ELECTRONIC FLIP SIGN

Group 16 - Blue Team
UCF Senior Design EEL 4915
04/27/2018

Richard Parise: EE (PM)
Dominick Pena: EE
John Meehan: CPE
Brennan Schild: CPE

Michael Young: Sponsor

Table of Contents

List of Figures

Table List

1. Project Narrative

The electronic flip sign is a handheld lightweight LED display that allows the user to preprogram their message via computer interface so that it can be visible for others to see with ease. This can allow the user to communicate their short messages to people around them. This device can be used for communication and entertainment. The ideal use for this device is to allow the user to communicate in situations like at a sports events, in traffic, or even as an idle display for a game room. This sign is able to hold custom messages and can display any capital letters, numbers, and even a few emojis. The flip sign can help with people in workplaces that have difficulties with hearing or communicating with others. For example, the flip sign can be used to direct construction works while they are operating loud machinery. The device will be composed of 2D surface mounted LED matrix display and is designed to be configured via micro USB and Bluetooth through a software interface. The micro USB cable provided also allows the user to charge the battery in the device.

The goal of this project is to create a device that can be used to convey messages that can be changed and used in a variety of areas. To accomplish this currently, items like paper signs or LED displays are used, but each of these have their own issues. There are various other LED signs out on the market. Some of these signs are programmable with text and icons, but they have to either be plugged in or mounted somewhere. The key reason why there is a need for our sign is due to a lack of portable LED signs. One that you can be used with one hand, on the go and without being plugged in. The problem with the current LED displays on the market is that some are not battery operated and the ones that are do not have portability in mind since most are for a sitting wall display. We are taking what the current LED signs have and making it not only portable, but also making it easy to change the display on the go as well as other features added to make using the device on the go easier and more efficient than anything on the market today.

This project's goal is not just to make a single working model. This project is to make a prototype for a production model that can then be sold at mass market. Making a product that can be used as a prototype for involves using parts that can be sourced for a much cheaper price in order to lower the bulk price. Other things that also have to be considered include the materials being used as well as the type of battery. Each of these things are done to lower production cost as well as make a better-quality product for the consumer. Below is an example of an LED sign that is already out on the market.

Figure 1: Eletechsup's 64X16 dot Matrix LED [8]

2. Prototyping and Modeling

This section includes the dimensions and constraints of the 3D model of the electronic LED sign. The design specification 10" wide by 4" tall by 1" thick(inches). Another modeling constraint is that the device must weigh no more than 8 ounces. The second photo is a mock device created that passes the size and weight constraints to give an idea of the dimensions for reference.

2.1 What Comes with the Device

- The Sign with battery and handle
- Micro USB Cable

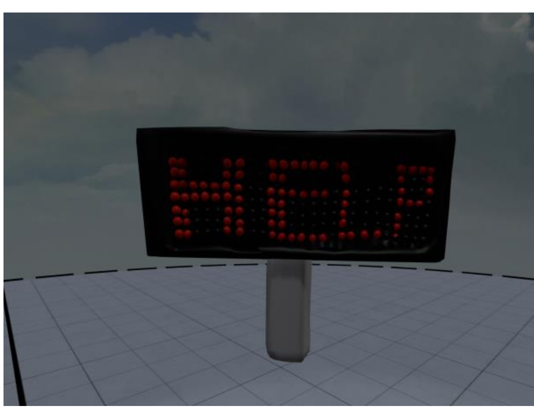


Figure 2: Front View of Simulation (3D)

Figure 3: Mockup Design

3. Design Specifications/Constraints

The Design Specifications and Constraints section is used to show both the important details that need to be met when creating the project and the types of constraints that may come up as the project is being designed and assembled. Design Specifications shows all the projects features, measurements and characteristics that the team will be working toward when design the project. The Design Constraints are the factors they could get in the way from the team's design process.

3.1 Design Specifications

The design specs are a collection of all the things that is desired of the device to be able to accomplish in the final device. The specs are for the hardware and the software and a combination thereof. On each table, there is an item that has the specification and then there is a definition of what the item is and then there is the list of what the specifications are.

Item	**What it is**	**Specification**
Dimensions	LED display	Approximately 10”x4”x1”
	Handle	Approximately 5”x1.5”x1.5”
Ambient Light Sensor	The sensor located on the front of the device	Sensor will adjust the brightness according to the ambient light
		Brightness will be adjustable manually and via sensor
Battery	Located inside the device	Able to operate the display at full brightness for 15 minutes
		Rechargeable via USB port to PC port or wall adapter
Connectivity	Ways that the device can connect with other devices	Bluetooth and USB interfaces
Weight	Total weight not including USB cable	Approximately 8 ounces

Table 1: Specifications table 1

Item	What it is	Specification
Power Switch	Three section switch located on the left side of the device near the top.	(Down): OFF, the device is inactive
		(Up): ON, the device is on and running but the display will only
Dim Control	Dimming is controlled by holding the top left button then using the lower 3 buttons to control the dimness	Lower Left: Decreases brightness
		Lower Right: Increases brightness
		Lower Middle: Sets the brightness to automatic
Button Array	There are 6 buttons labeled 1-6 laid out in 3 columns and 2 rows just above the handle	Each display a preprogrammed message (see programming) on the LED display upon being pressed.
Power Light	An LED that is located on the back of the device lower left	On when if the display is on
Micro USB Port	This is a micro USB port that will use the provided USB cable to plug into a computer USB port	To charge the device and can be used to program it as well
Programmable	allow the user to easily program each button to the desired text	via PC and or phone app
Visibility	The distance that the device can still be read clearly	35ft away

Table 2: Specifications table 2

Item	What it is	Specification
Software	Characters	All capital letters, !, @, #, $, and 4 emojis
	Multilane	This will be an option for the user to choose either one line of text and two lines of text, each line of text will have a given number of characters per line.
	Scrolling	This option allows the user to be able to set their messages as text that moves across the screen. This can only be done with single lined text, but with the text moving across the screen, the user will be able to fit longer messages. The scrolling feature will also allow the user to adjust the rate at which the scrolling is moving.
	Preview	when reprogramming the device, is able actively display the message on the display so the user has a real time understanding of what it looks like before disconnecting the device.
	Time Display	The amount of time each display is shown for on a single press can be customized to the user's desire.
	Reverse Text	This function would allow the user to be able to create a message that can be read while viewing in through a mirror.
	Flashing Text	Allows the user to make their message flash at a rate pre-set.
Cost	Price per thousand units	$40
	Prototype cost	$200
Durability	Usability	2.5 Nm force applied at handle and display joint
Display Operation	When **ON** upon the press of one of the **Display Buttons** the message will display for the programmed amount of time, this time will reset each time it is pressed..	

Table 3: Specifications table 3

4. Design Constraints

The project level design constraints discussed in this section are limitations that this project must follow in order to have a successful device that has a sense of standard. The types of project level design constraints that are addressed in this section are the economic, environmental, social, political, ethical, health and safety, manufacturability and sustainability.

4.1 Economic Constraints

The economic cost of designing and manufacturing the Electronic Flip Sign can cause various constraints on the project. For starters, this particular project was being funded totally by the design team and those causes a limitation to what parts the team can order to run trials on. This leads to a limitation on which parts the team can ultimately choose from when designing the device. Another economic cost constraint that needs to be addressed is if a part that is ordered malfunctions, does not work with the design anymore or is damaged, then the design team had to replace that part. This would have ended up costing more onto the budget. Therefore, the team had to take that possibility into account when deciding on which parts they used to develop the Electronic Flip Sign. Another important factor that needs to be accounted for is the fact that the team could not simply pick only the cheapest parts, because there is also the quality of the device that needs to be addressed. The quality of the design could not be hindered just to keep the price as low as possible. That being said the design team also could not have needlessly purchased the most expensive parts. There is a middle ground that needs to be met to achieve a low cost, high quality device. Another factor that was taken into account was time.

4.2 Time Constraints

Time was one of the most forefront constraints that could have caused problems for the design team if not prepared for. When designing and constructing an extensive project like the Electronic Flip Sign, a lot of time was needed. There was a lot that needed to be done and the team only had till the end of their second semester working on this project to achieve this goal. The team can plan out everything that needs to be done along with the amount of time it may take to complete this task, there still could have been outliers that causes the team to go overtime. For example, coding the microcontroller could be planned to take three weeks to do, but an error is discovered that causes the programming time to finish programming the microcontroller later than expected. Another example, is that the hardware could malfunction, causing more time needed to correct these issues. That is why it is important to overestimate the time it can take to complete tasks to account for errors. The team also had a lot of documentation that needed to be written during their time with creating this project. They needed to make sure all of that was done while also creating the project and taking care of their other responsibilities whether it was work or other college courses.

4.3 Environmental Constraints

When creating a portable device like the Electronic Flip Sign, the environmental constraints are very important to take into consideration. When a device is created to be used on the go, all different kinds of elements can cause problems for it if not taken into account. For example, if the Electronic Flip Sign is being used at a baseball game and it starts to rain, will the sign be able to withstand the rain. If it is expected to withstand the rain, then it needs to have the proper parts and casing that can make sure it will handle it. If it will not handle water, then it needs to be explained and stated in the manual. Another constraint can be if the sign gets dirty in the outdoor environment, will if hinder the use of the sign and will it be easily cleaned. The casing should not be a material that is very hard to clean to keep up a good quality for the device. Environmental constraints also involve making sure that the environment is not harmed due to the use of the device. The main concern that the Electronic Flip Sign may cause for the environment is battery it uses. Batteries are not good for the environment however research will be conducted with the possibility of having the sign run on solar energy. This would be very help to the environment.

4.4 Social Constraints

The Social Constraints to the Electronic Flip Sign are primarily involved with the safety when customers are using them. There is the possibility that the Electronic Flip Sign could be used by some customer when it is unsafe for them to do so. An example of this is if a customer chooses to use the sign when they are driving. This is a problem that my other companies face with having devices that are portable. The main companies that have to deal with this problem are cell phone manufacturers and just like those companies it will be a warning in the manual to make sure that customers are aware of the risks when using the device while driving or doing other tasks that involve their undivided attention. It also should be addressed that there is the possibility that users may also use the sign to display vulgar remarks. This is a social constraint that the team needs to keep in mind. Research could be conducted in adding a sensor that sensors out "inappropriate words".

4.5 Political Constraints

The Political Constraints for this project are mainly involved with the design of the software used by the customers to create the messages they wish to display. The design of the software should not have any designs on it that may be seen inappropriate which isn't really going to be much of a problem for this project anyway. Beyond that, there aren't really any more political constraints for this project.

4.6 Ethical Constraints

The Ethical Constraints for the Electronic Flip Sign are mainly involved with the quality of the design of the device. As seen when the Economic Constraints were discussed, there is a middle ground between cost and quality. The quality of the device should not suffer just to save some money. Components are researched to a great extent to make that they are reliable and when a good quality all around. The last thing the design team would want is to have the a component be a part of the Electronic Flip Sign and have it fail quickly and often. Also, the software that allows the users to create new messages will be safe to download without major security issues. The device will also be up to the specifications that have been designated to the project. Anything that is a part of the design specifications will be a part of the device unless otherwise specified. The Electronic Flip Sign will also not use anything copyrighted without the proper permissions and citation needed.

4.7 Health and Safety Constraints

The Health and Safety Constraints are some of the most important to be aware of when it comes to designing any device, especially if it is being designed for the average consumer. One important constraint for picking out components for the Electronic Flip Sign when keeping safety in mind is to make sure that known of the components get too hot. This is especially an issue when it comes to picking the proper LEDs. If the LEDS were too hot, they could be harmful to the user if they were to touch them when the sign is left on for a while. Another important safety issue that needs to be addressed is that the casing and handle are not sharp. If they were to be sharp, that could lead to harmful situations. The handle also needs to not be difficult to remove in case it causes safety issues. For example, the handle could cause damage to someone's hand if they pinched their skin while trying to remove the handle. The batteries that are a part of the design need to picked properly in the case of safety issues. Fault or poor-quality batteries can run it the problem of malfunctioning and possible exploding due to this. Due to this reason, the design team conducted extensive research to make sure that the battery chosen will be in proper shape to be used for this design.

4.8 Manufacturability Constraints

The Manufacturability constraints of the design involve the ability to be able to produce the device. This involves the availability of parts in the design along. If the parts are not consistently available, how can you expect the reliable to produce the product in a timely manner. Another constraint for manufacturability leads back to time constraints. To put it simply, the time it takes to manufacture the device is a constraint. Another constraint that needs to be considered is the use of software that is created by another company with copyrights attached to it. The company needs to be properly referenced when allowed to be used to create our software and other programming needs. This also goes for anything else that is done for the project by a company. The casing for the device needs to be a material that can be formed to fit the assembled parts properly without damaging the parts in any way. This means that the material chosen needs to be able to protect the components, but still be able to be manipulated to form the shape of the design of the Electronic Flip Sign.

4.9 Sustainability Constraints

The Sustainability Constraints are constraints that need to be considered when designing the device to be long last in both operation and in durability. The battery needs to be well equipped handle the operation of the device for at least fifteen minutes of constant operation and while not being plugged into a wall outlet. This means that the battery needs to be strong enough to handle the components it is power and can still hold a good charge. As discussed previously, the durability of the casing and assembly needs to be able to withstand heavy amounts of use and situations that may cause minor accidents.

5. Project Aids

During the course of the project the use of different aids will be used throughout and here will go over these. There are a few different kinds of aids that are used. Each of the different facilities that are used to help with the project from research to building and prototyping are noted. Some of the tools that were used to help in the testing building and production of the device are given. Last are the standards that help to guide the build to be functioning and usable and up to standards.

5.1 Facilities

Places to conduct meetings to design and build the project are a necessity in order finish this project. Professor Young has a lab in Lake Mary that is used to help in developing the flip sign. The Texas Instruments Innovation Lab at UCF is available for students to use and is free of charge. The facility is available for use during weekdays Monday through Friday between the hours of 10am to 10pm and can hold a maximum of 40 people [52]. To enter, students just have to sign in using their student ID. The facility is run by four student assistants and a lab director that. This lab gives students access to equipment that will help create the product that is being designed during the senior design course. The lab supplies students with plenty of work space with three tables that can fit up to ten people each and tools like an oscilloscope, function generators, power supplies, digital multimeters, a laser cutter which can be used to cut through or etch acrylic, paper, wood, and cardboard, a 3D printer, Texas Instrument’s microprocessors, soldering irons, microscopes, and other basic tools like screwdrivers, pliers, and saws[52]. The Senior Design Lab at UCF in Engineering II room 456 provided a place to meet, discuss and test in a quiet environment. This lab is open to students in senior design I or senior design II 24/7[53]. The facility gives students access to Oscilloscopes, function generators, multimeters, DC power supply, and computers which is all useful for the project. The John C. Hitt Library is a facility that is used for group meeting for writing and brainstorming the paper for senior design. The library provides computers and printing services and is accessible Sunday from 2pm to 1am, Monday through Thursday 7:30am to 1am, Friday 7:30am to 7pm, and, Saturday 9am to 7pm[54]. Idea Lab is a place that is used for brainstorming the design of the project. This lab provides whiteboard walls and tables that can be used to sketch out ideas for free. The Idea lab is located in room 434 in the Engineering II building[55]. The facility can hold a maximum of 50 people and is open to students Monday through Friday. Quality Manufacturing Services (QMS) offers many different types of services including such as PCB assembly, PCB manufacturing, testing, and repairing[67]. The company has offered help in printing the components to the custom PCB that are difficult to do by hand like the microcontrollers, transistors, capacitors, resistors, shift registers, decoders, etc. This is done by providing QMS with our PCB design.

5.2 Equipment

For the construction of the Electronic Flip Sign, tools and equipment were needed. There are labs at the University of Central Florida that provide equipment for students to use as well as equipment that can be purchased for self use. Equipment was needed in order to construct and test the Electronic Flip Sign. The following equipment found below were considered during the development of the device.

Tektronix MSO4034B Digital Mixed Signal Oscilloscope

The Oscilloscope available for use in the senior design lab is the MSO4034B and is manufactured by Tektronix Inc. This device can help find flaws in the distribution of power for the Electronic Flip Sign. This device can provide a bandwidth of 350MHz. There are four analog channels and sixteen digital channels available for use. The MSO4034B features a digital phosphor display that allows users to witness waveforms and be able to find issues within a circuit, triggers that can be used to detect issues, wave analyzing, automated measurement tools, power analysis, and voltage probing[57].

Agilent E3630A Triple Output DC Power Supply

The power supply available for use is the E3630A DC power supply found in the senior design lab. This device can be used to send safe, accurate and reliable measurements of power to prototypes in order to test functionality. The power supply offers 3 output channels that can offer a range of voltage and amps. The voltage can vary from 0V to 6V and the amperage can range from 0A to 2.5A. The max power that this unit consumes is 35 Watts[59].

Tektronix DMM4050 6.5 Digit Precision Multimeter

The DMM4050 6.5 digit multimeter can be located in the senior design lab and is available for use by students. This device offers the ability to measure the volts, amps, ohms, frequency, period, continuity, diodes, capacitance, and even the temperature which can be useful in verifying the circuits are connected correctly and gives a way to monitor current flow through the Electronic Flip Sign. The reading of DC voltage has an accuracy of about 99.9976% and has a measuring range from 200mV to 1000V. The current that the multimeter can measure ranges from 200µA to 10A. The resistance that this device can measure ranges from as little as 200Ω to 100MΩ. The device provides up to 6 measurement inputs, a USB port that can be used to store data, and a RS-232 serial port that can connect to the computer[58].

Stahl Tools SSVT Variable Temperature Soldering station

The soldering station belongs to one of the students developing this project and is brought in when soldering is required. This product features an on and off power switch, a hot iron light to warn the user the iron is still heated, a variable nobb that can adjust the temperature of the iron ranging from 302 degrees Fahrenheit to 842 degrees Fahrenheit which ranges from 5 watts of power to 45 watts of power in respect to the temperature, a cushioned rubber grip on the iron that provides a comforting non-slip stable grip, and the ability to replace tips in case one gets damaged to increase the lifecycle of the device[60].

Anet A8 3D printer

The 3D printer by Anet was considered to manufacture the case and handle of the Electronic Flip Sign. The frame of the 3D printer is made from laser cut acrylic. This model supports multiple types of 3D printing filaments such as ABS, PLA, Wood, Nylon PVA, PP, and Luminescent[65]. The max printing dimensions are 220 x 220x 240 millimeters. The pulleys, rail rods, gears, bearings, and connectors are all made of metal which will give each of the components longer life spans. The printer is equipped with quick-release feed gears[65]. This feature provides a faster and more efficient way of transporting filament. The printer has an Liquid Crystal Display screen that displays the current status of the printer. The device is powered by a 12V power supply. he Anet A8 works with Windows XP, Windows 7, Windows 8, Mac, and Linux. The supported file types are G-code, OBJ, and STL and the supported 3D printing software is Cura and Repetier[65].

3D modeling software

Software for 3D modeling needs to be used to develop models for the device's handle, casing for the PCBs, and battery casing. There are a lot of modeling software programs available that support 3D printing. Some of the options that are being looked into for modeling the objects needed are Blender, 3D Slash, and SpaceClaim.

Blender is a free open source 3D modeling software that is available to the public. This software allows people to model objects in 3D and can be used in a professional environment. Blender is one of the softwares that will allow for the group to design and create various models for the Electronic Flip Sign. Blender is able to support importing and exporting 3D files into the STL file format [61]. This is important due to the fact that this format is commonly used when designing a 3D printed model. Blender is also a free to use software which makes it much more accessible than some of the other options.

3D Slash is a proprietary software that can also be used in creating 3D models that can be 3D printed. Unlike Blender, 3D Slash is specifically meant for 3D model created for 3D printing. This could be very helpful since all its resources are meant just for 3D print, which is the reason for the use of the software for our project. The designs created in 3D Slash use blocks that together make various models [62]. 3D Slash is also known to being easy to use for those that do not have a lot of experience designing 3D models for the need of 3D printing. 3D Slash easy to use feature for beginners is something the team is considering, but Blender and other software used for 3D printing tend to have more features and tools for designing the models. 3D Slash also uses the STL file format when exporting designs.

SpaceClaim is another 3D modeling software that can be used to create models for our 3D printing needs. Just like the other options that are being considered, this software can also create models using the STL files. This is important, since this is needed when creating models for 3D printing. Similar to Blender, SpaceClaim can also be used to automate the repairing of STL models when needed [64].

The decision has been made to use the Blender 3D modeling software. There are a few varying reasons for this decision. For starts, the software is free to the public, which cause this software to be easier to gain access too. Another reason is that it has many modeling features that other programs like 3D Slash do not have. Another major reason is that it has a very user friendly user interface and can also export its models as STL files, which is needed when using a model for 3D printing.

10X-20X LED Binocular Stereo SE400-Z Microscope

The SE400-Z microscope is made by AmScope and will help give the person that will solder the LEDs onto the PCB an easier time because of the magnification this microscope gives. This microscope features 10x & 20x widefield stereo magnification, a 9 inch working distance, sharp true color images, and an arm with a Gooseneck Light attached[63].

5.3 Standards

There need to be standards set in place for coding, Security, Testing, Safety, Durability, and construction of the device. The standards are needed to be aware of when designing products. These set of rules are required so that the process of development will stay consistent, can be repeatable if followed, and prevent mistakes or issues from arising. These standards are absolutely pertinent for the production of the device as this device will have to meet standards to be able to be sold to the masses safely.

Security Standards

This device connects to other devices, PC and mobile, via USB and bluetooth. This device also contains memory storage and the software itself. Each of these aspects are something that has to be secured and protected to ensure the safety of all devices involved. Bluetooth is one of the biggest openings for security as it is remote access but there are security standards set in place for the use of bluetooth on how to safely and securely use bluetooth and these will be followed and up to date to maintain the highest level of security to prevent unwanted access to the device allowing for alterations of the device and use of the device to other means. The USB connection is similar to bluetooth in the way it can be abused but it is not as much of a risk as it does not allow for remote access but there is still risk in accessing unknown devices and even using known devices. There are securities set in place to keep it as secure as possible while in use and charging.

The software is a program that takes full control of the device each time it is connected thus making it a serious security matter. The program must not only be protected from other programs and infiltrations but also from the use of similar but modified code making use of the device to its own end while still being able to remain undetected. Security standards are used to prevent unauthorized access to the program are put into place to prevent access and control from other programs preventing their access. The prevention of mock programs is pertinent to preventing the spreading of malicious versions of the program getting to users and their devices and this will involve the protection of the program and its assets to prevent replication as well as an authentication that can be found to make sure the program is a version that is provided by a legitimate source to provide a legitimate program. Through each of these security standards is the protection of the protection of the device and its use not only for the protection of said device but as it does have some onboard storage this needs to be protected from being a carrier of malicious programs for the security of not just the device but all others that are in connection with it during use.

IEEE Standards

IEEE has a set of standards for all manner of things and these are set in order to be the foundation and fundamental baseline for all aspects of a device to make it acceptable in the eyes of IEEE in terms of safety, durability, usefulness, and accessibility. Each of the standards are given and made available to check with the corresponding component and it will provide the requirements and specs that expected to be upheld. These standards come from a lot of experience, trial and error and should be taken into consideration to make a better device that is useful and function to an acceptable level in many different occasions and for many different reasons to its use.

Standards for Testing

During testing there are a number of standards that were to be upheld in order to get results that are accurate and precise. Each device that is used has standards for the way the device should be used and the conditions needed to operate nominally this is essential to getting results that are accurate. There are standards for they way to test this includes making sure each variable that is not being recorded is as intended and during multiple test each of the variables should stay the same and remain unchanged unless intended to maintain reliable results.

Standards for Safety

Safety is number one priority. Each device, tool, and piece of equipment used has safety requirements and they are there to keep the user safe and to protect the device being worked on as well as the device or tool itself. Following the safety standards will make it so the results desired can be obtained without injury to the user the device being used and the one it's being used on this is the most important thing to remember when working on the project and must always be followed.

Durability Standards

The durability standards require the device to have the ability to withstand being dropped from about five feet and not breaking and allow users to hold the sign by the handle without breaking. There are standards for the materials that will be used. There is other device with similar functional designs and the standards can be related from them. These standards are so that the end product can be used in a manner suitable to the normal use of the product and still maintain working order and stability.

Standards for Coding

Every function in the code is well commented explaining how the code functions. This is important so that the programmer from this project has the ability to go back and improve code, fix bugs, or add on new features without having to sift through and remember how every bit of code works. This helps for quick fixes so the software developer can quickly spot where the code is that a specific issue is found and fix the problem within a few minutes because the comments that were done.

There is a standard for what language the program is written in. This is needed due to the fact that applications cannot mix languages. If the program was initially written in C then the entire program needs to be written in C, there cannot be any java or C++ or C# code or else the program will have compiling issues and would not function as intended. If another language was determined to be used then the entire program will need to be rewritten with that language's syntax.

6. Block Diagrams

The electronic device incorporates a built in LED display approximately 64 x 16 2D matrix panel, and a controller with a built-in pulse width modulator. This modulator will adjust the LED brightness depending on the user use of the device indoors or outside. Brightness is adjustable manually and via a ambient light sensor. Partnering modern day microcontroller chips and a LED display, the electronic flip sign offers solution to demanding networks of entertainment and mass communication.

6.1 Architecture

The device is composed of a power supply (the battery) that delivers power to the LED's and the system where a voltage regulator feeds a constant voltage to the micro controller. The power supply is powered by a rechargeable battery via USB. As depicted in figure 4, the signal is bidirectional, sending data communication to the controller in response to regulating the voltage across each diode. Bluetooth communication is incorporated to the controller design via UART to also receive and send information to the device. Multiple push buttons are built onto the board in order to alter the LEDs that are connected to the controller. And a PID controller with a photo sensor is implemented to adjust the brightness of the screen. When sensor receives a high intensity of light the LED voltage will maximize and if the intensity is not met by that threshold the led brightness will alter its max value.

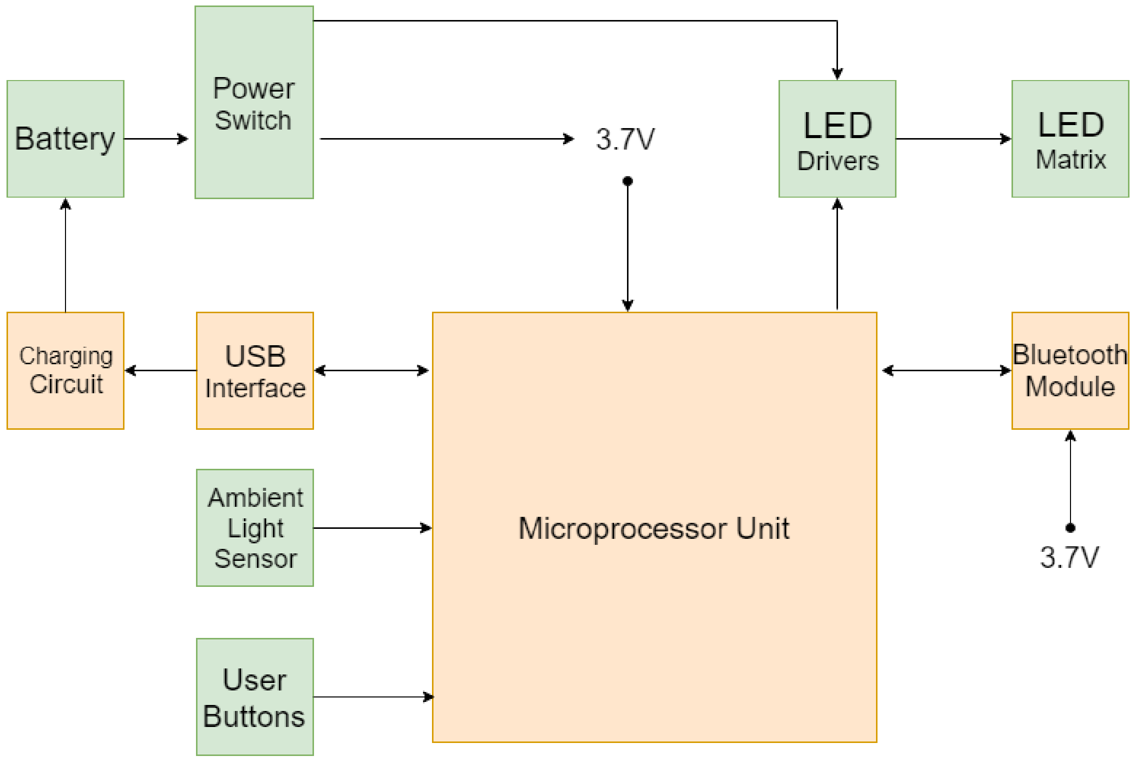


Figure 4: Hardware Block Diagram [20]

Key	
Richard	**Green**
Dominick	**Orange**

The LEGENDS are added at the end of each of the block diagrams to give some explanation to each of the blocks in the diagram. This also states the current progress in the completion of that current block and who is the main person responsible. There is a table for each of the two block diagrams and a spot on the legend for each of the blocks in the block diagram. The legends along with the use of the color coordination help to get the maximum amount of clear information into these charts as possible this will lead to a smoother execution of action down the line.

LEGEND Hardware Block Diagram	
Richard [**Green**]	Dominick [**Orange**]
Power Switch (Acquired) Used to power on and off device	**Charging Circuit** (Researched) Components used to charge the battery
Battery (Acquired) Used to power device	**USB Interface** (Researched) Components that allow for a USB connection and communication
Microprocessor Unit (Acquired) Chip used to processes information and control the device	**Bluetooth Module** (Acquired) Components allowing Bluetooth accesses
User Buttons (Acquired) User input on the device	**LED Drivers** (Acquired) Chips that control the LED's in the array
Ambient Light Sensor (Acquired) Sensor that reads the brightness	**PCB [Both]** (Designed) The board that connects all the components
LED Matrix (Acquired) An array of LED's in a 2D rectangle	
DC Power (Acquired) Distributes DC power	

Table 4: LEGEND Hardware Block Diagram

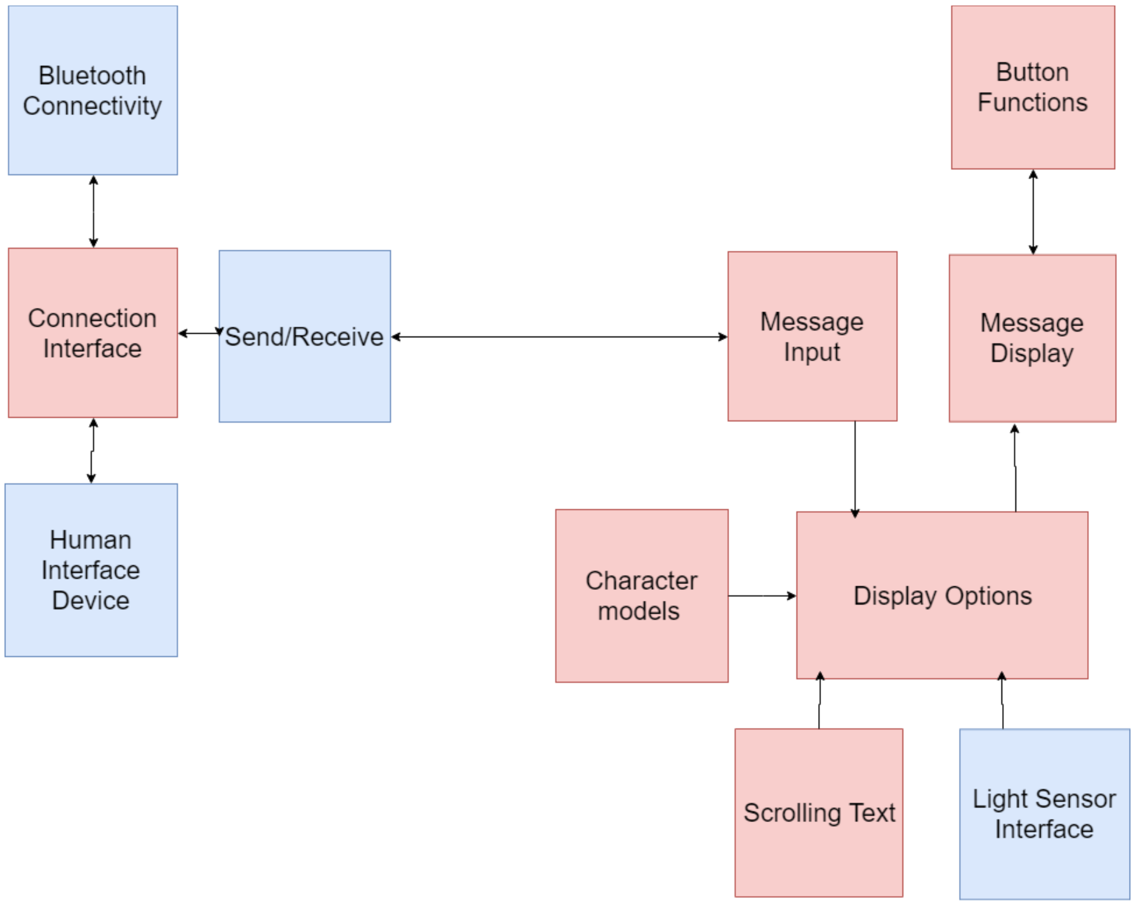


Figure 5: Software Block Diagrams [20]

Key	
Brennan	**Blue**
John	**Red**

LEGEND Software Block Diagram	
Brennan [**Blue**]	John [**Red**]
Send/Receive ● Transferring data between device and computer ● Inputs/Outputs: Transmits between computer and device.	**Button Functions** ● Buttons that are used to select message to be displayed ● Input from user ● Output to MPU
Table 5: LEGEND Software Block Diagram Part 1	

Bluetooth Connectivity ● Sets device as discoverable and allows device to be connected to another bluetooth enabled device ● Inputs/Outputs: transmits between computer and device	**Character Models** (Design) ● LED patterns for each charter ● Input GUI ● Output display
Human Interface Device ● Gives computer ability to recognize and use device ● Input: device to computer ● Output data to device from computer	**GUI – Graphic User Interface** (Design) ● The application that the user will use to program the device ● Input from user ● Output computer interface and memory
Light Sensor Interface ● Adjusts brightness of LED display ● Inputs ambient light level ● Outputs data for light level	**Message Display** ● Sending the message to the display ● Input GUI ● Output display
MPU - Micro Processing Unit ● Runs internal program that will be used to provide function to the device ● Inputs: Buttons ● Outputs: Message Displayed	**Scrolling Text** ● Allows the text to move across the display ● Input display message ● Outputs scrolling message
Message Input ● Takes in the desired text to display ● Input user ● Output display	**Display Options** ● Allows the input for all available options ● Input user ● Output GUI
	Connection Screen ● Interface to input all data ● Input user ● Output GUI

Table 6: LEGEND Software Block Diagram Part 2

7. Research

This section is used to document all the research we conducted as we designed our device. Our research also shows other established products that are similar to the Electronic Flip Sign. This also involves researching various components and discussing their specifications. All of the components researched involved looking into their features, statics and whether they will be able to fit the needs we need it too. Our research also involves creating software with a proper Graphical User Interface that meets the criteria for a user-friendly experience.

7.1 Existing Projects

The first project we looked into was Eletechsup's 64X16 dot Matrix LED. This Matrix LED board is built using 64 x 16 Red Pixel Matrix LED, UNO/MEGA2560 adapter board and adapter cable [8]. This device can display messages with letters, numbers and other symbols. The messages displayed can be created by the user using the software give with the purchase of this device. The difference with this device versus our design is that this device cannot be used on the go. The display must be continually plugged in to display the message. Our design is able to be used on the go without issues. This also leads to the fact that this device does not have a rechargeable battery while our design will.

The second project we researched when designing our project is the 8x8 LED Matrix MAX7219 [13]. This project involves assembling an 8x8 LED matrix that has text scrolling, and Bluetooth connectivity to an Android application. In the application, your phone can connect to the device using the HC-05 Bluetooth Module. The software also has two sliders, one for Scrolling Speed and the other for the brightness of the text. The last thing the software has is a textbox with a send button, so that the user can send their messages to the device, so it can be displayed. The current message being displayed on the device also is displayed under the textbox. That way the user can know which message they sent last. This project was very helpful with the idea that we can "daisy chain multiple drivers and matrixes and still use the same three wires connected to the Arduino Microcontroller" [13]. This can be seen in the figure below. It is very helpful to be able to connect all of these drivers and matrixes together using just three wires to cause less confusion and a more efficient circuit.

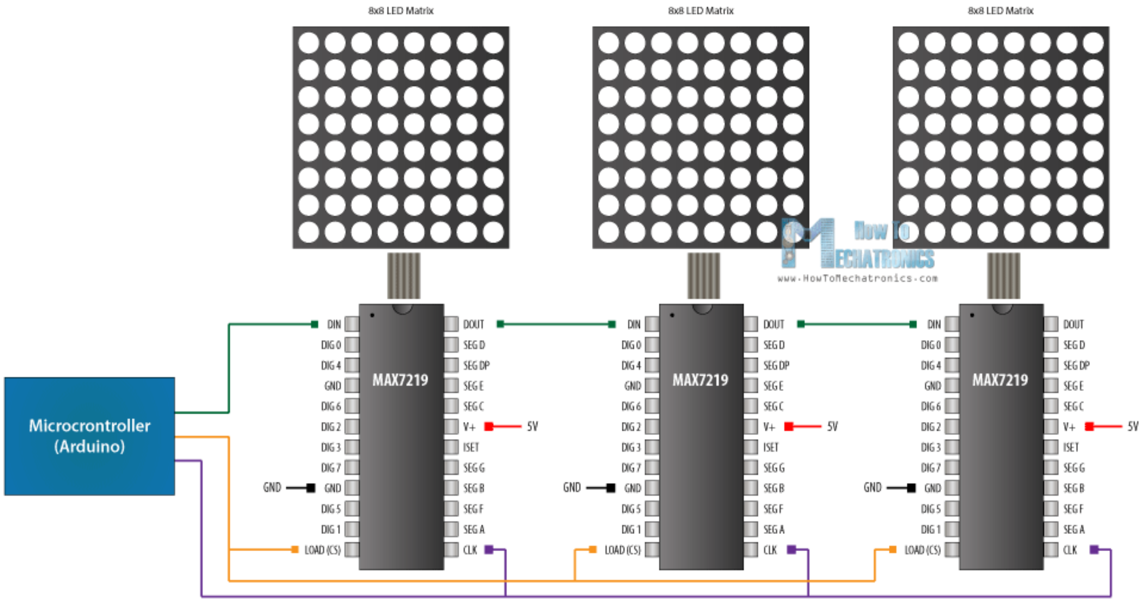


Figure 6: 8×8 LED Matrix MAX7219 Tutorial daisy chaining concept [13]

With these two projects already created we were able to gain a good idea on what is needed for our own take of the LED Matrix board. The first existing project helped establish an idea of how our LED matrix will physically look along with an idea of the direction in which we would go with software. The second project gave use an idea for how we will connect multiple LED Matrix boards, so that we can make our board the dimensions we are looking for in our design specifications. That project also told us how we can connect our device to an android application using Bluetooth technology.

Another project we had in mind when designing our device are the Hand-Held LED Stop Signs that crossing guards use to direct people across a busy street. These LED Stop Signs are stop signs that light up the text that says stop on the sign. The purpose for this is, so people can see this sign even during night time or in harsh weather. The LED’s in the sign are very important for the crossing guard, the citizens crossing the street and the drivers on the road, so that everyone is aware of their surrounds and they are all safe. This product has aspects similar to our device. Mainly in the fact that it is handheld just like our Electronic Flip Sign. Some of the features we considered based off this project are the use of Lithium Ion battery packs, the power saving feature of the sign turning off when it is put down.

7.2 Microcontroller

The microprocessor is the brain and the central control unit of the LED flip sign. This component stores and runs the code that will allow the user to interact with the device. Unlike CPUs, microprocessors are embedded to another device in order to carry out smaller tasks in comparison to a computer's CPU. Microcontrollers are also considered low power hardware, which allows them to be battery operated rather than having to always be plugged into a wall outlet [35]. There are a few aspects that need to be determined when picking the proper microcontroller for your design. Some of these aspects are the price, operating voltage, the number of bytes of RAM, the data memory size in bytes, the program memory in KB and the speed of the CPU in MHz. These aspects are compared for several microprocessors as seen in Table 4 below.

Microprocessor	Price ($)	Operating Voltage(V)	RAM (bytes)	Data(byte)	Program memory (KB)	CPU Speed (MHz)
ATmega32u4	3.98	2.7 - 5.5	2,560	1024	32	16
ATmega328p	2.01	1.8 - 5.5	2,048	1024	32	16
MSP430G2553 IPW20R	2.42	1.8-3.6	4096	NA	16	16
MSP430G2553 IN20	2.66	1.8-3.6	4096	512	16	16

Table 7: Microprocessor research table

Much like all other projects that use microcontroller, they have many tasks that they need to fulfill in order for the project to function properly. One of the tasks that the processor had to accomplish for the LED flip sign is to store up to 6 custom messages that can be written by the user along with any features the user chooses for how they want the message to be displayed. Another task is that the processor was able to communicate with the ambient light sensor(s) to adjust the brightness of the LEDs. The microprocessor is also responsible for the task of having communications with the buttons on the sign. If a button is pressed the sign should display the message that is assigned to that button. When the a USB cable is connected to the device from a computer the sign will charge and also communicate with the computer, which will allow the user to program custom messages onto the chip.

There are many different types of processors that can be used to handle these tasks provided by our design specifications. A few of them are the ATMega32u4, the ATmega328, the MSP430G2553IPW20R, and the MSP430G2553IN20.

The AtMega32u4 is a microcontroller that is a part of the Atmel AVR family. This means that Atmel Studio 7 can be used when programming this microcontroller. The AtMega32u4 chip costs $3.98 per unit. The operating voltage ranges from 2.7V to 5.5V. The program storage capacity for this chip is thirty-two kilobytes. The CPU speed is sixteen megahertz. This microcontroller also has a built in USB device peripheral that can be used if your device needs to be plugged in via USB.

The Atmega328P is also a part of the Atmel AVR family and the Atmel Studio 7 can be used when programming this microcontroller. The ATmega328P costs $2.01 per unit. This microcontroller is much cheaper than the AtMega32u4, but it does not have the built in USB device peripheral which is very beneficial to this project. The Atmega328P has an operating voltage range of 1.8V to 5.5V. This chip can hold thirty-two kilobytes worth of code. The CPU speed is also sixteen megahertz.

The MSP430G2553IPW20R is a microcontroller that is manufactured by Texas instruments. The MSP430G2553IPW20R is a microprocessor that has a price of $2.42. The voltage ranges this microchip has is 1.8V to 3.6V. This unit can hold up to sixteen kilobytes worth of code. This chip also has a memory size of 16KB. The MSP430G255IPW20R has sixteen I/Os which are all programmable, twenty pins, two timers and one UART channels. The RAM size of this microprocessor is 512B.

The MSP430G2553IN20 is also a microcontroller manufactured by Texas instruments that costs $2.66 and has a voltage range from 1.8V to 3.6V. This microcontroller is a part of the MSP430G2553 series just like the MSP430G2553IPW20R that was discussed previously. The processing chip can hold up to sixteen kilobytes of code for the program. This chip has a data RAM size of 512B, sixteen I/Os, two timers and flash program memory type.

The team had originally chosen to use the ATmega328p processor as an option for the project. This processor was cheaper than the ATmega32u4 and had double the program memory than either of the texas instrument msp430 chips. Programming the ATmega328p is a lot more user friendly than programming the msp430 chips. To program the ATmega series, Arduino open source IDE can be used. To program the msp430 series chips, the program code composer studio was used. Using the Arduino software will help with open source libraries available and are easy to integrate compared to code composer. We did however after further discussion decided to go with the ATmega32u4 as seen below.

It has been officially decided by the team that the ATmega32u4 microprocessor for the Electronic Flip Sign project. While the ATmega328p processor is a cheaper option than the ATmega32u4, the ATmega32u4 has the built in USB device peripheral that can be used if your device needs to be plugged in via USB. This microprocessor also has a lot more program memory than both the MSP430G2553IPW20R and the MSP430G2553IN20 Texas instrument msp430 chips. Programming for the ATmega32u4 will be a lot more user friendly than programming the msp430, because of the collection of libraries and functions that can be used along with the option of programming using the Arduino IDE or the Atmel Studio 7 environments. With Atmel Studio 7 being more complex, but more customizable than Arduino IDE we have the options of going as complex or simple as we want. Also, Atmel Studio 7 supports the Arduino libraries and functions as well if needed. To program the msp430 series chips, the program code composer studio will end up being used. Using the Arduino software will help with open source libraries available and are easy to integrate compared to code composer. The pin configurations for the ATmega32u4 microcontroller can be seen in the figure below.

7.3 LEDs

An LED is a light emitting diode. This component transforms electrical energy into light. The LEDs are one of the main components of the LED flip sign. They are used to illuminate the messages on the device. LEDs are turned on through LED drivers on the PCB. The different types of LEDs the team compared in our research can be seen in Table 8 below.

An LED is a light emitting diode. This component transforms electrical energy into light. The LEDs are one of the main components of the LED flip sign. They are used to illuminate the messages on the device. LEDs are turned on through LED drivers on the PCB. Each of these LED's prices are from a wholesaler AliExpress shipping from china, this is possible since even for one sign about 1000 LED's are needed, but it is also a necessity because even at the largest bulk from sellers like DigiKey the price in bulk was still $0.035 at some of the lowest prices. Prices like 3 cents apiece would be fine for a prototype but for production that is over $35 for just LED's, throwing the budget out the door.

First two types of LED's on the table is the through hole technology (tht) (seen on left on figure below) red LEDs and each are a different size either 5mm or 3mm. This type of mounting can be used in this type of application and are used in some of the competitor devices. This project dose not use that. Except for some outliers these LED's are the cheapest of each of these LED's.

The 1206 smd red LED is the next on the chart. Surface mount device (smd) is the type of mounting that is used by this and all the next LED's this means that there are no wires coming off the LED and they just have pads that get soldered to other pads on the PCB. The smd creates a much lower profile device this reduces the overall thickness of the device that's needed much lower. The number 1206 is not the LED part number but rather a measurement used for all the smd's and each of the numbers on the LED's are in fact a unit for measurement. The 1206 dimensions are 3.2mm x 1.6mm, these LEDs are relatively small and can allow them to be placed close but even spaced they can still be made out. The price is about the middle ground for smd LED's and offers a good amount of light.

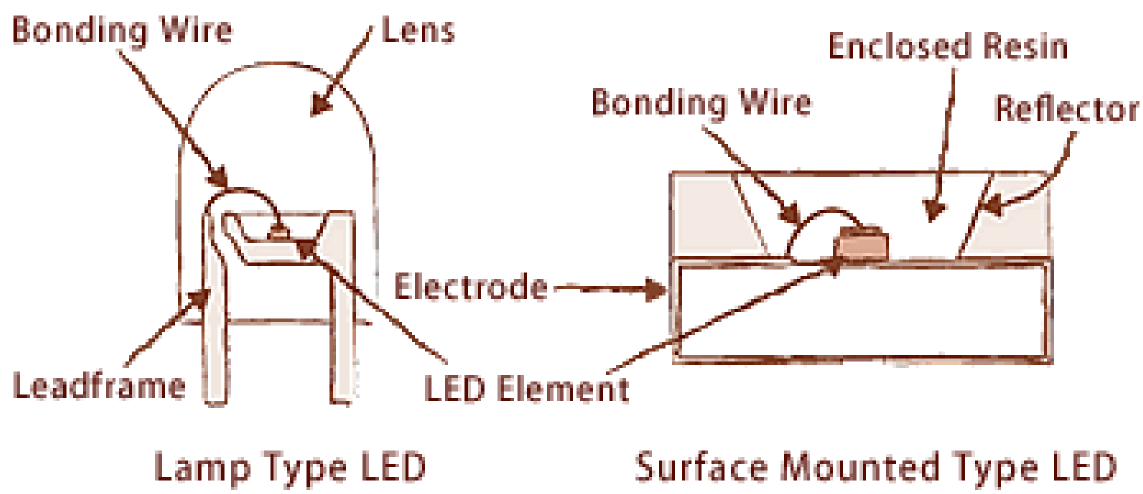


Figure 7: Through hole/ lamp type and SMD LED design styles [18]

The 3528 red smd LEDs are mostly an average price but there are some outliers for about half the price. The dimensions come in at 3.5mm x 2.8mm and are more square than most of the others although this is mostly irrelevant other than the size of the light emitter itself as this is what will be seen and change the appearance from LED to LED. On this LED and most of the other LEDs there is solder points that go from the bottom to the sides this makes it imperative to being able to solder them manually if that is the way that they are attached to the PCB.

The 2835 smd white LED is unlike the ones listed above white. The final color is still red but a white LED could be used and then a red tinted film could of been added over the display to make it display red. These LEDs are very expensive and would cost over $100 to make the display but this is in part that a wholesale china seller could not be found and that price is from DigiKey. If not for the higher price these LEDs produce much more light and are still sized like a 3528 and don't take a lot of power for the output.

The last LED on this list is the smd 5050 which can come is a few different manners it is not like the other LEDs this one has four LED units in the one package. This LED also has more connector and can come either in a RGB or all white form, with RGB you would get a multicolor experience, with the all white you would get something like the 2835 though still not as bright. This LED is larger than all the rest and measures at a 5mm square and though this is larger than the rest with the spacing that is being used for this project that would not be too large for this case. These are not all the LEDs that could of been chosen from, but some that embodied most major aspects and could fit the project's design plans.

LED Types	Cost Prototype (per LED)	Cost Production (per LED)
5mm tht	$0.00727	$0.00727
3mm tht	$0.00439	$0.00439
1206 smd	$0.00838	$0.00838
3528 smd	$0.0039-$0.00817	$0.0039-$0.00817
2835 smd	$0.099	$0.099
5050 smd	$0.01371	$0.01371

Table 8: LED research table

The LED that were used for this project are the 3528 LED. This LED is red and this was our choice in color for a few reasons. Red LEDs generally use less power than other colors. Red is a color that can be distinguished from surroundings easier than some being that since it is a darker color it is more likely to contrast its surroundings. The 3528 are cheap and at the bulk buy this was one of the better deals. The 3528 is not the largest and none of them are really large this is large for its category and it has solder mounts that curl around to the sides this makes it much easier to solder by hand this is important as soldering about 1000 LEDs needed to be made easy as possible. The brightness of these 3528 LEDs is enough to pass specifications given. The size of the actual region of luminance the area that glows is large enough to make a decent size to make a part of character alone from the distance this makes it so that if a “I” where to be displayed only one line of LEDs would be needed to make it, this is better for the room constraints of the given display and this is also good for manually soldering them as if it were needed to make the LEDs closer to make a more detailed display would of been hard to solder manually.

7.4 Bluetooth

Bluetooth is now considered a standard for connectivity from one device to another. Bluetooth allows the ability to transfer or access information on one device using another device. This wireless connectivity acquired by Bluetooth was very help when developing the Electronic Flip Sign. There are many different options when deciding on what Bluetooth chips can be used and if they should be used. The two major aspects when deciding this for our needs is by its price and which versions of Bluetooth the chip supports. The options looked into for this project can be seen in table 6 below.

The Bluetooth module on the LED flip sign allows computers and mobile devices that have Bluetooth to communicate wirelessly with the device. Having wireless connectivity to program custom messages is one of the goals of this project. Having the ability to customize the messages for the display without having to plug the device in, takes care of any hassle with using a cable, after the initial pair. The Bluetooth chip needs to be low powered so that the battery can last as long as possible. Security protocols were implemented so that only people with permission are allowed to access and modify the device's functions. There are many different types of Bluetooth devices, such as, the RN4020, HC-05, T1W1-uB1, and the RN42.

The RN4020 has a price tag of $8.83 per unit. The microchip is equipped with Bluetooth version 4.1 and is a low energy model. The low powered chip is beneficial to our project and will help with of our battery duration requirement. The antenna that comes built into the PCB is configured for about a range of one hundred meters. The dimensions of the chip are 11.5 x 19.5 x 2.5 mm. The operating voltage range is 1.8V to 3.6V.

Bluetooth Chips	Price($)	Bluetooth vers. supports
RN4020	8.83	4.1
RN42	15.27	2.1
HC-05 Bluetooth Module	7.68	2.0
TIWI-uB1	11.56	4.0

Table 9: Bluetooth research table

There are many different versions of Bluetooth available on the market. Bluetooth versions are determined by numbers and occur in a sequential order starting with 1.0. Some of the versions have updates to the milestones, for instance if version 1.0 had some improvements then the version would be labeled version 1.1.

Firstly, Bluetooth version 1.0 was the first version of Bluetooth to be released. This version had many issues. Manufacturers were not able to get their products to work properly with it.[7] There was an update to version 1.1 that fixed bugs in the previous builds were fixed, gave support for non-encrypted channels and has the ability to show the measurement of power that is given off by the receiving signal[7]. Bluetooth 1.2 came out in 2005 with a faster connection than 1.1. The data rate that version 1.2 could reach was up to 721 kilobits per second. Another feature given by version 1.2 was retransmitting damaged packets of data, this created better audio quality but in doing so, could cause an increase in latency between the receiver and sender[7].

Next, the next milestone in Bluetooth was version 2.0. This version was released in 2004 and provided a greater data transfer rate from the previous version by introducing an optional new technology. This new technology is called Enhanced Data Rate and provides a transfer rate of 2.1 megabits per second which includes packet time and acceptance of requests. The rated rate of transfer that

Enhanced Data Rate is known for is 3 megabits per second, this isn’t achieved due to other entities in the packet transfer process. This version improved functionality from the previous version by making it more of an easier process to pair devices from different manufactures. An updated version was released in 2007 called Bluetooth 2.1 with the available option of having Enhanced Data Rate. This version is most for the implementation of having secure simple pairing[7]. Secure simple pairing increased the quality of security between the connection of Bluetooth devices.

The next milestone after version 2.0 was released was version 3.0 + HS. Bluetooth 3.0 released in 2009 and provides a speed of eight times faster than version 2.0. To accomplish this speed Bluetooth is used to create a connection and while the traffic is handled through a collocated 802.11 connection[7]. This speed is only achieved with the optional +HS, a device supports this feature when the +HS logo is displayed.

The next milestone after version 3.0 was released was version 4.0. Bluetooth 4.0 was released in 2010 and comes with classic Bluetooth, Bluetooth high speed and Bluetooth low energy protocols[7]. “Bluetooth low energy” is Bluetooth 4.0 created using a new protocol. “Bluetooth low energy” was used to be known as wibree and Bluetooth ultra low power. Version 4.1 came out in 2013. This update provided an easier way for people to use the product. Version 4.2 was released a year later, which gave an increase in an increase in data packet length, secure connections, and privacy. Version 4.2 gave basic improvements to "Low Energy Secure Connect", privacy with improvements to the scanner filter and other support to the Bluetooth Smart things [7]. Version 4.0 also has the functionality of using a single-mode implementation and a dual-mode implementation as needed. The single-mode implementation only uses the low energy protocol stack [7]. The dual-mode implementation uses the Bluetooth Smart functionality along with the existing Classic Bluetooth controller implemented together [7].

The next milestone after version 4.0 was released was version 5.0. Version 5.0 was released in 2017 and is a big improvement from Bluetooth 4.0. Bluetooth 5.0 can transfer data at faster speeds and connect to devices that are at greater distances. The speed has been extended from 1 Mbit/s burst to 2 Mbit/s burst.

The chip we originally decided to use for this project was the RN4020 due to the higher Bluetooth version and low energy design. Going with the newer version provided a more reliable connection between the device and computer or mobile device. This makes pairing easier and faster. Out of all the options given in table 6, it is still the cheapest option and still supports Version 4.1 while the rest only support Version 4 or lower.

We ultimately decided to use the HC05 Bluetooth module. This module was used since there were problems trying to setup the RN4020 and using the HC05 is as simple as hook in and use.

7.5 Ambient Light Sensor

The ambient light sensor is used to tell the system the level of brightness surrounding the device so that the program can then send that information to the LED driver and pulse wave modulator to adjust the brightness of the LED's. Having a sensor detecting light is the main component of the auto brightness function of the display, this will dim the display as the ambient light gets darker this is to keep a similar perceived brightness while at the same time saving power. There are a few different types of sensors to get a reading that can send a value for the brightness to the processor.

First sensor is a light dependent resistor (LDR) also called a photocell. This is as the name states a resistor that is dependent on light, the more light it gets the lower the resistance is. In order to change the brightness of the LED's the value for light must be something that can be sent to the processor. To create a value for this sensor a circuit is made using voltage division so that the voltage can be read relative to the resistance of the LDR this voltage is then sent to an analog to digital converter and then can be used. A device that can be used to along with this the LDR is a LM393 which takes the part of a voltage comparator. The data was taken and tested to adjust the brightness to a desired level based on the new sensor data. Benefits to using an LDR is it is cheap and small, this is a larger factor due to the fact that the sensor will have to be located on the outside of the device to read the light levels. This sensor does require a small circuit and a analog to digital converter in order to be usable, this makes the system as a whole larger but the other components are on the PCB and in the processor.

The TSL235R is a light sensor that takes in the light level and gives a frequency. This sensor is rather expensive in comparison to the others. There are a few good attributes to using this device over some of the other options. This sensor requires no other circuits no boost in order to operate, this sensor is also pre calibrated. There is also some other reasons this sensor but they do not apply to this project. This sensor is self contained requires little effort to implement into a system but the cost of the device has to be taken into account.

The BH1750 is a light sensor package that outputs a digital reading. This sensor works like some of the others but it is a self contained package so it has a lot of smaller components like the analog to digital converter and analog light sensor on a small circuit board. This makes it less desirable for our purpose and in best case it would be disassembled and added to the PCB. this sensor would take up a good bit of real estate on the PCB for what it is. Running at about $2 this was not the most ideal for our needed application but does incorporate all pieces into a single functioning setup that depending on what is more important can make the area used smaller.

The OPT3001 is a single chip sensor that outputs a digital reading. This sensor is all contained in a single chip and is relatively small in size. The size and single chip aspect of this sensor makes this a nice sensor for this application but the price is also rather large in comparison. This chip is similar to the BH1750 but it is smaller and come fully contained in a smaller package which is better for this application.

Ambient Light Sensors	Input/Output	Price
Light Dependent Resistor	Light to voltage	$0.70
TSL235R	Light to frequency	$3.00
BH1750	Light to digital	$1.66
OPT3001	Light to digital	$3.84

Table 10: Ambient light sensor research table

7.6 LED Drivers

The LED Driver is used to control the LED's on the display. There are multiple ways and different types of components that could of been used to accomplish this goal. The goal of this device is to be able to have any of the LED's either on or off at any given time. To accomplish this goal using few of both power and chips the advantage of the speed of the processor and slowness of the human eye is taken into consideration and exploiting both of these attributes the driver uses shift registers to light up one column or one line section at a time then moving to the next column at a very rapid pace so that seen from the human eye the display seems to be fully lit when in fact only one line is lit at any given time. This is accomplished by having the rows and column equate to power and ground and when both are turned on the LED is lit making a grid that only at the specific cross point is the LED lit.

One of the ways this process can be achieved is by using shift registers. Shift registers can be set so each column and row is set to a shift register pin and then each LED can quickly be sorted through and lit or not as desired this can also be done in a few variations, such as using a multiplexer for the rows and shift registers for columns and then the height of each single lit line can be adjusted, such as using 16x64 or 8x128. Using this method each column or line has to have to have a resistor to set the current also as will be noted later this does require more of the chips themselves as well. Using this method can also limit the brightness of the LED's that they are capable of, this is mainly determined by the setup and the type of shift registers used.

Another method used is sole LED drivers these are chips that are made to drive LED's in this manner. Essentially these chips are specific shift registers and drivers that are built and optimized for the controlling of LED's. The Max 72219 can control a 8x8 LED matrix with the processor and one resistor (also some capacitors for power stability if needed) they can be daisy chained with more Max7219's to increase the size of the display. Benefits of using this chip over the shift registers themselves is that they will allow a greater current to get better brightness than the shift registers. The single chip also in most cases uses less chips and components to control the display. One of the downsides to using a chip like the Max7219 is that it does cost more but the Max7219 is a common chip for this application so it is possible to get them at a much better wholesale price, this however is not the case for the HT1632C which is not as common and even at better prices is still rather expensive, but it also has the ability to control 16x24 LED matrix on its own.

LED Drivers	Price	Type
Max7219	$0.50-$9.88	Full Driver
TLC5940	$2.90	Shift Register w/ PWM
HT1632C	$3.40	LED Driver

Table 11: LED Driver research table

7.7 Picking a Driver

This project has a few specific needs that will decide which chip was to be used. This design uses a wiring diagram that can be seen in the design section under LED wiring. The wiring for this LED matrix sets so each column of LEDs are all wired in parallel with each other. Using this information some calculations were done to find the power and current usage which is also a design aspect that has to be considered. The device has to be able to run for 15 minutes but also has to be lightweight, this means that the battery needs to be as small as possible as it is one of the heavier components. In order to reduce the size of the battery while keeping the same display time the power usage needed to be lowered and the device had to run more efficient.

Lowering the Power

The first way to lower the power of the device is to choose a LED driving method that uses the least power and this can be seen at the most basic level using the power equation P=IV. In this case the V is the voltage from the source minus the voltage drop across the LED and according to the LED specs the voltage drop is about 2 volts taking a median point. The source voltage is coming from a 3.7 volt battery and this can be used as is or could of been boosted to a higher voltage and there could of been two batteries used to double the voltage. The last part of this equation is the "I" or the current which is about 20 milliamps for static use which is what these calculations will be done in and the values are similar to the values for the cycling display at higher currents. Now that the information is known the value for the power can be found and this calculation was for the

LEDs only even though there are other things that will use power like the processor but here what were changing is the LEDs. For a source voltage of 3.7V the power was almost half the wattage of the 5V source and even less than the 7.4V of the double battery. In order to lower the power of the device the source going to the LEDs should be as close to the forward voltage of the LED as possible this lowers the power usage proportionately. This had to be done with some care from the voltage source as if the voltage fluctuates or deteriorates over time and drops below the forward voltage of the LEDs they do not function also the value of the source voltage is somewhat limited as the standard voltage level for lipo batteries is 3.7V in this range.

The source voltage is an aspect of choosing a drive because each driver has its own operating voltage levels for the source. The MAX7219 has a minimum input voltage of 4V and is recommended at 5V so this is not beneficial to the power usage as the higher voltage causes a greater loss as stated above. Using shift registers and multiplexers on the other hand have a much lower minimum voltage so for these the power can be sourced straight from the battery at 3.7V this gives a much better and lower power usage. This is one of the major flaws of drivers like the MAX7219 and any others that require higher voltages and is a deciding factor because of the increased battery life and therefore the smaller battery that is needed lowering the weight and possibly the size.

Current Control

Each of the LEDs has a certain current that is sent through the LED so that the brightness is at its best. In order to get this current there are a couple of different ways that the current can be controlled to get ideal conditions and not damage the LEDs. First way is a constant current driver like in the MAX7219 all the control points are put through a current controller and the current that they are set to is set by adding a single resistor to the circuit, after doing this each of the driven LEDs now receive the current that corresponds to the applied resistor which can be found in the drivers datasheet using the forward voltage of the LED and the desired current. The second way to keep the current constant is similar to the way it's done in the constant current driver but in this case a resistor is used for each of the columns and this will create a constant current for each of the LEDs. To find the value for these resistors there is some basic equations like Ohm’s law that are used to make the equation $R=V/I$ this is used where the V is the source voltage minus the forward voltage of the LED and then I is the current that is desired.

Deciding on a driver

The drivers are one of the key feature to this project and they determine a major part of the efficiency depending on their voltage needs. The drivers also change the layout and design of the device as some need more chips or less and others require new components to be added, each chip has its pros and cons and there are a lot to choose from ranging from chips built to run LEDs to using the core components of those chips to accomplish the same task. The choice of driver is a decision that will change the device fundamentally and is the heart of this device so the choice is both important and complex with no perfect answer and this sets up the design of the hardware at its core.

The first option for a design is a system that uses LED drivers that are designed specifically for the purpose of running LEDs. these types of chips like the MAX7219 are good at what they do and can get perfect results from the LEDs giving accesses to the max potential of the display they also take control of the PWM and require less chips in some cases and they easily stack with each other to make a larger display. The MAX7219 chip has 8 pins for the rows and 8 pins for columns these are the control pins that will make power and ground connections to make the LEDs operate the way that is desired and the MAX chip has pins for power a current set and data. The current set is just a pin that is hooked with a resistor and this will set the current that is sent to each LED. this has the drawback of requiring an input voltage larger than the 3.7V given by the battery and this causes the need for a boosted input voltage and this in turn will reduce the efficiency of the device overall.

For the other method, it is like a decompiled version of the full on LED driver and takes the core components of how that device works and what the underlying method is and it makes it using those core components. There are many ways to make this work but there is a core idea behind how each of these work and function. The main idea is that each of the columns are given power one at a time and this one column shift down the column at a rather rapid pace to give the effect of the display being on. On the rows each row is either be set to ground or not and then when these two meet at an LED that led turns on. The single current set resistor does not exist for this design and so the current is set in another way and this is still done with resistors but now there is a resistor on each of the rows that is tuned to the correct resistor value that it will produce the correct current for each LED. Even in this design there is some differences that change and there are a lot of different combinations of chips where some use the same chip for columns and rows while others use different chips for columns and rows and then the specific limitations of each chip plays its own role in how each design works. The method of using the same chip as seen the image below is possible because each chip can perform the role of both a shift registers and multiplexers.

The circuit shown below [43] does run into some issues that come from the limitations of the chips that are being used and this is that the chips being used cannot control enough current when scaling to displays larger than 8x8 and this will cause problems with the size of our device but this is overcome using the chips that are given using transistors to cut the current completely and this will allow the use of higher currents but this does require the use of extra chips to add this will up the cost and area used but sometimes this can be compensated by the lower cost of the initial chips and room is not much of a problem. Then next choice is a chip that has more robust as a multiplexer and able to handle the current along with a shift register or a decade counter this combination allows for a more flexible voltage option and also competes in the price range without the need for too many excess chips and area usage. This choice is ideal for our case as it allows us some flexibility and still does not require a lot of unnecessary chips and components to operate but still keeps cost low and does the job needed.[1]

Even though this new set up in the figure below [43] it is still not optimized for our application. The device were using has the dimensions of LEDs 20x48 and this will be require a lot more power to directly upscale the format below and in order to do this without using 2 chips for each 8x8 square then there is a problem using this setup. The chips that function as shift registers and multiplexers including the chips shown below cannot supply the current needed to turn on even a majority of the LEDs at full brightness and they would burnout as they are not related to that current. Any chip that is rated to higher currents only runs at 5 volts which will in turn cause more losses in the power consumption. In order to still run off of 3.7 volts from the battery there has to be some kind of device that can be an intermediary to move the power. To accomplish this task transistors are used they will be used so they take the power and then are turned on or off according to the drivers so each row or column has to have its own transistor [68]. Only one row will be lit at a time one side has to send power to all the on LEDs but on the other side it will only ever be one LEDs worth of power and that the drivers can handle on their own, this would be fine for supplying power but since there just current sinking they will have transistors as well. Using this method allows the use of a driver that can run on 3.7 volts and still provide the needed voltage needed to light each of the LEDs at full brightness since the transistors will be taking all the loads off of the chips and putting that on themselves and they can take a lot more power than the chips can making the system able to run many more LEDs in series then without the transistors.

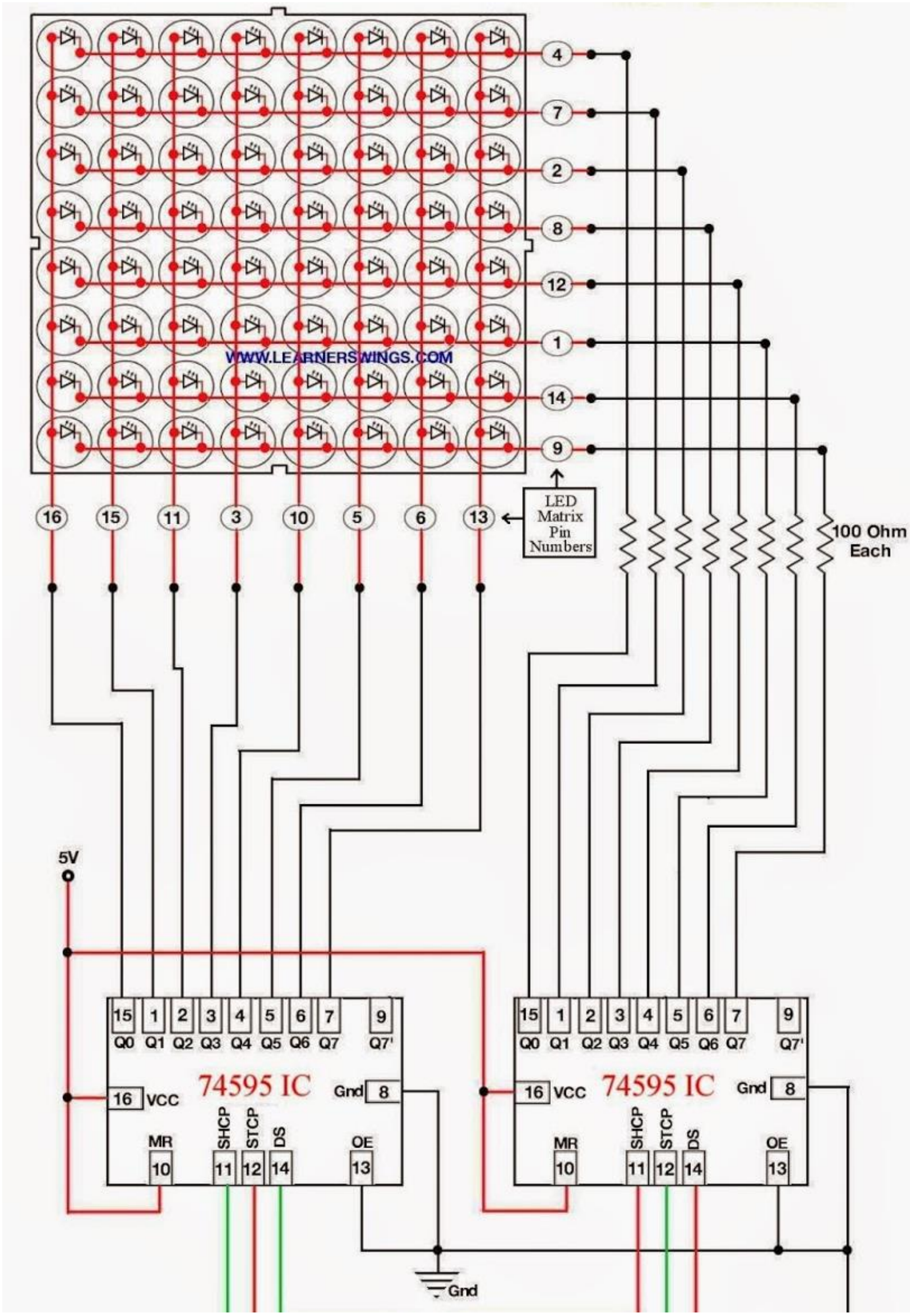


Figure 8: Circuit schematic for 2 74595 IC shift registers [43]

7.8 Battery

The battery has a number of requirements that have to be taken into consideration when choosing the correct battery. The end specifications for our project is that is had to be able to power the device for 15 minutes of display time, the battery had to supply enough power to display the LEDs with adequate brightness to make the visibility at least 45 feet, the battery also had to be rechargeable, lastly the battery needs to be relatively small to fit our small and lightweight design. These are the given specifications that must be obtained but otherwise the specific battery was just chosen based on which battery fulfills these specifications better than the rest.

Types

The types that will be discussed are all rechargeable this is the broadest specification and will allow the search to be narrowed down greatly. The first type of battery is a Lithium-ion, these batteries come in small form factors and are dry cells that come in packages that can be packed into a smaller design without causing problems these are often used in devices like cell phones. The lithium ion has some of the best energy density, its standard voltage is at 3.6V, it has a cycle life that is one of the longest, lead current can be all the way to 1C in good conditions allowing for adequate current draw in most cases, and lastly this battery requires no maintenance, this battery however does have some flaws such as the cost is higher than most other options and it can be fragile so safety circuits are used to protect it limiting the voltage and current. [37] This battery is a good choice because it can provide a lot of power in a smaller area and it has an operating voltage and current that will be adequate for this project in most cases, but this does come at a cost being one of the more expensive options available and there is the safety circuit that is required and the general increased safety risk but under normal correct operation this is not an issue. Another battery similar to the lithium ion is the lithium ion polymer battery and this battery is similar to the lithium ion in most way but it can be smaller and light weight and it does not have the same safety problems as the lithium ion dose but it does not have as much energy density and it is expensive to produce. [37]

[40] All electronic mobile devices or home appliances are mainly power by a DC voltage power supply. For the electronic flip sigh design a vital feature for power supply had to support a rechargeable electronic voltage source with a lithium ion battery or cell. How the portion of the device works depends on the lithium ions in which they move in a direction from the negative electrode to the positive electrode during the discharge process and reformed back to its original state. Another keen feature that is also satisfied in the design requirements of the electronic flip sign, to utilize the lithium ions lightweight properties to reduces the amount of weight and also supply an efficient cheaper solution to run power to the completed model of our design. Producing the same amount of voltage that any type of lead based batteries could produce. During the process of discharge, lithium ions (Li+) carry current through the system through the flow of charged

particles going from a higher potential to a lower potential. Some compounds that form Liquid substances to support current flow through the battery source include lithium salts, such as LiCLO4 an organic solvent. This liquid electrolyte acts as a medium channel for cations to move from negative to positive terminals during discharge.

For the procedure for charging to be implemented, there are three stages that are followed through. The first stage is called constant current (CC) and the second stage is balanced followed the final stage called constant voltage (CV). During the constant current phase, the charger applies a constant DC current to the battery at a continues increasing voltage, until the final voltage per cell is reached. For the next stage, the balance phase, the charger then reduces the charging current while the state of charging of each individual cell is brought to equal level as the balancing circuit. For the final stage, constant voltage phase, the charger applies a voltage equal to the maximum cell voltage multiplied by the number of cell in series to the battery. Then current exponentially decays back down to zero. This process will continue until the current value has reached a certain threshold which is about three percent of the initial current value.

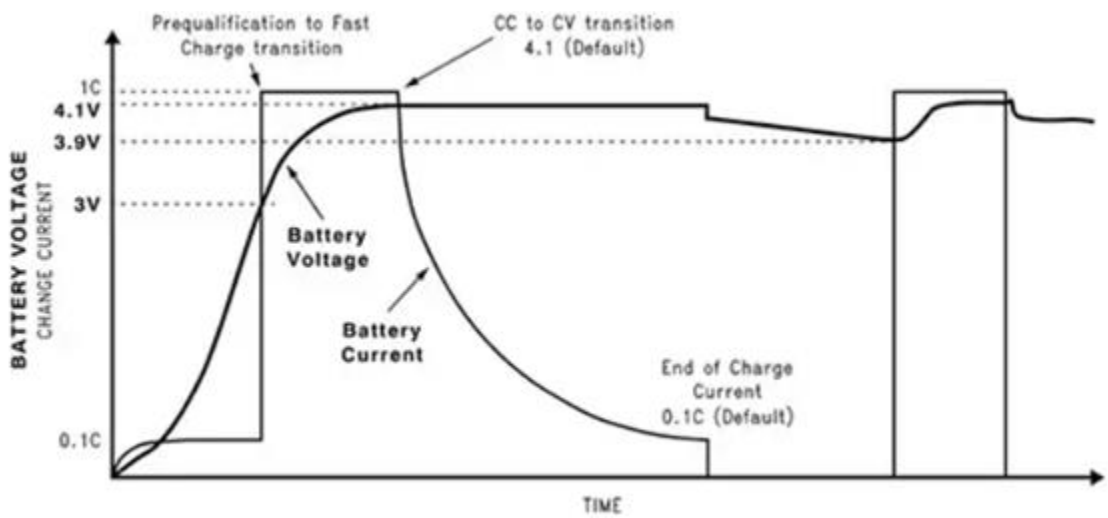


Figure 9: (CV) Output Voltage with respect to incremental time value [40]

Nickel-Metal Hydride (NiMH), this battery type is similar to the Nickel Cadmium (NiCd). NiMH has a much lower life cycle than NiMH and is a little more expensive, both base run on 1.25V and NiMH usually cannot draw as much current whereas NiCd has a high overall load current, both of these batteries require maintenance, the NiCd about every 1 or 2 months and double that for a NiMH. [37] NiCd has the fastest charge time for any of the batteries, works in a lot of temperatures, relatively stable and rugged, and at a low price. [37] The NiMH has a higher energy capacity than the NiCd and it does not contain all the toxins as in the NiCd, but it does not last as long and the lower charge current along with the maintenance is more than other batteries needing full discharges and can be sensitive to temperatures. [37] For this device the battery is going to be inside the device and for convenience the maintenance for these batteries is burdensome for the user and would be good to avoid but the price is lower than the ion batteries. The power output being lower than what is needed for the device to run means that these batteries would of had to either be paired and hooked in series or obtained in a series package giving a higher voltage and this can in some cases cause some issues that bring up complications that are not needed otherwise.

[40]Lithium-ion battery cells are composed of intercalation compounds. Intercalation is the structural chemistry of molecules or ions that form into material layered constructions. Main structure that for this type of model would be an example of the graphite crystalline stable structed as seen in a positive electrode system as seen in figure 2. This cysteine structure allows lithium ions to flow from between the layers. When discharge occurs the lithium ion battery, the ions move from negative electrode through an electrolyte to the positive electrode, thus causing the electrons to move in the opposite direction and produce current that moves in the opposite direction of the election to power the load of the system. The negative electrode acts as a storage facility, when the ions in the negative electrode are used up, current stops flowing because the electrons will cease to move once equilibrium is reached due to there being no total net charge of positive ions to be attracted to. That is why charging the battery forces the ions to form back to the negative electrode across the electrolyte and implant themselves back into their original position to undergo the process of discharge yet again.

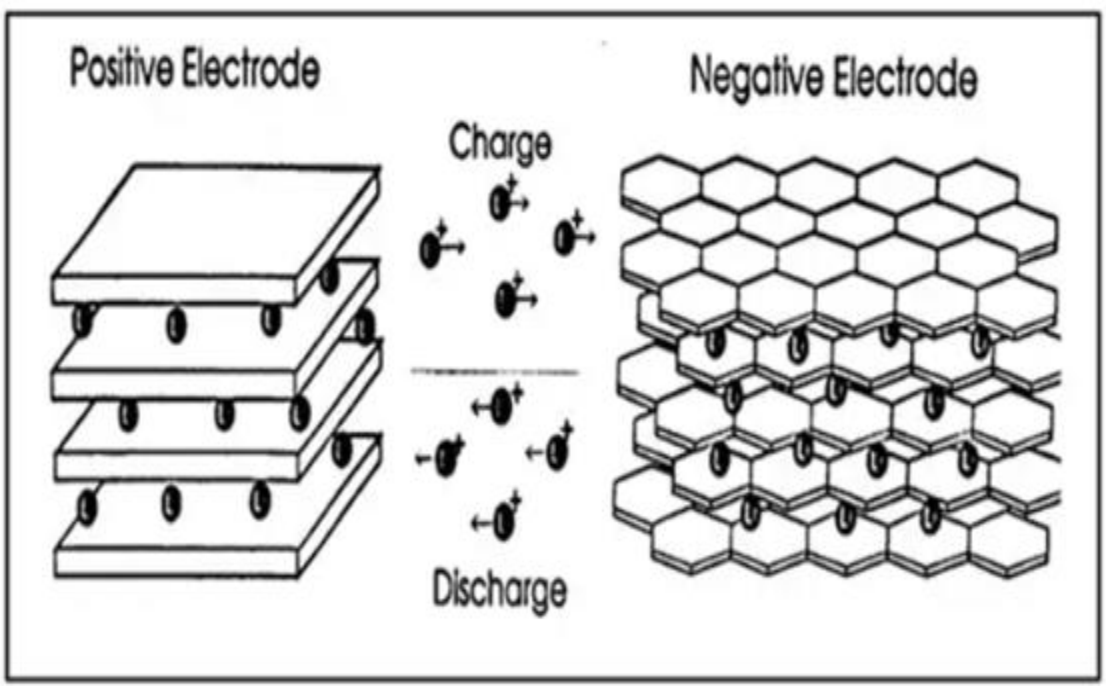


Figure 10: (Lio) lithium ion internal battery cell [40]

The final option was to look at a rechargeable Alkaline battery. These batteries are similar to the everyday batteries that most people use all the time. In most cases these batteries do not come in flat form factors as it is not as common as the usual cylindrical and other more 3 dimensional shapes and would not be as cost effective to do so. These alkaline batteries are not the lowest energy density but sits in the middle not great but not the worst. The cycle life of these batteries is the worst by far and don't tend to hold up to the test of time well. The voltage for a standard cell is 1.5V and as with the others there can be battery packages that contain multiple cells to increase the voltage. One of the main drawbacks to using this type is that the current draw is low but one of the best aspects of this type of battery is its cost is super low in comparison.

Batteries [37]	Energy Density	Cycle Life	Cell Voltage
Lithium-ion	110-160	500-1000	3.6V
Lithium-ion polymer	100-130	300-500	3.6V
Nickel-Metal Hydride	60-120	300-500	1.25V
Nickel Cadmium	45-80	100-200	1.25V
Reusable Alkaline	80	50	1.5V

Table 12: Battery comparison table

Battery Specs

The battery specs are list as each of the core components and identification that makes up a battery and how it is going to function while in use. Understanding these values will allowed for the best battery to be chosen that will work on the device and work the best of any other. These specs are a key factor to choosing the type of battery to be used as they do have different stats from battery to battery but different types have their own strong arenas and weak areas.

Charging Capacity

First and foremost is choosing the size of the battery or the capacity which in most cases is measured in milliamp hours and this is basically the milliamps that is can put out in a single hour before dead. This project uses a relatively small amount of power thanks to the way that the device shifts through the different columns making so that not all the LEDs need to be on at once, for a rough estimation of the power that will be used a single column of LEDs is calculated at a steady on state from a constant source, this will be an approximation but it will be close to the value of a single column being lit for a short time at a higher current. The size should be picked so that given the amount of current being pulled and the amount of time to be pulling that current also it should have some room for error as the battery discharges it can start to drop voltages and over time can reduce the capacity as well. Boosting the voltage will void the calculation of the amp hours and you would then have to find the power in watt hours by multiplying the voltage and the milliamp hours together to get milliwatt hours.

For today's typical uses for more stable and efficient rechargeable batteries we incorporate lithium cobalt oxide (LiCoO2), this structure is a lot safer than graphite be more reactive to lithium. Another import limiting portion of the battery is to account that this structure has imperfections within itself. Just like semiconductors utilizes PNP junctions to produce current flow through diffusion. The lithium battery still undergoes a similar process, when ions flow through the electrolyte medium similar to the intrinsic pure junction within the semiconductor some ions are lost in recombination. Thus, over time this causes the battery to gradually lose amounts of voltage over time. Overall putting a limit on the recharge cycles.

To address such a problem to sustain more voltage you can increase the energy density by compacting more lithium ions per cm^3 or volume. For throughput results ideally, we look at the charge time it takes for the battery to fully power and to be able to maintain a constant current supply for as long as possible. This time constant is ultimately determined by the capacity of the battery. Capacity is defined as unit "C", where equivalently 1 C will be enough to supply a maximum current the battery can supply for 1 hour. For example, if the electronic flip sign 2400 mAhr the battery will take twice as long compared to a capacity length of 1200 mAhr battery, when both batteries are charging at a constant current. Thus, engineer can use these values to develop more efficient means of producing better designs.

For instance, if the battery was being charged at by a constant current source of 1 A to a 4000 mAhr battery will equal a rate of 0.25C. With this result we can approximate that increasing the charging current will further decrease the recharge time. There are drawbacks to this condition that can harm the system thus the battery itself. First case if you increase the current the electron mobility is finite and will not change all the excess energy that is applied to the system will be dissipated and lost as excess energy further damaging the battery. For the other condition if too many ions enter the negative electrode by increasing the current the structure can become unstable and collapse. While undergoing this charging process you can see this has detrimental effects on the battery capacity. For that exact percentage change can be reflected by the time of charge spent. For example, charging under one percent can overall reduce the capacity around nine percent.

For these types of conditions, the battery should be designed to control the voltage at specific values to detect when the battery is fully charged. The type of battery that we are implementing will include a predefined circuit function that will automatically be able to detect when the battery is fully charged. Other detection methods include the current load and temperature. If the temperature goes above a specific threshold value charging will shut down its process to keep and maintain stability so the lithium battery will not receive damage.

Current Load

The battery is able to output the required amount of current that is needed for the device at max load safely. The current that can be drawn for a lipo battery is determined by the C rating which is a value that is given and is used to find the amount of current that can be drawn from the battery. This value can be found by multiplying the C value by the capacity and this will give the max current draw. Stated in the previous paragraph the circuit can be estimated and using this estimation the current can be found to be the number of LEDs in a column times the current through each LED this can then be compared with the current rating but not before the other currents that are being used are added such as the processor and other chips using power to get an accurate reading.

Voltage

The voltage is listed on the battery as a voltage that will most be given when operating at ideal ranges. The voltage of a lot of batteries varies with the charge, sometimes at full charge the voltage will read a little over the noted voltage but as the battery is used the voltage will start to drop. Some batteries have a system or are built to cut off completely after the voltage drops to a certain point. The batteries are also sometimes packaged in packs of more than one cell this means that the battery will output voltages higher than what is standard for that specific type of battery as the cells will be wired in series to double or triple and so on the voltage.

Reading Charge

The device is going to have the ability to read the current battery levels and then to display them on the screen. Some of the methods only work for certain types of batteries whereas others can work on a variety of batteries. There is a number of ways that the current battery levels can be read and each works in a different way and some are more accurate than others but some are easier to and cheaper to get than others and each of these factors are added to make a system that can get a reading of some value that relates to the battery level.

The first method to discuss is the voltage method which is one of the more simple methods used but has trouble being very precise. This method takes the voltage reading and compares that value with the battery's discharge chart finding the intersection point to give the charge remaining in the cell. This method is not accurate for a number of reasons one of which is that the temperature and type of cell effects the results. The temperature can change the voltage reading and this will then skew the results. The results can be skewed if the battery is in use so this method is only really accurate on a rested battery. Even if the battery is rested and inactive there is still a lot of room for error as in the case of lithium batteries there is a shallow nonlinear discharge pattern this makes it hard to find an accurate intersection point to get an accurate reading. The discharge pattern of lithium ion phosphate batteries is shown below and in this chart, it can be seen that the set voltage of 3.7V is just a median point that is more of an average. The cutoff can be seen at 3V the battery will no longer supply voltage and it can be charged to a capacity of 4.2V. The non-linear pattern is seen as there is a kind of s curve dropping faster at full and low charge.

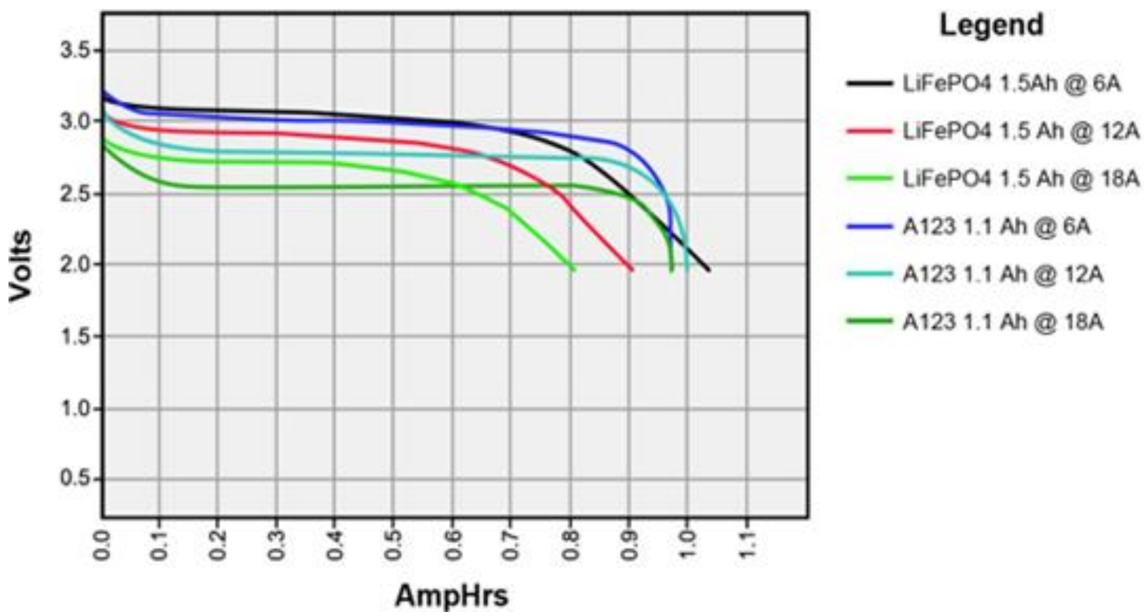


Figure 11: The discharge voltage for lithium ion phosphate [45]

The next method is coulomb counting the way that this method works is that it measures the current moving in and out of the device and uses it to calculate the remaining battery life. [45] This method can be very accurate and is even used in medical equipment. There are some drawbacks one of which is that the process that this method uses actually reduces the energy delivered. On the other hand, this method is very good for reading the voltage levels of lithium batteries even with their odd discharge patterns and this makes this a good method even with the power loss.

7.9 Voltage Regulator

A DC voltage is required to power essentially every electronic device including a computer and virtually for this project the electronic flip sign. To design a proper charging circuit to convert alternating current (AC) voltage to (DC) direct current with multiple devices. In addition, battery chargers for a portable device for the led matrix contains a charging circuit. To design a power supply the output voltage should be in a range of 3-5.5V to distribute power to the entire LED board. In the design, we constructed for the electronic LED sign display. The voltage regulator is used to power the entire LED matrix display at Vl=3.67V from a DC battery source.

For an ideal voltage regulator, the circuit used to build this model will include a non-linear Zener diode to properly regulate the value. As seen in the circuit in figure(25), shows the Zener voltage regulator. The output voltage should remain a constant across the Zener diode, when the input voltage varies with a specific range. The Zener diode can be conducting in reverse or forward biased. In order for this operation to function the Zener diode must be in the breakdown region and must not exceed it power limitations. The minimum current through the diode is Iz(min) and the load current is a maximum IL(max) when the source voltage is a minimum Vi(min) in the diode. When the current in the diode is a maximum Iz(max), when the load current is a minimum, IL(min) and the input voltage is a maximum VImax.

From circuit analysis, we can design the input resistance in a way so it can meet the design requirements in order for the device to operate on. When Ri is equal to the difference of the input max voltage and the Zener diode voltage over the max current through the diode plus the minimum current through the load or vice versa. This will allow us to maintain any output necessary output voltage, but for a more feasible design we replace this circuit with a more efficient component.

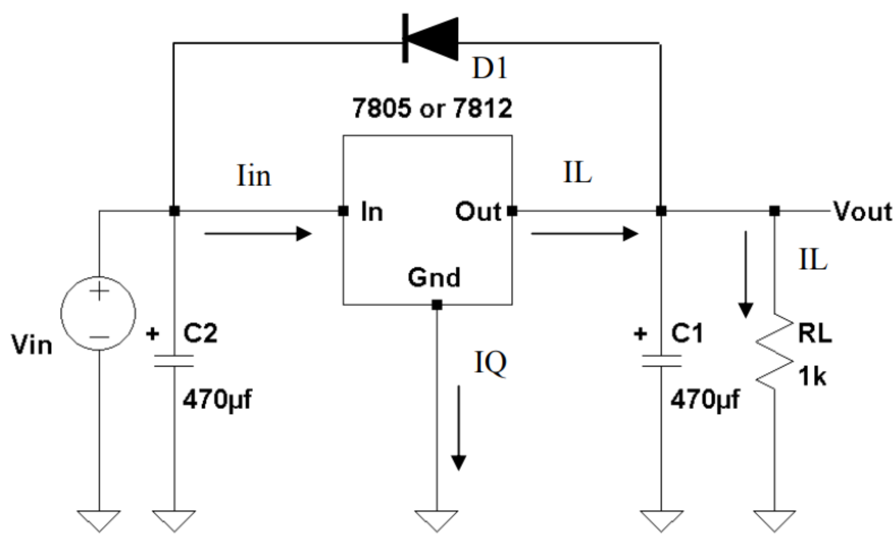


Figure 12: Voltage regulator [25]

Advances in Voltage Regulator Design

As seen in older models in previous designs in (figure), are not as feasible and less efficient as technology develops. Recently Manufacturers have developed high level of integrated circuits to reduce the size constraints of the device and power efficiency of voltage regulators. For our applications to implement onto the electronic LED sign device it is more usefully to incorporate the TI components of the TLV713 which uses an advanced internal control loop to obtain maximum stability within the system operation with or without the use of input and output capacitors. For this new type of device, it is designed to consume low quiescent current and supply line and load transient power. Also, very excellent characteristics include low noise and very high-power supply rejection ratio

(PSRR), making this the ideal decision for all mobile or portable devices. The power supply rejection ratio is used to describe the capability of an electronic circuit to suppress any type of power supply that affects the output signal. For the overall design of the voltage regulator there is a component called the under-voltage lock (UVLO) that the TLV713 utilizes. This circuit disables the output until the input voltage is greater than the rising UVLO voltage.

[39] This design will ensure that the device does not experience any unwanted behavior when the supply voltage is lower than the accepted range of the internal components, the minimal voltage. During this process, the under-voltage lockout disables, the output of the device, TLV713P portion is connected to ground with a 130 Ohm pull down resistor. The next component in the TLV713 voltage regulator as seen in figure (39) is the thermal shutdown, when the enable pin is active high. This enables the device by forcing the pin to read a minimum threshold voltage of 0.9V. The second case will turn off the device by forcing the enable pin to drop below 0.4V. The TLV713 also has an internal pulldown MOSFET that connects a 130 Ohms resistor to ground when the device is disabled. The discharge time is equal to the 130 Ohms resistor in parallel with the load resistance times the output capacitance. The fold back current limiter in the device has an internal fold back current built into the circuit, its main priority is to limit this current to help the regulator during fault conditions. It is also good to keep in mind the current to the supply is gradually reduced while the output voltage decreases. When the output is shorted, the LDO supplies the current of 40mA. When the current is limited the output, voltage will not be regulated. The PMOS transistor main role is to dissipate until the thermal voltage is equal to the thermal shutdown trigger voltage, the device turns off. To turn on the device is by another internal circuit during the cool down. There is also a built-in diode as you can see in the figure shown below. The diode conducts current when the voltage at OUT exceeds the voltage at IN. This current is not limited. The thermal protection core role is to disable the output depending on the change or variation of temperature. When the temperature increases to 158 Celsius the device output shuts off, allowing the device to thus cool down.

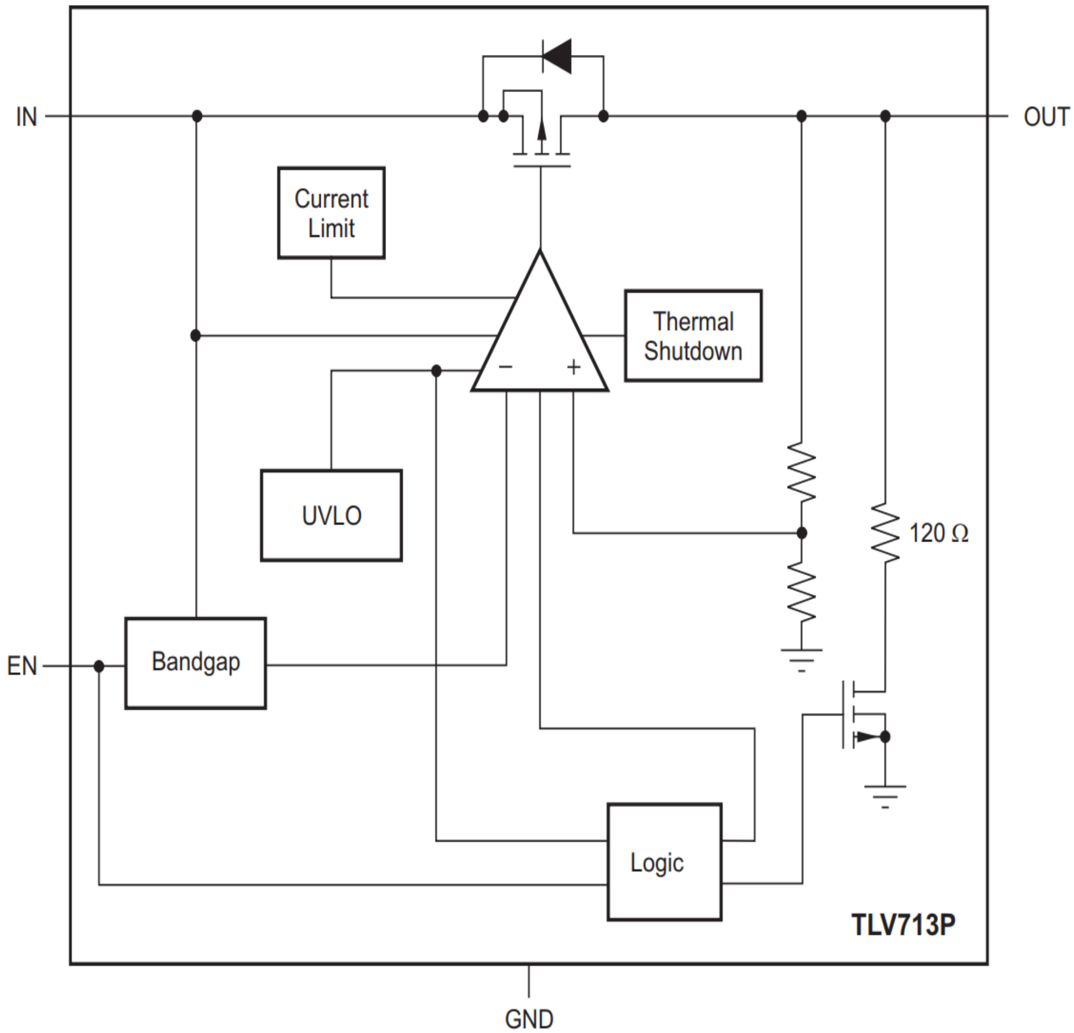


Figure 13: Regulator operation graph [39]

All in all, the devices normal operation of functionality is restricted to several conditions. The input voltage is high as the minimal input the device can contain. The input voltage is greater than the nominal output voltage added plus the dropout voltage. Also, the enable voltage has previously exceed the initial voltage threshold that is defined by the enable voltage and has not dissipated or leaked any current out from the input. The output current should be less than the current limit when coming from the voltage regulator and the device temperature is less than the junction temperature.
For an ideal voltage regulator circuit, the output voltage should remain constant. There are two types of voltage regulator linear regulators and switching regulators. The type for this project will include the microelectronic linear regulator due to the fact that it satisfies the design requirements for the our overall scope.

Also, this has a great trade off in price for higher performance. A linear voltage regulator is cost is ranged at 0.68$ per regulator. When looking at the voltage output vs input diagram as seen in figure 1. At initial input voltage ranges below the voltage threshold the output response is a linear response. Above this threshold point is the voltage remains constant.

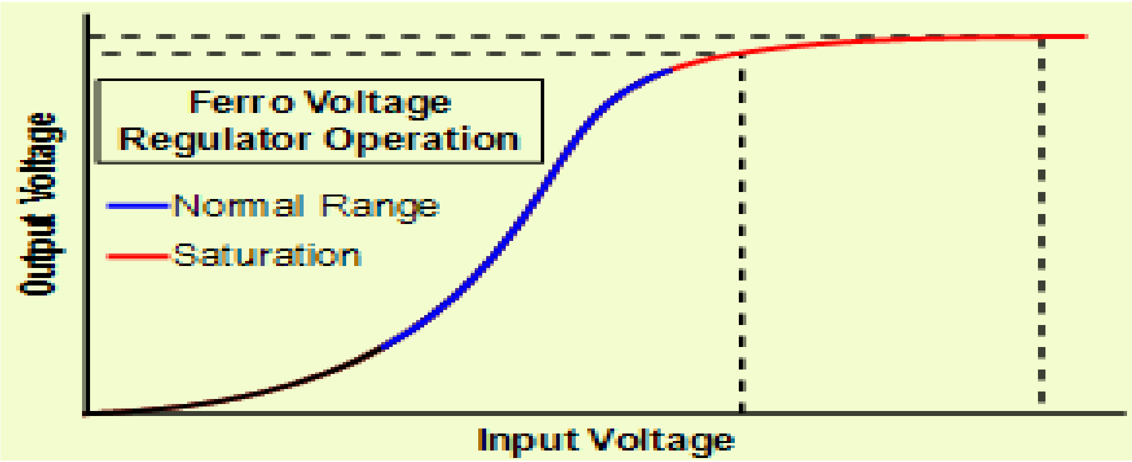


Figure 14: Regulator operation graph [30]

There are multiple parameters when looking at linear regulator. One important parameter is the load regulation. This unit defines the change Vout as the regulators output current, Iout is altered from the maximum current to the minimum threshold value.

Our goal is to use is to minimize the voltage across the regulator as much as possible, since the power of this sub device is proportional to the voltage across. Thus, increasing the efficiency = Power out/ Power in *100%(Page 3-5 e2l). This is a vital parameter that is the minimal amount of voltage across the regulator for the device to maintain a constant output voltage.

This minimum voltage across the regulator is called the drop off voltage. With this electronic component it has a maximum rating that must be adhered to, in other words there is a limit to allowable input current and input voltage. Finally, since these parts are short circuit protected, the max current is also given with the design specs. As this part does not exceed the maximum power dissipation and maximum operating temperature the device functions without fail with its output shorted to ground. There are three cases in which this linear device will not operate. The first is applying the wrong voltage polarity to the input.

The second case is when Vout greater the Vin. This commonly occurs if C1>C2. For protection for this circuit regulator we can add a diode across the output and input terminal. Where you install the cathode onto the output of the regulator and the anode which is the negative terminal into the input.

The diode is thus forward biased preventing any current flowing into the regulators from the output when Vout is greater than Vin. If this not the case under normal circumstances, Vin greater than Vout the diode is in reverse biases.

7.10 Universal Serial Bus

The device will be programmed, the electronic flip sign has an integrated universal serial bus (USB) interface, for programming the microcontroller. The USB communicates with the device and act as a channel or physical medium to send signals, as commands that must be followed completely. The USB has multiple advantages and functions such as a bidirectional communication from the main source via the host, known as the master to the device being implemented as the slave. Another vital feature the USB incorporates is the direct source as a power supply for charging, meaning you can use this universal device to plug into any type of source and draw current to the load[23].

There are distinctly four data transfer types Control which is regulated with devices such as sending commands between the device, Isochronous which is regulated with devices that are more time dependent and have no percent error in delay, such as cameras and audio devices, Bulk which includes a larger amount of information regardless of speed or time, and interrupts which are the focal point of the electronic flip sign because this is used by peripherals exchanging small amounts of data.

For the standard type of USB connection, A that we are using to connect to the port of the device, has a rectangular cross-sectional area. That includes four main wires that handle communication throughout the system, two pins that carry power Vcc+ and ground. The other two pins handle the data coming in and data going out of the device. For the other portion of the terminal port B it can be composed of different types of connection. For example, a mini connection which is 3x7mm. The more standard version for the microcontroller connection for the board will be integrating a micro USB which is 6.85 by 1.8 mm. USB is a serial bus.[28]

USB transmission is how data is transferred in between the data lines. This takes bytes of data and transmits the individual bits in a sequence. There are two states that define data as the J and K state. 0 bit is sent when toggling from J to K or vice versa.1 is transmitted if the state remains the same. To maintain a consistent signal transition for clock recovery is to happen in this bit field or bit stream, bit stuffing has to occur. In this case an extra 0 is applied to the data stream after any occurrence of 6 consecutive ones.

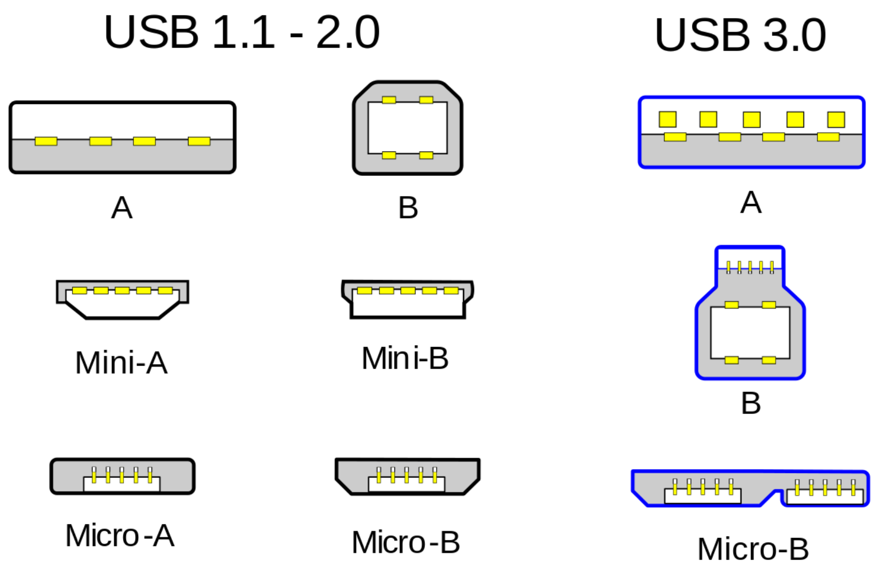


Figure 15: Type (micro) USB PORTA [22]

Interrupts are used from devices to obtain information from the pc host at a guaranteed a quick response. There are also special function registers for signaling these interrupts. The USB is also layered in subsections for the protocol procedure to send incoming and outgoing data streams. This is known as the protocol-layer. During this time of communication between the USB, data is transmitted as packets. Originally, packets are set from the host to other devices. A minority of these packets act as a subroutine to efficiently, direct a device to send some packets as a response or reply to the message. All packets contain 8-bits of vital data. Transmitting least significant bit first. The first bit is the (PID) packet identifier byte The PID has a total of 4 bits.

Packets come in three basic types. The main type is the handshake packet, which contains 8 bits. They only consist of a single byte. Error detection is then followed in response to see if the signal was obtained. This single byte can be represented in complement form. Which include three basic ACK asynchronous clock, indicating that information was successful obtained. NAK, indicating data cannot be received. And STALL, showing that the device has an error and cannot continue sending unless the correction has been made. The Token packet, consists of a PID byte in continuation of two payload bytes: 11-bit address and 5 bit CRC. In addition, USB that abide by token packets are only sent by hosts, and never the device. The tokens that are embedded in a micro USB follow a specific line of command tokens. These tokens consist of 7-bit number and a 4-bit function number. Those values contribute to how the data is being received and being sent out. The IN tokens expects a response from the device. This response can be one of the responses mentioned earlier NAK, STALL or ACK. The opposite operating function is the OUT token which is immediately followed by a bit stream. Meaning the device respond instantly sending out data and valued information including ACK, NAK, or STALL. Another command is the SETUP which is similar the OUT command, it will only operate for an initial device setup. The start of frame or SOF the USB host will send data every millisecond. Overall increasing higher bandwidth rate to the device. PING will communicate to the device and ask it if it is ready to receive OUT data accumulation pair. SPLIT is used to perform a unique function. Instead of sending data to a slower USB device, the nearest cable receives a spit token with two USB packets at high bandwidth. For the most important packet information that is provided by the USB is the data packets. Which also consists of PID and a 16-bit CRC. In this region, there are two data packets DATA0 and DATA1. Data packets have initial condition to allow transmission that include stop-and-wait ARQ. It begins with the a data packet must initially start with an address token and then followed by handshake token back to the receiver. If the host USB has not received a response for the data transmitted. This usually occurs when data is lost in transit or if the handshake response was lost. As a solution to this cause, the devices keep track of the data packets it has accepted. If the data is corrupted in anyway there are more checks and balances to maintain stable response.

7.11 Pulse Width Modulation

The PWM is the pulse width modulated signal. It is integrated into multiple circuits and electrical applications. For example, the electronic flip sign which is a mobile device. It is constructed to manipulate data signals using a microcontroller which has an integrated PWM. With this type of utility, it can be incorporated to encode a message into a pulsing signal. In the electronic sign, it main purpose will be able to control the power in the device. In which the average voltage applied to the load can be controlled by tuning the switch in between the supply and the load on and off at an extremely fast rate. In other words, the longer the time period the switch is to leave the device on there will be an increase in total applied power and vice versa when the power is turned off.

This pulsing frequency has to be much higher than to support the power supplied to the electronic device. This type of switch power on and off depends on the duty cycle. In which it is the portion of time spent on to a regular period. A low duty cycle thus corresponds to low power output. Since there is a ratio of time spent on and off. Duty cycle is expressed as a percentage in this case 100% means fully, 50%means on half the time and 0% means completely off.[29]

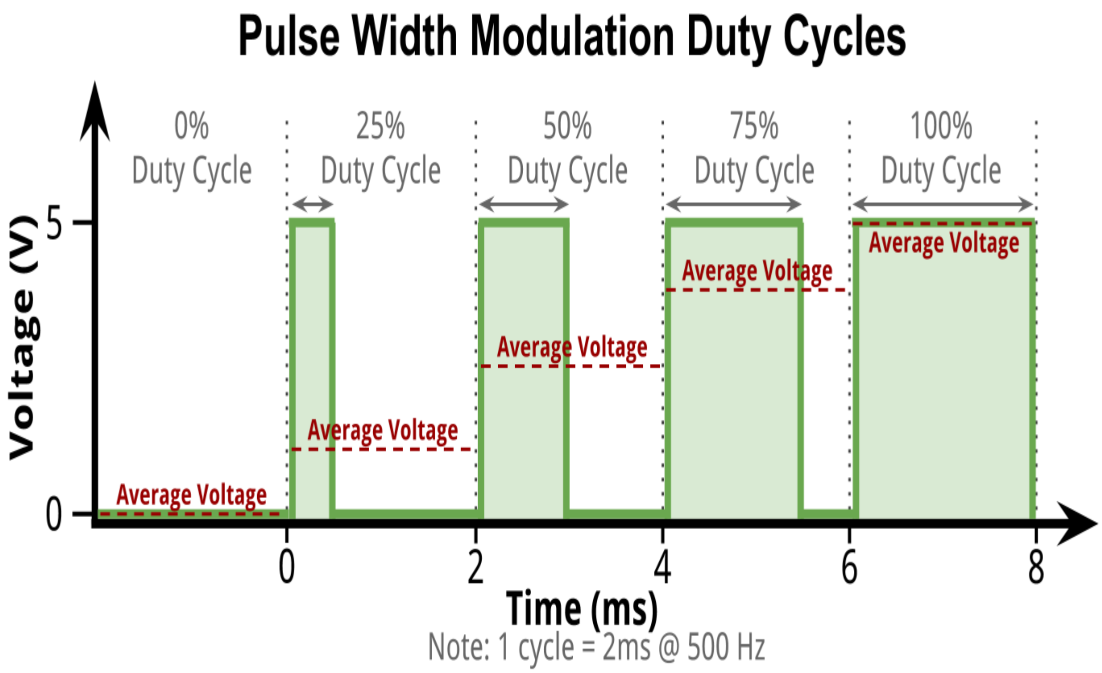


Figure 16: Duty Cycle %percent Value [26]

For example, the PWM we use to control the dimming on the LEDS. This is accomplished through the pulse wave allowing us to vary the amount of time the device is on in an analog domain. Since the signal can only be at +3.7V or low at ground 0V. We can thus adjust the brightness by altering the duty cycle.
The main advantage to PWM is the power loss is low due to turning the device off at specific intervals. Overall saves more power, thus being a more efficient method. When the device is off no current is flowing through the load thus the power, equaling the voltage times the current is totaled to net 0. Also, when the switch is activated there is no power loss across the switch since the voltage is also equal to 0. Utilizing a digital controller which is programmable through the user you can incorporate a rate in which you can manipulate any device to operate under the user's conditions. Can easier be set up utilizing the needed duty cycle.

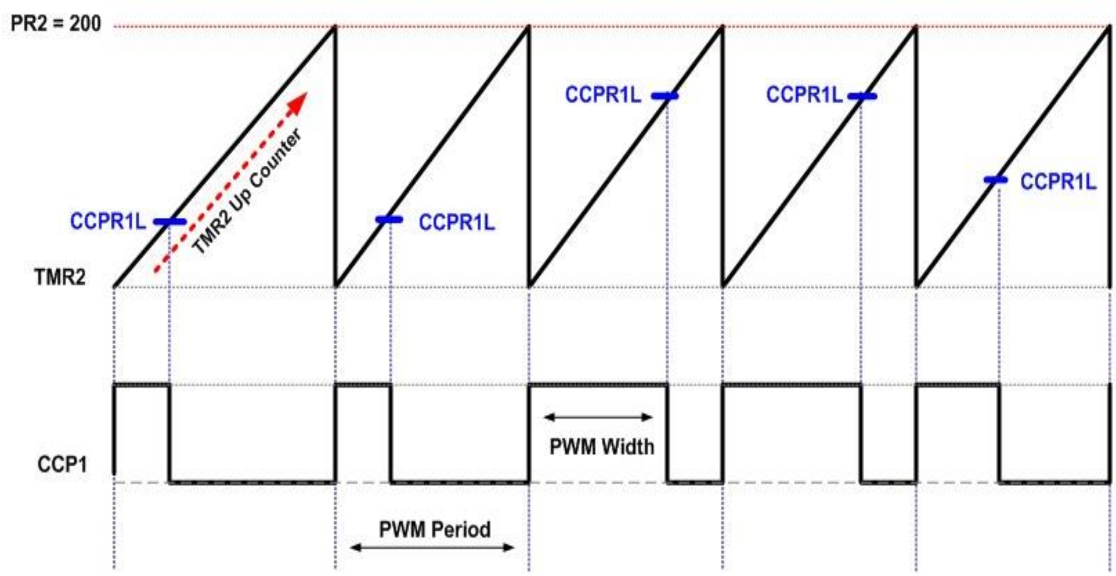


Figure 17: (PWM) Output Voltage with respect to incremental value [21]

The main principle and concept of the PWM can be constructed into a mathematical representation of a rectangular pulse wave. As its width equaling out to the average value of the waveform. The main way to create this waveform digital is through the saw tooth method. When incorporating a timer that counts up from an initial value sub 0 which can be set to 0. And incremented linearly by 1 every cycle and then slowly decrements down by 1 after it reaches it maximum value or peak value. We can have stored this information into a rectangular pulse waveform as a positive linear slope gradually increasing and negative slope decaying. On the pulse width modulator when the values are counting up we can leave it as a high value on the PWM after the comparator value programed into the controller. When the values a decrementing the waveform will be essentially off or all the values below the comparator will be also considered to be off. Depending on how much we design of chose to increment it to a specific value called the comparator value you can thus, manipulate the waveform that controls the device as a whole system. For example, the electronic flip sign has voltage regulators to turn on each individual LED in the matrix. In certain regulator which are usually switching regulators operate by switching voltage to the load that is manipulated through the duty cycle. When the voltage is below the desired value the PWM turns the switch on to regulate the voltage. When the voltage exceeds the desired limit the PWM shuts the switch off to turn down the voltage.

Programming with Arduino IDE Vs Atmel Studio 7

The Arduino IDE provides people with a simple environment to program in. This simplified environment helps make programming microcontrollers easy for newcomers. The Arduino IDE is one of the more common approaches when programming microcontrollers due to its use of the C programming language as the basis of its environment [33]. This IDE comes with plenty of built in libraries. These libraries are used to make the coding life easier for a programmer with a more streamlined form of programming the microcontroller. This is done by providing shortcuts when coding a program for a microcontroller. These shortcuts are created by using the Arduino IDE's standard functions and libraries, because the provided libraries and functions contain ways in which to create a process that would normally need several lines of code to manually create. For example, if a person wants to do programming over serial communication then the user can use the serial functions provided natively in the library. The native functions involve several processes running in the backend of the Arduino IDE environment. People who typically program using this software are hobbyists, due to the simplicity of the Arduino programming environment. This simplicity can be very helpful for us when creating our Electronic Flip Sign, due to the fact that we had a functional prototype that will be easy to test various features. The Arduino IDE was considered to be our final environment, but the Atmel Studio environment was also considered. This is because it has more flexibility.

The Atmel Studio provides an advanced programming environment for coders to program in. Atmel studio is used more for commercial products due to the advanced features that are provided. These advanced features allow the programmer to fully customize their program to meet the needs for their project. Programmers using this software can program microcontrollers using C, C++, and assembly. The specific version of Atmel Studio we researched is the Atmel Studio 7. This version can be used for the development of AVR and SAM microcontroller chips [34]. While using the Atmel studio, Arduino libraries can also be used, or programmers could use register operations to directly access individual registers. When using register operations, the user will find better performance when directly accesses the registers, instead of just using basic functions. Atmel Studio 7 also can import Arduino sketches into C++ projects [34]. Atmel Studio seems to have similar features and aspects that are shared with the Arduino IDE.

The Arduino IDE and Atmel Studio have some similarities. Both environments allow programming in the C++ programming language as well as programming in the C programming language. Both can also access the Arduino libraries as they are needed. The Arduino IDE and Atmel Studio environments can both be used to develop for AVR microcontrollers, such as the ATMega32u4 microprocessor. As a bonus, both have free versions to develop in, so we were able to practice in both environments before deciding, which will meet the needs for this project. While there are various similarities between the Arduino IDE and the Atmel Studio 7, there still many differences.

The differences between Atmel Studio 7 and Arduino IDE are found in the fact that Atmel Studio 7 is more advanced than the Arduino IDE. First of all, the Arduino IDE uses functions and libraries that are user friendly, but they do not allow further customization that may be needed for the project. The customization that can be done to functions in Atmel Studio 7 allows for more efficient processes than the ones you must use when programming in Arduino

IDE. As discussed previously, Atmel Studio 7 can also access registers directly and Arduino IDE cannot. Accessing registers directly can also allow more effective run times for interrupts and processes. There's also a lot more microprocessors that support Atmel Studio 7 and do not support Arduino IDE. This allows for a much larger selection of microcontrollers, so that the correct one can be selected to meet the requirements for your project. The main purpose for using the Arduino IDE over Atmel Studio 7 is the fact that Arduino IDE is more user friendly. The main purpose for using Atmel Studio 7 over Arduino IDE is for the amount of uses that it can be used for that Arduino IDE cannot be used for.

Interrupts

An interrupt is a signal that sent by a hardware component or from a program that tells the processor to stop what it is doing and do something else[48]. In the event of an interrupt is triggered the microcontroller will start an Interrupt Service Routine(ISR)[48]. An Interrupt Service Routine is a method that runs when the microcontroller is interrupted. Once the ISR event that the microcontroller was assigned is finished, then the program that was previously running will be resumed as long as it wasn't canceled during the event. Using interrupts helps bring multitasking to computing by allowing processes to take turns[48]. If there are multiple interrupts triggered then they will be placed in a queue. This queue can be configured by using a scheduler[48]. The scheduler can be created in the microcontroller's main code. To allow a microcontroller to use interrupts, the Global Interrupt Enabled bit and Interrupt Enabled bit need to be set[48].

Using interrupts will be crucial for the LED flip sign as this method can be used to get the device out of sleep mode. The microcontroller will use the buttons on the device to has a external hardware output as the activating component to release the processor from deep sleep and allow it to return to full functionality. When a button is pressed on the microcontroller the Interrupt Service Routine will run[49]. In the Interrupt Service Routine, the microcontroller will perform the main code and check to see what button was pressed, when the button is found the action that is required will take place to take care of the event.

Low power mode

Low power mode is when a microcontroller is put into a lower power state than when in normal use. This mode is a way to save power with a microcontroller when not in use and if battery powered, helps preserve the batteries life. There are many different variants of this mode that can be used to achieved a lower power consumption. These variants can be achieved from any combination of reducing clock speed, disabling external hardware, lowering the voltage, or by removing some functionality of the controller itself [47].

By reducing the clock speed this saves power consumption from the microcontroller. A person could reduce the clock speed of an Arduino if there are not that many instructions to be executed or if there isn't a need to worry about timing[x]. By dropping the clock speed, the current needed from the device. For example, having an Arduino run at 5 volts with a clock speed of 16MHz has a current running through it at about 12 milliamps [47]. After downclocking the microcontroller to clock speed in half to 8 MHz and keeping the same voltage, the measured current turned out to be about 8.5 milliamps [47]. By cutting the clock speed in half there is about a 66.67% decrease in power [47]. This result proves that when reducing the clock speed of the microcontroller that there is power that is being saved.

Power usage can be reduced by removing external components that are connected to the microcontroller. All components of a circuit use some amount of power. By removing components that are not needed at the time, this stops power from flowing to those components resulting in less power being used overall[47]. The LED flip sign will have many components connected to the microcontroller. There can be a sleep mode configured to remove all connections that will not being used during this time to save power.

By reducing the voltage going to the microcontroller this reduces the overall power usage. At 5 volts, the microcontroller uses about four milliamps and draws about 20 milliwatts of power. When the voltage going to the microcontroller is reduced to 3.3 volts the microcontroller uses about one milliamp and the unit draws about 3.3 milliwatts of power. By reducing the voltage by 1.7 volts there is about 7x less power being drawn by the microcontroller [47].

Features of the microcontroller can be removed to save power usage. There are components that are built into the microcontroller and each of these components draw out power. Given this information, by disabling some of these components this will result in some reduction of power usage from the microcontroller[47]. A programmer can prevent these features from being used by turning them off using software. Some of these components found in a microcontroller are the watchdog timer, a brown-out detect, and an analog to digital convertor[47]. The watchdog timer is used to prevent an automated unit from corrupting and having to wait for someone to manually fix the device. This timer will find out when a malfunction is happening and recover the system back to an operating functionality. If there is a problem with the hardware or the program of the device runs into a problem the timer will send a timeout signal. When this timeout signal is sent, the device will start correcting itself by restoring itself to a normal operating state[50]. The brown out detector is used to monitor the input voltage of the device and keeps controller on as long as it stays at a minimum value. The analog to digital convertor takes the analog voltage and changes it to a digital value that the microcontroller can use[47].

Timers

A timer is a circuit that is usually included on a microcontroller chip. Timers are very important when using microcontrollers. A timer is a good tool to help measure time by counting the clock cycles of the processor[51]. Timers usually have some size of a loadable count register, a common size is 8 bit. They also have an input clock signal and an output signal[51]. Data will be sent to the data bus of the microcontroller to count register and will start at a value with the range of the size of the register. An 8 bit register can hold a value between 0x00 to 0xFF and a 16 bit register can hold values ranging from 0x0000 to 0xFFFF. When the timer starts counting, it increments the value given from the software until the highest number the register supports is reached[51]. When this number is incremented to, the timer will overflow and an output signal will be initiated. This output signal can be used to set bits or trigger interrupts[51]. The timer can be reset by having a program send a new or used value to the count register. Counters can be set up two different ways. One way a counter can be set is to start from a value and count up to the maximum value of the register. The other way a counter can be configured is when the value is set, the counter will start to count from that value to zero and at the time the counter hits zero, the output signal will be triggered.

A semi-automatic timer is a timer that has an automatic reload capability[51]]. This type of timer uses a latch register to store the initial value. The processor writes to both the latch and count register in the beginning and then increments or decrements the count register. Once there is an overflow and the output signal is generated, the contents of the latch register are automatically pushed over to the count register and the cycle starts over from the beginning [51]. This timer can be used as a clock due to it's accuracy and automatic nature.

Using a timer with the microcontroller is a good way to measure the time that has been passed during a button press, while a message is being displayed on the LED flip sign, when to enter sleep mode, or when to check if an interrupt has been initiated. A timer is needed to see how long a message has been displayed so that when the time that the user set for the message to stay on for is up, that message go away and the LEDs on the LED flip sign stop displaying the message. Having a way for interrupts to be detected is very important for this project. so that the LED flip sign can be responsive to an input or signal and wake up from a low power state. If the device did not have the capability to wake up from a low power state then the sign will not be responsive for the user and thus become a faulty unit.

Charging Dynamic Speed

For design specification to be implemented in the electronic flip sigh, a mobile platform needs to last for an extraneous period of time and also be able to maintain that power to do so as quickly as possible for higher efficiency. It is ideal to find an optimal solution for accelerating charge rate. The constraints for faster charging were mainly concerns of high current withdrawal causing overheating within the system. With new generation of technology faster charging without this type of risk of overheating is possible. For instance, batteries now incorporate higher ion mobility. Overtime manufactures have further improved implementation of their designs. Recently in charging dynamics industries have a huge use in silicon to take advantage more stable and faster charging during the constant current phase. For great trade off that is produced by this process of accelerated charging is the quicker the charge, overall effects the capacity slowly decreasing over time when switching modes between constant current mode into the very slow constant voltage regimen. This type of features can be very useful tool in the design of our project since each diode takes about 0.67V to turn on each LED in a 12x18 matrix. For this design since the device capacity is so low on power consumption charging at 0.78C results in 60 to 70 percent 4-5V is reached.

By accelerating this rate to charging is less than 0.2C can result in a full charge at double the rate. Another valuable condition is raised temperature due to due to the electrons causing friction against the materials as current flow through the device itself. The manufacture might recommend that it is safe below a specific threshold but as an engineer any design can always be prone to defects. For example, as seen in the previous section “Charging Capacity” in the lithium ion battery cell can still be damaged. Resulting in reduced capacity and less duration of recharge cycles. Ultimately in this design lithium ion cell is the most effect design solution to implement in the electron flip sign LED matrix display. It offers multiple advantages such as high efficiency charging and energy storage for mobile devices.

7.12 Casing

The skeletal structure of the support bridge that connects each segment of the electronic flip sign is be composed of very efficient cheap material depending of the design can be implemented as metal, wood, or plastic. Using a 3D printer, pr laser cutter, we can model the design using a software package called splice. There will need to be two models that will be printed from a 3D printer, the handle and the casing that will hold the PCB, battery, and filter covering over the LEDs. By using a 3D printer the casing and handle can be constructed to the exact desired dimensions The handle for the flip sign can be designed using the 3D modeling software. Once the handle is designed, then the model is sent to the 3D printer which will print the digital model into a physical object. The next object to be designed, is the casing. This will need to be constructed so that it fits the PCB in a snug secure fit so that none of the components inside will move around while the Electronic Flip Sign is in use or being transported. We take into consideration structural analysis of trusses. At each joint there will be a support hinge that will apply the same amount of resultant force to keep and maintain the structure in equilibrium. This type of model will also take less amount of space and wait as a trade off to also conclude with the effective least amount of cost.

Another keen feature that can be implemented is layering the electrical components of the PCB board and wiring to decrease the device size. Making it more manageable to integrate more components into the spacing of the case. To define this skeletal structure we also applied software using matlab code. The skeletal structure is defined by a series of linear components. In a complex domain of a higher order structure you would need to solve for a non-linear structure do to tension and compression. You can set up a system of equation that include the sum of forces in x,y, and z direction. Since these are non-linear equation the only way to solve for the solution is solve through tedious repetition. A computer can solve these repetitive problems with seconds. With the analysis computation is complete we can finally test each weighted component onto the electronic sign to find a stable enough structure for the cheapest value.

As a result, technology will further develop, advance and its disadvantages and drawbacks. For example, Ion mobility being one of the greater of the weaknesses. As for our design for the electronic sign we are able to by choosing a battery that allows flexibility in the ideal charge rate. This can also be implemented into further programming the chip on the micro controller for power save option such as sleep modes. While the device is over it will ultimately conserve power increasing capacity thus also increasing the charge rate since it is proportional to the speed of the dynamic system. Rapid charging is a great solution to solve design specification demands that are implemented but also can bring into account other dangerous factors that are prevented using high safety protection circuit embedded into the device.

7.13 Solar panel power source

[42] Implementation of a Solar photovoltaic (PV) panels can be implemented to recharge our power supply in the electronic flip sign. Photovoltaic technology converts sunlight energy into electricity to power our design. The Photovoltaic cell is composed of a cell that is composed of semiconductor layering material mainly composed of silicon crystals. For the crystallized structure of silicon, it is at its state when it’s in a pure element. This mode is called the semiconductors intrinsic state where the semiconductor is composed of multiple pure elements and all the electrons are in their natural ground state.

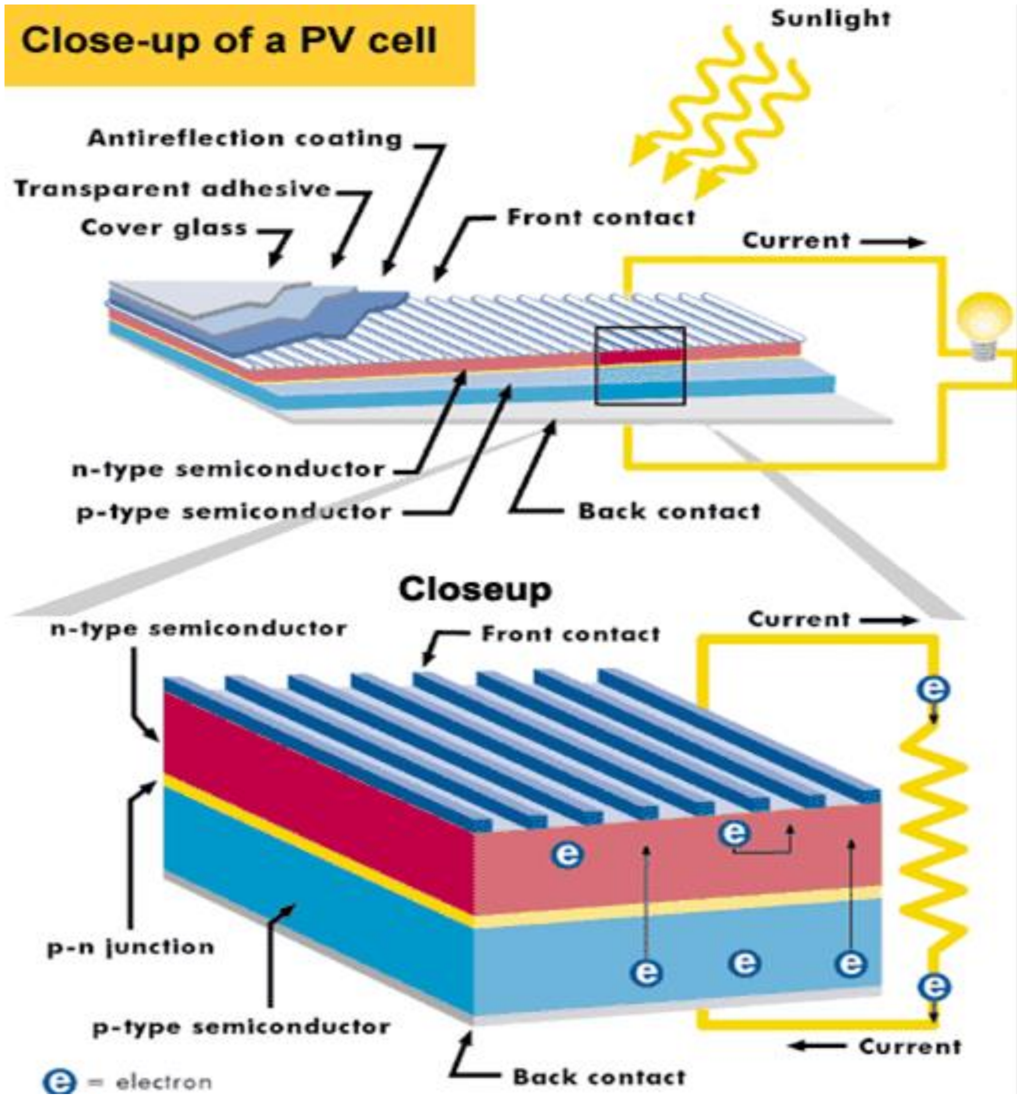


Figure 18: (PV) Photovoltaic cell[41]

In the semiconductor has unique special properties such as passing current in one direction than the other, showing variables resistances, and sensitivity to light or heat. To create a current flow you can add other impurities to this pure intrinsic semiconductor, with this high change is carrier concentration this due to the process called doping. During this stage of creating electrical current, the bottom of the photovoltaic cell is usually doped with boron, which has a valence electron of three. When bonding with silicon in the semiconductor boron will donate four electrons to silicon leaving behind a vacant space called a hole. This absence of space is called a hole. After bonding these two elements together the remaining concentration is non-intrinsic meaning the after doping has occurred from adding extra impurities the hole carrier concentration has rising. Leaving an overall p-type semiconductor. While the top layer is doped with phosphorous, which also bonds with the silicon atoms in another region to form a net gain of one electron per each bond. This creates a n-type doped semiconductor region.

The resulting p-type and n-type region is called the p-n junction. When the sunlight impinges in the cell of the photovoltaic panel, the optical energy physical knocks electrons loose in both layers. Since both of the layer have opposite polarities the electron will want to flow from the n-type region toward the direction of its opposite charge the positive p-type region. Due to the junctions electric field it will prevent the flow of electrons. Then the circuit need to be biased to meet the condition of reducing the magnitude of the electric field barrier preventing the electrons from moving and producing current.

How to bias the circuit you need to add a wire to each terminal and apply more electrons in the n-type region to overcome this barrier. A photovoltaic system are based on single square cells. By themselves, each cell generates very little power through-put (Watts), thus in general each individual cell are grouped together called modules or panels. Panels can also be grouped in unit arrays or matrices to harness more power through the sunrays.

Types of solar cells

[42] Single-crystal cells are designed and created in long cylinder sliced into thin wafers. They are the base material for silicon chips used in used in all electronic equipment today. During the process of creating the single-crystal has a high cost in energy and also uses more energy, to create high efficiency cells, to translate incoming optic light into electrical energy. The efficiency scale for single crystal cell are over twenty three percent. The other type of cell is the polycrystalline cells which are made from molten silicon cast into ingots then sliced into squares. The advantages of polycrystalline cells are the lower cost. As a trade off the structure is low on efficiency as well. The third basic solar cell is the thin film cell. In developing these devices, it involves depositing material onto glass film surfaces or metal films. Overall making the entire module in one swing instead of building each individual cell. Just like the polycrystalline cell this panel has lower efficiency but cheaper cost a balance of interest.

Efficiency of cells

Solar cell efficiency can be split into multiple subsections such as reflective efficiency, thermodynamic efficiency, charge carrier efficiency and conductive efficiency. The total efficiency is the total combination of all of the following efficiencies. A solar cell has a voltage dependent efficiency curve, temperature, and angels. Since the design measurement are very difficult to measure the parameters directly. Other values were substituted to replace these quantities. For example, thermodynamic efficiency, quantum efficiency, ratio, and fill factor. Reflectance losses take into account of quantum efficiency under external quantum efficiency. This mainly occurs due to the physical phenomenon of recombination which is the process of electrons filling up energy levels in the ground state to fill in a hole in a semiconductor. Recombination is another process of quantum efficiency. Resistive losses take into account of the fill factor.

The fill factor is the ratio of the absolute maximum value obtained by the power of the product of the open circuit voltage and the short circuit voltage. This vital and useful parameter in checking for performance. Cells with a high fill factor have a low equivalent resistance and a high equivalent shunt resistance.

Thus, the overall current is conserved throughout the system and less of it is dissipated in other internal losses. Specific semiconductors are reaching their limits of a single p-n junction are approaching a theoretical power efficiency limiting coefficient about thirty percent. The materials of solar cells mainly semiconductor elements that are used to compose the photovoltaic panels.

This element of material must have unique and special characteristics to absorb sunlight. For different types of efficiencies are can be categorized by this type of materials such as the single junctions or for multiple configurable junctions are the pnp to take greater advantage of types of absorption and charge separation mechanisms. For the higher efficiency cells, there are multijunction cells that pertain to multiple thin films, each staked with a solar cell on top of each other. Each layer has a different band gap energy to allow it to absorb more emitted light radiation of specific portion of a spectrum. These cells produce very high efficiency over all energy output. There are also triple junction cell as such semiconductors that are formed with gallium arsenide and germanium, today multijunction solar panels are the most efficient compared to any other design. There is also research in design to create an even higher efficiency rate with hybrid photovoltaics wafers.

Integrating Solar panel network

For the electronic flip sign in order to recharge the lithium ion rechargeable battery without the dependency of an external AC source. It is a great idea to extract energy from a renewable energy source such as light being emitted from the sun. Utilizing special electronic device such a photovoltaics(PV) panels we could connect and operate a self-support system at any time or rate needed on the go. For the first part to extract energy from the sun we need photovoltaic panel to absorb the sunlight to be processed into electric DC current. With the ideal solar panel to be selected it will be needed to fully supply the electronic sigh matrix to pass each threshold voltage on the diodes to activate the sign. In addition, to this concern the sign will also have to supply the constant power for over 15 mins will supply and produce a constant 4.5V at up 90mA, of course this is the maximum power it can supply if need to go below this minimal current threshold then for the electronic sign the device can still operate without being damaged.

For testing purposes to detect power from the solar panel is to place a load in front of the device and measure the voltage and current according with a voltage reader. We can also test for interruptions when the sunlight is out and the cloud can interrupt this light ray be emitted from the sun. Thus, there is an inefficient amount of energy and a waste of power. To save energy we can connect our rechargeable battery to the positive and negative terminals of the solar panel and store this energy. The lithium ion basically acts super capacitor storing charge within the module. For our design using the electronic flip sign the battery is great tool to supply an efficient amount of power delivered to the entire system since solar energy cannot produce as much do to the design specifications. Mainly due to the size of the solar cell has to be decreased in order to make the electronic sign more lightweight and durable. When adding the battery, it is important to prevent the solar panel from leaking any current. To prevent this limitation, between the battery and the solar panel a diode can be placed in between them blocking any reverse sinking current. Thus, only allowing current to pass one way. The only issue and concern about this design is that the LED is on all the time so it very energy inefficient. To address this problem there needs to be a feedback current that can sink and be stored later inside the rechargeable battery. To further add other external devices such as the micro controller to the charged circuit. The microcontroller needs very low current, efficiency and stops this process by controlling the led and shutting them off to save power utilizing the built pulse width modulator as seen inside the microcontroller.

8. Design

The design for this device is made for ease of use transportation and functionality. The case for this device is the most extrusive part of this device so it will have the end dimensions for the design and the only things that protrude from the device is the six buttons and the power switch. The device has two major parts to the design, the display module and the handle. The display module houses all the components inside as there will be no components located in the handle. The handle is designed to be sturdy yet lightweight. At the joint of the handle to the display there it is the strongest as this will be a weak point while in use. The handle is about 4 inches long and cylindrical in shape with a diameter of about a 1.5 inches. This joint has to withstand at least 2.5 Nm of torque to sustain use and waving without breaking. The display module is about 10 inches wide, 4 inches tall, and about 1 inch thick for its dimensions. The design for the casing front there is the PCB with the LEDs and behind that is the battery and connections. This display casing has holes for the micro USB the power switch and the six buttons to be fitted onto the display. The buttons all come through on the second board above the handle. The case is made of laser cut wood for the prototype and for this reason weighs more than what was expected from initial specs and the handle is layered sheets of wood making it strong but not removeable in the prototype state.

Another aspect of the design is the building process and the product relations with each other. The building process has a large effect on the design, if the building process is not thought out in the design you can run into a number of problems. If building is not designed for then there could be components that are placed in relation to another component that would make soldering them really hard and even impossible and some components could even interfere with others. Another thing that could cause trouble is the clutter of thermal producing components can cause hot spots that could have caused overheating. To be well designed the components had to be well spaced out this makes the components to better distribute the heat produced by the components, but it also made it easier to solder on by hand and even for a professional pick and place or other methods spreading the parts out can make it easier. The next part of a good design not only has the components spaced out but also makes the design of the PCB and the traces to be less scattered and less of a mess and more organized this will make it use less traces and less potential interference and less possible loss from long traces.

8.1 PCB

The PCB (Printed Circuit Board) is the main board that connects all the components together for the final product. The PCB is a board that has wiring inlaid to the board and at certain points it has the wiring surface to allow for connections at pads that can be soldered to. The PCB can be multi-layered, meaning that there can be more than one layer of the inlaid wire allowing for cross wiring, this is necessary as on a single layer it would be hard to make a good number of connection because you could not cross any of the wiring on a single layer. PCB can have many layers most notably 2-layer and 4-layer

The PCB for this project will be one of the largest pieces of the project and there is more than one way to design the PCB. Using 2-layer PCB to design this project uses multiple boards in the design. The LED’s uses their own PCB in a 2-layer design, so there is a single PCB that on one side would have all the LEDs on solder points and they are all be wired according to their design in the PCB and then on the back there is a connection 10 pin header to then connect this board to a second PCB. The second PCB for the 2-layer design has all the other components other than the LEDs these are located on one side on their solder points and they is wired in the PCB according to design and there is the other connector for the header pins to connect the boards allowing them to communicate and this board also houses the buttons that go through the casing to allow for user input.

The second design type that could of been used is a 4-layer PCB this design would of been similar to the 2-layer design but there would be no ribbon cable as they would be connected within the PCB. In this design, the device would have only one PCB and one side would have the LEDs and the other would have the components as well as the buttons. The cost of 2-layer runs about $33 for up to 60 square inches and then the 4-layer for about $66 for up to 30 square inches, the LEDs are going to take about 40 square inches so using the 2-layer PCBs it would be about $66 and then twice that for the 4 layered boards making the 2-layered PCB a more valuable option to use.

8.2 LED Wiring

The wiring for the LED matrix is done in rows and columns. When the LEDs are controlled using a grid layout with the ground on one axis and the power on the other axis and then when they are crossed the led will light. To accomplish this the LEDs are wired as shown in the figure below with the cathodes of the LEDs attached to the rows and the anodes attached to the columns but the rows and columns are not attached at any point. To prevent the display from showing the same line pattern across all the columns this is what the shift register is used for, when running only one column will get power at a time and then it will move to the next and can then change the pattern but it can do this faster than can be received by the human eye. It can be seen from this design that the number of rows and columns can be changed and is only limited by the power and that it can be turned so power is on the rows and ground is on the columns as well. There is some other ways that the wiring can be changed the biggest effect of this is the segmentation of each of the length and width of rows and columns, this can be changed to make the system a long row of 10 high stacked on top of each other or a single row 20 high this is a way to lower the current draw by a single column but taking long to make a full cycle thus needing a fast processor to make a constant feel. This design will use a longer column as the power usage at 10 is still higher than what is acceptable for the driver chip, using this method will also reduce the need for a lot more driver chips needed for using two rows separated.

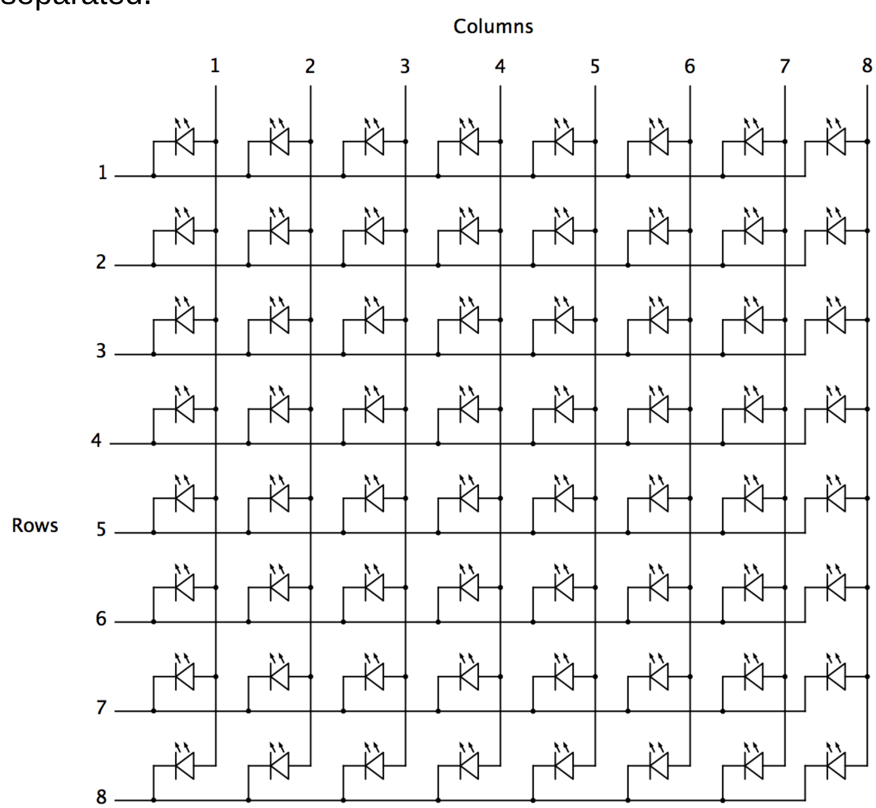


Figure 19: The LED wiring layout [14]

8.3 LED Layout

The LED layout is the pattern and spacing for the placement of the LEDs, this affects how they look and how they are perceived. This project specifications call for a viewing distance of up to 45 feet. To accomplish the set specification there are a number of factors that go into making the viewing distance best at the given range. Some of the things to consider is the brightness of the LEDs, the color, the filter, character size and style, and the spacing of the LEDs from each other.

The spacing of the LEDs changes a lot. The spacing with a set display area sets the maximum spacing based on the character design. The design of the characters and size have to be a style that is easily readable from the desired distance, for this project there will be a character height of about 2 inches while displaying two lines of text and about 3.5 inches for a single line of text, this size for the two lines of text can still be read at a distance of 45 feet and the single line can then be used to make the statement more pronounced or readable at greater distances.

The LED spacing is the distance from center point to center point of each LED and even though the character size can set the max distance is does not limit the lower end this is because if the spacing is too large the character are unreadable but at smaller distances the characters will just get more detailed. The goal here is to find the medium of a readable text but not to have too many LEDs causing greater power use. The LEDs can be placed so that if you were to have an “l” displayed on screen it could be 1 LED wide or 2 or more wide. To get good results while not using too many LEDs the spacing for this project with its given display size and character size is set to be at 5.5 millimeters spacing. The spacing of the LEDs can be seen in the picture below as shown on the prototype perfboard layout, note that the measurements appear off in the photo but each LED from center to center is about 5mm as seen in the first two columns of LEDs where they line up with the ruler best.

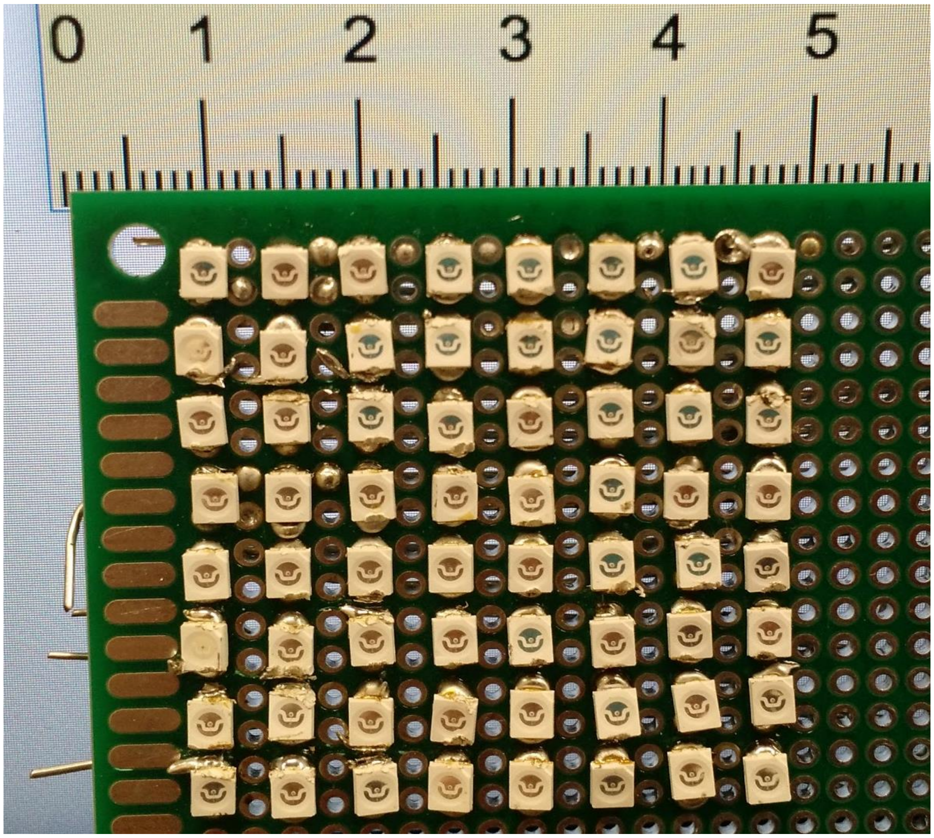


Figure 20: The perfboard LED layout (measurements is cm)

The filter is a piece of glass or plastic that covers the display going over the LEDs. The filter has a few roles first off it is a part of the case and is there to protect the components from outside elements and fall damage. The other purpose of the filter is to be a filter, although you do not need the filter or cover to be tinted or altered in any way and could be just a clear filter but you can to have certain effects. If the color is changed then this is a way to get the desired color output from the display even if white LEDs were used. If you make this filter have a fuzzy look like a fogged glass this can act as a light diffuser and will blur the light going through it and this can be of use for the display to make the characters more apparent by blurring the light to kind of connect the dots. This can make the display look nicer and smoother but there has to be a sweet spot as you don't want to blur it too much to make the characters less readable also you don't want to lose too much light using a filter. The LEDs can be individually cased from PCB to the filter this will prevent the blur from mixing the light too much making the characters unreadable but still lets the display have a smooth look.

8.4 Device Input

The input that is located and received through the device. The first input is the 6 buttons these will be located on the back of the device relative to the LED display and just above the handle. The handle is round and designed to be held with one hand and the device can be operated with the use of the palm and four fingers to allow for free use of the thumb. Since the handle is designed to free the thumb it can be used to press the 6 buttons that are located in reach of the thumb while using the device. The next input is the power switch which is located on the top left of the device, this switch is located so that it can be operated while the sign is being held in the right hand. Next the micro USB port is located on the left below the power switch for charging and programming. Last is the ambient light sensor which will be taking is the ambient light to adjust the brightness of the LED display when in auto brightness mode, this being the case the sensor will be located on the front of the display with the LEDs so that the light reading is relative to the light upon the display so that the auto adjustment will be the most appropriate for the given light situation.

8.5 Powering the device

The LED flip sign is powered using a 3.7 volt lithium polymer battery. The battery is connected to a charging circuit. From there this will power the microprocessor and the rest of the device. When the battery's charge becomes low, the user can connect a micro USB cable to the device's micro USB port.

While the device's battery is being charged, a symbol of a battery will be present on the LED display with four sections on it. Each section will light up depending on the charge percentage. When the battery reaches maximum capacity, the battery symbol's sections will stop flashing and will stay continuously lit.

8.6 DC Boost Converter

A DC to DC boost converter is required for our device in order to meet a voltage requirement to supply sufficient power to the LEDs when using a 3.7 volt battery. A basic boost converter as shown in figure 15 below consists of the voltage source connected in series to an inductor, this inductor is then connected to a switch that goes to ground and a diode, the diode is connected to a capacitor and a resistor in parallel. The voltage across the resistor and capacitor is the output voltage and would be higher than the input voltage. To achieve a boosted voltage the inductor plays an important part. When the switch is closed, the inductor begins storing energy. When the switch is closed, the inductor will release its stored energy, acting like a voltage source. When this happens, the input voltage will be added to this voltage and flow through the circuit allowing the output voltage to be higher than the input voltage. To achieve an efficient and agile method of flipping the switch, a MOSFET transistor can be used in place of the switch[56].

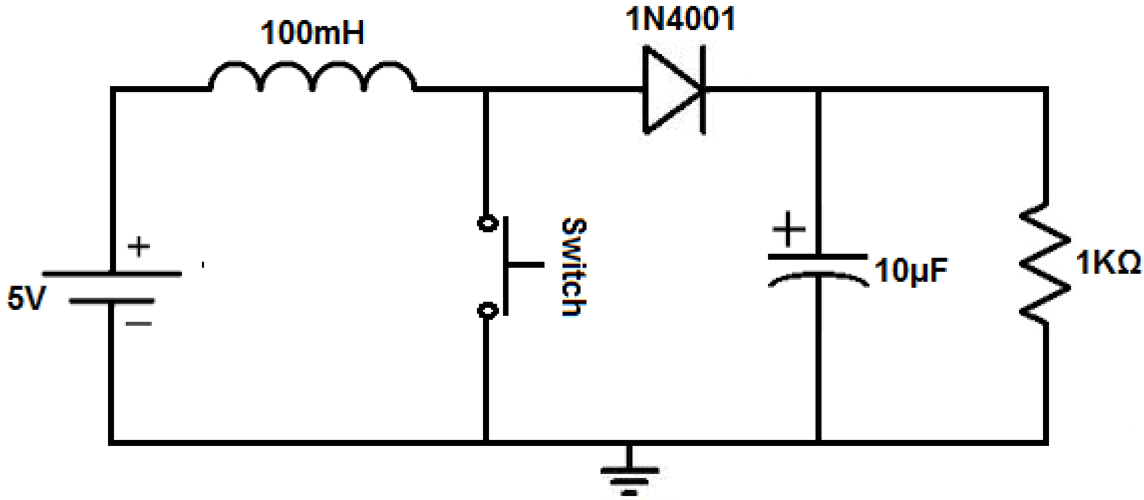


Figure 21: DC Boost Convertor[66]

One of the designs for the LED flip sign was going to use a DC booster circuit in order to create more voltage and have more power flowing through the board. This will allow more current to flow through the LEDs. With more current the LEDs could receive the maximum current they are allowed and this would provide the ability for the LEDs to achieve max brightness. This design will no longer be used for the device as a more efficient way is to power the device with the 3.7 volts that the battery supplies, this is because the LEDs being used take a max voltage of 2 volts and anything over that will only account for losses. By making this decision, there is a better overall runtime duration of the device on a single charge.

8.7 Charging Circuit

For the electronic power supply. The charging circuit contains multiple systems including the power AC input transformer, diode rectifier circuit, smoothing, Filter, and voltage regulator which is connected to a load. As shown in figure 1, the circuit charges the battery.

Power Conversion

The rectifier forms the first stage of a DC power supply, main concern is during the initial phase from transferring from AC equivalent voltage into alternating current from a positive and negative amplitudes of the sine function into positive response to get a more general function in terms of a constant regulating voltage. Basically, this is a process that converts an alternating current (AC) voltage that is limited to one polarity. This the first component of the charging circuit which is built in an arrangement of diodes that only allows current to flow in one direction. This will invert the negative portion of the AC signal into a more manageable solution.

The diode nonlinear characteristics will perform very vital tasks, current exists for one polarity, but is essentially zero for another. There are four diodes formed as a bridge seen in figure 2, this is the full wave rectifier unlike a half wave rectifier is more efficient since it does not obtain the full solution on the negative AC voltages not being inverted as half the magnitude response is equal to zero. A full wave rectifier allows current to flow during the input sine waveform, which can be verified as seen from an oscilloscope.

Each diode contains a voltage value V1 for voltage value in diode D1, V2 voltage value for diode D2, voltage value V3 in diode D3 and V4 for the voltage stored in D4. Each diode has the same cut-on voltage V (gamma) which is equal to 0.63V. When Vin > 2V(gamma) the electric field produced by the source of the Vi is in the opposite direction of electric field produced in the junction of the diode, since the electric field moves in the direction of high potential of the cathode terminal to low potential the anode. As a net result, the junction in between the diode is decreased thus turning diodes D2, and D4 are thus forward biases.

This will allow current to pass through the circuit. And since the electric potential in diodes D1 and D3 are in the same direction the bandwidth of the pn junction will decrease leaving them in reversed biased. If Vin < 2V(gamma) then the polarity is flip then the opposite outcome will occur D2 and D4 will be reversed biased and D1 and D3 will allow current to flow from Vin through the diodes to the load in the forward active region. When Vin>-2V(gamma) and Vin<2V(gamma), all diodes will be cut-off. The net result as displayed in figure 3, gives you a sinusoidal wave with inverted negative amplitudes and a slight voltage drop.

Filter

After producing a sinusoidal AC inverted waveform, the filter now converts AC into an unregulated DC voltage. Initially we obtain a voltage value from an AC source then that AC source is transcribed into a positive magnitudes of sin waveform. The next chain in the DC power supply is to convert this waveform into an unregulated DC voltage. With an inverted sinusoidal input, we obtain a DC value closer to the desired results.

To design a simple filter, we must first take into consideration the transfer of energy from what is placed into the system and what will leave the system. This type of characteristic is defined by the transfer function, that the output signal is related to the input of the system through an equation denoted as convolution in time domain as y(t)=integral of h(t)*x(t)dt or for a simpler solution H(s)=Vo(s)/Vi(s). This type of notation is dependent on frequency which is signified as s=jwo, where wo=2*3.14*frequency.

This type of analysis is formed from the Laplace domain which transfers nonlinear complex terms into simple algebra using superposition to form the results. If the system is linear then the total sum of the inputs should equal the total sum of the outputs. With these types of function has its own type of characteristics that are classified as filters. In this case for the incoming sinusoidal voltage the filter we will use is a simple RC filter. When a capacitor is added to the load resistor of the full wave rectifier, the waveform will begin to transform the sinusoidal output into a DC voltage. Initially, the during the first cycle the voltage across the capacitor follows the initial portion of the waveform when the rectifier diodes are under forward bias until signal voltage reaches its peak and begins to decay, which means the capacitor means to discharge at this time.

The only path to follow when the capacitor is discharging is to flow through the resistor since the diodes are off since the current can only flow in one direction for the diodes. The time it takes for this amount of charge depends on the RC time constant. Since the capacitor the larger portion of the AC sinusoidal signal. This is known as the filter capacitor. In time, causing the ripple effect as seen in the figure below. Overtime we can obtain an approximate DC value.

8.8 The Graphical User Interface

The Graphical User Interface, also known as the GUI, is used to help make the operation of software and other electronic devices streamlined, so that there is an easier way to operate the software without using the command line. Some of the features that are used when creating a proper Graphical User Interface are icons, dropdown menus, sliders, text boxes, etc. According the Argon Design, there are “five aspects of a good user interface” [12]. These five aspects are as follows, “Intuitive and Consistent Design”, “Clarity”, “High Responsively”, “Maintainability”, and “Attractiveness” [12]. “Intuitive and Consistent Design” means that once you have a certain design for your interface, you should not drastically change this design for the sake of the user. If you did a complete overhaul of the GUI after the customer is already acclimated in the current interface, it might end up being less user friendly to change it then it would be to just modify the current design. “Clarity” is rather self-explanatory. If the user is unclear as to how to function the interface, then it is not user friendly. “High Responsively” involves making sure the interface flows properly and that the functions of the Graphical User Interface are responsive. “Maintainability” simply means that the interface should be able to be updated without issues. The last aspect involved “Attractiveness”. This involves making the Graphics for the interface are aesthetically pleasing to look at. This means that the design is simple, has a good color scheme and arranged properly. When design our Graphical User Interface, all five of these aspects were used to make our design as user friendly as possible.

The first design for the Graphical User Interface can be found in figure 10 below. Figure 10 shows the custom window for this design, but before this window we will see the connection window. This window shows whether or not the Electronic Flip Sign is connected via Bluetooth, wired or not connected at all.

After the device is connected, the software will then open the custom window seen in Figure 10. This Graphical User Interface operates by first having the user select which button they want to edit first. This is done by toggling one of the buttons by pressing one of the circles next to the number that corresponds to which button the user wants to edit.

Once the button that the user wants to edit is selected, the user can now move onto editing the message and features. The interface below shows two textboxes. The first one is called Line 1 and the second one is called Line 2. This allows the user to type a message that takes up two lines with Line 1 on top and Line 2 on the bottom. The two lines option can be turned off by toggling the One line Text option. If the user types in a message that is too long to fit without scroll, the scrolling option can be toggled on to have the message scroll across the screen. The Scrolling speed can also be selected by toggling the speed with level one, level two, level three and level four as the four options available. Brightness also can be set with toggling levels one through four, much like the speed levels. These brightness levels will be available to select if the brightness mode is set to manual. If it is set to auto, then the light sensor will take over the brightness adjustment. Once all the changes the user wants to make have been made, then they will press the Save button which will trigger a dialog window that will ask if you are sure you want to save. If yes is pressed the changes will be saved, and if no is pressed they will not be saved. Once the changes are saved, the user can then move onto the next button they wish to make changes too.

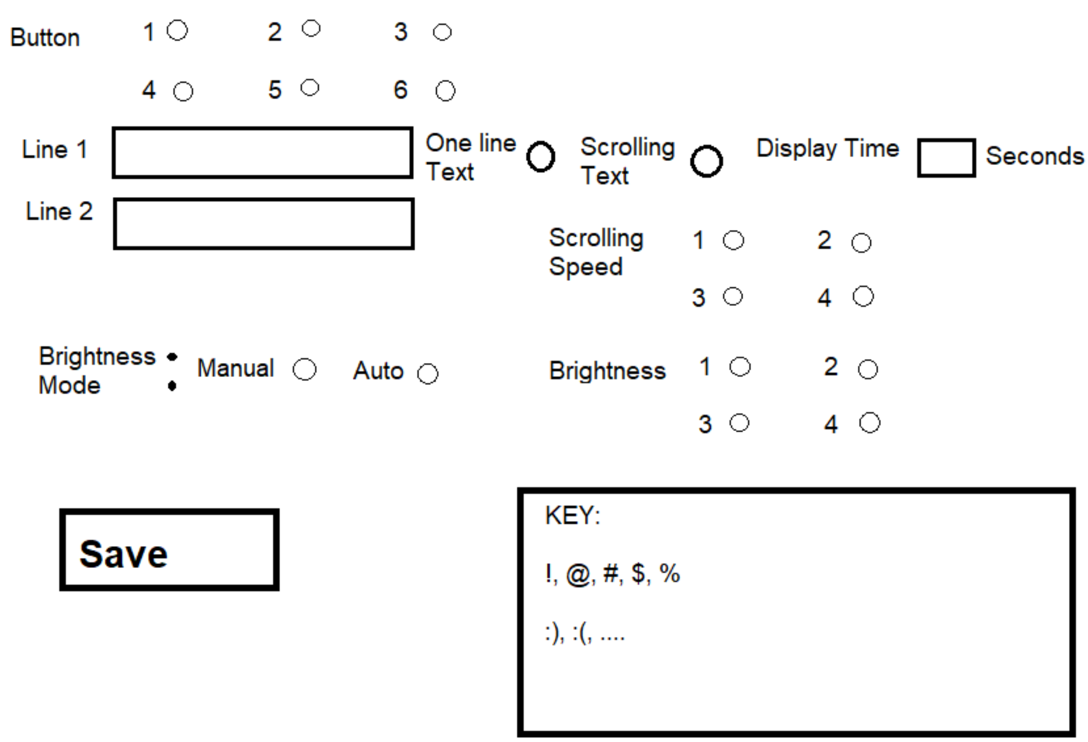


Figure 22: GUI Rough Model Design 1

A Second design was formed for the Graphical User Interface as shown below in figure 16. There will be a Connections screen (refer to figure 10) that will ask the user to connect the device if not already done so. This connection is done using Bluetooth or through a USB cable. Once the connection is successful a new screen will be displayed with the customizable options. This screen has six sections, one for each button on the device. The first long text box for each button is where the user can type the message they want displayed.

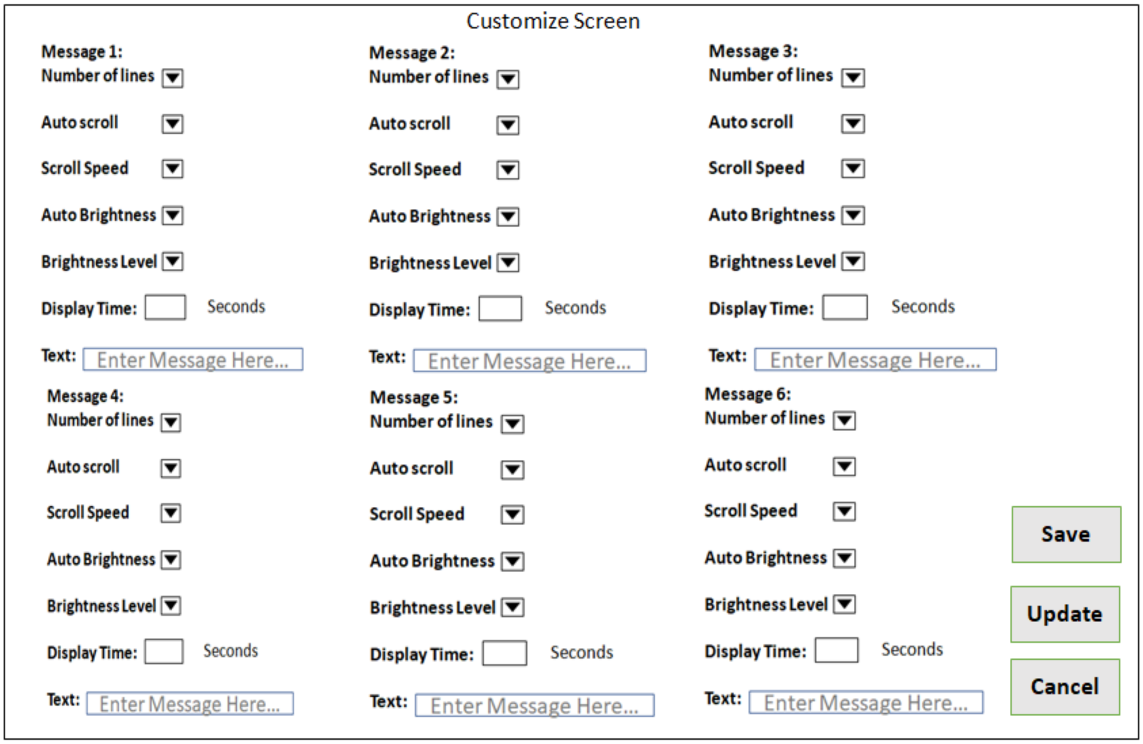


Figure 23: GUI Rough Model Design 2

The Customizable options that are presented for each message are Scrolling text enabled, brightness level, auto brightness, Single Line display, and Two-Line display. These options can be selected using drop down menus. The Scrolling text drop down menu and the auto brightness drop down menu have an on or off option. If the auto brightness option is on then the Brightness level option cannot be used. If the auto brightness option is off then the Brightness level drop down menu becomes active. The Brightness level drop down has options of intensity of brightness. The option available are 100%, 75%, 50% and 25% brightness with 100% being the brightest and 25% being the dimmest. The last drop-down menu has the option of choosing single line or two-line display for your message. If the single line option is selected, this will cause a character limit of twenty characters. If the two lines option is selected, the new character limit is increased to thirty characters. The Display Time section allows the user to determine how many seconds the message will display, if the user chooses to have a set time for display. If the user does not wish to have a set time for display, the user can leave this section blank. The drop-down menus described can be seen in figure 12 below.

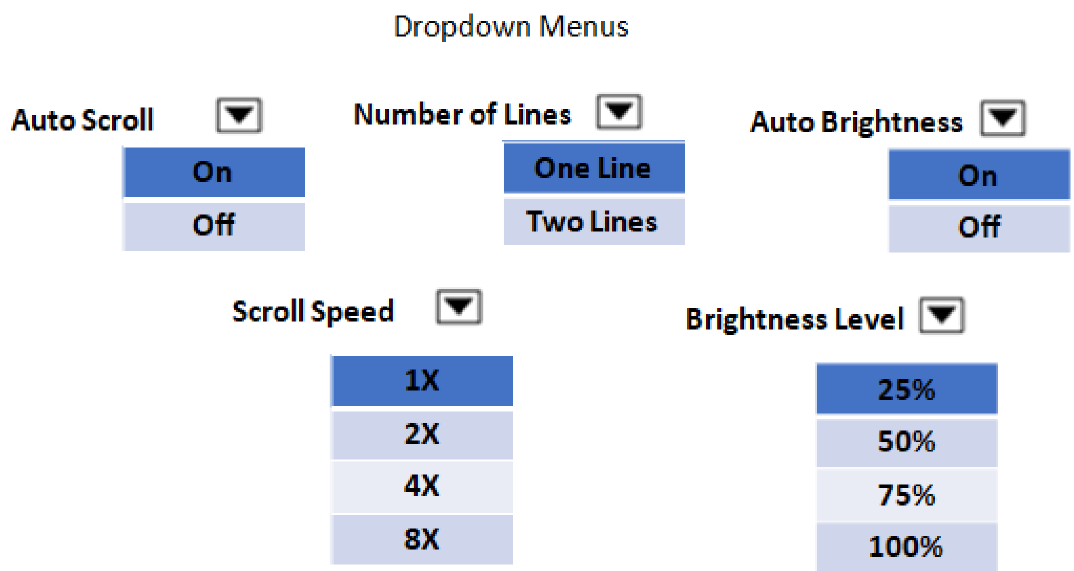


Figure 24: GUI Rough Model Design 2 Drop Down Menus

For this design, each textbox has a heading that tells the user which message number they are on. For example, Message 1 indicates that you are editing the message that will be displayed with the press of button 1. This is true for all of the rest of the messages along with their corresponding buttons.

Once all the changes that the user would like to make are made, the save button is pressed to save all of these changes done in the program. If the user is ready to have these changes apply to the connected device, the user presses the update button. When press this button, the device will now reflect all of the changes made by the user. Pressing the update, but will automatically press the save button as well that way the changes made to the device will stay saved to the program as well.

GUI design 3

A third design was constructed for the Graphical User Interface. The connection screen will not be used, and the customization screen will have a few changes from the second design. As seen in figure 23, the drop-down menus for single line display and auto scroll have been changed to checkboxes due to the fact that these features are either selected or not selected. If the checkbox for single line was checked, then line 2 for that message will gray out and the user will not be able to type inside the textbox, this is similar to design 2, when the option "enabled" was selected. If the checkbox for auto scroll was checked and the message was saved, then this will cause the message to scroll across the screen and loop when the button is pressed. There was a drop-down menu called "scroll speed" that was added below the checkbox for auto scroll. This drop-down menu has four different speed operations for how fast the auto scroller will operate.

Speed one is the slowest and speed four is the fastest Speed. The rates will be divided up evenly with speed one starting off at a rate of one LED movement per second. Movement speed setting two will have a movement speed of one and a quarter of LEDs per second while movement speed three will move at a rate of one and a half LEDs per second and finally movement speed four will move at a rate of one and a quarter LEDs per second. The brightness feature from both designs were removed due to the fact that the brightness will not be controlled through the GUI but rather controlled from the device itself by a combination of button presses. The display time is the same as design 2 by using a dropdown menu with a list of preset times the message will be illuminated on the display. There is also a preview button that has been added for each button. When this button is pressed the current message and current configuration will be displayed on the LED flip sign device for about thirty seconds. This is useful so that the person programming the sign understands how the message will look and act when the button is pressed. There were two more checkboxes added mirror and flash. These will set the messages to be mirrored and/or start flashing if selected in case the user wanted to preset the functionality.

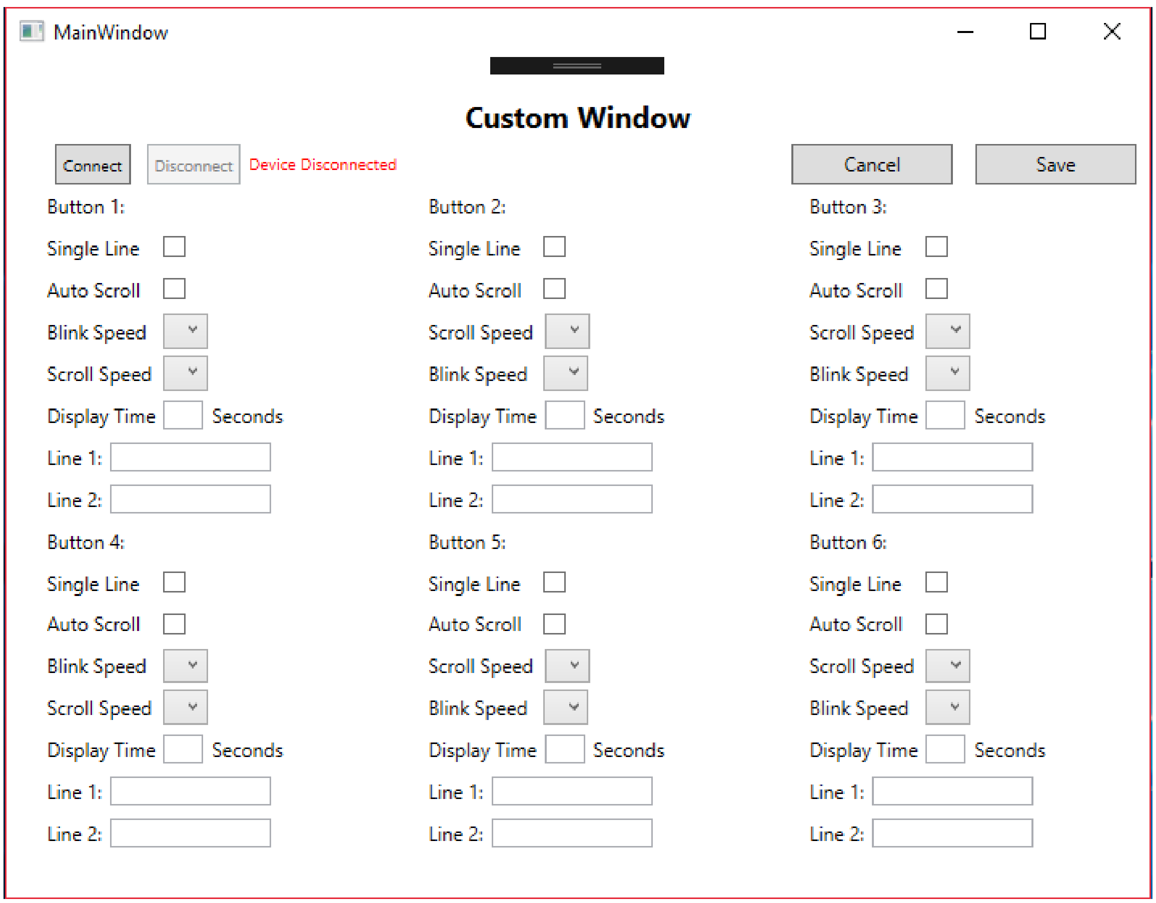


Figure 25:Custom Screen for GUI design 3

When the save button is pressed and the messages are transferred to the microcontroller successfully, then a pop up window will show up notifying the user that the messages were saved(shown in figure 21). If the connection was not successful, then there will be an error message(as shown in figure 22) that will pop up that notifies the user that something went wrong and to make sure the device is connected. On the popup window there is two options, try again and cancel. When the user presses try again the program will try to save the custom messages to the device again. If the user presses the cancel button, the user will be brought back to the custom message screen with the current changes that he or she made.

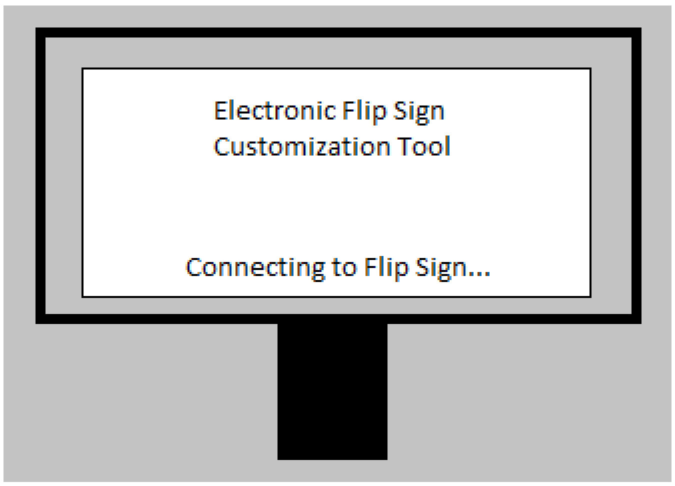


Figure 26: Connection Screen

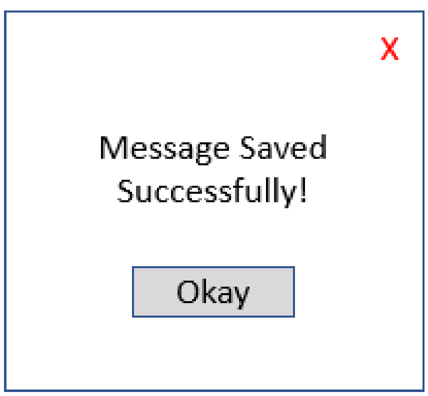


Figure 27:Saved Notification

Figure 28: Error Notification

GUI Design Mobile

This GUI design is used for the application that can be used on Android phones. Just like the other GUI designs, the first thing that is asked once the software is installed and launched is to attempt to connect to the Electronic Flip Sign. Once the sign is connected to the Android device, whether it is using Bluetooth or by hardwire, the app will then ask the user to which device to connect to. Once the user selects the device the app will open up to the customize screen as seen in the figure below.

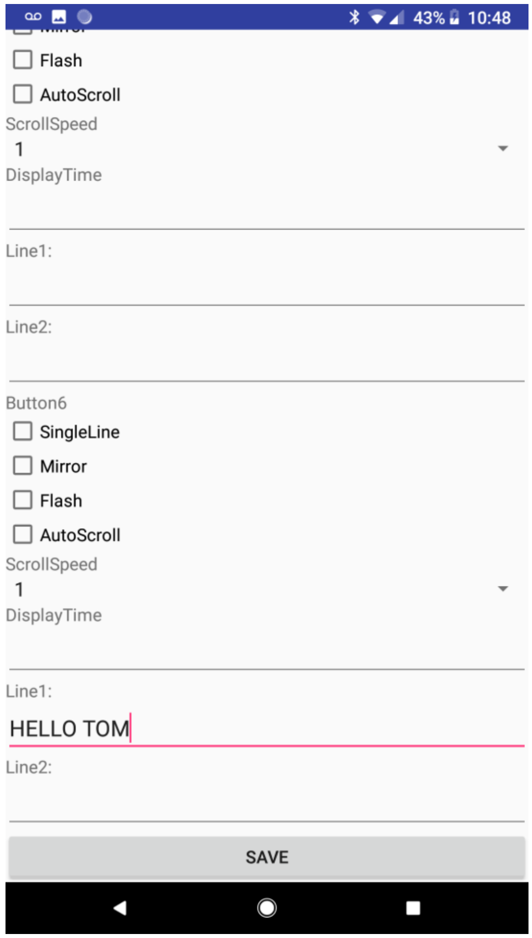


Figure 29: GUI mobile Design

The layout is similar to the computer application GUI. There are checkboxes to select single line, auto scrolling, flashing, and mirrored. There are dropdown menus available to select the scroll speed and brightness presets. There are textboxes for each line that the user can type into to store the messages they want. The buttons are stacked from top to bottom starting with button 1 available to button 6. Each button has their own controls to allow individual customization. At the bottom there is a save button which sends the data of each button to the device and saves the new configurations to the memory of the device.

GUI Functionality

The following flowchart shows the process in which the code goes through once the user is ready to create messages. This starts with the user opening the customization window. Once the window is open, the software creates a new grid for the user to work in. The Graphical User Interface then asks the user if they want to load saved messages. If yes, then the window opens with the loaded messages. If the user says no, then the software just opens a blank window. Once the user is in the customization window, the user selects the button they want their current message to be contained in. Now that they have created their message and placed it in the button section they want it in, the user can now add the features they want.

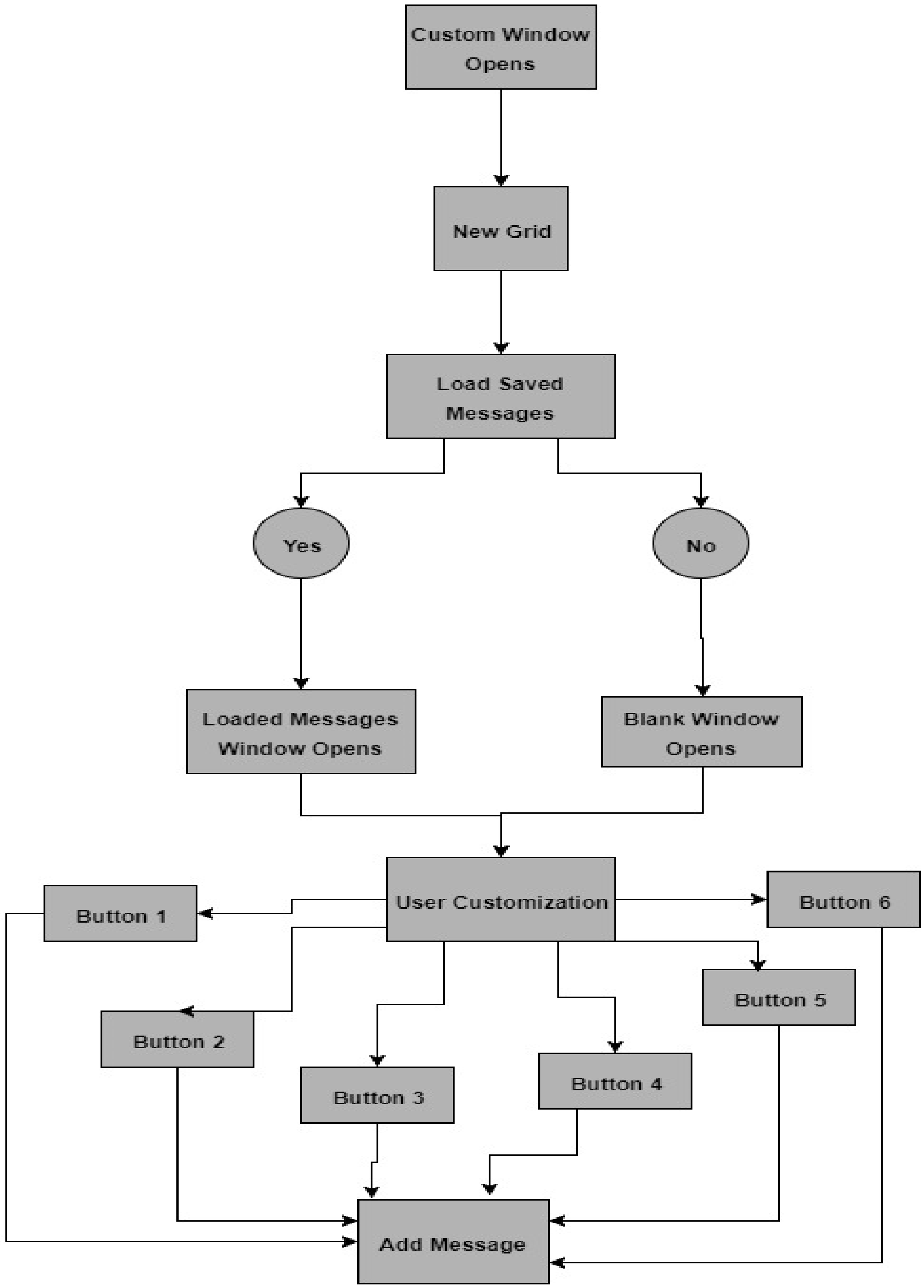


Figure 30: Message Creation Flowchart [20]

With the creation of a message and its placement in the proper message section the user wants it to be in, the message features can now be selected. This starts with the features window opening. The window has all the different features available to the user with checkboxes for weather this feature is on or off. Once the feature is toggled to its expected mode, the code goes back to the select features section if the user chooses too. If not, the message is created and the changes are saved.

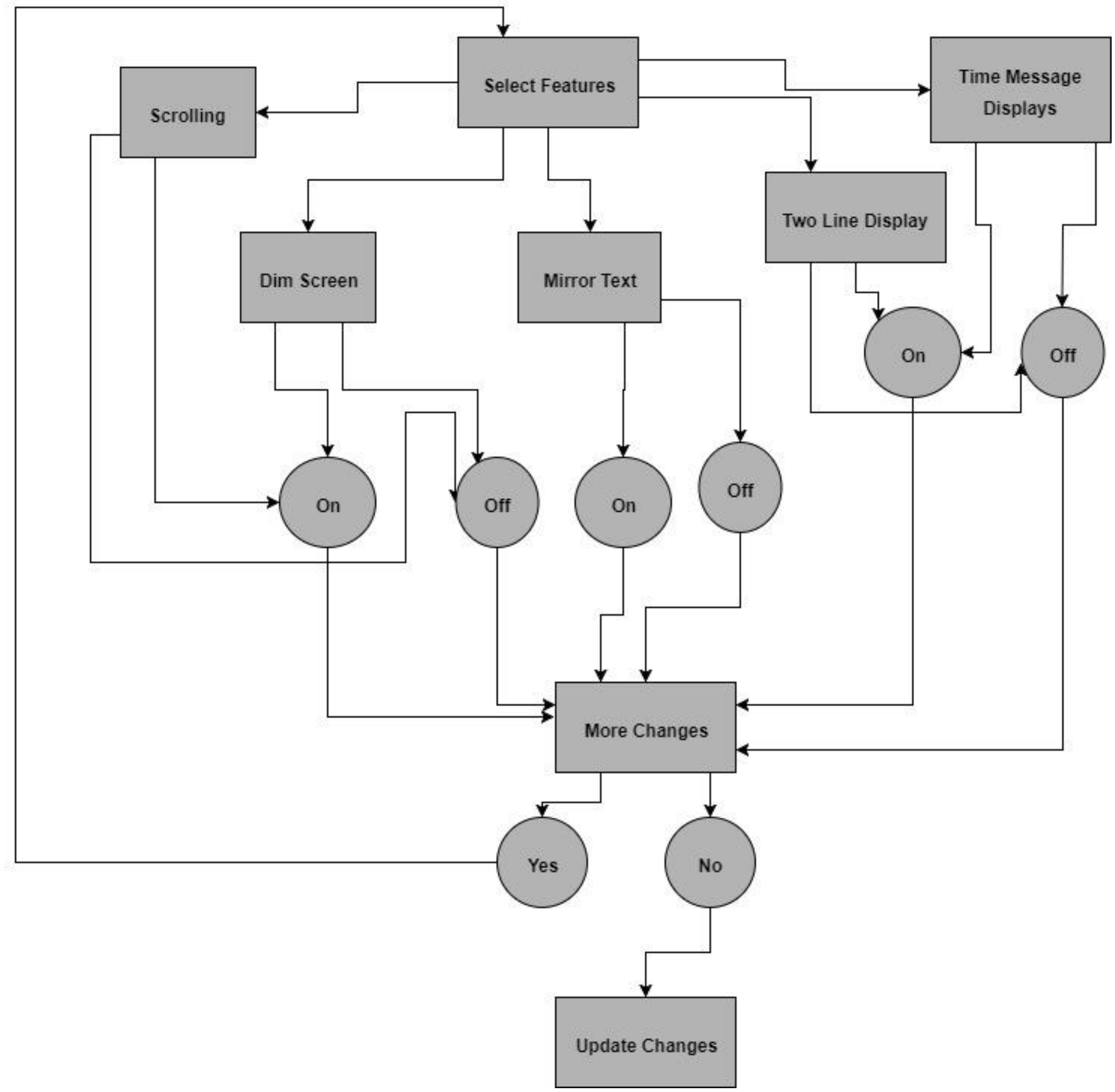


Figure 31: Feature Selection Flowchart [20]

8.9 Portable USB Charging Module

A major potent feature for the electronic flip sign design it will be able to be able to function on the go via mobile. In this case if the battery source dies out it would be extremely convenient for the user to recharge their power supply anywhere via USB. For instance, if you outside on the go the device will signal a red led to turn on indicate that the battery is low currently at a low voltage. Then you can connect a USB to the device into any suitable port that is universal to any charging outlet or other electronic devices such as computers. For this type of implementation to recharge the lithium ion battery a charging circuit module must be integrated with the design. For a more efficient model and cheaper advance solution that follows the same design specification is more preferred to use an integrated circuit component such as the TP4056. This circuit design, is mainly composed of two LEDS that reference if the device is in use or being fully charged. For the initial portion of the circuit a voltage regulator is required that takes any input higher order voltage that ranges from 7V to 20V and steps it down to 5V with 1A of current. To test performance to see if the charging is working similar to the TP4056 it is vital to include an LED and 120-330 Ohm resistor in-between the 5V and minus terminal of the battery. For the internal resistors on the board R32 and R65. These voltages main priority is to ensure that the proper current is supplied for charging. The rectifier diode is then added as a setback or for safe precaution just in case if the battery terminals are applied the in the opposite polarity. It also reduces the input voltage as well for a simpler step-down conversion, thus also reducing heat dissipated throughout the device. To also solve this problem of the circuit dissipating too much energy as heat. A heat sink will be applied to reduce the amount of temperature being absorbed through the circuit that possible damage minor components. Since this design is relatively low-cost due to the circuit elements the only factor to keep in mind is the energy dissipated throughout the circuit. The power loss inside the voltage regulator is too high thus less efficient. To obtain greater design using more advanced integrated circuits.

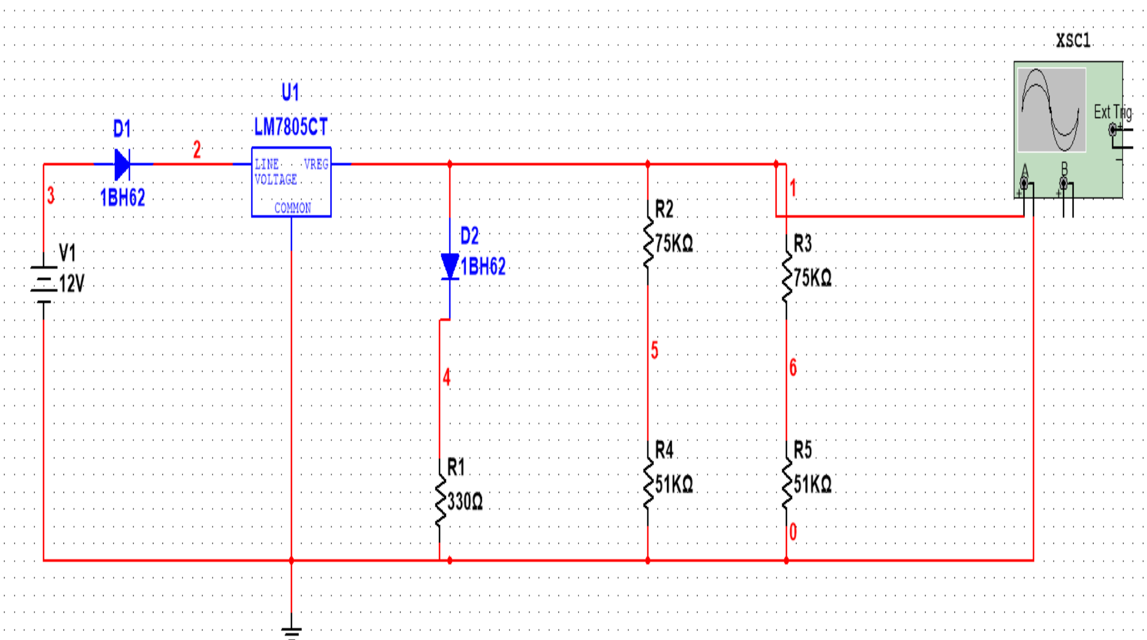


Figure 32: Charging circuit built in MultiSim

Advances in charging technology

As technology advances, there are more critical and efficient methods to simulate a linear charging circuit with more active components vs an inactive elements. The advantages due to a more advanced design are the size scaling of the circuit with replacing resistors with transistors. Another potent advantage would be the power dissipation. Instead of power being reduced into a wasted heat source do to the linear voltage regulator you can replace these components with integrated op amps to obtain a similar net result. For the ideal power supply, it is more optimal to apply an integrated circuit for such a case using the TP4056. It is complete constant current and constant voltage linear charger for a single cell lithium ion battery. This component can also work with USB and a wall adapter.

This design also does not include a diode the blocks the current for electric discharge since the design is foolproof. Which allows safe stable structure mainly due to the PMOS transistors integrated internally. Thermal feedback is also implemented with the circuit and regulates the charge current to limit the die temperature when the input power is a very high value. There is also an option to alter the input current by externally programming the resistor value. It does this by automatically terminates the incoming charge value drops to one tenth of the initial set programmed value after the final voltage value has been obtained.

Other unique characteristics include current monitoring, basically meaning the battery will automatically recharge if needed. There is also a status pin to indicate the cutoff charge rate and the presence of an input voltage. For the first pin which is the temperature sensor input, this sensory basically acts as a switch depending on the temperature the system will suspend the input voltage if the temperature is too high. To read this value there is a small thermistor. When the temperature changes so will the resistance. The second pin which is the constant charge current monitor pin. Its sole purpose is to measure charge current the charge current is set by connecting a resistor from that pin to common ground. Pin 3 is the ground pin. Pin 4 is the voltage supplied from the power supply source, when the voltages drops too low the circuit will automatically enter sleep mode. Pin 5 is the battery input pin. This pin supplies charge and regulates power to the internal circuit. Pin 6 Open drain output charge status. This voltage value will indicate when the battery charge termination or when the standby pin is pulled low since it is always at high impedance. Pin 7 Is the open drain output pin this lets you know that the battery is being charged the pin will be pulled low by an internal transistor otherwise the charge pin will be in high impedance. And the final pin8 will enable the input.

For a more accurate design using MCP73831. This advance design can automatically set itself to conserve power when the input voltage falls below the threshold voltage for the circuit. Over this feature prevents voltage to be drained out of the battery when the voltage supply is not remained. For fast charging the current can also be scaled by altering a resistor value. For quicker response to adapt to the system programming this resistor can be at quicker to fix than fully replacing a component already built in or soldered into the circuit. For the output for the P channel metal oxide semi field effect transistor (MOSFET). The MCP73831 maintains a constant voltage and constant current depending on the time by manipulating a mosfet biased in the linear region. For the internal components in the circuit to indicate what state the battery similar to the previous designs such as the TP4056 the MCP73831 has a multicolor led the changes color depending on the given voltage viewed from the battery. The three states are High, Low, and High Impedance. For main application, the charging circuit is used to power a microcontroller or a small programmable chip. Which will be powered by a lithium ion power battery. The main constraint to the charging circuit design are the cost and the thermal design. Which vary depending on the input voltage and the output current. When the battery charging circuit enters a constant current mode, maximum power dissipation occurs. The internal resistor value was selected into the design to alter the input current to have higher performance since the lithium ion battery can maintain a charge rate of 1C. Charging at this rate yield the shortest charge cycles without distorting the battery life. The purpose for external capacitors is to maintain good AC stability in the constant voltage mode. This capacitance also provides great compensation when an external battery source is not applied. These elements are also in the control feedback loop to act as bypass capacitance to compensate the battery nature that behaves as an inductor at high frequencies. There are also built protection circuit for unwanted discharge and potential short circuits thought the design. Multiple heat sinks are also used to reduce overall temperature.

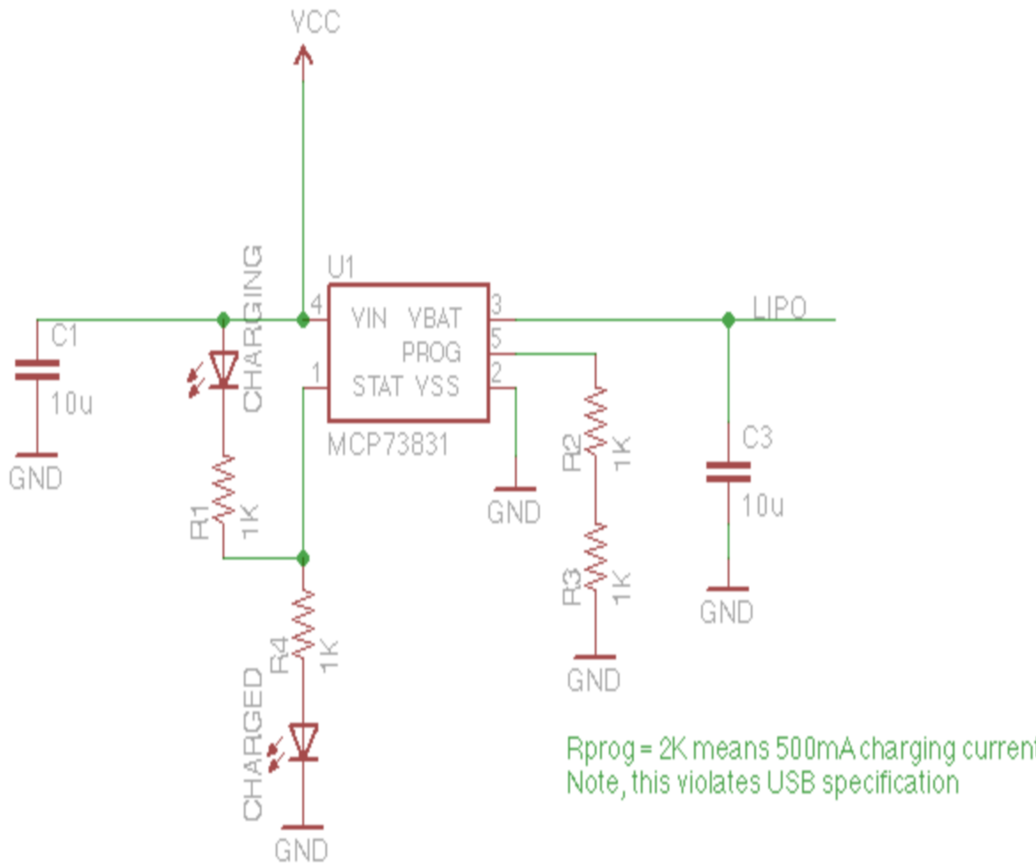


Figure 33: Charging Circuit Schematic[51]

9. Code Concepts

Code concepts are important and need to be explained in order to understand how each feature will work and be implemented. One of the requirements is being able to connect to the device wirelessly and this will be done with a Bluetooth chip that is integrated into the LED flip sign and code needs to be written to handle how the Bluetooth works. Another requirement is being able to plug in the device using a USB cable and communicating with a GUI through it There needs to be a concept of code that will describe how scrolling through the characters on the LED screen will work. The code for how a charging battery indicator that will show the user the current power level of the battery will take the readings of the battery and displayed them will need to write. The concept of how the letter characters will be stored and sent to the LED screen needs to be described. There needs to be an explanation of how the coding of brightness control will happen. There needs to be an explanation of how a message will show up on the LED sign as reversed text. The way the buttons will function needs to be explained along with any combinations of button presses that will be implemented. There needs to be code described for how the LED flip sign behaves when it first starts up. There needs to be an explanation for how the code for previewing messages from the GUI to the device work. The way the microcontroller behaves and handles inputs needs to have an explanation. Code for multiplexers and shift registers are needed for turning on and off the LEDs of the display panel on the flip sign.

9.1 Bluetooth Code

There is code that communicates with the Bluetooth chip on the LED flip sign. The code searches for a button combination, using a conditional statement, that turns on and off the module. When the device is connect to the android app the app will show the customization screen. This action will indicate that the display has been paired. There will be a loop of code that will be checking the connection. This will be used to make sure that the connection is still made. If the connection is dropped, then it will try to reconnect the flip sign to the most recent paired device every twenty seconds. This loop will also handle checking for any data transfer. If there are data packets being transferred over from the Bluetooth device that the LED flip sign is connected to, then the code will take the data and flash the microcontroller with the new custom-made messages that were created by the user from the computer or mobile device.

9.2 Launching program code

This next flowchart shows the flow in which the executable program goes on the surface. When the executable file is opened, it starts with checking for a connection to the device. If the connection is not found, a dialog box comes up with the option as to whether we should try again with connecting the device. If yes, check the connection again and if no, the program is closed. When the connection is found, the code moves onto checking for existing messages. If no existing messages are found, then the program opens the custom window with all the message sections blank and the feature options unchecked. The custom window will open with existing messages and features in the appropriate message positions. Now that the custom window is opened, the user can then edit their messages with other features available as well. Once the user is done making changes, they can save them. This is done by pressing the save button. The dialog box then opens asking if you are sure you want to save. The user hits save if they want to save and hits cancel if you want to cancel saving your changes. The custom window also has an exit option to exit the window. This causes a dialog option with the option of exit to confirm the exit or the cancel option to cancel the exit.

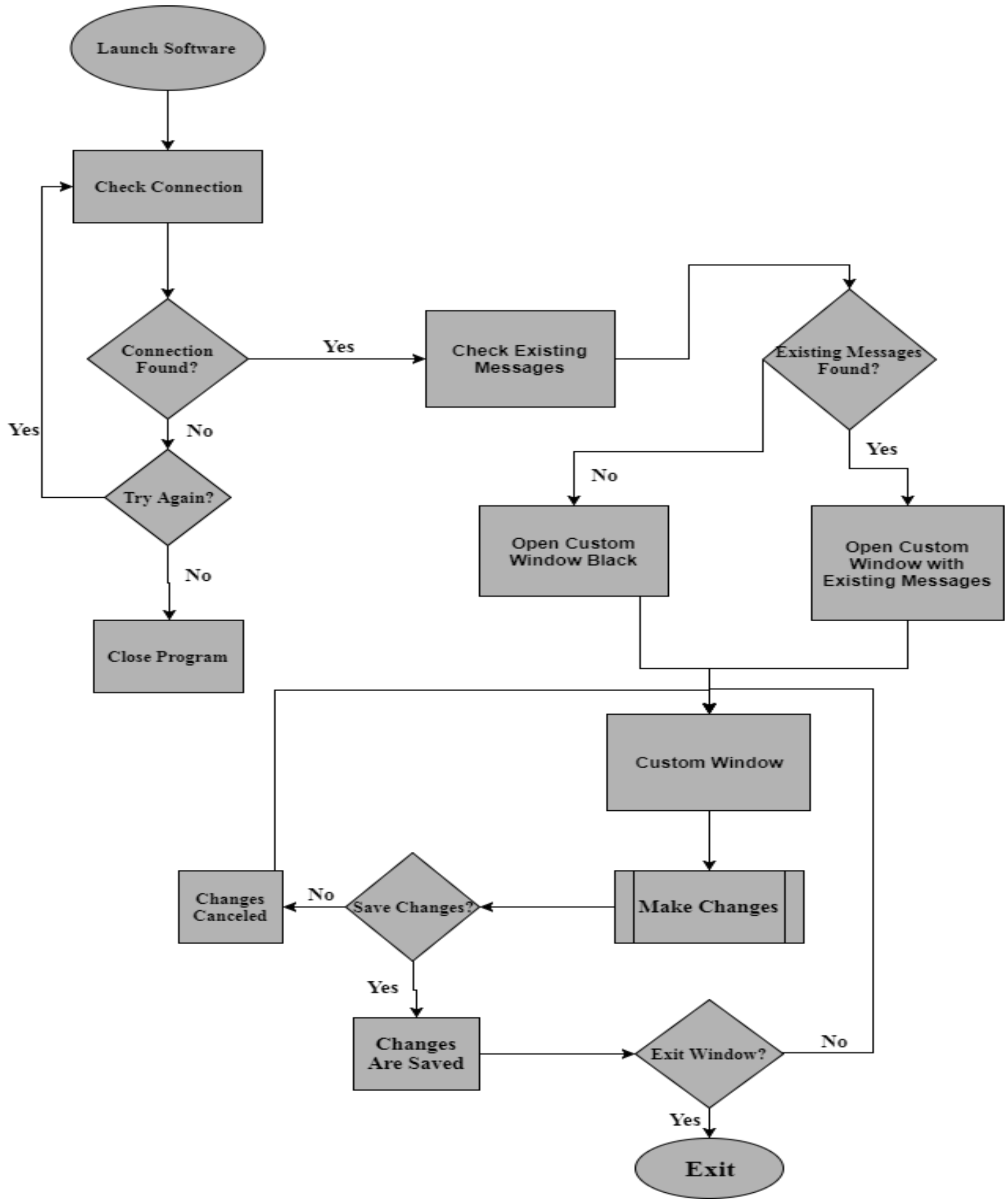


Figure 34: Launching Software Flowchart [20]

9.3 Auto launching program code

This code will handle making the program conveniently useable by bringing up a pop up message on the computer or mobile device and asking the user if he or she would like to run the program that will allow him or her to customize the LED flip sign when the user plugs connects the device to the computer or mobile device. If the answer is yes, then the code will close the pop up window and launch the program and the user-friendly GUI will show up. If the answer is a no, then the code will close the pop up window. This feature will no longer be used in the final design of the LED flip sign due to because the customer has changed the specification for this feature and would rather have the program be transferred from a CD to a computer, USB drive, or downloaded from the internet. This function ended up not using this process due to dynamic memory space.

9.4 Human Interface Device Code

The code that handles the human device interface checks to see if there is a connection from the USB port. If there is connection, the code on the microcontroller will send device information to the computer it is connected to. The receiving device will then display what is connected. The name of the connected device is displayed as “Electronic flip sign” in under the devices window.

9.5 Scrolling Text Code

This code operates the scrolling of the text on the display when enabled from the graphical user interface by the check box selection. This code loops through the message that the user is displaying on the sign and have the letters scroll across the screen of LEDs in a consistent motion. The movement of letters will be handled by shifting the display one column of LEDs at a time to the left at a specified rate, this rate is determined the option selected in the GUI. The rate will be passed in from the GUI in terms of speeds. There are four speeds and are represented as a one, two, three, or four as an integer data type. When the speed is passed in, that number will trigger a conditional statement that will set the speed of the LEDs. When the message reaches the last letter, the message will start over and repeat, starting with the first letter after five blank columns These actions will give a scrolling illusion to the display which allows for the user to create a message that is too long for the screen to normally display as a still message with the limited size of the screen.

9.6 Charging Battery Indicator Code

The code for charging battery will be operated from the microcontroller. The device will check to see if there is a USB connection. If there is a USB connection, then the LED flip sign will show a blinking battery symbol that will show the current charge. The Battery symbol will be hollow and have four segments in it and split the charging indicator into fourths. The battery will fill the segments from bottom to top. While plugged in and charging the current segment of the battery that is being charged will flash once and if a segment is done charging than that corresponding bar will illuminate and stay lit. If the battery has a charge less than twenty five percent then the battery will be completely hollow with a blinking first segment. If the battery has a charge level between twenty five percent and fifty percent, then the battery will have one lit bar showing that one fourth of the battery is filled. If the battery is between fifty percent charge and seventy five percent charge, then the battery will have two bars of led lights full. If the battery has a greater than or equal to seventy five percent and has a less than one hundred percent charge, then three bars of the battery will be illuminated. Once the battery has been completely full the battery symbol will have all of the segments filled and turn off after a few seconds. This function was not implemented due to hardware constraints.

9.7 Light Sensor Code

The code that handles the light sensor. When auto brightness is turned on this sensor will be used. While the LED flip sign is on, the program will check to see if auto brightness is on. If auto brightness is on then the code will check the data that the sensor is giving off. Depending on the readings that the sensor gives off, the code will reduce the brightness or increase the brightness of the LED display to accommodate for the intensity the light that is being shined on the LED display. The range of the sensor will be found by placing the device in really bright environments and dark environments. This will be used to create conditional statements that can compare values with the sensor to adjust the brightness accordingly. If the light is really bright against the sign, the LEDs will automatically to be turned on with a higher brightness while if the light source that is hitting the light sensor has a low intensity then the LEDs don't need to be on as bright and the code will reduce the amount of power going to the LEDs. By reducing the power, the LEDs will become dimmer. There will be a set minimum and maximum LED brightness that the sensor can adjust to. These values will be found through testing and observing the LEDs in bright and dark environments and adjusting the intensity of the LEDs until a brightness is found that is satisfied.

9.8 Message Display Code

This code will take in the message that was stored into a string that was passed from the graphical user interface and display the message on the LED panel. This will be done with by converting the string to a set of byte values. Once we have these values, the program can then convert the values to be able to be displayed on the LED matrix. One major method to displaying messages on a LED dot matrix is to scan the columns to get the correct values for the rows [38]. This concept can be seen in figure 17 below. The data is pushed to the shift registers using SPI.transfer. While the data is being pushed, the decade counter needs to scan through all the rows to illuminate the message. The decade counter needs the clock pin to be cycled to move row by row.

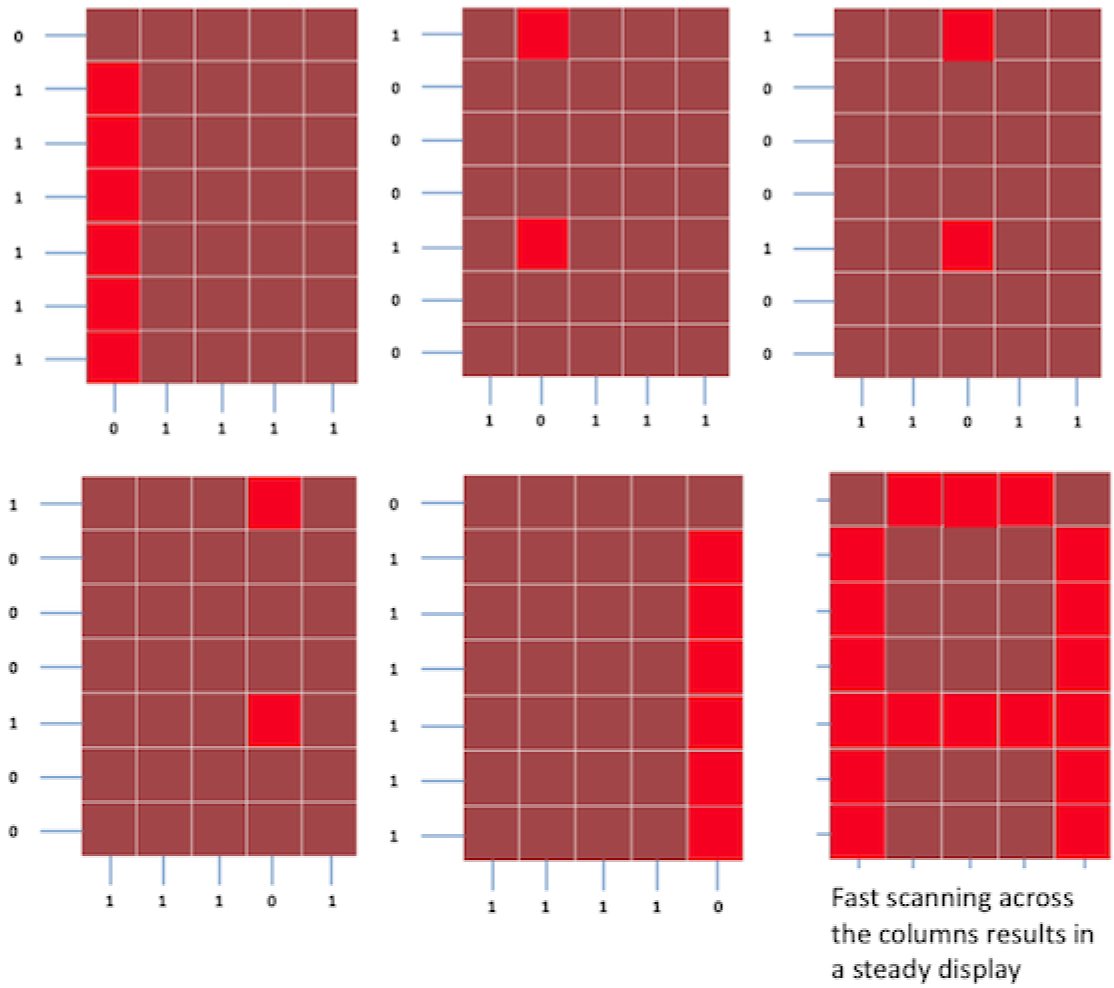


Figure 35: Scanning Method for LED Display [38]

9.9 Dim Control Code

The code that handles the brightness control will run when a certain combination of buttons are pressed. While the sign is on, the program will check to see how long a button is pressed. If the first button (top left) is pressed and held down for three seconds or more then the code will enter a dimness function. This function will check for the status of three buttons. If button 4 is pressed (bottom left) the brightness will be decreased and the auto brightness will be turned off by setting the auto brightness variable (bool type) to false. If button 5 (middle button) is pressed then auto brightness will be set to true and the brightness will be controlled by the ambient light sensor. If button 6 is pressed then the brightness will increase and auto brightness will be turned off.

9.10 Reverse Text Code

The reverse text function will be handled in the microcontroller. This event will run when the option is selected by the user in the Graphical User Interface. This function takes in the message that the user has typed into the textbox in the GUI and will replace the message's letters with mirrored letters. The function holding this code will hold an array with the same order as the character array but will have byte values that will illuminate mirrored letters on the led display. Once the letters have been switched, then the new mirrored message will be returned from the function to the main loop of code.

9.11 Button Functions Code

This code handles input from the buttons. When the user presses a button, the code will trigger certain commands depending on the button or combination of buttons. There will be conditional statements handling these actions. When a conditional statement is true, the code for that statement will be executed and the corresponding action will be taken. Once the action is completed the code will continue to run and loop. If there are no buttons being pressed the code will loop for five seconds. If there are no buttons that were pressed within those five seconds, then the microcontroller will go into a low power state to save power and wait for an interrupt to power back on. If the first button is being pressed, then the conditional statement will bring the code to a section that will send the message that was assigned to button one to be sent to the LED display to be displayed. If the second button was pressed, then a conditional statement will take the code to a route that will send message two to be displayed. This will be the same for buttons three, four, five, and six. There will be a conditional statement that looks for how long a button is pressed and if a button is pressed for more than three seconds then the message that is displayed from that button will flash. There will be a conditional that looks for a combination of buttons like holding down buttons and entering dim control.

The process in which the code implemented in the device goes through during the operation of the Electronic Flip Sign. When the user is operating the device, the user starts with powering on the device. If the power is off, then the code doesn't continue to loop till the power is turned on for the device. Once the device has been turned on, the code moves on to checking if any of the six buttons are being pressed. If none of them are being pressed, the code waits a set amount of time before moving on to turning the power back off on the device. If the set amount of time has not been passed then the code follows back to checking which buttons are being pressed. The purpose of this portion is to continue to display a message based on the amount of preset time give in the software when customizing the messages.

When a button is pressed the code moves to whichever button has been pressed and displays its respective message along with the features given to this message by the user. There is also a side feature that whenever Button 6 is held down for 2 seconds, it will trigger the brightness setting feature. Once Button 6 is held down for 2 seconds, the code moves onto checking whether button 4, 5, or 6 have been pressed during this brightness feature phase. Whenever button 4 is pressed during this feature, the messages brightness becomes lower. If button 5 is pressed, the brightness is set to auto. And finally, when button 6 is pressed during this phase, the brightness is increased. Once the proper message is displayed and the brightness setting is set to the desired specifications, the code loops back around to checking the buttons pressed.

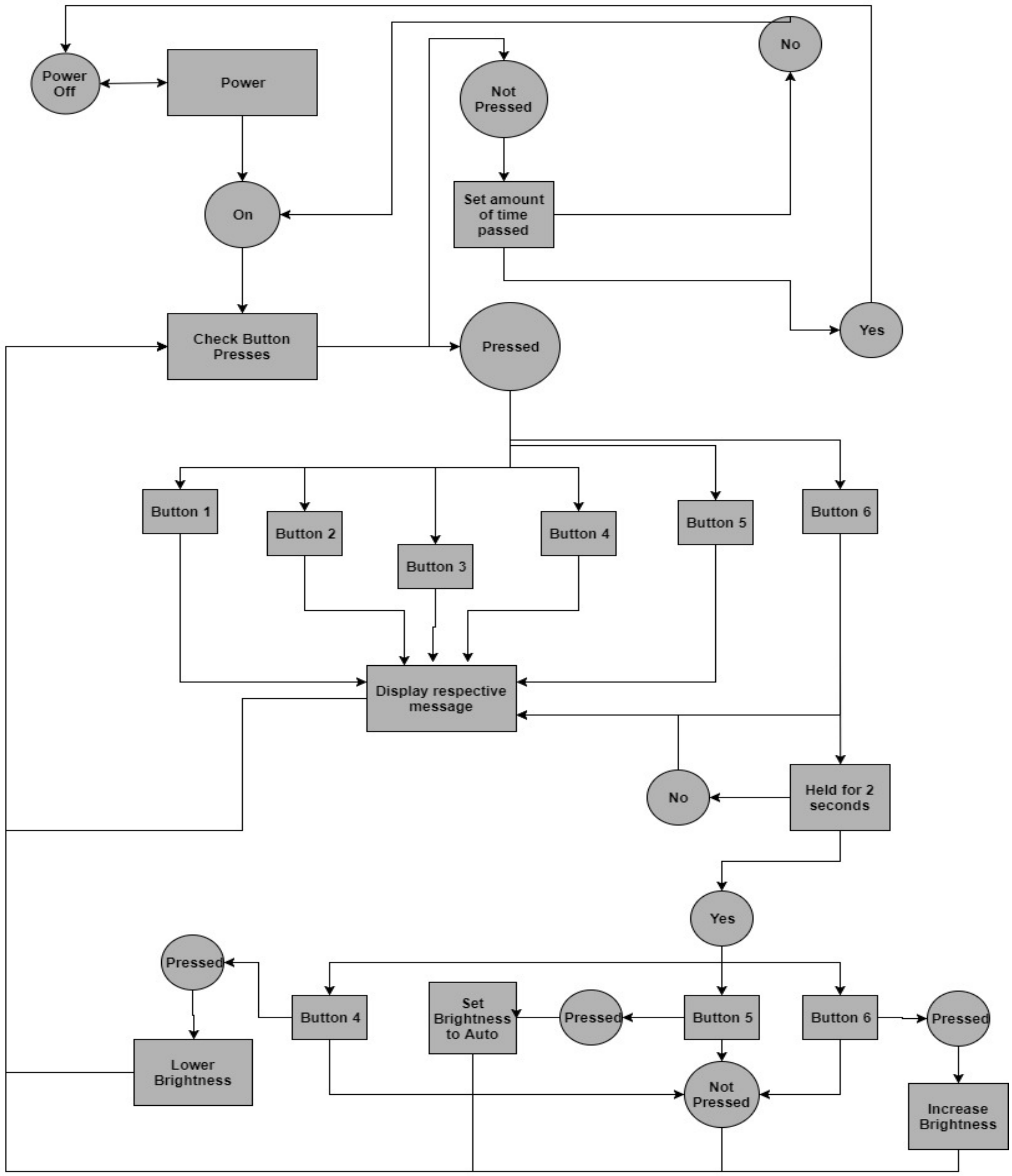


Figure 36: While LED Flip Sign is in Use Flowchart [20]

9.12 Code on Microcontroller

When the device is powered on, the microcontroller will start to run the program for a few seconds. Then, after a few moments of inactivity, the microcontroller enters into sleep mode. This will then lead to the low power state and will then wait for an interrupt to happen in order to wake up. This will cause the power save and extend the battery life of the device.

9.13 Character Models Code

This code uses arrays that hold byte values. These byte values power the LEDs in a certain pattern to resemble the character or symbol that will be displayed. The first index of the array will resemble the letter A. After the letter A, letter B will be the next index in the array and so on until Z. All of these letters will be capitalized. After the alphabet, numbers zero through nine will fill the next indices in the array. After the numbers will be a byte value that will represent a space. After a space, there will be an exclamation point then at symbol then hash tag then dollar sign then carrot then and symbol then star then left and right parenthesis then dash then plus and equal sign. After the symbols, emojis will be added. A smiley face, sad face, and winky face will be added.

Controlling Multiplexer using code

There needs to be code handling the multiplexer that powers each row for the LEDs. In order to control a multiplexer, there needs to be input/output pins connected from the microcontroller to the input terminals of the multiplexer. The program should set all the pins connected to the multiplexer to output because all the signals are being sent to the multiplexer and the microcontroller is not receiving any input. The enable pin needs to be set to on for each multiplexer in order to have the LEDs turn on. When no LED rows are active the enable pin can be set to one. When a row is going to be used, the enable pin will be set to zero[46]. There is a bool array that will hold as many values as there are selector registers. A bool array is used rather than a integer array because a bool array is more memory efficient and only ones and zeros are going to be used for the select registers. The array will hold the sequence of bits that will be used for the current item being displayed and pass each value to each selector register accordingly[46]. This turns on and off each row at the time of execution. This code was ultimately discarded due to the decision of using shift registers and decade counters instead.

Controlling Shift registers using code

There needed to be code to handle the shift registers that control each column of the LED matrix. There needs to be data sent to the data pin so that the registers have data that can be displayed. The latch pin needs to be set to ground in order to keep the LEDs at the current state. There is a function that passes in which LEDs in the particular column need to be turned on and off according to the layout of the letters that are being displayed to the data pin[32]. Then the latch pin will be set too high to light the LEDs based on the data that was passed to the data pin[32].

Code for Turning on

When the LED flip sign is turned on this code runs once during first start up. This code gives a signal to the user that the sign has been turned on and is ready to be used. The LED flip sign's display will flash the message "HELLO!". There is a pre-written string that will stay constant holding this message and display for two seconds using a loop. Once this action is done the code will go to the main loop of the program.

The flowchart below, in Figure 37 is to show the flow in which the code runs through the various functions when the user plugs in the device to either the wall or your local computer. Once the user plugs in the device using the Micro USB Cable, it starts with checking whether the device is plugged into a computer or in a wall outlet. If the device discovers that it is just plugged into the wall, it will just display the charge level until the battery is full. If the device discovers it is plugged into the PC, the code moves into finding whether the software is installed on this computer. If the software is found, the code moves onto prompting to run the exe file. If the software is not found, the computer is then prompted to install the software and then moves onto the prompt to run the program. If the user decides to run the program, the exe program begins to run. If not, the dialog closes and the device continues to charge.

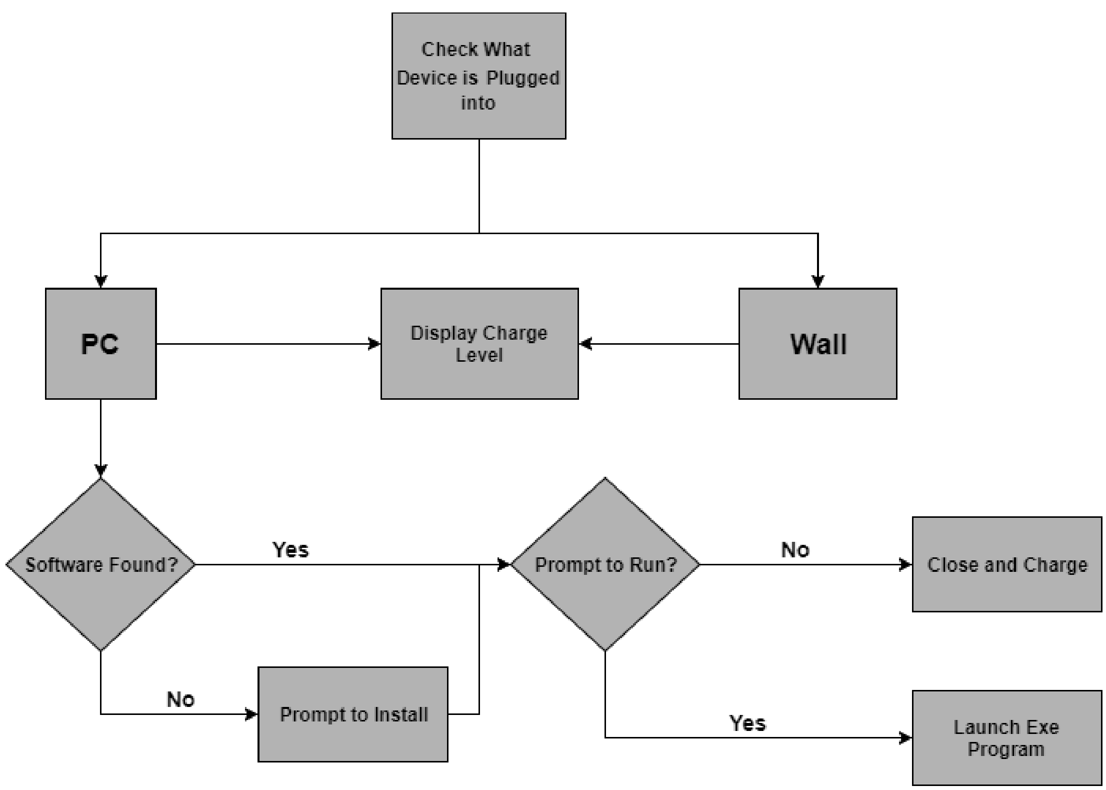


Figure 37: Plug In Flowchart [20]

Code for Previewing messages

There is code that will handle showing the user the current changes made to a button's message that the user made on the LED flip sign. While the LED flip sign is connected to a device that is using the GUI to customize messages, there will be a button that will send the newly configured message to a function on the microcontroller. When the microcontroller's main loop detects a connection from a device it will look for a signal from this button being pressed. This signal will be a bool and if true, then the button has been pressed and the code will send that message to the LED drivers and keeps that message displayed until another preview button is pressed or the timer of 30 seconds is up, whichever comes first. If the bool is false, then the button hasn't been pressed. Some of the features for this function's design ended up being removed due to memory constraints.

9.14 Programming Languages Available for Use

The C programming language is a wildly popular language that has many different uses. One of C programming's most useful features is that it can be used when programming various kinds of embedded systems. This is very helpful when programming the microcontroller. The microcontroller that is used in this project will be programmed in the C programming language. The Arduino integrated development environment and AVR studio both utilize C to program microcontrollers.

The programming language C# allows developers to create applications that run on windows. [4] C# can be used to create a physical executable file that can be launched from the computer to program the device. The code will search for the device using COM ports and device names. C# has a simpler form of syntax compared to other languages such as C++ [4]. C# is an object-oriented programming language, which allows the programmer to create objects as instances of classes. For this project, C# is being considered for creating the Graphical User Interface. This can be done by using Visual Studio C#. C# also supports the use of language integrated queries [17]. This can be used to create a basic SQL database that can store the messages that the use creates using our software, along with the options of features they choose to select to go along with each message.

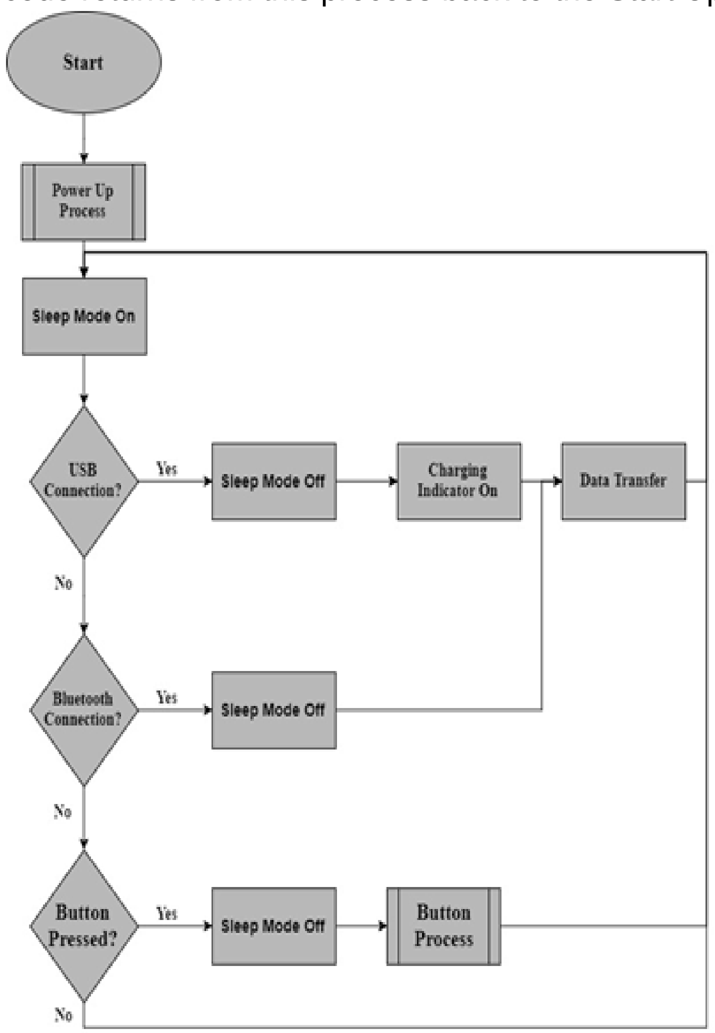
The programming language HTML could be used for designing the Graphical User Interface. By using HTML, the LED flip sign can be customized by using a website, so that a computer, Android, or iOS device can connect to. This is created using HTML, CSS, and JavaScript. We also have the idea of creating a website to go along with our software, so that users have a way to access the installation for our software. This way, users have another method for installing the software in case the installation could not be found through just plugging in the device to their local computer.

The other possibility for a language of coding that could be needed for this project is assembly language. When using the Atmel Studio 7 environment for programming microcontrollers, assembly is supported. This will allow the programmers to be able to directly manipulate the registers for more efficient processes for the microcontroller.

9.15 Microcontroller Function

The Start Process, as seen in Figure 38 below, shows the flow in which the code goes through when device turns on. This structure allows the device to be able to function in low power mode. Being in low power mode conserves the batteries energy while it is waiting for its next task. The code starts with the Power Up Process, which shows the procedures that the device goes through while the device is turning on. Once the device is completely powered on, the code enters a loop. This loop keeps the device powered on while allowing it to go into sleep mode when it is not in use and come out of sleep mode when it is being used again. This starts by checking if the device is connected to a USB cable. If it is, then sleep mode turns off. Then the charging indicator displays, the device checks for the software to be able to transfer data between and the code loops back to turning sleep mode back on. If a USB connection is not found, then the code moves on to checking whether a Bluetooth connection is found. If it is found then, the sleep mode is turned off and the device checks for the software to transfer data, just in the USB connection. The Bluetooth connection also loops back and turns sleep mode back on. If the Bluetooth connection is not found, then the device checks for a button pressed on the sign. If a button is pressed, then the device turns off sleep mode and moves into the Button Process. This process tells the device what functions need to be called based on which button is pressed, how long it is pressed, and based on how many times it is pressed. Once the Button Process is completed, the code loops back to turning on sleep mode. This process never ends until the device is fully turned off and no power is being used to keep it low power mode. Whenever the device stays in sleep mode, the code is constantly running and waiting for an action is taken to allow it to turn off sleep mode to perform the action.

The Power Up Process, as seen in Figure 39, is used to tell the Electronic Flip Sign it needs to do whenever it is turning on. This is only needed to be done once per time the device is to be turned. This is why this process is not a part of the loop found in the Startup Process. The first step is to have it set the time's value to zero, so that it is at its starting value. The next step is to store the message "Welcome" in the message variable. Once this initial message is stored, the message is then displayed on the LED Matrix to let the user know that the device is turning on. The welcome message stays displayed for three seconds. This is done by using the timer to tell when three seconds has happened. If time is still less than three seconds, the code loops and continues to check the time until it becomes three seconds. When the time is no longer less than three seconds, the code than moves into turning off all the LEDs, so that the sign is ready to start displaying the user's programmed messages. Now that the LEDs have been turned off after the device's initial powering on procedures, the code returns from this process back to the Start Up Procedure.



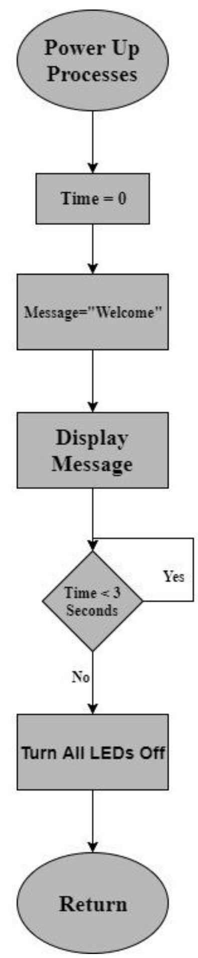


Figure 38: Start Up Process Flowchart [20]

Figure 39: Power Up Process Flowchart [20]

The next process and the most complex in this section is the Button Process as seen in Figure 40. This process tells the Electronic Flip Sign what it needs to accomplish whenever a specific button is pressed. This means that the device can also distinguish between a button press, a button being held down for a period of time and being pressed more than once. The code starts by checking if button one has been pressed. If the button has been pressed, then the code checks to see if the button is continued to be held down for a few seconds. If this is the case for button one, then the activation of the Bluetooth device is toggled on or off depending on the mode it was previously in. If the button was not held down for a few seconds, then code checks to see if button one is pressed again during a short window of time after the first press. If this does happen, then the message stored in message one will display in reverse. If the button is not pressed a second time, then message one is just displayed normally. Once the code is done with checking button one's presses, the code loops back to checking the buttons. If button one is not originally pressed, then the code checks button two for a button press. If button two has been pressed the code just goes through the same process as button one for the button being pressed again in a short period of time. If this does happen then the message is reversed and then displayed. If this does not happen, then message two is just displayed and then the code loops back to check for the next button press. If button one and button two are not pressed then button three is checked. When Button three is pressed, the code goes through the exact same process as button one, but if button three is held down for a few seconds, the Brightness Process becomes active. Once this process is resolved, the code loops back to the button checking. If button four is pressed, then the same code flow that button two goes through happens. This means that button four's message is reversed if the button is pressed again and displays as normal if it is not pressed again. Button five and Button six also goes through the same reverse text code as button four does. If none of the buttons are pressed after all of them have been checked, then the code flow returns back to the Start Up Process code and the device goes back into sleep mode until the devices next function is to be made.

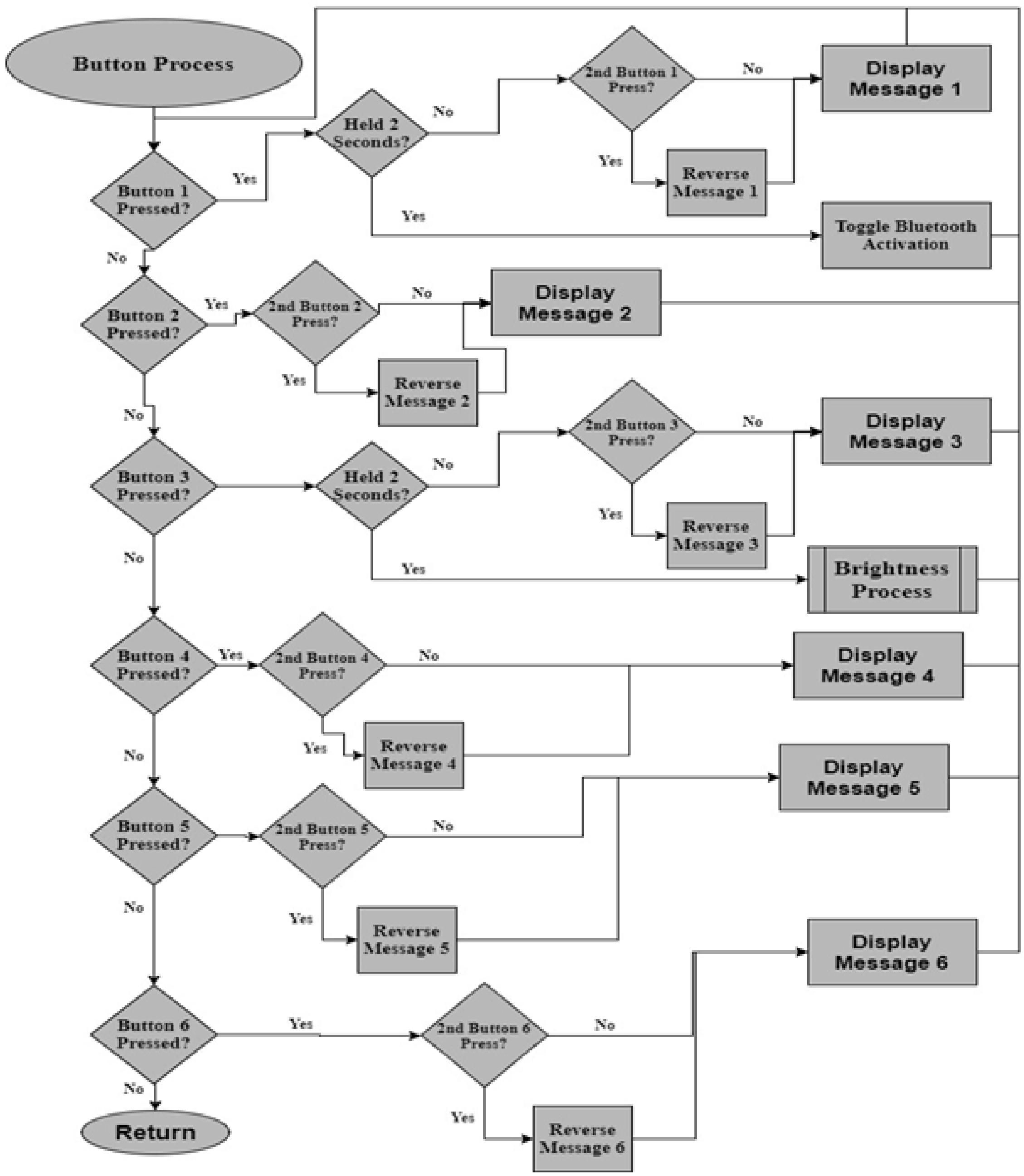


Figure 40: Button Process Flowchart [20]

As seen in the Button Process Flowchart, button three can be used to activate the brightness process. The function flowchart for the Brightness Process can be found in Figure 41 below. This process shows how the device functions the brightness on the LED Matrix. The process starts with checking if button four is pressed during the brightness mode. If it is pressed, the brightness of the LEDs is decreased and then the code loops back to the button checking section. If button four is not pressed then the code moves onto checking if button five is pressed. When button five is pressed, then the auto brightness mode is toggled on or off depending on whether it had been active previously or not. When the auto brightness is toggled on, the code moved to the Auto Brightness Process before looping back to the button checking. Otherwise, the Auto Brightness Process is skipped over. If the code finds that button four or button five are not pressed, the button six is checked to see if it has been pressed. If it has, the brightness of the LEDs increases. If not then the code checks if button three has been pressed. When button three is pressed, then this process ends and moves back to the Button Process. If none of the buttons are pressed, the device stays in the brightness mode until button three is pressed to close the brightness mode. Buttons being held down was changed to two buttons pressed at the same time for user friendly use.

Figure 42 shows the flowchart for the Auto Brightness Flowchart. If the sensor comes back with a low light, then the brightness is increased. If the sensor detects a medium light level, then the brightness is set to 50% of its max brightness. If neither of these options are found to be true, then the brightness is set to a low level.

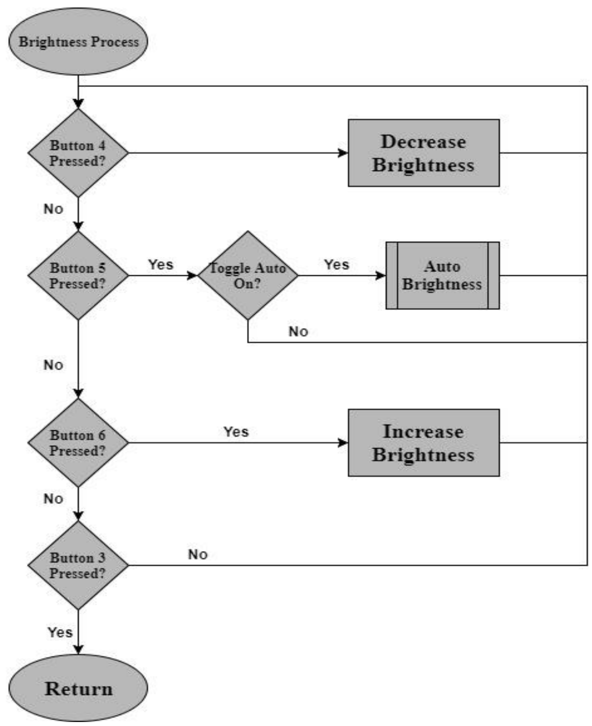


Figure 41: Brightness Process Flowchart

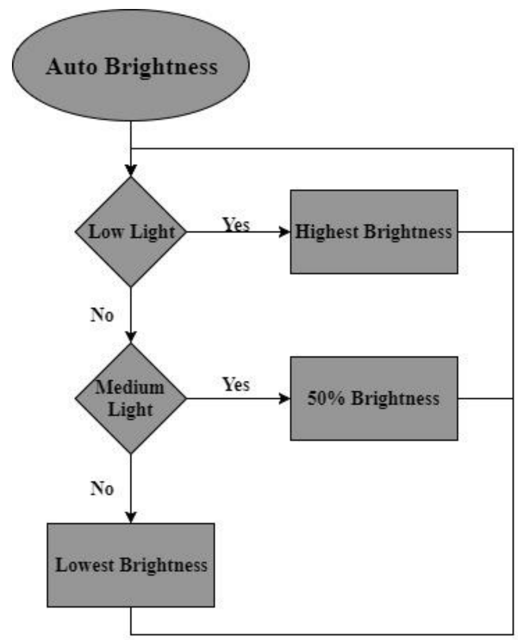


Figure 42: Auto Brightness Flowchart

10. Prototyping

This section on prototyping will go over each of the prototypes that have been made, what they were built to test and how they were used and what the desired outcome it to be. This is important because, trial in error is essential for perfecting a project by tweaking out bugs, repairing structural problems, and fixing design flaws.

10.1 Display Prototype

The first prototype is a simple functioning device that can perform the basics as a proof of concept. This prototype display is made of a perfboard that and 8x8 square of LEDs is attached manually. The first attempt at this display the LEDs were soldered onto two holes of the perfboard and then it was attempted to be wired on the back this caused problems as the wiring diagram requires that the cathodes are wired on the columns and anodes are wired on the rows making each LED wiring cross and there was some issue where the connections would cross. The second prototype display was improved upon the first design to make it easier to wire and solder. The new design had the LEDs wired diagonally across 2 the comparison can be seen in the 2 images below. Soldering the LEDs diagonally improved the ease of soldering the LEDs giving more room to get the solder iron there and this also made wiring the LEDs easier. Since the LEDs are now diagonal the anodes line up with anodes and cathodes with cathodes in their own rows, they still cross but they now have their own rows this makes the wiring diagram easier to understand and set up. The second prototype was wired before soldering the LEDs and this was done by threading a wire through the hole in the perfboard on the top along the rows and on the columns on the bottom this made it so the wires could cross without causing errors and this let the prototype be wired similarly to the way that it will be wired on a 2-layer PCB. Once this new display is soldered each LED was tested by getting a source with the correct current and voltage to touch to each row and column combination possible to test that each of the 64 LEDs not only turn on but also that they do not turn on multiple LEDs so each of the row and column combinations should turn on a single LED and each of the LEDs should show the same brightness and be as bright as if tested alone.

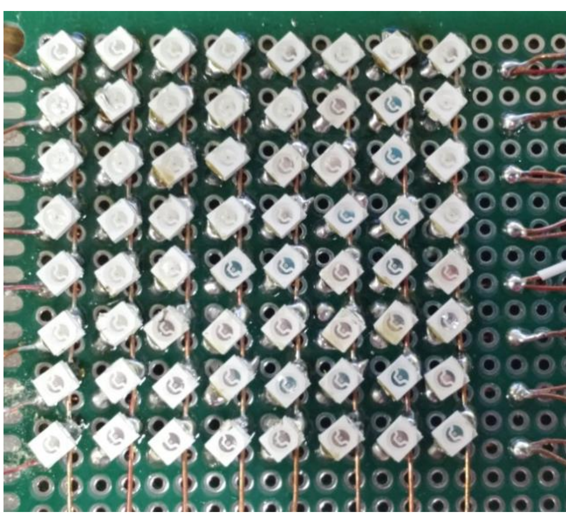


Figure 43: Prototype display V.2 front

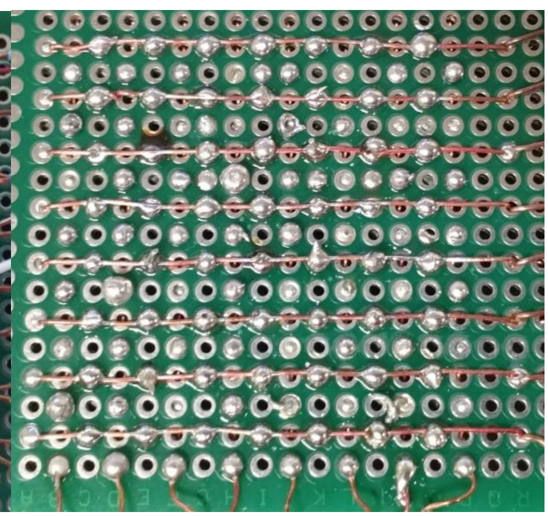


Figure 44: Prototype display V.2 back

The new display to be fully tested and used must have something to run it as a display. The first prototype LED drive was an Arduino prebuilt LED display that has removable display segments as seen in the figure below. The display is operated off an Arduino board that is getting power from a USB port. This serves 2 purposes first the provided LED display is removed and the prototype LED display is added and now sample code can be run to see the display functioning correctly. The second use for this board is to allow the coders to implement code from sending data to and from a GUI and to testing display code. It is of note that this board used a MAX7219 and the way that this is programmed is different than that of shift registers and multiplexers. The second prototype for the LED drivers is to take the chips that are being used and wiring them into a breadboard but since the parts that we are using are surface mount devices so they cannot be directly added to a breadboard so a breakout board is used for these chips this is a small board that has smd pads and then wires out to pins that can be used in a breadboard. This new circuit is wired on the breadboard with the breakout chips and then for the resistors simple parts such as resistors and any capacitors are represented using regular size components even though they will be smd components in the final device. With the wiring completed the prototype is wired into the circuit and then the Arduino board is used to run the processor since it runs the same processor. The new prototype device will be run off a battery rather than a USB and power from the Arduino. This all together is a functioning prototype that can run and function just as the final device for a single 8x8 section of LEDs. There are a few things this cannot test such as checking the data being sent from chip to chip as it will be in a full display as well as seeing the full size of the display as to see the size of the text and brightness but this will be enough to get a good bearing on the project.

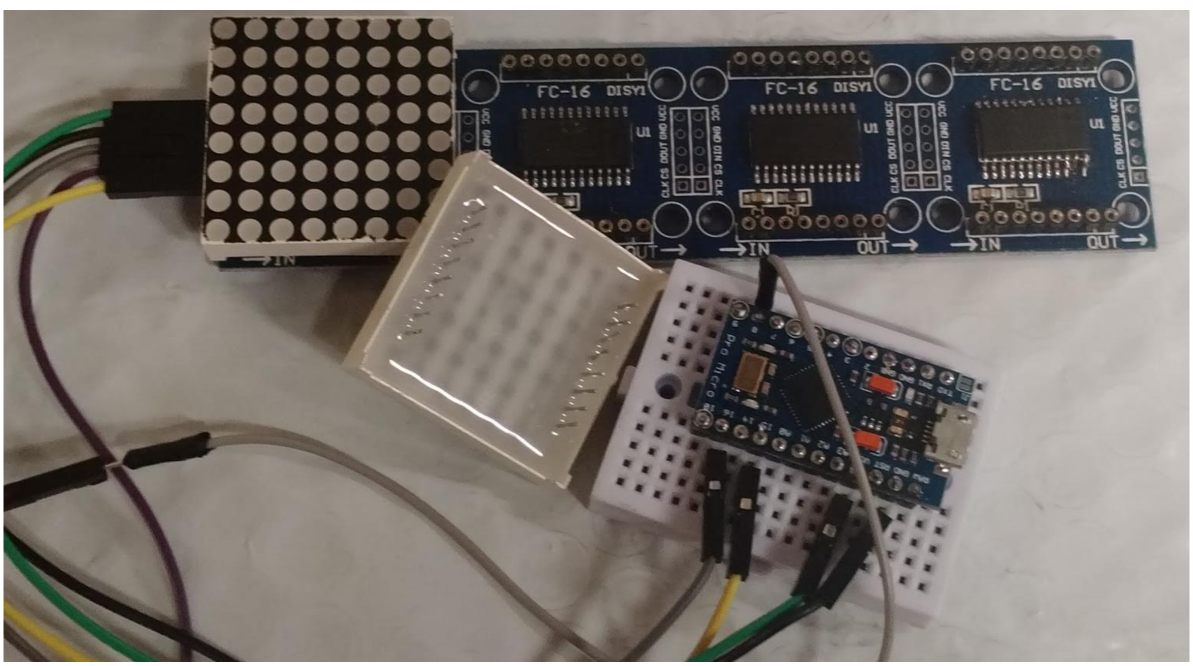


Figure 45: The test prototyping hardware with MAX7219 and the Arduino

This is a prototype that will test the functionality of the LEDs and the shift registers wired together to make a display that can change and control each LED and its brightness. The first step of this prototype is the LED matrix and this will consist of an 8x8 matrix that are attached to a perfboard 5mm apart and they will be wired according to the schematic shown below. Next the shift register will need to be connected in this case the MAX7219 will be used. This prototype will be wired to a Arduino board so it will be a Arduino sending data and power to a MAX7219 then that will be connected to the 8x8 matrix this is shown in the figure below. The figure below shows some other components and these will be included in the prototype and each have their own purpose. The resistor RSet is there to set the current through the LEDs and this will be calculated using both a chart given in the MAX7219 manual using data from the LED manual to find the correct value for that resistor. The two capacitors located near the bottom of the schematic are there to reduce the noise from the power source which for this prototype will come from the Arduino board [15].

Testing the prototype is done by programming the Arduino to display some given characters and the program will be run to see if the LEDs display the corresponding character. The current through the LEDs should be measured and the voltage from the source and this current should also be taken when the light level is then programmed to dim using the PWM that to see the change for different light level outputs and that each value is still in the LEDs parameters. Running this prototype will test a number of things it will test that each of the LEDs work and this will also give a small test model that will allow the sizing of character and the brightness of the LEDs as well as the spacing to make sure that each of these are to the desired specifications for the project and will allow the final device to have a display that will pass the specifications given. The prototype will also test the MAX7219 chip this will allow us to test its capabilities of controlling the LEDs making sure it can provide the right power to each LED and that it can use PWM to adjust the light levels. This prototype only used one MAX7219 chip so this makes it so it is not able to test the working of multiple chips connected together this can be done later by make another LED matrix and connecting the MAX7219 to it then the wires seen going out the bottom of the schematic below will be connected to the next MAX7219 chip to be passed data and power.

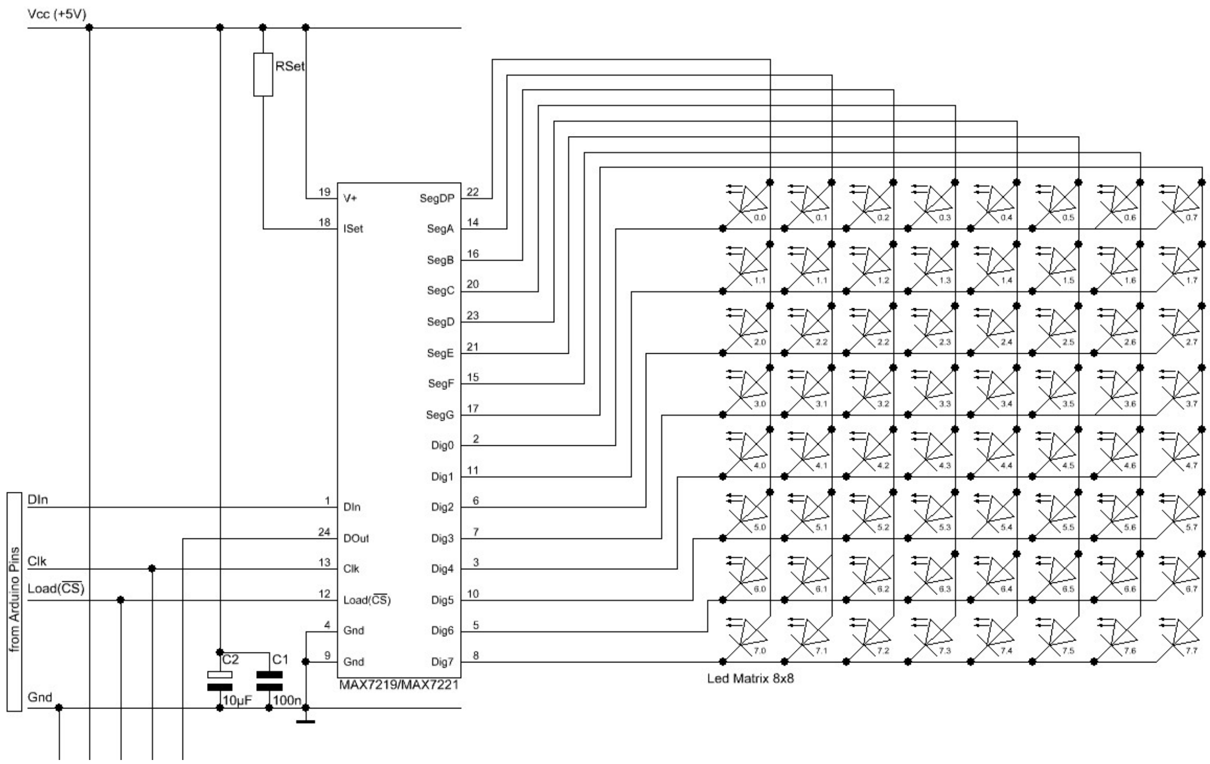


Figure 46: Wiring schematic for display prototype using MAX7219 [15]

10.2 GUI Prototype

The first prototype GUI is a basic design and is not functional. The purpose of this prototype is to give visualization to the design and purpose of the GUI this will allow the later full programming to be set toward a goal and to have an outline. The prototype is simply an image or snapshot of what the GUI will be, so this GUI will have all the functions that will be in the GUI such as the message text boxes for line one and line two, a check box to switch between single line and two-line displays, a textbox for the message display timer as an input in seconds, a preview button, a dropdown menu for scrolling speed, a checkbox that will enable and disable scrolling. A second prototype of the Graphical User Interface will be a written program that displays a connection screen that will search and verify that the flip sign has been found and a connection has been established successfully through USB. The next, prototype of the GUI after a successful USB connection would be a basic mockup of the Graphical User Interface with only one basic programmable message that will be assigned to button one. This setup will consist of a page with the heading “Custom Flip-Sign Screen”, a textbox for the message to be typed into, and a button that the user can press to send the message through the serial interface to the microcontroller.

10.3 Build Prototype

The final device was not the first one built and it was the second device using shift registers as there was a basic prototype build. The first build will be a prototype this is because after putting all the pieces together there will be testing that can only be done in a final product and things do not always go together nicely. After building the first device it will first be tested that it can run the main function of displaying the messages and if not, it will be analyzed to find the problems for the future editions. The displaying of the messages is only one part of the test that this will need to pass as there are a lot of constraints that the device will need to pass so this final device will have to go through each of the testing phases in the testing section and a detail spec sheet for the current model will be kept and then all of the places that the model failed or did not do as well as could be done are then edited and modified to ideal create a better device and after all aspects of the device are tested the new device model schematics will be drafted and the second device will be made and this processes was repeated until the device is at peak operation.

11. Testing Phases

Testing the device is a necessity in order to verify that the requirements and specifications have been met. There are many differing aspects in creating this project that all need to be well tested for all kinds of situations. As seen below, the major topics of testing are testing the battery, testing for our project specifications, other pieces of the hardware, testing the software in separate units then as a whole and test the project as a whole with everything assembled and using the combination of the software and hardware.

11.1 Testing Battery

To test the battery, there were many trials. These trails were conducted to make sure the results are consistent with the advertised battery specifications. We started by testing the battery time at one hundred percent brightness with all of the LEDs being turned on for the whole duration of the test. The time was recorded and checked with our specification of at least fifteen minutes of battery time. We tested this several times to make sure that this time is consistent with each attempt. Also, the time will be check at each other brightness setting to determine how much more time the user has with the device running at differing intervals of brightness. Lastly the device will have two lines of text running and the brightness set at 80% and the time will be recorded to get an average runtime value.

The other specification for the battery that needed to be addressed while doing these trials of testing, was the battery level being displayed. The battery level needs to be displayed with a consistency of what power level that is actually left in the battery. In order to do this, we will need to be checking to see if the level displayed is consistent with the level tested during our trials described previously. For example, when the battery is full, the display battery level should reflect that that battery is full every time it is checked until the battery is not full anymore. We also need to test that when the device is being charged the charging in process indication is displayed and when it becomes full, the device is able to indicate that it is full.

11.2 Testing Display

The viewing distance for this project has a specification set to be forty five feet, this means that the text on the display must be readable at forty five feet. To test this specification the sign can be programmed with six messages that only the person holding the sign knows and are each something that could be quickly understood. Next three other people with good vision stood at forty five feet and not knowing the messages they will each be shown for three seconds and then each wrote the message down and after all six messages are shown each person should of been able to understand each of the messages. If after the messages are shown and there was a miss but it is found that this was mainly caused by a miscommunication then the test will be redone with new messages. The test needed to turn a success rate of 17 out of 18 to pass.

Confirming the specification is not the only thing that was tested. A new test was done in order to find the maximum viewing distance as well as some real-world tests. The maximum viewing distance was found by repeating the test for the specification but increasing the distance by 10 feet each test until there is a fail then the test will be dose a last time at 5 feet short of the failed distance and the furthest viewing distance that passed will be tested again and taken as the maximum viewing distance.

The real-world tests allowed us to find the limitations and performance of the device. The first test had the user of the device be walking and the sign is meant to be held casually as to make is lightly bounce and the viewers will stand varying distances from the user as in the maximum distance test and its best distance will be taken. The next test had the device deal with different light levels the tests above will be used while the light mode is on auto and outside in the daylight, where the sun is not at noon but an hour or 2 past noon. The tests were taken with the user facing both to the side, towards, and away from the sun. The sign visibility was much less in the sun and had trouble viewing in direct sun at distance.

To expound on the interference of the light the glare angles would be found if any by tilting the device toward the sun or large light source. The Maximum viewing angles would be found by staring in front of the device and walking around it till a displayed message becomes unreadable at 30 feet. Next the automatic light adjustment will be measured, first the speed of the sensor was tested and that it does change this will be done by changing the ambient light with the use of a light to greatly change the room light levels then if possible recording the time to switch the light level but if it is less than a second it is a pass. Next the light level was measured starting in a dark room and then an adjustable light slowly increases the ambient light levels and each time the device adjusts the brightness that current room light level will be recorded. Lastly the output light levels of the device was recorded by going in a dark room and reading each of the brightness levels on manual and the maximum brightness will be checked to the specification.

11.3 Testing Durability

The durability of the device is a constraint specification of the device that needs to be tested and confirmed or fixed. To test the durability the device will be attached to a clamp at the handle at a 90-degree angle and then weights will be added to the other end of the device to meet the specification based on calculations using the weight and length to find the torque. The device will also be given some real world test waving it in an extended had to again test the durability of the device mainly at the weakest point where the handle and display meet and then it will be placed outside in direct sun for 30 minutes, this is to check that it will be able to withstand the heat of normal use without overheating or something melting for this reason the device will be run through all of its messages and the battery will be checked that it has not dropped a lot and that it will recharge.

11.4 Testing Hardware

Each piece of hardware has been tested to make sure all of the components work without any malfunctions and that each part will work for the particular needs for this project. Every LED, shift register, and button will have to go through testing before they are selected to be used for the product.

Our initial tests for the various LEDs we considered for our main matrix involved connecting them to our 8x8 LED Matrix LED driver which is then connected to our Arduino Micro Pro board. The LEDs have been tested by using a basic circuit. The circuit consists of using a 3.7-volt battery as a power source that is connected to a one-hundred-ohm resistor and an LED. The tester uses a multimeter (check battery level using multimeter give how to) to verify the battery has charge. Once the battery is certain to have power the tester will hook up the led to the circuit. If the LED lights up and maintains light for three seconds then the led passed the test. With a led that has successfully passed the test, the tester will isolate the successful led into a container that will be marked “passed”. With a led that failed the test, the tester took the LED and place the component into the trash.

The buttons were tested by using the basic circuit from the LED’s test and attaching a button to the circuit between the battery and resistor. The tester pushed the button and check to see if the led illuminates. If the LED lights up after the button is pressed then the button works correctly and can be placed in a container marked “passed”. If the LED does not turn on then the tester should throw away the button.

The shift registers were tested by connecting one LED per output pin. The tester then use a circuit with the 3.7-volt battery, a resistor, and one shift register. Once the circuit has been made and hooked up properly, one led should light up then turn off. When the first led turns off the next led should turn on then off. Once the second led turns off the third led should turn on. Once the third led turns off the first led should turn back on and repeat the cycle. If this sequence runs then the shift register works properly. The tester will now take out the shift register and replace it with a new one to test until enough shift registers have been tested to meet the requirements of the led flip sign.

Testing connectivity by USB cable will be conducted. The tester takes the LED flip sign and take the USB cable provided and connect the micro USB side to it. Next, the tester will connect the other side of the cable which, will be a USB A connector, to a computer. If the connection works successfully, then the Graphical User Interface should recognize and connect to the Electronic Flip Sign and the charging battery indicator should show up on the LED display of the sign.

11.5 Testing Software

In order to test the software, we went through a series of phases for testing that involve different aspects of our software. Once we know that the various software aspects operate independently, we will then test implementing them together. It is expected that incorporating the differing parts together will take a lot of time to debug to make it work. First, we did unit testing for each software component. This included USB connection, Bluetooth connection, auto launching program from device, letter characters, string of characters, text scrolling, Button to display.

There needed to be code written for a computer to recognize the device has been connected. To test this, connect the micro USB cable to the device and the USB 2.0 end to the computer. Navigate to the start menu then to file explorer. On this window under the devices and drives tab there should be a device called LED flip sign. If this shows up, then the test was successful. If the device doesn’t show up, then the test was not successful.

There needed to be code that will handle connecting to the device wirelessly using Bluetooth. To test this, hold down buttons five and six also known as the bottom middle button and bottom right button on the sign for three seconds. After three seconds one of the LEDs of the sign will start to blink. This means the device is discoverable. Using a computer with Bluetooth, search for Bluetooth devices. The led flip sign should pop up as a discoverable device. Click on it and select pair. If the pairing was successful then the device should show up under the This PC folder- devices and drives in file explorer. If the pairing was unsuccessful then an error message should be displayed from a pop up window stating that the device was not successfully paired and to try again at a later time.

There needed to be code that will handle auto launching the program once the device is connected to the computer and the proper program has been installed. To test this, the user will plug in the LED sign into a computer using a USB to micro USB cable. Once the device is connected, a pop up window should appear and the user should be prompted with a question asking if he or she would like to customize the LED flip sign with a yes button and a no button. If the user presses the yes button, then the GUI will open and present the user with the tools to customize the LED flip sign. If the user clicks on no, then the pop up window will close and nothing else should happen. If these actions function the way they were specified then the test was successful. We ended up not using the auto launch feature in the end of this project.

There needed to be code that will handle creating characters on the LED display. To test this system, the user passed in the first index of the array that holds all of the letters, numbers, and symbols to the testing displayed on the led matrix. The led matrix should illuminate the letter A. The user will go one by one through the array sending each index of the array to the led matrix and make sure the LEDs light up correctly. If all the letters, numbers, and symbols appear on the LED matrix display correctly then the test had a successful run.

The Graphical User Interface has to search for a connection to a USB port through built in function of the programming language. This allows the user to see which comport the device is connected too. This method was tested by plugging in a USB device into the computer and running the program. The program should be setup to display a message on the screen saying, “Connection Successful” if the USB device was discovered and connected to. If there was no device found or the connection could not be established, then there should be an error message stating no device detected and the code had to be adjusted accordingly.

The Graphical User Interface needed to search for a connection of the flip sign through Bluetooth. In order to test this, the tester had to have successfully paired the LED flip sign to a computer or mobile device. Next, the tester needed to run the application that was made for the flip sign. When the application starts up, there will be a splash screen that will be searching for a connection, if the screen displays that there has been a successful connection then the testing was a success. If the search times out after thirty seconds, then the testing has failed and adjustments to the code will be required to fix this issue.

The Graphical User Interface stores the characters and the configuration settings of each message in objects and using those objects full of data, display the information on the GUI. To test this, a test must have connected the flip sign to a computer or mobile device and launch the GUI, have a successful connection to the sign and when the customization screen appears with all the message boxes and configuration tools, there should be existing data from the sign for the buttons that have been programmed. If a button has scrolling text, then the check box should be checked. If the message was on one line, then there should be a check in the single line checkbox and the line 2 should be grayed out. Depending on the scroll speed that was set, the number one, two, three, or four should populate the dropdown box to the respected message. If all the information is correct and match the right buttons then the test was successful. If any of the information is missing or out of place then the test was a failure.

The GUI will need to be able to send a preview of a message the user wants to see while in the custom message screen. To test this the tester will set up a button with a configuration and push the preview button located in that button’s section. If the sign lights up with the correct message and performs the correct actions set by the configuration in the Graphical User Interface then the test had performed successfully. If the message came out wrong or any of the features did not perform as expected, then the test had failed.

Once the unit testing is done, integration testing began. Integration testing went through each of the units of code that work together and test in pairs. Once all the combinations are tested and work. then the group expanded to three units together and so on until all of the combinations become one unit. When the testing code becomes one unit, a system test is conducted.

A system test was black box testing. The tester was able to input a custom message to each button and once that information is sent to the device the tester unhooked the USB cable and test the inputs and make sure the outputs match. The tester then checks to make sure all of the button functions as expected with all of the button combinations. The Brightness setting has to be tested and see if the output is correct. The tester has to see if the Bluetooth works with the given button combinations and see if another device can detect the sign. The Tester will then plug in the USB cable to the flip sign and see if the charging indicator appears.

11.6 Testing Button Functions

The buttons not only each activate a message to be displayed for a certain amount of time but they also and settings to display that message for unlimited time and then there are button combinations that change the brightness of the display. To test the basic message display each of the buttons are programmed to display different messages and at different time periods then the buttons will be each pressed after the device is disconnected and the displayed message and display time for each message are compared to what was programmed and this should result in 100% accuracy to pass. Each of the six messages will then be tested for the infinite display time and this will test by setting it to this mode and leaving it for 15 minutes. The button combination used to adjust the brightness is tested by is first switching the auto mode and changing the ambient light and seeing if the brightness is changing or not then when in manual mode the button combinations for increasing will set it to the maximum value and then to the lowest and then back to the maximum and the number of light settings will be recorded and compared to their correct value.

12. User Guide

This device is a handheld device that is lightweight and portable being fully battery powered LED sign that allows you to display one or two-line messages to people in your area. This electronic flip sign is one hand operated and easy to use in both the active use and the programming. This device is easily programmed using an intuitive program that lets you control how the messages are displayed and what they say with the option to save up to 6 different messages that can all be accessed on the fly. The electronic flip sign can connect to Android and PC corded and wirelessly. This device will make covey a message in a whole new light for all kinds of situations and at all different events making communication easy and efficient.

12.1 Charging

Before first use the device, the device must be fully charged. The device can be charged by connecting the device to a computer via micro USB or using the micro USB wall charger adapter provided with the LED flip sign. The micro USB port on device is located on the left-hand side of the device as seen in figure 34 below. The sign will light up a battery symbol and show the progress of the battery’s charge. When the battery is empty, the battery will be dark in the middle and only an outline of the battery will be shown. When the battery is full the display will show a battery that is fully illuminated.

12.2 Powering on and First Use

When the device is charged it can be used. The power switch is located on the top left-hand side of the device and can be switched up to the on position to start the device, see figure below. The micro USB port is at the bottom of the display, near the handle and can be used to charge the battery as well as connect to the software on a Windows device. Once on, each of the buttons will now display the default message “HELLO!” for 30 seconds. These messages can be changed please see the programming section below.

12.3 Connecting the Device

To connect the device using USB, take the USB type A end or larger end and plug it into the computer. Take the other end, the Micro USB end of the cord or the smaller end and this will plug into the micro USB port located on the bottom left side of the device as seen in figure 47 below. Connecting via Bluetooth can be done to either an Android or a PC. First make sure that the Bluetooth is on the device that you are trying to connect to. To turn on the Bluetooth on the device hold down buttons 1 and 3 the top left and right buttons to start the connection. On your device, you should now see the electronic flip sign show up on your computer's Bluetooth interface and it should now be selected and the devices should then proceed to connect completing the connection.

12.4 Button Function

There are a few different button functions that are used to control the brightness and the methods of display. To change the brightness both from auto to manual modes the button 1 is held at the top left and then the middle bottom button is then pressed button 5 this will change the mode from auto to manual. To change the brightness, hold button 1 the top left and then press the bottom left or right buttons 4 and 6 to decrease or increase the brightness respectively. The buttons are also used to display the messages that correspond to the button that is pressed. If the message is given a specific amount of time, in seconds, to be displayed on the LED matrix, then the user will just press the button and then message will display for that amount of time. Once the time is up, the message will be removed from the display and the device will go into sleep mode until another button if pressed. If a button is pressed that does not have a specific display time, then the message will stay displayed as long as the button is being pressed and only as long as it is being pressed.

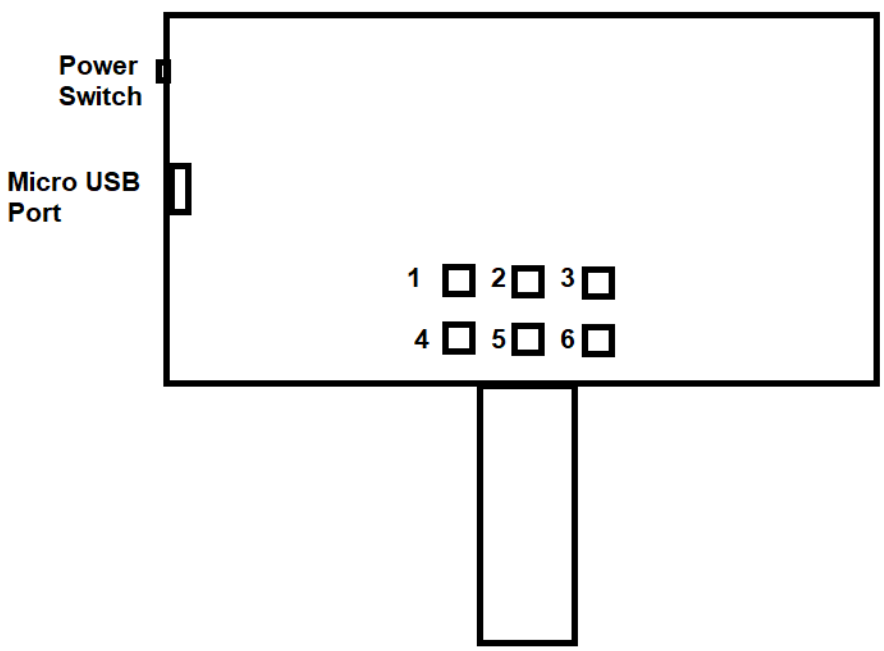


Figure 47: The rear of the device showing key components

12.5 Programming

To program the device it must be connected to a computer or phone, see connecting the device. Once connected to a device there should be a prompt to open or download the software. If the prompt to download the software appears on the screen then proceed to click OK to install after which you will be prompted to run the program, press OK. Once the program is running you will see a screen as below and can begin programming.

Single line of text: The **One Line Text** box must be checked and then input the message into the **Line 1** this will be limited to capital letters, numbers, and some emojis and symbols listed in the **KEY**, this message will also be limited to 10 characters. Once the message is as desired check the bubble of the **Button** you wish to program and fill in the **Display Time** text box with seconds that it should be displayed on a single press, then click the **Save** button and that button is now programmed you can close the program and disconnect the device if programming is complete.

Scrolling Text: To make a single line of scrolling text first check both the **One Line Text** and the **Scrolling Text** boxes. Fill **Line 1** with the desired message that can have capital letters, numbers, and the values in the **KEY**. There is no limit to the number of the characters in this line. Note: the longer the message the longer it will take to cycle going the same speed. Next check the **Scrolling Speed** bubble with the speed that the text will scroll from the given 4 speeds. Now the duration should be filled in the **Display Time** text box with the desired display time in seconds and check the bubble for the **Button** that is to be programmed. Once all is as desired click the **Save** button to program that button.

Two lines of text: First make sure the **One Line Text** is not checked and then you can add your message to both **Line 1** and **Line 2** each of these messages will appear on the line that they are written and each line has a character limit of 10 per line. The character limit still applies for capital letters numbers and the **KEY**. Now the duration should be filled in the **Display Time** text box with the desired display time in seconds and check the bubble for the **Button** that is to be programmed. Once all is as desired click the **Save** button to program that button.

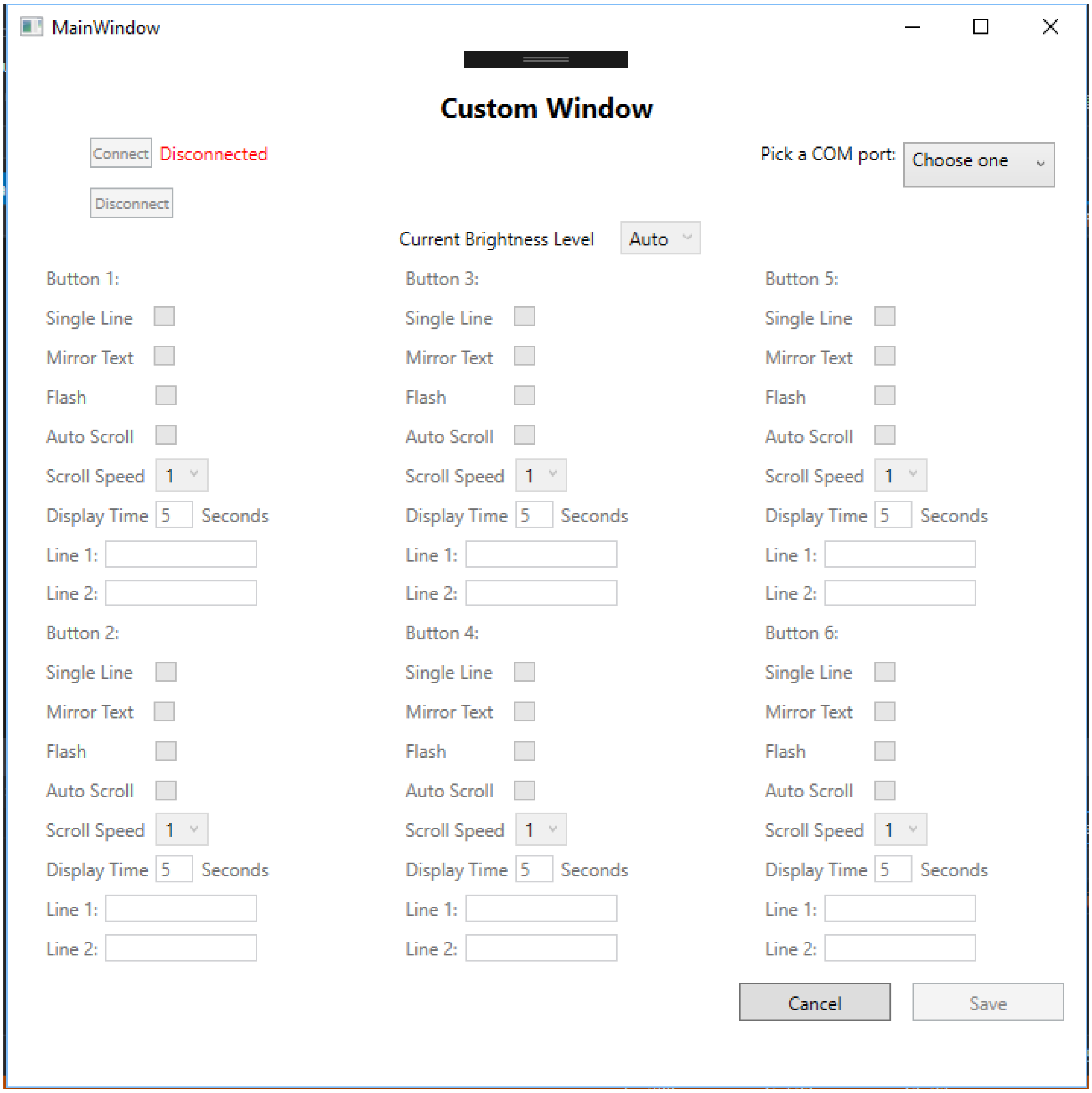


Figure 48: Prototype GUI Design

Button: This is a bubble check for which button is to be programmed and is arranged as it is on the device.

Line 1: This is the primary message input text box and is limited to 16 characters in most cases. Both Lines are limited to capital letters and numbers as well as the symbols and emoticons located in the KEY.

Line 2: This is only used when one line text is NOT checked and is for the second line of text.

Display Time: This textbox only accepts numbers. The number entered here will be the time the text is displayed for a simple button press in seconds.

One Line Text: checking this box will deactivate the Line 2 text box and will only display the characters in Line 1

Scrolling Text: Checking this box will deactivate the line 2 box and will only display the Line 1 Text but the character limit is no longer enabled. Note: "One Line Text" must be checked to check this box.

Scrolling Speed: This is a 4-drop down menu options each representing a speed that the text will move across the display.

KEY: This has emoji text inputs that will be accepted in the text lines as well as all accepted symbols.

Brightness Mode: This has 2 check bubbles that is for Manual and Auto. Manual will set the brightness to a steady state set as the Brightness bubble below. Auto will set the brightness to an automatic level.

Brightness: This is a bubble check that will set the brightness at the given level. Only effective if Brightness Mode is in Manual.

Save: This is a button that will save the current message to the current button on the device.

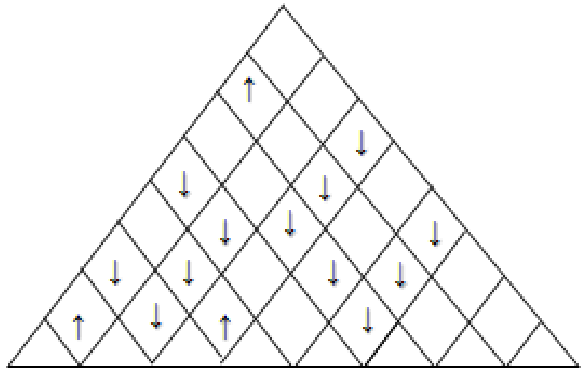
13. House of Quality

The House of Quality, as seen in Figure 18, is a diagram used to show the relationship between the requirements the device needs as a product and the requirements needed for the device to meet all of its specifications. A part of this diagram is there to be requirements that can be tested and viewed in the device as a test.

For the Electronic Flip Sign, the customer would want the device to function and be user friendly while creating their messages and using the sign on the go. The device should also be able to be visible to the people that the messages are meant to be seen by. That is a unique aspect to this design, because the customer that purchase this device is not the only one that needs to be satisfied by the device. The design requirements discuss more on how the device will meet the specifications. The key aspects that are a part of the design requirements is the Brightness level, the battery's durability and battery life, the dimensions of the device, the weight of the device, the cost, the durability, View Distance in which the sign can be read, and the User Interface. The brightness level means that the sign's display brightness and be changed by the user and that brightness can be maintained even at full brightness. The battery's durability and battery life means that the battery a part of the device can take and maintain a charge as well as has the battery life required to maintain the minimum battery life discussed in the specifications. The dimensions of the device simply mean that the dimensions of the device stays within the proper dimensions discussed in the specifications to allow for portability and screen size. The weight of the device also needs to be within the range of the weight in the specification for the portability and usability aspect. The cost is to make the design as cost effective as possible. The durability of the device is to make sure that sign can last even on the go. The view distance of the device needs to be a good distance to meet the user's and reader's needs. The User Interface needs to be user friendly in both the software and the hardware. These all can be seen at the top of the House of Quality found in Figure 18.

The Customer Requirements show what the customer would want from this device and whether or not the device met them. These requirements are the viewing distance, the quality of the device, how easy the sign is to use, the amount of time the device can be used, the weight, size and cost of the sign. The viewing distance needs to be far enough of the customer to be able to show their messages at reasonable distances. The quality of the device needs to be something that the customer feels like they got their money's worth. The device also needs to be easy to use in both adding messages with the features they want to use and being able to display the sign with ease. The time use can be within the proper battery life time. The weight and size needs to be right for the user to be able to carry and use without issue and the size is still big enough for a good-sized display. The cost needs to be a little as possible for the customer to want to buy the device. All of these requirements can be seen on the left side of the House of Quality found in Figure 18.

The house of quality also shows the targets for engineering requirements which are the goals that need to be met for the projects design specifications to be completed. Each section on the bottom of the house of quality corresponds to the section on top that is was discussed previously. If each of the targets for engineering requirements are met when designing the device, then the design requirements are also met. The targets for engineering requirements are also all specific requirements that can be proved through demonstration, measurement and testing. For example, the design requirements state the battery as a requirement and the targets for engineering requirements states the battery to be able to stay on with a message displayed for fifteen minutes. This can be tested by simply displaying a message to wait and see how long the device stays on before turning off. If the device lasts for at least fifteen minutes then the requirement has been proven to have been met.



Direction of Improvement		+	+	-	-	-	+	+	+
Customer Requirements (What)	Design Requirements (How)	Brightness	Battery	Dimensions	Weight	Cost	Durability	View Distance	User Interface
Improve +	Viewing Distance	↑↑	↑↑	↓	↓	↓↓		↑↑	
	Quality	↑↑		↑		↓↓	↑↑	↑	↑↑
	Ease of Use			↑	↑↑			↑↑	↑↑
	Use Time	↓↓	↑↑	↓	↓	↓		↓↓	
Reduce -	Weight	↓	↓↓	↑	↑↑		↓↓		
	Size		↓	↑↑	↑		↓	↓↓	
	Cost	↓	↓			↑↑	↓	↓	↓
Targets for Engineering Requirments		1500 nits	15 Minuets On Time	10" x 4" x 1"	0.5 lbs	$40 Prouction	2.5 Nm at Handle	45 ft	6 Buttons

Figure 49: House of Quality

This figure shows the correlations between design requirements and consumer requirements whether it be beneficial or detrimental to each other. Engineering requirements also add some design specifications to be obtained by the end product.

14. Plan

This is the details of each thing, activity, milestone that should be done. It is important to have a plan for each of the goals this makes it so that things can be completed in an orderly manner and that each task and milestone has a designated person that is responsible for that task giving someone specific lead and authority in each task to keep thing organized.

14.1 Senior Design 1

The milestones seen in table 9 are the milestones the team accomplished during their time in Senior Design 1 for the design of the device and the creation of the prototype for the project. One of our overarching goals for Senior Design 1 is to create a working prototype that involves work components for the major aspects of the project. Some of the milestones found in table 9 that lead a hand to this are the prototype schematic, the embedded code, the basic GUI, testing the module, a breadboard prototype and finally a working prototype. The working prototype will help in designing the final product, because we will be able to reference our prototype and see what parts work and what parts do not work. The other major goal for our project in Senior Design 1 is to have everything prepared for the assembly and testing of our final project. This involves having the working prototype, as well as the LED Display Design, a basic GUI design and having all the hardware selected that will be used for the final project design.

Milestones	Completed (Y/N)	Week Due Completion*	Responsible
Hardware Selected	Y	6	EE
LED Display Design	Y	7	EE
Prototype Schematic	Y	8	EE
Test Code	Y	10	CPE
Basic GUI	Y	10	CPE
Module Testing	Y	10	EE
Prototype Display	Y	12	All
Breadboard Prototype	Y	13	EE
Working Prototype	Y	15	All

Table 13: Milestones SD 1

The Electronic Flip Sign's creation is documented by the Team as the process to completing the project is done. Table 13 shows the documentation that has been done during the team's time in Senior Design 1. This is done to help organize the team and their tasks that need to be done. The documentation is also needed to help map out the design and researched needed when creating the Electronic Flip Sign.

Course Documentation	**Description**	**Due Date**
Group creation	Team created	8/25/17
Project Idea	Project defined	9/1/17
Initial Project Paper (10 Page)	Document giving details and constraints of the project	9/22/17
Table of Content	List to of topics to be covered in the final paper	10/6/17*
First Draft Document (60 Page)	Rough draft of final paper	11/3/17
Second Draft (100 Page)	Draft of final paper	11/17/17
Final Document	Final documentation full details of the project	12/4/17

* estimated

Table 14: Course Documentation SD 1

14.2 Senior Design 2

The milestones seen in table 15 are the milestones the team accomplished during their time in Senior Design 2 which involved the creation of the final product. This means that the programs have been finalized along with the software, the final product has been assembled and that all aspects have been tested and adjusted as needed. The process may take a few revisions, but once everything has been assembled and tested till the device meets all the specifications given, the final device as complete.

Milestones	Completed (Y/N)	Week Due Completion*	Responsible
PCB Design & Layout Rev A	Y	3	EE
PCB Schematic Rev A	Y	4	EE
PCB Manufactured Rev A	Y	5	EE
GUI Rev A	Y	6	CPE
Code Rev A	Y	6	CPE
Assembled Rev A	Y	6	EE
Test Rev A	Y	6	EE
PCB Design & Layout Rev B	Y	7	EE
PCB Schematic Rev B	Y	8	EE
PCB Manufactured Rev B	Y	8	EE
GUI Rev B	Y	9	CPE
Code Rev B	Y	9	CPE
Assembled Rev B	Y	9	All
Test Rev B	Y	10	EE
Final Device	Y	12	EE
Final Code	Y	13	CPE

Table 15: Milestones SD 2

Table 16 shows the documentation that the design team has to complete during their time in Senior design 2.

Documentation	Description	Week*
CDR Presentation		2
Conference Paper	Paper to present as summary	5
Middle Term Demo	Draft of product	8
Final Bill Materials	Final total of parts for production and prototype	14
Production Report	information to start production	14
Final Presentation and Demo	Completed project	15
SD Day/Exit Interview	Open demo	16

Table 16: Course Documentation SD 2

To summarize, the tables above describe the milestones along the project for both senior design 1 and 2. The other tables are for the course documentation that will be submitted through senior design 1 and 2 for the completion of the project. The goal of these tables is to outline the work that was needed to be done in order to have a successful project at the end making a path to victory.

15. Cost Estimation

One of the objectives of this project is not just to make a working project but to design it for mass production and for the price at mass production levels to be under $40. Listed in the table below is a chart that shows the components needed and then a list of their prices per unit for both buying at a single part by part to build a single device and then the cost of a single unit if enough parts were bought to build 1000 units. The price of each unit greatly differs with the single prototype costing much more than the bulk price, this is good because the goal is to make the cost in bulk as low as possible because to make the device into a production device the prices need to be low so that the price to consumer is lower making it marketable. Though the goal of production price is $40 this is not what the consumer will pay, this is why the price is as low as some of the competitor devices, even though they do not do the same thing, they are priced around $60 and we will have to be in this realm in order to compete in this market. The cost estimates can also be seen in Table 17 below.

Components	Rev A Cost Prototype	Rev B Cost Prototype	Production Cost 1000 units
PCB	$90.00	$90.00	$3.00
Case	$10.00	$10.00	$2.00
Battery	$5.00	$5.00	$2.00
LED’s	$8.00	$8.00	$6.00
LED Drivers	$20.00	$20.00	$8.00
Bluetooth Module	$10.00	$10.00	$5.20
Ambient Light Sensor	$0.70	$0.70	$0.30
6 Buttons	$5.00	$5.00	$0.50
MPU	$4.20	$4.20	$3.50
USB Port	$0.80	$0.80	$0.50
USB Interface & Cord	$3.55	$3.55	$2.20
Power Switch	$0.50	$0.50	$0.30
Passive Components	$0.75	$0.75	$0.40
Power Circuitry	$1.00	$1.00	$0.60
Total	**$159.50**	**$159.50**	**$34.50**

Table 17: Cost Table

15.1 Part List

This is a picture, seen below, of all the components that are going to be used in this project. Each part was ordered in excess for most of the parts this is so that if parts do not work or are destroyed in the processes of building we will have the parts to replace them. There is another reason to have extra and also some pieces were ordered that are not used in the build but these are used to make other builds to make the build in other ways to test and see different outcomes of different methods of making the circuit.

The bottom right of the figure below depicts a reel of LEDs that are red 3528 smds and there are 2000 in the reel shown. To the left of the LEDs in the Bluetooth module in the reel casing. Above the LEDs is the battery and above that is a tape of 22 74HC595 shift registers. The top right of the figure below is the battery charge controller in the reel casing and to its left is a tape of 10 decade counters. Below the decade counters is a tape of 100 NPN transistors. The last 2 things on this list is the are the buttons and switch it should be noted that these may change as the case has yet to be designed and it is unknown if they will be usable in the case design.

15.2 Consultants, Subcontractors and Suppliers

The main consultant that was used to help with designing this project was the project sponsor and advisor, Professor Michael Young. He has given us suggestions and information where research can be done on various aspects to the design of this project. He also gave us valuable information on companies that can be used as subcontractors when developing our final design. One of the subcontractors that was advised to us is Quality Manufacturing Services Inc. After touring the facility and speaking with the employee that was the tour guide they have offered to provide their services for printing some of the components to the custom circuit board as mentioned in the facilities section of this paper.

The suppliers that were used to distribute the purchased parts for this project are Digikey, Amazon, and AliExpress. Digikey was used to purchase the ATmega 328P microcontroller, 4020 Bluetooth chips, transistors, shift registers, IC decoder, Lithium Ion 4.2V controller, shift registers, and switch. The testing LED 8x32 matrix array, the Arduino micro pro, and 3.7V battery were purchased and supplied by Amazon. The components purchased from AliExpress were the buttons, switch and LEDs.

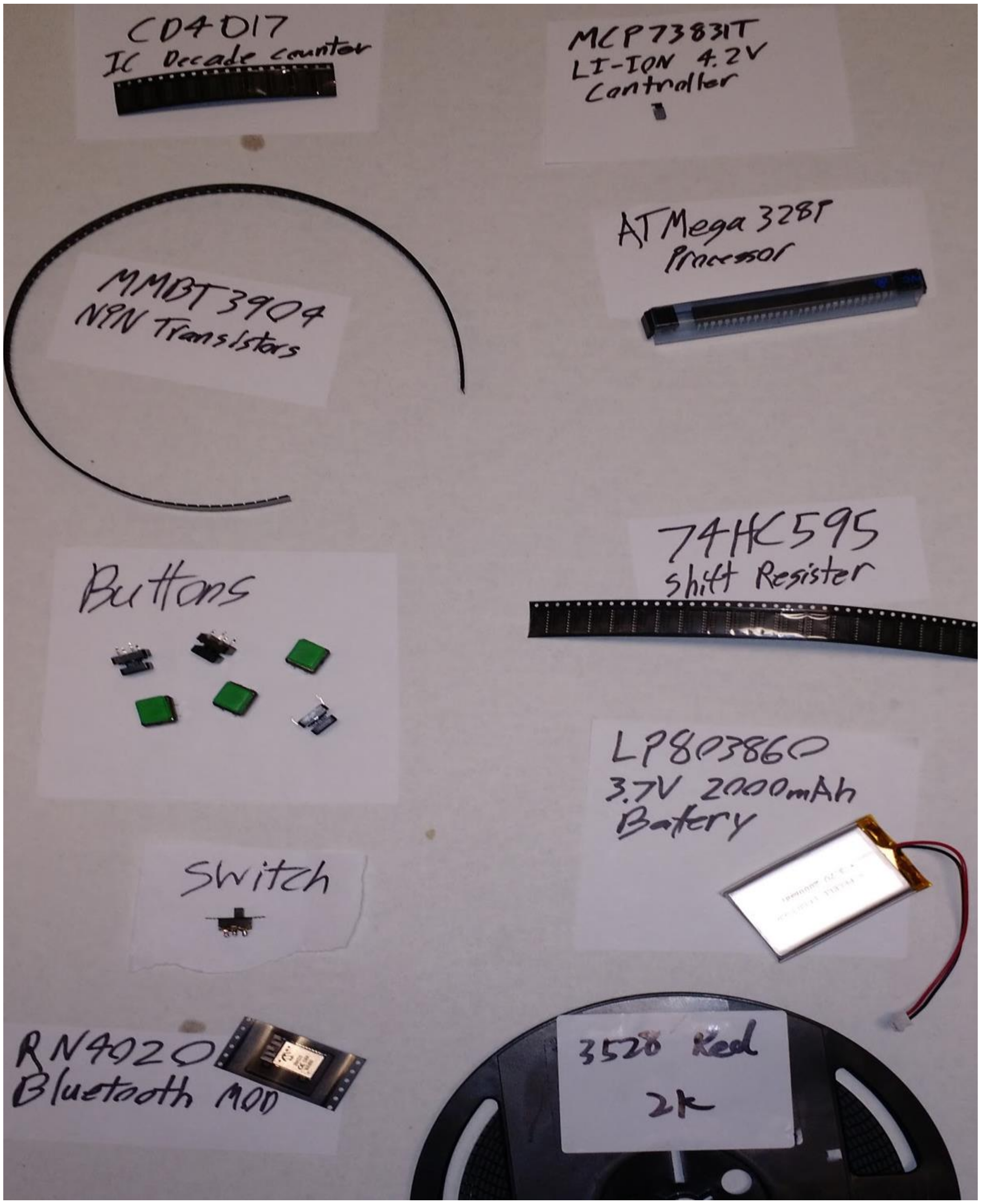


Figure 50: Parts for project

16. Conclusion

The Electronic Flip Sign has many different ways to be built and designed. Each method that has been discussed and researched has its own pros and cons. In the end, the device can only have one design and one build that has to be decided upon in order to not only pass each of the tests on the specifications, but it should ideally be the one that can outperform all other designs and builds. The decided build also needs to be able to meet customers' needs when operating the device. In order to find the best build without building each of the designs, each component was compared to others that could also be used to find the one that best did its job. The process doesn't end there each component not only has to be checked with others like it but also with the system in general, since all the components are part of a single system each one affects each other this means that the part that is the best in its own category might not work in the overall system at all or just make the overall effectiveness of the device go down compared to some other options. To prevent this each component is compared with the system as a whole and then checked again as another component are chosen or changed out for a better option. Below is discussed the final composition of the design and build for what will be the first revision of the device this will be the design that after research will create the best device within the specifications and restraints given.

16.1 Final Hardware Design

The first core component of the design that holds and connects all the components of the device together is the printed circuit board (PCB). The main changes in the PCB is the use of 2-layer verses 4-layer, for this device the best option was to use the 2-layer option this turned out to be cheaper and with the design restraints there is not a problem with the size that will be taken by the use of two 2-layered boards one for the LEDs and one for all other circuitry. The LEDs are going to be wired across the full length of each row and column with cathodes and anodes respectively. The LEDs are going to be spaced at about 5.5mm from each other. The device will be powered off of a 3.7 volt lipo battery there will be no boosting of the voltage and this voltage level will drive both the LEDs and the components.

The processor that will be running this device will be the ATmega32U4 this will accomplish each goal while also being cost effective. The LEDs used are 3528s these are large so that could be soldered easily while still small enough to fit the design and they also provide a brightness that can accomplish the goals of the design. The Bluetooth chip was a decided as it was able to be added to the PCB in surface mount form and it is a later version of Bluetooth and it has a good power usage. Light sensor is a light dependent resistor setup for cost while still accomplishing the goal. The drivers consist of six 74HC595, npn transistors, NMOS MOSFESTS, and decade counters to create a simple shifting and multiplexing system with the transistors to protect the drivers. The battery will be a li-po 2000 mAh single cell running 3.7 volts providing a medium between the cost and effectiveness and durability. Each of these components make up a system that is within the budget but still having a high output and good efficiency while maintaining a small and light form factor.

The schematic as seen below is the layout of our device although not in its complete form has the major parts of the build present, each was made with

Autodesk Eagle [69]. The first figure below (figure 51) is of the left hand side of the schematic highlighting the CD4017. This image shows how the decade counter will be connected. On the right side of the picture the LED matrix can be seen and this is wired as described in the LED wiring section earlier in the paper. There are wires going down from the 4017 and these go to another 4017 but each is controlled independently by the processor, completing the height of 20 LEDs for the device. The wires going from 13 and 14 pins go to the 13 and 14 pins of the next chip as this is ground and clock. From the processor there is the data and clock going independently to each of the 2 chips. Each 4017 will also have power and a ground not shown in the figure.

The second figure shown below (figure 52) is of the same schematic but it is the top left corner top side instead of left and this figure has been rotated clockwise 90 degrees. This figure highlights the 74HC595 shift register and shows how it is wired. The charging circuit is set up to take power from the USB VIN and will then charge the circuit and in the current setup has an indicator LED as well. In the middle right hand side the shift register is located and can be seen going to each column as it is rotated and can control 8 columns so if the full schematic was shown there would be 6 of these chips across the top of the LED matrix with 2 4017 on the left but this is a very large circuit and showing the entire circuit is just a green blur but a figure of both the photos combined can be seen in the 3rd figure below (figure 53) this shows a part of the whole schematic and the cascading drivers, this figure goes the full height but only a fourth of the width, the LDR circuit can also be seen in this figure. The processor is sending the clock, latch, and data to the 74HC595. The shift register has 5 wires going down and these go to the next shift register they sync the clocks and are a ground as well as the latch but the data is also sent but it is received through the 14 pin and then sent to the next shift register through the 9 pin. Together these two drivers control the NPN transistors that in turn control the LEDs and the drivers are controlled by the processor all being powered from the 3.7V battery and the grounds are controlled by the NMOS that is run by the 4017 and this make the workings of the device. The NPNs can handle up to 100mA of current but will only ever be lighting one LED making it more than adequate, PNPs were tried since it is sourcing but the voltage at the 74HC595 output was lower than the supply voltage so the LEDs would not turn all the way off, the NPN worked even though there is the voltage drop of about a volt it was still able to light the LEDs. On the ground side there is the MOSFETS which can handle up to 7.5A more than the current even if all the LEDs were on and pulsing at 80mA. The second board contains all the other components that are in standard setup of the processor, charging circuit, fuel gauge, buttons, and Bluetooth.

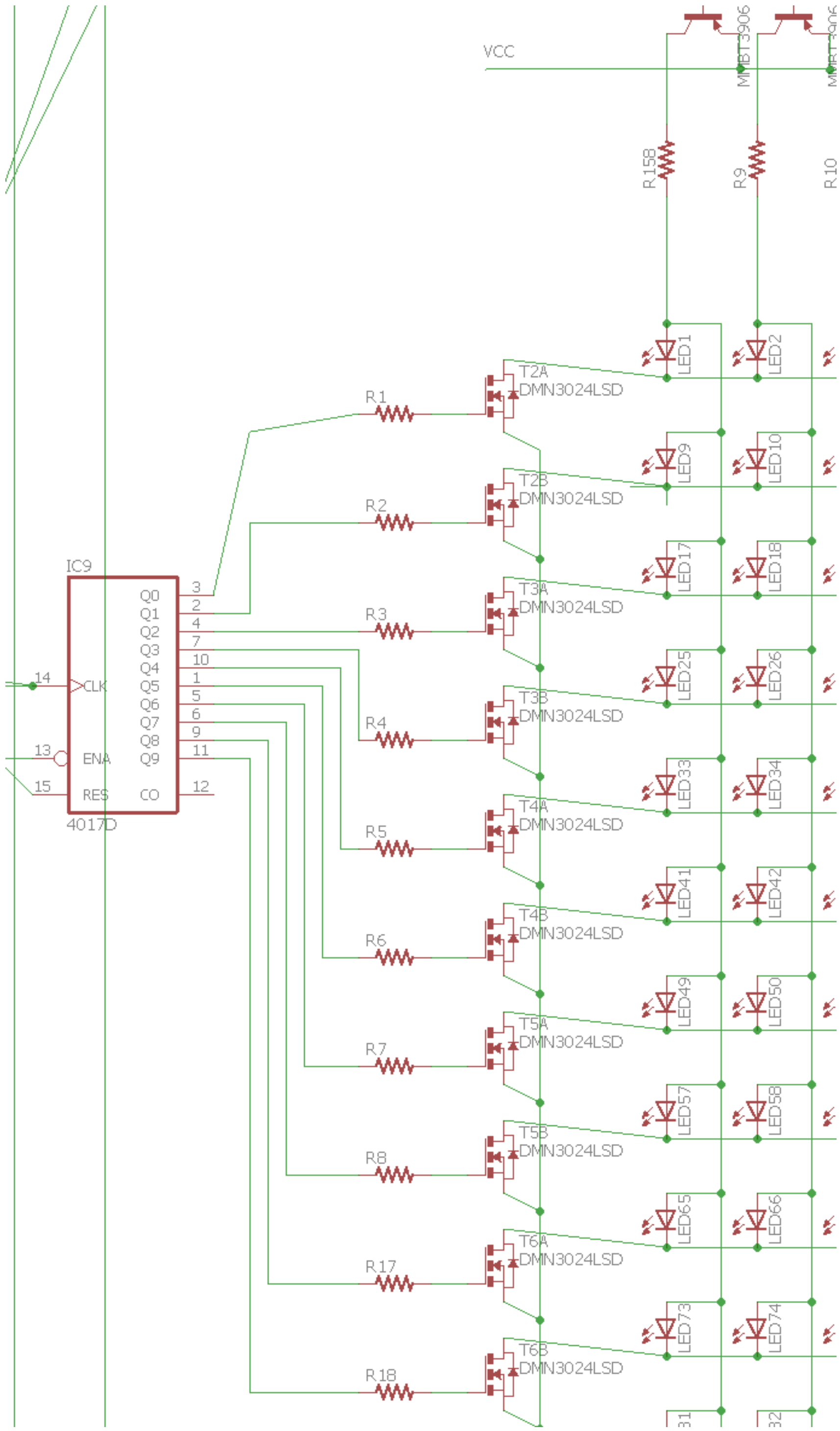


Figure 51: Final schematic part 1

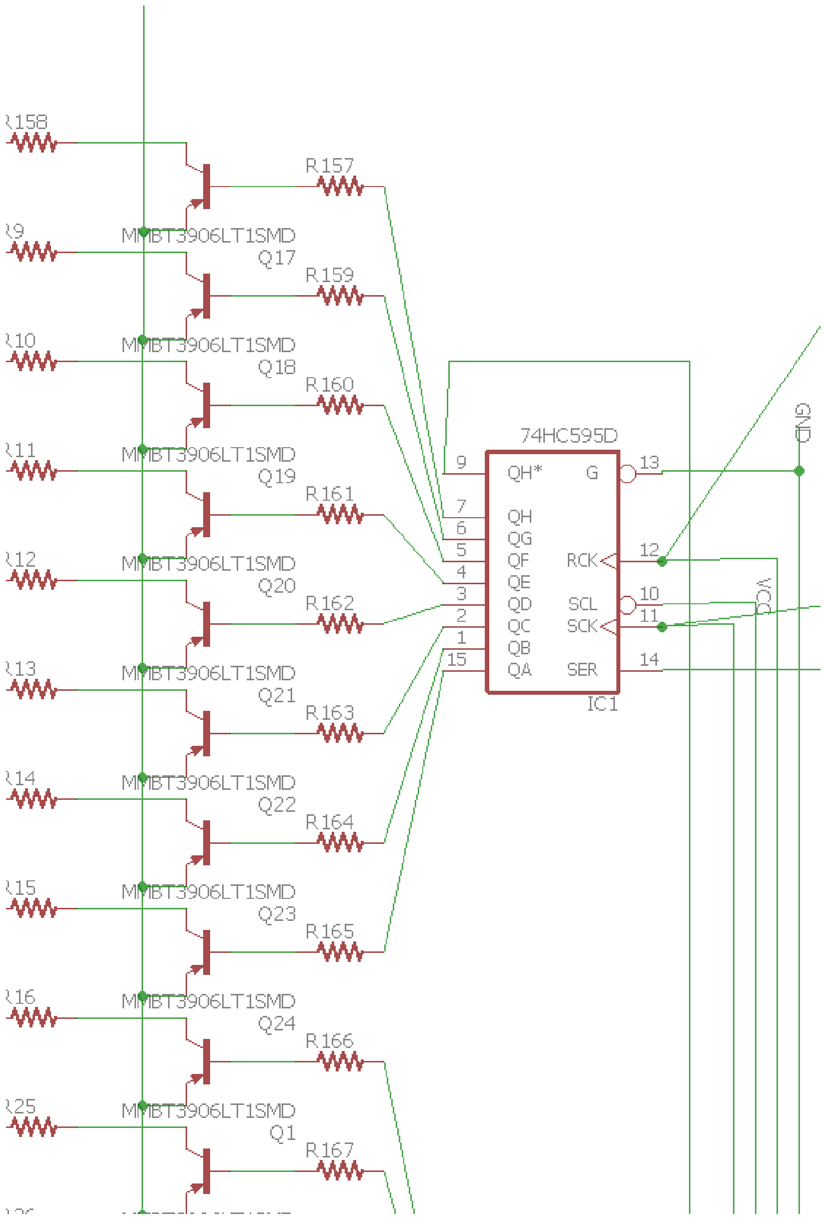


Figure 52: Final schematic part 2

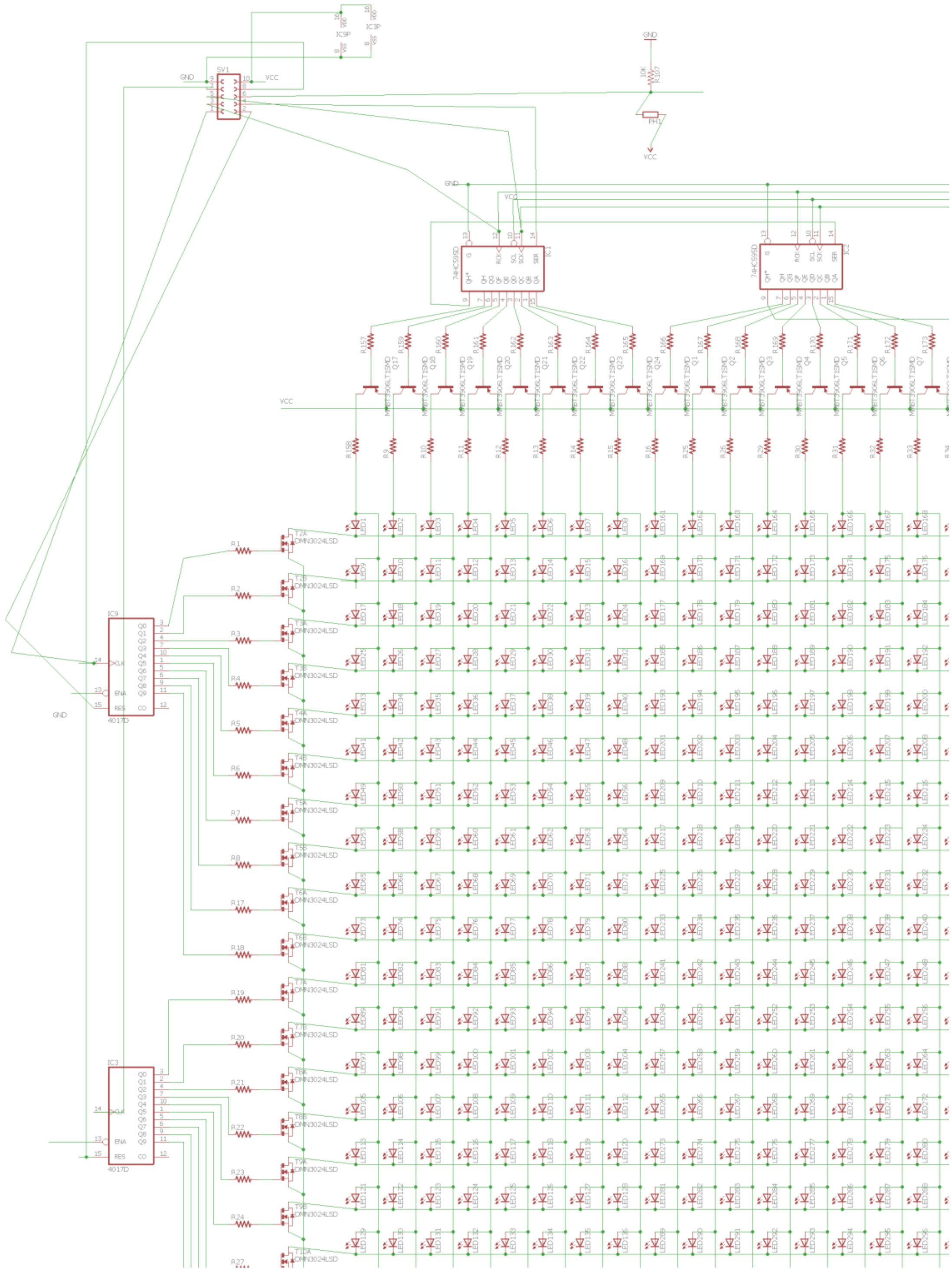


Figure 53: Final schematic part 3

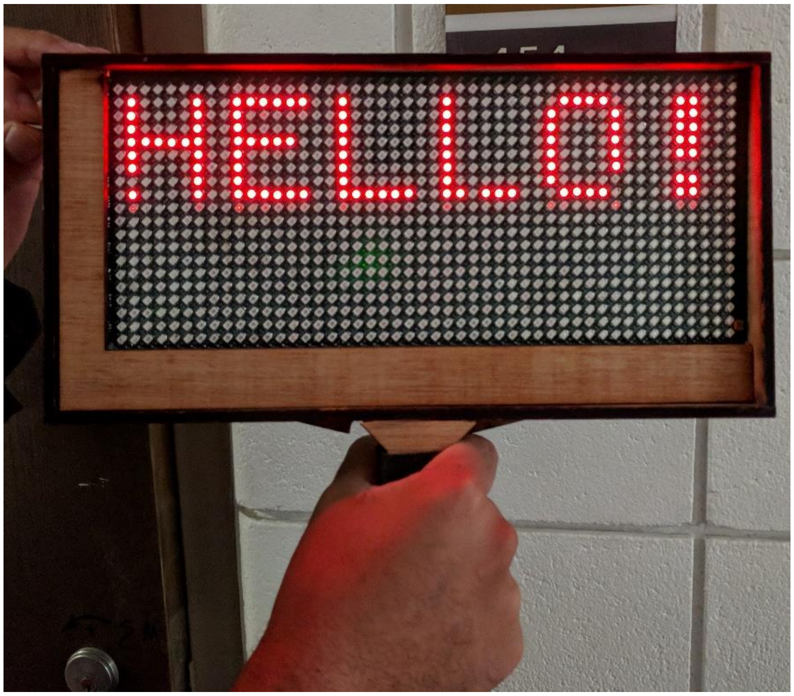


Figure 54: Final device

16.2 Final Case Design

The case design is designed as a 3D model and then laser printed built to use the least amount of materials while still keeping within the form factor and durability needed to be used freely. The case is wooden and have a cover and contain the LEDs and internal components. The case houses 2 PCBs a smaller one and a larger one and then the battery while each PCB will be spaced from each other. The case has port for power and connection via the USB port. There is a power switch that goes through the case for simple toggling and then there are 6 buttons on the device.

16.3 Final Code and GUI

The final code includes a number of functions that allows the ease of use of the device for the user as well as allow the device to reflect all the features that have been decided for the device when compiling the specifications. The code controls the storing and displaying of messages, as well as the connections for sending and receiving data via both Bluetooth and USB. There is also code for controlling and operating each of these operations. The display buttons have a number of different operations such as, displaying the programmed message, how it was programmed and has options to invert the text, flash the text, scroll the text and the speed of the scrolling. The buttons also have combinations to operate the activation and deactivation of the Bluetooth feature. The brightness is controlled using the button combinations and the GUI as well as the auto brightness that the code will determine using the data from the light sensor.

Graphical user interface (GUI) is the software that allows the user to program the device to perform the desired actions while in use. This GUI is designed to be simple and easy to use. The GUI allows the user to change the text that is displayed for each button and then for each button there are options that allow the user to change how long the message is displayed, how fast it scrolls, how many lines of text there will be, if it will scroll or not, and if it will blink for attention before displaying the message.

16.4 Final Product

The final goal of this device is twofold, to create a device that works and functions with ease and optimal results, and then to make a mass-produced device that can be made to sell. The device uses the components chosen and makes a device that is easy to use and is functional making a device that is user friendly and can produce results that will accomplish the initial goals and the needs of the device. The second goal of this device is the mass market prep this comes in the part price line that was set as constraint and this was done so that the mass market price is as low as possible while still making a quality piece of hardware and for this the pricing of the device at bulk order was also a constraint for this device as this is one of the goals for the device.

17. Appendices

This section contains the references used in the Senior Design project document. There will also be emails and other forms of contact that indicate copyright permissions.

17.1 References

[1] T. Scully and V. →, "Everything about LEDs: Basics of High Power LED Lighting", Ledsupply.com, 2017. [Online]. Available: http://www.ledsupply.com/blog/what-you-need-to-know-about-leds/. [Accessed: 05- Oct- 2017].
[2] "A Guide to Selecting a Bluetooth and BLE Chipset · Argenox Technologies", Argenox.com, 2017. [Online]. Available: http://www.argenox.com/bluetooth-low-energy-ble-v4-0-development/library/a-guide-to-selecting-a-bluetooth-chipset/. [Accessed: 05- Oct- 2017].
[3] J. Bartlett, "Software Testing Life Cycle Phases (STLC) - TestLodge Blog", TestLodge Blog, 2017. [Online]. Available: https://blog.testlodge.com/software-testing-life-cycle/. [Accessed: 09- Oct- 2017].
[4] "Introduction to the C# Language and the .NET Framework", Docs.microsoft.com, 2017. [Online]. Available: https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework. [Accessed: 23- Oct- 2017].
[5] V. Mahidov, "Bluetooth Versions and Their Differences | CloviTek", CloviTek, 2017. [Online]. Available: https://clovi.tech/blog/bluetooth-versions-and-their-differences.html. [Accessed: 29- Oct- 2017].
[6] Digikey.com. (2017). DigiKey Electronics - Electronic Components Distributor. [online] Available at: http://www.digikey.com [Accessed 27 Sep. 2017]
[7]"Bluetooth",En.wikipedia.org,2017.[Online].Available: https://en.wikipedia.org/wiki/Bluetooth#Specifications_and_features. [Accessed: 30- Oct- 2017].
[7] “Electronic Flip Sign” Project Proposal Powerpoint by Michael Young
[8] P. Bright, "Provide UNO MEGA2560 code 64x16 dot Matrix LED for Arduino AVR MCU diy Christmas Gifts Sign Light Neon Bright-in LED Displays from Electronic Components & Supplies on Aliexpress.com | Alibaba Group", aliexpress.com, 2017. [Online]. Available: https://www.aliexpress.com/item/Provide-UNO-MEGA2560-code-64x16-dot-Matrix-LED-for-Arduino-AVR-MCU-diy-Christmas-Gifts-Sign/32481337965.html?spm=2114.search0103.3.58.leY9SY&ws_ab_test=searchweb0_0,searchweb201602_3_10152_10065_10151_10068_10130_5490020_5

550020_10307_10137_10060_10155_10154_10056_10055_10054_5470020_10059_100031_10099_5460020_10103_10102_10052_10053_10142_10107_10050_10051_10324_10325_5380020_10326_10084_10083_10080_10082_10081_10178_10110_10111_10112_10113_10114_10312_10313_10314_10078_10079_10073,searchweb201603_15,ppcSwitch_3&btsid=2c4541bc-524f-4330-8643-87f30b0d26d9&algo_expid=5ffd914e-9187-4d96-8388-a3f55cb5c4e4-7&algo_pvid=5ffd914e-9187-4d96-8388-a3f55cb5c4e4. [Accessed: 17- Nov-2017]. - Example of an LED display that is already on the market.

[9] Find Quality Wholesalers, Suppliers, Manufacturers, Buyers and Products from Our Award-Winning International Trade Site. Wholesale Products from China Wholesalers at Aliexpress.com.", Aliexpress, 2017. [Online]. Available: https://www.aliexpress.com/. [Accessed: 17- Nov- 2017].

[10] "Cite a Website - Cite This For Me", Lighthouseleds.com, 2017. [Online]. Available: http://lighthouseleds.com/downloads/dl/file/id/85/1210_plcc2_smd_red_led_datasheet.pdf. [Accessed: 17- Nov- 2017].

[11] G. Definitions and C. Hope, "What is a GUI (Graphical User Interface)?", Computerhope.com, 2017. [Online]. Available: https://www.computerhope.com/jargon/g/gui.htm. [Accessed: 17- Nov- 2017].

[12] "5 aspects of a good user interface – Electronic design services with Argon Design Ltd.", Argondesign.com, 2017. [Online]. Available: http://www.argondesign.com/news/2014/feb/5/5-aspects-good-user-interface/. [Accessed: 17- Nov- 2017].

[13] "8x8 LED Matrix MAX7219 Tutorial with Scrolling Text & Android Control via Bluetooth - HowToMechatronics", HowToMechatronics, 2017. [Online]. Available: http://howtomechatronics.com/tutorials/arduino/8x8-led-matrix-max7219-tutorial-scrolling-text-android-control-via-bluetooth/. [Accessed: 17- Nov- 2017].

[14] Electronics.stackexchange.com. (2017). Driving 10x10 LED matrix with AVR?. [online] Available at: https://electronics.stackexchange.com/questions/30501/driving-10x10-led-matrix-with-avr/30504 [Accessed 1 Nov. 2017].

[15] E. Fahle, “Hardware and schematics,” LedControl, 2015. [Online]. Available: http://wayoda.github.io/LedControl/pages/hardware. [Accessed: 02-Nov-2017].

[16] "STOP/STOP, STOP/SLOW Flashing LED Hand-Held Paddle Signs | Traffic Safety Corp.", Xwalk.com, 2017. [Online]. Available: http://www.xwalk.com/pages/stop-stop-LED-paddle.htm. [Accessed: 17- Nov-2017].

[17] "C Sharp (programming language)", En.wikipedia.org, 2017. [Online]. Available:

https://en.wikipedia.org/wiki/C_Sharp_(programming_language)#Syntax. [Accessed: 17- Nov- 2017]

[18] “LED FUNDAMENTAL,” Footprintz Inc., 2013. [Online]. Available: http://footprintz.in/tutorial/led-education.html. [Accessed: 02-Nov-2017].

[19] "C (programming language)", En.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/C_(programming_language)#Embedded_C. [Accessed: 17- Nov- 2017].

[20] "Flowchart Maker & Online Diagram Software", Draw.io, 2017. [Online]. Available: https://www.draw.io/. [Accessed: 17- Nov- 2017].

[21] rwb, “PIC18 Pulse Width Modulation (PWM) DC Motor Speed Controller with the RPM Counter Project,” ermicro shop, 2009. [Online]. Available: http://www.ermicro.com/blog/?p=1461. [Accessed: 02-Nov-2017].

[22] “File:USB3.0 connectors.svg,” File:USB3.0 connectors.svg - Wikimedia Commons. [Online]. Available: https://commons.wikimedia.org/wiki/File:USB3.0_connectors.svg. [Accessed: 02-Nov-2017].

[23] “File:USB3.0 connectors.svg,” File:USB3.0 connectors.svg - Wikimedia Commons. [Online]. Available: https://commons.wikimedia.org/wiki/File:USB3.0_connectors.svg. [Accessed: 02-Nov-2017].

[24] “File:USB3.0 connectors.svg,” File:USB3.0 connectors.svg - Wikimedia Commons. [Online]. Available: https://commons.wikimedia.org/wiki/File:USB3.0_connectors.svg. [Accessed: 02-Nov-2017].

[25] UCF, “Electronic 2 lab manual,” 2012. [Online]. Available: http://www.ece.ucf.edu/files/labs/EEL%204309%20Jan%202012.pdf.

[26] UCF, “Electronic 2 lab manuel,” 2012. [Online]. Available: http://www.ece.ucf.edu/files/labs/EEL%204309%20Jan%202012.pdf.

[27] “Powerful 160A - 12V, 24V, 36V, 48V DC PWM Consists of 2 80A Units, combined.,” Pulse Width Modulator - PWM. [Online]. Available: http://www.hho4free.com/pulse_width_modulator__pwm.html. [Accessed: 03-Nov-2017].

[28] “USB,” Wikipedia, 02-Nov-2017. [Online]. Available: https://en.wikipedia.org/wiki/USB#Transmission. [Accessed: 03-Nov-2017].

[29] “Pulse-width modulation,” Wikipedia, 16-Oct-2017. [Online]. Available: https://en.wikipedia.org/wiki/Pulse-width_modulation. [Accessed: 03-Nov-2017].

[30] “AVR Guide: Constant Voltage Transformer Operation,” UST. [Online]. Available: https://ustpower.com/comparing-automatic-voltage-regulation-technologies/avr-guide-ferroresonant-transformer-type-voltage-regulator/avr-guide-constant-voltage-transformer-operation/. [Accessed: 03-Nov-2017].

[31] "CD4067 Multiplexer used to control LEDs | code, circuits, & construction", Tigoe.com, 2017. [Online]. Available: https://www.tigoe.com/pcomp/code/arduinowiring/540/. [Accessed: 17- Nov- 2017].

[32] B. (White), L. pack), J. Pack), R. pack, S. 74HC595, A. SMD and S. Arduino, "SIK Experiment Guide for Arduino - V3.2 - learn.sparkfun.com", Learn.sparkfun.com, 2017. [Online]. Available: https://learn.sparkfun.com/tutorials/sik-experiment-guide-for-arduino---v32/experiment-14-using-a-shift-register. [Accessed: 17- Nov- 2017].

[33] "Cite a Website - Cite This For Me", Web.csulb.edu, 2017. [Online]. Available: http://web.csulb.edu/~hill/ee400d/Technical%20Training%20Series/02%20Intro%20to%20Arduino.pdf. [Accessed: 17- Nov- 2017].

[34] "Atmel Studio 7 | Microchip Technology Inc. | Microchip Technology Inc.", Microchip.com, 2017. [Online]. Available: http://www.microchip.com/avr-support/atmel-studio-7. [Accessed: 17- Nov- 2017].

[35] [H. Garden and S. Electronics, "How Microcontrollers Work", HowStuffWorks, 2017.[Online]. Available: https://electronics.howstuffworks.com/microcontroller1.htm. [Accessed: 17- Nov- 2017].

[36] ["Texas Instruments - MSP430G2553IPW20R", Ciiva, 2017. [Online]. Available: https://ciiva.com/m/part/msp430g2553ipw20r-2954934. [Accessed: 17- Nov- 2017].

[37] Buchmann, “What’s the Best Battery?,” Battery University, 2017. [Online]. Available: http://batteryuniversity.com/learn/archive/whats_the_best_battery. [Accessed: 15-Nov-2017].

[38] "Lab 12: Basics of LED dot matrix display - Embedded Lab", Embedded Lab, 2017. [Online]. Available: http://embedded-lab.com/blog/lab-12-basics-of-led-dot-matrix-display/. [Accessed: 17- Nov- 2017].

[40]“A Designer's Guide to Lithium (Li-Ion) Battery Charging.” *A Designer's Guide to Lithium (Li-Ion) Battery Charging | DigiKey*,

[41]"template31", Ffden-2.phys.uaf.edu, 2017. [Online]. Available: http://ffden-2.phys.uaf.edu/631fall2008_web.dir/wallace_webpage/7_PV.html. [Accessed: 17- Nov- 2017].

[42]"template31", *Ffden-2.phys.uaf.edu*, 2017. [Online]. Available: http://ffden-2.phys.uaf.edu/631fall2008_web.dir/wallace_webpage/7_PV.html. [Accessed: 17- Nov- 2017].

[43] realfinetime electronics, “Method to Control 8*8 LED Matrix using Shift Register IC 74595 and Arduino Mega,” Method to Control 8*8 LED Matrix using Shift Register IC 74595 and Arduino Mega «, 2014. [Online]. Available: http://www.learnerswings.com/2014/08/simple-method-to-control-88-led-matrix.html. [Accessed: 16-Nov-2017].

[44] "Cite a Website - Cite This For Me", Atmel.com, 2017. [Online]. Available: http://www.atmel.com/Images/Atmel-7766-8-bit-AVR-ATmega16U4-32U4_Datasheet.pdf. [Accessed: 17- Nov- 2017].

[45] “BU-903: How to Measure State-of-charge,” Battery University, 25-Oct-2017. [Online]. Available: http://batteryuniversity.com/learn/article/how_to_measure_state_of_charge. [Accessed: 17-Nov-2017].

[46]"Multiplexing With Arduino and the 74HC595", *Instructables.com*, 2017. [Online]. Available: http://www.instructables.com/id/Multiplexing-with-Arduino-and-the-74HC595/. [Accessed: 17- Nov- 2017].

[47]"Reducing Arduino Power Consumption - learn.sparkfun.com", *Learn.sparkfun.com*, 2017.[Online].Available:https://learn.sparkfun.com/tutorials/reducing-arduino-power-consumption. [Accessed: 17- Nov- 2017].

[48]M. Rouse, "What is interrupt? - Definition from WhatIs.com", *WhatIs.com*, 2017. [Online]. Available: http://whatis.techtarget.com/definition/interrupt. [Accessed: 17- Nov- 2017].

[49]T. Agarwal, "Types of Interrupts in 8051 Microcontroller | Interrupt Programming", *ElProCus - Electronic Projects for Engineering Students*, 2017. [Online]. Available: https://www.elprocus.com/types-of-interrupts-in-8051-microcontroller-and-interrupt-programming/. [Accessed: 17- Nov- 2017].

[50]"Watchdog timer", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Watchdog_timer. [Accessed: 17- Nov- 2017].

[51]Open source Lithium battery charger modules. [Online]. Available: https://wemakethings.net/2013/02/02/lithium-battery-chargers/. [Accessed: 01-Dec-2017].

[52]K. Lewis, "Texas Instruments Innovation Lab « CECS", Cecs.ucf.edu, 2017. [Online]. Available: http://www.cecs.ucf.edu/texas-instruments-innovation-lab/. [Accessed: 01- Dec- 2017].

[53]"ECE Department - Senior Design Lab", *Ece.ucf.edu*, 2017. [Online]. Available: http://www.ece.ucf.edu/labs/sdl.php. [Accessed: 01- Dec- 2017].

[54]"Homepage - UCF Libraries", *UCF Libraries*, 2017. [Online]. Available: https://library.ucf.edu/. [Accessed: 01- Dec- 2017].

[55]K. Lewis, "Idea Lab « CECS", *Cecs.ucf.edu*, 2017. [Online]. Available: http://www.cecs.ucf.edu/idea-lab/. [Accessed: 01- Dec- 2017].

[56]"Boost converter", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Boost_converter. [Accessed: 02- Dec- 2017].

[57]"Tektronix MSO4034B Oscilloscope | Apex Waves", *ApexWaves.com*, 2017. [Online].Available:https://www.apexwaves.com/analyzers/tektronix-inc/mso4000b/MSO4034B?gclid=CjwKCAiAu4nRBRBKEiwANms5Wyo7NUW7ur6mdYepI8nvsVfIonGyd_DTJT7ZrqyCQgdNiW5fUDkrcxoCp-4QAvD_BwE. [Accessed: 03- Dec- 2017].

[58]2017.[Online].Available:http://www.testequipmentdepot.com/tektronix/digital-multimeter/digital-multimeter-dmm4050.htm?gclid=CjwKCAiAu4nRBRBKEiwANms5W9bky_UK9TmxxOh-

vNMPBcOTsqgw1Q4E928ETI1iOOIrcjD0nyM_xBoCbeoQAvD_BwE. [Accessed: 02- Dec- 2017].

[59]0. Keysight (Agilent) Technologies E3630A 35 W Triple Output, "Used Keysight (Agilent) Technologies E3630A 35W Triple Output, 6V, 2.5A & ±20V,0.5A|MATsolutions",*Matsolutions.com*,2017.[Online].Available:https://www.matsolutions.com/productdetails.aspx?pk=49153?sra=191503&sra=190018&gclid=CjwKCAiAu4nRBRBKEiwANms5Wx4VWggvqFrGfeEKIJyAMQfba507AXTwUTkAt4JfGs8wF1nMaas9-xoCdV4QAvD_BwE. [Accessed: 02- Dec- 2017].

[60]P. Express, ""Stahl Tools STSSVT Variable Temperature Soldering Iron Station" from www.parts-express.com!", *Parts-express.com*, 2017. [Online]. Available:https://www.parts-express.com/stahl-tools-stssvt-variable-temperature-soldering-iron-station--374-100. [Accessed: 03- Dec- 2017].

[61] En.wikipedia.org. (2017). *Blender (software)*. [online] Available at: https://en.wikipedia.org/wiki/Blender_(software) [Accessed 3 Dec. 2017].

[62] En.wikipedia.org. (2017). *3D Slash*. [online] Available at: https://en.wikipedia.org/wiki/3D_Slash [Accessed 3 Dec. 2017].

[63]"10X-20X LED Binocular Stereo Microscope Boom Arm with Gooseneck Light",Amscope.com,2017.[Online].Available:http://www.amscope.com/10x-20x-led-binocular-stereo-microscope-boom-arm-with-gooseneck-light.html?gclid=CjwKCAiAu4nRBRBKEiwANms5W9bwK10-JwwsHlKredSuSur-81uuoTDVGss4ogkMcgLEzcwafKO3XxoCa4YQAvD_BwE. [Accessed: 03- Dec- 2017].

[64] Spaceclaim.com. (2017). *Software for Engineers | 3D Prototyping | SpaceClaim*.[online]Availableat:http://www.spaceclaim.com/en/Solutions/PrototypingandManufacturing/AdditiveManufacturingandRapidPrototyping.aspx [Accessed 3 Dec. 2017].

[65]3. 3D Printers, "GearBest USA: Anet A8 Desktop 3D Printer Prusa i3 DIY Kit US PLUG-$203.83", *GearBest*, 2017. [Online]. Available: https://us.gearbest.com/3d-printers-3d-printer-kits/pp_343643.html?wid=24. [Accessed: 03- Dec- 2017].

[66]"Cite a Website - Cite This For Me", *Learningaboutelectronics.com*, 2017. [Online].Available:http://www.learningaboutelectronics.com/images/DC-to-DC-boost-converter-circuit.png. [Accessed: 03- Dec- 2017].

[67]"Quality Manufacturing Services, Inc. | Electronics Manufacturer, PCB Repair, Printed Circuit Board Assembly, Quick Turn PCB", *Quality Manufacturing Services PCB Assembly & Prototyping*, 2017. [Online]. Available: https://www.qmscfl.com/. [Accessed: 04- Dec- 2017].

[68] Instructables.com. (2017). Make a 8x10 L.E.D Matrix. [online] Available at: http://www.instructables.com/id/810-LED-Matrix-with-4017/ [Accessed 4 Dec. 2017].

[69] Autodesk.com. (2017). Download EAGLE | Free Download | Autodesk. [online] Available at: https://www.autodesk.com/products/eagle/free-download [Accessed 4 Dec. 2017].

[70] L. Ada, “Atmega32u4 Breakout,” Intro | Atmega32u4 Breakout | Adafruit Learning System. [Online]. Available: https://learn.adafruit.com/atmega32u4-breakout. [Accessed: 27-Apr-2018].

[71] “SparkFun LiPo Fuel Gauge,” TOL-10617 - SparkFun Electronics, 09-May-2011. [Online]. Available: https://www.sparkfun.com/products/10617. [Accessed: 27-Apr-2018].
[72] N. Seidle, B. Arango, and P. Alberts, “SparkFun LiPo Charger Basic - Micro-USB,” PRT-10217 - SparkFun Electronics, 15-May-2015. [Online]. Available: https://www.sparkfun.com/products/10217. [Accessed: 27-Apr-2018].

17.2 Email Picture

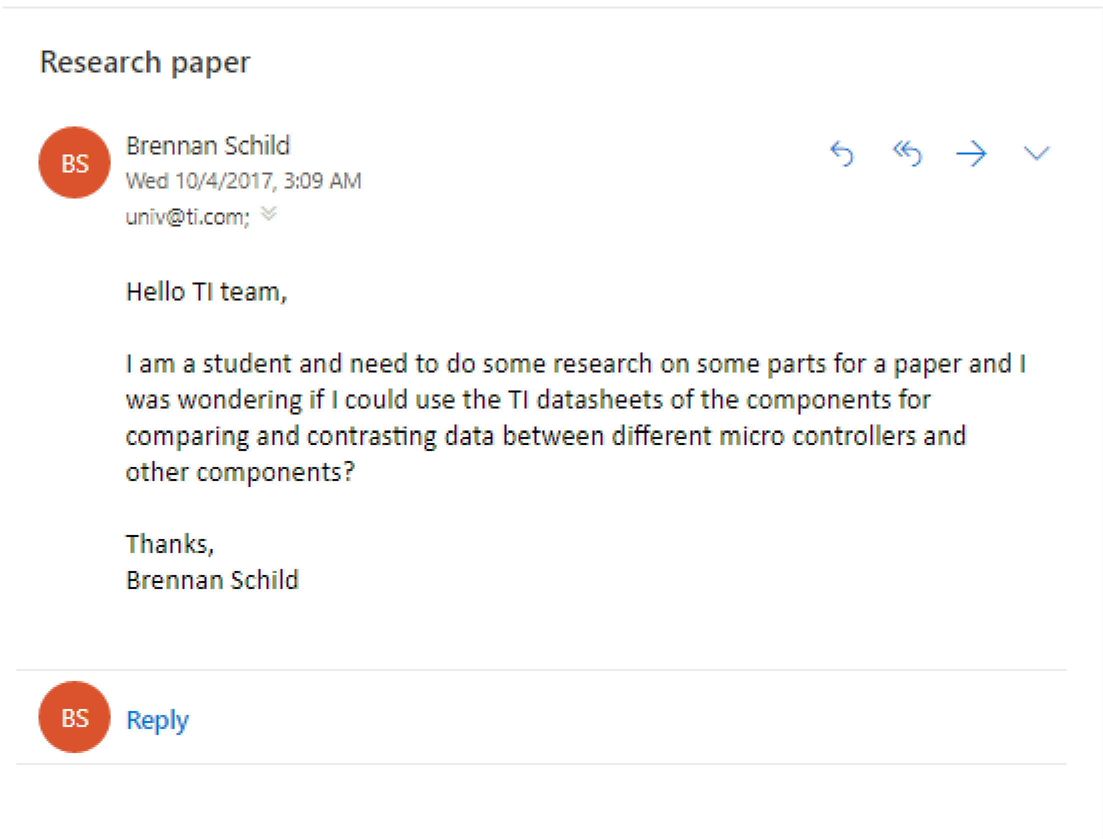


Figure 55: Email sent to TI