller. Espressif has several different versions of the ESP32, and each version of microcontroller can be purchased as a bare SOC, a module with the SOC, a crystal, an RF matching network, and an antenna connector, or a module with the SOC, a crystal, an RF matching network, and a trace antenna.

Table 3.2.2.1 shows the peripherals included in several of the ESP32 options that we considered. The different versions of the ESP32 and ESP32 modules optionally include trace antennas on a small PCB, different amounts of GPIO, and support for varying versions of the Bluetooth standard.

Starting from the top of the table, three of the modules we considered had trace antennas, while the bare SOC did not. While designing our own antenna would sound cool, it doesn’t add any value to our system, and creating successfully functioning antennas and matching networks is outside the intended scope of our project. We therefore decided to focus on the ESP32 modules that included either an onboard antenna, or a matching network with a provided connector for an antenna on the module.

We additionally considered the Bluetooth versions supported by each model of ESP32. While we have decided that our primary wireless communication protocol should be Wi-Fi, we hope to use the bluetooth functionality in our stretch goals. This will occur when we create a small wireless accelerometer module that will attach to the lid of the shaking canister. With the speeds that the ball mixing mill moves at, wires are impractical for transmitting data between the stationary main board and the shaking accelerometer module. We therefore plan to use bluetooth. Since that module must be completely wireless, it will also need to be battery powered. In order to minimize the drain from the module’s battery, we decided that we want our microcontroller to support Bluetooth Low Energy. This additional requirement further ruled out the ESP32-PICO-V3.

The next criteria we considered was the amount of volatile and nonvolatile memory available on the board. The specifications for the four ESP32 products we considered are shown in table 3.2.2.1. We plan to use nonvolatile storage to save information about the current system configuration and timer state for recovery after potential power outages in addition to the long-term health monitory statistics. The volatile RAM memory will be used for storing program variables while the program is running.

While we intend for our system to primarily use FeRAM as discussed previously in section 3.1.4, we want to have sufficient flash memory capacity as a fallback. This memory could be used in case of damage to, or unreliable communication with, the FeRam module. Given that flash memory wears out orders of magnitude faster than FeRAM, any fallback system based on flash memory needs to have sufficient capacity such that it will wear at a reasonably slow rate. This is a point in favor of either the ESP-32-WROOM-32E or the ESP32-S3-MINI-1.

For our application, the standard 512KB or 520KB of SRAM should be plenty. We don't really need the additional 2MB of PSRAM that some variants of the ESP32 offer. Having more memory is always beneficial for future extensibility, but we decided that it should not be a major factor for our project.

The next criteria we considered was the processor technology included in the ESP32 options, show in table 3.2.2.1 As these devices are all variations on the ESP32, there is not a lot of difference between the processors

All four of the ESP32 versions that we seriously considered for our selection of primary system controller had 240MHz dual-core microprocessors. The newer ESP32-S3 module that we considered had a newer version of the architecture than the older three ESP32s. This is nice to have, but certainly not essential for our application. Our Wi-Fi transmissions should be using a fairly low amount of bandwidth, and the tasks that our system needs to complete will otherwise be computationally simple. There are additionally single-core versions of the ESP32, but we those out in an earlier stage of the part selection process due to the fact that they have extremely low amounts of GPIO pins, going as low as 5 or as high as 15. The single core would likely have been sufficient, so the fact that the ESP32-S3 has a slightly better version of the dual-core architecture is insignificant.

The final criteria we considered were the physical dimensions and the cost of each variant, shown in table 3.2.2.4. The cost differences are fairly insignificant for our small scale use case. The $1.18 difference between the cheapest option and the most expensive option does not matter when we only need to buy a few copies of the chip. This cost difference would be a more significant consideration if we were planning to manufacture our project at a larger scale.

	ESP32-WROOM-32E [8]	**ESP32--MINI-1 [9]**	**ESP32-PICO-V3 [10]**	**ESP32-S3-MINI-1 [11]**
Included Antenna	Yes	Yes	No, SOC only	Yes
Available GPIO	26 Pins	28 Pins	29 Pins	39 Pins
Bluetooth Version	4.2, LE	4.2, LE	4.2	5, LE, mesh

Flash	4, 6, or 8 MB	4 MB	4 MB	8 MB
Ram	520 KB SRAM and 2MB PSRAM	520 KB SRAM	520 KB SRAM	512 KB SRAM and 2 MB PSRAM
Processor	240 MHz dual-core LX6	240 MHz dual-core LX6	240 MHz dual-core LX6	240 MHz dual-core LX7
Dimensions	25.5 x 18 x 3.1 mm	13.2 x 19.0 x 2.4 mm	7.0 x 7.0 x 0.94 mm	20.5 x 15.4 x 2.4 mm
Board Area Consumed	459 mm^2	250.8 mm^2	49 mm^2	315.7 mm^2
Footprint Type	Castellated Mounting Holes	QFN	TFQFN	QFN
Current Cost from Digikey	$2.50, $2.80, or $3.00 depending on memory size	$2.30	$3.48	$3.10

Table 3.2.2.1: ESP32 Part Comparison

The size is a very significant factor. We have a constrained maximum size for our PCB, so reducing the size of our microcontroller will make it much easier to design the rest of the board and accomplish all necessary routing in the space we have. This gives us a real incentive to use the bare SOC of the ESP32-PICO-V3. Unfortunately, that benefit is outweighed by the cost of not having an included antenna. The next smallest option, the ESP32-MINI-1 is 20.56% smaller than the ESP32-S3-MINI-1, and 45.36% smaller than the ESP32--WROOM--32E, which suggests that the ESP32-MINI-1 may be the most practical choice. It gains this size improvement compared to the ESP32--WROOM--32E primarily by having a QFN footprint instead of using castellated mounting holes. This increases the difficulty of soldering the part to the printed circuit board, which we originally believed to be a fairly minimal trade off for the decreased size. Discussions with a team that is currently working on an unrelated senior design project in senior design 2 made us reconsider this trade-off. They found the ESP32-MINI-1 to be extremely difficult to work with. They said that it was difficult to solder, and that it was challenging to tell if you had soldered it correctly since the pads are completely under the package. An additional difficult that we had not considered was that the under package pads are impossible to probe with an oscilloscope or digital logic analyzer. The castellated mounting holes of the WROOM package are very easy to touch with a probe. These newly acknowledge difficulties made us change our initial decision.

We originally decided that the ESP32-MINI-1 was the best microcontroller for our project. It has enough GPIO pins for our needs, has plenty of flash memory and SRAM, includes a trace antenna, and is a little bit cheaper than the other options. Most importantly, the ESP32-MINI-1 was the smallest version we considered that also included a trace antenna. This product would give us sufficient resources for our project, while also consuming a reasonable amount of space on our circuit board. After hearing firsthand about the practicality of consuming a little bit more board space to gain the benefit of having the exposed castellated mounting holes, we reconsidered our decision. While using up an additional 208 square millimeters is unfortunate, we ultimately decided to use the ESP32-WROOM-32E module. This module includes an antenna and should have sufficient GPIO pins for our needs. It additionally offers versions with more flash memory and has 2MB of PSRAM on package to augment the 520 KB of flash. This extra memory isn't really required with our current plan for the project, but could be used in future extensions. Extra memory is never a bad thing to have.

If we decide to add more peripherals to the project, and therefore require a greater amount of GPIO pins, we will use the ESP32-S3-WROOM-1 instead. The ESP32-S3-WROOM-1 has 36 GPIO pins in exchange for a slightly increased price [20]. It has the same area as the ESP-32-WROOM-32E, and sports many of the same improvements as the equivalent part update between the ESP32-MINI-1 and the ESP32-S3-MINI-1 including the newer processor and increased memory.

3.3 Wireless Communication Comparison

3.3.1 Bluetooth

A major component of our project is communication between sensors and the controller (ESP32), and between the controller and the network port (Raspberry Pi 4). We have a lot of sensors and controlling. For sensors, most of our sensors will use wired communication via I2C or SPI. But there are sensors on the market that use Bluetooth or even Wi-Fi. These might come in handy especially if we have the need to communicate to a sensor that is in an awkward spot on the mill such as the arm. For Home Assistant, we will need wireless communication between our controller and network (ESP32 and Raspberry Pi). This is because the mill is located in a spot that is hard to get an ethernet cable to, and we want the freedom to move it or connect to other equipment in the lab. For these reasons, it is good to know what the differences are between the different standards, and what situations they might be useful in.

There are two Bluetooth standards; they are classic Bluetooth and Bluetooth Low Energy (BLE). Both utilize the ISM frequency band of 2.402 GHz to 2.480 GHz. They use a method called adaptive frequency hopping (AFD). They both have designated channels in the band. BLE has 40 with 2MHz spacing while classic Bluetooth has 79 bands with 1MHz spacing. The transmission protocol will swap between the designated channels at a rate of 1,600 hops per second. This is relevant because Bluetooth is a low powered application and so maintaining reliability is more difficult. Bluetooth uses master and slave architecture. These factors are what make Bluetooth a personal area network and is the main difference between Wi-Fi which is a local area network. Bluetooth must work while being the lowest priority on the ISM band. This means it does not get assigned any bandwidth unlike a LAN which has full reign of its band. The practical differences between BLE and Bluetooth are listed in the table below. (Note: Mbps stands for Mega bits per second and Kbps is Kilo bits per second and so is the transmission rate and/or bandwidth).

	Classic	**BLE**
Data Rate	1 / 2 / 3 Mbps	0.125 / 0.5 / 1 / 2 Mbps
Communication Type	Point to point	Point to point, broadcast, mesh
Applications	Data transmission, audio	Data transmission, audio, location, device networking

Table ___: Classic Bluetooth vs BLE

To explain the features listed above, classic Bluetooth is simple and is designed to provide one service very well. BLE is designed to have more services in exchange for speed. Both do point to point connections where both devices are acknowledged before transmission occurs. This transmission type is asynchronous and synchronous connection oriented. This is used for audio and data transmission. Classic Bluetooth can transmit data much faster than BLE, so it is better for these applications. BLE has two added transmission types: asynchronous / synchronous connectionless transmission, and isochronous connectionless transmission. Connectionless is where transmission occurs without any acknowledgement between master / slave. This allows for broadcasting where devices can connect without permission. Isochronous transmission is data transmission between one master and two slaves where the slaves can transmit data to the master unlike broadcast. Isochronous allows for mesh (or distributed) networking among multiple device kind of like a daisy chain. BLE is also able to find the location of a device. Location can include presence, direction, and/or distance of a device.

Reliability

	Classic	**BLE**
Transmission Power	20 dBm	20 dBm
Receiver Sensitivity	-70 dBm	-70 dBm (1/2 Mbps) -75 dBm (0.5 Mbps) -82 dBm (0.125 Mbps)

Table ___: Classic Bluetooth vs BLE Signal Quality

The transmission power is limited to 20 dBm (or 100mW) according to the standard set by SIG (initially IEEE). The minimum bit error rate (BER) is also set in the standard to no more than 0.1% loss at least -70 dBm. The receiver sensitivity is the minimum power the receiving device needs to receive to achieve the 99.9% correct data requirement. This means that a signal can attenuate from 20 dBm to no less than -70 dBm. This allows for a certain level of noise and/or distance to affect the signal before it is lost. BLE has the advantage of being able to meet the BER requirement at a much lower power of -82 dBm if data transmission of 125 Kbps is used. Another option exists for BLE for -75 dBm for transmission rate of 500 Kbps. So BLE allows for a trade off of high bandwidth / transmission rate for a higher receiving sensitivity. The table below lists the advantages of classic Bluetooth and Bluetooth Low Energy.

Classic Bluetooth	**BLE**
• Fast Transmission ○ Higher bits per second allows for faster transmission of data • Wide Bandwidth / High Quality ○ Higher bits per second allows for more data to be sent at which allows for higher quality • Simple Architecture ○ Few features allows for less exposure	• Higher Receiver Sensitivity ○ Ability to exchange transmission rate for higher sensitivity • Broadcast Feature ○ Send data to unlimited slave devices at once • Isochronous Feature ○ Data transmission between 1 master and 2 slaves

Table ___: Pros of Classic Bluetooth vs BLE

BLE has 3 advertising channels (out of 40) that are used by devices with BLE to constantly broadcast and receive presence signals. This is what allows BLE the ability to create a mesh network. The device is always available for another device to connect to. The problem is that it leads to vulnerabilities since any device can connect, there is no discrimination. This also means bandwidth is constantly consumed. Our project does not have any use for this mesh network feature, so this is why the simple architecture of classic Bluetooth is an advantage. The main draw to classic Bluetooth is the high transmission rate. The main draw to BLE for our project is the ability to use a much lower transmission rate in exchange for higher sensitivity. Another feature of BLE that may be useful for connecting sensors, is the broadcasting and multiple connection topologies. However, if we were to only use Bluetooth to connect from the ESP32 controller to the Raspberry Pi, the additional connection features of BLE are useless since we only need a point-to-point connection. On one hand, it may be more beneficial to use the higher transmission rate of classic Bluetooth because higher transmission rate translates to a higher bandwidth. This will lessen the likelihood of congestion. Congestion is bad because then buffers are needed which consume CPU bandwidth and could possibly lead to buffer overflow. This means data will need to be resent. On the other hand, BLE might be more useful because it can offer a higher sensitivity to nose and path loss in exchange for a lower transmission rate. So, if we end up needing to send a lot of data, classic Bluetooth would be more advantageous. If we end up not needing to send that much data, BLE would be more advantageous. This will mostly depend on the type of data that is sent and the quality/resolution requirements of the data.

3.3.2 Wi-Fi

The ESP32 module we chose, as do most ESP32s, supports 3 different Wi-Fi standards: 802.11b, 802.11g, and 802.11n. This is notated as 802.11b/g/n. Each standard represents a different generation of Wi-Fi. 802.11b is gen 2 (1999), 802.11g is gen 3 (2003), and 802.11n is gen 4 (2008). For reference, the most recent generation of Wi-Fi is 802.11be, gen 7 (2024). The table below shows the main features of each of these standards.

	802.11b	**802.11g**	**802.11n**
Data Rate (Mbps)	1 – 11	6 – 54	72 – 600
Frequency Band (GHz)	2.4	2.4	2.4/5

Table ___: Wi-Fi Specs Comparison

The ESP32-WROOM-32E however, offers up to 150 Mbps and only a center band pass of 2412 MHz to 2484 MHz. The Wi-Fi specifications are given again below, however this time, it only supported the options that are supported by the ESP32 chip we selected.

	Data Rate (Mbps)	**Frequency Band (GHz)**	**Standards**
WROOM-32E	1 – 150	2.412 – 2.484	802.11b/g/n

Table ___: Supported Wi-Fi Specs Comparison

Wi-Fi does not have standardized signal strength levels so the typical signal strength range is from -67 dBm to -70 dBm depending on the router but can be higher or lower. The reason there is not a standard is because there's no reason to limit the amplitude. This is because it is not physically possible to create a signal so high in amplitude that it becomes disruptive to other networks using the same band, while also supporting Wi-Fi. There is a tradeoff between signal strength and bandwidth which affects the number of devices supported, and/or quality (or throughput). The stronger the signal is, the more noisy the area becomes, and the more bandwidth is used meaning there is less bandwidth for others to use. Less bandwidth means less devices can connect, or devices will have less quality of signal, i.e., devices will have less throughput (bps). So, Wi-Fi routers are variable in terms of number of devices they can host, and how much throughput they can provide each device. The environment will also affect these deliverables. A larger area, an area with more obstructions, and more noise present will lessen the abilities of a Wi-Fi router.

Bluetooth and Wi-Fi

Bluetooth and Wi-Fi share the same frequency band. Both Bluetooth and Wi-Fi can operate in the same area, but each will eat each other's bandwidth when transmitting. Bluetooth has less range than Wi-Fi but takes up the entire frequency band to support one device. Devices connected to Wi-Fi take a designated frequency and time slot dictated by the router. This means that if a lot of devices are connected in a small area, Wi-Fi will have more ordered transmission and can assign priority since it has a connected network. Bluetooth is independent so no priorities can be assigned, and no ordered transmission can occur. Since Bluetooth is lower powered than Wi-Fi, if spread out, it can host more devices. If devices start to interfere with each other however, there is no way to collaborate to avoid this and the problem becomes like "the tragedy of the commons". There are so many complicated factors that are difficult to put numbers, and each of these factors affect each other in ways that make it impossible to plot, so no real hard numbers can be used for comparisons between Bluetooth and Wi-Fi. This is why generalizations are used to describe each. These generalizations were explained above and are graphed in the tables below.

	Bluetooth	BLE	Wi-Fi
Speed (Mbps)	1 / 2 / 3	0.125 / 0.5 / 1 / 2	1 to 150
RSSI (dBm)	-70	-82 / -75 / -70	-30 to -80 (Variable)

Table ___: Wi-Fi vs Bluetooth Speed and Power

Wi-Fi	Bluetooth
1. Network protocol	1. Data transmission protocol
2. Delegates bandwidth	2. Uses entire bandwidth
3. Higher transmission power and RSSI	3. Lower transmission power and RSSI
4. Non-disruptive protocol	4. Disruptive protocol
5. LAN (higher range)	5. PAN (lower range)

Table ___: Wi-Fi vs Bluetooth General Comparison

To summarize, Wi-Fi can support much more versatility in a larger area due to a centralized networking scheme. It can divide up bandwidth and do different optimizations to mitigate different scenarios. This includes assigning priority, assigning higher throughput, increasing the RSSI, or pausing communication to various devices on the network. One optimization is discussed later in this document and is the MQTT protocol. For this reason, Wi-Fi is supports faster throughput and is better in a crowded network. Bluetooth is low powered so it must operate in close proximity and with lower speeds. It mitigates noise by resending lost bits and using the entire frequency bandwidth. Bluetooth can operate in noisy environments but if too many devices are communicating using Bluetooth in close proximity, communication becomes drastically harder and may make communication infeasible. For our case, the micro controller could be used to divide time between different sensors communicating via Bluetooth.

3.4 Wired Communication Comparison

3.4.1 SPI

SPI (Serial Peripheral Interface) is considered a ‘de facto’ standard for synchronous serial communication in both embedded and real time systems. SPI uses master/slave architecture to perform device communication at short distance between integrated circuits. The communication it does provide is ‘full duplex’, meaning it can receive and transmit (hear and speak) at the same time. SPI’s master/slave architecture allows the system to organize data flow by selecting one device – the master – to control the data flow to the other devices – the slaves. The master device accomplishes this by generating a SCLK (serial clock) pulse to be the reference to all of the slave devices – similar to a drum major to a marching band. The slave devices cannot communicate to each other directly and can only communicate through the master device, however the slave devices can still send and receive data to and from the master device simultaneously per its full duplex capabilities.

The process of sending and receiving data from the master device and slave device(s) is performed in the signal lines of the integrated circuit(s). Using MOSI (Master Out Slave In) data lines and MISO (Master In Slave Out) data lines running separately but simultaneously together to transmit and receive data from each device – similar to a street’s architecture allowing traffic in one direction on one lane and traffic in the opposite direction in the other lane, so that all traffic may happen at the same time. The CS (chip select) allows the master device to choose and indicate which slave device is being addressed and interacted with for data transmission. Chip select works utilizing clock edges which in turn work in relation to pull up and pull down resistors. These resistors keep the input pin in a constant ‘natural’ state, whether high - a logical ‘1’ or ‘on’ - using a pull-up resistor to connect to a constant voltage source VCC – or low – a logical ‘0’ or off – using a pull down resistor. Pull up and pull down resistors keep a default voltage or state when not being interacted with to ensure predictable behavior and processing.

The advantages and disadvantages in SPI lie in its innate capabilities and qualities. Due to the nature of the master/slave architecture of SPI it is capable of operating at higher speeds relative to other communication protocols such as I2C since SPI uses a much simpler architecture and only uses a chip selector to initiate conversation between devices, operating at the capabilities of the master and slave devices which is often ranging in the Mbit/s (Megabits per second). Also due to SPI’s master/slave architecture is the aforementioned full-duplex communication, which other architectures such I2C are not capable of, operating at only half-duplex. Despite these advantages, SPI has a higher cost because of them. At the least, SPI requires four pins (MOSI, MISO, SCLK, and CS), and an additional CS line for each slave device. This potential of pins needed for SPI may exceed what we have available on our microcontroller board.

For our ball milling machine project our Senior Design group will use SPI to interface with the OLED and interact with our user-interface (UI) on the ball milling machine's front panel. We made this decision for SPI's simplicity regarding button presses and its simplicity to display and change images with those button presses.

3.4.2 UART

UART (Universal Asynchronous Receiver/Transmitter) is another serial communication protocol used to communicate with other microcontrollers. Being asynchronous, UART does not utilize a clock for sending and receiving data. Instead, the devices must agree on a baud rate (data rate) beforehand to communicate – similar to two radios needing to be on the same frequency. When the data is transmitted, it is packaged into a framework that includes start bits, data bits, parity bits, and stop bits. These parts of the framework serve necessary purposes to organize the data exchanged. The start and stop bits naturally signal the receiving device the beginning and the end of the data being sent - and the data bits (usually eight, but can vary) being the actual data transmitted. The parity bits in UART can be used to troubleshoot errors or monitor and log error rate during receiving and transmitting the data by checking for even or odd parity. The data bits are of course the desired/requested data to be transmitted and received being sandwiched between the start and end bits.

The capabilities of UART include being full duplex, being able to transmit and receive data at the same time. UART also has a customizable baud rate to adjust and communicate with other devices with data frames. UART only needs two lines for its communication - RX (receiving) and TX (transmitting) - and unlike SPI and I2C does not use a master/slave architecture for serial communication. Per its name, UART is universal and can be used in various devices across multiple platforms.

The advantages and disadvantages in UART are useful in specific scenarios. Being a universal serial communication architecture allows it to be used in any project and can be useful where other architecture can't be used or would be severely limited. Its full duplex capabilities with allows for faster communication and response to conditional logic. With only two wires – its RX wire and TX wire – the layout for UART would be more simple to design and implement. The parity bits implemented as a tool in UART make troubleshooting its simplistic design even that much easier. Despite these helpful and useful benefits, UART has its share of disadvantages. UART has a significantly shorter range of use compared to SPI or I2C being only designed for short range – typically only a few inches distance.

And, despite the parity bit troubleshooting, UART has no intrinsic or built-in error correction. UART is also a slower serial communication method as increasing the baud rate only increases the chances of errors occurring. Moreover, UART has no hardware addressing, which consequently means that all devices and peripherals connected to the UART bus hear the transmitted data and need to be configured to filter relevant data sent along the bus.

3.4.3 I2C

Inter-Integrated Circuit (I2C, pronounced "I squared C") is another major and widely used serial communication protocol used to send and transfer data between integrated circuits. I2C is a multi-master, multi-slave protocol that is primarily used for short-distance communication between various integrated circuits and peripherals on a printed circuit board (PCB). I2C is used in a multitude of applications including but not limited to: Embedded systems, sensor interfacing, real-time clocks (RTCs), display modules, and battery management.

The capabilities of I2C gives it a diverse list of uses and wide range of potential. I2C utilizes a multi-master multi-slave architecture primarily used for short distances. With multiple masters and multiple slaves, an I2C formatted integrated circuit can allow multiple devices (the masters) to initiate conversion IE send data. I2C is also capable of synchronous communication, allowing data to be transmitted in a clocked fashion. Moreover, I2C has two bidirectional data lines – Serial data (SDA) and Serial Clock (SCL). The serial data line is for carrying data while the serial clock line is for synchronizing the data. I2C is also capable of hot-swapping devices even while the bus is still hot, allowing for the addition and/or the removal of devices on the fly, which may prove to be one of its most helpful capabilities.

Like both SPI and UART, I2C has its share of advantages and disadvantages. Utilizing the master/slave architecture, I2C is relatively more simple to implement, and with only two wires/data lines I2C has efficient communication in terms of pin management. In addition to its simplistic architecture setup, the multi-master support allows for the capability to build and support complex and sophisticated projects with each master being able to initiate and independently control data flow. Despite these strengths, I2C, much like UART is limited in range being designed and oriented to short ranges – virtually only the length of the PCB – exclusively. I2C is also limited in the amount of devices it can support – at only seven bits I2C can support at max only 128 (2^7) devices per its addressing scheme. And, due to I2C's dependence on its clock for synchronization, I2C is susceptible to letting devices on its integrated circuit to fall out of synchronization if said devices on its network have different clock speeds/rates, allowing for potential problems to befall the project and will require troubleshooting.

3.4.4 Serial Communication Decisions

Both SPI & I2C

Our senior Design Group has opted to use both SPI and I2C. SPI will be used for the OLED and the user interface because the OLED responds better and is easier to program with an SPI standard given that the OLED is run mainly off of peripherals. Using SPI for this makes the incorporation of button presses much more efficient due to its efficient nature and duplex communication capabilities. As noted earlier, this efficient nature is due both to its master-slave structure. This structure allows the speedy transmission of data for receiving and transmitting, as the user interface will be interacted with constantly and will need to be fast responding for all of the button presses as the user scrolls through the user interface and for the user interface to be constantly updating the image its displaying.

3.5 Display Comparison

3.5.1 Display Technology Selection

We need a display that is capable of being read from a considerable distance and in various light conditions ranging from brightly lit to dark environments with relative ease. Given that there are many affordable small display technologies currently available on the market, we wanted to compare two types of commonly used at displays LED lit LCD and OLED. Beginning with LCDs (Liquid Crystal Display) which rely on a few interesting physical properties of the components used to make them to create the desired image, the first being able to polarize light, which involves orienting the electric vectors of light into the same plane or filtering the light that doesn't have the desired orientation this helps to reduce the glare and sharpen the image quality. The second being that liquid crystals can either allow light to pass or block it out depending on their orientation of the crystal and polarization of the light. The third property allows us to manipulate the orientation of the crystals by applying an electric current. Finally, what I would consider the most important property is that electrodes can be made transparent not only allowing them to conduct electricity but also allowing light to pass through them, making them ideal for optical applications. LCD screens are constructed in multiple layers, each taking advantage of one or more of these properties.

The bottom layer is an array of Light Emitting Diodes are extremely small semiconductor devices that generate light when holes and electrons are recombined. These LEDs can be arranged into individual pixels. Next is a Polarizing layer which utilizes a filter to polarize the light to a specific direction. The third layer is a substate that has groves oriented in the same direction as the previously mentioned layer with a transparent electrode. In the fourth layer is a Liquid crystal in a helical shape referred to as twisted nematic. Followed by another substrate layer with groves oriented perpendicular to the groves the first substrate with transparent electrodes in a desired shape such as a pixel. The final layer is another polarizing filter which has an offset orientation to the first filter by 90 degrees. When a current is applied to specific electrode the corresponding liquid crystals untwist changing the polarization of the light passing through them as a result the light is blocked by the filter. Some of the main advantages of LCD screens are their relative low cost, energy efficiency. A main disadvantage is its low contrast as blacks tend to appear grey.

Next we look at OLED (Organic Light Emitting Diode) which does not utilize conventional light emitting diode as we know them with a p-n structure but rather an organic electroluminescent diode. EL diodes are made from a thin film of organic material that emits its own light when an electric current is applied. Like traditional diodes doping is used to increase the efficiency of recombination but also to adjust the wavelength of the photons. The structure of the OLED display consists of the following layers in order a cathode layer which inserts electrons into the electron transport layer. The ETL (electron transport layer) aids the delivery of electrons to the emissive layer. Next, blocking layer restricts the electrons to the emissive layer. The most important layer is the emissive layer. This is the layer that converts the electrical energy into light of varying color. The hole transport layer helps to deliver holes to the emissive layer. To introduce holes an anode layer is used. Finally, a substrate which is generally made of glass, plastic or thin metal serves as the foundation. Note this is a simplified description modern OLED displays can vary, implementing a lot more layers in their designs. Advantages of OLED are lower power consumption, high contrast fuller color range with deep blacks. The main disadvantage is its higher cost in comparison to other display options. That being said, OLED displays have become increasingly affordable as technology improves making it an ideal choice for our desired application.

Our display will serve as the primary canvas for our user interface so we would like it to be as smooth as possible. It will need to display various sensory information upon request as well as display the current operating status of the mill. When considering displays the primary interface options were I2C, SPI and 8 Bit Parallel. Each has advantages and disadvantages for communicating between OLED displays and micro controllers. The simplest to implement seems to be I2C (Inter Integrated Circuit) due to its low pin count and universal standards. I2C can support multiple devices at ones such various sensors. The downside of I2C for a display application comes from its slow speeds, limited bandwidth and relatively short cable length. An additional consideration for I2C is the PCB real estate needed for the pull up resistors.

Another option and the most complex option would be 8-bit parallel. Which has the advantages of higher speed data transmission and performance. However, this increased speed comes at the cost of additional wires. Additional wires require additional considerations such as line spacing. The possibility of crosstalk increases for every line added. Any time you increase the number of physical components you increase the number of potential points of failure. Another disadvantage of parallel communication is the additional cost associated with it due to power consumption and components.

The final possible communication protocol for our display is SPI (Serial Peripheral Interface). This protocol offers significantly higher speeds than I2C. However, similar to parallel it requires more pins than I2C but less than the amount required for parallel. Additionally, SPI is more complex to implement than I2C again not as complicated as 8-bit parallel and much more supported than parallel. The average power consumption of SPI is the lowest of the three as well.

OLED Interface Protocols	**Pros**	**Cons**
I2C	• Simple to implement • Low pin count • Supports multiple devices	• Slow Speeds • Limited bandwidth
8-Bit Parallel	• High speed data rates • Low latency ideal for real-time applications.	• Complicated to implement • High Pin Consumption • Not as widely supported as I2C and SPI
SPI	• High speed data rates • Supports multiple devices • Commonly used across microcontrollers	• More complex than I2C to implement • More pins required(less than Parallel)

For our display we chose SPI interface protocol due to its increased data rate as compared to I2C which will provide a smoother display refresh rate. We also chose SPI since it requires fewer pin connections than 8-Bit parallel protocol which will allow us to connect additional sensors to the digital I/O pins of our microprocessor in the future.

3.5.2 Display Part Selection

We looked at 3 options in similar price ranges that had the features we desired. The first was Crystal Fontz CFAL12864K-Y display. This display features the standard 128x64 dot pixel resolution. The model we selected has an integrated SSD1309 common driver with controller. This display also has the thinnest profile measuring only 2.1 mm. Additionally, it uses a 31 pin Molex connection making it secure and unlikely to come loose in the high vibration environment.

Next, we looked at the New Haven Display NHD-2.7-12864WDY3-M which also has a 128 x 64 dot pixel resolution same as the Crystal fonts display. However, this display features a SSD1322 dot-matrix high-power OLED/PLED segment/common driver with controller. The New Haven display also features a smaller 20 pin Molex connector. Both share high contrast ratios of 10,000:1 and average brightness of 80 candela per square meter.

Finally, the last display was the Display Visions EA OLEDL128-6LGA. It is the most compact design having the smallest overall footprint when comparing the active screen area to overall size, although at a cost. This display only features a 102 x 64 dot pixel resolution and has a direct to PCB soldering connection type which limits placement. While it is the brightest of the three displays with a brightness of 100 candela per square meter it falls behind in contrast ratio. Having a contrast ratio of only 2,000:1 it has significantly lower contrast ratio than the other two displays.

While the New Haven Display and Crystal Fontz display were closely matched, we decided to go with the slightly less expensive New Haven display. The main reasons being a smaller Molex connector and double the lifetime rating. The New Haven display boasts a 100,000 hour life span while both the Crystal Fontz and Display Vision options only offer 50,000 hours

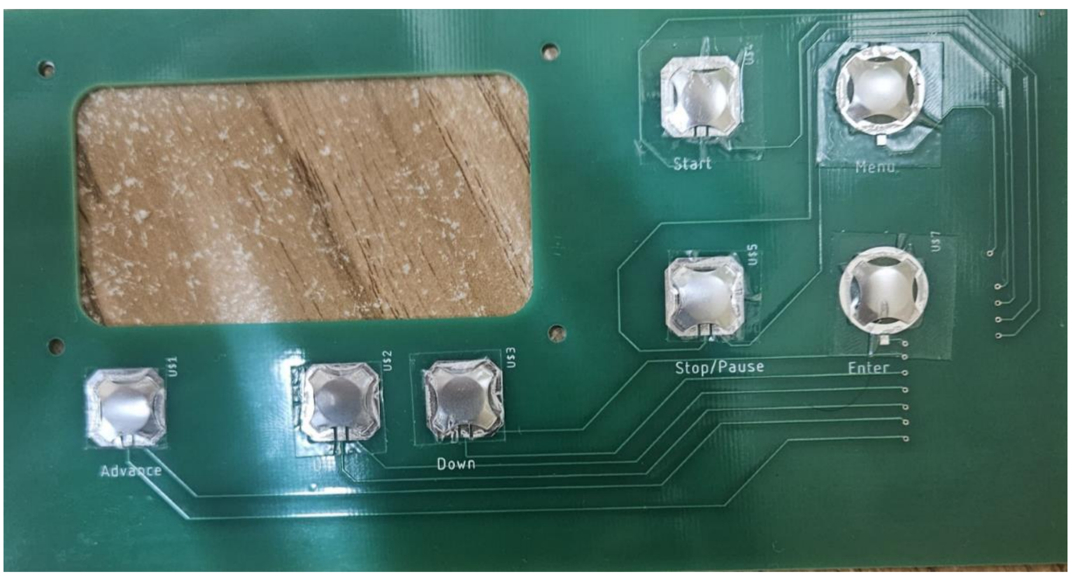
OLED display	Crystal Fontz CFAL12864K-Y	New Haven Display NHD-2.7-12864WDY3-M	Display Visions EA OLEDL128-6LGA
Resolution	128 x 64	128.64	102 x 64
Contrast Ratio	10,000:1	10,000:1	2,000:1
Brightness	80 cd/m2	80 cd/m2	100 cd/m2
Dimensions	73.0 (W) x 41.86 (H) x 2.15 (D)	82.0 (W) x 47.5 (H) x 6.4 (D)	39.0 (W) x 20.75 (H) x 2.6 (D)
Connector	Molex 31 pin connector	Molex 20 pin connector	20 pin header direct solder
Interface	8-Bit Parallel, SPI, I2C	8-Bit Parallel, SPI	SPI, I2C
Integrated Controller	Yes	Yes	Yes
Lifetime	50,000 hrs	100,000 hrs	50,000 hrs
Price	$38.85	$38.08	$30.42

Table #: OLED Display Comparison

3.6 Membrane Keyboard Technology

One of our Senior Design group's most fundamental goals with the ball milling machine project is to have it remotely operated and monitored on the HomeAssistant application for user convenience. To have our printed circuit board mounted behind the front panel and – most notably – inside the ball milling machine where the sensors can interact with the motor and measure the various status parameters we would be unable to transmit data outside of the ball milling machine. Since the ball milling machine is essentially a metal box of conductive material, per the physics the interior of the ball milling machine is a textbook example of a faraday cage, and will block electromagnetic fields, not allowing any of our signals outside of the ball milling machine to communicate with the HomeAssistant app.

To circumnavigate this dilemma, we are swapping the current metal front panel of the ball milling machine for a membrane keyboard that is not made of conductive material – which will essentially provide a hole for signals to escape the ball milling machine faraday cage and out to the server(s) to be displayed on the HomeAssistant app and from the HomeAssistant app back to the ball milling machine for its remote operation..



Above is a prototype membrane keyboard our Senior Design group will use for testing. Each of the buttons on the membrane keyboard will complete a circuit when pressed down like any other button. Uniquely thought out, however, is implementing a hole below the button to allow air to press out and to give a more enjoyable button press. The traces for the buttons are completed on the other side of the membrane keyboard through the vias.

Thinking ahead, in the top left our Senior Design group had accounted for and measured the OLED screen so that on it can be incorporated onto the membrane keyboard and screwed and secured onto it with the screw holes implemented around the screen cavity.

Currently, our Senior Design group will create our own membrane keyboard with our own design. The membrane keyboard shown above was design and produced by Dr. Blair as a proof of concept and so that our Senior Design group could use a membrane keyboard for prototyping.

3.7 Nonvolatile Storage Comparison

3.7.1 Nonvolatile Storage Technology Selection

One of the major goals of our project is that the mixer mill needs to successfully recover and continue operation after a power outage. In order to complete this goal, the system needs to have some type of non-volatile memory on board. We need to store the system state frequently during operation, and recover the state during startup if the system underwent an abrupt shutdown due to a power outage or fluctuation. This of course, requires that the memory is able to retain information without power, so the volatile RAM on the microcontroller will not be sufficient. Additionally, ensuring that a recent state is saved will require a significant number of read and write throughout the operation of the mixer mill. This mill is used frequently, and for long periods of time, so we need to make sure that the non-volatile memory technology we choose does not degrade quickly. We are considering several different technologies to achieve these goals, including flash, FRAM, EEPROM, and Toggle MRAM.

Flash memory uses MOSFETS with floating gates and has two common types: NAND flash, and NOR flash [15]. As one might expect, given the names of these technologies, NAND flash uses MOSFETs in an arrangement that operates like NAND gates, while NOR flash uses MOSFETs in a configuration that behaves similarly to NOR gates. For NAND flash, the bit line is brought to ground, only when all bits of the word line are high. For NOR flash, the bit line is low when any of the word lines are high. This is accomplished by connecting the cells in NAND flash to the bit line in series, while the cells in NOR flash are connected to the bit line in parallel. The parallel connection allows any cell in NOR flash to be accessed individually, while NAND flash has to be read at a block level. For reads, the individual accessing makes NOR flash faster than NAND flash, but the ability to erase bits at a block level makes NAND flash faster to write and erase. NOR cells are more expensive and physically larger than NAND cells, with NAND flash cells consuming around 40% of the area that NAND cells consume [15]. However, NOR flash is more reliable than NAND flash, with better data retention and fewer bit-flips. Table 3.7.1.1 summarizes the differences between NOR flash and NAND flash. For our project, with only extremely low-scale manufacturing of our new mixer mill controller, the reliability of the memory is far more important than a small difference in size or cost. If we decide to use flash, we would therefore prefer NOR flash to NAND flash. Additionally, NOR flash is the type that is often included on-chip in microcontrollers such as the ESP32.

	NOR Flash	**NAND Flash**
Area	Higher	Lower
Cost	Higher	Lower
Read Speed	Faster	Slower
Write Speed	Slower	Faster
Bit Flips	Less Frequent	More Frequent
Data Retention	Longer	Shorter

Table 3.7.1.1: NOR flash vs NAND flash

The next technology we considered is Ferroelectric RAM (FRAM). FRAM cells are dipole capacitors that use a ferroelectric crystal between the plates [16]. The crystal has two stable states that are used to store the bit. Since these states are stable, they are maintained when power is removed, making FRAM a type of non-volatile memory. Reading the state of this crystal requires applying an electric field that changes the state of the crystal, so every time that a bit is read, the data needs to be written back to the FRAM. The most important properties of FRAM for our use case are the data retention and endurance of the memory. FRAM far exceeds flash in these areas. Data retention in FRAM is reliable for a period of 10 years at 85 degrees Celsius, and can be much longer if the memory chip is kept at a lower temperature [16]. Additionally, the endurance of FRAM is around 10 orders of magnitude higher than flash memory, with FRAM chips being reliable for around 10^{15} write cycles, against the approximately 10^{5} write cycles that flash memory is able to endure [16].

One difference here is that since FRAM requires a write after every read this endurance is limited by both reads and writes, while flash memory has non-destructive reads, and can be read from an essentially indefinite amount of times. However, we anticipate our application having a large number of writes, since we want to ensure that our mixer mill controller is able to recover from disruptions with accurate and recent state information. We do not want to have these disruptions affect the amount of time that a sample is milled for. Additionally, the 10^{15} read/write cycles supported by FRAM is a large enough number, that FRAM has effectively unlimited cycles for either operation. We therefore strongly prefer FRAM over flash for our use case. The largest downside of FRAM is that it is less commonly included on-chip in various microcontroller architectures. If we decide to use FRAM, we will need to purchase a separate module that we will integrate on our PCB.

Another type of memory technology that is an option for a system that requires high reliability and write endurance is toggle magnetoresistive random access memory (MRAM). In toggle MRAM, the data is stored in a magnetic state, rather than being based on the energy state of a crystal as in FRAM, or stored in a floating gate MOSFET like in flash. Each cell in toggle MRAM storage includes one transistor, and one magnetic tunnel junction [17]. The magnetic tunnel junction is the element that actually stores the information. A bit is stored as the current state of the orientation of the magnetic moment of the free layer. When the moment is parallel to the ferromagnetic layer the magnetic tunnel junction has low resistance, and when the magnetic moment is perpendicular to the ferromagnetic layer the magnetic tunnel junction has high resistance. Reading the resistance of the magnetic tunnel junction is a non-destructive operation, so unlike in FRAM reads do not require a write-back and do not impact the lifetime of the device. Additionally, the write endurance of toggle MRAM is above 10^{14} cycles, significantly larger than flash, and only one order of magnitude short of FRAM [17].

A write endurance of 10^{14} cycles is large enough to be effectively unlimited; we will not get anywhere near reaching it during the lifetime of the mixer mill. If we wrote state information to memory once every second, it would take us 3.171 million years to reach 10^{14} writes. The primary downside to MRAM is the cost. It is more expensive than flash memory or FRAM. Since we do not need a large amount of non-volatile memory, the increased cost is not a major concern, but it is still a consideration. Another minor factor is the read and write speeds of the memory. MRAM is quite a bit faster than FRAM, which was already faster than flash. Since we are not storing massive amounts of data, the difference in speed would be relatively insignificant, and would certainly be small enough that it will not make a noticeable difference to the user.

Table *3.7.1.2* shows a comparison of the non-volatile memory technologies we have considered. One aspect that we left off of the table was the speed of the memory technologies. While we mentioned that MRAM is the fastest, we would not be able to practically take advantage of the increased memory speeds. We will likely be using a memory module that has an SPI interface. When using a module like that, the read and write speeds are limited by the clock frequency of the interface, rather than by the memory technology. In practical application, flash memory would be the fastest, because several of the microcontroller options we are considering include flash memory in the package which would prevent the additional delay of going off-chip. We do not have the same ability to purchase microcontrollers that include the other features we are looking for in addition to FRAM or MRAM on-chip.

Since speed is not a factor in our decision, it primarily comes down to cost and reliability. We have considered two major components of reliability, data retention and write endurance. Of these two factors, write endurance is more important for our application. The ball milling machines in Dr. Blair's laboratory are used extremely frequently. Any state information that we are intending to save on the machine will be read and overwritten the next time the machine is turned on. Therefore, any of the technologies we considered will be more than sufficient. The shortest data retention is over ten years, and Dr. Blair's machines will not be sitting for anywhere near that long. The write endurance varies more dramatically between these technologies. NOR Flash has a write endurance of approximately 10^5 cycles. We will burn through these cycles relatively quickly. If we save state data once per second, it will take us just under 28 hours of continuous operation to complete 10^5 writes.

We will not actually burn through the chip that quickly, because we would not write to the same addresses each time. With a more thoughtful scheme that rotates through all of the memory addresses, we can get a reasonable lifetime from the flash chips. While we haven't finalized our state data format yet, a reasonable starting point is a 16-bit number measuring milling time per cycle in seconds, a 16-bit number measuring break time per cycle in seconds, an 8 bit number representing the number of cycles left along and the current cycle type, and another 16-bit counter for the current time. Our baseline state information requires 56 bits. If our scheme evenly uses every address in the memory, it should take around 15 years of operation to ruin every memory cell in the 256 kilobit flash. We could easily increase this time further with a larger flash memory, however we likely will not perfectly wear out the memory, and can afford to pay a few dollars to ensure the longevity of the memory system. Since one write per second will not reach 10^{15} writes for 31 million years, we decided that an FRAM module is worth spending a few extra dollars for the increased reliability of the product. Even if the module we purchase is an outlier with shorter than normal endurance, an FRAM module should never need to be replaced. A flash module with shorter than usual write endurance could conceivably need replacement within the lifetime of the ball milling machine. This especially holds true since your final state saving scheme will likely use more than the 56 bits, and will probably not rotate memory addresses every write. Even if we do change the address that the state information is saved to on every single write, that would require an additional write somewhere to update the pointer to the state information, which would increase the speed that the flash wears out. All this to say, flash memory will probably have sufficient reliability, but since there is a slim possibility that it will not, it is not a significant hurdle to spend a little bit of money on better memory technology.

The differences between FRAM and toggle MRAM are less important. Both technologies have much better data retention and significantly higher write endurance than we require. We therefore prefer FRAM over toggle MRAM for its lower cost. Additionally, FRAM is more commonly found in even smaller modules than 256 kilobits compared to toggle MRAM, which can reduce the cost of the module further. Since the ball milling machine *probably* is not going to run for 31 million years, even a 128 bit FRAM should be sufficient to store state information, if FRAM modules came in that size. We will likely choose something more along the lines of 16 kilobits, because we will use some of that additional memory to store information like the lifetime health monitoring statistics, including average motor temperature, average humidity, motor runtime, and accelerometer data that tracks the level of vibration the system has been subjected to.

	NOR Flash [15][16]	**FRAM [16]**	**Toggle MRAM [17][18]**
Cost for a 256 Kbit Module	~ $0.25	~ $4	~ $6
Data Retention	20+ years	10+ years	20+ Years
Write Endurance (Cycles)	10^5	10^{15} (reads also incur writes)	10^{14}

Table 3.7.1.2: Non-Volatile Memory Technology Comparison

3.7.2 Nonvolatile Storage Part Selection

When it comes to selecting an FRAM module, there are a few different factors that are important for this project. Firstly, we considered the size of the FRAM memory. FRAM modules are measured in bits, and are easily available in memories from as small as four kilobits to as large as sixteen megabits. For our application, even four kilobits should be enough, but it would not be a bad idea to purchase a larger memory in case we want to log a greater amount of data. The number one reason we need an FRAM module is to store state information and be able to successfully recover from power outages. This requires a very small amount of memory. We also intend to store health monitoring statistics, such as average motor temperature, average temperature, average humidity, and total motor runtime. Storing averages doesn't require a massive amount of memory, but we want to leave the possibility of storing more than just averages available.

With a larger memory we could store a large number of measurements of these different data points. That would allow for the humidity, temperature, and other factors to be plotted over time. This could be useful if something is wrong with the sample. Dr. Blair would be able to plot the environmental data and see if something happened, like maybe the humidity in the math and science building increased dramatically due to a temporary issue with the building's air conditioning. Or maybe something is wrong with the ball milling machine, and the accelerometers are recording a greater magnitude of vibration than is typical. Recording enough data to detect events such as these will require a larger memory. We decided to look for an FRAM module with at least 128 kilobits of storage available.

The four options we considered are shown in table 3.7.2.1. Since we decided that 128 kilobits is sufficient, the most notable variations between these four options are the interfaces and the footprints of the parts. The interface is important because SPI and I^2C have a very different impact on our GPIO pin utilization, as discussed in the communication method technology comparison section. The I^2C device can be added on to a bus that will already exist for several of our sensors. The SPI devices would require a separate interface with their own GPIO pins. However, the SPI devices are significantly faster than the I^2C device due to the lower reliability of the I^2C bus at high speeds. Deciding which style of communication we want to use is largely based on how much data we intend to write. If we were using this as our main memory, or performing more memory intensive tasks, we could potentially have problems with the slower speeds of I^2C. Since we will be frequently writing the extremely small state data structs, and writes of sensor data can be a bit more limited, we decided that the increased speeds of SPI are not worth it. We are better off with the smaller pin utilization of I^2C.

The footprint type is the second major factor in our decision. Three of the four options, the Fujitsu Semiconductor MB85RS128TYPNF-GS-BCERE1, the Fujitsu Semiconductor MB85RC256V, and the Infineon Technologies FM24V01A-GT have essentially the same footprint. The 8-SOP and 8-SOIC packages are slightly different, but it’s extremely marginal. They are just the two different manufacturers versions of the same package. This package is a plastic casing for the integrated circuit, which is attached to four pins on either side. The pins bend down, and then lie flat on the PCB. The pitch and pin lengths for 8-SOP and 8-SOIC are not centered at the same number, but their minimum and maximum values are only slightly different. The most significant difference is that the Infineon 8-SOIC package can have the plastic casing sit slightly further above the pins than the Fujitsu package. The third option of 8-WLP is a more substantially different package. WLP is an extremely compact package with 8 small solder balls beneath the chip for mounting. This package is what allows Fujitsu to put so much more memory in a smaller area than the other two options we considered. While this package is really cool, it would be incredibly difficult to work with. It also would not allow us to probe the pins with a logic analyzer to see what is being written to memory if we’re having trouble debugging the hardware.

Due to its more convenient package and decreased pin usage due to its usage of I^2C, we decided to use the Fujitsu Semiconductor MB85RC256V FRAM module. This module provides a simple interface with more than sufficient memory for our intended application.

	Fujitsu MB85RS128TYPNF-GS-BCERE1 [24]	**Infineon FM24V01A-GT [25]**	**Fujitsu MB85RS1MTPW-G-APEWE1 [26]**	**Fujitsu MB85RC256V [42]**
Size	128 Kbit	128 Kbit	1 Mbit	256 Kbit

Memory Interface Type	SPI	I^2C	SPI	I^2C
Maximum Interface Clock	33 MHz	3.4 MHz	40 MHz	1 MHz
Part Dimensions	5.84 x 5.30 x 1.50 mm	4.978 x 5.842 x 1.727 mm	3.092 x 2.28 x 0.25 mm	6.00 x 5.05 x 1.75 mm
Board Area Consumed	30.952 mm^2	29.081 mm^2	7.050 mm^2	30.300 mm^2
Footprint Type	8-SOP	8-SOIC	8-WLP	8-SOP
Current Cost from Digikey	$3.36	$3.70	$5.97	$4.63

Table 3.7.2.1: FRAM Module Comparison

3.8 Relay Comparison

3.8.1 Relay Technology Selection

We need to switch a 115 VAC supply that will be used to control a second relay that drives the ball milling machine's motor. The second relay is already in use and is mounted to the chassis of the machine. This relay has a coil current between 17 and 20.4 mA with a 115 VAC supply, so we need a part that is able to supply this current [53]. We considered three different technologies to accomplish this task: Electromechanical relays, reed relays, and solid state relays.

Electromechanical relays are a popular choice for high power applications. They use a current through a coil to create a magnetic field that closes contacts through the armature. This approach, combined with strong insulation, provides superb isolation between the circuit controlling the relay and the power being provided through the relay. These types of relays also tend to have large contacts that are able to survive large amounts of power. Unfortunately, large contacts cause the overall relay package to also take up a larger amount of space, so this type of relay is not necessarily the superior choice for a relay that needs to be surface mounted on a printed circuit board in a limited space. Additionally, electromechanical relays are very slow to switch when compared to the other technologies, often taking up to 15ms to settle [54]. This is one of the trade-offs caused by the extra isolation between the two circuits. In our case, switching speed is not a huge concern. The milling machine will be running for periods on the order of minutes or hours, so a few millisecond delay is not an appreciable problem. The larger concern for our application is the lifetime of the relay. One potential problem with this technology is that electromechanical relays have a shorter mechanical lifetime than our other options. The electrical lifetime of the relay is, however, much longer than its competing technologies. For our use case, the electrical lifetime of the relay is a more significant factor. We will be switching the relay relatively infrequently, as the motor typically runs in thirty minute increments.

Our second technology operates on a very similar concept to the electromechanical relays we just discussed. Reed relays work through physical contacts that are moved with a magnetic field in order to create or destroy an electrical path. The big difference between reed relays and electromechanical relays is the size and movement mechanism of the physical contact. Reed relays have small contacts that are both attracted to each other under a magnetic field to form the connection. This double movement and smaller physical size means that reed relays are able to switch much faster than the standard electromechanical relays. The smaller size is an additional plus for our use case, where we intend to mount the relay on a circuit board. Reed relays also have a longer physical lifetime than electromechanical relays. The average reed relay is able to switch around 10 times more than an electromechanical relay prior to failure [54].

These properties do, however, come with some disadvantages. The primary disadvantage is in survivability of the relay when any sort of electrical glitch occurs. Reed relays arc more easily than electromechanical relays, which creates more of a potential for this kind of relay to weld itself closed. This is not a good thing if your relay is powering a motor. If this happened in our project, it would remove our ability to control the milling machine motor through our microcontroller, and would require that someone goes to physically unplug the machine in order to turn it off. With the current sensor that we will be putting on the machine, we would be able to detect this fault and alert the user through HomeAssistant, but we wouldn't be able to do anything to stop it automatically.

The final technology we considered was solid state relays. Solid state relays are, as the name suggests, solid state, with no moving parts. These relays are built around photo-sensitive MOSFETs that can be controlled with an LED. This provides the same high degree of isolation between the control circuitry and load that we found in the two other types of relays. The solid state relay comes with the benefit of high life expectancy, nothing is moving, so there are no moving parts to break. An additional benefit of having no moving parts is that switching time is much faster, these relays can switch in periods of under one millisecond [54]. These relays also have significant downsides over the older two styles. The first downside is that solid state relays easier to break than electromechanical relays. They do not have contacts to weld shut like reed relays, but the internal circuitry can be damaged. Solid state relays also have a larger amount of resistance when they are conducting. Electromechanical relays and reed relays both close a contact that is used to form an electrical path, so the resistance in one of these relays can be similar to the resistance in the wire that is leading up to the relay. In solid state relays, the current needs to travel through a MOSFET, leading to resistances on the order of 100 ohms [54].

The advantages and disadvantages of the three types of relays are summarized in table 3.8.1.1 below. For our project, we chose to go with a reed relay. While we do not need the fast switching speeds of the reed relay, we place a high value on their relatively small size. Solid state relays also could have easily fit on our PCB, but they introduce a resistance in the conducting path that isn't there in the existing design, and could cause a greater amount of power dissipation in our board. This factor, plus the increased cost of purchasing a solid state relay, led us to decide on a reed relay.

	Electromechanical Relays	**Reed Relays**	**Solid State Relays**
Size	Largest	Middle	Smallest
Isolation	Excellent	Excellent	Excellent
Switching Speed	Slowest	Middle	Best
Mechanical Durability	Worst	Middle	Best
Electrical Durability	Best	Middle	Worst

Table 3.8.1.1: Relay Technology Comparison

3.8.2 Relay Part Selection

Once we selected the relay technology, it was time to look at specific relay part options. We want a relay that has a coil voltage of either 3.3V or 5V, and is able to carry 20.4 mA of 115 VAC. Regardless of the voltage selected, we can power the coil in the relay using a MOSFET between the relay's inductive load and ground. This will help prevent excessive current draw stemming from the general purpose input/output pins on the ESP32 microcontroller.

We looked at four different relay options. The first is the Cocto Technologies 8L41-05 relay. This is the relay that is on the existing ball milling machine controller. It uses a five volt coil to switch loads of up to 100 VAC with currents of up to 0.25A. We noticed that this relay is actually being used outside of its specifications in the existing board. It is currently carrying a 115 VAC source. We suspect that the board designer believed that this was okay because the current running through this relay is relatively low. The only thing that this relay needs to power is the coil of a second relay – which is the one that is used to actually provide power to the ball milling machine's motor. As we mentioned earlier, the coil on the larger relay has a maximum coil current of 20.04 mA, so the Cocto 8L41-05 relay on the controller board only needs to carry under a thousandth of its maximum current rating. Otherwise, the main benefit of this part is that we know it works, since it is being used right now.

For our design, we would like to make sure that all of our parts are being used within their specifications, even if other parts have proven functional outside of their strictly in-spec behavior. To keep it similar to the existing part, we looked at the other options in the Cocto 8L catalogue. They have a part called the Cocto 8L02 that is very similar to the 8L41, except that it is specified for voltages up to 200 VAC, and for currents up to 0.5 A. It also has a higher rated insulation resistance than the 8L41. Outside of the specifications, the primary functionality difference between the Cocto 8L41 and 8L02 reed relays is the configuration of the pin connections. The 8L41 relay has a single-pole double-throw configuration. You could use it to select between two different sources for the bottom two pins, which are tied together. The 8L02 relay, on the other hand, has two single-pole single-throw connections in parallel. You can use the relay to connect two different loads to two different sources. For us, this difference is not an important factor. We really just need this relay to either connect 115 VAC to the coil of the larger relay, or to not connect the 115 VAC source to the coil. With either configuration, we only need to use one of the input pins, and one of the output pins.

We additionally wanted to look at some options outside of the Cocto lineup, so we examined the Littlefuse HE721C05 relay. It has very similar specifications to the Cocto relays. It has a smaller switching voltage than the Cocto 8L02, but it is higher than the 8L41, and is above the requirement for our application. The coil current and voltage are identical to the Cocto offerings, and the size and switching current are both very similar. This part seems to be a good option, but since we know that Cocto parts have proven reliable in the existing ball milling machines, we do not believe that this part is worth the change just to save about 4.8 square millimeters of board area.

The final part we considered is the Standex-Meder SIL03-1A72-71D. This relay is interesting because it has a 3V coil voltage, and a 500 ohm coil resistance, so it only requires six milliamps to switch. These specifications mean that we could potentially drive this relay directly from one of the ESP32 general purpose input/output pins. This reed relay also offers an overkill switching voltage of 200 VAC with a current of 0.5 A, matching the Cocto 8L02. It has the additional benefits of having the best insulation resistance of the four relays we considered, and the smallest size. We believe that the primary reason for the reduced size is the simplicity of the relay connections. One of the Cocto and the Littlefuse relays are single-pole double-throw, and the other Cocto relay is two single-pole single-throw switches in parallel. The Standex-Meder relay is functionally just a single single-pole single-throw switch. For our application, that's all we really need.

A summary of the relay specifications is shown in table 3.8.2.1. We decided to go with the Cocto 8L02-05 relay. While we do not need the second single-pole single-throw switch on the device, it is an overall safe choice. We know that the 8L41-05 has worked well for some time, and it does not hurt to go for the higher voltage rated, cheaper version of the same relay. We could save some board area with the Standex-Meder relay, especially since it does not require a MOSFET to drive it, but we decided it is worth it to go with the tried and tested brand. Additionally, while we can drive the Standex-Meder option directly from the general purpose input/output, it does not hurt to reduce current load from the ESP32 by driving the relay separately from our 5V source. Another factor that we considered was the isolation of the relay from the ESP32, we want to reduce the chances of any AC coupling making its way through the system. The Standex-Meder relay has a higher insulation resistance, but would be driven directly by the ESP32 pins. Using a MOSFET in between the microcontroller and the relay provides an extra degree of isolation between the ESP32 and the high voltage power source.

	Cocto 8L41-05 [57]	**Cocto 8L02-05 [57]**	**Littlefuse HE721C05 [58]**	**Standex-Meder SIL03-1A72-71D [59]**
Coil Voltage	5 V	5 V	5 V	3V
Coil Current	25 mA	25 mA	25 mA	6 mA
Coil Resistance	200 ohms	200 ohms	200 ohms	500 Ohms
Max Switching Voltage	100 VAC	200 VAC	120 VAC	200 VAC
Max Switching Current	0.25 A	0.5 A	0.35 A	0.5 A
Insulation Resistance	10^9 ohms	10^{10} ohms	10^{10} ohms	$5*10^{12}$ ohms
Part Dimensions	19.68 x 7.62 x 6.60 mm	19.68 x 7.62 x 6.60 mm	19.05 x 7.62 x 5.5 mm	19.8 x 5.08 x 7.8 mm
Board Area Consumed	149.96 mm^2	149.96 mm^2	145.16 mm^2	100.58 mm^2
Current Cost from Digikey	$10.21	$4.48	$4.65	$4.48

Table 3.8.2.1: Reed Relay Part Comparison

3.9 Current Sensor Comparison

3.9.1 Current Sensor Technology Selection

We need a current sensor that can be easily integrated into the power supply of the mill. We would like to monitor the power consumption of the mill. We would like to monitor Prolonged spikes or drops in the current. The mill operates at 1060 cycles per second with a 115V/60Hz power supply and the full load amperage should be less than 8 amps. There are multiple sensor options available to detect current. We decided to two at two of the least invasive options available a current transformer and a Hall effect sensor.

We begin with the current transformer sensor which operates like a traditional transformer in that it relies on an alternating current passing through the center of a magnetic core. This creates an alternating magnetic field in the core that in turn induces a current in a secondary winding proportional to the ratio of turns between the primary and secondary windings. In the case of a CT sensor the primary winding is the wire that passes through the middle of the magnetic core. The secondary is wrapped around the magnetic core. This acts as a step-down transformer as it lowers the current and generates a voltage typically between .333V and 1V. The cores can either be split or solid. The split core has the advantage of being easier to install while being less accurate than a solid core CT sensor. Solid core CT sensors are typically smaller and more accurate and may require you to cut your primary wire to pass it through.

The material used to make the core can impact size, phase shift, and accuracy. Typically, CT sensor cores are either FeSi (Silicon Iron), FeNi (Nickel Iron) or Ferrite. FeSi cores are cheap but offer inferior performance in the areas of linearity and phase shift. They are better suited for high current applications that do not require high precision. Additionally, their size and weight make them difficult to use in limited space environments.

FeNi cores offer much more accuracy than FeSi, making them more suitable for lower current applications. As can be expected the increase in linearity and phase performance comes at a cost, they are much higher in cost than the FeSi and ferrite CT sensors. Similar to the FeSi cores they are larger than ferrite core sensors.

Recent improvements in ferrite cores have increased their permeability, making it suitable to operate at low frequencies in the 60Hz range. These sensors have good accuracy and linearity at lower currents. Additionally, these sensors have low phase shift. They are much smaller than the other two cores as well and cheaper. However, they are not suitable for high current applications as they are limited in size by manufacturing capabilities.

Core Material	FeSi	FeNi	Ferrite
Accuracy	Low	Higher	Good/high
Cost	Low	High	Low
Current Suitability	High Currents	High Currents	Low Currents
Linearity	Poor	Good	Decent
Phase Shift	Large	Low	Low
Size	Large	Large	Small
Weight	Heavy	Heavy	Light

Table #: Pros and Cons of Core Materials FeSi Vs FeNi Vs Ferrite

Hall effect current sensors are another valid option for our needs as they are compact in size and can offer similar performance. The sensors rely on the basic principles of the Hall effect. When a current flowing through a conductive material experiences a magnetic field perpendicular to its current flow, a Lorentz force is generated. This force pushes flowing electrons to one side of the semiconductor. The collection of electrons that build on one side of the semiconductor, create a negatively charged end. Consequently, an opposing positive charge is created on the opposite end and a secondary potential difference is formed. This potential difference is referred to as the Hall voltage. The strength of the Hall voltage is proportional to the strength of the magnetic field that induced it.

Hall current sensor takes advantage of this phenomena by passing a steady current through a thin semiconductor material. When it is exposed to a magnetic field that is proportional to the current of interest, we can use the resulting hall voltage to determine the strength of the current. There are generally two approaches in hall current sensors open and closed loop configurations.
In the open loop configuration, the hall sensor sits in the gap of a magnetic core. When a current passes through a wire (or conductive material) that is running through the center of the core, a magnetic field is generated. This field is proportional to the current in the wire and is concentrated by the magnetic core. This concentrated field is measured by the hall sensor passed through an amplifier to generate an output. These sensors are more susceptible to saturation and temperature drift.

The closed configuration is set up identically to the open loop configuration with a few additions. In this configuration the hall sensor feeds back a current to a secondary winding around the core that zeroes the flux produced by the field of the primary current. The benefit of this configuration is that it reduces drift and improves accuracy. Furthermore, the secondary winding behaves as a current transformer at high frequencies increasing the bandwidth.

Open loop Hall sensors are cheaper than closed loop but suffer from drift and low accuracy better suited for higher currents. Although closed loop circuits are slightly more expensive, they boast a significant improvement in accuracy and have their drift reduced to a tolerable minimum. Closed loop sensors also perform well in noisy environments which is something we need to account for in our application.

Hall Current Sensor	Open Loop	Closed Loop
Accuracy	Low	Higher
Cost	Low	High
Noise Tolerance	Low	High
Installation Ease	Difficult	Difficult
AC Current Precision	Low	Good
Linearity	Poor	Good
Size	Small	Small

Table #: Pros Cons Open Vs Close Loop Hall

We need our current sensor to measure and AC current between 0 and 10 Amps in a relatively noisy environment. So, our best options would be a split core CT sensor or a closed loop Hall sensor. Both options offer high accuracy and can be incorporated without much need or any need to modify the circuit. We would also like to be able to easily install this sensor with minimum invasiveness. Many Hall sensors need to be physically wired into the circuit making them slightly more challenging to implement. Although, some are available in a passthrough configuration with a magnetic core similar to a CT sensor. These pass-through type of Hall sensors are significantly more expensive than CT sensors, making them less attractive.

Current Sensor Type	Split Core CT Sensor	Closed Loop Hall Sensor
Accuracy	High	Higher
Cost	Low	Higher
Noise tolerance	High	Higher
Installation Ease	Easy	Difficult
AC current precision	High	Good
Linearity	Good	Good
Size	Larger	Smaller

Table #: Pros and Cons CT Vs Hall

While the Hall sensor boasts higher noise tolerance, better linearity, and drift values. The cost and complexity of installation make it less appealing than the CT sensor. The main purpose of the sensor is to measure an AC current in a noisy environment. Given that the CT sensor is better at AC current measurements and is easier to install. Additionally, this type of sensor is available with a 3.5mm jack. This will make swapping out the sensor in the case of a failure as easy as plugging in a pair of headphones. We are also afforded more freedom in the placement of the components required for this type of sensor. This is because we will not be required to hard wire it into our circuit

3.9.2 Current Sensor Part Selection

When looking at CT current sensors, our primary points of interest are accuracy, ease of installation/Invasiveness and operating ranges. The AC motor we are using has a Full Load rating of about 7amps. The mill itself is equipped with 8-amp slow burn fuses to protect the system, while simultaneously allowing for short term current spikes during initial motor startup. Given these operating specifications for the mill, a CT sensor that is rated for 10-amps should be enough with room to spare.

Three options that were available in a split core 10-amp package were the SCT-0400-010 by Magnelab, SCT013-010 by YHDC and finally the CR9580-10-M by CR Magnetics. These sensors are all rated to support 60 Hz low frequency operations at 115V. Each of these come in slightly different configurations requiring specific alterations to the PCB circuit design. The first CT Censor we'll look at is the CR9580-10-M.

The CR9580-10-M has the highest accuracy rating of the three, being ±0.5%. It is rated for an output of 0-5 Vdc. Since the ESP32's ADC pin is only rated for Max of 3.3 Volts the voltage will need to be lowered through a voltage divider circuit. Additional measures may need to be taken to limit the maximum voltage allowed to pass to the ADC pin to prevent damage. Additionally, this sensor requires 1 megaohms of Burden impedance for accuracy so an additional resistor is required as well.

Another benefit of this sensor is that it has built in average sensing that outputs the RMS of the signal. This comes with a greater overall size compared to the other two options by quite a bit. Another downside is the cost of this sensor being almost twice that of the SCT-0400-010 and 7 times that of the SCT013-10.
Next, we'll look at the SCT013-010 current sensor. This sensor offers a slightly less accurate option but at a considerably reduced price. This sensor has an accuracy of ±1% which is still very accurate and more than enough to suit our needs. It comes with a built-in 3.5mm output making it easily replaceable.
Since its rated output voltage is only 1 V a voltage divider isn't necessary reducing the overall additional components needed on the PCB. This is a voltage type sensor, so it has a built-in burden resistor reducing the need for another component. Its overall footprint is significantly smaller than the CR9580-10-M, it requires less components on the PCB and has a smaller package size.
An even smaller option than the SCT013-010 is the SCT-0400-010. This sensor has an accuracy of ±1% as well, making it a viable option. This sensor has the

lowest output voltage of the three sensors, being only .333 V. The low voltage reduces the required components similarly to the SCT013-010. The Cost is of this sensor falls between both sensors.

Sensor	CR9580-10-M	SCT013-010	SCT-0400-010
Accuracy	±0.5%	±1%	±1%
Frequency Range	50 – 400Hz	50-1000 Hz	50-1000 Hz
Output Voltage	0-5 V (DC)	1 V (AC)	0.333 V (AC)
Rated/Max input	10A/24A	10A/35A	10A/12A
Burden impedance	1 Mega-ohm	Built in	Built in
Dimensions	49.57mm(W) x 25.42mm(D) x 50.84mm(H)	32.00mm(W) x 22.00mm(D) x 57.00mm(H)	24.00mm(W) x 27.00mm(D) x 40.00mm(H)
Opening Size	10.17mm	13mm	10.0mm
Cost	$50.39	$6.90	$26.00

Table #: CT Current Sensor Comparison

While the CR9580-10-M is very accurate the required additional cost from the sensor itself and the additional components required for voltage dividers and voltage protection it doesn't make it very appealing or necessary for our desired applications. When choosing between the SCT013-010 and the SCT- 0400-010 their accuracies are identical while the latter has the smaller overall size though not by much. The low output voltage of the SCT-0400-010 means it has an overall lower resolution than the SCT0-013-010.
Since our sensor will be used to measure current draw of the motor as well as power consumption over time. The SCT013-010 will be more than capable of providing accurate readings at a significantly cheaper price than the other two. Another benefit of this model sensor is that they come with a 3.5mm output Connection allowing for off the shelf plug and play replacement. This also reduces the need to purchase additional components to connect the sensor to the circuit aside from a 3.5mm jack to mount on the PCB.

3.10 Thermal Sensor Comparison

The thermal sensor(s) needed for the ball milling machine must meet several criteria. First, it must be able to measure to at least 100C to be able to register the motor of the ball milling machine as it reaches its critical threshold of overheating. As the sensor hits this magnitude in temperature a flag will be raised to the controller and scheduler to stop and let the ball milling machine motor rest and cool down. However, to err on the side of caution and higher quality design and greater range of flexibility the desired thermal sensor ought to ideally measure up to 150C (1.5X the threshold) to have a more than comfortable offset for the resolution. The next criteria is the resolution itself. The resolution should at least be within ±1.5C (150C/100 or 1% margin of error of maximum temperature) to accurately measure the temperature of the ball milling machine's motor and give correct as possible feedback for the other devices relying on this information. The budget of the ball milling machine upgrades must also be considered. This project's budget is sponsored up to $1,000 cumulatively; therefore, the sensor's cost must be relatively cheap being just one component of a larger, grander design. A reasonable price for the thermal sensor should be in range of about 20 – 50 dollars, but more can be spent if any of the other components are cheaper than anticipated.

The size of the sensor is also a crucial parameter to be taken into account. The size of the sensor, though not as critical to the project as the above criteria, is still a constraint to be weary of and to consider diligently. The sensors and PCB together in a most ideal situation fit neatly and comfortable just behind the front panel of the ball milling machine – a panel that is 168.275mm by 85.600mm. This criteria is a soft criteria for the reason that we could be flexible with both our PCB design and its placement. We can design the PCB to be double sided to give us extra space and we are also capable of using longer connections to connect the PCB and sensors to the interface.

That being said, the PCB would most comfortably fit if it were able to be placed right behind the interface panel, which requires careful and compact design with its components including the sensors. Since we are planning to incorporate as many as four to five sensors for the ball milling machine project, preemptively securing space for the sensors and allocating/rationing that space. Currently, we're planning conserving half of the space behind the ball milling machine interface panel for our sensors, and of that half space to divide that space evenly between the five sensors – but this is subject to change as necessary given the space of the other sensors if some are smaller or larger than the others.

3.10.1 Thermal Sensor Technology Selection

Infrared

Infrared (IR) sensors work by measuring and reading infrared radiation emitting from an object. All objects above absolute zero (0 Kelvin or –273.15 Celsius) emit radiation. The hotter the object the greater this radiation is and the shorter its wavelength. This is then measured, processed, and then converted into an electrical signal to give the thermal reading of the measured object. This is a potential method for measuring the temperature of the ball milling machine motor.

Measuring the temperature of the ball milling machine motor could be accomplished with an infrared sensor mounted on the PCB facing towards the motor. Since the motor would have significantly the highest temperature inside the ball milling machine casing, it would stand out and be simple to identify for the sensor. With the infrared motor sensor facing towards and isolating the motor, measuring its temperature, and waiting for it to – if ever – reach a critical threshold, of which our customer Dr. Blair has informed us the motor's critical temperature is at approximately 100C. At this point – the sensor would send a flag to the microcontroller (we're very likely going to use the ESP32 WROOM for this ball milling machine project) to send a command to shut the motor off. From there we have a couple options: To use the infrared sensor to read the motor's temperature and reactivate the ball milling machine at a specific yet to be determined temperature, or set a timer for the ball milling machine to reactivate after 30 minutes which is currently a manual process for the Blair Research team.

The advantages of using an infrared sensor are pronounced. An infrared sensor is on average much more accurate than the typical thermistor – often having a resolution less than 1C. The physical setup of the infrared sensor too would be relatively simple compared to other approaches/methods. The infrared sensor would need only be fixed to the microcontroller and facing the motor – or at least enough so to have the ball milling machine motor in its range of sight. From there it would need only further programmed through and interfaced with the microcontroller using the appropriate code and pin interactions. Infrared sensors also have a high temperature range compared to thermocouples and thermistors.

The disadvantages of using an infrared sensor for the ball milling machine project are not overwhelming but these disadvantages are still quite palpable and offer their own challenges. The simplest of these disadvantages is the cost: The cost of an infrared sensor is typically twice the amount of either the typical/average thermocouple or thermistor. Though granted, this problem is less of a problem than it appears – even at twice the cost, the price of an infrared sensor would only be an extra 10 dollars towards the budget, which isn't ideal but isn't out of the question if it makes the task of monitoring the ball milling machine's motor easier. However, the disadvantages continue. Using an infrared sensor isn't as easy as just mounting it to the microcontroller and programming it; for the infrared sensor to work and measure the temperature of an object correctly, we must know the desired object's emissivity. Emissivity is an innate quality of the material – similar to impedance/admittance - that is the amount of radiation the object gives off. For the ball milling machine motor thermal sensor to work as intended our group would need to measure the motor's emissivity and factor it into calculating the motor's actual temperature. The infrared red sensor may gradually drift over time and will have to be recalibrated if it does drift. Furthermore, even with these considered, given the nature of the ball milling machine and the chemicals and powders used in it, the infrared sensor may get dirty and disrupt the infrared vision, requiring some level of further and periodic maintenance for the ball milling machine infrared thermal sensor. Notwithstanding these concerns, an infrared thermal sensor remains a potential path for selection for monitoring the temperature of the ball milling machine's motor.

Thermocouple

A Thermocouple uses two different metals formed together at two points forming two junctions. When there is a temperature difference between the two junctions a small voltage is produced, and this produced voltage is directly related and proportional to the difference in temperature and is used to calculate it. This physical phenomena used in thermocouples is known as the Seebeck effect and is an electromagnetic force (EMF) - similar to (but should not be confused with) thermal voltage using the Boltzmann's constant. Measuring temperature using a thermocouple is done by placing one of the junctions where the temperature is intended to be measured while the other junction in placed in an area with a stable temperature – such as room temperature – to act as a controlled variable for the measurement.

The advantages of using a thermocouple to measure the temperature of the ball milling machine motor are adequate, noticeable, and in a few ways unique from the other approaches for thermal sensors. Thermocouples are very cost effective compared to infrared thermal sensors. Thermocouples are roughly 60% the cost (though with few exceptions depending on the manufacturer) of most infrared thermal sensors. Thermocouples also have a wide range of temperature readings going up well into 1000C. Due to its their small sizes, thermocouples have exceptionally quick response times giving near instant readings in change of temperature. Thermocouples are also very rugged and simple to use making it an excellent for harsh environments such as the inside of the ball milling machine and any of its contents that may spill out onto it. Most unique to the thermocouple's advantages over the other options of thermal sensor is that the thermocouple is self powered, not needing any other outside power source for its operation.

The disadvantages of thermocouples are however just as nearly numerous as its benefits. The accuracy/resolution of a thermocouple isn't nearly as accurate as an infrared thermal sensor, usually being at least a full degree celsius as its lowest resolution. Thermocouples require specialized cabling for it to perform its measurements. Thermocouples also often need a whole other integrated chip to amplify its low voltage measurements, otherwise the low voltage will be greatly susceptible to noise. Still related to noise negatively affecting the thermocouple, the wires of the thermocouple are greatly influenced by electromagnetic interference (EMI). The motor of the ball milling machine will likely produce a significant amount of electromagnetic interference relative to the small thermocouple wires and could potentially give varyingly inaccurate measurements – which could prove disastrous for one of the more important and crucial sensors given its task and role for the ball milling machine project.

A thermocouple could still be a potential option for a thermal sensor to measure the ball milling machine's motor, but many of its shortcomings will require workarounds and fairly clever ones at that. Most importantly if our senior design group were to go forward with a thermocouple to use to measure the ball milling machine motor's temperature will be to verify how impactful the electromagnetic interference from the ball milling machine motor will be on the thermocouple, and if it is a problem we may have to implement a creative measure to overcome this. Depending on how difficult this may turn out to be will likely be the most crucial factor in making or breaking the thermocouple as the choice we make for the ball milling machine project. Notwithstanding these disadvantages, our senior design group will continue to research and consider using a thermocouple to measure the temperature of ball milling machine's motor. The thermocouple's ruggedness make it an attractive option so much that it may be worth the extra work arounds to overcome the electromagnetic interference problem from the ball milling machine's motor's operation.

Thermistor

Thermistors are used to measure the thermal energy of an object or substance. Thermistors are often made with ceramic material often composed of metallic oxides. These materials either have a high negative temperature coefficient (NTC) or a high positive temperature coefficient (PTC). This means the thermistor material's resistance changes significantly with temperature. NTC thermistors decrease in resistance as temperature increases, while PTC thermistors increase in resistance as temperature rises. This resistance change relationship to temperature is how a thermistor measures thermal readings.

To make these measurements, the thermistor is incorporated into an electrical circuit, often a voltage divider. In this circuit, the thermistor is connected to a known, fixed resistor. Then, when a constant voltage is applied to the thermistor circuit, the applied voltage across the thermistor changes with the changes to the resistance – which is in turn changed by the temperature. With this process, temperature is related to the changing voltage and calculated. This process is often fed into an analog-to-digital- converter (ADC) found on a microcontroller such as the ESP430 or the ESP32 WROOM. The calculations for converting the voltage reading to a temperature reading is nonlinear and requires the use of the Steinheart-Hart equation or a beta parameter.

Thermal Sensor Summary and Decision

Thermal Sensor Type	Pros	Cons
Infrared	• High accuracy • Simple setup • Measures from distance	• Expensive • Dirty lens will degrade performance • Extra emissivity calculations
Thermocouple	• Wide Temperature Range	• Less accurate • Low Output voltage leads to Noise

	• Fast Response Time • Simple • Rugged • Low Cost • Self-powered	• Drift over time • Requires Specialized Cabling
Thermistor	• High accuracy • Compact Size • Low Cost	• Non-linearity • Limited Range • Self-heating • Drift over time

Table 11: Thermal Sensor Comparison

Above is a table of potential thermal sensor candidates for the ball milling machine project. In the table the sensor candidates are listed with the aforementioned criteria to better organize our selection process. Considering all of the criteria, our group has selected the B075J9K44Y for the ball milling machine project. We chose this sensor for the ball milling machine project because it had not only the single best resolution giving it the most accurate data readings to be used for the product – and important readings too considering the its use to measure for the ball milling machine's motor overheating which could lead to critical failure and damage – but the B075J9K44Y thermal sensor was also middle in cost and maintaining the desired threshold of measuring up to a minimum of 150 degrees celsius.

The B075J9K44Y thermal sensor's biggest disadvantage compared to all of the other thermal sensors is its size. Being 1.8 times the size of the next largest thermal sensor (B075J9K44Y/PCC-SMP-K-5 (28x16)/(15.5x16) = 1.8). However, the size of the sensor is only a soft requirement – more of a consideration - and can be worked around in various methods provided the volume of the B075J9K44Y thermal sensor's size wasn't excessively large. The B075J9K44Y thermal sensor is also capable of contactless measurement to aid in the event the size of the thermal sensor proved to be an issue.

Several close contending selections were the WTK-10-60-TT and AD597ARZ. Both of these thermal sensors met the hard requirement to reach up to and beyond 150C – which is again crucial for measuring the thermal condition of the motor to detect if it's overheating. What made both the thermal sensors appealing is their cheaper price tag than the B075J9K44Y. However, both the thermal sensors WTK-10-60-TT and AD597ARZ thermal sensors had poorer resolution and therefore less accuracy in their measurements on something that's relatively crucial for the ball milling machine project. With these considerations and above reasoning, our group has ultimately selected the B075J9K44Y thermal sensor to be utilized for the ball milling machine project.

The B075J9K44Y thermal sensor uses I2C architecture for its operation, which aligns nicely with our design preferences – the I2C architecture allowing our Senior Design team to make use of the I2C shared bus and save GPIO pins on the PCB for other sensors and peripherals. The B075J9K44Y thermal sensor's power requirement is the standard 3.3V ~ 5V supply if used with an MCU. The B075J9K44Y thermal sensor is a High-precision contact-less temperature measurement device that fills its advertised role for our Senior Design ball milling machine project to measure and record the temperature of the ball milling machine motor, detecting for a critical temperature of 100C signaling overheating.

Model	Minimum Temp	Maximum Temp	Resolution	Cost	Size in mm

Table 12: Thermal (Selected Type) Sensor Comparison

3.10.2 Thermal Sensor Part Selection

3.11 Humidity Sensor Selection

The humidity sensor for the ball milling machine project – though not as critical as the thermal sensor – still plays a helpful role and task for the ball milling machine. Though the humidity sensor doesn't play as direct of a role as the thermal sensor or the other sensors, given that this device is utilized for research, having the extra information and context of the environment surrounding the ball milling machine gives a better insight to the output of the ball milling machine's output mixture and can help troubleshoot from the chemistry perspective should the ball milling machine not give the expected output. This humidity sensor is therefore still pertinent to have for this ball milling machine upgrade project.

The plan for the humidity sensor is simple – mostly due in part in how loose the design limitations are for the sensor. Since measuring the humidity is measured from the general moisture in the vicinity or room, and since it is not urgent nor critical to the function of the ball milling machine, the humidity sensor measurement is relatively relaxed. The current criteria for the ball milling machine humidity sensor to be put simply is to just work. This humidity sensor just needs to read the humidity in the room at consistent intervals and to record it accordingly. Again, this sensor isn't inherently integral to the ball milling machine's functions or fundamental purpose; the humidity is being read for the sole purposes of research and troubleshooting so only measuring and recording the data and displaying the recorded data on the home assistant app for monitoring purposes. Unlike the ball milling machine motor thermal sensor there will be little to no logic usage or determination (IE flags) from the humidity data.

Most if not all the humidity sensors our group researched for part selection for the ball milling machine were dual sensors – also incorporating a thermal sensor in addition with the humidity sensor. Though there wasn't a need to also measure the temperature of the room, since nearly all of the humidity sensors had incorporated one already, and there isn't any harm in also knowing the room temperature also.

Model	Minimum Reading	Maximum Reading	Resolution/Accuracy	Cost	Size in mm
SHT41	0% RH -40C	100% RH 125C	1.8% RH, 0.2C	3.12	1.5x1.5x0.5
SHT30	0% RH -40C	100% RH 125C	2% RH 0.2C	4.94	2.5x2.5x0.9
DHT22	0% RH -40C	100% RH 80C	2-5%RH 0.5C	9.95	1.05x2.32 x0.53
SHTC3	0% RH -40C	100% RH 125C	2% RH 0.2C	2.86	2x2x0.7
Phidgets 1125_0	10% RH -30C	95% RH 80C	5% RH 0.75C	74.33	35.56x50.8 X8.53
CHS-UPS	5% RH 0C	95% RH 60C	3% RH	77.17	9x6.5x3.7
Adafruit BME688	5% RH 0C	100% RH 80C	3% RH 1.0C	19.95	25.5x17.6 x4.6
SHT20	0% RH -40C	100% RH 125C	3% RH 0.3C	5.40	3.0x3.0x1.1

Our Senior Design group has decided to use the SHTC3. We have selected the SHTC3 to be our humidity sensor for its overall criteria meeting or surpassing most of the other humidity sensors seen in the similar minimum and maximum humidity readings and resolution readings while also being much cheaper than the other humidity sensors. This choice is a rather straightforward decision considering the above criteria. Therefore, our Senior Design Group will move forward with the SHTC3 humidity sensor for the ball milling machine project.

3.12 Accelerometer Selection

One of the goals of this project is to monitor the health of the ball milling machine.

The number one most important factor in the health of the machine is whether the machine is actually able to move. We intend to put an accelerometer on the platform in the ball milling machine so that we can determine if the mill is properly performing its job. We will be able to use the accelerometer to determine a baseline acceleration in each axis for when the machine is functioning properly. We will then monitor the milling machine while it is running. If the accelerometer starts reading heavily increased or reduced values in any given direction, we can conclude that something is wrong. It may be a broken belt, a part that has come loose, or simply that the sensor stopped working. Regardless of what isn't functioning correctly, any of these factors is a problem.

To accomplish this function, the selected accelerometer must be reasonably accurate and consistent. If the readings have a large amount of variance, this sensor will not be as effective at detecting problems. We also want the accelerometer to be low power, and use a common communication protocol, such as SPI or I^2C. Of the two, we would prefer I^2C so that we can take advantage of the shared bus, and not use additional GPIO pins for this sensor.

As we have discussed in other part selections, we are additionally concerned with the trade offs between size and practicality. For debugging purposes it would be extremely useful to have exposed pins. That way we can read the signals from those pins with a digital logic analyzer or oscilloscope. If something isn't working correctly with the module, this ability to use an analyzer or oscilloscope would help us determine if the issue is the commands that are being sent to the sensor from our microcontroller, or if the issue is because we incorrectly connected the module, or if the module itself is simply not functioning the way it is supposed to. This advantage, however, comes with the trade off of requiring substantially more area on a space-limited printed circuit board. Since the accelerometer is a small module this concern is not as significant as when we selected our microcontroller module, but it's always something we consider. The more relevant issue with the package type for an accelerometer is that most of the reasonably priced accelerometer modules only come in packages where the contacts are completely underneath the chip.

Additional factors we want to consider are the output bit width of the accelerometers and the supported scaling factors for the readings. By selecting an appropriate combination of accelerometer reading scale and bit width, we can have a large amount of precision from our accelerometer data.

The four accelerometer modules we compared are shown in table 3.12.1. We looked at a cheap offering from Memsic, a marginally pricier option from Kionix, an offering from Bosch, and a more expensive option from STMicroelectronics.

	STM IIS328DQ [27]	**Memsic MXC6655XA [28]**	**Kionix KXTJ3-1057 [29]**	**Bosch BMA400 [41]**

Supported Scales	±2g/±4g/±8g	±2g/±4g/±8g	±2g/±4g/±8g/±16 g	±2g/±4g/± 8g/±16g
Output Bit Width	16 bits	12 bits	8/12/14 bit modes	12 bits
Output Data Rate	0.5 Hz to 1 kHz	100 Hz	0.781 Hz to 1.6 kHz	125 to 800 Hz
Interface Type	SPI or I^2C	I^2C	I^2C	SPI or I^2C
Shock Survivabilit y	10,000g	200,000g	10,000g	10,000g
Part Dimensions	4.0 x 4.0 x 1.8 mm	2.0 x 2.0 x 1.0 mm	2.0 x 2.0 x 0.9 mm	2.0 x 2.0 x 1.0 mm
Board Area Consumed	16 mm^2	4 mm^2	4 mm^2	4 mm^2
Footprint Type	QPFN	LGA	LGA	LGA
Current Cost from Digikey	$6.69	$1.70	$2.80	$3.45

Table 3.12.1: Accelerometer Module Comparison

Starting with the supported scales and the output data width, all three of the accelerometers seem like reasonable choices. While we are not sure precisely how much acceleration the ball milling machine platform experiences, it is certainly less than eight Gs, and should be an appreciable factor of 2 Gs on the low end, so all four of the accelerometers have appropriate options for the output scaling. Additionally, all four of the accelerometers have an output data width of at least eight bits. Eight bits gives us 256 steps of quantization below the accelerometer scaling. Since we need to receive both positive and negative readings, it will provide 128 steps for the negative measurements, and 127 steps for the positive measurements. If we were to use the ±2g setting, this provides us with measurements to the nearest 2g/128 = 0.15323 m/s^2, for a maximum quantization error of 0.07661 m/s^2. If we need to use the ±4g setting, our maximum quantization error will double, while our precision will halve.

One of the benefits of the STM IIS328DQ accelerometer and the Bosch MB85RC256V over the other two options is that they support either I^2C or SPI. In most cases, this would be a nice feature that allows us to trade off the number of pins that we are consuming in exchange for a higher interface clock, or to tap in to our existing I^2C bus at the expense of the data transfer speed. That is still something that happens here, but it is rendered insignificant by the output data rate of the accelerometers. Since we will only be transmitting between 24 and 48 bits at a time depending on the output data width of the accelerometer, the data transmission time will be extremely minimal. This increase would not allow us to attain more frequent readings from the device because the bottleneck in that process will be the accelerometer measurement time itself. The maximum output data rate of these accelerometers is 1.6 kHz for the Kionix KXTJ3-1057, and the other three devices are slower. For this reason, we would choose I^2C over SPI even if we select the STM IIS328DQ or Bosch MB85RC256V sensors, and are happy that it is the provided mode of communication for the other two accelerometer options.

The shock survivability of these accelerometers is another interesting parameter. The STM IIS328DQ, Bosch MB85RC256V and Kionix KXTJ3-1057 both guarantee survivability up to 10,000g, while the Memsic MXC6655XA claims an astounding 200,000g. It did not seem right to us that the cheapest of the four options claimed to have the highest shock survivability, so we decided to dig into the datasheets a bit further. The STMicroelectronics, Bosch, and Kionix datasheets all specify shock survivability without any additional qualifiers [27][29]. The Memsic datasheet, on the other hand, specifies the “shock survival of the MEMS sensing structure” [28]. None of the four datasheets indicate any standards that were used for their shock survivability testing, but the difference in phrasing leads us to believe that the Memsic MXC6655XA may be surviving to a lower standard than the other three. It only needs to have an intact MEMS sensing structure after experiencing a 200,000g force, while we believe that the other three sensors could conceivably be still fully functioning at their 10,000g limit. Regardless, the 10,000g shock survivability will be more than enough for our application. If the ball milling machine were to fall apart while in motion, there could be significant shocks, but even then it would not reach 10,000g. And even if there were some sort of freak 10,000g event, the survival of the accelerometer module would be the least of our concerns.

We ultimately selected the Bosch MB85RC256V accelerometer. While it is not the most performant of the four options, it is more than sufficient for our needs. For our use case we do not need an incredibly accurate or fast accelerometer, and we are not particularly concerned if the shock survivability rating is not entirely reliable. We also like the small size of the LGA package over the QPFN footprint type found in the STM offering. This size does create challenges if we want to probe the I^2C bus at the accelerometer, but as long as our PCB is designed reasonably well, the signals on the I^2C bus at the microcontroller should be nearly identical to the voltages at the accelerometer. If we end up having more concerns in this area, we can intentionally create exposed probing points on the PCB without solder masks that come off of the I^2C bus traces near the accelerometer.

The Bosch MB85RC256V accelerometer has simple input/output ports, with a

1.62-3.6V supply voltage pin, a digital IO supply voltage pin, a ground pin, the two I^2C pins, three additional pins for SPI mode, and two interrupt pins. The interrupt pins are interesting, because they allow us to ensure that the microcontroller is notified when the orientation changes or when the accelerometer starts shaking. Either of these notifications could indicate some sort of catastrophic system failure. The platform that this accelerometer will be secured to should not have an orientation change. If that scenario occurs, then either something has gone horribly wrong with the physical ball milling machine, or the accelerometer is reporting an event in error. The shaking interrupt could be used for when the ball milling machine is supposed to be turned off. It could indicate that a relay is stuck on, or that the ball milling machine is being physically moved.

3.13 Motor Detection Sensor Selection

One of the variables that we want to monitor with our sensors is the current state of the motor. We want to be able to detect whether or not the motor is actively running. This could help us detect certain failure modes in the ball milling machine. If, for example, there is a short somewhere in the motor, the current sensor will still detect current entering the motor, but the motor itself may not be spinning. If this scenario were to occur, we would want to promptly shut down the relays that control the flow of current to the motor.

In order to detect the state of the motor, we will use a hall effect sensor. Such sensors are used to detect magnetic fields, like the fields produced by a spinning AC motor. We can profile the fields when the motor is on and when the motor is off, then use this profile and the current sensor output to determine if the motor is on at any given time.

There are a few different factors that are important to use when it comes to selecting an appropriate hall effect sensor. The first such factor is the sensitivity of the sensor. Since this sensor will be outside of the motor’s metal casing, we need to ensure that it is still sensitive enough to detect the fields that make it outside of the motor. The second factor we consider is the sensing range of the device. We need to make sure that the range is wide enough to properly distinguish “off” from “on”. The third factor we considered is the number of axes detected by the sensor. Many of the cheapest hall effect sensor only detect fields along a single axis, while slightly fancier sensors will be able to detect fields along all three axes. A single axis sensor will likely work just fine for us, since we want to simply detect the magnitude of the magnetic field, and do not particularly care about the precise orientation of the field. However, with a single axis sensor, we would need to be more careful about how the sensor is physically oriented with respect to the motor. With a three axis sensor, we would be able to orient the sensor in any manner, and still successfully detect the magnitude of the magnetic field that is produced by the motor.

Outside of the sensing parameters, we also considered several factors that affect the usability of the sensor. The first is, of course, the communication method that the sensor supports. Several of the cheaper single-axis options simply output one analog voltage that corresponds to the detected field with a linear scaling factor. Many of the more sophisticated sensors support either SPI, I^2C, or both options. I^2C is, as always, nice because it does not consume any extra GPIO pins. SPI is faster, but in this case, we do not want to use the GPIO pins. The analog output will use one extra ADC pin. Most of the other sensors we have selected communicate digitally, so we are not hurting for GPIO pins that are connected to the ADC, but we are still already using a fairly large amount of GPIO. We therefore would prefer a sensor that supports I^2C, but would not be completely against a sensor that uses an analog output. The other two practical concerns relate to the physical footprint of the device. Per usual, since we are hand soldering, we would like a larger package with pads that are not underneath the device. This generally corresponds to the sensors with an analog output. We would also like a small sensor that does not consume a lot of surface area on the PCB.

The three hall effect sensors we considered are shown in table 3.13.1 below. We looked at options from Texas Instruments, Allegro Microsystems, and Infineon Technologies. We selected the Infineon Technologies TLE493D-W2B6 hall effect sensor, for its 3 axes, I^2C support, and convenient package. We decided that the lower sensitivity was an acceptable trade off for these conveniences and the wider sensing range.

	Texas Instruments DRV5053VAQDBZR [43]	**Allegro Microsystems ALS31300EEJASR-1000 [44]**	**Infineon Technologies TLE493D-W2B6 [45]**
Sensitivity	-90 mV/mT	20 LSB/mT	0.13mT / LSB
Sensing Range	±9 mT	±100 mT	±160 mT
Number of Axes	1	3	3
Interface Type	Analog voltage	I^2C	I^2C
Part Dimensions	2.92 x 2.37 x 1.12 mm	3.0 x 3.0 x 0.75 mm	2.9 x 2.8 x 1.1 mm
Board Area Consumed	6.92 mm^2	9 mm^2	8.12 mm^2
Footprint Type	SOT-23	10-DFN	TSOP6

Current Cost from Digikey	$1.01	$2.05	$2.97

Table 3.13.1: Hall Effect Sensor Module Comparison

3.14 Infrared Distance Sensor Selection

3.15 Power Conversion Comparison

3.15.1 Power Conversion Technology Selection

The power to our design will come from a 12V DC rail that is provided by the mill's original supply. We will need to take this 12V supply and convert it to a 3.3V DC source to supply power to the microcontroller which consumes approximately 350mA. However, various reports indicate that when transmitting Wi-Fi this peak at nearly double. Likewise, the OLED display also requires a 3.3V and consumes a max of 360mA. We will need to supply power to the relays that control the power to the latches and motor. Additionally, the various sensors will also require power although low. Two suitable options for PCB mountable buck converters are non-isolated and isolated.
Non-Isolated Buck Converter

A basic buck or stepdown converter is a simple circuit consisting of a voltage source, switch, rectifying diode, inductor, and a capacitor. When the switch is closed the Inductor and capacitor begin to charge until the switch is opened. At this point the inductor attempts to maintain its steady current flow, it induces an opposite polarity voltage that can now supply power to the load. Likewise, the capacitor maintains a steady voltage smoothing the output and providing power to the load. This is allowed thanks to the rectifying diode being in the forward bias position when the switch is open thus allowing the current to flow.
In a standard non isolated buck converter module like those found on a PCB the switch is replaced with a MOSFET and a control signal to open and close and a specified duty cycle. The control signal is also contained in and supplied by the module making the circuitry incredibly simple. The duty cycle that is produced determines the output voltage as the output voltage is proportional to the duty cycle ratio. These modules are relatively compact and offer high efficiencies with some modules offering upwards of 90% efficiency.
Isolated Buck Converters

The basic principle remains the same between the isolated and non-isolated buck converter with a few differences. The first is that the inductor is replaced by a transformer. The second is ensuring that the secondary coil has an independent ground providing galvanic isolation. This can protect sensitive

equipment from current spikes on startup. Another benefit is that they reduce the amount of EMI noise generated by the switching. Their Isolation also protects them from electrically noisy environments.
All these benefits come at the cost of price, size, weight, and efficiency. The introduction of the transformer increases the size and weight of the module in comparison to the non-isolated versions. The addition of a transformer and isolated ground accounts for a loss in efficiency as many isolated buck converters only offer efficiencies of 80-85% on average. More expensive modules can provide increased efficiencies comparable to non-isolated modules.

When choosing between the two we must consider the environment in which it will be in. The mill utilizes a high-power electric motor which can generate significant noise. The added noise can significantly lower the output efficiency of the non-isolated modules. The added size of the isolated modules is relatively small in comparison to the added protection and benefit of absorbing the loss of efficiency on the front end. When we consider that the power will come from an outlet and not a battery in our design, we can afford a slight loss in efficiency. The sacrifice in efficiency is worth it for the gain of safety and resilience to our design. An isolated DC to DC buck converter will be the best option the electrically harsh environment it will be operating in.

3.15.2 Power Conversion Part Selection

4 Related Standards and Design Constraints

4.1 Standards

4.1.1 IEEE 802.11 (Wi-Fi)

The Institute of Electrical and Electronic Electronics Engineers (IEEE) is a professional organization with a variety of different technical societies across the field of electrical engineering. In addition to such technical societies, IEEE includes a standards association that has developed more than 2,100 standards [40]. One of the standards that they have developed is incredibly relevant to any project or device that intends to wirelessly connect to the internet.

IEEE 802.11 is a standard that describes and supports the creation of wireless local area networks (WLAN) [23]. This standard uses the existing IEEE 802.2 logical link control sublayer, so that IEEE 802.11 WLANs appear the same as IEEE 802.3 ethernet LANs to the protocol layers that exist above the logical link control sublayer. This ensures compatibility with all types of communication that are sent over a traditional ethernet based local area network.

One place where IEEE 802.11 expands on the existing local area network standards is with the medium access control (MAC) sublayer. The IEEE 802.11 MAC has three functions [23]. The first is to reliably deliver data to all users of the MAC. The second is control access to, and reduce contention for, the shared physical device that is used to send information wirelessly. The final function is to provide secure delivery of the data. Security is a little bit less of a concern in traditional wired local area networks, because someone needs to have access to the physical ethernet port in order to intercept data. Access to these ports can be physically controlled. In a WLAN, the data is being broadcast wirelessly, meaning that it is much easier for other devices to intercept it. Security is ensured through encryption of the transmitted data.

Reliable delivery of data is ensured by the MAC level frame exchange protocol [23]. Each frame that is delivered to a recipient is required to have an acknowledgement sent back to the sender. Once the sending MAC has received this acknowledgement, it knows that the data has been delivered. If the acknowledgement is not received, the MAC is able to re-transmit the frame. This process is something that does not need to occur at the MAC level on traditional ethernet LANs since wired transmission is more reliable. The unreliability that is inherent to wireless communications is what necessitates this additional overhead for the MAC in the IEEE 802.11 standard.

Another part of MAC functionality enforced by the IEEE 802.11 standard is the format of the MAC header. Each IEEE 802.11 frame starts with a MAC header before the Frame Body, then concludes with a frame check sequence field [23]. The MAC header consists of a Frame Control Field, a transmission ID, four address (source, destination, and two addresses to indicate the transmitter, receiver, or basic service set type), and a sequence control number. The frame control field includes information such as the protocol version, the frame type (control, data, or management), a field that indicates whether the frame is going from a station to an access point, an indication for whether or not this frame is the last part of the data being sent, a retry field that indicates a retransmission, a power management field that indicates whether the station will be available after this transmission, a bit that indicates that there will be another frame immediately after this one, a bit that indicates encryption status, and a bit that requests strict ordering.

Beyond the base IEEE 802.11 standard, the ESP32 microcontroller supports IEEE 802.11 b/g/n. These are three extensions to the standard that provide superior functionality compared to the original IEEE 802.11 version. These three extensions are all improvements to the original IEEE 802.11 physical layer or PHY. Before we look at what these extensions modified, we need to understand the original PHY.

The PHY is the bottom layer of the Open System Interconnection (OSI) stack [23]. The PHY is the part of the system that takes data from the MAC and converts it to physical signals that can be transmitted over the wireless media. The IEEE 802.11

standard uses specifications that ensure compliant PHYs will meet the emissions guidelines of US, European, and Japanese authorities. There were three types of PHYs supported in the original version of the standard. The direct sequence spread spectrum PHY supported data rates of either 1 Mbit/s or 2 Mbit/s when using differential binary phase shift keying or differential quadrature phase shift keying respectively [23]..

The second type of PHY supported by the base IEEE 802.11 standard is the frequency hopping spread spectrum (FHSS) PHY. The FHSS PHY supports different data rates for different components of the transmission. The preamble and header are transmitted at a standard rate of 1 Mbit/s, while the bulk of the data transmission supports rates from 1 Mbit/s to 4.5 Mbit/s in half megabit increments [23].

The final PHY specified in the IEEE 802.11 standard is the infrared PHY. The previous two PHYs both used the 2.4 GHz spectrum as their mediums, but the infrared PHY uses infrared light. This medium requires direct line of sight, which makes transmission more difficult, but it also makes intercepting the signal more challenging, adding a little bit of intrinsic security. The infrared PHY supports data transmission rates of either 1 Mbit/s or 2 Mbit/s [23].

The first PHY extension supported by the Espressif ESP32 line of microcontroller is the IEEE 802.11b standard. IEEE 802.11b is for high-rate direct sequence spread spectrum (HR/DSSS) communications [23]. This version of the standard uses better modulation techniques to increase the data rates to 5.5 or 11 Mbit/s, while still supporting the original 1 Mbit/s and 2 Mbit/s data rates from the original DSS PHY. The second extension to the standard that is supported by the ESP32 is IEEE 802.11g. IEEE 802.11g offers substantially larger improvements that push data rates up to 54 Mbit/s [23]. IEEE 802.11g supports backward compatibility with parts of IEEE 802.11b, while also adding support for orthogonal frequency division multiplexing. The final extension supported by the ESP32 devices is IEEE 802.11n. IEEE 802.11n is designed to deliver data rates that are higher than the maximum 54 Mbit/s that was supported by IEEE 802.11g. The supported data rates are in excess of 100 Mbit/s. The precise version of the standard on the ESP32 WROOM module supports data rates up to 150 Mbit/s [8].

The IEEE 802.11 standard is critical to our project. It ensures that our enhanced ball milling machine controller will be able to connect to a wireless local area network using any router that also supports the IEEE 802.11 b, g, or n standards. There is no requirement to buy equipment from a specific manufacturer or to install proprietary software to ensure compatibility. These standards mean that we can implement wireless connectivity in a manner that will be very easy for the end user to set up.

4.1.2 MQTT (Oasis Standard):

MQTT stands for Message Queuing Telemetry Transport. It uses TCP / IP where MQTT is the layer above TCP in the “protocol/network stack”. TCP stands for Transmission Control Protocol and is an internet protocol that is connection oriented. This means that acknowledgment of connections, and reception of packets (a packet is a packaged chunk of data) is sent by the receiver and the transmitter is listening for these signals. This is compared to UDP which is connectionless meaning the sender does not have communication with the receiver. MQTT uses the terminology of client and broker where a broker is the central server maintaining connections between all its clients. In a typical configuration, and in our configuration, the broker and client connect to each other using a router. The router provides the broker with a public facing IP address and the client/s with private IP addresses and NAT. NAT stands for Network Address Translation and connects an assigned private IP address to a public address that can be used to connect to the broker. More than enough private IP addresses can be assigned, but the translator (NAT) will only have so many public facing IP addresses, so this may or may not be our bottle neck with respect to how many clients we can host. In our case, each client and the broker are local, so no internet service provider is required and every client must share the same frequency band. MQTT uses publish and subscribe models where any client can publish and subscribe data to a topic. This means that the broker must keep every connection alive. A topic can be thought of as a location like what a website or a computer will have for accessibility (ex. Drive C:/Users/Me/Desktop/). Data can be collected from this location (subscribe) or posted to this location (publish). The advantages MQTT provides for IoT applications include: small packets of data (light weight), organization (topic architecture), and the option for assurance of delivery (QoS 0-2). QoS stands for Quality of Service and describes 3 packet delivery options. The first option is QoS = 0 and does not include any assurance of delivery. QoS = 1 has ensures the packet was published. This means if something goes wrong with sending a packet, the broker will resend the packet. QoS=2 ensures the packet was received by the client.

Protocol Details

A packet is made up of two headers and the payload. The first header is the fixed header and every packet has a fixed header. The second header is the variable header and is not included in some packet types. The fixed header is 2 Bytes long. The first Byte has 4 bits (0-3) designated for flags and the last 4 bits (4-7) designated for the specification of the control packet. Since there are 4 bits in the second field, there are 15 control packet types (2^4 - 1). The second Byte contains the total size of the remaining message. This structure is laid out in the following table.

Level	Size	Description

Fixed Header	2 Bytes	Specifies packet type, flags, and total message size
Variable Header (ID Field)	0-2 Bytes	42 total IDs for different usages
Variable Header (Properties Field)	0-4 Bytes	contains integers, strings, or binary depending on usage
Payload	Any	Typically, the msg data from client in PUBLISH

Table ___: MQTT Packet Structure

The maximum payload can be set in the properties of the variable header for the CONNECT and CONNACK packet types. A packet type is a packet with a defined set of parameters such as if flags are possible in the fixed header, what fields are included in the variable header, and if there is a payload. The CONNECT is sent by the client and the CONNACK is sent by the broker so the client and the broker can set a limit on the maximum payload they will receive. The CONNECT packet type is a request from the client to connect with terms. The CONNACK is a connection acknowledgement sent from broker accepting the clients' terms and giving its own terms to the client. Both packet types include a maximum payload field which is a term that restricts the other node from sending a packet beyond this size. The field is 4 Bytes long so the theoretical maximum would be 32 bits long which if set to all 1s, would be 4 Gigabytes. However, the standard states that in any Byte in these fields, the value cannot be 0xFF which in binary is 1111 1111; at least one of the digits must be zero. So, then the field drops from 32-bit combinations to 28-bit combinations which would mean the theoretical maximum packet size is 256 Megabytes. This size is much outside the typical range and much outside the range of what our system could handle. The typical range is within 256 Kilobytes with 256 Kilobytes being for a large-scale operation. There are 15 total possible packet types. Two packet types were mentioned earlier, but a full list is provided in the table below.

Name	Value	Variable header Contains ID	Variable header Contains Properties	Contains Payload
Reserved	0	N/A	N/A	N/A

CONNECT	1	No	Yes	Yes
CONNACK	2	No	Yes	No
PUBLISH	3	YES (for QoS > 0)	Yes	Optional
PUBACK	4	YES	Yes	No
PUBREC	5	YES	Yes	No
PUBREL	6	YES	Yes	No
PUBCOMP	7	YES	Yes	No
SUBSCRIBE	8	YES	Yes	Yes
SUBACK	9	YES	Yes	Yes
UNSUBSCRIBE	10	YES	Yes	Yes
UNSUBACK	11	YES	Yes	Yes
PINGREQ	12	NO	No	No
PINGRESP	13	NO	No	No
DISCONNECT	14	NO	Yes	No
AUTH	15	NO	Yes	No

Table ___: MQTT Packet Types

Packet types 1 and 2 were explained earlier. Most of the packet types are self-explanatory. Type 3 is used for publishing messages or data from either the client to the broker or vice versa. Type 8 is requested by the server and 9 is the confirmation sent by the broker. 10 and 11 function the same as 8 and 9. 12 and 13 are used to maintain connections after a period of no activity. It is a way for the broker to check to see if the client is still online. If there is no response, the broker will terminate the connection. 15 is used by the broker to check if the client is valid. Packet types 4 through 7 are used for quality of service options 1 and 2.

QoS 1

Packet type 4 is PUBACK and is short for publish acknowledge. It is for QoS option 1 and is an acknowledgement of the receiver that the message or data from the publish packet was published to the topic. The client or the broker can fill either role as the receiver or the sender. If the sender does not receive this acknowledgement, it will resend the publish packet until the receiver receives it. This option ensures the payload in the publish packet is delivered at least once unlike the default QoS 0 which cannot ensure anything. The following figure depicts this interaction.

Figure ___: MQTT QoS 1 Interaction

QoS 2

Packet types 4, 5, 6, and 7 are used when a senders' publish packet calls for a quality of service of 2. The first packet in the transaction is PUBREC and is short for publish received. It is an acknowledgement that the receiver got the packet from the sender. When the sender gets this packet, it deletes the initial publish packet and saves the PUBREC packet. The sender then sends a PUBREL packet which is short for publish release. Once the receiver gets this PUBREL packet, it then deletes all the stored packets and sends a PUBCOMP packet which is short for publish complete. After this packet is sent, the sender deletes the remaining stored packets and both parties know for certain the message was published not more than 1 time. This is the advantage of QoS 2. QoS 1 can guarantee delivery but cannot guarantee the publish packet was not sent more than 1 time. The following figure depicts the QoS 2 interaction.

Figure ___: MQTT QoS 1 Interaction

Application

For the most part, the protocol takes care of itself meaning the protocol is already designed to be efficient and there's not much we can do to change how much bandwidth will be consumed. Changing the size limits might be useful for the devices the packets are being sent to and from but will not affect the bandwidth or response time. Properly understanding the quality of service options is important as certain sensors or user inputs might be deemed critical in various ways. For example, scheduling a timed run of the machine will be sent as a publish packet. It is critical that the machine receives this packet so we will need a QoS 1, but bugs might occur if the same schedule is sent twice meaning we might need to invoke a QoS 2 for our scheduling messages. Sending event update messages from the ESP32 controller will require a QoS of 1 since these updates potentially describe significant events, but it does not matter if it is accidentally sent more than once. Sensor readings should use the default QoS 0 but there might be a circumstance in which a sensor will require QoS 1 or 2.

4.2 Design Impact of Relevant Standards

4.3 Realistic Design Constraints

4.3.1 Economic and Time Constraints

Economic and time constraints are crucial and critical criteria that can break any project if not accounted for and planned for with astute detail and diligence. Failing to meet either an economic constraint or a time constraint is more often than not a project breaking event or disaster – though can be relatively benign in some cases depending on the context in which the situation occurs and how it happens.

Economic constraints are obviously monetary in nature but can be more than just a simple budget. Economic constraints include not only the budget but the everchanging environments that affect the budget often seen in the form of inflation. Inflation is an economic term that refers to the general rise in prices of goods and services over time. When inflation occurs, each unit of currency buys fewer products and services. Inflation is essentially a decrease in the purchasing power of money. Various factors can cause inflation. One of the primary causes is demand-pull inflation, which happens when the demand for goods and services outstrips their supply. Another form of inflation is cost-push inflation. This type occurs when the costs to produce goods and services increase, leading companies to raise prices to maintain their profit margins. Various engineering materials and supplies are susceptible and vulnerable to inflation. So, in addition to a budget, it's imperative to also be aware of the general financial environment as prices have the potential to change – at times drastically.

Time constraints are straightforward. Much like our classes and assignments, this ball milling machine project has a series of objectives that need to be completed by a certain time. In industry, time constraints can have many factors, most often set by - and to satisfy – the customer. For this project, its function as an assignment serves as the purpose for a time constraint – our graduation being the deadline of all deadlines. This ball milling machine project – like most projects in industry – is subdivided into multiple time constraints, dividing the project over time into smaller objectives to keep pace or to secure a special or important deadline, say like a weather sensitive task for in some niche cases for example.

These time constraints can be imposed by a supervisor, customer, or self-imposed by those working/building the project to manage the project's pace and to gauge completion vs time. For Senior Design, this ball milling machine project encapsulates a little of all of the time constraints. We have a hard time constraints in the form of assignments issued to us by our management – IE our professors – in the form of the Divide and Conquer and page deadlines. We have a customer wanting us to have parts of the project completed by certain times in the form of Dr. Blair. And, we have given ourselves time constraints as goals to have parts of the ball milling machine project completed by to set ourselves a pace and to keep ourselves on that pace. Not meeting these time constraints can be disastrous, from falling behind to even outright failing the project. It's imperative that we keep track of and make our time constraints so that we our successful in our endeavor to build a quality project for our customer Dr. Blair and the Blair Research Group and for ourselves so that our Senior Design Group does well in Senior Design and passes. To accomplish this, we have marked multiple dates with each important milestone and aim to achieve it by the appropriate date or sooner.

Seen much further below in chapter '10.2 Milestones' our Senior Design Group's ball milling machine project is divided into significant milestones to help keep our Senior Design Group on pace. Our Senior Design Group chose major tasks in an order that would make sense – for example you wouldn't design a printed circuit board before knowing what sensors you would be using or incorporating - and then chose realistic dates that would give our Senior Design Group appropriate time to complete these tasks, while also being compliant with the hard deadlines of the project provided to us in the assignment and the nature of the course. This method has allowed our Senior Design Group to set ourselves goals and pace ourselves within our own autonomy. Currently, our Senior Design Group has met our milestones and plan to continue to meet these milestones so that we may stay compliant for both our customer Dr. Blair and for the Senior Design class.

Our Senior Design project is sponsored by Dr. Blair and the Blair Research Group of the Physics Department of the University of Central Florida. We are very fortunate and grateful for Dr. Blair for financing this ball milling machine project for his lab. For this ball milling machine project, Dr. Blair has given our Senior Design Group a budget of $1, 000.for us to spend on the project to build it, including each and all of the sensors, the OLED screen, the microcontroller, the membrane keyboard, and any of the various necessities to put it all together such as solder and jumper cables. To plan out and organize our spending to keep it within the budget, we have researched our various parts and components and took notes on all of their prices and took averages. With these averages of prices for our various components our Senior Design Group decided to make a BOM (Bill of Materials) to both attempt to predict how much our group will spend on the ball milling machine project on its various materials and as practice for when we build the actual bill of materials and finalize the project.

The importance of maintaining budget constraints is important, though admittedly not as critical as our time constraints. Maintaining our Senior Design Group's budget – again generously provided by Dr. Blair and the Blair Research Group – allows our Senior Design Group to not have to pay ourselves for the parts needed for the ball milling machine project. Otherwise, we the college students in Senior Design would have to finance the project personally. As stated before, going over the budget constraint isn't as disastrous as breaking time constraints. Breaking a budget constraint isn't the end of the world or the project – it's simply an inconvenience in most cases. In most of these cases its just paying more for materials than initially projected. Though in some cases breaking an economic constraint can be a much costlier and heavier failure if the budget was already at its maximum, or it could reflect poorly on the manager of the project to his superiors. In other times it can upset the customer who had initially only agreed to pay for the original budget. In our Senior Design Group's case anything going over the budget would need to be financed by us as individuals – which again is inconvenient but not disastrous.

As seen much below in chapter 10.1 in budget estimates and funding, it can be seen our Senior Design Group's initial projected budget for this endeavor on the ball milling machine project. With our $1, 000 budget provided to us from Dr. Blair and the Blair Group Research team, our Senior Design Group currently predicts this ball milling machine project to cost in the price range of $222 after initially considering what parts we will need to upgrade the ball milling machine with its requested modifications. Currently projected $222 of our $1, 000 budget we are in a rather safe position with more than comfortable room for growth or error for the ball milling machine project. This is outstanding for multiple reasons. The first and obvious reason is that our Senior Design Group is projected to be well under its budget, but the subsequent reasons are consequence of this. Being comfortably under budget as much as our Senior Design Group is gives a large margin of error – which in addition to the privileged luxury of error, our Senior Design Group can order multiple copies of each part if necessary, especially in the case that one breaks or malfunctions. Moreover, if needed, our Senior Design Group could express order parts if there was a time crises.

4.3.2 Size Constraints

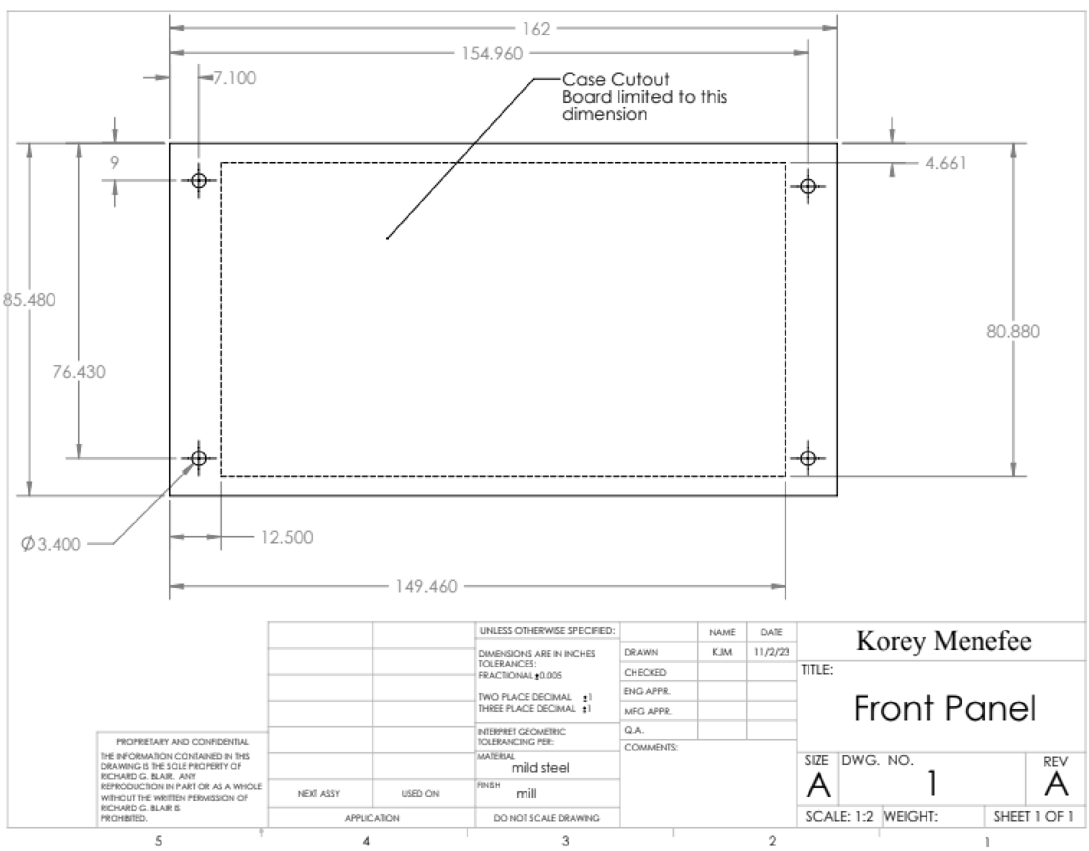
Since our project is to modify and upgrade an existing piece of equipment, we had size constraints imposed on us from the beginning. While some components may fit in the same spot as their predecessor many of the sensors, we would like to incorporate will require new mounting. The addition of new components to any piece of equipment that was designed to operate without it, is always tricky. This is why keeping the new components as small and the least intrusive as possible is our only option.

This is a task we must overcome in multiple ways. The first being figuring out how to incorporate the new sensors into the old design of the mill/mixer. There is limited useable space on the inside of the mill when it is still, once it begins to operate this space is further limited. This is due to the highspeed movements of the mills vial shakers. The force generated by the shaking is transferred to movement in the X and Y plane as well as the entire assembly is mounted to a table that allows fixed movement in both those directions.

Our sensors need to be small enough to fit in confined areas and strong enough to detect accurate readings from a distance. Additionally special care must be taken to ensure all wires are routed and secure in a manner that keeps them clear of all the moving parts inside the mill. One loose wire could cause a chain reaction of damage to our project and the mill itself. We need to limit the size and number of wires placed near the moving parts which will be difficult as most of the measurements we are interested in reading are near said moving parts.

We have more freedom with the placement of some sensors than others. However, a few of our sensors will require precise placement in extremely confined places such as In between the motor and the flywheel. These will be the hardest to mount as routing the wires will need to be done carefully as previously mentioned.
We have done our best to balance size, cost, and accuracy throughout our design process. The size constraints we have are extremely strict, as any violation could result in irreparable damage. Likewise, violations of the size constraint can create a safety issue further increasing the importance of adhering to the size constraints.

Any Sensors that must be mounted in tight places will take priority. Size will be considered ahead of cost in these situations as size constraints are directly tied to safety in our project. Any sensor that can be mounted to the outside of the mill will be done so as they are not subject to the same limitations. It is in these areas we can balance additional cost incurred by adhering to size constraints.



Above is a Solidworks CAD of the current ball milling machine front panel. For our Senior Design Group’s ball milling machine project we must take into account the size of the front panel – especially the screw placements. This consideration will be important to our design constraint for multiple reasons. Firstly, our replacement panel – a membrane keyboard – must be similar in size to fit and cover the same surface, minding the size and button placement to fit within this space. Similarly, and the second significance of noting the size of this panel, will be for the design of the printed circuit board to be mounted on the other side of the panel. Since this board needs to be screwed into the existing ball milling machine, both the membrane keyboard and the printed circuit board most importantly need to focus on the screw placement as this placement is hard fixed to the ball milling machine to be secured in – the much less attractive alternative if this isn't considered would be to machine new screw holes in the front of the ball milling machine to fit the redesign for the board.

This is being avoided also for multiple reasons. The first reason is that this would create extra labor not just in the form of machining the new screw holes but also in having to make new designs for the board and membrane keyboard. Designing with new screw locations will also defeat a goal of this project: to make this upgrade to the ball milling machine a 'plug and play' product. The current size of the ball milling machine is currently standard for most other ball milling machines, and of course for the same brand/producer of the Blair Research Group/s ball milling machine. To redesign this and make it custom made would subvert the goal of allowing another research group or customer to be able to have our designed product and easily and quickly utilize it for their own ball milling machine.

4.3.3 Manufacturability Constraints

Manufacturability constraints play a large role in the decision making progress of several components of our project. There are several different aspects of our project that are affected by such manufacturability constraints. We need to ensure that we will be able to build our project, and that it will be possible to fix any errors that occur during the process.

Our number one concern when it comes to manufacturability is that we will be assembling our boards by hand. Since we will be using many surface mount parts, hand assembly can be quite difficult and time consuming. This becomes a major part of our decision making process when selecting parts. While smaller parts are very nice in terms of their reduced consumption of board area, they can be much more difficult to work with. Hence, in our part selection section of this report, we frequently chose parts that were a little bit larger than absolutely necessary. Even though we often made these decisions, there were some cases where it is difficult to select parts that will be easy to work with.

One such case is with the accelerometer component. For some reason, manufactures do not like to offer accelerometers in larger packages. During our searches for larger packages, we found that unfortunately the vast majority of accelerometers currently available on the market are only available in LGA packages. This is an extremely compact package with twelve pads that are completely below the chip. There are some slightly larger accelerometers with external pins in packages similar to SOP available from suppliers like Digikey, but these accelerometers are all intended for applications in the automotive industry. We are not doing anything overly complicated with any of our sensors, so for components like the accelerometer we would be willing to use older parts intended for automotive industry use. In fact, we would be happy to use components that have been extensively tested and are known to be reliable.

Unfortunately, it is not that simple. These parts all require orders in large quantities. Since we only are using one of each sensor on the board, and we currently intend to create one final product with a couple of iterations on the way, we only need a few sensors. We are not able to buy these bulk automotive sensors, and are therefore stuck with the LGA footprint for some components like the accelerometer. Soldering these components by hand is a more difficult process that will increase the potential for making mistakes. In order to simplify the surface mount soldering process and reduce the likelihood of mistakes, we plan to purchase a stencil for our printed circuit board. Unfortunately one of the manufacturability constraints attached to LGA is that stencils used for the small pitch should have a thickness of 1mm or less to decrease the chance of manufacturing errors. These small thickness stencils are more expensive than the standard 1.3mm stencils from most PCB manufacturing shops. These manufacturability constraints of LGA, combined with the availability constraints of the parts, will require us to spend more money on certain parts of the project.

Despite our best efforts it is very possible that we will make some mistakes during this process, so we need to ensure that our manufacturing process allows us to detect and rectify the problems. This requires that we will need to add extra pinouts or testing points in order for us to see what has gone wrong. We may want to also ensure that traces are a little more separated from each other than strictly necessary so that we can more easily cut traces and insert extra wires for any required fixes. Hopefully we will not need to make such modifications to the wiring on our circuit boards, but it is likely that there will be some amount of errors in our first iteration of the system.

4.3.4 Sustainability Constraints

Sustainability is another major concern in this project. We want our system to be long-lasting and reliable so that it is not necessary to replace it in the near future. The ball milling machine is an important component of the work done by the Blair Research group. When in operation, it runs for long periods of time at high intensity, shaking its contents at incredibly high speeds. It is a rather harsh environment for electronic components. There is a large amount of vibration in the machine, the power supply is noisy when many machines in the lab are operating, there are occasional, but not incredibly infrequent power outages, and the air conditioning in the building has previously failed, leading to humidity above 90% in the room.

Ensuring that our new ball milling machine controller will survive this environment is not the simplest task. That is why a large number of the sensors being used in this project are intended to decrease the risk of pre-mature component failure. Starting with the humidity sensor, we will be monitoring for any cases where the air conditioning may fail again in the future. We do not want to operate our electronics in conditions where humidity is that high, so we will implement an automatic shut down procedure that will help ensure the sustainability of the system in such cases. Such humidity can be bad for both an operating motor and for the controller, so we will shut down both parts of the ball milling machine. Before completing shutdown, the controller will also send a notification to the user, so that they can be made aware of the detrimental environmental conditions that the ball milling machine is currently experiencing.

We also intend to protect against mechanical failures in the system due to the high vibration occurring in the ball milling machine. We will have an accelerometer on the main platform that will allow us to detect how much vibration is occurring at any given moment. We will be able to use this information to prevent any mechanical failure from cascading. If, for example, one of the belts breaks, and only part of the ball milling machine is receiving power from the motor, we expect that this scenario would create a noticeable change in the vibration patterns that the accelerometer is experiencing. We would also expect that this change in operating conditions could create additional wear on the belt that is still functioning. In cases such as this, we would use the information from the accelerometer to automatically stop the motor, and report to the user that something has gone wrong in the ball milling machine. Using our controller to prevent part breakages from cascading to additional wear on other parts is one way that our new controller will help improve the sustainability of the ball milling machine.

To help isolate the electronic components from the potentially poor power quality that the ball milling machine experiences, we will use on board DC-DC power conversion. Such power conversion provides a layer of separation between the AC power source that is affect by power draw from the motor, and the more sensitive electronics. That is not to say that the this will completely isolate the electronics from poor power quality, but it will help reduce the impact. The power conversion circuitry itself will, however, be exposed to the poor power quality. This is acceptable because the electronic components that will make up the power conversion circuitry will all be durable, and either large or commonplace.

These components will be relatively easy to replace. Replacing larger components such as the chips or inductors used in power conversion would likely require ordering new parts, but their larger size makes the physical process of desoldering and replacing the component much simpler. Other components that are part of the power conversion circuitry are extremely common, such as resistors and capacitors. These may be slightly harder to physically replace, but will likely be components that the Blair Research group already has on hand. We will also ensure that components involved in the power conversion circuitry are no smaller than 0805, so that replacing them will be feasible.

Even if other parts of the system fail, our hope is that it will be simple to find and replace the components that are no longer properly functioning. For the purposes of finding the failing components, in many cases the users should be able to simply look at the readings being reported by the sensors through the home assistant application. If any sensors fail, they will have obviously incorrect reporting. Other places where components fail may be harder to detect, such as if one of the components in the power conversion circuit fails. For a case like this, we intend to have test points on the PCB. We will use these when initially testing our system, and they will also be available for if any components fail. This is not necessarily the most user-friendly way to debug a failure, and we hope that it does not become necessary. Once the failing component has been identified, there are certainly limits to the ability to easily replace it. We will be using many surface mounted parts that can be a bit difficult to swap, such as the LGA 12 accelerometer. In cases like these, we would like to have alternatives to physically de-soldering and replacing the part. For example, we may provide another header that allows the user to directly tap into the I^2C bus. This would provide the opportunity to simply attach a different accelerometer breakout board, which is much easier to do, especially since third-party breakout boards using the same accelerometer as us are available.

Overall, the sustainability of this system is a major focus in our project. We want our new smart ball milling machine controller to help monitor the health of the machine, and make sure that any problems are found quickly, so that they can be addressed before the size of the issue grows. We also want to make sure that fixing any part failures on the controller itself is as simple as possible. These aspects of the project will help ensure that the ball milling machine is sustained for as long of a lifetime as possible.

4.3.5 Power Constraints

The ball milling machine is a typical appliance that is operated off house power, being plugged in to a typical American standard outlet at 120V 60Hz. The motor of the ball milling machine requires 115V to operate.

4.3.6 Supply and Availability Constraints

We have taken care to identify the most important components to our project so that we can compare which one best meets our needs. While all the components required to complete our project are currently available with sufficient lead times, most can ship immediately. Conducting thorough research on the components is critical to the success of our project for multiple reasons. The more research conducted before designing the higher the likelihood that the component will work as intended. This takes time to do properly, given that we are currently living in a highly sensitive geopolitical climate supply chains cannot be assumed to maintain stability. Due to the fact these isolated conflicts have the potential to escalate and spread to other regions at any moment, we must plan for the worst-case scenario. This will ensure the successful completion of our project. Should a global conflict arise, we must consider supply chains and the availability of components as a luxury we may not always have.

During times of global conflict governments often redirect supply chains to support the defense of their respective countries. Since many of our nation's defenses rely on sophisticated technologies securing the components to replenish and manufacture these technologies becomes a national security concern. Additionally, the current world relies on international trade of all resources. Global conflicts affect not only trade agreements, but also the ability to physically import and export goods. As we recently learned from the COVID pandemic parts can easily go from being readily available to having months of lead times.

We are currently on a deadline to complete our project 6 months from now. Any lead time even approaching this could potentially result in an incomplete project. Everyone on the team has experience with this firsthand. During our junior design we experienced global supply chain issues for many of the components needed to create our projects. As a result, we were only able to prototype our projects, never actually building them. We not only have a commitment to our sponsor but an academic requirement to complete a fully complete and functional product as well.

All these things considered we have a constant looming supply and availability constraint we can't ignore. We must research and design our project with near flawless accuracy. Any significant redesign may not be possible for our project as it was in the past for other groups. Thus, it is imperative that we Identify and secure any an all components as soon as possible. Another consideration is the need for backup components. These will need to be ordered as well to account for damaged components and to manufacture at least one complete backup device if necessary.

We are conducting timely but thorough research to balance the uncertainty of supply chains with designing a functional product. In the best-case scenario, our supply chains remain as they are at this moment, and we are afforded the ability to make last-minute component purchases if need be. But as it stands now our group is going to err on the side of caution and treat the supply chains and the availability of components as highly unpredictable.

4.3.7 Safety and Reliability Constraints

Aaron

4.3.x More constraints (If needed)

5 Comparison of ChatGPT and Similar Platforms

5.1 ChatGPT/Bard Limitations

The introduction of home use AI models such as OpenAI's ChatGPT and Google's Baird have changed the way that most students conduct research and complete certain tasks almost overnight. The introduction of such a powerful tool capable of answering questions directly, writing code, translating different languages, and even generating songs or stories. On the surface this is a gift to humanity, providing unprecedented capabilities to virtually anyone with access to a computer or smartphone. The caveat is that they must be used properly if they are to be accurate or effective.

Unlike their title implies AI models are slightly misleading even in name. They aren't true Artificial Intelligence so much as a highly advanced machine learning model. They still rely on preprogrammed algorithms to generate text responses based on patterns. This is the first obvious limitation like all programs and algorithms they are only as good as their programming allows them to be. While previous versions lacked the ability to access current information this limited the responses to information from the year prior. The newer version of ChatGPT allows it to access a browser feature that can access up to date articles for information. Googles Bard allows this also though it is based on googles search algorithm a limitation in it of itself. This is still not perfect for either model since they are both still limited to publicly available knowledge and incapable of creating new information.

Since these programs are designed to provide answers in a certain manner, they are naturally biased to answer in a way that its' programmer deems suitable. For instance, when asking ChatGPT a controversial question it answered including the direct sources for the information it was providing. It also provided a note of caution as controversial topics are often disputed. In any ChatGPT provided a valid response with sources and a note of caution allowing the user to use their own discretion. When asking the same question Bard would refuse to provide an answer and direct you to the google search engine. This is far from acceptable for a research tool as many topics of research can be controversial in nature. When comparing the two sources for bias I would say that Bard is more biased than ChatGPT as the latter still attempts to provide you with the information you as best as it can with a note of discretion.

Another limitation is that they lack a complex understanding of context. As a language model they do not possess any emotional intelligence or understanding of complex social cues that can alter meaning and intent. This can lead to irrelevant or inappropriate responses. This can be looked at in two ways a limitation of the model or a limitation of the user. Like all tools they require proper use to avoid harm and digital tools are certainly no exception. The quality of your responses depends on the quality of your question and the limitations of the model. If a question is poorly articulated for example can generate a false response or fail to generate a response at all.

The response may be limited by the complexity of the question. The models struggle with generating longform responses resulting in repetitive information or sometimes freezing or errors. To avoid this, I recommend parsing the questions into smaller more specific questions that build and answer the larger question allowing for a wider breadth of sectioned information. This can be difficult for inexperienced users which will limit the range and quality of their responses.
To be fair I asked each to list their own respective limitations and the results were relatively honest.

Although, I would argue that some of what ChatGPT considers “limitations” I would consider safety protocols such as lack of access to personal information, interaction with the physical world through IoT devices. The rest of the list can be accessed through the following link https://chat.openai.com/share/dae58b62-0ed0-4923-8d25-a870c1a57d4b.
Similarly for Bard the following link can be used to access the full response https://g.co/bard/share/ee25d59916b6. In comparing both responses I would say that ChatGPT’s response was more specific. Bard was short and quite vague in its own response three out of five all essentially say it has difficulty following instructions. It does however parallel some of the ChatGPT limitations. They both admit to having difficulty executing complex tasks or instructions. Overall ChatGPT’s response was far superior to bards. It provided actual informational limitations that can help influence how you can use it more effectively. While Bard took the “enter at your own risk” approach.

5.2 ChatGPT General Use Cases

In the past I have used both ChatGPT and Google’s Bard with varying results. My initial attempt to use ChatGPT was asking it to explain a math concept. After performing the calculations by hand I wondered if ChatGPT could perform the calculations for me if I provided it with the variables. It replied that it could with confidence and proceeded to give me the wrong answer. When I notified it of the possible error in the calculations it apologized and agreed that it made error, then recalculated and provided a new wrong answer.

This is where I changed by approach instead of asking it to solve the problem at once. I gave it individual components of the equation it provided and had it solve them one step at a time. The calculations ended in the same error that it had initially provided, which was off by a factor of 10^3. By this point I was beginning to second guess my own knowledge of the metric prefixes. I again asked it to redo the calculations but to explain every step along the way. It proceeded to explain the steps correctly as it performed them correctly up to the final step. Once it reached this point it explained everything in the correct units and again provided the wrong answer off by the same factor. I then proceeded to spend the next few minutes convincing it that it continues to make the same error and why. Then it agreed with me and finally provided the correct answer. I then put the same problem into google bard to which I was given an even worse answer than what ChatGPT provided. Having spent quite a bit of time and frustration with ChatGPT I immediately decided that for complex or important calculations I would never fully trust an AI model to set up and perform any equations in the correct manner.

Both Bard and ChatGPT explained the concepts and how to perform the calculations correctly, which is arguably the most important part. However, the confidence with which it delivered the incorrect answers was disturbing. Where I found it most beneficial was in helping me tailor google searches. It has become frustratingly difficult to sift through the ads and pages of irrelevant links before finding what your actually looking for. It saves time is when researching new topics you have very little understanding of. I found that for general information or explanations ChatGPT and Bard both provide time saving accurate information that can be used to help you narrow searches.

Taking this approach works best due to the information you receive from ChatGPT and Bard being general or arbitrary in nature. They are both capable of expanding further on various topics. However, more detailed and up to date information are best obtained through traditional sources. Due to the limitations in data that the AI models have access to and the amount of contradictory information on the any give topic. By using the information acquired through your preferred AI model you can make more specific searches in google helping you find more up to date and detailed resources. I don't think either Ai model is suitable a replacement for more established tools or resources. They can however, serve in the role of an assistant or tutor.

Another area I found the AI models useful is in code analysis. The models can't create full functioning codes, but they can explain how to structure simple tasks and help you find errors in your own code. I had to create a cache simulator as a project in another class. The program needed to be able to scan a trace file of a couple million entries. My code worked perfectly with the smaller test trace file and both Ai models were able to identify the programs' purpose without me having to tell them.

However, upon running the full trace files It would return the correct values for one file and the wrong values for another. I decided to scan the trace file line by line till I saw the problem out of nowhere the hex value size shot up by 4 digits making the int value too large causing overflow. I asked what ChatGPT what would be an appropriate size int to the value it informed me to use long long int and recommended that I allocate memory to the to help protect against the overflow issue I was having. I had never allocated memory before so I asked ChatGPT to explain how to do it and it provided me with two options calloc and malloc explained the differences between them and provided a skeleton with labels on the proper syntax.

I believe this will be the most beneficial use to our senior design team, as we are entirely comprised of EE students. Our coding skills are strong enough to accomplish most of the task required given enough time. The fact that we are constrained in time having a tool that can help you troubleshoot and recommend ways to improve your code will greatly improve our ability to meet the time constraints and learn more sophisticated ways of accomplishing tasks that we are currently performing more efficiently.

5.3 ChatGPT and Bard for Website Design

Everyone on our senior design team is an electrical engineer. It has not been a problem for the purposes of project work, because several of us enjoy programming anyway, but the kind of programming that we have experience with is largely lower level programming. We all took embedded systems, so we're all good with C, a couple of us have used C++, and we have varying levels experience in scripting languages like Python mixed in. However, none of us have experience with website design. For the purposes of Senior Design 1, it seems like a simple black text on white background webpage might be sufficient for the time being, but since tools like ChatGPT and Bard exist, we figured we might as well see if they can help us by sprucing up our basic webpage.

We started by creating a simple, purely HTML webpage. We did not want generative artificial intelligence to be responsible for any of the content, we wanted to use it solely for formatting, color, and appearance in general. The barebones HTML webpage included all of the text that is on the current version of our website, along with some simple HTML tags to give the generative AI a starting point for understanding how the website is organized. These included header tags, paragraph tags, and line breaks. We also included some $ensp spaces to try to align our group member's "Electrical Engineer" labels and our reviewers' departments.

Since it seems to be more of the name brand, we started with ChatGPT. We used the free version of ChatGPT that is built on GPT 3.5. We started by priming ChatGPT with the question "Can you add CSS to a simple HTML webpage to make it look nicer?" It requesting with "Of course!" and an example of some CSS tags. We then fed it our human-written HTML and asked ChatGPT to improve it. It gave us a webpage with the same text, some blue background coloring on the headers, and additional centering. The colors were a little harsh, so we asked ChatGPT if it could "tone them down a bit", and it requesting with new CSS that had a nicer blue color. We also asked it to add a picture after the title, and align the places that say Electrical Engineer.

This is where we hit our first snag with ChatGPT. It successfully added a placeholder for the picture, where all we had to change was the image filename, but the way it tried to align the group members with spans did not work. We simply told it that its solution failed, and asked ChatGPT to try again, and it changed its approach. This time it used tables, and the alignment worked. In addition to aligning the “Electrical Engineer” titles, it also aligned the reviewer’s departments. This was not something we had asked it to do, so it could be indicative of ChatGPT making unnecessary changes to inputs. However, we were about to ask ChatGPT to make that change, so it may just indicate that large scale transformers are very good at modeling context.

Our final request for ChatGPT was to left-align the group member’s names. This showed us why it’s good not to rely on ChatGPT entirely, because there was an error generating the response, and the regenerate button did not work at that time. Thankfully, it had been kind of lazy to ask ChatGPT to do the left-alignment for us, because after actually looking at the CSS it generated, it was incredibly obvious that we could make that improvement by changing the text-align parameter from “center” to “left”. Out of curiosity, I clicked regenerate after writing this section, and ChatGPT produced new output but it did not work right. It created a new class with the “text-align: left” setting and then added that class to all of the elements in the table, instead of simply changing the existing table’s text alignment setting from center to left. This also meant that the new ChatGPT output did not align our reviewer’s departments, while our human-made change did. I asked it to try again, and this time it succeeded by using divs instead of changing the alignment parameter. When I asked ChatGPT to also align the professor’s departments, it failed again, and also removed the left-alignment on our group member’s names.

Overall ChatGPT was an incredibly useful tool for creating the initial version of our website. As shown in figure 5.2.1, it allowed us to take a very simple black text on white background style webpage and add color, add images, and improve the formatting, without having to dig through a bunch of random CSS tutorial websites. I was also impressed that ChatGPT never changed the text we provided it. It modified some of our HTML tags themselves, but it never touched anything that was between body tags. While this webpage was quite simple, we were happy that the output from ChatGPT required only minor changes and bugfixes.

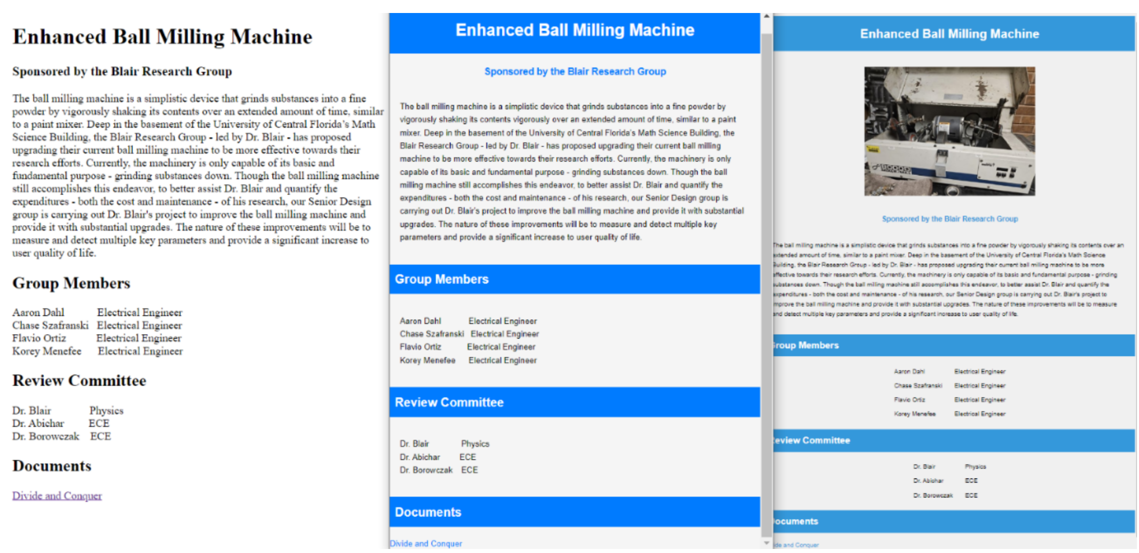


Figure 5.2.1: ChatGPT Results – Original, ChatGPT First Output, With Additional Prompting

Next, we tried Bard, and had an incredibly different experience. We wanted to keep the comparison fair, so we started with the same type of primer question before giving Bard our HTML source and asking it for improvements in the second prompt. The response from Bard was lackluster. It provided a few CSS snippets that it suggested adding to our code. I want to stress here that Bard provided CSS code out of the context of our webpage, when ChatGPT simply added it to the correct part of our code, and did not change anything else. Additionally the CSS “improvements” suggested by Bard did not make a noticeable change to the webpage.

We then asked Bard to “add some color and make it look nicer”. The new output from Bard made the background a shade of light gray, and converted the text to a darker shade of gray. At this point, we were pretty confident that we were not going to be able to have Bard create a better output than ChatGPT on its own, so we gave it the final result from ChatGPT and asked Bard to improve that instead. Bard suggested new CSS snippets, once again out of context of the rest of the webpage, that simply removed the coloring from the ChatGPT output, uncentered the image, and decreased the spacing between the group members’ names and majors and the reviewers’ names and departments.

The initial result from Bard, the result from ChatGPT followed by Bard, and the result of Bard with the ChatGPT starting point plus additional prompting are shown in figure 5.2.1 below. The text in all three versions is the same, the differences are all in the formatting. We feel that ChatGPT produced results that were far superior to Bard’s results. ChatGPT requesting to our prompts with the requested changes, fixed most of the errors, and required only minor tweaks and bug fixes. Bard did not do what we asked it to as accurately, and also required that we integrate its CSS snippets with our existing HTML. We were able to get Bard to restore what was essentially the ChatGPT output by requesting more color, black text, and an uncentered image, but it is still subtly worse. It is a bit difficult to see in the figure, but there is a white background to some of the table cells that we were unable to convince Bard to remove. We were not able to generate a comparable result with Bard alone.

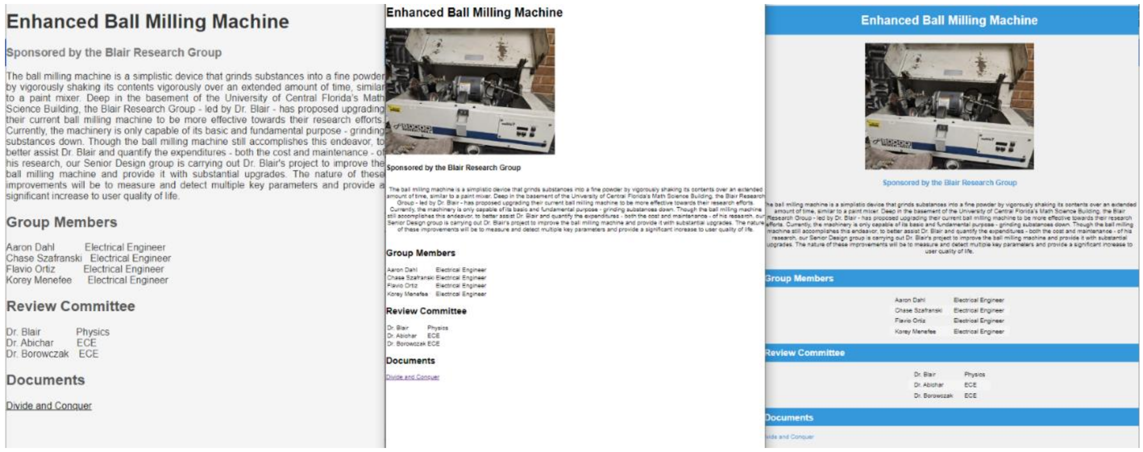


Figure 5.2.2: Bard Results – Bard First Output, Bard with ChatGPT Start, Bard with ChatGPT Start and Additional Prompting

6 Hardware Design

6.1 Reverse Engineering the Existing Controller

One of the major goals of this project is to provide a drop-in replacement for the existing ball milling machine controller. We want it to be as simple as unscrewing and unplugging the original board, and putting ours in its place. In order to achieve this goal, we need to ensure that our board is compatible with the existing interfaces that the ball milling machine has with the other components. This process involves identifying the headers on the controller, as well as determining what each pin in the header does.

The existing ball milling machine controller interfaces with three different components of the ball milling machine through two different pin headers. These three components are the safety interlock, power supply, and motor relay. The power supply circuit board is connected to the mill controller printed circuit board through a 16 wire ribbon cable, that is terminated with a 0.1 inch pitch eight by two pin header. This plugs in to the controller with a 3M 3408 connecter header that helps create a secure connection by looking the ribbon cable in place. The second connector is a six pin header with 0.1 inch pitch that connects the controller to the safety interlock switch, the 115 volt ac power source, the neutral, and an output to the motor relay. The existing milling machine controller with the six pin connector on the left, and the sixteen pin connector on the right is shown in figure 6.1.1 below.

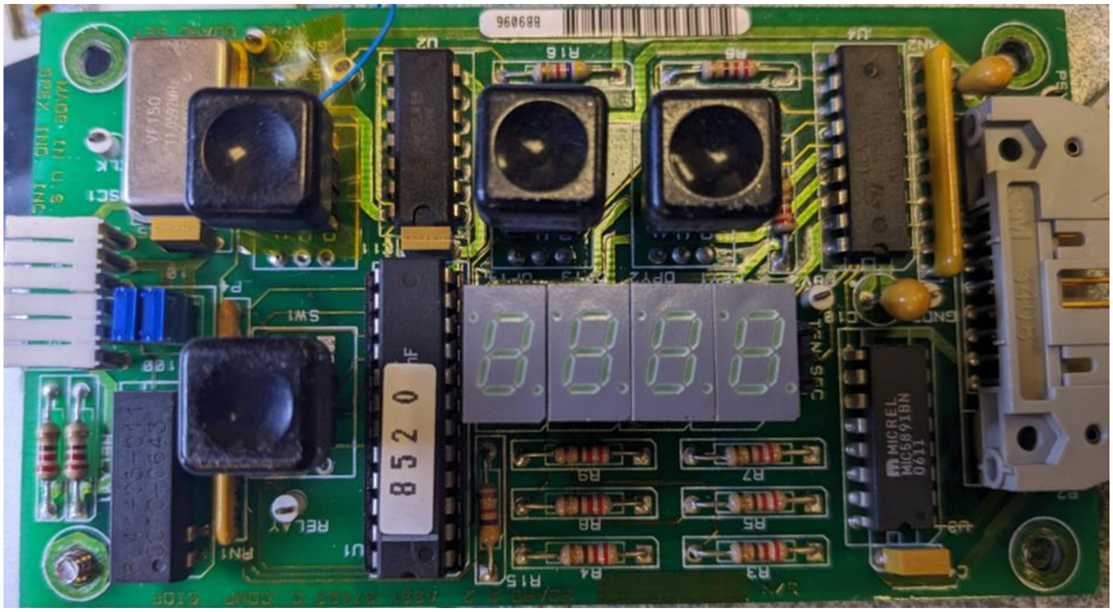


Figure 6.1.1: The Existing Ball Milling Machine Controller

We started by examining the four wires attached to the six pin header prior to more thoroughly disassembling the ball milling machine. The two of the four wires are attached to the safety interlock switch, one to the motor relay, and one to the 115V main power source. Tracing the on board connections revealed that the wire from the motor relay is connected to the smaller Cocto Technologies 8L41-05-011 relay that can be seen in the bottom left of figure 6.1.1. This relay is also connected to the 115V pin. The power supply for the motor uses a two relay system. The first relay on the board is used to switch a larger, off-board relay that provides the actual power for the motor. This means that the current flowing through the 115V trace on the main controller board is negligible. The large, off-board relay that powers the motor is an Omron G7L-2A-BUBJ-CB, which has a coil current of between 17.0 and 20.4 mA when the coil voltage is between 100 and 120 VAC [53]. It is possible to switch this relay with a smaller voltage, but doing so requires a much larger current. It requires about 1.9W to switch the coil, so our simplest option is to maintain the current design, especially because this solution will provide the simplest install of our new board, with no need to change the wires attached to the six pin connector. In addition to a replacement for this Cocto technologies relay, in our design we will need to use a transistor to switch the on-board relay because the ESP32 is not able to drive enough current to switch even the small surface mount relay that will be on the board.

We were able to determine what the wires on the 16 pin connector carried by examining the datasheets of the onboard parts, following visible on board traces, and verifying connections with a multimeter. The bottom of the ball milling machine controller is shown in figure 6.1.2, with the 16 pin header on the left. While this header has 16 pins, it doesn't support 16 different functions. Several of the pins are directly connected to each other, these pins are in boxed groups in figure 6.1.2.

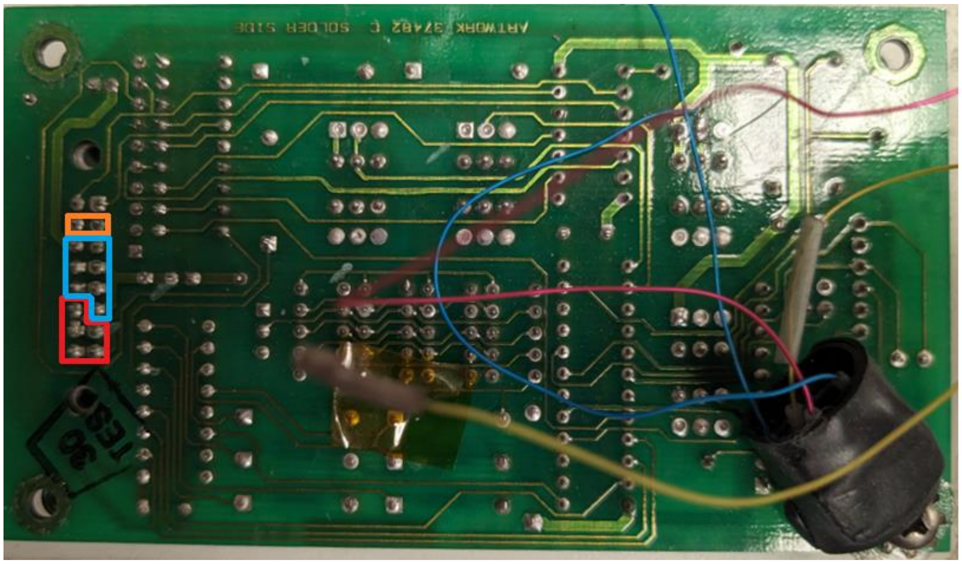


Figure 6.1.2: The Bottom of the Existing Controller

The top row of pins is used to drive the mechanical interlock latch system. The left pin inputs the active low reset signal from a National Semiconductor LM2925 low dropout voltage regulator that is on the power supply PCB. This reset signal is low when the voltage output from the regulator is unstable. This signal then goes to an input pin of the on board Texas Instruments SN74LS04N inverter. The corresponding inverter output is then fed through a pair of the Darlington array pins, before being sent back out the top right interface pin. If the inverted output is low, this signal grounds the cathode of an optoisolator on the power supply circuit board, which triggers a thyristor on the same board, which ultimately supplies power to the interlock latch, preventing anyone from opening up the cover while the ball milling machine is running. This is an important safety feature for us to replicate. The ball milling machine is moving extremely quickly when in operation, and has the potential to cause serious damage if the containers are exposed while moving.

The second row of pins (the orange group), provides a 5V input to the resistor network that is directly adjacent to the header. This is the yellow component that runs alongside the STMicroelectronics Darlington Array, and provides it with the voltage inputs. It also provides power to the other integrated circuits that accept 5V. We do not intend to use this 5V supply as the primary power source in our project. The regulator that provides this 5V supply is the LM2925 mentioned previously, which has a maximum output of 750mA [52]. Transmitting information wirelessly can have a pretty large power draw, which was not something required in the original design. The ESP32 is known to draw up to 379mA under peak load [8]. Between this increased load, and the fact that we would be wasting additional power in the conversion from 5V to 3.3V logic, we have decided to use the higher voltage power source to ensure that our system is always able to be supplied with a sufficient current, even for the high power state of the microcontroller and sensors.

The last two groups of pins are for power. The blue group provides ground, while the red group of pins supplies a higher voltage power source. We measured it at 13.8V while the controller was disconnected from the interlock and motor relay, but suspect it is intended to be a 12V source under load.

6.2 Subsystem Block Diagram(s)

6.3 Schematic Diagram(s)

6.4 Simulations

7 Software Design

7.1 Flow Chart

7.2 User Interface

7.2.1 Scheduling

7.2.2 Other Features (TBD)

7.3 Publishing to HomeAssistant

7.3.2 Scheduling

7.3.3 Other Features (TBD)

7.4 Polling Sensors

7.5 Timekeeping System

7.5.1 Flow Chart

7.5.2 Scheduling

8 System Fabrication and Prototype Construction

9 System Testing
9.x Sensors
9.x Membrane Keyboard

10 Administrative Content

10.1 Budget Estimates and Funding

This project is being sponsored by the Blair Research Group from the Department of Physics here at UCF. Funding for the components will be provided by our sponsor. Initial estimates for the budget of each section of our project are provided in Table 2.

The final version of this project will be used to control one of the Blair Research Group's ball milling machines. Additionally, our project will be designed with the intention of creating a system that is easy to integrate with the other existing ball milling machines owned by the research group. They will be able to manufacture additional circuit boards to connect the other machines to the Home Assistant system.

Part	Estimated Cost
PCB Shops	$50
Power Supply	$10
Antenna	$5
Raspberry Pi	$45
Humidity Sensor	$5
Temperature Sensor	$20
Current Sensor	$15
Magnetic Field Sensor	$5
MCU	$3
IR Sensor	$5

Nonvolatile Storage Module	$2
OLED	$37
Housing Design	$20
Total	$222

Table 11: Bill of Materials

10.2 Milestones

In Senior Design 1, we plan to go beyond completing the 120 page design paper. Table 10.2.1 shows our current project timeline for the project. We plan to select our major components before the second week of October, so that we can spend time in October and November on circuit design and breadboard tests for prototypes of the various subsystems. This will allow us to iron out a lot of the issues in our design by the end of this semester, so that we are able to order the first version of our PCB early, and can hit the ground running in Senior Design 2.

Task	**Timeline**	**Status**
Senior Design 1 and Documentation		
Project Idea Brainstorming	August 28th, 2023	Complete
Project Selection	September 1st, 2023	Complete
Divide and Conquer Document	September 15th, 2023	Complete
MCU Selection	October 6th, 2023	Researching
Sensor Selection	October 6th, 2023	Researching
60 Page Checkpoint	November 3rd, 2023	Complete
Schematic Capture	November 3rd, 2023	Not yet started
Subsystem Breadboard Prototypes	November 17th, 2023	Not yet started
PCB Layout	November 28th, 2023	Not yet started

120 Page Final Document	December 5th, 2023	
SD1 Committee Review Meeting	TBD	
Senior Design 2		
Order Initial PCB	December 15th, 2023	Not yet started
PCB Assembly	TBD	Not yet started
Integrated Baseline Code Ready for Testing	TBD	Not yet started
PCB Redesign and Reorder	TBD	Not yet started
PCB Assembly #2	TBD	Not yet started
Testing and Bug Fixing	TBD	Not yet started
Live Demo	TBD	Not yet started

Table 10.2.1: Milestones

10.3 Table of Work Distribution

Table 10.3.1 shows our current plan for the distribution of work. We divided work into multiple subsystems that can be tested separately before being integrated in Senior Design 2. Each subsystem has a primary person who is responsible for completing the task, and a secondary person who will be assigned to assist the primary person and to hold them accountable to completing the task.

Task	Primary	Secondary
Power System Reverse Engineering	Chase	Korey
UI Design	Flavio	Korey
Membrane Keyboard Design	Korey	Chase

PCB Design	Korey	Aaron
Power Supply	Flavio	Korey
Antenna	Aaron	Chase
Microcontroller Main Loop	Chase	Korey
MCU Wifi and MQTT Programming	Aaron	Chase
Raspberry Pi Home Assistant Setup	Aaron	Chase
Home Assistant Programming	Chase	Aaron
Temperature/Humidity Sensor	Korey	Flavio
Magnetic Field Sensor	Flavio	Korey
IR Sensor	Korey	Flavio
FeRAM	Chase	Korey
OLED Driver	Flavio	Aaron
Housing Design	Flavio	Chase

Table 10.3.1: Work Distribution

References

[1] J. Scislaw, E. Reina, A. Umashankar, and A. Gilliam, “Chemical Manufacturing Process Automation,” UCF Senior Design, https://www.ece.ucf.edu/seniordesign/fa2019sp2020/g22/ (accessed Sep. 12, 2023).

[2] https://www.corelis.com/education/tutorials/spi-tutorial/

[3] https://oled.com/oleds/

[4] https://www.oled-info.com/oled-technology

[5] Jeff Tyson "How LCDs Work" 17 July 2000. HowStuffWorks.com. <https://electronics.howstuffworks.com/lcd.htm> 29 September 2023

[6] Arduino, “Nano 33 IoT” Datasheet Modified Sep. 29, 2023, https://docs.arduino.cc/resources/datasheets/ABX00027-datasheet.pdf (accessed Sep. 30, 2023)

[7] Arduino, “Nano 33 BLE” Datasheet Modified Sep. 29, 2023, https://docs.arduino.cc/resources/datasheets/ABX00030-datasheet.pdf (accessed Sep. 30, 2023)

[8] Espressif Systems, “ESP32-WROOM-32E, ESP32-WROOM-32UE” Datasheet Version 1.6, 2023, https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf (accessed Sep. 25, 2023)

[9] Espressif Systems, “ESP32-MINI-1, ESP32-MINI-1U” Datasheet Version 1.3, 2023, https://www.espressif.com/sites/default/files/documentation/esp32-mini-1_datasheet_en.pdf, (accessed Sep. 25, 2023)

[10] Espressif Systems, “ESP32-PICO-V3” Datasheet Version 1.3, 2023, https://www.espressif.com/sites/default/files/documentation/esp32-pico-v3_datasheet_en.pdf, (accessed Sep. 26, 2023)

[11] Espressif Systems, “ESP32-S3-MINI-1, ESP32-S3-MINI-1U” Datasheet Version 1.1, 2023, https://www.espressif.com/sites/default/files/documentation/esp32-s3-mini-1_mini-1u_datasheet_en.pdf (accessed Sep. 26, 2023)

[12] Espressif Systems, “ESP32 Series” Datasheet Version 4.3, https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf (accessed Sep. 30, 2023)

[13] Espressif Systems, “ESP32-S3 Series” Datasheet Version 1.7, https://www.espressif.com/sites/default/files/documentation/esp32-s3_datasheet_en.pdf (accessed Sep. 30, 2023)

[14] STMicroelectronics, "STM32G0B1xB/xC/xE" Datasheet Revision 4, https://www.mouser.com/datasheet/2/389/stm32g0b1cc-2042221.pdf (accessed Sep. 30, 2023)

[15] "Nand vs. NOR flash memory for Embedded Systems," NAND vs. NOR Flash Memory For Embedded Systems, https://resources.system-analysis.cadence.com/blog/nand-vs-nor-flash-memory-for-embedded-systems (accessed Oct. 4, 2023).

[16] P. Thanigai and W. Goh, "MSP430 FRAM Quality and Reliability", Texas Instruments, Application Report SLAA526A. Accessed: Oct. 4, 2023. [Online]. Available: https://www.ti.com/lit/an/slaa526a/slaa526a.pdf

[17] M. Durlam et al., "Toggle MRAM: A highly-reliable Non-Volatile Memory," 2007 International Symposium on VLSI Technology, Systems and Applications (VLSI-TSA), Hsinchu, Taiwan, 2007, pp. 1-2, doi: 10.1109/VTSA.2007.378942.

[18] J. Shepard, "How to use MRAM to improve reliability, lower latencies, and reduce power for Edge Computing," DigiKey, https://www.digikey.com/en/articles/how-to-use-mram-to-improve-edge-computing (accessed Oct. 5, 2023).

[19] New Haven Display, "NHD-2.7-12864WDY3-M" DataSheet" Revision 7, https://newhavendisplay.com/specs/NHD-2.7-12864WDY3.pdf (accessed Oct 1,2023)

[20] Espressif Systems, "ESP32-S3-WROOM-1, ESP32-S3-WROOM-1U" Datasheet Version 1.2, 2023, https://www.espressif.com/sites/default/files/documentation/esp32-s3-wroom-1_wroom-1u_datasheet_en.pdf (accessed Oct. 10, 2023)

[21] Soldered Electronics "Parallel vs Serial Communication" https://soldered.com/learn/parallel-vs-serial-communication/ (accessed Oct. 10, 2023)

[22] Total Phase inc" The Differences Between I2C and SPI (I2C vs SPI)" https://www.totalphase.com/blog/2021/07/i2c-vs-spi-protocol-analyzers-differences-and-similarities/ (accessed Oct. 10, 2023)

[23] B. O'Hara and A. Petrick, IEEE 802.11 Handbook:: A Designer's Companion. New York, NY: IEEE, 2005.

[24] Fujitsu Semiconductor, "Memory FRAM - MB85RS128TY" Datasheet DS501-00050-1v0-E, 2017, https://www.fujitsu.com/uk/Images/MB85RS128TY-0050-1v0-E.pdf (accessed Oct. 11, 2023)

[25] Infineon Technologies, "128-Kbit (16K × 8) Serial (I2C) F-RAM - FM24V01A " Datasheet 001-90869 Rev. *K, 2018, https://www.infineon.com/dgdl/Infineon-FM24V01A_128_KBIT_(16K_X_8)_SERIAL_(I2C)_F-RAM-DataSheet-v12_00-EN.pdf?fileId=8ac78c8c7d0d8da4017d0ec94a3541b4&utm_source=cypress&utm_medium=referral&utm_campaign=202110_globe_en_all_integration-files (accessed Oct. 11, 2023)

[26] Fujitsu Semiconductor, “Memory FRAM - MB85RS1MT” Datasheet DS501-00022-6v0-E, 2016, https://www.fujitsu.com/uk/Images/MB85RS1MT.pdf (accessed Oct. 11, 2023)

[27] STMicroelectronics, “High-performance ultra-low-power 3-axis accelerometer with digital output for industrial applications” Datasheet DocID027250 Rev 3, 2015, https://www.st.com/content/ccc/resource/technical/document/datasheet/24/b9/50/c5/61/ae/4d/58/DM00150114.pdf/files/DM00150114.pdf/jcr:content/translations/en.DM00150114.pdf (accessed Oct. 16, 2023)

[28] Memsic, “Digital Three-Axis Accelerometer” Datasheet MXC6655XA Rev A, 2020, https://mm.digikey.com/Volume0/opasdata/d220001/medias/docus/495/MXC6655XA_Rev_A_1-22-20.pdf (accessed Oct. 16, 2023)

[29] Kionix, “Tri-axis Digital Accelerometer Specifications” Datasheet KXTJ3-1057 Rev 5.0, 2019, https://kionixfs.azureedge.net/en/datasheet/KXTJ3-1057-Specifications-Rev-5.0.pdf (accessed Oct. 16, 2023)

[30] Hardkernel, “ODROID-C4” Specs, https://www.hardkernel.com/ko/shop/odroid-c4 (accessed Oct. 17, 2023)

[31] Raspberry Pi, “Raspberry Pi 4 Tech Specs”, https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/ (accessed Oct. 17, 2023)

[32] Home Assistant, “Home Assistant Green”, https://www.home-assistant.io/green/ (accessed Oct. 17, 2023)

[33] Github, “Documentation/boards/hardkernel/odroid-c4.md”, https://github.com/home-assistant/operating (published Mar. 13, 2023)

[34] Open Energy Monitoring, “CT sensors - An Introduction”, https://docs.openenergymonitor.org/electricity-monitoring/ct-sensors/introduction.html (Accessed Oct 21, 2023)

[35] Elkor Technologies “Introduction to Current Transformers”, chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.elkor.net/pdfs/AN0305-Current_Transformers.pdf (Accessed October 21, 2023)

[36] Innovators Guru “Learn everything about Split Core Current Transformer”, Elkor Technologies “Introduction to Current Transformers”, chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.elkor.net/pdfs/AN0305-Current_Transformers.pdf (Accessed October 21, 2023)

[37] EE Times “Which current sensor for power measurement?”, https://www.eetimes.com/which-current-sensor-for-power-measurement/ (Accessed October 21,2023)

[38] Electronic Tutorials “Hall Effect Sensor”, https://www.electronics-tutorials.ws/electromagnetism/hall-effect.html (Accessed October 22,2023)

[39] Digikey “The Basics of Current Sensors”, https://www.digikey.com/en/articles/the-basics-of-current-sensors (Accessed October 22, 2023)

[40] IEEE Standards Association, https://standards.ieee.org/ (accessed Oct. 26, 2023).

[41] Bosch, “3-axes ultra-low power accelerometer” Datasheet BST-BMA400-DS000-11 Rev 2.0, 2023, https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bma400-ds000.pdf (accessed Oct. 27, 2023)

[42] Fujitsu Semiconductor, “Memory FRAM - MB85RC256V” Datasheet DS501-00017-3v0-E, 2013, https://www.fujitsu.com/us/Images/MB85RC256V-DS501-00017-3v0-E.pdf (accessed Oct. 27, 2023)

[43] Texas Instruments, “DRV5053 Analog-Bipolar Hall Effect Sensor" Datasheet SLIS153D, 2023, https://www.ti.com/lit/ds/symlink/drv5053.pdf (accessed Nov. 1, 2023)

[44] Allegro Mircrosystems, “ALS31300 - 3D Linear Hall-Effect Sensor" Datasheet ALS31300-DS Rev. 12, 2021, https://www.allegromicro.com/~/media/Files/Datasheets/ALS31300-Datasheet.ashx (accessed Nov. 1, 2023)

[45] Infineon Technologies, “TLE493D-W2B6 - Low Power 3D Hall Sensor with I2C Interface and Wake Up Function" Datasheet Rev. 1.2, 2019, https://mm.digikey.com/Volume0/opasdata/d220001/medias/docus/757/TLE493D-W2B6_v1.2_4-9-19.pdf (accessed Nov. 1, 2023)

[46] Forbes , “The Top 10 Limitations Of ChatGPT”, https://www.forbes.com/sites/bernardmarr/2023/03/03/the-top-10-limitations-of-chatgpt/?sh=244f3b388f35 (accessed Nov 1, 2023)

[47] The Times of India, “Google lists five ‘limitations’ of its ChatGPT rival Bard”, https://timesofindia.indiatimes.com/gadgets-news/google-lists-5-known-limitations-of-llm-based-ai-chatbots-like-bard/articleshow/98943051.cms (accessed Nov. 1 2023)

[48] Bluetooth, “Bluetooth Wireless Technology”, https://www.bluetooth.com/learn-about-bluetooth/tech-overview/, (accessed Nov. 3 2023)

[49] Bluetooth, “10 Frequently Asked Questions on LE Isochronous Channels”, https://www.bluetooth.com/blog/10-frequently-asked-questions-on-le-isochronous-channels/, (accessed Nov. 3 2023)

[49] Bluetooth, “Understanding Reliability in Bluetooth Technology”, https://www.bluetooth.com/wp-content/uploads/2020/10/EN-Understanding_Reliability.pdf, (accessed Nov. 3 2023)

[50] Medium, “Inferring distance from Bluetooth signal strength: a deep dive”, https://medium.com/personaldata-io/inferring-distance-from-bluetooth-signal-strength-a-deep-dive-fe7badc2bb6d/, (accessed Nov. 3 2023)

[51] Metageek, “Wi-Fi Signal Strength Basics”, https://www.metageek.com/training/resources/wifi-signal-strength-basics/, (accessed Nov. 3 2023)

[52] National Semiconductor, “LM2925 - Low Dropout Regulator With Delayed Reset” Datasheet, 1998, https://datasheetspdf.com/pdf-file/62633/NationalSemiconductor/LM2925/1 (accessed Nov. 8, 2023)

[53] Omron, “G7L Power Relay” Datasheet J055-E1-15, https://omronfs.omron.com/en_US/ecb/products/pdf/en-g7l.pdf (accessed Nov. 8, 2023)

[54] How to choose a relay: Electromechanical, Reed, SSR, or Fet, https://www.ni.com/en/shop/electronic-test-instrumentation/switches/what-are-switches/how-to-choose-the-right-relay.html (accessed Nov. 8, 2023).

[55] How to Select a Transformer When Designing an Isolated Buck Converter, Yaxian Li, https://alenconsystems.com/learning/new-current-war-isolated-vs-non-isolated-dcdc-converter/ (accessed Nov. 13, 2023)

[56] The New Current War: The Isolated vs. Non-Isolated DC:DC Converter, https://alenconsystems.com/learning/new-current-war-isolated-vs-non-isolated-dcdc-converter/ (accessed Nov. 13, 2023)

[57] Cocto Technologies, “8L Series/Spartin DIP Reed Relays”, Datasheet, 2019, https://www.cotorelay.com/wp-content/uploads/2014/09/8L_series_reed_relay_datasheet1.pdf (accessed Nov. 13, 2023)

[58] Littlefuse Incorporated, “Miniture [sic] Dual In-line Reed Relay”, Datasheet HE700, 2020, https://www.littelfuse.com/media?resourcetype=datasheets&itemid=5d4d8689-206e-482c-a2f9-47d233a978eb&filename=littelfuse_reed_relays_he700_datasheet.pdf (accessed Nov. 13, 2023)

[59] Meder Electronics, “SIL03-1A72-71D Reed Relay”, Datasheet 3303172171 Version 5, 2012, https://4a30d8fd18dae1bf393d-df49f4cedb726ad03ad145d2e3d346bd.ssl.cf5.rackcdn.com/datasheets/33/3303172171e.pdf (accessed Nov. 13, 2023)