EEL 4915 -- Final Document
Senior Design II

Group #6

Matias Canter	nickcanter3@knights.ucf.edu	EE
Taymas Musaev	tmusaev@knights.ucf.edu	CpE
Brandon Carty	bccarty218@knights.ucf.edu	EE
Jonathan Espinal	Espinal_j@knights.ucf.edu	EE

The Smart Skateboard

Table of Contents

1.0 Executive Summary

This senior design project discusses the design and assembly of an electronic device to be attached to the bottom of a skateboards with the sole purpose of classifying each individual maneuver the user does based on the classifications that already exist in the skateboarding community (i.e. “kickflip,” “pop shuvit,” etc.). This electronic device will be paired with a mobile application, also researched, designed and implemented in this senior design project, which will allow the user to get real-time updates of the trick he/she just performed. The device will be attached to the bottom of a skateboard, where it will be relatively unnoticed by the user. This electronic device involves creating a custom PCB with multiple sensors integrated to track the orientation of the board at all times. This PCB will also have a wireless communication chip to allow the PCB to send information to the mobile application as fast as possible.

The printed circuit board we customize will be created using a computer aided design software, most likely EagleCAD. This will ensure that the device works exactly as our specifications outline it. The holster we create to hold our electronic device will also be created using a computer aided software in order to create a three-dimensional design that can later be printed and implemented into our project. Although the creation of this holster is mainly a mechanical engineering problem, it is an integral part to the creation of a successful final product.

This documentation outlines the entire research, construction, and implementation process of the SMART Skateboard. It also breaks down the selection process of this idea along with the motivations and goals that guided our group during the creation of this idea. We also talk about current products that are out on the market, or in the production phase of development, that are similar to the SMART Skateboard, as well as the exact specifics that make the SMART Skateboard unique in comparison to those products.

There are currently two companies that we found to be implementing this sort of device, Syrmo and Trace. The current technology we envision is not on the market, since Syrmo could not find proper funding to continue with their venture and Trace retracted their pursuit for the skateboarding community. Our designed device, the SMART Skateboard, will continuously track the motion, acceleration, and height of the skateboard, while constantly sending the information to the mobile device for the user to assess. The classified tricks will be visible on the mobile device long enough for the user to finish their current session and still be able to see their progress. This is the essential summary of this senior design project: to design and build a PCB device to track motion and orientation, and then to design and build a mobile application that can receive the information from the PCB then immediately display the information for the user to see.

2.0 Project Description

2.1 Project Introduction

Very rarely have the skateboarding world and the electrical engineering world ever met. Even in our fast-paced, open-minded society skateboarding is still somewhat thought of as an "underground" activity within most intellectual circles. This and tradition may be the reasons why skateboarding really hasn't much innovation, technologically speaking, in our great culture. We don't see very many scientists working on new skateboard materials, or doctors deriving formulas for sturdier joints for when skaters fall down a flight of stairs. We want to be the generation that brings the skateboarding industry a new light.

Our idea is to design a smart chip that anyone can attach to the bottom of their skateboard and, by linking it to a mobile app, it will keep track of the past tricks the user has landed. The application will also keep track of the user's speed, acceleration, as well as jump height. All this data should be readily available on a mobile phone application to skate with your friends.

Before this great idea (Original Intellectual Property: Syrmo), skaters had no way of logging their "gnarly" tricks unless there was a cameraman nearby. Our focus was to change this nuisance. By implementing a mobile app, where skaters can save every ridiculously dangerous trick they have ever landed, we can progress this very traditional culture towards the 21st century. No longer will skaters have to break bones to prove to their friends that they indeed did land the triple kickflip, now they can just pull it up on their phones.

Obviously, this idea has a lot of different complications and implementations, yet nothing will quite be able to stop a "skater" from just grabbing the board with his hands and showing off a trick he didn't really land on the fancy new app he just downloaded. Nonetheless, the application will be a great tool for honest skateboarders.

Since we are on a quite strict time restraint with this project, we can't really take this device as far as we would like. We have decided to implement only "flat-ground tricks" meaning that there will be no ramps or railings to worry about, and we can just focus on the considerably simpler flat ground. Also due to time constraints, we will not be able to log every single trick ever landed. Although we would like to do this and it will be attempted, a reasonable goal for this project will be storing the last 5-10 minutes worth of tricks.

The ultimate goal for this project is to design a device that can be attached to the bottom of a skateboard, link with a mobile application, and tell the user what tricks he landed in his previous skating session. If we can accomplish all of this within the timeframe provided, we would also like to work on a possible way to link multiple devices to play a good old fashioned game of H.O.R.S.E. (or S.K.A.T.E.

in the skateboarding world). However, this addition would just be the “cherry on top” so-to-speak.

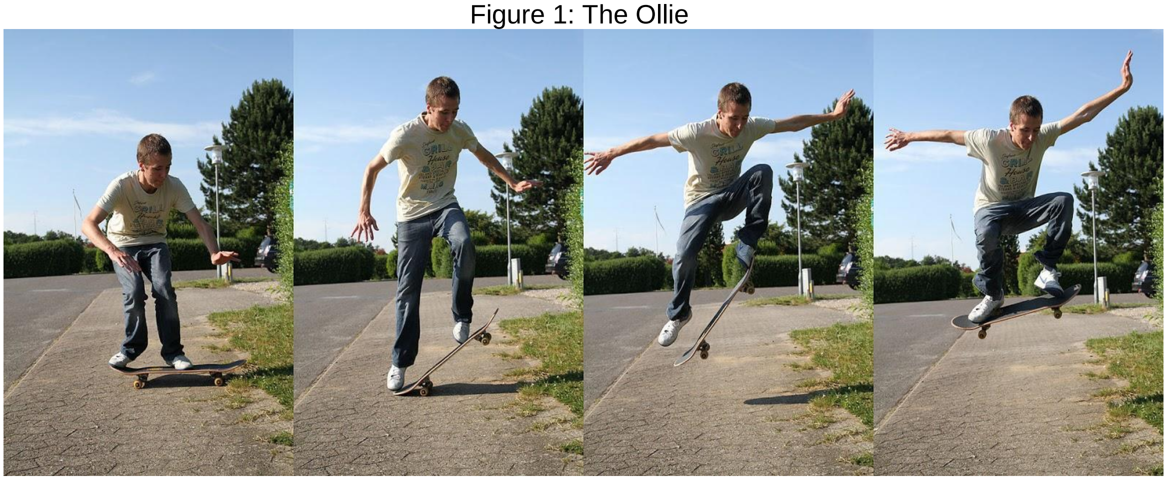
2.2 Skateboarding Trick Classifications

There are seven basic skateboarding maneuvers, referred to as tricks, performed on flat ground. Nearly all of the advanced tricks are combinations of the basics. To deliver a product and meet user satisfaction it is important that engineers understand the activity that they are developing a product for. We wish to be able to classify the basic tricks explained below.

Ollie

As shown in Figure 1 below, the rider stomps down on the tail of the skateboard. When the tail hits the ground the board experiences a reactive force upwards. This propels the board into the air. The rider must jump synchronously with the skateboard and use the other foot to level the board out so that it is parallel with the ground and then land. In terms of sensor readings, the Ollie is just acceleration in the positive z-axis with no rotation.

Figure 1: The Ollie



Used under Creative Commons Attribution-ShareAlike 3.0 Unported license

Backside Shuv-It

The rider stomps the tail of the skateboard while scooping it behind them. This causes a reactive force that propels the board into a 180 degree rotation in front of them. The sensor would read acceleration in the positive z-axis with a rotation along the vertical axis.

Frontside Shuv-It

Similar to the backside shuv-it except that the scoop is done forward and the board spins 180 degrees behind the rider. The sensor would read acceleration in the positive z-axis with a negative rotation along the vertical axis.

Backside 180

Similar to the Ollie, except the rider spins their shoulders 180 degrees which causes the board to turn 180 degrees along with the rider while moving backwards relative to the axis of motion. The sensor would read acceleration in the z-axis with a rotation along the vertical axis just like the backside shuv-it which would make it tough to distinguish.

Frontside 180

Similar to the backside 180 except that the shoulders are spun the other way so that the rider moves forwards relative to the axis of motion. The sensor would read positive acceleration in the z-axis with negative rotation along the vertical axis. These are the same readings as the frontside shuv-it so we are faced with the same challenge as before.

Kickflip

As shown in Figure 2 below, the kickflip is similar to the Ollie, except that the front foot now applies a flick of the toe on the edge of the skateboard that causes it to flip 360 degrees on it's side. The sensor would detect acceleration in the z-axis, no rotation along the vertical axis, and a positive rotation along the horizontal axis.

Figure 2: The Kickflip

Used with permission from Skatepark of Tampa

Heelflip

Similar to the Kickflip except the flick is caused by the heel and the board rotates the other direction 360 degrees on it's side. The sensor would detect acceleration in the z-axis, no rotation along the vertical axis, and a negative rotation along the horizontal axis.

Advanced Tricks

If time permits, the group can also classify the advanced skateboarding tricks described below:

Varial kickflip

A combination of the kickflip and backside pop shuvit. The board spins 180 degrees forwards along the vertical axis while it also flips a full rotation clockwise about the vertical axis.

Hardflip

A combination of the kickflip and frontside pop shuvit. The board spins 180 degrees backwards along the vertical axis while it also flips a full rotation clockwise about the vertical axis.

Varial Heelflip

A combination of the heelflip and frontside pop shuvit. The board spins 180 degrees backwards along the vertical axis while it also flips a full rotation counterclockwise about the vertical axis.

Inward Heelflip

A combination of the heelflip and backside pop shuvit. The board spins 180 degrees forwards along the vertical axis while it also flips a full rotation counterclockwise about the vertical axis.

360 Shuv It

A pop shuvit performed with a 360 degree spin about the vertical axis instead of 180 degrees. The spin can be forwards or backwards.

360 Flip

A combination of the 360 Shuv It and the kickflip. The board spins 360 degrees about the vertical axis forwards, while the board flips a full rotation clockwise along the horizontal axis.

Laser Flip

A combination of the 360 Shuv It and the heelflip. The board spins 360 degrees about the vertical axis backwards, while the board flips a full rotation counterclockwise about the horizontal axis.

Bigspin

A combination of the 180 and the shuv it. The board spins 360 degrees either forwards or backwards along the vertical axis and the rider spins 180 degrees in the same direction.

Bigflip

A combination of the Bigspin and a kickflip. The board spins 360 degrees either forwards or backwards along the vertical axis with the rider also spinning 180 degrees in the same direction and the board also does a full rotation along the horizontal axis.

2.3 Project Goals

Main Goal:

With this project we seek to achieve many different goals. However, there is one main, underlying goal that we visualize with the SMART Skateboard that is imperative towards the success of our senior design team. That goal is to have a fully functioning electronic device, comprised of a main processing chip with sensors and regulators implemented into it, that can be mounted onto a skateboard, that can assess motion, altitude and acceleration whilst communicating with a mobile phone by means of an application we develop. This main goal encompasses the entire focus of our senior design project. Having met this goal essentially ensures the success of the entire project.

We cannot achieve any of our subsequent goals without accomplishing the main focus of this design. All of the subsequent goals of this project essentially break down the main goal into smaller, manageable goals that we can ultimately accomplish one after the other. In the following text, we will review some of the subsequent goals needed to realize the main goal of this project. Please note that we dissect these objectives, and more, in the Requirements and Specifications section (Section 2.4) of this report document.

Subsequent Goals:

Some other goals we are focused on include being able to correctly identify each skateboarding maneuver using the sensors on the device, writing embedded and mobile app codes that can fully assess the motion detected by the sensors, creating a device holster that can withstand the regular wear and tear most skateboards are accustomed to, developing a communication method that can minimize latency between the device and mobile phone, and creating an overall design that is lightweight and relatively unnoticeable to the user without compromising the performance of the device.

Identifying each trick

The importance of this goal needs little explanation. The entire project is focused on creating an enjoyable experience for the user. If the device fails to correctly identify each trick he/she accomplishes, then the entire project is considered a failure. This goal, just as all of the ones following it, is crucial to the success of our senior design team.

Writing code that interprets the sensors

The SMART Skateboard would be rendered useless if it could not precisely identify the motion being detected by the sensors. This is a crucial objective towards being able to fully identify each trick. The sensors are developed to measure spatial coordinates and movement to an uncanny level of precision. The responsibility, therefore, lies on our software development team to create a code that can fully comprehend the information attained by these sensors and in turn create meaningful results for the user.

Developing a reliable holster

This goal is discussed in greater detail in the research section (Section 3) of the document. Ultimately, the focus here is the fact that the entire design is relying on a stable holster. Without accomplishing this goal, our entire project could break after just the first round of testing.

Creating communication with minimal latency

This goal seems a bit obvious, however it is essential to the success of the design. The user need to know as quickly as possible what he/she has accomplished. Having a large latency, or delay, would jeopardize the effectiveness of the device and could ultimately result in a failed product. Having minimum latency means that our code needs to be as small as possible without compromising the functionality of the device.

Building a subtle overall SMART device

These are the main objectives of our senior design group. We feel that if we accomplish these goals, our design project will be a success. We also have some other goals in mind if we can finish the entire project ahead of the deadline, but those will be discussed in a later section in this report.

2.4 Motivations

The group knew from the beginning that it was important to pick a fun and interesting idea for Senior Design so that our passion for it would help us through the work. Group member Taymas has been skateboarding for almost 10 years and has always wanted to apply his Electrical and Computer engineering knowledge to the world of skateboarding. Taymas pitched the idea to the group and we knew that it was a perfect fit for our team. A lot of work has already been done to electrically power skateboards, but not so much has been done in terms of classification and analytics. We knew this project would keep us motivated and it was also firmly within the scope of the Senior Design requirements.

This project spans multiple areas of Electrical and Computer Engineering, such as embedded systems, power, MEMS devices, PCB design, mobile application design, wireless communication, and data classification. Embedded systems have long been an important field of electrical engineering and it will continue to be. This project will build the team's experience in embedded systems and make us competitive in the job market.

Power has also long been a dominant field in electrical engineering and it is important for emerging electrical engineers to have real life experience in powering components and systems. MEMS devices is a field where our group has little experience in so learning more about these kinds of devices such as Gyroscopes will help grow and round out our electrical engineering knowledge. PCB design experience is extremely important and unfortunately an area that we have not been taught in our electrical engineering education. Thankfully, PCB design is a requirement for Senior Design so we will learn all about it and be prepared to enter the professional electrical engineering field.

Now mobile application design has taken the software world by storm and it is now an expectation of every successful venture to have a mobile application. This project will give us an opportunity to work with a real time user interface and develop these skills that are so crucial in today's digital world. Wireless communication will also grow in necessity in the future where the internet of things could transform the world as we know it. The future home can have dozens of small computers in every application such as toasters and microwaves that all communicate with each other to provide the user with the optimal experience.

Finally, in a world filled with vast amounts of data it is crucial to be able to process and make sense of that data or else it is rendered useless. This project will give us experience in taking real world data and transforming it into useful feedback for the end user. Also, this project will teach us how to implement multiple ideas into a concrete design.

2.5 Requirements & Specifications

The following section breaks down the different market requirements along with the corresponding engineering specifications that our team must abide by during this project. The market requirements essentially focus on the user’s end of the design, keeping the details to a minimum. Meanwhile the engineering specifications are much more detail oriented. These specifications are the backbone to the SMART Skateboard. The only way our project will be a success is by meeting each and every one of these requirements and specifications. The team understands that this is one of the most important sections to keep track of during the creation of the SMART Skateboard device.

2.5.1 Market Requirements

Weight

This is an important requirement for any wearable device, especially a skateboard. Too much weight will put extra strain on the user and lower their enjoyment of the product. The weight can negatively affect the physics of skateboarding, making it harder for the user to flip the board or to soar in the air. This is why it is important to market the product with the minimum weight possible.

Battery Life

This is also important for any embedded device, especially wearable ones. If the battery life is too short, the user will be burdened with frequent charge sessions and they will be unable to skateboard as long as they want. The battery could also drain just as they are performing their best maneuver and the device would fail to capture it. This is why it is important to market the product with the maximum battery life possible.

Durability

Being attached to the bottom of a skateboard will subject the device to unpleasant impact forces. Skateboards can last many months for a casual user, but as little as several days for a professional. Shown in Figure 3 on the next page is what typical wear and tear of a skateboard is like. It's clear that there are obvious high stress locations on a skateboard that the device should not be attached to if the user has high capability on the board. For these reasons, the device should be marketed as durable as possible.

Figure 3: Skateboard wear and tear

Permission pending from Oasis Skateboard Factory

Cost

It is important to market the product at as little cost as possible because a typical price for today's skateboards is about $150. Also, the skateboarding market is predominantly teenagers and their parents. Skateboarding has typically been a middle class hobby throughout the years so money is not abundant in this market.

User Interface

The product will be marketed with a real time user interface so that the user will receive instant feedback for the tricks they perform. The results should come in quickly because the user would hate to wait too long for the result and grow impatient with the product.

Skateboarding is a fast paced action sport where many tricks can happen in a short span of time depending on the speed of the rider so the user interface should be made to keep up with these demands.

Basic Flatground Trick Classification

The minimum market requirement is that the device can distinguish the basic flatground skateboarding tricks between each other. These are the tricks that almost all skateboarders learn because they are the most accessible. Basic flatground tricks do not require any special ramps or terrain so all skaters end up learning and practicing these tricks at home.

Wireless Range

The range should be as long as possible because skateboarding is a very mobile sport. Skateboarding cannot be confined to several feet because skateboarders see the whole world as their playground. The user should be able to have a friend hold the mobile device while the rider performs tricks.

2.5.2 Engineering Specifications

Below we have outlined the exact specifications by which we will be conducting this design. All of the subsequent research, designs, and implementations will have these specifications in mind.

Weight

The total weight of the product should not exceed 1 pound as the absolute maximum. This specification will ensure that the attached product will not interfere with the user’s ability to perform skateboarding tricks. We chose to keep the design under a pound to ensure that we had a bit of freedom in developing the holster for the SMART Skateboard.

Battery Life

The device must operate for a minimum of 5 hours starting at a full charge before being depleted. A typical LiPo battery has 2000 mAh of life. To achieve a 5 hour charge, our PCB should not draw more than 400 mA. This can allow the user to use the device for a full skateboarding session without having to find a power source to recharge the SMART Skateboard.

Durability

To achieve our market requirement of high durability, a good objective test is to subject the product to realistic pressures and see how it performs. The product should be able to withstand a mild session of skateboarding without falling off or malfunctioning. It should also be able to withstand a free fall of a couple feet while attached to the skateboard.

Cost

To meet our market requirement of low cost and ensure that the device is made within a fair and realistic price range, it should not cost the group more than $250 to design and develop the first prototype of the device. This would create a reasonable foundation for developing the right price range if this device were to ever hit the market. Cost is one of the most important criteria that we must focus on to create a marketable product in the SMART Skateboard device.

User Interface

To meet the market requirement for a robust user interface, our group needs to develop a mobile application on one of the 3 main brands of smartphones. The user interface must not only be robust and easy to use; it must also be aesthetically pleasing to the user. This ensures that the user will keep using the product over and over again after the initial download.

The user interface needs to have Bluetooth compatibility to take in data from the device. The user interface needs to clearly display results to the user. It needs to have menus that are standard in mobile application development. It needs to be developed with all best practices of software development in mind. It should display the results to the user at a maximum time of 5 seconds.

Trick Classification

To fulfill our market requirement of basic flatground trick classification, our sensors must be accurate within 10% to distinguish between full rotations along the horizontal axis and 180 or 360 rotations along the vertical axis. This ensures precision in identifying each flatground trick without any informational errors and, most importantly, maintains the user’s satisfaction. Without trick classifications being properly implemented, the SMART Skateboard will be rendered useless to the user and will ultimately be considered a failure.

Wireless Range

To make the device friendly to the user and permit the free roaming of skateboarding, the wireless range of the device should be at least 6 feet. This will enable the user to never lose connection with his/her device as long as the user is standing on the board or is near the skateboard. This allows the skater to fall off the board or give his/her phone to a friend to record the tricks that they attempt. This is yet another crucial requirement for the implementation of the SMART Skateboard.

2.6 Tradeoff Matrix

The house of quality table on the next page shows the correlation between each of the marketing requirements (left) and all of our engineering specifications (top) of our senior design project.

Table 1 - House of Quality Tradeoff Matrix

			1.)	2.)	3.)	4.)	5.)	6.)	7.)
			Power Consumed	Weight	Wireless Range	Sensor Accuracy	Wireless Latency	Cost	Impact Resistance
			−	−	+	+	−	−	+
1.)	Weight	−		↑				↓	↓
2.)	Battery Life	+	↑		↓	↓	↑	↓	
3.)	Durability	+		↓		↑		↓	↑
4.)	Cost	−	↓	↓	↓	↓	↓	↑	↓
5.)	Real Time Results	+			↑	↑	↑	↓	
6.)	Trick Classification	+			↓	↑		↓	
7.)	Range	+	↓		↑	↓	↓	↓	

Here are the **seven marketing requirements** we implemented into the matrix:

1. **Weight**: it is crucial to minimize the total weight of this device.
2. **Battery Life**: this category, as mentioned earlier, needs to be maximized to allow optimal usage.
3. **Durability**: the durability of this device is also critical as far as marketing the device towards the user.
4. **Cost**: the cost of the device needs to be as small as possible.
5. **Real-Time Results**: latency is defined as the time or delay between the analog signal and the final outcome on the user's application. This category should be minimized to ensure the least amount of delay or "lag" for the user. Responsiveness is crucial.
6. **User Satisfaction / Trick Classification**: this is the most crucial marketing specific. If the user is not maximally satisfied, the product fails in the market. Everything we create must fully satisfy users.
7. **Communication Range**: the effective range between the device and the mobile phone should be maximized. This is due to the fact that sometimes phones fall out of pockets, or a peer is holding the device while the user is skateboarding.

Here are the **seven engineering specifications** we implemented into the matrix:

1. **Battery Life**: our device does not look to deliver any power to anything except for the PCB and its sensors. Therefore, we want to be able to minimize the amount of power being drawn by the device itself while maximizing the total battery life.
2. **Weight**: we want to be able to minimize the total weight of this device. To be more specific, we want to keep the weight under 32 ounces. This will ensure that the device will be unnoticeable to the user while they use the SMART Skateboard.
3. **Wireless Communication**: the range of the wireless communication technology need to be as high as possible. (minimum 6 feet) This range must be achieved to create a sustainable product for the user. The user will not want his mobile application being inhibited by the sheer fact that his phone went out of range.
4. **Accuracy**: The accuracy of the sensors should be maximized. We want to keep the error rate of the sensors under 5%. This one is obvious, we want the device to be able to relay the information to the user as accurately as possible.
5. **Latency**: the latency of the device needs to be minimized. This is defined as the delay between the action and the device's reaction. To test this, we want to have the delay between a completed trick and the display on the device be a maximum of 5 seconds.
6. **Cost**: The overall cost of the device needs to be minimized. (under $250)
7. **Impact Resistance**: this category needs to be maximized as well. To test the durability of the device, we will drop the skateboard from a 5 foot free fall to show that the electronics were not negatively affected.

The +/- besides each category indicates whether we are trying to maximize or minimize the respective category for optimal design.

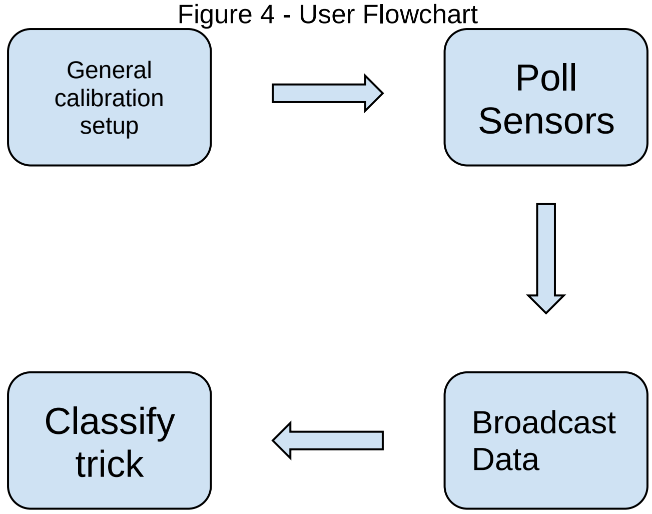
The upward green arrows within the matrix indicate that the two categories being represented have a direct positive correlation, meaning the optimization of one category immediately leads to the optimization of the other. (ex: A lower latency is directly connected to a greater quality in wireless communication technology)

The downward red arrows within the matrix indicate that the two categories being represented are inversely related, meaning that optimization in one area would immediately lead to a decrease in quality in the other. (ex: The more durable we make the device, the larger the device will be and the greater the overall cost)

2.7 User Flowchart

Figure 4 below shows the user flowchart which is how the user will interact with the SMART Skateboard from initial calibrations to actual use. Below the flowchart we break down all four steps of this process in full detail to give our team an idea of what we are trying to accomplish.

Figure 4 - User Flowchart



Step 1
When the user first opens the application, he/she will be asked to input general settings, such as stance preference. The device will also use the beginning elevation as the baseline elevation for the application.

Step 2
Sensors will track skateboard's orientation at all times. If the power is on, the device will be constantly tracking the board's orientation and acceleration through the integrated sensors implemented by the senior design team.

Step 3
Device will transmit sensor readings to the mobile device. If START SESSION is toggled ON from the mobile phone, the information collected by the electronic device will be constantly communicated to the mobile phone via the wireless communication chip chosen.

Step 4
SMART Skateboard device classifies trick performed and then the results are displayed onto the mobile device. If sensor readings correspond to maneuver known to algorithm, the trick will be properly displayed. If the maneuver is not recognized, the mobile device will state either ERROR or Unrecognized Trick.

3.0 Research

This entire section is dedicated to research that will ultimately guide us to creating a successful final device. This section is headlined by digging into similar projects and companies that have created similar products as the SMART Skateboard. Then, further research is done on relevant hardware components and materials that will aid in the physical implementation of the SMART Skateboard. These include finding the most optimal sensors for creating the device, the best power supply configuration for the device, the easiest and most efficient method of holding the device in place. This hardware research is soon followed by the software research section of the project, which inspects different options and methods for selecting the proper embedded architecture, mobile devices, and classifications of skateboarding tricks. All of the research will detail the different methods of selection alongside the pros and cons of each method over the rest. The final selections for all of our components and software will be officially announced in the design portion of this document.

3.1 Similar Projects

Syrmo

Syrmo was an ambitious project launched by a group of Argentinian skateboarders on Kickstarter in 2014. They sought to offer a lightweight and portable product that would attach to any skateboard and collect data. It would have an Android and iOS application to receive data transmitted over Bluetooth.

On the application it would have a 3D animation to replay which trick was performed. The user would be able to share this on multiple social media platforms such as YouTube and Facebook. Syrmo’s design would have a geolocation system to know where each trick is performed and for the ability to share new locations with your friends. This is a much more in-depth version of what we are trying to accomplish with the SMART Skateboard.

Syrmo took to Kickstarter to raise money for mass production, but the campaign was cancelled with only $7,164 out of $40,000 raised. As of July 2016, the product is still yet to reach the market with the Syrmo websites only accepting pre-orders.

We believe that the Syrmo project was too ambitious and overestimated the market demand for a skateboarding sensor. Our Smart Skateboard project is different because our goal is not mass production. Our goal is simply to apply our Electrical and Computer Engineering skills to an interesting project to grow our knowledge.

Trace

Trace was another ambitious wearable action sports project launched on Kickstarter. Unlike Syrmo, Trace would track and provide analytics for surfing and snowboarding as well as skateboarding. This also meant that the way the device attaches to the board was not optimized for skateboarding so that it would be compatible for surfboards and snowboards as well. The team also had more experience than Syrmo, with a PhD and pioneer of GPS tracking systems, Dr. Lokshin, as their CEO.

Trace also turned to Kickstarter to fund their campaign for mass production and they actually reached their goal, raising $161,260 out of $150,000. They are currently selling the product for $199 on their website and on Amazon but curiously, it no longer offers support for skateboarding.

3.2 Relevant Hardware Research

This section covers the relevant hardware components that will be necessary towards the completion of the design of the SMART Skateboard. All of the different options for each component will be discussed as well as the advantage of having said component in the SMART Skateboard design. The hardware research has figures and tables with proper annotations to help clarify ideas and concepts. Please note that the final design decisions, although they may be discussed here, will not be officially made until the design portion of this document. Although these decisions will be announced later, it will seem very obvious what the right decision will be after all of the research is properly presented.

3.2.1 Embedded Device Options

A microprocessor is crucial to the success of this project to measure output from the sensors and to control what to send over Bluetooth and when. There are various microprocessors on the market with different prices and functionality. It is important to compare the I/O provided by the microcontrollers, the cost, the power consumption, and what resources are available in the community to aid development. Below we compare several different microcontrollers.

Arduino Mega

The Arduino Mega is based on the ATMega2560 microprocessor chip. It features 54 digital I/O, 16 analog inputs, operating voltage of 5 volts, and 16 MHz clock speed. The development environment can be any editor that allows uploading to Arduino. The language is C++ with libraries that make it feel like Java as well.

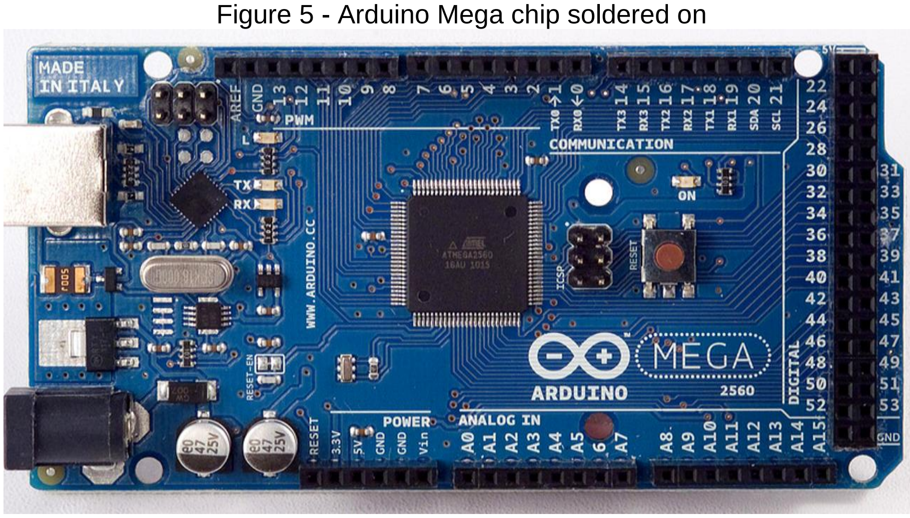
Advantages:

- Abundant digital I/O.
- 3 RX and TX communication lines.

Disadvantages:

- The processor chip is soldered on so it is tough to detach, shown in Figure 5 below. This mean we would need to buy an additional processor chip and burn code to it through a bootloader. This was deemed as a risky and inconvenient way to design our PCB.

Figure 5 - Arduino Mega chip soldered on



Permission pending from Arduino.cc

Arduino Uno

The Arduino Uno is based on the ATMega328P microprocessor chip. It features 14 digital I/O pins, 6 analog inputs, operating voltage of 5 volts, and a 16MHz clock speed. It was the very first release of the Arduino and therefore it is the simplest model available. The development environment and programming language is exactly the same as the Arduino Mega.

Advantages:

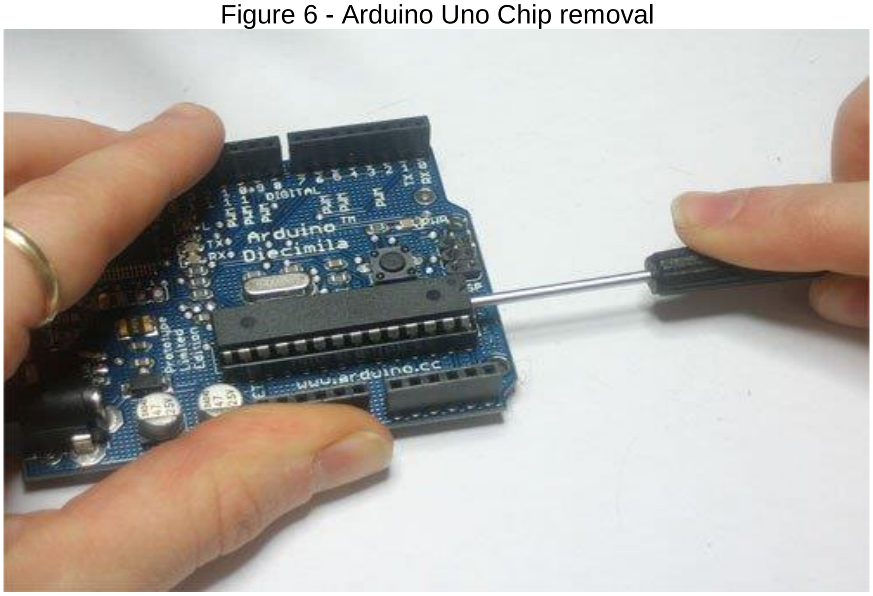
- The processor chip can easily be detached from the board. This allows us to upload code to it and then detach to a breadboard. In the final design we can simply solder the same chip to our PCB. As shown in figure below.
- The Arduino community has much more documentation and open source code for the Arduino Uno because it has been in production longer than the Arduino Mega.
- Lightweight and simple. Only needs a few lines of code to run properly.

Disadvantages:

- Much less digital I/O pins than Arduino Mega.
- Only one RX and TX communication line.
- Low memory. 32 KB flash.

Below, we show Figure 6, provided by Adafruit, of the correct method of removing the main chip from the Arduino Uno. This is a great reference for us to use when we remove the main chip from the Uno to create our prototype.

Figure 6 - Arduino Uno Chip removal



Permission pending from Adafruit

Texas Instruments MSP430

The MSP430 is Texas Instrument's main microcontroller that they produce. The MSP430 is based on a 16-bit CPU and are known for their low cost and low power. The development environment is done in TI's Code Composer Studio in assembly or C language.

Advantages

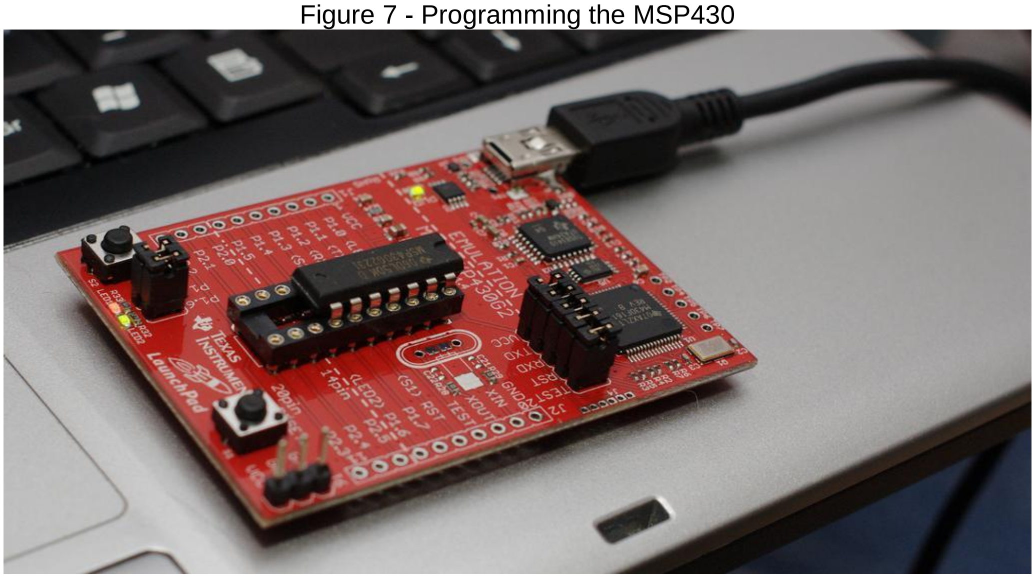
- The MSP430 is cheap, costing $9.99 on a popular distributor such as Mouser Electronics.
- There is no need for an external crystal because the MSP430 chip has a built in clock.
- Processing power and power consumption is efficient.
- The MSP430 is used and taught in multiple classes at UCF so the group is very familiar with the MSP430 architecture and how to program it.
- Easy to program. You just need a usb connection to a computer. Shown in figure below.

Disadvantages

- Another negative is that MSP430 development is limited to TI's Code Composer Studio, while the options for Arduino are much more varied.
- The MSP430 doesn't have a large open source community to get a jump start on development or to seek help when something is not working correctly.

Below in Figure 7 we show the configuration of how to implement our code into the MSP430. This figure shows the relative size of the device as well as how to connect it to the computer via USB to allow for the transfer of information.

Figure 7 - Programming the MSP430



Permission pending from Tested.com

Raspberry Pi

The Raspberry Pi is by far the most powerful microcontroller of all of the options. It's CPU speed ranges from 700 Mhz to 1.2 GHz. The on board memory can have up to 1 GB of RAM. It is similar to a Linux environment and the programming language of choice is Python.

Advantages:

- The Raspberry Pi is a general purpose computer.
- Being a computer gives it the ability for multiprogramming and makes the Raspberry Pi powerful enough to act as a light traffic server for web traffic.
- The raspberry Pi has all of the modern usb, ethernet, and SD ports as shown below.

Disadvantages:

- The Raspberry Pi is one of the most expensive microcontrollers on the market, costing around $40.
- Also, the Raspberry Pi can't run X86 operating systems.
- Thus, some Windows and Linux distributions are not compatible with the Raspberry Pi.
- The Raspberry Pi also performs poorly on heavily CPU bound processes. The Raspberry Pi runs Linux. The group is not familiar with Linux so it will be a challenge learning the Linux operating system.
- The Raspberry Pi also runs Python. The group hasn't had much experience in Python so learning the Python language would be a further challenge.

Below, we show Figure 8. This image shows the different power supply connections on the Raspberry Pi, which is an advantage over the other options.

Figure 8 - The Raspberry Pi's many functionalities

Permission pending from RaspberryPi.org

Summary

The group considered several microcontrollers to be implemented in our project. Table 2 on the next page summarizes the advantages and disadvantages of each microcontroller we considered. It was important for the group to select a microcontroller at the right price, with the right power, and the right amount of open source materials available online to help us jump start our design

Table 2 - Comparison of Embedded microcontrollers

Comparison of Microcontrollers Summary		
Brand	**Advantages**	**Disadvantages**
Arduino Mega	● Abundant digital I/O ● Rich open source community	● Cost ● Excess pins ● Hard to deconstruct into breadboard
Arduino Uno	● Ideal number of I/O ● Rich open source community ● Sample code readily available ● Easy to deconstruct into breadboard	● Low memory
TI MSP430	● Cost ● Taught at UCF	● No sample code readily available
Raspberry Pi	● Powerful ● Rich open source community ● Large memory	● Cost ● Unnecessary components

3.2.2 Embedded Power Connection Options

Our customized PCB has many different possibilities as far as power connection goes. In the following section, we will break down every possibility and the advantages and disadvantages of each as they pertain to our specific senior design project. With our prototype Arduino board, there are three main methods to connect the PCB.

The following is a systematic break down each of these three methods to use as a reference when constructing our SMART Skateboard. Each of the three options will be highlighted by the respective advantages and disadvantages as they pertain specifically to the SMART Skateboard. Please note, that the final selection for the power connection option will be discussed in the Design section of this document. This was one of the most important sections of hardware research, due to the fact that it is the foundation of the design process. The following section breaks down the three main connection methods associated with the Arduino printed circuit board.

Connection Method #1: Barrel Jack/ Plug (AKA: Japan Jack)

The Arduino has a 5.5mm/2.1mm DC barrel plug, which details a possible input of 5V - 20V at a current rating between 0.5A - 2A. On the Arduino, this connector lies on the edge of the board to disallow interference between the power system and the PCB itself. This barrel can be easily installed in our customizable PCB using EagleCAD in the second half of this project.

Advantages

It seems as though this is the common barrel size for almost all PCBs on the current market. This means using this connection method on our Arduino prototype would allow for a realistic model of the final product in terms of hardware. It also means that finding an adapter for the battery to connect to the barrel plug would be a relatively hassle-free process. The fact that most PCBs have this connector plug can ensure that our prototype assembly is not going to waste. This can allow us to spend more time on the other, more time-consuming aspects of the project.

Another key advantage to using this connector port is the fact that the power input would pass through the voltage regulators set up on the PCB. If there was any noise or sudden increase in voltage due to the power supply chosen, the voltage regulators built into the Arduino would therefore null the negative effects. This in turn would somewhat protect the entire system from ESD (Electrostatic Discharge).

The current rating for the barrel jack seems to be yet another advantage over the other options. The barrel jack supports currents ranging from 500mA to 2A, with a minimum requirement of 250mA to power the Arduino with ensured stability. We will discuss the other methods' current ratings in the following sections, however, after some rudimentary comparisons, it seems as though the barrel jack has the most promising current output ratings. This is important to the SMART Skateboard due to the fact that our final PCB will have up to three spacial sensors and a wireless chip that will all need to consume a significant amount of current to run correctly and efficiently. However, since our design has so many other components that need to be powered, it might prove to be more beneficial and more stable to just power the components individually with another external power supply and let the PCB be powered by whichever power supply system we ultimately deem fit.

Disadvantages

This method doesn't come without some disadvantages, unfortunately. The connector barrel (or barrel jack) is detailed to have the ability to power the PCB anywhere from 5V - 20V. However, after doing some research, there is an issue with this voltage range. Apparently, the Arduino will malfunction quite frequently if the input voltage dips below 7 Volts. This is a major problem for our purposes. Due to the fact that our project cannot be plugged into a wall or large car battery, we have to use smaller mobile batteries such as AA or D for example.

The issue arises when we realize that almost all mobile batteries on the market peak at around 3 Volts. This one disadvantage cuts down our possible powering options by almost 100%. At first glance, it seems as if our only options left would be to use a car battery, or a 9 Volt Alkaline/Zinc-Carbon battery. We will discuss the different battery types and their implications on the SMART Skateboard in the next subsection.

Also, the Arduino we use as a prototype will have some internal issues if the power supply reaches the detailed maximum of 20 Volts. If that is the case, the internal voltage regulators will run at full capacity. This will, in turn, cause any extra voltage to be dissipated as heat. Ultimately, this heat can damage the PCB and cause overheating. Multiple online sources state that the most optimal range for the barrel jack connector is between 9V-12V, which is slightly different than the original range we had assumed of 5V - 20V. In conclusion, the main problem with this connection setup is the fact that most mobile batteries wouldn't supply enough voltage to power the PCB.

There are fixes for this main problem however, an example would be to just line up multiple batteries in a series connection to scale-up the input voltage, which would essentially solve the minimum voltage issue but create some space/ logistics problems. This ultimately is a good option for the SMART Skateboard device, but will be difficult to implement in terms of the voltage necessary to power up the board.

Connection Method #2 USB Port

The Arduino PCB can also be powered up by the USB port attachment located right above to the barrel jack (pictured below). The USB is detailed to require 5V at 500mA to power the Arduino. Powering the Arduino using only the USB essentially transposes the voltage directly to the 5V rail on the PCB. This method is similar to the barrel jack in that it is a universal port that almost every type of electronic device can be adapted to. Since the Arduino is the PCB we use as our prototype device, it is the main focus of the hardware technologies we discuss. Ultimately, we will use the data we gather on the Arduino as the foundation to designing our own PCB through EagleCAD.

Advantages

The Arduino has many perks to using the USB connection method. As discussed above, the USB is a universal connection port (Universal Serial Bus). It is a cable system that encompasses almost every connection and communication protocol. This, in turn, makes it easy to find compatible cables and adapters to connect the power supply to the PCB. This great for the SMART Skateboard since USB cables come in many different styles and lengths, and the overall size of our device is crucial to its effectiveness.

Another big advantage to using a USB connection is the fact that it requires only 5 Volts of input. This is easier to achieve than the recommended 9V-12V range specified by the barrel jack in the previous breakdown. This advantage keeps multiple battery types a possibility for ultimately powering the PCB.

The biggest advantage to using the USB connection method is the fact that it allows the user to also use the Japan Jack method. The Arduino can detect whether or not there is a barrel source connected along with the USB supply source. The Arduino will automatically switch to the barrel voltage, as long as a minimum of 6.6V is connected to the PCB. So there is a large possibility of us being able to power the PCB using both Methods 1 and 2.

Disadvantages

There are a few issues with using a USB to power the Arduino and ultimately our designed PCB. One of the disadvantages was just discussed as an advantage over the Japan Jack method. The USB only needing 5V to power up seems as an initial advantage, however the problem is that this method is strictly restricted to 5V, meaning any deviation higher or lower than the required 5V will cause issues in the hardware of the PCB. The main reason this is a problem is due to the fact that the input voltage bypasses the voltage regulators built into the Arduino, and directly powers the 5V rail voltage. If the power supplied to the USB reaches 6V of input, the 5V voltage regulators are bypassed, causing damage to the PCB as well as each of the components that use the same power supply.

Another possible disadvantage to using just the USB connection method is the fact the port has an overcurrent protector. If there is anything on the board that uses more than 500mA of current, or if there are multiple components that draw more than 500mA combined, the board's overcurrent protectors (polyfuses) will be triggered, and the board will not work properly and most likely begin repeatedly resetting. Regardless of what the supply power current is detailed to be, say 2A, the USB port will limit the current to 500mA. This could be a major issue when implementing multiple sensors and wireless communications to the PCB.

If all of our sensor, in addition to our wireless communication, aggregate to over 500mA of required current, then the SMART Skateboard will not be able to properly operate. This is the biggest issue when dealing with the USB connection port. If we do decide to use this connection method, it is more than likely that it will have to be coupled with the barrel jack method to allow some leeway in the amount of input power.

Method #3 I/O pins

The Arduino prototype board we are developing has a couple of input/output pins that can also be used to power up the PCB. These pins can either be configured as inputs or outputs depending on the power system being supplied to the Arduino. The first of the three power pins is 3.3V. This pin actually cannot be used as an input. The 3.3V is directly connected to the 5V pin, rendering it useless as far as input voltage to power the PCB goes.

The next of the pins is the 5V designation. In case there is no power being supplied from the barrel jack or the USB, the 5V socket can be used directly to power the Arduino. The only requirement is that there needs to be a regulated, stable 5V source being inputted into the pin.

There are some online sources that state there might be an issue using this method in that the regulators don't "like" voltage being applied to the output pins, however it turns out that the Arduino is developed to accept input voltages. This is something we need to keep in mind when designing the final PCB for the SMART Skateboard.

The Vin socket is yet another dual function pin. The input function, used for powering the PCB, is not protected by inversions in polarity. The input voltage goes directly into the regulator. This voltage also goes below the barrel plug's diode. This means that no input should be applied to the barrel plug, this would in turn cause some serious issues in the PCB such as ESD and component damage. Since the polarity has to match the PCB, the negative pole of the input voltage is located on the GND pin.

Advantages

After some investigation, it seems as though using the pins to input power shows to have very few advantages over the other two methods (Japan Jack & USB). The most notable positive aspect of this method is similar to that of the USB. The pins require much less input voltage than the Japan Jack (5V in comparison to 9V - 12V). This means that there is more space in the overall device due to the fact that the amount of batteries used will be significantly less than using the Japan Jack alone. Another slight advantage to using this method over the others is the relative ease of application. To get power in through the pins on the PCB, all that is required is attaching/soldering a couple of wires connected from the battery (or batteries) chosen to designated pins. This method of connection seems to be much less complex than the other possibilities, however it doesn't come without some major flaws.

Disadvantages

There are a few issues with using the I/O pins as an input powering system. One of the more obvious issues with using this method is the fact that once we decide to designate an I/O pin as an input, there is no ability to use that same pin as an output to power connected components on the PCB we develop. This is a crucial issue for our purposes, due to the fact that we need multiple components, we will need to find exterior power sources to power up our sensors and wireless communication chip.

Another disadvantage of using these pins as an input is the fact that using this method provides no protection, unlike the Japan Jack. The diode and the PTC (positive temperature coefficient device) fuse are found above the 5V socket and thus have no function when using the pins as inputs. This means that any type of disturbance in input voltage cannot be regulated, which in turns means that the possibility of component damage is quite high using this method.

The 3.3V pin not being able to be used as an input is another slight disadvantage. There is a voltage regulator right next to the 5V pin used solely for the purpose of generating a 3.3V output. If the 3.3V pin could've been used as an input, it would've been ideal for the SMART Skateboard due to the fact that most store-bought batteries output right around 3.3 Volts. However, this is not the case here, and the inability to use the pin as an input voltage is ultimately a flaw in regards to the SMART Skateboard project.

The last issue with using I/O pins as the power input is sturdiness. The pins would have to be connected to the battery via copper wires, which would make it difficult for us to ensure that the wires will stay connected throughout a SMART Skateboarding session. This could ultimately cause our product to fail, and even damage the device.

3.2.3 Power Supply Options

In addition to the power connection option of the Arduino prototype we developed, we also must consider what type of power supply the device will be using to operate. There are two major methods of powering the Arduino and most other PCBs. The first option is to use a battery and the second is to use a plug in adapter as a power source. In the following text we break down these two main distinctions into the multitude of options that lie before us. The first section in the following text talks about the different battery types and the characteristics of each of these types as they pertain to the SMART Skateboard The latter section of the following text talks about different wall plug in options and how they could possibly be implemented into the SMART Skateboard.

3.2.3a Battery Power Options

In the following section, we identify six relevant battery types as well as the implications of each type as they pertain to our project. We detail the output voltage range of each battery as well as the respective shape of each. The overall weight of the batteries is also discussed, as it plays an important role in the creation of the SMART Skateboard. After detailing these battery types, it should become clear which option will work best for our purposes.

AA, AAA, C and D shaped batteries (Alkaline/ Zinc-Carbon):

These batteries have a nominal charge of 1.5 Volts. This could prove to be ideal for the SMART Skateboard. If this is the case, we would need to combine multiple batteries in a series combination to ensure that enough voltage can be reached to power on the PCB using any of the three source powering options.

A possible issue with this battery type is the fact the final product we design as a power source might outweigh our initial weight specifications, causing the device to add significant weight to the skateboard, which in turn would affect the efficiency of the project.

Also, the fact that these batteries are unable to be recharged forces our hand in the design of the device. If we go with this option, not only will we need multiple batteries to be strung in series, but we would also need to devise a method for these batteries to be removed from the device itself.

The fact that they cannot be recharged makes the design of the holster (detailed in the subsequent subsection) a much more meticulous process. Below, is a visual example of the series connection we would require for the SMART Skateboard given 1.2 Volt AA batteries.

9V Alkaline/ Zinc-Carbon batteries:

These batteries seem to be ideal to power the barrel jack (recommended 9V - 12V). The nominal voltage outputted by these batteries is 9 Volts, meaning that we would only need one to power on our PCB. This option seems to work for the SMART Skateboard project in that it delivers just the right amount of voltage to the PCB’s Japan Jack plug.

There are, however, a couple of issues with implementing this battery type. First of all, the dimensions of the battery would make it difficult to create a compact device that creates an essentially unnoticeable experience for the user. Having a bulgy or heavy battery creates a heavy device, which completely alters the entire weight distribution of the skateboard.

The other, more significant, issue with this option is the fact that these batteries are not rechargeable. Therefore, we would need to design a holster that would allow for the battery to be replaced without damaging or altering the existing rest of the device. This is the same issue encountered with the previous battery type option.

Coin cell shaped Lithium batteries:

These coin cell batteries have a nominal output voltage of 3 Volts. This is not enough to power any of the three connection methods discussed in the previous subsection, however if placed in series orientation we could build a small enough battery pack that would power our designed PCB as well as take up a minimal amount of space to allow the holster to be as small as possible. That is the advantage of using these batteries over the previous two.

The only problem with the coin cell Lithium batteries is, once again, the inability to recharge these said batteries. This slight issue affects the overall design of our device by forcing us to be make battery replacement a possibility for the user.

Since we are trying to keep the overall device as small and simple as possible, a non-rechargeable battery seems like a nuisance which we should ultimately avoid. An ideal battery option would be to have small, rechargeable batteries, where the user can plug the device in to recharge the SMART Skateboard and there would be no need to ever replace the batteries in the device unless there was a mechanical breakdown.

Silver Flat Pack shaped LiPo batteries:

This battery type seems to make the most sense for the SMART Skateboard. It is quite useful in that it is rechargeable. This would essentially ensure that using this battery option would ensure a single purchase for the user as far as batteries go.

Also, the ability to recharge these batteries significantly improves the overall design of our project. We would not need to worry about the device having a battery slot that can allow constant battery replacement.

These batteries are quite easy to connect in series to optimize the output voltage, and they take up a minimal amount of space due to their flat shape. This makes them a very viable option for the purposes of our senior design project.

The nominal voltage for these LiPo cell batteries is 3.7 Volts. Combining just three of these in series would essentially ensure the Japan Jack/ barrel jack would receive just the right amount of input voltage to power the entire device. These batteries are flat and compact, which makes designing the packaging holster much easier on us.

These batteries are also lightweight (~ 1 gram) meaning that they will not significantly affect the overall weight of the SMART Skateboard device. This would keep the total weight of our device well under the initial predicted weight.

The only possible issue with using the LiPo battery is the fact that multiple sources claim that they are quite sensitive to tempering. This means that when a skateboarder lands a trick, the sheer G-forces may negatively affect the battery. Some instances even claim that the battery could rupture or explode. This is something that will definitely need to be tested going forward if we mean to use this as the power source for the SMART Skateboard.

Looking ahead towards the design aspects of this project, the LiPo flat pack battery style seems to make the most sense. It's minimal design, light weight, wide range of voltage outputs and rechargeable capabilities all make it a very viable option to power our SMART Skateboard device.

AA, AAA, C, D shaped rechargeable (NiMH/ NiCd) batteries

These batteries types are quite similar to the silver flat pack batteries in that they are also rechargeable, meaning constant replacement is not a significant issue for the user. This is a major advantage over the first three options detailed above.

Also these batteries can be easily placed into a series combination. This method will almost certainly be needed to optimize the output voltage driving the PCB. This characteristic is quite similar to the LiPi flat pack batteries detailed earlier.

However, after some deeper inspection, there are some unfortunate significant differences between these NiMH/ NiCd rechargeable batteries and the LiPo flat pack rechargeable batteries. These NiMH/ NiCd battery types are restricted to the standard typical battery shapes we see at most supermarkets. This restriction makes designing the overall device to be as small as possible (and essentially undetectable) a bit more difficult.

The dimensional specifications we outlined at the beginning of the project could be a limiting factor if we decided to use these batteries to power up our PCB device. Also, a less serious dimensional problem is the overall voltage output of these batteries. The nominal voltage outputted by these batteries is around 1.2 Volts. When comparing to the standard 3.7 Volts of the LiPo flat pack batteries, this is a significant disparity between the two.

This difference ensures that we would possibly need to combine more than 7 batteries to properly power the SMART Skateboard's PCB. This is an issue that not only affects the overall dimensions of our final device, but also is an issue that affects the final economic cost of developing the SMART Skateboard. If this product were to ever be on the shelves of most general stores, the overall cost of production will be a major contribution to the price of the device. This is something we need to consider if we were to choose this option to power our device.

Car battery (Lead-Acid)

Although this sixth option is quite ridiculous in regards to our project, it actually would be very beneficial to our project were it not for the actual dimensions of car batteries. Most car batteries supply around 12.6 Volts, which is just outside the recommended range for the Arduino barrel jack, but well within the detailed specific range of the port (5V - 20V).

Also, car batteries have a quality that the first three options above cannot accomplish. They are rechargeable, which would be ideal for this project with regards to replacement of the battery itself.

However, there is a quite obvious issue when it comes to the overall weight and size of the battery. There would be no way to implement a car battery into our SMART Skateboard device without the user noticing a significant change in his/her experience. Not to mention the fact that there is essentially no spot on a skateboard that would allow for a 250cm X 175cm X 175cm battery to rest without some major conflicts.

Summary

Table 3 below summarizes the different battery power supply options we have considered and their advantages and disadvantages. After carefully analyzing the following table, it should be clear which option makes the most sense in terms of the creation of the SMART Skateboard device.

Table 3 - Battery type summary

Battery Type	**Advantages**	**Disadvantages**
AA, AAA, C and D shaped batteries (Alkaline/ Zinc-Carbon):	• Easy to configure batteries in series for optimal output voltage. • AAA sized batteries are small enough to be unnoticeable by the user	• Other than AAA, these batteries are too large to create an optimal minimal design. • These batteries are not rechargeable. • Need multiple to get the right voltage output.
9V Alkaline/ Zinc-Carbon batteries:	• Would only need one battery to properly power the barrel jack connection port.	• Heavy and bulky, making minimal design difficult. • Non-rechargeable

Table 3 (continued) – Battery type Summary

Coin cell shaped Lithium batteries:	● Small in size and weight allowing a minimal design for the device.	● Little voltage output, meaning multiple are required to create optimal power supply. ● Non-rechargeable
Silver Flat Pack shaped LiPo batteries	● Flat in shape and light, allowing minimal design. ● Wide range of output voltages, meaning only one battery is needed. ● Rechargeable	● Possibility of damage after enough endured force.
AA, AAA, C, D shaped rechargeable (NiMH/ NiCd) batteries	● AAA sized batteries are small and light enough to allow for a minimal design. ● Rechargeable	● Other sizes are too large to allow for minimal design. ● Output voltage is too low, meaning multiple batteries are required.
Car battery (Lead-Acid)	● Optimal output voltage to power the barrel jack connection	● Way too oversized and overweight to be considered an option.

3.2.3b Three Wall Plug Power Options

The Arduino doesn't only run on batteries. The Arduino and every other type of PCB also allows a wall plug-in option to power up the board. This method is essentially unfeasible when it comes to our project, since mobility is essential to the design. There would be no way for a skateboarder to ride around and attempt complex tricks while the device is plugged into a wall.

It should be noted that, although this method of powering up the PCB is not ideal for our purposes, the Arduino boards need a DC input source voltage. This requires an adapter that can convert the AC supplied by wall outlets into a constant, direct voltage to power up the device.

The only possible way to implement this type of plug-in power source into the SMART Skateboard project would be if we decided to implement rechargeable batteries. If that is the chosen route for our project, the user would need to use a wall plug to charge the device until fully charged, and then unplug to allow for free roaming. In the following text, we will briefly discuss the three major types of direct current, plug-in power supplies.

Unregulated Linear Power Supply

Although this power supply configuration is a possibility to use in most PCBs, it is not necessarily a smart idea. Unregulated power supply systems output a DC power which, at first glance, seems to be a viable option for powering up almost any PCB. However, we will detail some significant issues with this power supply system in the following text which should prove to make this method essentially unusable.

Unregulated linear power supplies are usually especially unreliable for the purposes of powering up electronic devices. This unreliability could end up damaging the SMART Skateboard device and ultimately cause our entire project to become a failure. This unreliability stems from the notion that the DC output voltage depends on an internal voltage reduction transformer, and is also related to the amount of current used by the electrical load. This means that any variations in the load could change the amount of voltage being delivered to the PCB.

These power sources also often offer no stability in their power output. The fact that these power sources are unregulated means that any type of noise or sudden rise in voltage could ultimately destroy our SMART Skateboard device. This is due to the internal structure of these devices relying heavily on a voltage reduction transformer instead of a voltage regulator, unlike the following two examples of power supplies.

After doing even a nominal amount of research, we found that unregulated power supplies are known for damaging electronic devices quite regularly. This problem alone would end up causing expenses to rise through the roof with every replacement in the SMART Skateboard device.

Ultimately, this power supply system should be avoided at all costs for almost any PCB not just the SMART Skateboard device. It will almost certainly ruin any PCB.

Regulated Linear Power Supply

A regulated linear power supply seems to be a much better option for powering up our PCB or even for simply recharging the device's batteries. These power supply systems are essentially just voltage regulators with a few more internal electronic components. These are consequently much more reliable and stable than the previously discussed unregulated linear supply.

Also, the stability of these regulators is much higher relative to the unregulated power supplies. This means that the ripple voltage encountered by the device would also be relatively miniscule.

Although this plug-in option seems to be much better than any unregulated power supplies, there is still an issue of performance. This is a major issue for our purposes. With an efficiency rating hovering between 40% - 60%, most of the power being converted by these plug-ins is usually being dissipated to the voltage regulators which are built in.

In terms of our project, this means the user will spend more time charging the SMART Skateboard device and a significant portion of that time is proven to be wasted. This is a major problem if we mean to appeal to the user in any way. The user will not want to waste extra time waiting for the device to charge, and he/she will certainly not want to know that 40% - 60% of the power used is being wasted.

The regulated linear power supply also has a small chance of harming the device connected if the inputted voltage is too low. This may not be as serious of a dilemma as we found with the unregulated supplies, however this option ultimately doesn't seem to be ideal for the purposes of the SMART Skateboard.

Switching Power Supply

This system is the most recent option as far as development goes. It is the only one that seems to have the ability to step-down voltage (meaning that the output voltage of the supply can be made less than the input voltage) and also step-up voltage (supply more than the input).

One of the biggest advantages with using this method is that, even though the voltage output is always switching, the output voltage stability of the switching power regulator is known to be very high. Also, the efficiency of this option is much better than the previous two (80% - 90%). This means that there is minimal time and power wasted when powering up the rechargeable batteries of the SMART Skateboard device.

One possible downside to choosing this option to power the PCB is directly correlated to the main characteristic of the supply. The switching nature of the supply means that there exists a ripple voltage and it is is quite high with respect to the other choices above. This could damage components within the circuitry of the Arduino or SMART Skateboard's PCB. Also the switching power supply can suffer from high frequency noise, rendering this option almost completely useless as far as powering up the actual PCB.

However, many of these switching power supplies are exclusively designed for the purpose of charging LiPo batteries. This is a huge advantage for the SMART Skateboard since LiPo batteries could very well be the battery used to power the SMART Skateboard. This essentially means that, if we chose LiPo rechargeable batteries to run our project, the switching power supply is the most viable option.

Summary

Table 4, on the next page summarizes the three wall power options we considered and their advantages and disadvantages.

Table 4 – Three Wall Power Options Comparison

Three Wall Power Options	**Advantages**	**Disadvantages**
Unregulated Power Supply	● Converts AC to DC	● Unreliable ● High instability in output voltage ● Low efficiency
Regulated Power Supply	● Converts AC to DC ● More stability in output voltage ● Low ripple voltage	● Very low efficiency, wastes around 40%-60% of the power from the plug.
Switching Power Supply	● Converts AC to DC ● More stable than the other options ● Relatively high efficiency	● Due to switching nature, high ripple voltage in output.

3.2.4 Sensors

In the following text, we will discuss research relative to the types of sensors that the SMART Skateboard device might need to have implemented within. The research revolves around studying different possibilities of sensors as well as their functionality and how they would be incorporated into the overall design of our project. Please note that the final selection for the sensors chosen will not be made completely clear until the design portion of this document.

3.2.4a Gyroscope

The gyroscope is going to be the sensor playing the main role in our project. The gyroscope is going to be implemented in order to be able to measure the orientation and rotation of the skateboard, the main factor in determining the type of trick, and the magnitude of the trick. The gyroscope provides the exact precision the user will need. For example, if the person riding the skateboard would like to know how many flips he/she was able to accomplish, or the exact degrees of a trick, let’s say they want to do a 180 degree move, we would need to use the gyroscope in order to detect the movement of the board and send the information to the microchip to be processed. Being able to use a gyroscope to monitor the orientation of the board is incredibly helpful in being able to complete our project. Many times, the gyroscope comes integrated already in a board that can complete the most popular tasks users are looking for.

An example of such boards are the Arduino boards. Many Arduino boards already come with the gyroscope integrated, as well as wireless or Bluetooth communication capabilities, that enable the user to obtain the information recorded by the gyroscope. In our case we will not be able to use a board that comes equipped with a gyroscope, since we are required to design our own printed circuit board. We have made it our mission to find a gyroscope that meets all the criteria we are looking for.

The gyroscope we are looking for will be low power consumption, since we would like our users to be able to enjoy the device for long periods of time before having to be recharged, while saving them money at the same time. The smaller size of the device possible, while providing accurate readings, and withstanding rigorous shocks, preferable 10,0000 G-shock and above, since our users will be exposing the device to high impacts.

The gyroscope we choose will measure the angular motion of the skateboard with very high accuracy at almost real-time, such that we are able to deliver a high quality product, it will also be able to communicate with the microchip in an uncomplicated manner, such that the programmer can find it simple to complete the main tasks of the project.

There is a variety of gyroscopes to choose from on the market, all which offer different features we can benefit from. It could be because of price, or because of performance, but each gyroscope has a specific design advantage and disadvantage as far as creating the SMART Skateboard goes. In order to achieve our goals for this project, the group needs to make sure the gyroscope meets the specifications needed to provide a reliable and desired feedback coming back to the microchip. The following bullets are the exact specifications which we are looking for:

- Triple axis with digital output (provide feedback for the direction of the X, Y, and Z axis)
- Has to be digitally programmable
- Low energy consumption (looking for approximately 6-7 mA operating current.
- Be able to support a wide supply voltage range of around 2-4 V

SparkFun ITG-3200 Triple Axis Gyroscope

This gyroscope is from SparkFun, it is according to the description the world’s first single-chip gyroscope that offers digital output and uses 3-axis to process MEMS motion, and it is dedicated and designed with the gaming industry in mind, as well as 3D mice and other top-technology devices, such as motion-based remotes and smart tv’s with the ability to connect to the internet. It has a temperature sensor already embedded and uses an internal integrated oscillator with a 2% accuracy. This gyroscope is incredibly interesting and very useful for our project because it’s size has been reduced by a respectable 60% from the size of similar gyroscopes that would complete similar tasks, this allows our team more space on the printed circuit board, and the ability to add more features to the project, without compromising space its actual size is 4x4x0.9mm, that is a huge improvement for this device, all while reducing the heat dissipated by the parts, and saving a considerable amount of energy, which is up to 30% less energy than many of its competitors.

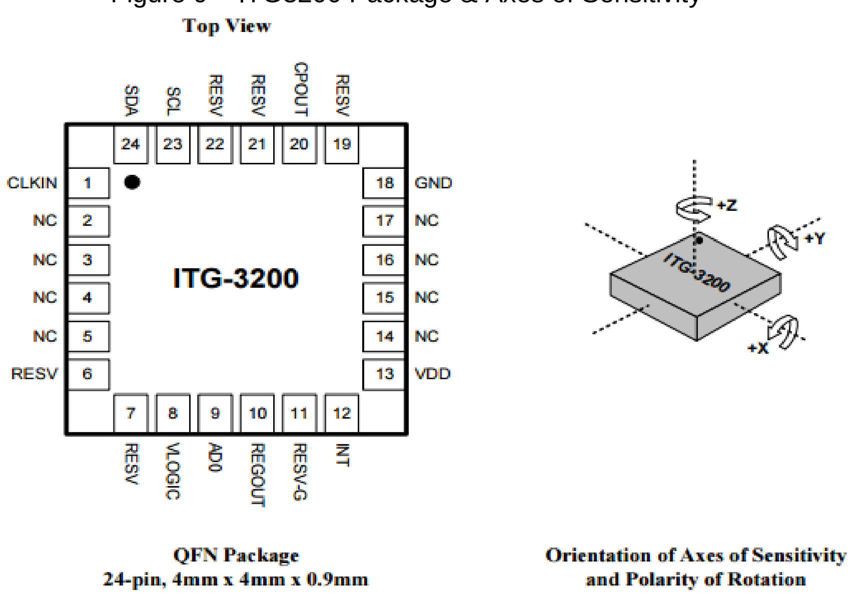
Features

- It uses digital outputs and senses movement in the X, Y and Z direction using integrated sensors in a single circuit (Triple Axis)
- Has a low pass filter that can be digitally programmed
- It has low energy 6.5mA current consumption for long lasting battery benefit
- Runs on very low standby current of 5µA
- Fast mode serial interface 400 KHz specifically
- wide supply voltage range of around 2-4 V

Something else that is very important to our team, is that the ITG-3200 was designed with intense sports in mind, therefore they made it shock resistant, one of the features our team is focusing the most on, since our product is going to be involved in jumps, intense tricks and flips, this feature is of our utmost interest, knowing that there is a gyroscope in the industry that can withstand the amount of activity out project requires is alleviating, since it is one less problem we have to worry about, the device has a 10,000g shock tolerance, plus the added security we will provide, makes it more than sufficient for our team to work with.

Figure 9 below shows the size of the ITG-3200 and the axes of sensitivity that it can measure.

Figure 9 – ITG3200 Package & Axes of Sensitivity



The permission for the use of this image has been granted by SparkFun

Maxim Integrated MAX21000

This Gyroscope is also a great choice for any team looking to accomplish motion detection using low energy. The MAX21000 from Maxim integrated is a 1.8V 3mm x 3mm in size gyroscope, something very beneficial for our team, since we are trying to complete our project using the minimum space possible, this gyroscope like the ITG-3200 offers digital output, as well as very precise accuracy and sensitivity over time and temperature, this gyroscope is also interesting since it offers us to a selection of finely tunable bandwidth, something that would allow us to tune the gyroscope to the specific bandwidth required by our project in order to deliver high accuracy results to our target users.

Features

- Small in size (3mm x 3mm x 0.9mm LGA)
- Does not have the need of external components in order to operate
- Low power
- Can be ran in Eco mode in order to save battery life Eco mode runs at 100Hz with 3.0mA typ
- Can be operated with a minimum 1.71V minimum supply voltage
- Has a 9µA type current mode (sleep mode)
- Extremely fast turn on time of 45 milliseconds from sleep mode
- Extremely fast turn on time 5 milliseconds from standby mode
- Very minimal delay of approximately 3 degrees at 10Hz
- High bandwidth availability of 400Hz
- It is shipped factory calibrated, to save the team the time it would take to calibrate the device
- Able to withstand shocks at an incredible 10,000 G-shock
- Offers LSB data mapping
- Single data capture trigger
- Multiple data capture trigger: to make the processing of data in almost real time

SparkFun MLX90609 Gyro Breakout Board

Although the MLX90609 breakout board from Melexis Microelectronic Integrated Systems has been discontinued and it is no longer for sale by SparkFun, it is still a good candidate to use as a reference, to compare with other products available on the market and to research the possibilities and options we have as a team when it comes to sensing the X, Y, Z movements of the skateboards. The MLX90609 has both, digital as well as analog output, and low acceleration and angular rate cross sensitivity, it can sense movements in the three axis just as we need in our project design, and it is also all integrated on one chip.

Even though there is higher technology in the market this gyroscope can accomplish many features and it is cost efficient, it has low zero rate output drift, on-chip calibration ability and an operating range temperature of -40 degrees Celsius to 85 degrees Celsius.

One downside to this board is that it consumes a lot more energy than its replacement the ITG-3200, as it uses 5V supply in order to operate correctly. Whoever prefers dealing with analog input instead of digital, and who is not on a power restraint can benefit from this device. However, since our team is trying to keep power consumption as low as possible, with other options out there, we would opt-out of using the MLX90609 gyro breakout board.

Summary

Table 5 below shows a summary of all of the gyroscopes considered for our project and everything they have to offer in terms of features. This table will make the final selection much easier to figure out.

Table 5 - Gyroscope research summary

Gyroscope	Features
ITG-3200	• Low pass filter that can be digitally programmed. • Low energy current draw • Runs on very low standby current • Fast serial interface of 400 KHz • Wide supply voltage range • 3 Axes of sensitivity
MAX21000	• Small in size • Does not need external components to work • Low power • Can be ran in economy mode to save energy • Extremely fast turn on time from sleep or standby • Minimal delay • High bandwidth • Shipped factory calibrated
MLX90609	• Low zero rate output drift • On-chip calibration capability • Wide operating temperature range

3.2.4b Barometer

The use of the barometer was something that we deliberated on for a while. We couldn’t think of a proper way to figure out the height of a microcontroller. Unlike the accelerometer and gyroscope, the barometer did not come as a given sensor to implement into the SMART Skateboard. After doing much research and inspecting Syrmo’s designs carefully, we finally discovered the importance of having a barometer in the device.

A barometer is going to be very indispensable to our project, as it will play an important role determining the altitude of the board at all times. Many times the user performing a trick wants to know how high he/she was able to jump, in order to determine what adjustments need to be made to perform the trick better, or simply to practice performing a different trick after they are able to reach a certain altitude milestone.

The barometer is going to help us achieve the altitude feature of our project, by using the barometer we are going to provide the user with real time feedback of the altitude of their jump. This is crucial to providing the most possible information to the user of the SMART Skateboard. This feature alone is one of the most distinct features of this project. Without it, the user would have no way of comparing different trick heights and competing with friends on specific trick height.

We will implement the barometer to perform along with the gyroscope in order to provide feedback on the altitude as well as the orientation of the trick performed by the user. The combination of these two sensors, alongside the accelerometer will provide all of the analog data that we need to make the SMART Skateboard's sensors a reality. There are a wide variety of barometers on the market. We have chosen a select view to compare and contrast to decide which one to select in our project.

Barometric Pressure Sensor MPL225A1

The MPL225A from SparkFun is a digital barometric sensor and it uses one of the latest technology in MEMs to be able to deliver the most accurate measurements possible to the user. This barometer can measure pressure between 50 kPa and 115 kPa. The sensor is designed to use low current consumption to save battery life and allow the user to design a device that can perform more while consuming less.

The MPL225A uses only 10µA at a measurement per second. This sensor is able to output both temperature and pressure information on an SPI bus. It is also designed with the space restriction we face nowadays in mind, and it was kept at only 5 x 3 mm and 1.2 mm in height, this would allow our group to stick with the design with started, since it provides the appropriate size for us to integrated into the circuit board without sacrificing too much space.

Features

- Supports a wide range of supply voltage
- Can operate at a temperature range of -40 degrees Celsius, to 105 degrees Celsius
- Low power consumption
- Can output monotonic pressure and record temperature
- Very high Kpa accuracy
- Small in size 5 x 3mm (1.2mm height)

Barometric Pressure Sensor BMP180

The BMP180 barometric pressure sensor is very similar to the MPL225A except for its overall size. This pressure sensor is slightly larger but uses less energy than the MPL225A. This is a feature which we would have to take into account when selecting the proper barometric sensor. Since we are trying to accomplish our project using the minimum amount of energy, we will have an end product that the users can use for a long period of time without having to worry about recharging all the time. This barometer would take up more space in our board which could be enough to render it useless. The energy consumption is not too out of our desired range to consider selecting the BPM180 over the MPL225A.

This barometer is of high precision and it is digital, two of the important features we are considering for our project. It offers a range measurement of 300-1100 hPa with an incredible accuracy of 0.02 hPa error. This range is perfect for anybody trying to obtain the most accurate data information possible, which is something our senior design group is trying to accomplish. This barometer uses piezo resistive technology in order to provide the robustness, the high accuracy required by most users, the linearity, and the long term stability necessary to record the tricks our SMART Skateboard users will be performing.

Since our skateboard sensor is being designed with hobbyist and professional competitors in mind, we need to make sure we provide a sensor that records and sends the most accurate information possible at all times, with no exceptions. Many tricks are extremely hard to perform and athletes practice a very long time to perform it just once, therefore we have to make sure our sensor is able to deliver at the highest quality possible every single time it is being used.

The voltage supply supported by this barometric pressure sensor is between 1.8 V DC and 3.6 V DC which means it operates at a slightly lower range than the sensor we previously described. One feature that really caught our attention with this sensor is the fact that this sensor is designed to be connected to any microcontroller we choose directly, by using the very common I²C connector.

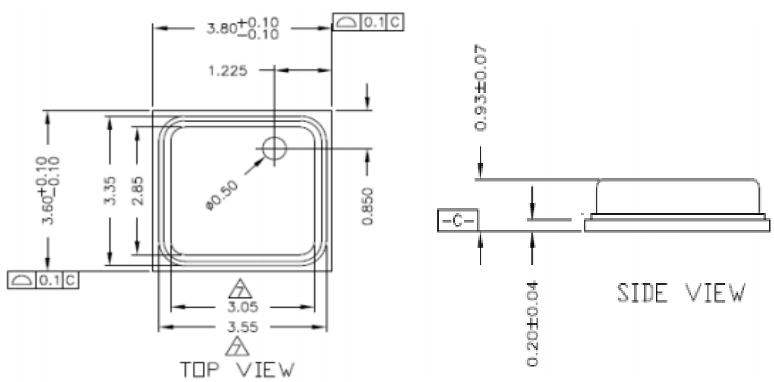
This sensor is an ultra thin one and is made using a ceramic leadless material, which means we would have to use an oven to solder it. This is also something we would have to put into consideration. An oven for soldering is not as accessible to us as an ordinary ironing iron, which we could use at any time at home or in the senior design lab.

We have to consider this because of time restriction implemented in designing the SMART Skateboard. Even though our board would be cleaner and the design would be more optimized, there are other, more dire features for us to consider. Below we list some of these distinct features.

Features

- Supports a wide range of supply voltage
- Has a digital I²C interface
- Super low power consumption
- Can measure results with very low noise
- Fully calibrated out of the box
- Can measure temperature
- Very thin and small in size (Figure 10 shown below)

Figure 10 - Top and side view of Barometric Pressure Sensor BMP180:



The permission for the use of this table has been granted by SparkFun

Having the top and side view of the barometer (shown above) is essential to the design of the SMART Skateboard. Using these figures as reference, we know exactly how much space the sensors would take up when we implement it in our customized board. These images help us design a better enclosure, where our device does not overheat or create any kind of problems in the long-run. The size of the barometric pressure sensor above is not the smallest in the market, but it is something that we can work with. This barometer can be implemented without having to sacrifice too much space. This option fits the initial constraints our team defined early on, but we would ideally still like to find a barometer with leads. Having leads implemented with our barometer would make it less complicated to implement this sensor into the overall design of the SMART Skateboard. This sensor not having leads would also make any changes to the board, in case the sensor has to be replaced or re-mounted, much more difficult in comparison to the other sensors detailed earlier in this section.

Summary

Table 6, shown on the next page summarizes each barometer considered for selection by the group and the features they offer. This table makes the final selection much simpler for our group, and it is a great reference for our group to when making this selection.

Table 6 - Barometer research summary

Barometer	Features
MPL225A1	• Wide range of supply voltage • Wide operating temperature range • Low power consumption • Measures monotonic pressure and temperature • High accuracy
BMP180	• Wide range of supply voltage • Low power consumption • Fully calibrated out of the box • Very thin and small • Digital I^2C interface

3.2.4c Accelerometer

The Accelerometer is the sensor that is going to bring the most excitement to the project. It will help us determine the acceleration and speed of the user traveling on the skateboard. Since speed is very important when performing a skateboard trick, this feature of the SMART Skateboard will allow the user to know exactly how fast he/she is going while performing their favorite maneuvers. The accelerometer is going to be used to give the microchip feedback with information on how fast the user is traveling. This will help the user determine what specific measureable adjustments need to be made in order to perform certain desired tricks.

The accelerometer used in combination with the gyroscope is going to help us provide the velocity to the user, as well as the altitude and the position of the trick. For example, if the user is wanting to accelerate and reach 10 mph before attempting a 10ft jump and a 360 degree flip, we will make the feedback possible for them. Our project will let them know if the trick they were looking to performed was achieved or if they failed to perform, that would give them a sense of the changes needed in order to be successful the next time they try it again, and accelerometer is a must have sensor when providing statistics on any sport, specially skateboarding.

We need our accelerometer sensor to meet certain standards in order to deliver a quality product that is energy efficient, all while keeping the device at a minimal size. There are a wide variety of accelerometers on the market. In the proceeding sections we present our research into which models we reviewed in our considerations.

SparkFun Triple Axis Accelerometer Breakout Board ADXL377

The SparkFun ADXL377 is a very simple breakout board, but it performs with very high accuracy. The ADXL377 was designed with space restriction in mind, it was made very thin and occupies very little space. It is a complete 3-axis board with all the features needed for our senior design project. It is low power sensor, with an operation voltage range of 1.8 volts to 3.6 volts. At first glance this is not the best range currently on the market, but that specified range is for a whole breakout board, not just the sensor alone. When we do our customized printed circuit board we will already have most of the components found in this board. The ADXL377 uses very low typical current of 300 µA, which is very impressive for the functions it performs when compared to other sensors similar to the ADXL377. The board contains all the pins needed to receive all of the necessary information from the accelerometers. It is easy to set up and can measure acceleration in the X, Y, and Z direction. We can implement the accelerometer on this board to our printed circuit board without much difficulty, since it lines up with the breadboard tests we have run so far.

Features

- Supply voltage from 1.8v - 3.6v
- typical current of 300 µA
- Supports a wide range from - 200g to +200g
- Adjust the bandwidth with one capacitor per axis
- 4x mounting holes for easy implementation
- Minimal size requirement

Triple axis Accelerometer Breakout Board ADXL362

The ADXL362 is very similar to the breakout board ADXL377, since it is in the same series. This accelerometer is also a 3-axis sensor and uses MEMS technology for the measurement of acceleration. This acceleration measurement system can operate at a level where it draws extremely low energy, which is a huge benefit in regards to designing the SMART Skateboard device. It is capable of measuring the dynamic acceleration resulting from both motion and from shock as well as static acceleration that can be caused by the board tilting, a feature that could be extremely helpful in determining at what speed our user lands a trick alongside the impact of the trick.

One advantage we have determined in the ADXL362 is that it was designed so that the it is easy to communicate between the processor and the sensor. This makes our team's work a lot easier and saves us tremendous amounts of time designing the SMART Skateboard device and its schematics. Since the programming for the ADXL362 is much simpler than its counterparts, we can easily implement this sensor into our design without worrying about the software constraints of doing do.

Another feature of this specific accelerometer is that this device allows us to use is the "wake-on-shake" feature. This feature can be hugely beneficial for our design, since it saves the user a substantial amount of energy while sitting idle. Many times the user can forget to turn off the device and as a result their battery life suffers, which is an inconvenience for our users.

With the “wake-on-shake” feature, our users are enabled to forget to turn the device’s power off and their battery life would be essentially unaffected. Once the skateboard is on the move again, the device enables and starts recording the acceleration/speed of the skateboarding session being performed.

The ADXL362 allows our team to add extra features if needed, without compromising the user experience. The features offered by this breakout board are very intriguing, and can add a tremendous value and capability to our project.

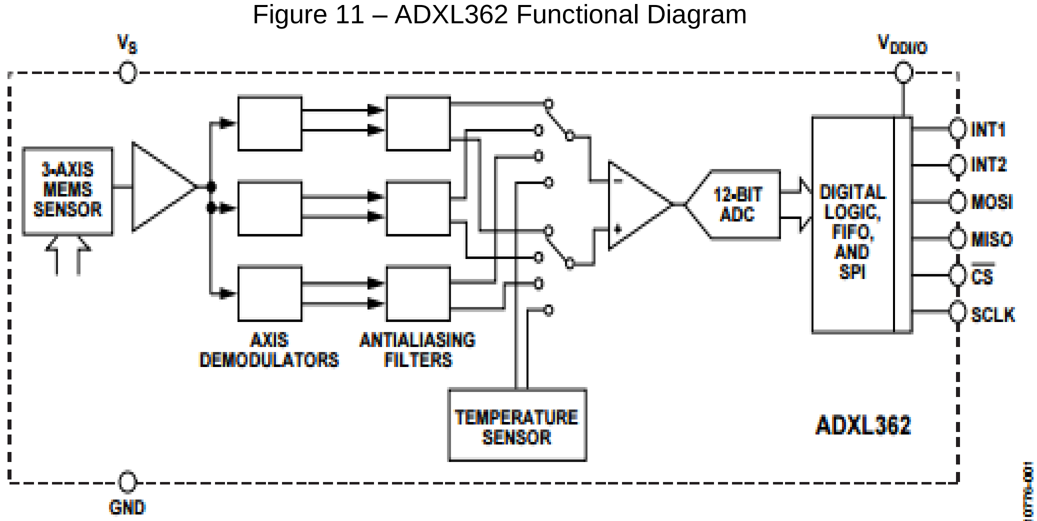
Features: we need this accelerometer sensor to meet certain standards in order to deliver a quality product that is energy efficient, all while keeping the device at a minimal size. Listed below, are the features of this accelerometer that are crucial to the development of the SMART Skateboard device.

- Measures in the 3-axis
- Wide measurement range could be selected from +-2 ; +-4 ; or +-8g
- Super low power consumption
- Low noise 175 μg/√Hz
- Wake-on shake feature
- Thin size board

Functional Diagram:

We have included a diagram of the ADXL362 accelerometer, Figure 11 below, showing how this specific sensor breaks down analogous information and converts it to a digital output easily recognized by the main processing chip. This diagram is crucial to understanding how the ADXL362 works, but more importantly it allows us a reference when creating schematics of our final design. By including the figure below, we can better understand the leads of the ADXL362 sensor as they pertain to developing the SMART Skateboard’s customized printed circuit board.

Figure 11 – ADXL362 Functional Diagram



The permission for the use of this diagram has been granted by SparkFun.

This diagram shows the simplicity of the ADXL362, with the few components shown above, the device is able to perform one of the most important functions of our project, accurately measuring the speed of the user riding the skateboard. The sensor used temperature as well as position to be able to determine and record extremely precise measurements of the speed/acceleration and in turn sends the information to the processor. The size of this device would make it easier for our team to implement it on our PCB, since we are trying to keep the device to the smallest size possible. This accelerometer makes for a very viable candidate, not just because of its size, but also because of its diverse features in comparison to its counterparts.

Summary

Table 7 on the following page summarizes each accelerometer considered for implementation in our project and what features they offer. This table will make the final decision much easier.

Table 7 - Accelerometer research summary

Accelerometer	**Features**
ADXL377	● Measures wide range of acceleration ● Easy implementation ● Minimal size requirement ● Low current draw
ADXL362	● Low power consumption ● Low noise ● Wake-on shake ● Thin size board

3.2.5 Serial Communication Interface

While it is very important to select a reliable microcontroller and accurate sensors, all of that hardware would be useless if there was no way to interface them together. That means selecting the right serial communication interface is extremely crucial to the success of the project. The following section shows our research into the options that are available to the group in the area of serial communication.

3.2.5a Serial Communication Options

The group researched several serial communication options for consideration into our project implementation. We outline each on the following page.

SPI

Serial Peripheral Interface (SPI) bus is a synchronous serial communication interface used widely in Embedded Systems. It was created by Motorola and has become an industry standard ever since. It can be implemented with single master to single slave or single master to multiple slave configuration.

Advantages

1. High throughput
2. Full-duplex (easier to program)
3. Low power
4. Slaves don't need addresses
5. Full control over message size

Disadvantages

1. Can only have one master
2. Short distance
3. More pins on IC packages than I²C

I²C

Inter-Integrated Circuit (I²C) is another type of communication interface between chips. Its best use is to connect low speed components to microcontrollers which would make it perfect in the SMART Skateboard. It was invented by Philip Semiconductors and now has no fees to use; however, fees are still necessary to obtain slave addresses.

Advantages

1. Multi-master
2. Fewer connections than SPI
3. Chip addressing makes it easier to add more devices
4. Flexibility in bus voltage

Disadvantages

1. More software overhead
2. Low speed
3. Half-duplex
4. Need to select addresses at circuit design stage

RS-232

RS-232 is another standard for serial communication. It defines the two devices as a DTE (Data Terminal Equipment) or DCE (Data Circuit-Terminating Equipment).

RS232 used to be very prevalent in personal computing but it was hampered by slow speed and has since been replaced by USB communication; however, many industries still use RS232 or industrial grade equipment.

Advantages

1. Cost effective

Disadvantages

1. Lots of connections
2. We would have to change the circuitry

3.2.6 Wireless Communication Interface

An essential component to our project is how we wirelessly create a communication link between the user's mobile device and the hardware unit located below the skateboard. With that being said, the range will not need to exceed five to ten feet. Also, there is very little data needed to be transmitted. We will only classify a select few ground tricks for the purposes of our demonstration. As we compare the different options, some things we will need to consider when picking a suitable wireless technology are the power consumption, cost, and the physical size of the transmitter and receiver. A number of relevant technologies come to mind, however these constraints and requirements limit us to a handful of possibilities. To name a few, there are Bluetooth, Bluetooth Low Energy (BLE), Wi-Fi or WiMAX.

3.2.6a Communication Options

Bluetooth (Classic)

Bluetooth (BT) is a wireless connection technology that allows you to simultaneously pair with several devices. Similar to Wi-Fi, you can use Bluetooth to exchange files and operate electronics. However, this is done at a shorter range and lower bandwidth (more reliable than other wireless alternatives). This technology was considered for the following reasons:

Assembling a Bluetooth connection between two devices is a quick and easy setup. The exact interfacing varies depending on the device, but to connect the devices you need make one discoverable while the other one scans. Once the scanning device finds the other, you initiate the connection and enter the PIN as directed by your user's manual. Now that the devices have been paired, you shouldn't have to run through the reconnection process again. Bluetooth technology is compatible with any other peripheral device that supports BT, regardless of make, model or design. You can apply this technology with your mobile phone or even pair essentials to your gaming console or computer for easy chatting online. For example, your Bluetooth keyboard can work with your computer and your PlayStation. The only issue you're likely to encounter is if the device in question can only pair with a limited number of devices.

The processing and battery power required to operate BT technology is very low compared to alternatives like Wi-Fi. When enabled, the power consumption is rated at 1W with a peak current < 30mA. Physical data rates are 1-3 MBit/s with an effective application data throughput of up to 2.1 Mbit/s. Serving as an ideal tool for our project, this technology can be implemented to almost any device.

Bluetooth Low Energy (BLE)

Similar to the “classic” BT, Bluetooth Low Energy (BLE) is a wireless personal area network technology that’s designed to provide reduced power consumption and cost, while maintaining a similar communication range. BLE is supported by a number of operating systems, some of which are listed below:

- iOS 5 and later
- Windows Phone 8.1
- Windows 8 and later
- Android 4.3 and later
- BlackBerry 10
- Linux 3.4 and later through BlueZ 5.0
- Unison OS 5.2

This is an attractive aspect to our product requirements, since it will likely save us time on compatibility issues between various transmitters and receivers. Below are a few technical details comparing the traditional Bluetooth with Bluetooth Low Energy.

Table 8 below, allows for quick and easy comparisons when selecting the most optimal option between the two technologies:

Table 8 - Comparisons of Technical Details

Technical Specification	Classic BT Technology	BLE Technology
Distance/Range (theoretical max.)	100 m (330 ft)	>100 m (>330 ft)
Active slaves	7	Not defined; implementation dependent
Latency (from a non-connected state)	Typically 100 ms	6 ms

Power consumption	1 W (as the reference)	0.01 to 0.5 W (depending on use case)
Peak current consumption	<30 mA	<15 mA
Primary use cases	Mobile phones, gaming, headsets, stereo audio streaming, smart homes, wearables, automotive, PCs, security, proximity, fitness, etc.	Mobile phones, gaming, smart homes, wearables, automotive, PCs, security, proximity, healthcare, fitness, Industrial, etc.

Wi-Fi

Wi-Fi is another wireless local area network (WLAN) technology known to operate within the 3 GHz frequency band. There are two general types of Wi-Fi transmissions: DCF (Distributed Coordination Function) and PCF (Point Coordination Function).

Our focus is on the DCF transmission, or better known as “Ethernet in the air.” This transmission utilizes a packet-based structure. This structure is very similar to that of the Bluetooth.

However, this technology is optimized for large data transfer, using high-speed throughput; which consequently rely on a high power consumption rate. This issue begins to stray away from our needs, as it will be less efficient and require a battery of substantial size in respect to our device.

WiMAX

Worldwide Interoperability for Microwave Access (WiMAX) is easily one of our more compelling options. This power intensive technology provides about 10 megabits per second of throughput, at distances up to 10 kilometers from a single base station. With that being said, this does require a large amount of electrical support, as well as huge operational costs. For obvious reasons, this is unacceptable to our needs and eliminates WiMAX as a considerable selection. Table 9 below summarizes the advantages and disadvantages of WiMAX.

Table 9 - Pros and Cons of WiMAX

Pros	Cons
Larger broadcast (30 mile radius)	Limited support for sparse/rural areas
Access to widest array of devices	Costly option
Delivers fast, low cost internet	Limited to dense/populated areas

Near Field Communication (NFC)

Serving as an advancement for an outdated technology, Near Field Communication exchanges data by transferring information to a nearby transceiver. NFC not only beats Bluetooth in connection time, but in power consumption and the area of interface. Which would prove useful in our application where there could potentially be up to four or more skateboards all broadcasting their own signal to a user device.

Nonetheless, this technology is only effective when devices are held within a close proximity to one another. Typically, NFC is a popular convenience for in-store payments, allowing the user to digitally complete a transaction through a smartphone or wearable.

The image below depicts the range required for connectivity between devices. This figure shows that, for the purposes of the SMART Skateboard, the optimal range is much smaller than the specified 5 to 10 feet we outlined earlier in the requirements and specifications section of this document.

Figure 12 – Visual of NFC Range

Permission Pending from MasterCard

This feature however, excludes NFC from the list of wireless options due to its low range potential. Near Field Communication is best suited to operate in the four-centimeter range, which obviously does not meet our project requirements.

After careful consideration, Bluetooth low energy seemed to be the most reliable and consistent option in regards to our project. Not only does this technology meet design requirements, but it also gives us the freedom to establish an easy-to-use wireless link between devices. The next aspect the group will need to consider is the type of module (chip) that will be integrated into the SMART device.

3.2.6b BLE Module

The purpose of a Bluetooth module is to essentially gather all information from external sensors; both analog and digital data. Once the information is collected, the module will exchange this content via radio frequency to the external device.

Bluetooth modules also come in two different package options: Quad Flat No Leads packages (QFN) and Wafer Level Chip Scale packages (WLCSP), also known as (CSP). There are a few noted differences between the two. QFN packages are a little costlier since it already contains the required components to operate the module, whereas using a WLCSP will be slightly more expensive when used on a PCB design. This is because it requires tighter tolerances and more than two layers of tracing. Depending on one's requirements, QFN might be the more inexpensive option when used on a PCB design, though CSP is more suitable for really small product designs where QFN doesn't fit. Below, this section will cover the several types of Bluetooth modules and review the pros and cons of each manufacturer.

Texas Instruments CC2541

As we weighed the options, one module that initially stood out to the group was the CC2541 chip by Texas Instruments. The CC2541 is highly suited for systems where a flexible, integrated and low-power Bluetooth Low Energy solution is required. With its software stack, this device enables easy integration of a BLE solution for developers. This chip combines the excellent performance of a leading RF transceiver with an industry-standard enhanced 8051 MCU, in-system programmable flash memory, 8KB RAM, along with many other powerful supporting features and peripherals. The list below details a few more specific features offered by the CC2541:

- RF Features
    - 2.4-GHz low energy Compliant
    - Supports 250-kbps, 500-kbps, 1-Mbps, 2-Mbps Data Rates
    - Excellent Receiver Sensitivity (-94 dBm at 1 Mbps), Selectivity, and Blocking Performance
- Microcontroller Features
    - High-Performance and Low-Power 8051 Microcontroller Core With Code Prefetch
    - Hardware Debug Support
    - Retention of All Relevant Registers in All Power Modes
- Applications
    - 2.4-GHz Bluetooth low energy Systems
    - Human-Interface Devices (Keyboard, Mouse, Remote Control)
    - Sports and Leisure Equipment
    - Mobile Phone Accessories

Table 10 below outlines the specifics of the CC2541 in a chart to make it more manageable as a reference when selecting the most optimal chip.

Table 10 – CC2541 Information

Parametrics: CC2541	
Device Type	Wireless MCU
Bluetooth Type	Bluetooth Smart (BLE)
Layout	6-mm × 6-mm QFN-40 Package
Operating Temperature Range (C)	-40 to 85
Rating	Catalog
Approx. Price ($US)	1.79 | 1ku
Technology	Bluetooth Low Energy

Nordic Semiconductor nRF8001

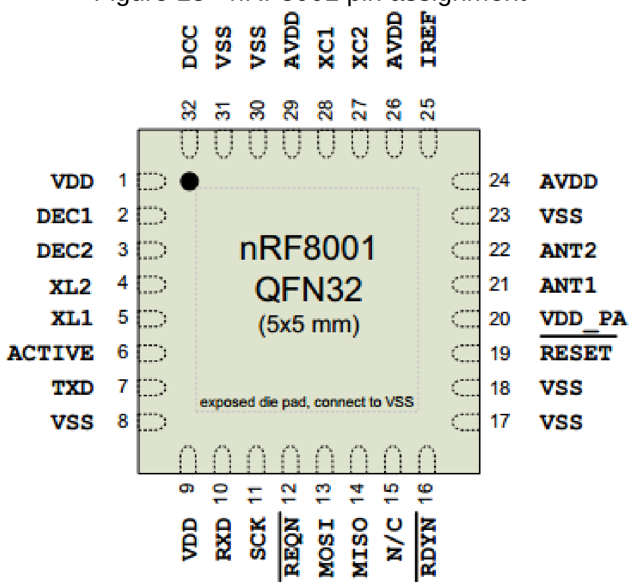
Another module that we carefully considered was the nRF8001. The nRF8001 has on-chip non-volatile memory for storing service configurations. Which means this on-chip storage allows you to select/combine the necessary services for your application. Ultimately giving us the freedom to reduce certain requirements on the application controller for handling real-time operations related to the BLE communication protocol. These features are all accessible through the Application Controller Interface (ACI).

The nRF8001 also provides an optional output timing signal “ACTIVE” that is activated before the radio starts to run. This signal enables you to control the application circuitry; avoiding noise interference when the nRF8001 radio is operating. This single chip is tailored specifically for Bluetooth low energy applications that operate in the Peripheral Slave role. Examples include: proximity tags, PUID watches, remote controls, and sports/fitness/healthcare sensors. The on-chip Link Layer and Host stack also include support for Peripheral GAP role, client, server and security functions.

In addition to these unique features, the nRF8001 also hold the best-in-class power consumption rate. By integrating a battery level monitoring system, this chip has a low tolerance 32kHz RC oscillator which eliminates the need for an external 32kHz crystal. Also incorporating two voltage regulators: there’s a linear voltage regulator providing a 1.9-3.6V supply range, and a DC/DC voltage regulator that can further cut current consumption by up to 20% when running from a 3V battery cell. The nRF8001 is available in a 32-pin 5 x 5mm QFN package. The backplate of the

QFN32 capsule must be grounded to the application PCB in order to achieve optimal performance. The figure below displays the pin assignment for nRF8001:

Figure 13 - nRF8001 pin assignment



Permission pending from Nordic Semiconductor

Microchip RN4020

An alternate product the group considered is Microchip's RN4020 Bluetooth low energy module. The RN4020 is a fully-certified, Bluetooth Version 4.1 low energy chip for designers who want to easily add low power wireless capability to their devices. This surface mount module has the complete Bluetooth stack on-board and is controlled through simple ASCII commands over the UART interface. Other features include the complete set of Bluetooth SIG profiles, as well as MLDP (Microchip Low-energy Data Profile) for custom data. Users also have the leisure to enable the standalone operation without a host MCU/Processor by applying the scripting feature.

There are many different features and characteristics of the Microchip RN4020. To give a better idea of this product, the following tables provide a few conventional specs:

Table 11 – RN4020 Features, Characteristics, Parameterics

General Features	
Specification	*Description*
Standard	Bluetooth 4.0
Frequency Band	2.4 - 2.48 GHz

Max Data Rate	1 Mbps
Interface	UART, PIO, AIO, SPI
Operation Range	100 m
Sensitivity	-92.5 dBm at 0.1% BER
Temperature (operating)	-30°C to +85°C

Electrical Characteristics	
Specification	*Description*
Supply Voltage	1.8 - 3.6V DC
Working Current	12 mA (typical)
Standby Current (disconnected)	<0.5 mA

Parametrics	
Specification	*Description*
Dimensions	19.5 x 11.5 x 2.5 mm
Weight	1.2 g

3.2.7 Printed Circuit Board Options

As our group started the project research, we realized there are many ways to go about the circuit board that we will be designing and ultimately using on our project. After much brainstorming, the group decided we have to design the board taking durability into consideration, due to the fact that it will be used in rigorous activities (tricks, and jumps from high altitude) during its demonstration. We concluded the best way to achieve this is to keep the printed circuit board to a small size constraint, that way we could fit it under the front wheels, where it would be the safest since the metal truck and the wheels are the toughest part of the skateboard. This will provide the most security, and prevent unnecessary shocks and damage to the chip and sensors, which could cause inaccuracies in our readings.

In order to achieve our goals and meet our board's design needs, we must find a company that will keep their word with excellent service and fast shipping. We must also find a company that can sell its products to us at a reasonable price. We have a limited budget and are on a time constraint, since we need to deliver a good quality working project at the end of the semester. There are many circuit board manufacturers out there, but not all meet our needs of reliability, having a low price and being able to accomplish a short-term delivery.

Numerous of the companies we researched that can provide a competitive cost are international, which means we will have to wait a longer amount of time to receive our product. That forces our team to work at a faster pace under more time constraints, forcing us to finish the circuit board design before the time we had originally planned for. We need a somewhat local company. That way we have a proper amount of time to test the board, deal with any failure that might arise, replace parts that have shorted out or any other unforeseen changes that could be made.

3.2.7a Possible Manufacturers

Many of the PCB manufacturers seem promising and offer reliable service, but so far, through research and recommendations we have an intuition that Bay Area Circuits is the most appropriate one to choose to create our 4-layer printed circuit board. They are offering a reasonable price, and they promise a delivery time that we could work with comfortably. So far we have taken many companies into consideration, amongst those companies are Sunstone Circuits, ExpressPCB, PCB Zone, PCB cart, JDBPCB, Bay area circuits and others.

Sunstone Circuits instant quote was $195.98 for a 4-layer printed circuit board, not taking into consideration the shipping and handling costs. ExpressPCB had it at a similar price of $208.97. PCB Zone had it at a much more reasonable price of $78.94. PCB Cart quoted us for $68.71. JDBPCB had a good price of $45.00. And finally, Bay Area Circuits offers, with a student discount, a price of $30.00 including 5 days shipping.

Their offer definitely meets our budgetary and time needs, since it is cost efficient, and in case we need to make a new order, five days is a reasonable amount of time to have the board build and shipped. All the above prices are excluding shipping and handling, except the student discount from Bay Area circuit.

The manufacturers had various prices depending on how fast we wanted to have our board build and shipped. Having the board shipped through express shipping could add a cost of anywhere between $40.00 and $80.00, which we could not afford under our constraints and budget.

On Table 12, shown on the next page, we included a table of all the possible manufacturers that we could use to order the custom printed circuit board as well as the price of the product alongside the location of the main manufacturing plant of the company. This table allows us to quickly weigh our options and decide on a manufacturer that can most satisfy our printed circuit board needs.

Table 12- PCB manufacturers quote

PCB Manufacturer	Price	Manufacturer Location
Sunstone Circuits	$195.98	United States
Express PCB	$208.97	United States
PCB Zone	$78.94	New Zealand
PCB Cart	$68.71	China
Bay Area Circuits	$30.00	United States
JDB PCB	$45.00	China

3.2.7b PCB Software Design Options

Something else we have to take into consideration is which software we will be using to design our custom printed circuit board. Some of the PCB software out there offer a free version. For example, ExpressPCB have their own CAD software they offer to their customers for free, so that the design process as well as the manufacturing process goes smoothly if we were to choose to do the job with them. That would be of help to our budget and would save us the time of having to download and pay for other software. However, since their price is significantly higher than other companies, we would most likely have to make the decision of going with another company that might not offer a free version of a CAD software tool.

One such software we have been taking into consideration to build our printed circuit board is EAGLE CAD, the reason why we are inclining into using EAGLE is because they offer a free version of the software as well as a paid version with features that seem very appealing and that meet our needs. Since the software is one of the most common in the PCB design industry, we are able to find more help and guidance on how to design a printed circuit board, or in case we run into any inconvenience, there is a faster and easier way of solving the issue, since many people out there are already using the software.

Aside from EAGLE CAD and the ExpressPCB software, there are many offered at no cost, and can help our team finish the job of creating the design of our PCB in case we do decide to go another route. Some of the software we are able to use at no cost are, ZenitPCB, PCBWeb Designer, TiniCAD, Osmond PCB, BSchV amongst others. What makes the difference between the software mentioned above is that each one of them offer a different way of importing parts, and some give you the ability to complete tasks easily that would take you a great amount of time otherwise. For example, EAGLE CAD offers the ability to do auto routing on their paid version of the software, in some situations this feature could save our teams many hours of work, that we could use to improve our design, it also generates a parts list and a bill of materials that makes the process of ordering and manufacturing the board significantly easier than if we were to do everything manually ourselves. Our team has the responsibility of becoming proficient at using any of the software mentioned above in a timely manner, in order to be able to deliver a working product at the end of our senior design.

3.2.8 Device Holster Options

Once the entire SMART Skateboard device is fully designed, tested and built, the next step is to configure the most efficient way to holster this device to the bottom of a skateboard with minimal interference for the user. It seems as though there are many viable trains of thought for doing this effectively. Below is a review of three of the most relevant options with details about each method for clarification.

Adhesive (Glue/Tape)

The most obvious, as well as elementary, method of fastening anything securely is to use an adhesive to bond the device to the skateboard. Although this method would definitely ensure that the device will stay in place during skateboarding sessions, there are several flaws with this option of holstering.

The major issue with using glue to hold the PCB in place is the fact that the glue offers no external protection for the device from debris or water. This could prove to be detrimental to the electronics of the SMART Skateboard after just one session. Not to mention that the glue itself could actually damage the PCB circuitry as well. Using tape or glue also has the possibility of losing adhesion over time, which will also in turn damage the device and leave the user unsatisfied.

The other glaring concern with using an adhesive to secure the device is the notion that the user will at some point want to purchase a new skateboard and will be unable to detach the SMART Skateboard device from the existing skateboard.

Using an adhesive such as superglue would force the user to have to purchase a new SMART Skateboard device every time he/she acquires a new skateboard. This would get quite expensive and tedious for the users of our product. Implementing this method would surely cause major dissatisfaction on the user's end.

Although this constant reacquisition of the SMART Skateboard device may sound like a major profit in the waiting, we believe the SMART Skateboard will become an impractical, ineffective product very rapidly due to the non-user-friendly nature of this holstering option.

Velcro

A very simple and non-complex method for fastening the SMART Skateboard device to the bottom of a skateboard is to use a couple strips of Velcro tape. The adhesive sides of the tape would be fastened to the bottom of the skateboard as well as the backside of the PCB. A big advantage of using Velcro strips is that it would allow easy attachment and detachment of the device from the board. Also this method would be easily implemented by the user as long as he/she was given the strips with the product.

However, there are a few downsides to using this method of holstering. First of all, the user has to be accurate the very first time with the strip placement given that there is only one chance to place the adhesive tape. Any subsequent attempts will pull the paint off of the skateboard and the given Velcro strips will have lost most of their adhesive qualities. This limited implementation is also a problem when the skateboarder needs to change skateboards due to wear and tear.

Another major issue with this method is device safety. The fact that all of the electronic components are not being protected can be a major problem if this product were to ever hit the market. If the user rides over the slightest puddle of water, he/she risks completely ruining the device simply due to the fact that the device is not being protected at all. This is not an ideal scenario to a consumer, seeing as skateboarders are frequently known for attempting dangerous tricks, sometimes over bodies of water. This may not be a major dilemma for the purposes of this senior design project, since the test environments will be strictly controlled throughout the semester and final presentation.

The other major issue that should be addressed using Velcro strips is whether or not they can keep the SMART Skateboard device in place after enduring the shock of a landed trick. It would prove to be very ineffective if the first time we test our finalized device, it falls off of the skateboard and breaks. This is a major issue that should be inspected as we go forward with the SMART Skateboard.

3D Printed Holder

The third method seems to make the most sense in terms of creating a secure, customizable holster for our electronic device. This much safer method (as far as the SMART Skateboard device's safety goes) is to use a similar method that the original developers, Syrmo, used to attach their device to the skateboard. This method is to create a plastic casing for the device that can be attached, using the skateboard's screws, between the trucks and the wooden board itself. Figure 14, shown below, is an image of Syrmo's device alongside an image of the device attached to skateboard trucks for clarification.

Figure 14 - One of Syrmo’s holster designs (left) as well as its integration into the board (right).

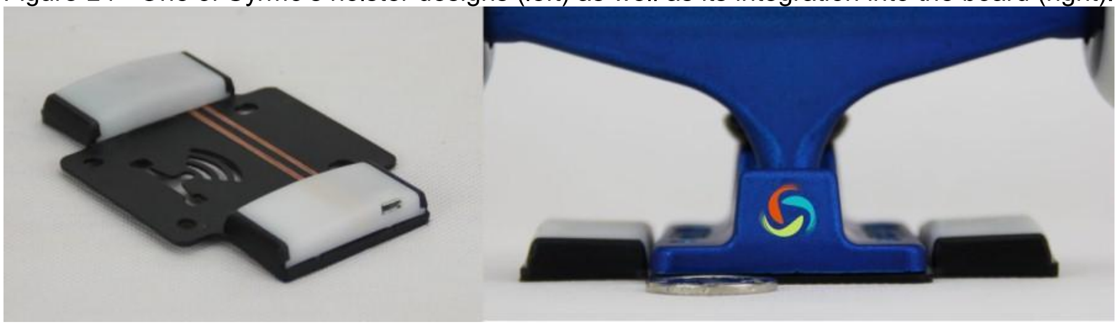


Image use permission from TechFastener.com is still pending

Although this is a tested and proven method for holding the device underneath the skateboard, there are a few complications as far as the SMART Skateboard goes. Initially, the main difference here is the fact that we do not have access to the same type of manufacturing facilities to create a high quality holster for the device. The SMART Skateboard device holster would have to be created using the 3D printers made available to us by the University of Central Florida. The problem with that is the fact that most of the materials used by these printers can be easily cracked or broken. It seems almost inevitable that a 3D printed holder would fail after repeated tricks being landed.

However, after some deeper insight, it seems as though UCF has multiple material options for their 3D printers. This possibility may actually make this holstering option a viable one. We need to ensure that the material we chose to create the holster of the PCB can withstand the force created by a skater landing a trick multiple times.

If the material can hold up to this force, then 3D printing a holder to be secured underneath the trucks seems to be the most realistic method for the SMART Skateboard.

One other possible problem with 3D printing the holder for our device is the possible size of our overall device. From the initial prototypes and research, it seems unlikely that our device will match Syrmo’s in spacial dimensions. This may prove to be a major issue if we try to design a holder to be attached under the trucks of the skateboard.

If the device is too large to fit underneath the wheels, then that could hinder the trucks’ range of motion causing the user’s skateboarding experience to be negatively affected. This method of securing the PCB seems to be a viable option as long as our device isn’t too bulky and the 3D printed material can sustain the impact of regular skateboard use.

Summary

The table below summarizes the different holster options we considered to attach the SMART Skateboard device to the bottom of the user's board and their advantages and disadvantages. The table below will summarize the information given above in a clear, concise manner to make the selection process easier.

Table 13 below will make selecting the proper method of holstering the device a much easier process. The senior design group will refer back to this table when creating a holster for the final SMART Skateboard device.

Table 13 - Holster Summary

Holster Option	**Advantages**	**Disadvantages**
Adhesive	• Simple to implement into the overall design. • Little risk of device falling off of skateboard.	• Adhesive substance may damage the printed circuit board. • Not water resistant causing possible damage. • Not shock resistant causing possible damage.
Velcro	• Easy to implement into the overall design of the device. • Easy to acquire. • Easy to remove the device for charging or new board.	• Not as secure as using an adhesive material. • Not water resistant causing possible damage. • Not shock resistant causing possible damage.
3D Printed Holder	• More secure than the other options. • Somewhat water resistant depending on design. • Somewhat shock resistant depending on the design.	• More difficult to implement than the other options. • Possibility of breaking if designed using wrong material type.

3.3 Relevant Software Research

The following section outlines all of the different mobile device options for our mobile app, also including the advantages and disadvantages of each in terms of the SMART Skateboard. The design portion of this document will discuss how the software will interpret the motions of the skateboard and then turn analogous information into simple and definable classifications.

3.3.1 Mobile Device Options

The mobile device is another key feature to fully bring the project to life. These days a robust mobile application is no longer just a nice addition to a product, it is a necessity. All new successful ventures are expected to have a mobile application if they are to fully succeed in the market.

This is why software development and user interface skills are so highly sought after. The mobile device must run a user interface that displays which maneuver was performed on the skateboard. Below we compare the three large device families. Each device family would fulfill the needs of the project so choosing one is mostly a matter of preference.

Android

A mobile operating system developed by Google. Based on Linux. First was a startup that wanted to make smarter operating systems for digital cameras. Was then acquired by Google. Used in smartphones and tablets. Also extended to Android TVs, cars, and watches. Android is the best-selling mobile OS in the world.

Advantages:

- Android skills are in demand so it is a good technology to learn and put on the resume.
- Also, Android has the largest global user base out of all the three big device families.
- Android coding is closely based on Java and since Java is the main object-oriented language taught at UCF, the team is adequately prepared to develop on Android.
- The team owns an Android device. This makes it easy for us to develop and test code at home.
- Most devices support Bluetooth.
- Some devices also support Near field communication.

Disadvantages:

- Low memory for storage, not good for storing many games and movies.
- Large apps might be forced to close.

- Lots of background processes can rack up data connection bill.
- Some devices have significant problems with batteries.
- Google Play store has problems with malicious applications being uploaded.
- Planned obsolescence. Every couple of years phone will need to be replaced.

iOS

A mobile operating system invented and developed by Apple. Used on devices such as iPhone, iPad, and iPod touch. Only bested by Android in terms of global sales. Very popular in western culture.

Advantages:

- iOS is a very in demand technology and would also be good to learn and put on the resume.
- iOS development is also more secure than its competition but this is not required for our purposes.
- In terms of getting a broad Western user base and app monetization, iOS is the way to go. The iOS user base is always ready and willing to pay for quality applications so this would be the ideal platform to launch the final version of a product on.
- The team owns an iPhone. This would make it easy for us to develop and test code at home.
- Most devices support Bluetooth.
- Apple is known for their crisp and clean user interfaces. This would make the user experience the best out of all the three device brands.
- Top of the line hardware means large memory and dependability.
- Apple constantly updates their software to fix bugs.

Disadvantages:

- One negative is that nobody in the group has experience in these languages so it would take longer to start developing.
- We are still in the prototype phase so we do not need to consider the size of the user base for our project.
- The iOS development environment is based on Swift and objective-C languages, none of which the group have experience with.
- Not open source, if something goes wrong cannot look at the source code to help.
- iOS devices are very expensive.
- Some devices have poor battery performance.
- No Near Field Communication support.
- Planned obsolescence, Apple wants customers to replace their phone every couple of years.

Windows 10 Mobile

A mobile operating system developed by Microsoft. It succeeded the Windows Mobile and Windows Phone 8.1 operating systems. Part of Microsoft's plan to consolidate all of their devices into one main operating system. Makes it easy for developers to make apps for all devices that Microsoft develops.

Advantages:

- The group has a lot of experience writing C# code on Windows platforms.
- Making the application for the project would be very simple in Visual Studio.
- Most people have grown up around the Windows desktop operating system so it will be more intuitive for the user.
- Microsoft has pushed the Windows mobile technology and has constantly improved it since Apple and Google started to dominate the market. Now the gap between all three device brands isn't so large.

Disadvantages:

- The Windows phone brand is in a state of uncertainty because it is not clear whether Microsoft will eliminate the brand as a whole. Windows development skills may lose their value rather quickly.
- Staying with the same technology all the time is not good practice so the group would rather learn a new technology.
- To make matter's worse, none of the team members own a Windows phone.
- Planned obsolescence. Need to replace phone every couple of years.

Summary

All three major smartphones have their own upsides and downsides as we've outlined in the preceding sections. In this section we summarize all three major smartphone brands and what they have to offer to our project. In Figure 15 shown on the next page we can see the difference between the familiar layout of Windows mobile, the robust layout of iOS, and the efficient layout of Android. Then in Table 14 shown on the next page we summarize the advantages and disadvantages of each major smartphone brand.

Figure 15 - Comparison of Smartphone UIs

Permission pending from Digiarty

Table 14: Comparison of Smartphones

Comparison of Smartphones		
Device Family	**Advantages**	**Disadvantages**
Android	• Largest global market share. • Java, the object-oriented language taught at UCF. • Readily available to the team. • Bluetooth and NFC	• Need to gain experience with Android Studio
iOS	• Largest U.S. market share. • Readily available to the team. • Secure • Best user interface • Apple always fixes bugs	• Team has no experience with Objective-C or Swift • No NFC
Windows Mobile	• Team has experience with C#. • Easy to develop in Visual Studio.	• Smallest market share • Future uncertain • Device not readily available to the team

4.0 Related Standards and Design Constraints

This entire section is dedicated to discussing the standards and constraints that pertain to our specific project. Below we discuss, in depth, the different standards that we will have to abide by when designing the SMART Skateboard. The standards are then followed by the relevant constraints that we can currently foresee happening in the design phase of this senior design project. There might very well be constraints that we have not yet thought of, but this is the list as we see it so far.

4.1 Relevant Standards

Bluetooth - IEEE 802.15.1:

IEEE 802.15 is a working group of the Institute of Electrical and Electronics Engineers (IEEE) 802 standards committee which specifies wireless personal area network (WPAN) standards. There are 10 major areas of development, not all of which are active. These standards differ by which frequencies they use and this affects the data rate and range they can cover.

Abstract: The IEEE 802.15.1 standard is the basis for the Bluetooth wireless communication technology. Bluetooth is a low tier, ad hoc, terrestrial, wireless standard for short range communication. Designed for small and low cost devices with low power consumption, this technology operates with three different classes of devices: Class 1 - (100m range), Class 2 - (10m range), and Class 3 - (1m range).

Similar to Bluetooth, Wireless LAN (WLAN or WiFi) can operate in the same 2.4 GHz frequency band, but the two technologies use different signaling methods to prevent interference. The application of WLAN however, has been most visible in the consumer market where a number of portable devices support at least one of the many variations (ie: IEEE 802.11a/b/g/n).

These wireless networks commonly have two distinct modes of operation, ad hoc & infrastructure. Infrastructured wireless networks typically have some sort of “base station” that acts as a central node; essentially connecting the wireless terminals. These base station are usually provided in order to enable access to a wide range of wireless networks. But in most cases, they are within a fixed location. The downside over ad hoc networks is that this fixed location can also be a central point of failure. If it stops working none of the wireless terminals can communicate with each other.

On the other hand, ad hoc networks can be formed “on the fly” without the help of a base station. Self-organization is the key to forming an ad hoc network because initially there is no central node to talk to. In ad hoc networks the wireless terminals may communicate directly with one another while terminals in infrastructure networks must use a base station to relay their messages. As previously mentioned, Bluetooth adopted this mode of operation and consequently this can affect the capabilities of our project in multiple ways.

Android App:

The topic of android app standards is very broad, Google inc has set a numerous amount of standards in place since the release of the android platform. Google has set standards for android in all departments, Including the optical creation, style, layout, usability, the components and patterns, the animation, security, as well as the compatibility with their own application store, the reliability of the software, as well as how the platform should execute in compatible devices.

Google, allows users to be as creative as possible when using the android platform, however certain things must never change, so that the usability of android can remain the same. They also have certain rules, such as not allowing users to use certain modified versions of the android robot for marketing purposes, or how the brand is represented. Figure 16, shown below outlines some of these standards.

Figure 16 - Android Standards

Navigation	UX-N1	App supports standard system Back button navigation and does not make use of any custom, on-screen "Back button" prompts.	CR-3
	UX-N2	All dialogs are dismissable using the Back button.	CR-3
	UX-N3	Pressing the Home button at any point navigates to the Home screen of the device.	CR-1
Notifications	UX-S1	Notifications follow Android Design guidelines. In particular: a. Multiple notifications are stacked into a single notification object, where possible. b. Notifications are persistent only if related to ongoing events (such as music playback or a phone call). c. Notifications do not contain advertising or content unrelated to the core function of the app, unless the user has opted in.	CR-11
	UX-S2	App uses notifications only to: a. Indicate a change in context relating to the user personally (such as an incoming message), or b. Expose information/controls relating to an ongoing event (such as music playback or a phone call).	CR-11

Use of this image was published by the Android open source project, our group is using it in accordance with the Android Open Source Project Rules

Lithium Polymer Batteries:

Our team is using Lithium polymer batteries to power our project, the decision to use these batteries makes a big difference in our product, and the standards tied with such batteries are very important to our design. The safety of the user as well as the durability of the product is our main concern. We need to keep in mind the standards and safety of the batteries at all times, from beginning to end. Our group should keep in mind to follow all standards and guidelines throughout the development of the project in order to prevent injuries, and have a successful end-product. Some of the Lithium battery standards are listed below.

- IEC 60086 : specification when utilizing lithium batteries on air vehicles
- IEC 61960 : Using Lithium battery cells for portable applications
- IEC T1828 : Recommendation by the United Nations on the Transportation of Lithium batteries

There are also standards on how to store Lithium batteries. They must be kept at a certain temperature in order to avoid explosions or mechanical failures. There are also standards on how to charge the batteries, in order to follow the battery specifications, to make sure they are charged at the correct rate. This ensures that we get the correct output performance, while being able to avoid unnecessary accidents.

4.2 Possible Design Constraints

The following section outlines the ABET design constraints that our group will undoubtedly have to face in order to achieve success with the SMART Skateboard project. These include economic, environmental, social and many other constraints that could cause problems later in the design process of this project. The ABET design constraints are then followed by possible challenges that we may face throughout our creation of the SMART Skateboard. Those are then followed by more specific hardware and software design constraints which may impede our development of this senior design process.

4.2.1 ABET Design Constraints

ABET design constraints will keep our project within a realistic scope. These include: economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability. We break each one down to specifics below.

Economic:

The product will be designed with appropriately costed materials and technologies such that it can compete in the wearable device market. If the price is too low our design and development cost would not be reimbursed and if the price is too high the market will not demand our product and we will have failed our ABET requirement. We will turn to the best distributors in the market to ensure we get quality parts at market price at a reasonable time.

Environmental:

The product will be designed with the environment in mind. Using the final product should not harm the environment in any way, shape, or form. Wherever possible, materials should be used that can be recycled or reused. No dangerous elements should be used in the design of this product. Our wireless communication should not cause harm to nearby wildlife.

Social:

The product should provide a benefit to society. The product will seek to meet the human need of self-actualization. It will help us grow our engineering knowledge while helping the end user track their performance and improve their skateboarding skills. The product should not alter the skateboard in such a way that it becomes unrecognizable from a social standpoint. The mobile app could feature social connectivity if we have time and in that case we would have to provide a safe social haven for our users.

Political:

The design, development, and use of the product should violate no United States laws and it should not harm the United States in any way. The product will not be marketed towards or against any specific race or gender. All copyright laws will be followed during the development of the project. All creative commons licenses will be abided by and cited where needed.

Ethical:

All of the work done in this project will be our own except that which is used under the appropriate licenses from the open source community. The product will be made with parts of appropriate quality. It will not be made at the lowest possible cost simply to be resold at a very high cost for personal gain. The product should work as we promised and as the user expects.

Health and Safety:

Using our product should not put the user at any extra risk than if they would have gone without our product. The product should not cause any potentially harmful wireless interference. It will use safe best practices of electrical engineering to make sure the electronics work as intended to minimize risk to the user. This is why it is important for the weight to be as small as possible so it will not hinder the natural abilities of the user.

Manufacturability:

The project will be designed with the current standards in manufacturing in mind. Our Eagle schematic will be designed such that any modern manufacturing facility would be able to manufacture the schematic. This means that components and traces will be appropriately scaled and distributed across the board.

Sustainability:

The product will be designed such that it will be easy to reuse in the future if future work needs to be done. The battery should be attached in such a way that it can be detached if it fails. The components will also be soldered on so if they fail they can be easily desoldered. The mobile application will be developed with standards for code reuse in mind so that the functions could later be used in other similar projects.

4.2.2 Possible Challenges

Listed below are possible challenges that the group will face due to our limited experience and time constraints. We list them out and detail each one to be better prepared for the challenging times that may lay ahead.

Please note that there may be additional, unforeseen challenges that we must deal with in order to make this project a success. This following section just outlines the possible challenges we may face looking ahead in the design process of this project.

Optimal PCB Design:

PCB design will be a big challenge because UCF does not offer a course teaching how to develop printed circuit boards. This means that we will be learning on the fly during this project in addition to a strict time constraint. We will have to learn quick to design a PCB that is cost effective and optimal.

Communication Protocols:

Another area where the team has limited experience is integrating multiple chips with a common communication protocol. Some chips use I2C and some use SPI and there are also many others. It is challenging to design some of these protocols from scratch.

Power Optimization:

The team also has limited experience in power electronics so this will be the first time we have had to limit the power drawn from a device. We will need to make sure the device is in a sweet spot in terms of power drawn, so we can meet the market requirement for long battery life while also keeping it realistic. Outputting too little power means the device will fail to turn on, while feeding too much power to the SMART Skateboard will cause serious, irreparable electronic failures.

Device Holster:

Nobody in the group is a mechanical engineer, so it will be challenging to design the most optimal way to attach our product to a skateboard. We will have to learn 3D printing and 3D printed wire traces if we want to attach the SMART Skateboard device to the board in the best location where it fully minimizes harm to the product.

Developing Code:

Most of the team is studying electrical engineering and have limited experience in deploying production level code that runs perfectly every time. The embedded code will have to flawlessly poll the sensors and send it over Bluetooth the the mobile device without losing any of the data. The mobile device will have to interpret the data without losing accuracy. All of this and displaying the results to the user in a robust and elegant way will be a new challenge to us all.

4.2.3 Hardware Design Constraints

Minimal Design:

To ensure the best possible design, the SMART Skateboard needs to be as small as possible. This means that both the hardware components of the electrical device as well as the holster itself need to be designed to be as small and light as possible. This is crucial to creating a device that is relatively unnoticeable by the user.

Having a heavy or large design will impede the user's ability to perform tricks, in turn rendering the SMART Skateboard device unusable. This is a constraint due to the sheer difficulty of designing a small device in comparison to having no limits on the dimensions of the device. The SMART Skateboard needs to be light and small in dimension in order to not restrict the user's ability to land a skateboarding trick.

Device Durability:

Another constraint of this design project is creating a durable electronic device that can withstand constant impacts endured throughout a skateboarding session. This is a serious constraint due to the fact that most electronic hardware breaks easily after just the slightest level of stress. Looking ahead, this constraint is most likely going to be the most difficult one to deal with in terms of hardware. From our little knowledge and experience in the field, the electronic device won't allow for much leverage in terms of creating more durability.

Ultimately, creating a durable device will lie mainly with the holster of the device. To allow for multiple failures, we will most likely need to create multiple electric prototypes. The durability of the electric hardware will be hard to alter, since we are not creating the components themselves rather ordering the parts from a company. This in turn means that the materials of each components are already predetermined and the durability is not easily altered.

Holster Durability:

Mentioned above, the durability of the device lies solely on the holster we design. The electrical components will be made of rigid silicon, and will break easily under stress. Therefore, the holster we design will need to not only be small enough to be unnoticed by the user, but also durable enough to protect the electric components of the device from the repeated wear and tear endured by the skateboard.

The sheer force of a 3-foot drop creates enough force to destroy our components. It is a major constraint to determine the right building material and design of the holster to absorb this force in order to fully protect the electric components which we have developed.

4.2.4 Software Design Constraints

Data Accuracy:

As we exchange data between devices, the IEEE standards described earlier also play a role with the accuracy of our throughput. The observation is that several standards operate in the 2.4 GHz band, including Bluetooth's IEEE 802 standard.

Along with these standards we also have common household appliances like microwave ovens and certain wireless phones which might congest the 2.4 GHz frequency spectrum. This unfortunately brings the dilemma of traffic within the spectrum band; causing these signals suffer from interference and noise. Ultimately leading to delays or false data exchanges within our devices.

Minimal Latency:

All wireless technologies have limits on how fast they can transmit data; faster connections generally mean higher energy consumption. Since BLE was designed to be very energy-efficient, it sends data relatively slow. This application was never intended to be a substitute for faster technologies such as Wi-Fi or USB. This is a major constraint as far as selecting the proper communication technology between the SMART Skateboard device and the mobile phone application.

5.0 Hardware and Software Design

The following section discusses the design plans behind creating the SMART Skateboard. Broken down into two main subsections, this section discusses all aspects of the hardware and software specifics of this project. Each subsection will discuss the chosen method for the design, detailed earlier in the research section of this document, the implementation of this design, as well as any images of our progress throughout the project.

5.1 Hardware

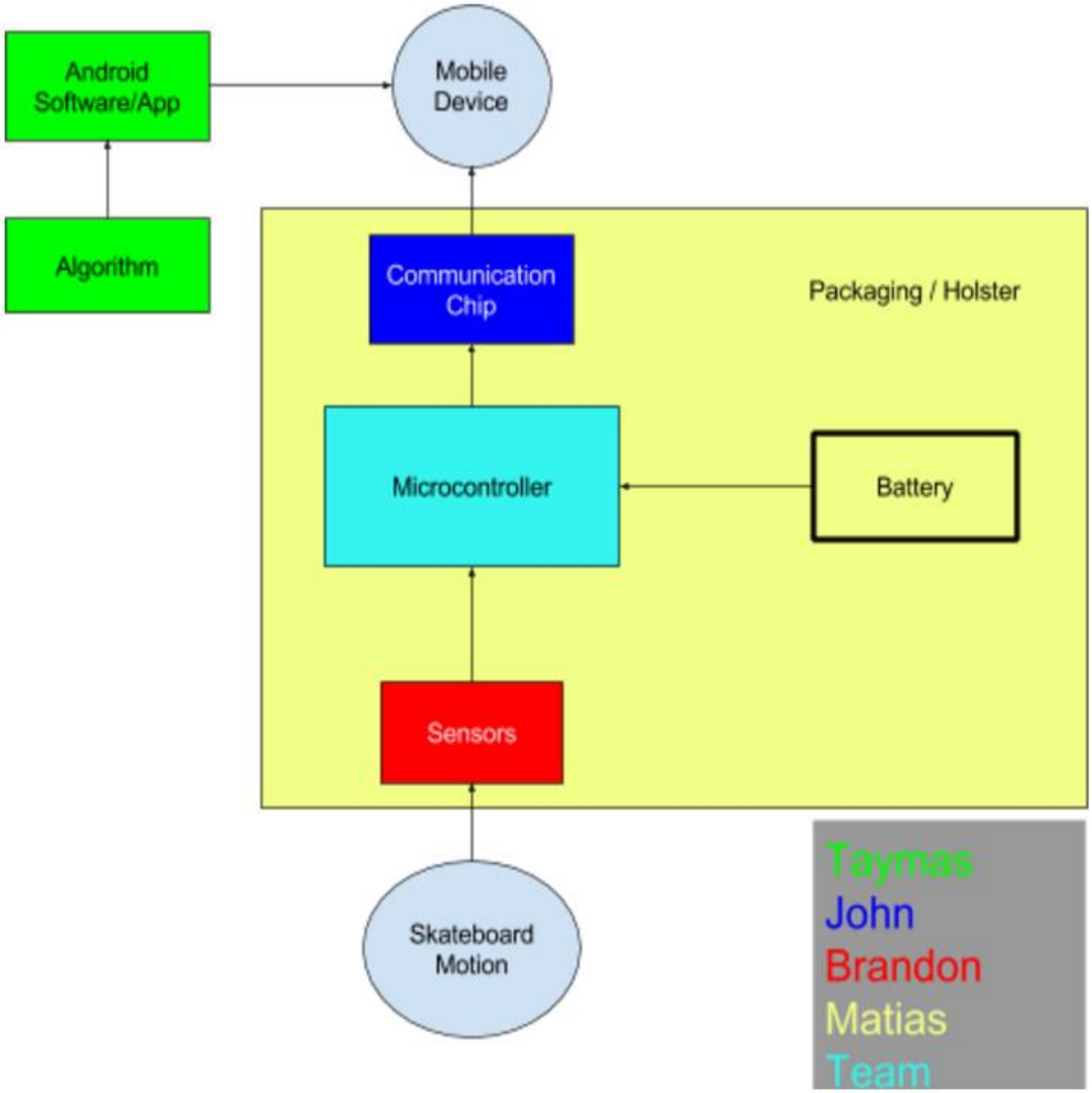
This section implements the design aspects of the hardware portions of our senior design project. We begin by outlining a basic block diagram which we will use as a reference for individual responsibilities for each team member as well as a reference towards the entire process of the SMART Skateboard. After the block diagram, we discuss the design plan for each component of the project in great detail. We begin by stating the selected option we chose in the respective section then explaining the implementation behind said option as it pertains to creating the SMART Skateboard.

In order to have a reliable device, all of our components need to meet our specified requirements detailed earlier in the beginning sections of the document. These specifications are imperative towards creating a sleek and fully functional design. The SMART Skateboard needs to have the best possible hardware at the most affordable price to ensure it will become a successful, desirable product.

5.1.1 Hardware Block Diagram

Figure 17 on the following page breaks down the hardware and software designs of the SMART Skateboard along with each group member's responsibility pertaining to the design. It should be noted that Taymas is in charge of the software parts of the design, which does not pertain to this section specifically. However, the rest of the diagram shows how we decided to break down each component and the respective team member responsible for each part of the device. As a quick summary of the diagram, John is responsible for the communication chip of the device, Brandon has the responsibility of finding the right sensors for the SMART Skateboard, Matias has the task of determining the optimal battery type and holster design of the device, and the entire team is responsible for creating the PCB microcontroller.

Figure 17 – Hardware Block Diagram



5.1.2 Microcontroller Implementation

Figure 18 on the next page shows the pin layout of the ATMega328P chip. Then Table 15 on the next page shows how we implemented the ATMega328P chip in our design. These figures will assist us in the printed circuit board design process by reminding us where our traces will need to go.

Figure 18 - ATMega328P Pin Layout (from datasheet)

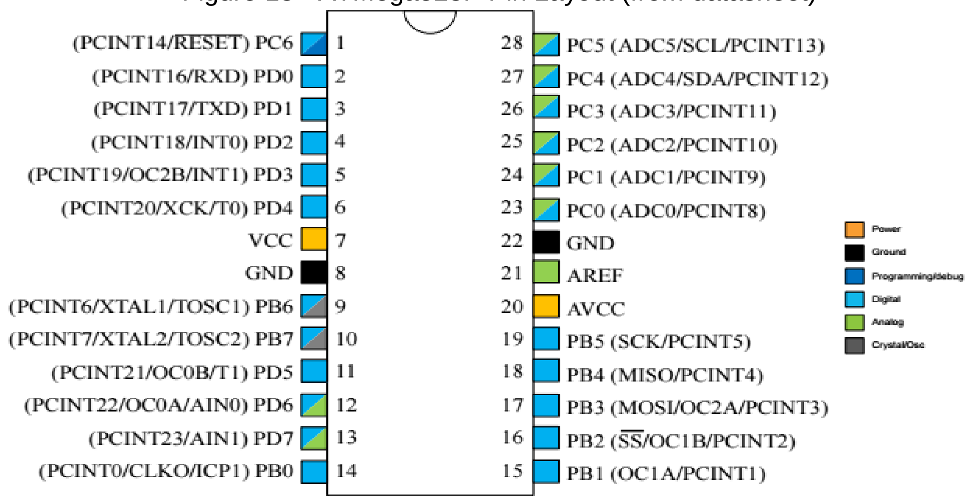


Table 15 - ATMega328P Pin Connections

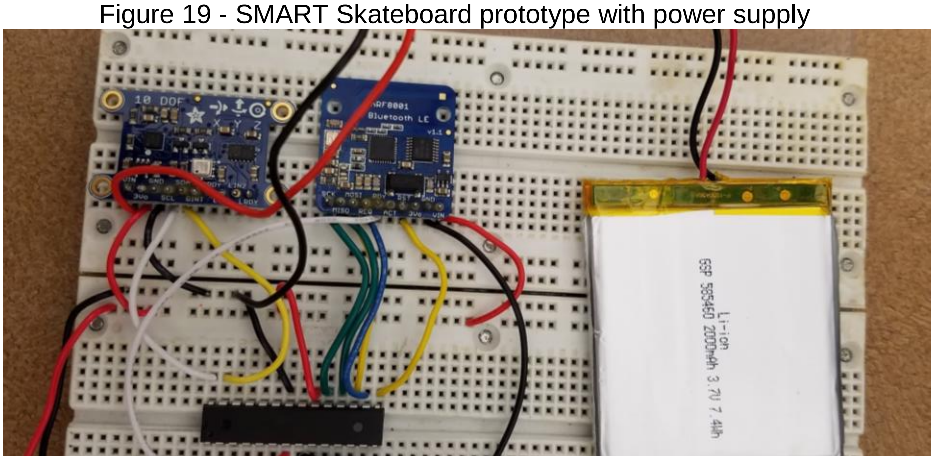
ATMega 328P Pin Connections			
Pin #	**To**	**Pin #**	**To**
1	n/a	15	BT RST
2	n/a	16	BT REQ
3	n/a	17	BT MOSI
4	BT RDY	18	BT MISO
5	n/a	19	BT SCK
6	n/a	20	Vin / Reg. Out
7	Vin / Reg. Out	21	n/a
8	GND	22	GND
9	n/a	23	n/a
10	n/a	24	n/a
11	n/a	25	n/a
12	n/a	26	n/a
13	n/a	27	Sensor SDA
14	n/a	28	Sensor SCL

5.1.3 Power Supply

The following section deals with the selection and implementation of the power supply for the SMART Skateboard. We will divulge into the selected power supply option of our design then the battery configuration of the design. With the selection of each hardware component, we will also discuss the reasons behind why we chose the specific option over the others as they pertain to our SMART Skateboard senior design project.

Figure 19 below, shows an image of the designed device connected to the chosen power supply option. This shows the battery type we selected, the connection method we selected, and also the way we wired the power supply into the SMART Skateboard device.

Figure 19 - SMART Skateboard prototype with power supply



5.1.3a Power Connection

The SMART Skateboard will be powered using the I/O pins on the main chip of the device. We decided to use the pins for a few important reasons. Although using this method disallows us to use the respective pins as outputs, it does allow our design to be much more concise, simple, and pragmatic. If we chose to use either the Japan Jack or the USB, then that would mean that we would need to keep a much larger section of the PCB together. Since we’re using the I/O pins, it allows us to take apart the original prototype PCB and only use the exact chips, sensors, and processors that we need specifically for the SMART Skateboard. More specifically speaking, the power supply of the device is connected to the Vin pin of the main chip. This connection runs through the onboard voltage regulators of the device before powering the main entire device. This regulation allows the electronics of the device to not be damaged by electrostatic discharge. Having the Vin pin power, the SMART Skateboard gives our design team much more freedom to construct a small overall product. This freedom allows us to work around the dimensional constraints discussed earlier in this report.

5.1.3b Battery Configuration

The six different battery types discussed in the above section have many advantages and disadvantages, however selecting the right one was quite simple. Due to the the perfect range of voltage outputs and the rechargeable nature of the batteries, we decided that the LiPo flat pack batteries are the best option in regards to the SMART Skateboard. These batteries output anywhere from 3.7 Volts to 22.2 Volts. We should have no issues finding a flat pack that can power up the 5 Volt input power pin found on the PCB. This means that we will not need to take the batteries apart and configure them in a series connection to optimize the output voltage. Since the range of these batteries is so diverse, there is no need buy multiple batteries and connect each one to one another. One battery would get the job done, as long as we chose a LiPo flat pack with a high enough output voltage to power the 5 Volt input power pin.

Another reason behind selecting this battery is the fact that it can be easily and quickly recharged with a wall plug. This means that the user will never need to worry about replacing the batteries of the device unless there is a mechanical failure. The last and possibly most important distinction of this option is the actual shape of these batteries. The flat nature of this battery allows us to discretely implement this battery into the SMART Skateboard without compromising the size and weight of the overall device. The LiPo flat pack is the best battery option for the SMART Skateboard.

5.1.4 Sensors

Here, we talk about the design of the specific sensors that we will implement into the SMART Skateboard’s main electronic device. We will first talk about the selection of the sensors that we made after extensive research and deliberation. Following the detailed selection, we include the planned methods of implementing each sensor into the device. These steps are critical to the design of the SMART Skateboard, and without the proper sensors being selected we would face many unavoidable and unnecessary dilemmas.

5.1.4a Sensor Selection

While it might sound simple, selecting the correct sensors for our project was one of the hardest tasks for our team, since we all have had very limited experience when it comes to designing a new electronic device. What we first did as a team was gather ourselves to brainstorm, and from there we decided what features we are going to be integrating to our project, after we decided all the features needed, we started researching what sensors have the capability of performing such features, and up to what extent.

After long hours of searching the web our team had gathered all the information needed to determine the sensors we would be using. In the following text we dive into the different selections made for each respective sensor as well as the specific reasoning behind choosing each sensor.

Accelerometer:

We decided we need an accelerometer in order to measure the speed at which our users would be going while performing the trick.

Even though the team had enough information to tell an accelerometer sensor would measure the speed and acceleration of the user, we had little information on how it would collect such data or how to program it, so the team decided to look for devices that use an accelerometer in order to deliver to it's users the information we need, doing that took us to the right path and we now have enough information to program the accelerometer to perform the actions we need.

After searching and doing research on a variety of accelerometers, we had to select the right one for our project, this accelerometer must meet the criteria of being as small as possible, it should record data accurately, and it has to be easy to program, and easily available in the market. That way our team would make sure we are able to deliver a product that is working properly and on time. After much consideration we decided to use the LSM303DLHC accelerometer from STMicroelectronics, this is a high performance accelerometer and magnetometer, it is the perfect fit for our project, since it has all we are looking for and more to offer. The LSM303DLHC is of high performance and is ultra-compact with measurements of 3x5x1mm. This accelerometer is incredibly helpful, since despite of its extremely small size is packed with amazing features our team can use to our advantage. The accelerometer is capable of going on sleep mode when it detects it's not in use, which saves our team a tremendous amount of energy, and allows us to provide the user with longer battery life.

Features: LSM303DLHC has all the features needed to complete our design

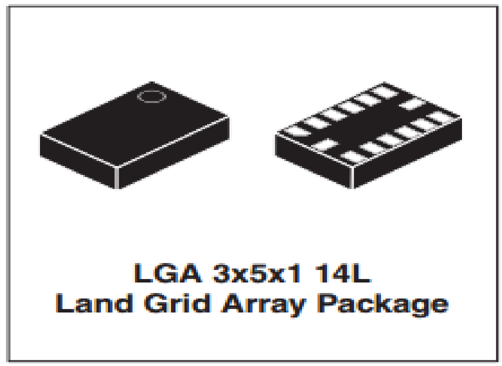
- Contains 3 magnetic field channels as well as 3 acceleration channels for high accuracy
- High magnetic field range of +-1.3 to +-8.1 gauss full scale
- Power-down mode for battery saving and higher user satisfaction
- Contains a pair of independent programmable interrupt generators that detect free-fall as well as motion
- Has an embedded temperature sensor
- Position detection
- Activates on motion

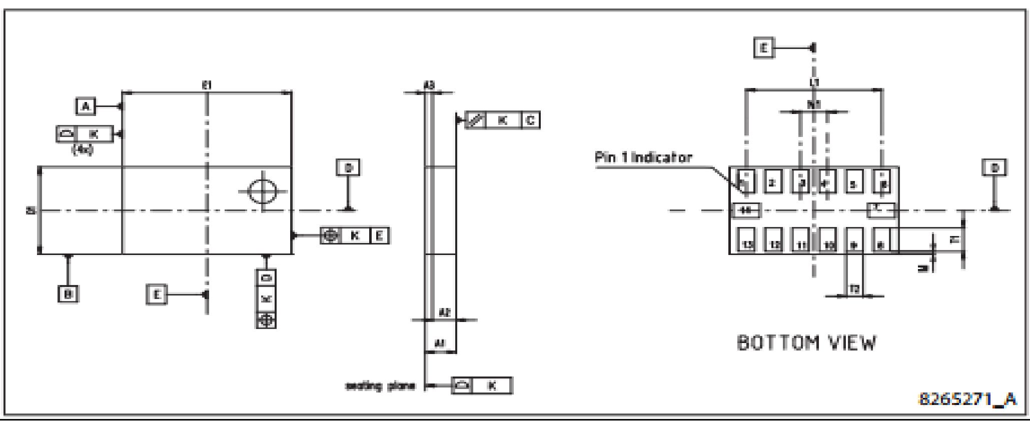
Figure 20 below clearly shows the dimensions of the accelerometer we chose, as well as the ports and the functionality of them. The mechanical data is of use for us, because it helps us understand more the sensor, and it makes it easier for the team to be able to implement it on the PCB without running into problems.

Figure 20 – LSM303DLHC Dimensions

Ref.	Dimensions (mm) Min.	Typ.	Max.
A1			1
A2		0.785	
A3	0.16	0.2	0.24
D1	2.85	3	3.15
E1	4.85	5	5.15
N1		0.8	
L1		4	
T1		0.8	
T2		0.5	
M		0.1	
k		0.05	

Outline and mechanical data





The permission for the use of this diagram has been requested from STMicroelectronics and it’s pending.

Gyroscope:

Selecting the Gyroscope sensor was a hard task for the team, since we did not have previous knowledge of what a gyroscope could accomplish, but since we knew we needed to be able to measure the position of the board and the flips we started looking at sensors that couple possibly accomplish such a task. After a long time searching we ended up with a couple of sensors that could potentially work, but seen similar projects helped our team determine a gyroscopic sensor would be in our best interest.

Since we were new in the world of the gyroscope, it was difficult to select a specific sensor, so in our research we found more information on it, such as the dimension of the sensor, something really important to our team, since we need to keep the printed circuit board to a certain size. We were impressed with what a small sensor could do for our project. Another requirement we had to keep in mind while selecting the gyroscope was how much current it was going to draw, we need to keep it at a minimum, in order to be able to avoid overheating our board, we also need to make sure we are providing our customers with the greatest satisfaction, and we believe we could accomplish that by designing our device so that it consumes low power and provides the user with long battery life.

Something else to keep in mind when selecting the gyroscope was the compatibility with the rest of the devices we had already chosen, as well as the code those other devices would be using. We made sure the gyroscope we selected was going to be easy to integrate with the other components, and that our team is going to be able to program it with no issues, and at the end we will have a working product. Having that in mind we selected the L3GD20H 3-axis gyroscopic sensor, since it provides the team with the necessary features to accomplish our goals.

We knew when selecting the L3GD20H that we would have no issues overheating, since it uses very low power, we also knew when selecting this specific gyroscope, we will have no problem programming it, since it is compatible with the other components we are using, and it is a sensor that is widely used by hobbyists, especially those who use Adafruit, there is a blog with code we can use a reference, in order to be able to code our device.

Having sample code to study and read is a big help to the team, because we would not have the need to start from scratch, and if we come to a stop where we need to ask questions there is the blog to help us figure out certain problems.

As some of our previous parts, the L3GD20H is capable of going into sleep mode when not in use, this is the feature that allows it to conserve energy and defines it as a low-power consumption device. Another feature that was a determining factor while selecting the sensor was its ability to withstand high shocks.

We need it to be able to have such feature because our device needs to be designed for heavy use, since our users will be attempting all types of tricks, at different heights, our team needs to make sure the device is ready for the worse shocks possible. Making our device reliable and durable is a big plus, and a must, and having a gyroscope that can offer this gives the team peace of mind.

Features: LSM303DLHC has all the features needed to complete our design

- Very high accuracy for the user to determine trick changes
- Low power consumption
- Contains 3 full-scale selectable of up to two thousand dps
- Can output data at a rate of 16-bit
- Has 2 integrated line, 1 for the interrupt the other for data ready
- Designed to withstand high shocks
- Embedded temperature sensors to prevent overheating
- Wake-on-shake compatible

The table below, shows the parameters and conditions of the LSM303DLHC gyroscopic sensor. Such table was to great advantage when the team was in the sensor selection process. It made it less complicated to select the gyroscope, since we knew what we were specifically looking for, and the table provided us with additional information that could be used to our advantage.

The datasheet below shows tests that we have not been able to accomplish as of yet since we do not have the required equipment not the board done. This is information that is imperative to the project, and knowing such information guarantees our team that choosing this gyroscope as one of our main sensors is an excellent choice.

Having test results that we are not able to acquire ourselves is an important and necessary step in configuring the main device to accept the external hardware. We need to know what the failure rates of the gyroscope are without actually destroying the sensors themselves.

The sensitivity to movement as well as the rate noise density if perfect for the task we are trying to accomplish. Knowing such data helps us further the project and make other important decisions.

The implementation of this gyroscope is not going to be as easy as other gyroscopes with leads, because it will make the soldering process harder, however, with all the features it offers, we decided we must select it as our sensor.

Table 16 on the following page breaks down the different aspects of the gyroscope we selected (LSM303DLHC). This table will immensely help our group when we begin configuring the final version of the SMART Skateboard.

Table 16 – LSM303DLHC Details

Symbol	Parameter	Test condition	Min.	Typ.(1)	Max.	Unit
FS	Measurement range	User selectable		±245 ±500 ±2000		dps
So	Sensitivity			8.75 17.50 70.00		mdps/digit
SoDr	Sensitivity change vs. temperature(2)	From -40 °C to +85 °C Delta from T = 25 °C		±2		%
DVoff	Digital Zero-rate level	FS = 2000 dps		±25		dps
OffDr	Zero-rate level change vs temperature(3)	FS = 2000 dps		±0.04		dps/°C
NL	Non linearity(3)	Best fit straight line		0.2		% FS
Rn	Rate noise density(3)	BW = 50 Hz		0.011		dps/($\sqrt{Hz}$)
ODR	Digital output data rate(3)			11.9/23.7/ 47.3/94.7/ 189.4/ 378.8/ 757.6		Hz
Top	Operating temperature range		-40		+85	°C

The permission for the use of this Table has been requested from STMicroelectronics and it's pending.

Barometer:

One of the features our team discussed we wanted to add to our device is the ability of detecting how high our user jumps. We know we wanted the feature, however we did not know how to accomplish such a task. Even though we knew it was possible based on the fact that other devices on the market are able to accomplish the same or similar task, we were not aware what type of sensor could be used to measure height, or what information that sensor will be using to generate such information.

After researching similar projects, we came to the conclusion that using a barometric sensor was the best for our own project, since it collects all the necessary data in an extremely accurate way. Accuracy is our number one concern when it comes to the entire design of the device, and having a barometer that is as accurate as possible is our priority. Users would not be interested in using our device if we are not able to deliver a product that provides the most accurate information in the market, since the whole project would lose its purpose, users would not want to use our device if they perform a trick and the data they are receiving says otherwise, that is why we are working to make sure we meet the highest standards possible.

We will be using The BMP180 barometric pressure sensor from SparkFun, since it offers very high accuracy, all while consuming low-energy, low noise and staying at a reasonable size. The BMP180 is widely used for Leisure and sports, which gives us confidence that it is a good choice to add to our project. Aside from consuming low power and being of small size 3.6mm x 3.8mm, the BMP180 also comes fully calibrated and it is compatible with the other components we have already selected. This barometric sensor was designed with mobile application in mind, so it is ready to be implemented in mobile devices, such as GPS, phones, and PDAs, now we will be implementing it in a new device, which will help us measure the height our users are able to achieve.

The only downside to the BMP180 is that it does not have leads, which makes its implementation to the printed circuit board more difficult, since soldering is not as easy.

Features: BMP180 has all the features needed to complete our design, as well as more to offer

- Low power consumption
- Fully calibrated
- High accuracy
- Can output data at a rate of 16-bit
- Withstand high shocks
- Low noise
- Supports standby feature

Table 17 on the following page shows how accurate the BMP180 barometric pressure sensor is, it can give us detailed information on the accuracy depending on the condition, it also provides us with the voltage usage and the range we are allowed to stay in. it is good to have this information when we get to the Printed circuit board design, since these requirements are going to make a big difference in what we can and cannot do, having the information in the table below helps the team make the right choices and prepares us when it comes to the implementation.

We will need to keep in mind the number below, because all the devices being used in our project must be compatible, and we need to have the right current going to the correct places, having the numbers below allows us to know how we should divide the current throughout the board, and which components should be getting more or less voltage, that way we would take precaution not to burn any devices, or not to create excessive heat.

Table 17 – BMP180 Details

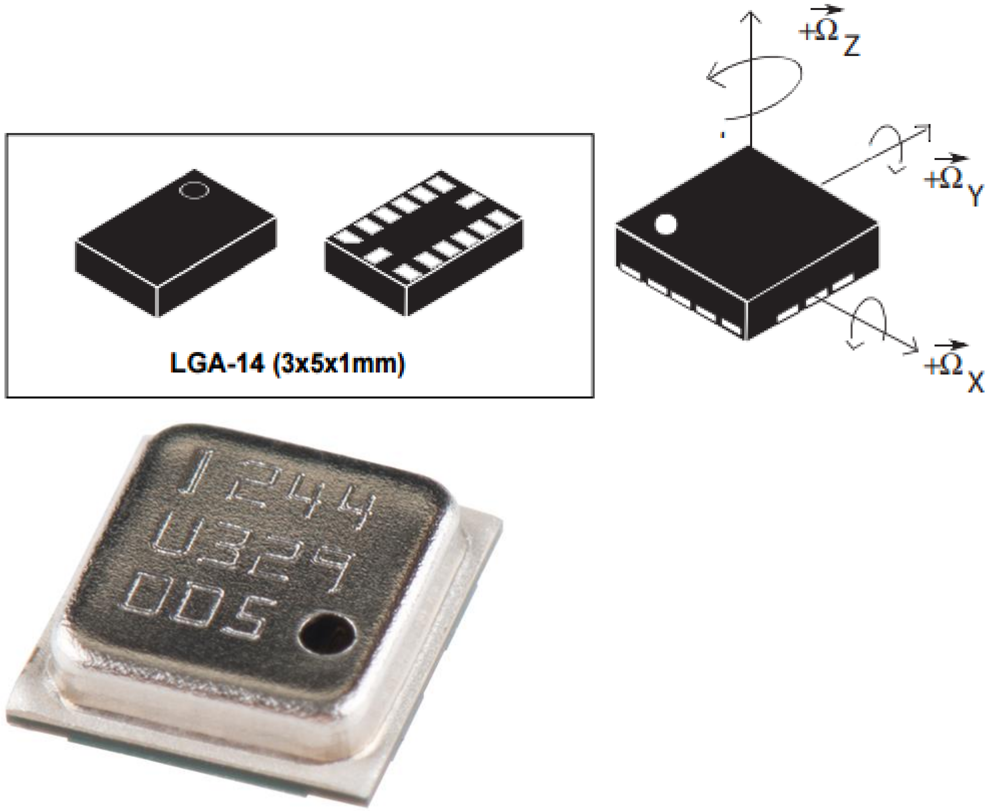
Parameter	Symbol	Condition	Min	Typ	Max	Units
Operating temperature	T_A	operational	-40		+85	°C
		full accuracy	0		+65	
Supply voltage	V_{DD}	ripple max. 50mVpp	1.8	2.5	3.6	V
			1.62	2.5	3.6	
Supply current @ 1 sample / sec. 25°C	I_{DDLOW}	ultra low power mode		3		μA
	I_{DDSTD}	standard mode		5		μA
	I_{DDHR}	high resolution mode		7		μA
	I_{DDUHR}	Ultra high res. mode		12		μA
	I_{DDAR}	Advanced res. mode		32		μA
Peak current	I_{peak}	during conversion		650	1000	μA
Standby current	I_{DDSBM}	@ 25°C		0.1	4[1]	μA
Relative accuracy pressure V_{DD} = 3.3V		950 . . . 1050 hPa @ 25 °C		±0.12		hPa
				±1.0		m
		700 ... 900hPa 25 . . . 40 °C		±0.12		hPa
				±1.0		m
Absolute accuracy pressure V_{DD} = 3.3V		300 . . . 1100 hPa 0 . . . +65 °C	-4.0	-1.0*	+2.0	hPa
		300 . . . 1100 hPa -20 . . . 0 °C	-6.0	-1.0*	+4.5	hPa
Resolution of output data		pressure		0.01		hPa
		temperature		0.1		°C
Noise in pressure		see table on page 12-13				
Absolute accuracy temperature V_{DD} = 3.3V		@ 25 °C	-1.5	±0.5	+1.5	°C
		0 . . . +65 °C	-2.0	±1.0	+2.0	°C

The permission for the use of this Table has been granted by SparkFun.

5.1.4b Sensor Integration

Our project consists of three main sensors, an accelerometer, barometer, and a gyroscope, the accelerometer will measure the speed/acceleration of the user, the barometer will measure the height of the jump, and the gyroscope will measure the skateboard's position. We will be integrating these three sensors using a printed circuit board, which will also include a voltage regulator, since the voltage range is not the same for all the sensors, we have discussed keeping the printed circuit board at a minimum size, which will make the integration of these sensors a bit more challenging. Figure 21 below are a representation of how our design will be seen, these sensors are small enough to stay between our original size restraint goals, these are from upper left to right, the accelerometer, the gyroscopic sensor, and the picture on the bottom left is the barometric pressure sensor, all of them smaller than a quarter of a dollar.

Figure 21 – Dimensions and Axes of Sensitivity



The permission for the use of this images have been granted by SparkFun.

5.1.5 Wireless Communication Design

This section discusses the selection and integration of the wireless communication chip for the SMART Skateboard device. This section is also broken down into a selection portion followed by an implementation process that we plan on following during the final creation of the SMART Skateboard much like the other sections of the design section of this document.

5.1.5a Wireless Selection

After close examination and a few cross-comparisons, the group unanimously decided to go with Adafruit's Bluefruit LE (Bluetooth Smart, Bluetooth Low Energy, Bluetooth 4.0) nRF8001 Breakout. The breakout version of this module is not only inexpensive, but it also simplifies our PCB design to a certain degree. This board allows the user to create an easy wireless link between devices, and is compatible with your conventional iOS or Android (4.3+) applications. The nRF8001 breakout works by simulating a UART device beneath the surface that sends ASCII data back and forth between devices, and lets you decide what data to send; giving you the leeway to do with it on either end of the connection.

Typically, classic Bluetooth have lengthy contracts to sign and major hoops to jump through in order to create iOS/Android peripherals that you can legally design and distribute in the App Store. BLE on the other hand avoids these extensive restrictions, making it a great choice in comparison. And now that Android officially supports Bluetooth low energy (as of Android 4.3), it's also a universal communication channel covering the main mobile operating systems consumers use today.

The nRF8001 module all together is an attractive product for just a BLE 'peripheral' (client) front-end; being that you can use any microcontroller with SPI to drive it. However, since it is not a stand-alone module, some microcontroller is required. With that being said, all the pins you need are broken out on the bottom of the PCB and are all 5V compliant. So you have the option to use this peripheral with 3V or 5V micros.

Figure 22 – nrF8001 Pinouts (our breadboard)

Pin layout (starting from left to right):

- *SCK* - this is the SPI data clock pin, connect to your SPI master clock out
- *MISO* - this the SPI data out pin, data is sent from the module on this pin. Data level is 3V but that is fine for 5V microcontrollers.
- *MOSI* - this is the SPI data in pin, data is sent to the module on this pin.
- *REQ* - this is basically what the nRF8001 considers the 'SPI Chip Select' pin, it's an input
- *RDY* (ready) - this is the data-ready pin, an interrupt output from the breakout to the microcontroller letting it know that data is ready to read
- *ACT* (active) - this is an output from the module, it lets the host know when the nRF8001 is busy
- *RST* (reset) - this is the reset pin input.
- *3Vo* - this is the output from the onboard 3.3V regulator, you can grab up to 100mA from this pin.
- *GND* - common ground for data and power
- *VIN* - 3-5 VDC input to power the breakout

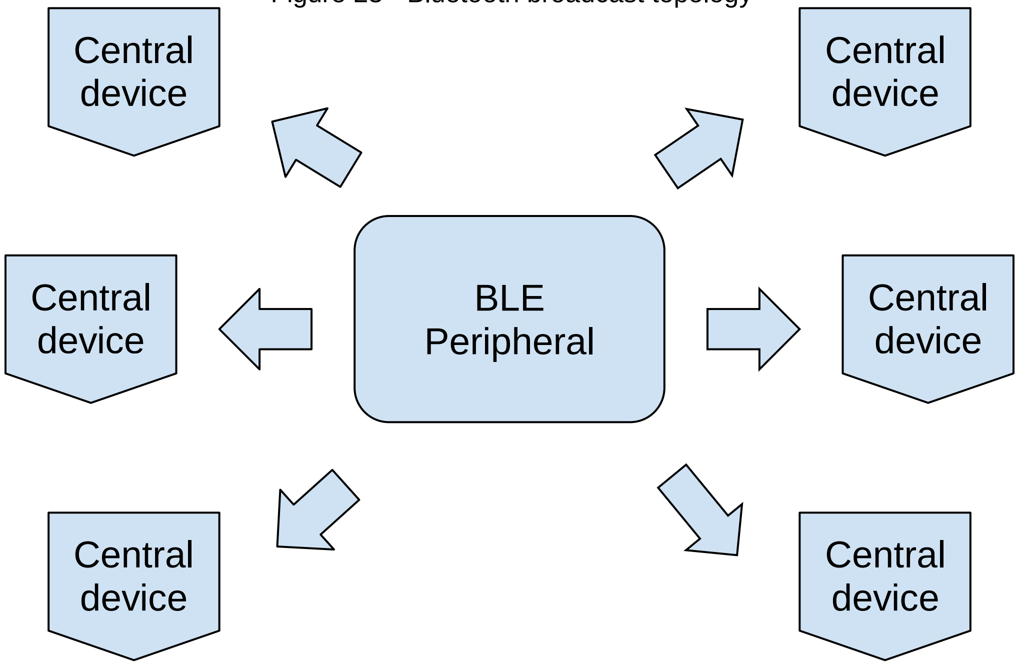
5.1.5b Wireless Integration

BT Generic Access Profile:

The Generic Access Profile, better known as GAP, is what makes a Bluetooth device visible to other devices. The GAP basically controls the connection and advertising in Bluetooth low energy; as well as being responsible for deciding device interaction. Scan response payload and advertising data payload are the two ways that the Generic Access Profile can communicate. Both payloads are similar, however the advertising data payload is a bit more critical to the profile. Essentially there are two main roles in the Generic Access Profile: central devices and peripheral devices. Advertising data payload is a vital aspect because it needs to be transmitting at all times so that the central device is able to recognize the peripheral device. On the other hand, the scan response
payload is optional and is used to fit more information within the advertising payload. Typically, a string consisting the name of a device would fall under the scan response payload.

Additionally, the Generic Access Profile's peripheral device is able to connect to more than one central device. Broadcasting in Bluetooth low energy is when a peripheral device sends data to other central devices within range. Figure 23 on the following page gives a visual representation of this concept. Since the data sent/received can only be seen by two connected devices, this type of configuration can only be possible by using the advertising packet.

Figure 23 - Bluetooth broadcast topology



BT Generic Attribute Profile:

As previously mentioned, Bluetooth technology requires at least to devices. While one device will be transmitting data, the other will be receiving data. Now in order to understand how and why these devices need to switch roles, this section will cover the Generic Attribute Profile commonly known as GATT. The Generic Attribute Profile clarifies how Bluetooth low energy or Bluetooth Smart devices are able to send data back and forth.

A typical central device could be a phone, computer, or a tablet, where as an example of a peripheral device could be a Bluetooth speaker or headphones. Essentially you can expect a central device to have more processing power and memory while the peripheral devices are considered to be low power and resource constrained devices. The underlying difference among devices is that a central device can be connected to multiple peripherals, and peripheral devices can only be connected to one central device at a time. Once both devices are successfully paired, the communication between these devices can take place in either direction. However, in order to exchange data between two peripheral devices, a Bluetooth “mailbox” system is needed to run the data through the central device.

Additionally, the Generic Attribute Profile differentiates the peripheral device as the “GATT Server” and the central device as the “GATT Client.” The purpose of the Client is to send out requests to the Server, while the Server looks up this data sends back a follow-up response. Having a slave/master device interaction. In most cases, the central device usually initiates the request to pair with the peripheral device, but to avoid causing any noise or interference with other devices, the central device uses a method to to limit its radio transmission as it scans for a connection. Another practice used to limit radio transmission is to utilize short intervals to accomplish faster detection/connection speeds. The only downfall to using this method is the fact that it uses a much greater power consumption.

On the contrary to accelerated connection speeds, slave latency tends to be limited at times in order to conserve some of that power consumption on the GATT Server. Slave latency is what defines how many times the server can ignore a consecutive connection. If this happens, it means that the peripheral device is not sending back data to the central device; allowing it to remain in sleep mode for a longer period of time.

Allowing BT Connection:

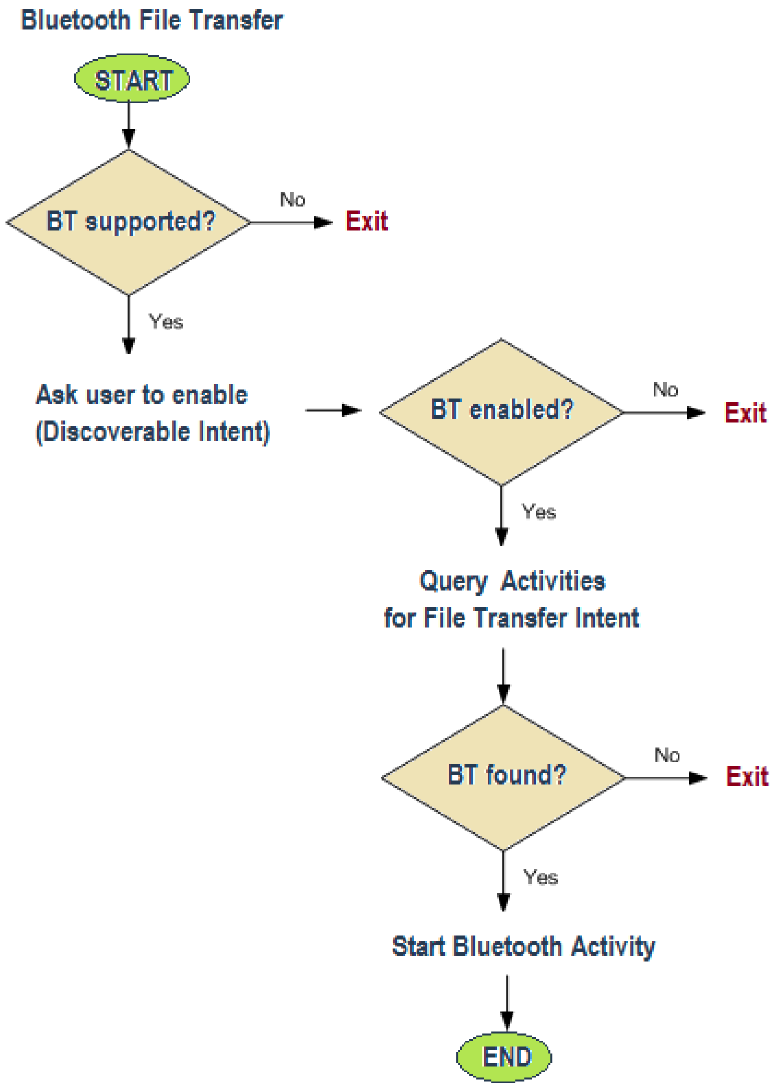
Another critical aspect to making a Bluetooth connection, permission must be granted by both devices. Requesting a connection, transferring data or even accepting a connection depend upon this permission. As our SMART pad is turned on, an established mobile device is needed to make that connection. All this is done through microcontroller.

Our peripheral device (SMART Pad) will broadcast an advertisement signal for the central device (Android Mobile Phone) to detect and make that connection. Once a connection has been made, another signal is sent from the SMART Pad to authorize a successful connection.

After the devices have paired, the peripheral will implement a single device configuration. This is an advantage to our project because single configuration consumes less power than the network processor configuration. Single device configuration is also the most common and easier method to apply. Since scanning for an external device can drain the battery, we will need to set up a timer to avoid this issue.

Furthermore, if the SMART Pad doesn’t have any data to transmit, it should skip through the number of connection acknowledgments and go into sleep mode. Ultimately, extending the battery life of our design.

Figure 24 – Bluetooth Permission Process



Permission pending from Java Code Geeks

The diagram above runs through the basic steps of the permission process and file transfer. As noted before, there are a network of connection acknowledgements in place prior to starting the Bluetooth activity.

5.1.6 PCB Design

The printed circuit board is the most important part of the project, since it serves as the brain of the operation, we are designing the printed circuit board to be as efficient as possible, not only size wise, but we are making sure that our design fits our budget, since depending on the design the circuit board could end up costing our team more or less money. Designing the printed circuit board is not an easy task, since none of our group members had previous experience designing circuit boards, we have had to start watching video tutorials on how to use EagleCAD and how to design a printed circuit board.

We have decided to make our printed circuit board a 2-layer board, since it should be sufficient to run the lines our project needs, that way we can keep it at a low price, while still having it perform as needed.

The top layer will be used to place the components, such as the gyroscopic sensor, the accelerometer, as well as the barometer, we will also be adding a voltage regulator and some form of power supply input jack, we will also be running some of the lines on the top layer of the board. The bottom layer will be strictly used to run those lines which could not be ran on the top layer, due to conflicts or intersection between lines, which is not allowed because it would cause a short circuit.

5.1.6a PCB Software Design Selection

Our team researched all the possibilities when it comes to the software we could be using to design our printed Circuit board. There are many good candidate software that we could have used for our design. However, we decided to choose EagleCad, we had many reasons on doing so, but one of the main reasons is the amount of tutorials available to us is more than any other software. EagleCad is simple to use, after watching a couple of videos and spending a few hours learning our way around the program, we were able to complete simple tasks, and a few weeks after installing the program the team was able to manage the program in a way that creating the schematic became a simple task.

We obtained the student version of EagleCad free of charge, a big plus since it helps keep our budget as low as we initially set it to be. The tutorials we assessed were also free of charge. Many of the features offered by EagleCad make the designing of the printed circuit board a lot easier than it would be if we had to start learning from scratch. The software recommends some of the best printed circuit board manufacturers in the market, it also generates a file with the bill of materials when the project is complete, this saves the team many hours of work that it would take to figure out the materials needed to put the board together.

If for any reason the team runs into trouble trying to wire the board Eagle has a feature that could assist us finding where the error is located, something that could also save the team time and money, many software in the market do not recognize many of the mistakes starters like us make, and many boards are printed with mistakes, cause the team money and time, that could put any team in trouble, especially if running out of time. EagleCad was the perfect choice, not only because it is not complicated to use, but also because every manufacturer we have looked into accept the files generated by eagle, which gives the team peace of mind.

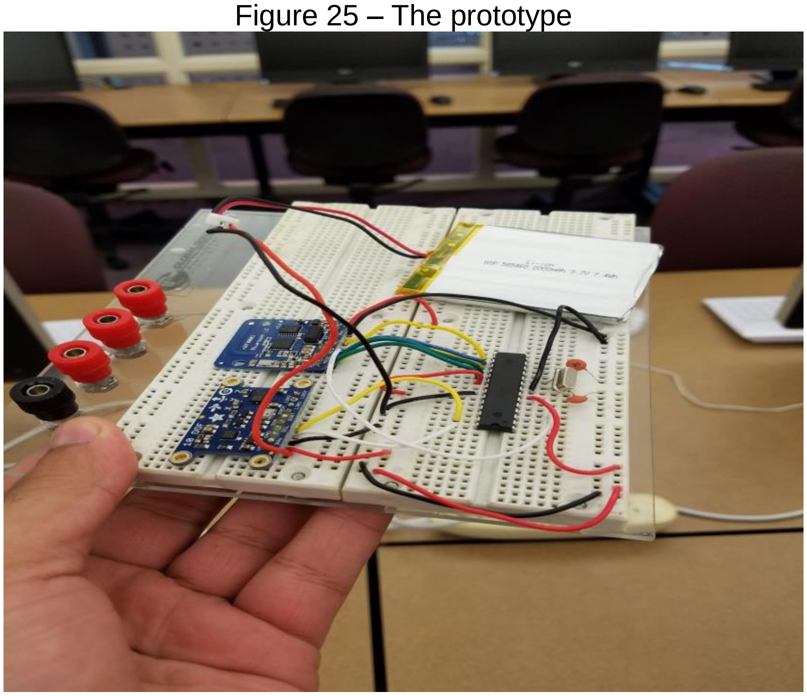
5.1.6b PCB Circuit Design

Our team knew coming in that the circuit design was going to be the hardest part of the entire project, therefore we immediately started to research what parts we would need in our circuit and we made sure every single part is available in the market. After the research step, we went ahead and ordered all the parts we thought are necessary for the initial tests, and started the breadboarding process. We successfully completed the circuit in a breadboard and were able to test the components. One of the challenging tasks of circuit design was the implementation on the components, many of the components use different voltage than others, therefore we had to keep in mind to add a voltage regulator when we were at the circuit design stage.

After successfully being able to breadboard the components, the team took to the task of testing the communication between the breadboarded devices and our android phone, something we were able to do without a problem as well. The picture below shows our circuit design, the breadboard of components while we were testing the functionality of the gyroscope via Bluetooth, while in the process we were getting information as we changed the position of the board, indicating the main function of our project has been taken care of.

Figure 25 below is a Bluetooth breakout board from Adafruit, connected to the Adafruit 10 degrees of freedom and the chip from an Arduino Uno, we are using a lithium polymer battery to power all the devices, we have it connected to a voltage regulator to make sure we do not burn out our components.

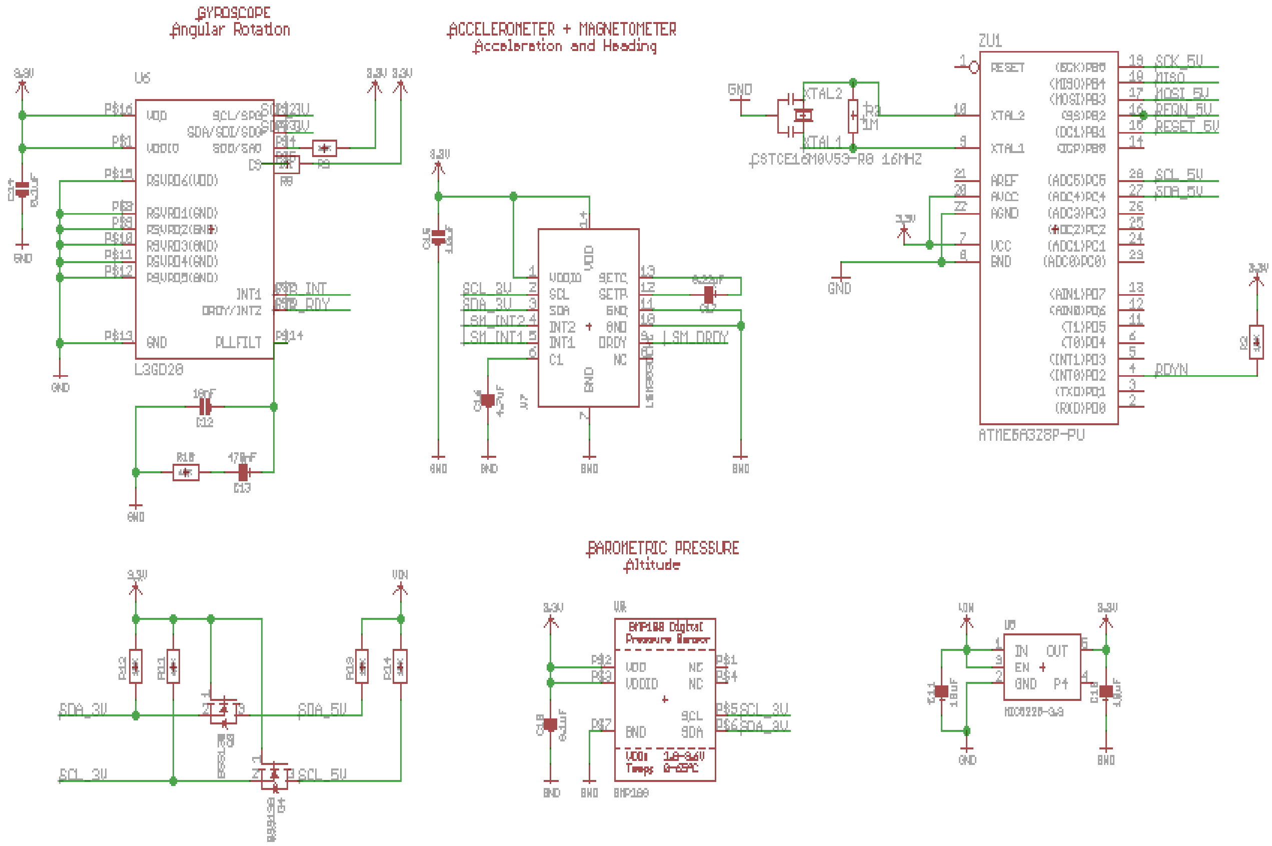
Figure 25 – The prototype



5.1.6c Schematics

The schematic below is a representation of the schematic of the three sensors we are using in our project, below you could see the schematic for the gyroscope on the left side followed by the accelerometer, then below in the middle is the barometric pressure sensor, the other devices seen with the sensors are very important in order to the the sensors working, these devices are the ATMega chip which is the device that will carry the program that controls the data received by the sensors and a voltage regulator which will regulate the voltage to 3.3V

Sensors Schematic



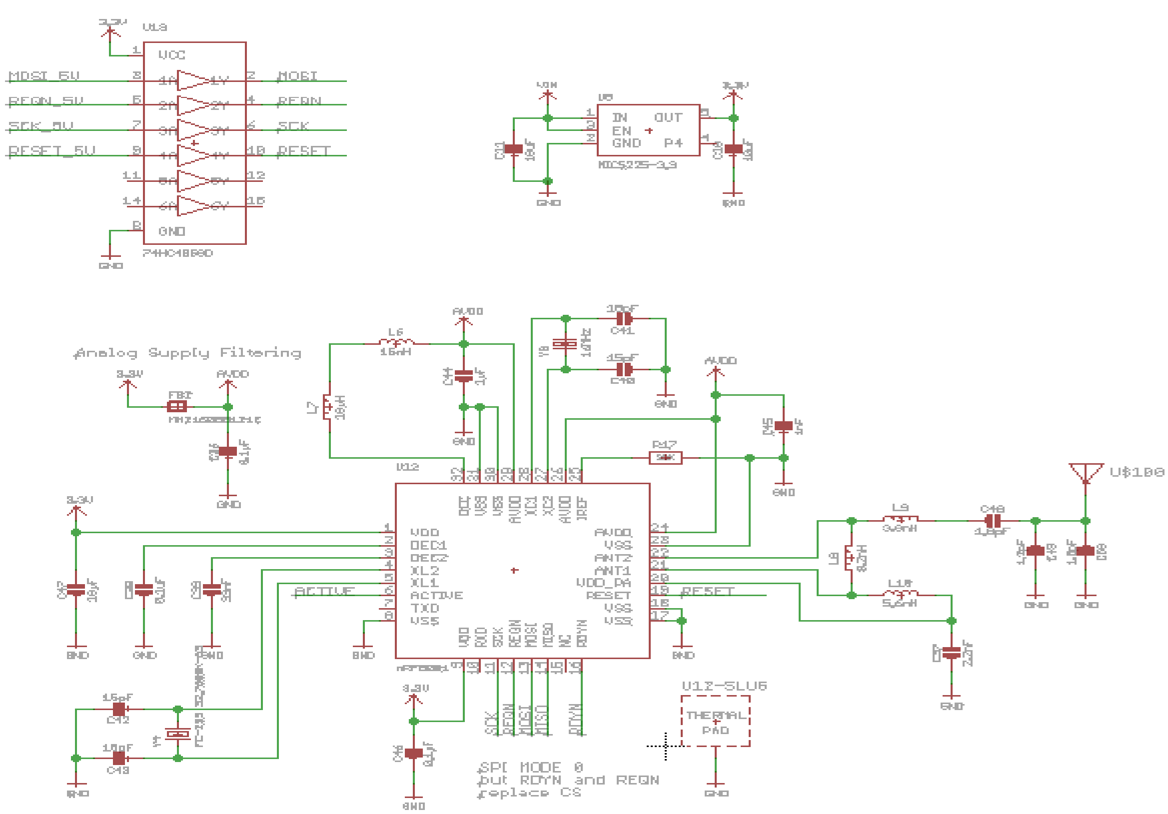
The picture below is that of the bluetooth schematic for our device. The schematic was created using EagleCAD, and it is based on previous bluetooth devices that have been proven to work exceptionally well, our team made sure the schematic is well organized and optimized as well as possible. That way we save space when we construct the entire printed circuit board.

Besides the main chip, you can see a voltage regulator, needed to regulate the voltage coming from the battery into the chip and the rest of the components that make up the bluetooth device.

The schematic below is designed and created in a way that it will be compatible with the rest of the devices, we made sure to make many of the values the same and kept a good consistency through the entire printed circuit board project, to avoid any confusion in the future, as well as to make it easier to the team to wire the board at the end of the design process.

The voltage regulator in the schematic above will make sure that all the components in the schematic will only be receiving up to 3.3V and not any more, to avoid damage to the board or overheating of the device.

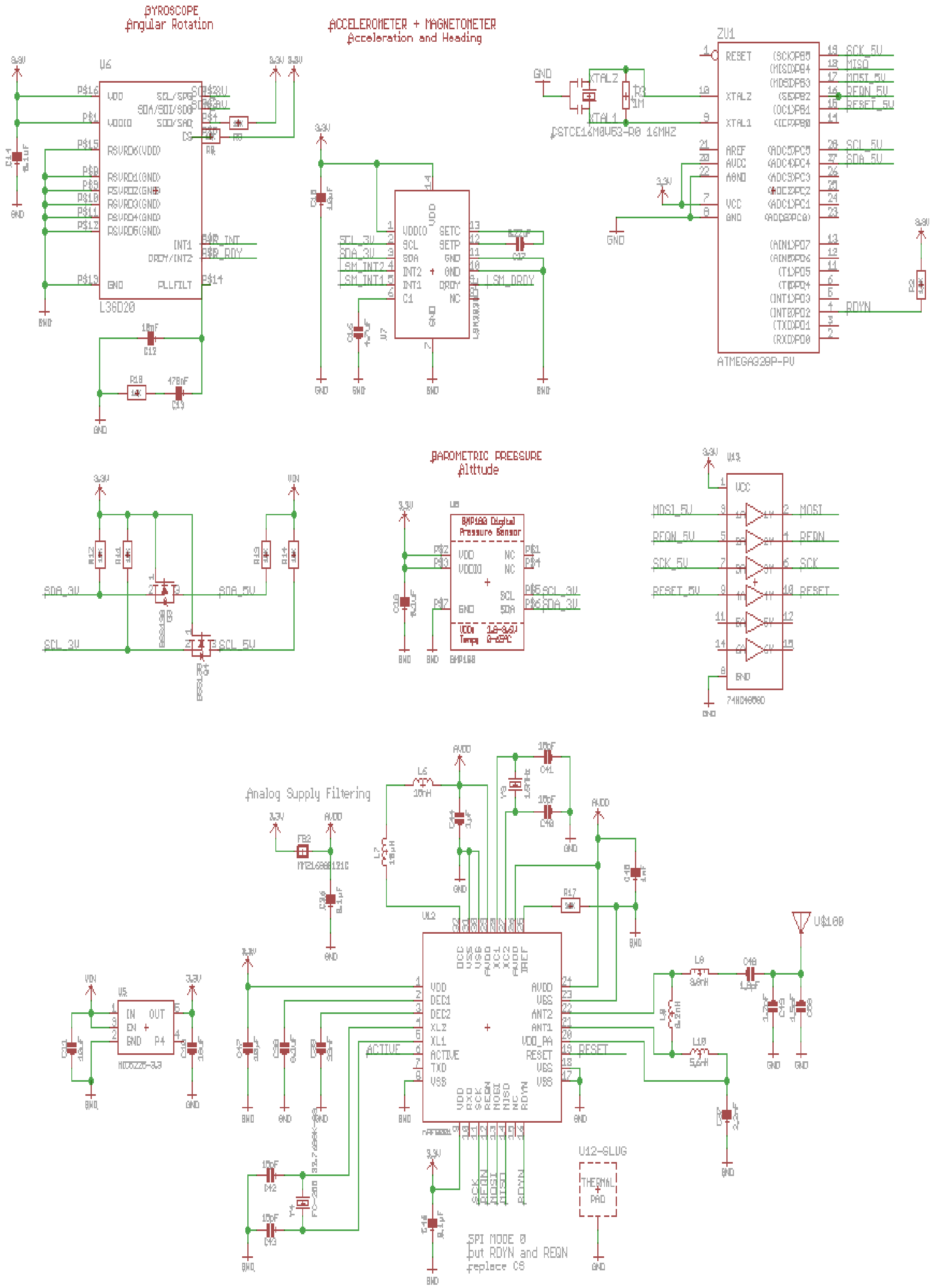
Bluetooth Schematic



The Final Schematic shown on the next page is the representation of our overall schematic, we have to make very few changes to the schematic below and we will have the printed circuit board ready to be manufactured. One of the features we will add is the schematic of a charging circuit, since our battery will be rechargeable we have to make sure our printed circuit board has the capability of charging it. Something else we are planning to add to the schematic, is an LED bulb to let the user know when the device is turned on or off. Besides that the only other step we have left is converting this schematic into a board and making the connections.

Making the connections will be the most challenging part of the board, because the team has to make sure none of the wires are overlapping and we also have to make sure the spacing between the wires is correct, otherwise the manufacturers will not make our boards since there is a certain standard of spacing to be met.

Final Schematic



5.1.7 Holster

This section is dedicated to the design plan of the holster of our project. There were three main holstering options for the SMART Skateboard discussed in the research section above.

In the following sections, we discuss the method selected for holstering the device as well as how we plan on implementing that method into the overall design of the SMART Skateboard.

This section is then concluded with a few rough renderings, or first drafts, of the holster we plan on instilling into the SMART Skateboard project.

5.1.7a Holster Method Selection

The decision here was simple. We chose that the best option for securing the electric components of our design was to develop a 3D printed holster using a sturdy material. We could also line the inside of this designed holster with a softer, foam-type material to ensure that the forces endured are completely dissipated before they reach the components.

The 3D design of the holster will closely mimic the one implemented by the team at Syrmo, but will most likely be larger due to the size of our battery pack and electric components inside the device. Using a 3D printed holster has many advantages over any other method.

The biggest of these advantages is the ability to customize the holster to the exact specifications we need. Any other method would restrict our design to a specific shape, weight, or location on the skateboard.

The tailor-made characteristics of using a 3D printed holster allows us to alter the design to our senior design group project’s exact specifications.

5.1.7b Holster Design Plan

The design of the holster will be very similar to the figure we see in the research section above made by Syrmo. The 3D printed holster will be designed in order to fit underneath the trucks of the skateboard.

The design should also allow for screws to be placed in the exact same locations as the screws would normally be to secure the trucks of the skateboard. This requires some elementary measurements of the distance between the screws on the skateboard as well as a knowledge of the acceptable thickness of the holster to disallow any impedance created from the additional elevation of the holster.

Once these measurements are made, we will use computer aided design software (SolidWorks or AutoCAD) to create a 3D design to be printed at one of the 3D printers found locally at the University of Central Florida.

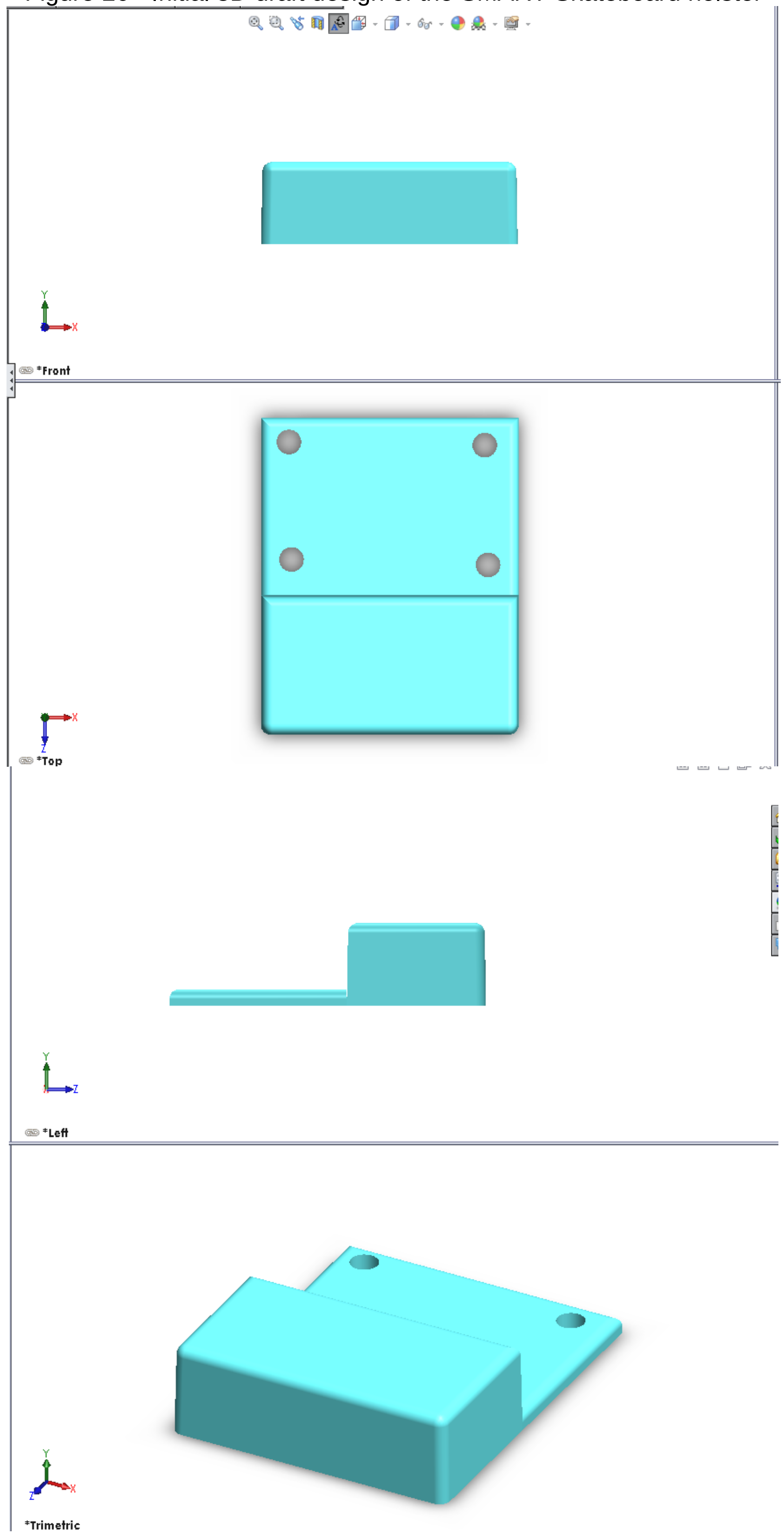
Figure 26 on the next page is the initial rendering of the 3D holster, designed by our team in SolidWorks studio. Please note that these images are far from the final design of the holster. These four images give a rough estimate of what we plan on creating to hold the electronic components underneath the skateboard.

The design of the final holster will need to have multiple parts that are hollowed out to allow the electronics to be implemented within. None of the specifics for the actual design of the holster have been finalized yet. The four figures below simply show how we plan to design the holster aesthetically as well as logistically.

The measurements, and specific implementations of the holster had not yet been taken into consideration whilst designing the holster shown in the figures below. The reason for the 3D designs below is to serve as a reference and foundation to creating the 3D printed holster we plan on implementing into the SMART Skateboard device.

Figure 26 on the next page shows the four different view of the holstering method which we selected to implement into the SMART Skateboard device. Once again, these are just a rough estimate of what we plan on developing to hold together our electronic device and are by no means the final version of the holster.

Figure 26 - Initial 3D draft design of the SMART Skateboard holster



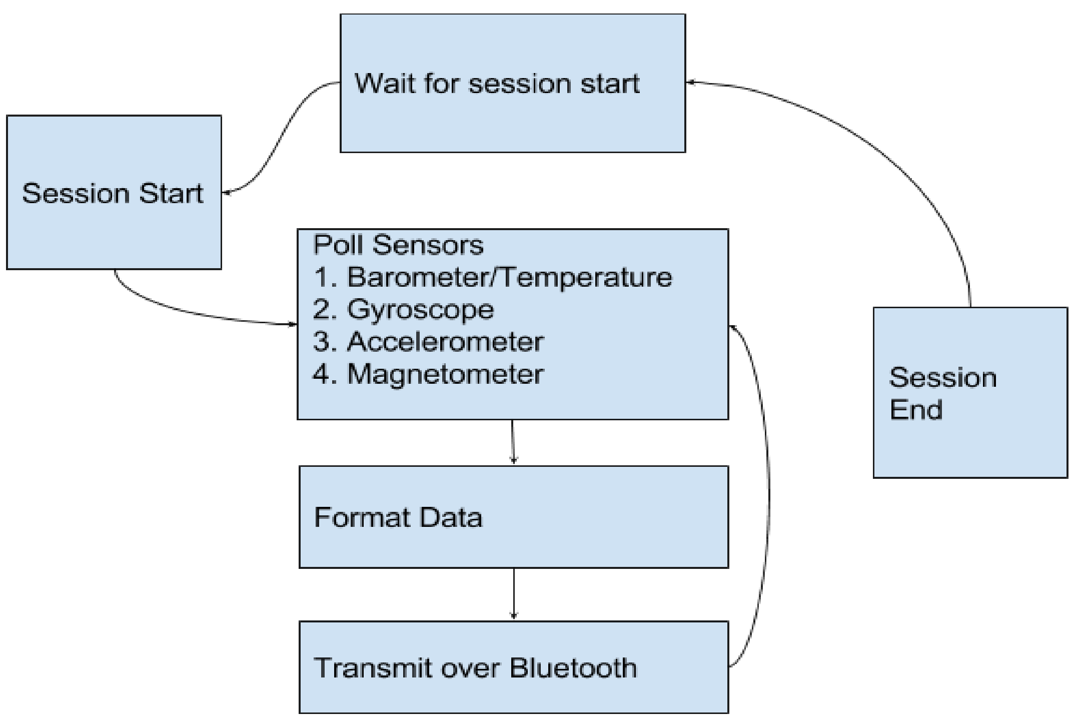
5.2 Software

This is the section with the design plan for all of the software specifics of the SMART Skateboard. These specifics include choosing the right type of embedded code, with the corresponding selection and block diagram of the embedded device, as well as the mobile device that will be configured to analyze the information being transmitted. Each subsection of this main section will discuss the selection of the corresponding software, followed by the implementation of the software into the project, and finally adding any additional figures, tables, or diagrams that could make the design process more achievable.

5.2.1 Embedded Code

The embedded code is the heart of the project and must control all of the hardware logic. The sensors should continuously be polled and that data should be sent over Bluetooth. When the device is not in use, the code should not poll the sensors to conserve energy. This will be achieved with a way to toggle on and off the session start and session end. When the device is not in use it will wait for a Bluetooth connection to pair with. After it establishes a connection it must now wait for the user to request for the session to start. After the session begins, the sensors will now be polled continuously in a loop. In every iteration of the loop, the data must be formatted into a compact form and sent over Bluetooth. When the session finally ends and the Bluetooth connection is terminated, the device can return to idle wait.

5.2.2 Embedded Block Diagram



5.2.2a Selection

To summarize, the group decided to pick the Arduino Uno as the microprocessor to control the project due to it's simplicity, the group's familiarity with the technology, and the wide array of resources available in the open source community such as break out boards and code repositories. The Arduino Uno showed to have many preferable characteristics when compared to its counterparts.

5.2.2b Implementation

The sensors were interfaced with SPI protocol and connected with the ATMega328P microcontroller. The Bluetooth chip gets paired with a mobile device during normal use. The microcontroller continuously polls the sensors to read in acceleration, rotation, pressure, temperature, and heading. Then the microcontroller formats this data in a compact way to send over Bluetooth. Finally, it sends the packet of information over Bluetooth to the paired device.

5.2.3 Mobile Application

A robust mobile application is crucial to the success of this project. Without a way to deliver results to the end user, our sensor data is useless. The user must receive this data in an intuitive and clear way. Our mobile application will be a user interface that receives data over Bluetooth and processes it in the back end and then displays a result on the front end.

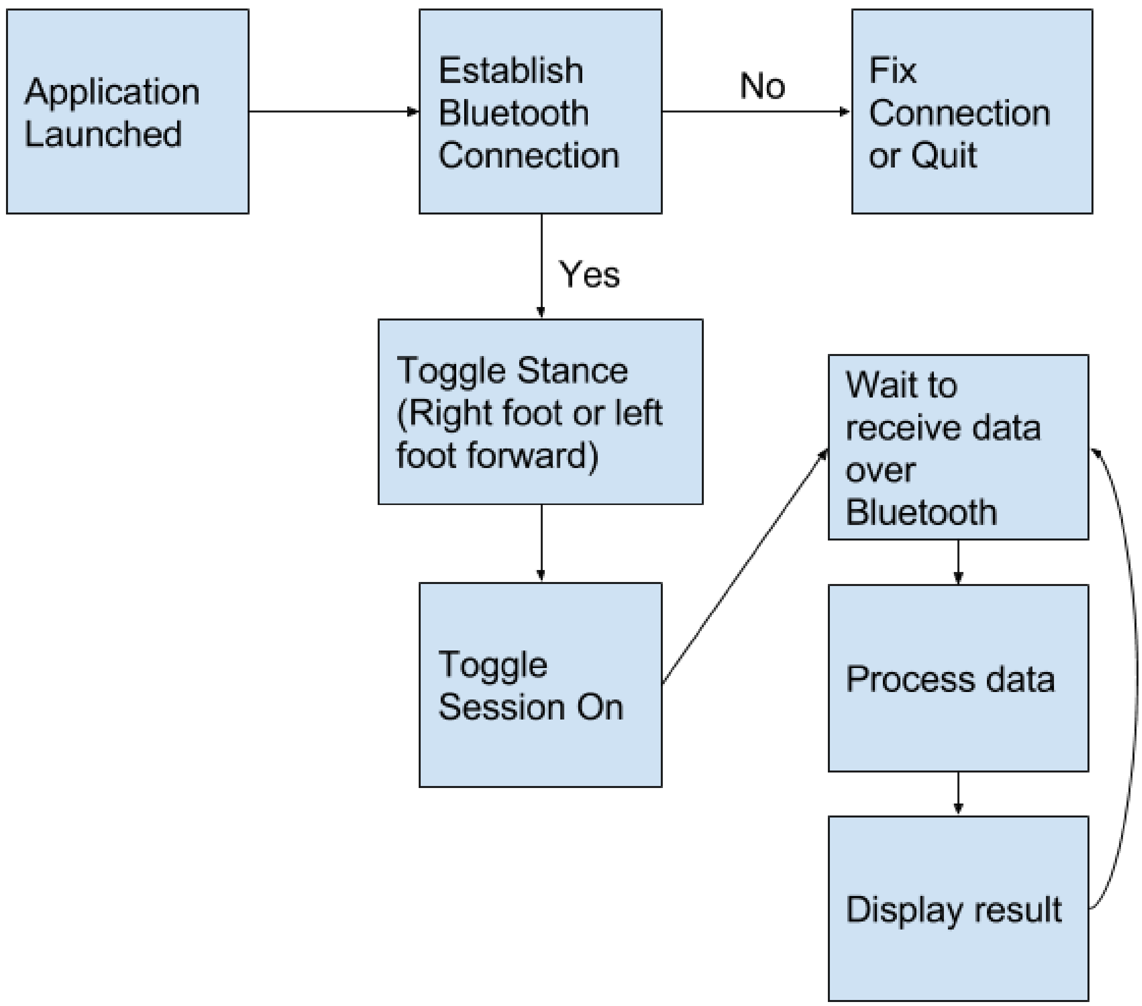
When the application is first launched it must establish a connection with our smart device over Bluetooth. If this fails, the connection must either be fixed or the application will exit. Then the rider will be able to set whether they will ride the board with their right or left foot forward because the tricks mirror each other when the stances are swapped. Then the rider will toggle the session ON which will send the smart device into a state of continuously polling the sensors.

The application then waits for a packet of data to come from the smart device. If this data is valid it will process it and determine which trick was performed. This result will be displayed to the user in a clear way on the user interface and also stored for the duration of the session. This process will repeat until the session is toggled off which will prompt the smart device to stop polling the sensors and the history of the session will be erased.

The mobile application must be developed with relevant standards and modern design practices in mind. Efficiency is very important. We want our application to use the phone's resources efficiently. Battery and data should be consumed in moderation to ensure a smooth user experience and user satisfaction. We also must make sure the application is highly responsive to user input. The user cannot be left waiting several seconds after they press a button. The application should immediately respond to user input and refresh the form quickly. The application should not contain any malicious software that could harm the user's device.

In the future we may wish to monetize the application. In that case we would have to register it in the application store of the device family that we end up using. That would cost a fee such as the developer's fee. The application would be tested to ensure that it is not malicious and then we could deploy it to the application store and begin making money.

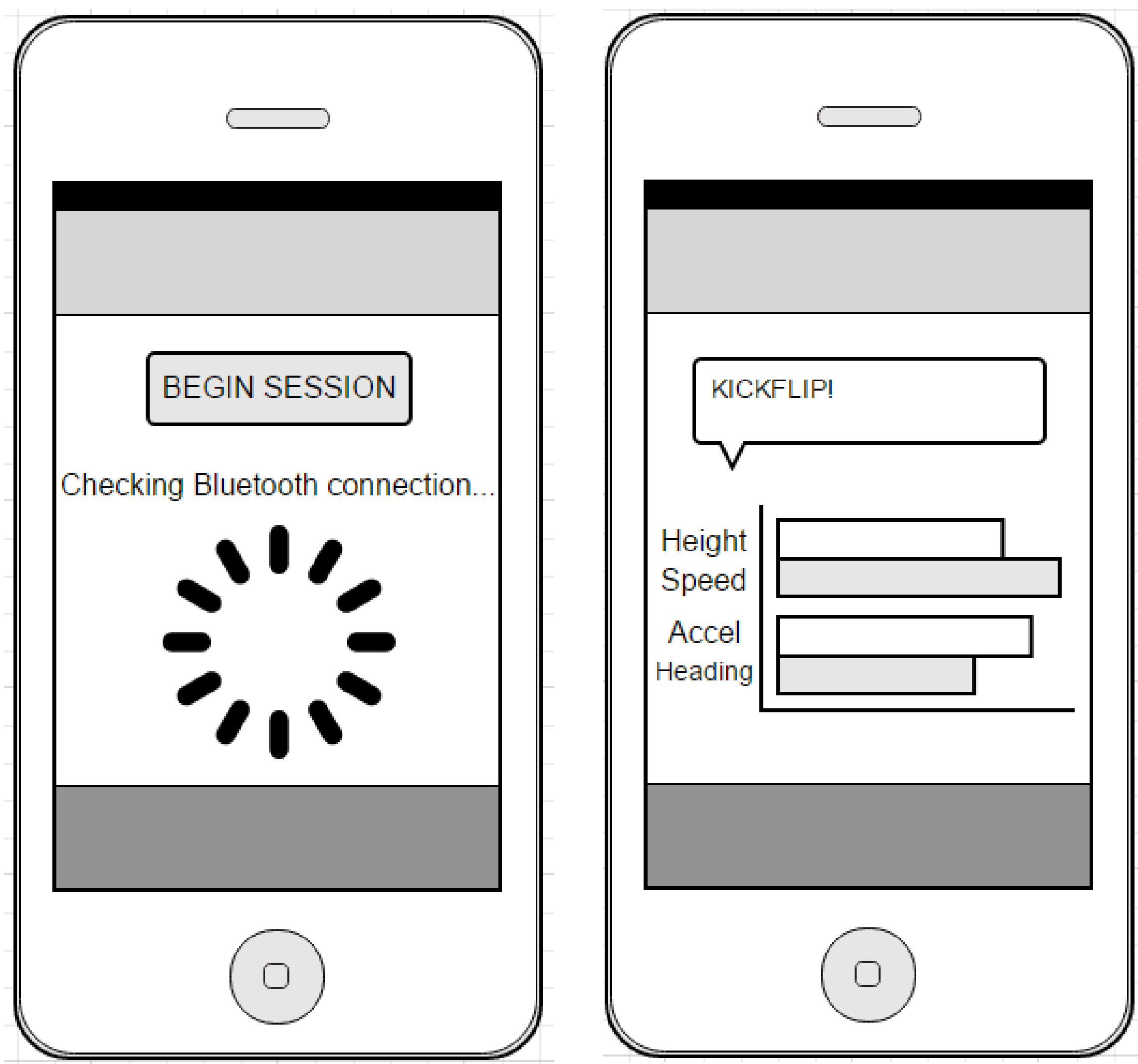
5.2.3a Mobile Application Block Diagram



5.2.3b Mobile Application Mockup

Figure 26 below is the mockup of the mobile application for clarity. This shows two of the most important menus of the mobile application. To the left we have the user pressing the begin session button and the application will try to establish a Bluetooth connection with the device. If successful, the application will move on to the application on the right. When the user starts performing tricks the application will display which trick it detected and stats such as height, speed, acceleration, and heading. These are just an idea of what it will look like at the end and the final layout is still in progress.

Figure 26 - Mobile Application Sample UI



5.2.3c Selection

The team has decided to develop an Android application because the team's developer has access to an Android more readily and has a slightly better knowledge of writing Android applications. We are excited to acquire more knowledge about the technology. Android is fully compatible with Bluetooth low energy so it is the perfect environment for us to start developing on.

5.2.3d Implementation

Interoperability plays a pivotal role in the development of our project. Again, a failed connection between the mobile application and the smart device will initially cause the application to exit the interface. The first step in establishing a connection between devices is to ensure that the BT module and mobile application are actually compatible. The module selected for our design will be the low energy version of Bluetooth (BLE). Although Bluetooth Low Energy is not backwards compatible, most phones on the market today already come BLE compatible. However, there could be other software compatibility issues if the Android API level used for the application is newer than the API used by the device. These API levels can be used as a tradeoff between having more features or more compatible devices.

Once this link is made certain, the module must now stabilize this connection in order to verify that the mobile application is ready to be used. The act of this is better known as pairing. Both the Bluetooth adapter and the mobile phone have to begin searching for each other. To do this, the mobile application will scan for a peripheral device and return any devices using the supported GATT services.

After a connection is maintained, you now need permission to exchange information within the source code to ensure security. Typically, Android mobile applications do not have automatic permission-to-use BT features. So the developer must declare the Bluetooth permission in the application manifest file. This permission is needed in order to use Bluetooth communication for actions like requesting a connection, accepting a connection, or transferring data.

5.2.4 Classification Algorithm

Now for our software to classify the trick correctly, it needs some type of algorithm to process the data in an intelligent way. The group had several ideas that each come with their own challenges.

Idea 1:

The microprocessor will use some sensor reading to determine whether a maneuver is currently being performed. For example, a rider simply cruising on the board should not have significant acceleration in the z-axis or any significant pitch, yaw, or roll readings. If we read x-axis acceleration, we know a maneuver has been started. Then the microprocessor will pay attention to the sensor changes. We will take note of whether the board spun 180 or 360 degrees and whether it flipped 360 degrees. Then classifying the trick is simply a matter of seeing which maneuver in our pre-programmed database fits those descriptions. For example, a kickflip is simply rotation of the board along the horizontal axis. If our horizontal rotation boolean is true, then we know the user could have done a kickflip. The rotation about the vertical axis boolean would have to be false; however, so that we really know that a kickflip was done and not some other trick.

Challenges:

Since the sensors are strictly on the skateboard, we have no way to determine whether the rider's body rotated also. Some riders perform complex rotations with the skateboard as well as their body. A simple example: a shuvit is when the board spins 180 degrees under the rider's feet. But a 180 is when the board and the rider spin 180 degrees. We believe each rider has noticeable differences between their shuvit and their 180 but these differences would not be picked up by simply seeing whether the board spun or flipped.

Idea 2:

Use a machine learning algorithm. Machine learning algorithms are very powerful. Recently one such deep learning algorithm was created by Google to defeat one of the world's highest ranked players in Go. Google accomplished this with huge hardware infrastructure to compute the game tree very quickly and to traverse it. Furthermore, it used state of the art algorithms to simply and prune the game tree so that it could traverse as deep as possible in the least amount of time.

Unlike Google, we will manually teach the algorithm which skateboarding maneuvers correspond to specific sensor data. When the device encounters similar sensor readings, the algorithm will turn to its past data and output the maneuver that best fits the data. This might be overkill to solve the problem though because most skateboarding maneuvers are very clear cut. It either rotated or spun in the specific direction or it didn't.

Challenges:

Every skateboarder has their own style. The same maneuver may have a dramatically different height or small differences in motion. The algorithm may have trouble accurately distinguishing these differences.

Idea 3:

We can store all of the tricks we wish to classify on a web server. Most modern applications such as Instagram use web server hosting such as Amazon's services. This is because it is very cost effective. The mobile application will compare the data that it's receiving from our device and try to compare to what is on the web server. If a match is found it will display a success for the user, otherwise it will display a fault.

Challenges:

Web hosting costs money so it will be an additional expense to our project. Hosting a web server to store all of our trick profiles can become quite expensive and will ultimately cost way too much for group to be able to find proper funding. Furthermore, none of our group members are doing an Information Technology or Computer Science degree so our experience is very limited with web hosting and server side technologies. This means that we would need to spend valuable time learning the technology, when that time is much better spent on actually designing and building the SMART Skateboard.

6.0 Prototype

This entire section is dedicated to the creative process of the first prototype for the SMART Skateboard. The first half of this section discusses the different stages of creating the prototype. In other words, once the parts are all ordered, prepared, and tested, we outline the implementation of the prototype. The second half of this section discusses the different facilities and equipment we used to create the prototype as well as properly test each feature and electric component of the SMART Skateboard.

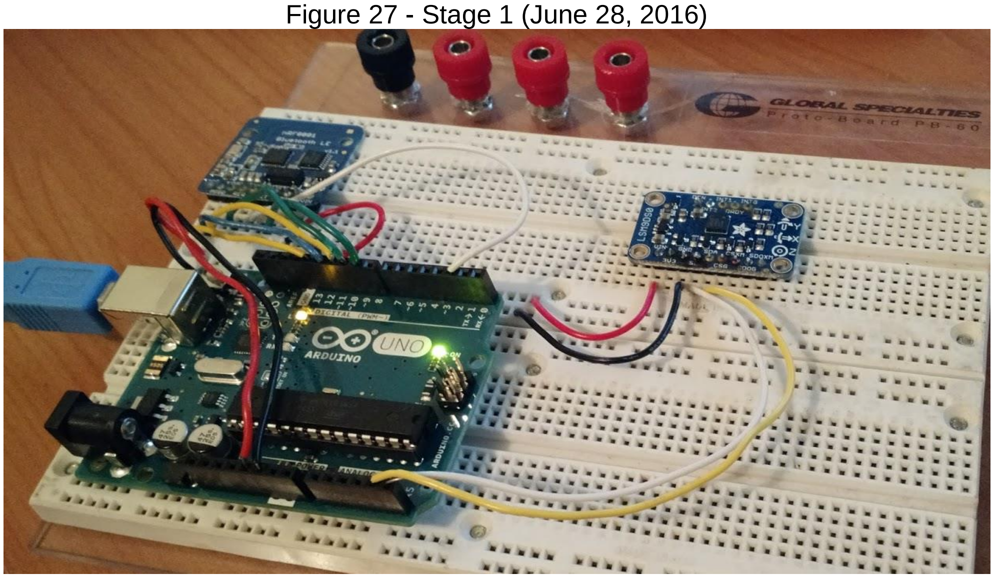
6.1 Prototype Iterations

This section will break down the different phases of the prototype development of our project. After doing all of the necessary research, design planning, and ordering of the proper components, the next step was to create a functional prototype of the SMART Skateboard. The following three steps outline the major milestones that we achieved when creating the first prototype of our device.

Step 1

The first step shown in Figure 27 below is after acquiring the right embedded device and the corresponding sensors and communication chips, was to put it all together. We powered our Arduino Uno with a USB connection to a nearby computer. This allowed us to see if the device was functioning properly. Using wire found in any electrical engineering lab, we configured the Bluetooth communications chip along with the three sensor Adafruit “10 degrees of freedom” (gyroscope, accelerometer, barometer) to the Arduino Uno. After some testing to see full functionality, we took this picture.

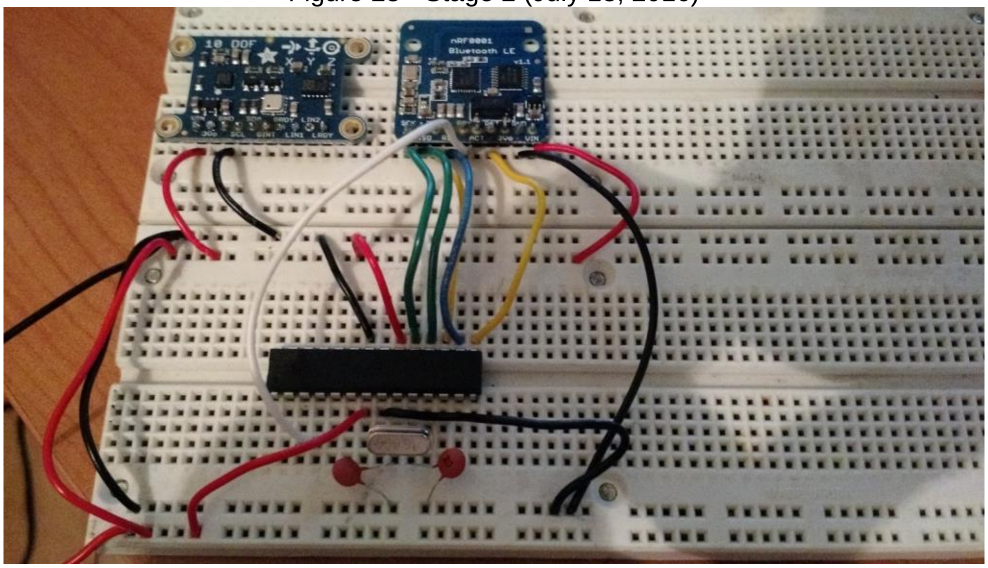
Figure 27 - Stage 1 (June 28, 2016)



Step 2

The second phase of the prototype development, shown in Figure 28 below, was to find a way to minimize the design of the SMART Skateboard device. This entailed removing the main chip from the Arduino Uno, which took some basic research, and figuring out a way to connect the chip to the external components of the SMART Skateboard. After some more research, we found out that the connections can be made possible simply by adding a pair of capacitors and a crystal oscillator to regulate the power being drawn to the external components.

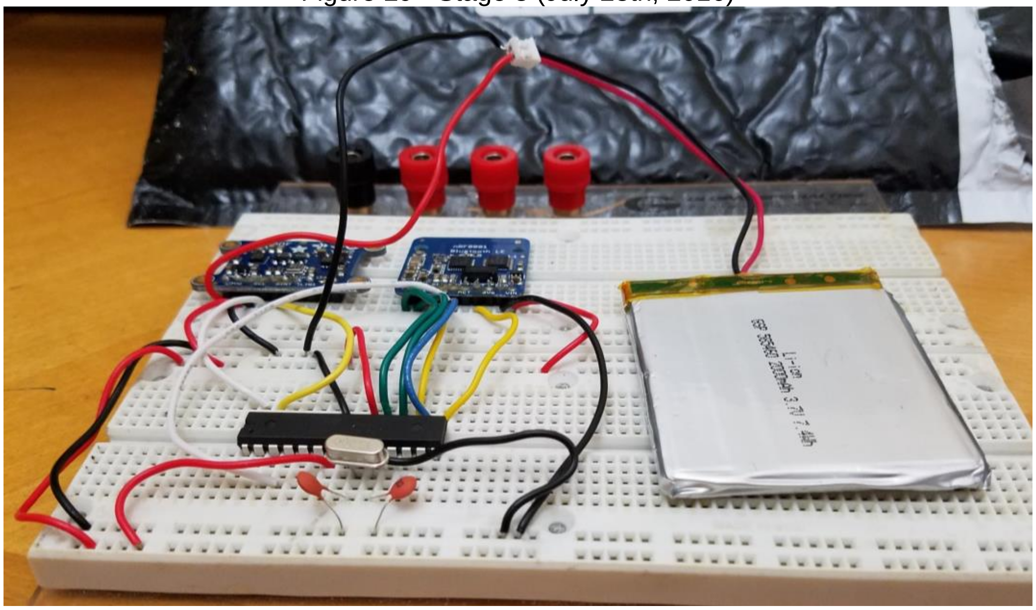
Figure 28 - Stage 2 (July 13, 2016)



Step 3

The third and final stage of creating the first prototype, shown in Figure 29 on the next page, is to implement the power source into the device to ensure that the entire prototype can be mobilized. Once we decided to use the LiPo flat pack battery to power the device, we connected the two poles of the battery to the voltage regulated pins already build in the main chip of the prototype. After many tests confirming the orientation capabilities of the device worked properly, we took a photo of the final rendering of our first prototype. Below, is the image of our final, fully mobilized prototype.

Figure 29 - Stage 3 (July 28th, 2016)



6.2 Facilities and Equipment

The University of Central Florida provided us with many state of the art facilities and modern equipment to develop our prototype and work on our design. The following sections detail the facilities available to us and what equipment we had access to.

Senior Design Lab

In the senior design lab, we had access to multimeters, oscilloscopes, power supplies, breadboards, computers, and other basic electrical supplies necessary to complete our project.

TI Smart Lab

The Texas Instruments Smart Lab provided us with quality soldering irons, computers, and an all-around creative space to work in. The group would frequently meet up there several times a week to solder new components or test new hardware.

Harris Engineering Computer Lab

The Harris Engineering computer lab was crucial to our success in writing our document. It equipped the group with brand new Windows 10 computers which allowed us to be productive and quickly collaborate and write our draft documentation.

Multimeter

The multimeters found in the TI Smart Lab as well as the Senior Design Lab allowed our group to properly test the prototype of the SMART Skateboard right after we received the parts from the suppliers and constructed the design. The multimeter allowed us to make sure that the data sheets of the device we created are all accurate and we received no faulty equipment from our suppliers.

DC Power Supply

The power supply allowed the group to power up our prototype before we acquired our battery. We were able to supply the same amount of input voltage that the battery would supply, and by doing that, we were also able to read the total current drawn from the device. The power supply generator was a key part in testing the prototype before we could select and acquire the proper battery.

Soldering Iron

The soldering iron allowed the group to quickly assemble the breakout boards and begin testing them on our breadboard. Having group members that had previous experience with soldering made the creation of the prototype a quick and easy process. This allowed the team to focus more on the design rather than learning how to use the equipment provided to us by the University of Central Florida.

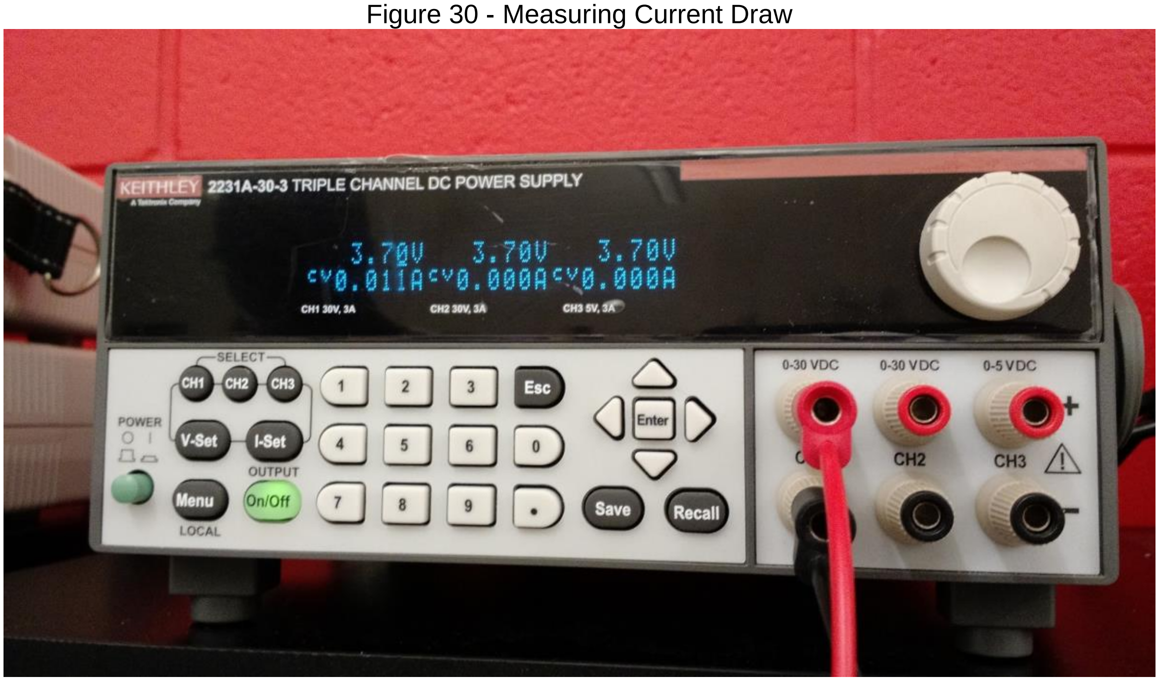
7.0 Testing

This section is dedicated to showcasing the testing protocols of every aspect of the SMART Skateboard device. This main section is broken down into two main parts. One part is dedicated to the hardware testing of the prototype and its components. The other part is dedicated to the software testing of the project. In the software section we test the embedded communications and the information being passed to the main chip from the external chips. Both sections include multiple images captured by our team to show the testing results of the testing that we implemented throughout this project’s creation.

7.1 Hardware Testing

This is a brief section that outlines the voltage and current specifications of our prototype once we created the first version of the SMART Skateboard. Shown below in Figure 30 is the current draw of our prototype when connected to the power supply in the TI Smart Lab at the University of Central Florida. We were measuring 0.011 Amps when supplying 3.7 Volts. This is the same specification that is described in the datasheets of the main chip of our prototype. Below we show an image we took to prove the input voltage and output current of the prototype SMART Skateboard device.

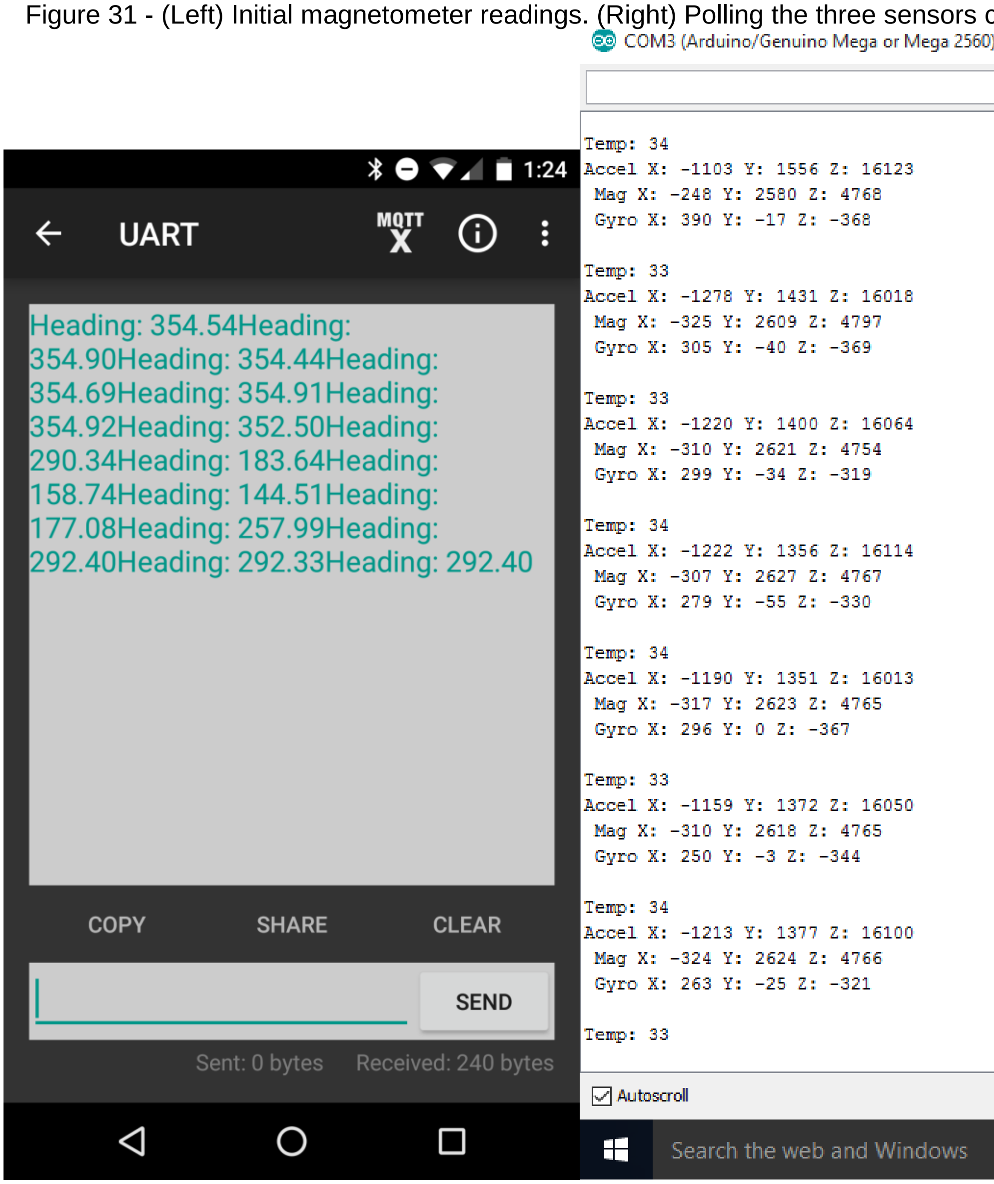
Figure 30 - Measuring Current Draw



7.2 Software Testing

This section briefly shows the results of the prototype's software being tested after its initial creation. These tests involve hooking up our prototype and launching the mobile application provided by Adafruit. Then, the device is constantly running and feeding information to the mobile device. This constant feed of data is what we are testing in this section. Figure 31 below shows our early stages of testing software. To the left we are using the sample Android application provided by Adafruit to receive the magnetometer reading information from our device and displaying it onto the screen. The figure on the right shows how we are using the Arduino serial monitor to poll all of the sensors and display their combined output on the screen.

Figure 31 - (Left) Initial magnetometer readings. (Right) Polling the three sensors continuously



8.0 Administration

Administration is a key part to the success of every team. The division of labor in a fair and agreed upon method is the most important part as far as making a project run smoothly and efficiently. The second part to this division of labor is the creation of deadlines and the meeting of these deadlines. The following section is focused on the administration of our project. The first subsection breaks down the division of responsibilities of our group members as we accomplish different milestones (with the deadlines included) throughout the semester, which is then followed by another subsection showing the estimated budget of accomplishing the full creation of the SMART Skateboard.

8.1 Milestones

#	Task	Start	End	Duration (weeks)	Responsible
	Senior Design I				
1	**Introduction and Project Selection**	05/20/2016	05/27/2016	1	Everyone
2	**Initial Document (10 pages)**	05/27/2016	06/03/2016	1	Everyone
3	**Table Of Contents Due**	05/27/2016	07/01/2016	4	Everyone
4	**Current draft due (Document)**	05/27/2016	07/08/2016	5	Everyone
5	**Final Document Due**	05/27/2016	08/02/2016		Everyone
6	**Meeting With Instructor**	06/07/2016	06/07/2016		Everyone
	Research parts & software				
7	Accelerometer	06/03/2016	06/17/2016	2	Taymas
8	Gyroscope	06/03/2016	06/17/2016	2	Nick

9	Barometer/Altimeter	06/03/2016	06/17/2016	2	John
10	Bluetooth/Wi-Fi	06/03/2016	06/17/2016	2	Brandon
11	Power/ Battery	06/03/2016	06/17/2016	2	Brandon
12	Software/code	06/03/2016	06/17/2016	2	Taymas
13	PCB parts	06/03/2016	06/17/2016	2	John
14	Microcontroller	06/03/2016	06/17/2016	2	Everyone
15	Case	06/03/2016	06/17/2016	2	Nick
	Design PCB & Software				
16	Accelerometer	06/17/2016	07/22/2016	5	Taymas
17	Gyroscope	06/17/2016	07/22/2016	5	Nick
18	Barometer/Altimeter	06/17/2016	07/22/2016	5	John
19	Bluetooth/Wi-Fi	06/17/2016	07/22/2016	5	Brandon
20	Power/ Battery	06/17/2016	07/22/2016	5	Brandon
21	Software/code	06/17/2016	07/22/2016	5	Taymas
22	PCB parts	06/17/2016	07/22/2016	5	John
23	Microcontroller	06/17/2016	07/22/2016	5	Everyone

24	Case	06/17/2016	07/22/2016	5	Nick
Senior Design II					
25	**Order parts**	07/22/2016	08/22/2016	4	Everyone
26	**Build prototype**	08/22/2016	10/13/2016	7	Everyone
27	**Test Prototype**	10/13/2016	10/27/2016	8	Everyone
28	**Enhance prototype**	10/27/2016	11/10/2016	2	Everyone
29	**Peer Presentation**	TBA	TBA		Everyone
30	**Final Report**	TBA	TBA		Everyone
31	**Final Presentation**	TBA	TBA		Everyone

8.2 Budget

ANTICIPATED BUDGET	
Component	**Price**
Processor/Microcontroller	$25.00
Accelerometer	$10.00
Gyroscope	$9.95
Barometer	$8.50
Battery	$12.95
Wireless Communication	$22.00
PCB Manufacturing	$60.00
Harness	$25.00
Miscellaneous	$35.00
Total Cost	*$208.40*

8.3 Bill of Materials

The Bill of Materials for our printed circuit board is shown in Table 18 below. The BOM file is necessary to order our printed circuit board. The different manufacturers ask for the bill of materials to build the board, but we also need it for ourselves if we would like to purchase the devices on our own, we also need to be able to account for every single component in the board and having a bill of materials is the best way of achieving such a task.

Table 18 – Bill of Materials

Qty	Value	Device	Package	Part
1		ANTENNA_PCBNRF8001_1.6MM	ANT_PCB_2.4GHZ_NRF8001_1.6MM_1OZ	U$10
1	0.1uF	CAP_CERAMIC0805-NOOUTLINE	0805-NO	C38
2	0.1uF	CAP_CERAMIC_0805MP	_0805MP	C14,
2	0.1µF	CAP_CERAMIC0805-NOOUTLINE	0805-NO	C36,
1	0.22µF	CAP_CERAMIC_0805MP	_0805MP	C17
1	1.2pF	CAP_CERAMIC_0402MP	_0402MP	C49
1	1.5pF	CAP_CERAMIC_0402MP	_0402MP	C50
1	1.8pF	CAP_CERAMIC_0402MP	_0402MP	C48
1	10K	RESISTOR0805_NOOUTLINE	0805-NO	R2
7	10K	RESISTOR_0805MP	_0805MP	R8,
1	10nF	CAP_CERAMIC_0805MP	_0805MP	C12
3	10uF	CAP_CERAMIC_0805MP	_0805MP	C10,
1	10µF	CAP_CERAMIC0805-NOOUTLINE	0805-NO	C47
1	10µH	INDUCTOR_0603	_0603	L7
1	15nH	INDUCTOR_0603	_0603	L6
4	15pF	CAP_CERAMIC0805-NOOUTLINE	0805-NO	C40,
1	16MHz	CRYSTAL2.5X2.0	CRYSTAL_2.5X2	Y3
1	1M	R-EU_R0603	R0603-ROUND	R3
1	1nF	CAP_CERAMIC_0603	_0603	C45
1	1µF	CAP_CERAMIC0805-NOOUTLINE	0805-NO	C44
1	2.2nF	CAP_CERAMIC_0402	_0402	C37
1	22K	RESISTOR_0805MP	_0805MP	R17
1	3.9nH	INDUCTOR_0402	_0402	L9
1	33nF	CAP_CERAMIC0805-NOOUTLINE	0805-NO	C39
1	4.7uF	CAP_CERAMIC_0805MP	_0805MP	C16
1	470nF	CAP_CERAMIC_0805MP	_0805MP	C13
1	5.6nH	INDUCTOR_0402	_0402	L10
1	74HC4050D	74HC4050DTSSOP	TSSOP16	U13
1	8.2nH	INDUCTOR_0402	_0402	L8
1	ATMEGA328P-PU	ATMEGA328P-PU	DIL28-3	ZU1
1	BMP180	BMP180EXT	BMP180_EXTENDED	U8
2	BSS138	MOSFET-NWIDE	SOT23-WIDE	Q3,
1	CSTCE16M0V53-R0 16MHZ	RESONATORMU	RESONATOR	Y2
1	FC-255 32.7680K-A3	CRYSTAL4.9X1.8	CRYSTAL_4.9X1.8	Y4
1	L3GD20	GYRO_L3G2400D	L3G4200D_LGA16L	U6
1	LSM303DLHC	LSM303DLHC	LGA14	U7
1	MIC5225-3.3	VREG_SOT23-5	SOT23-5	U5
1	MMZ1608B121C	FERRITE0805	805	FB2
1	nRF8001	NRF8001	QFN32_5MM	U12

8.4 Division of Labor

This following section breaks down the team's division of labor pertaining to developing the SMART Skateboard senior design project. This is a brief description of everyone's role in the senior design project.

Taymas is in charge of the software parts of the design. His duties included creating the most optimal software design. He is in charge of coding the embedded system as well as coding the mobile device.

As a quick summary, John was responsible for the communication chip of the device. This means that he was in charge of selecting and implementing the most efficient wireless communication chip possible within our budget, space and time restrictions.

Brandon had the responsibility of finding the right sensors for the SMART Skateboard. This means that he was in charge of detailing each sensor component, finding the most optimal option for the SMART Skateboard, then implementing it into the design of the project.

Matias has the task of determining the optimal battery type and holster design of the device. He was in charge of finding the optimal voltage necessary to power up the SMART Skateboard, then configuring the correct type of power source that could power up this device while also being mobilized. He was also in charge of developing the 3D holster design that will ultimately hold the SMART Skateboard's electronic components.

The entire team was responsible for creating the PCB microcontroller. This means that we were all present in the research, design, and analysis of the prototype SMART Skateboard's PCB. Although some of us have more experience in developing PCBs, we all wanted to be present to learn and be a part of the main foundation behind the SMART Skateboard.

8.5 Personnel

Taymaskhan Musaev - Computer Engineering

Taymas Musaev is a Computer Engineering student at the University of Central Florida. He has been part of the Lockheed Martin College Work Experience program since March 2015, designing hardware and software for automation systems. He hopes to in the future pursue a Master's degree in Computer Science at the University of Central Florida and pursue a career in Software Engineering.

Matias Canter - Electrical Engineering

Matias Canter is also an Electrical Engineering student at the University of Central Florida. As an immigrant from Budapest, Hungary, Matias came to America with the hopes of furthering his education as well as pursuing an athletic endeavor. Due to a recent series of injuries, his athletic hopes had to take the back seat. This in turn forced Matias to become fully involved in the electrical engineering pursuit. Inspired by his family's academic accomplishments, where his mom and grandfather were both physicists, Matias declared to be an electrical engineer due to the sheer difficulty of the program at UCF. Matias excels in many technical fields such as circuit theory, electrical analysis, and coding, as well as non-technical skills such as writing, team management, organization, and networking. Matias hopes to graduate in December ready to enter the electrical engineering industry.

Jonathan Espinal - Electrical Engineering

Jonathan Espinal is currently a first generation student at the University of Central Florida. Jonathan has been involved on campus activities since he started his freshman year, he is a part of SHPE the society of Hispanic Professional Engineers at UCF and NSBE, the National Society of Black Engineers on campus. Jonathan is also a Brother of Lambda Theta Phi Latin Fraternity Inc where he held numerous roles, including Administrative office, Recruiting and Retention Advisor, as well as the Public Relations Representative for the chapter. In the Spring 2015 Jonathan joined Earthrise Space Foundation: Google Lunar Xprize team omega envoy as an intern where he worked improving the electrical and communication system of Sagan the Lunar rover who is scheduled to launch to the moon in December 2017, the internship mentioned provided Jonathan with enough hands on experience which helped him get a second internship with the option of a full-time position upon graduation in the NSBE 2016 national conference in which he is currently in. He is now an intern for United Technologies, Carrier corporation, he is now getting experience in the controls engineering field, learning how to program VAVs and how to control simple sensors on the jobsites. All the knowledge gained in the previous mentioned internships are helping Jonathan contribute to the Project in Group 6.

Brandon Carty - Electrical Engineering

Brandon Carty is a first generation student currently wrapping up his electrical engineering degree at the University of Central Florida. Originally from South Florida, Brandon branched out and committed himself to extending his network in the Orlando region. Since freshman year, he has joined a number of organizations to diversify his experience and along the way he also picked up a business minor broaden his knowledge within the workforce. At the peak of his sophomore year, Brandon started working at the UCF Fairwinds Alumni Center where he then lined up his first two internships. From then he managed to gain the hands on skills to which he uses in contribution to the project.

9.0 Conclusion and Future Work

In the conclusion and future work section we will be summarizing all we have done throughout our project as well as what we have planned for future improvement. Since the time for senior design is limited, we have left out some of the hardest features for a later time, where we could fully develop the app and the device as time permits and as we learn more about the project ourselves.

9.1 Conclusion

<u>Senior Design I:</u>

In conclusion, the first half of the senior design project has been mainly about research and preparation. We had to obtain the proper knowledge of the necessary technologies for an acceptable idea, create a testable prototype, and then implement all of this information into a final documentation report. The first goal we had to accomplish was to think of a suitable design idea that would be acceptable by the UCF electrical engineering faculty. The SMART Skateboard encompasses all of the requirements that were laid out in front of us at the beginning of the year. It requires a custom built PCB, as well as many implemented electronic sensors and components, not to mention a mobile application that is to be coded entirely by our software development team.

Once a proper idea was agreed upon, the next step of the semester was to conduct research relevant to the project on which we decided. This step took up most of the time, in terms of the first semester of the development of the SMART Skateboard. This is very evident by simply looking at the sheer volume of relevant information we discovered in the research section. Many of us came into this project with little knowledge of the specific microcontrollers, processors, sensors, integrated chips, wireless communication technologies, power supply options, battery options, and holstering methods all needed for a successful design of the SMART Skateboard. The research for this project was paramount.

The third phase of the first senior design semester dealt with creating an acceptable prototype, after the proper testing of the hardware and software components needed, for full implementation of the SMART Skateboard device. At the beginning of this project, we assumed that this phase of the process would have taken the longest time.

However, after we gained the necessary knowledge about all of the different intricacies of this device, the actual design of a prototype fell into place much simpler than we originally expected. This simplicity is in part due to the fact that we have obtained new knowledge about each part of the design, and also in part due to the way electronic components are designed to be integrable with one another as well as all of the schematics and information being open to anyone looking to use the products.

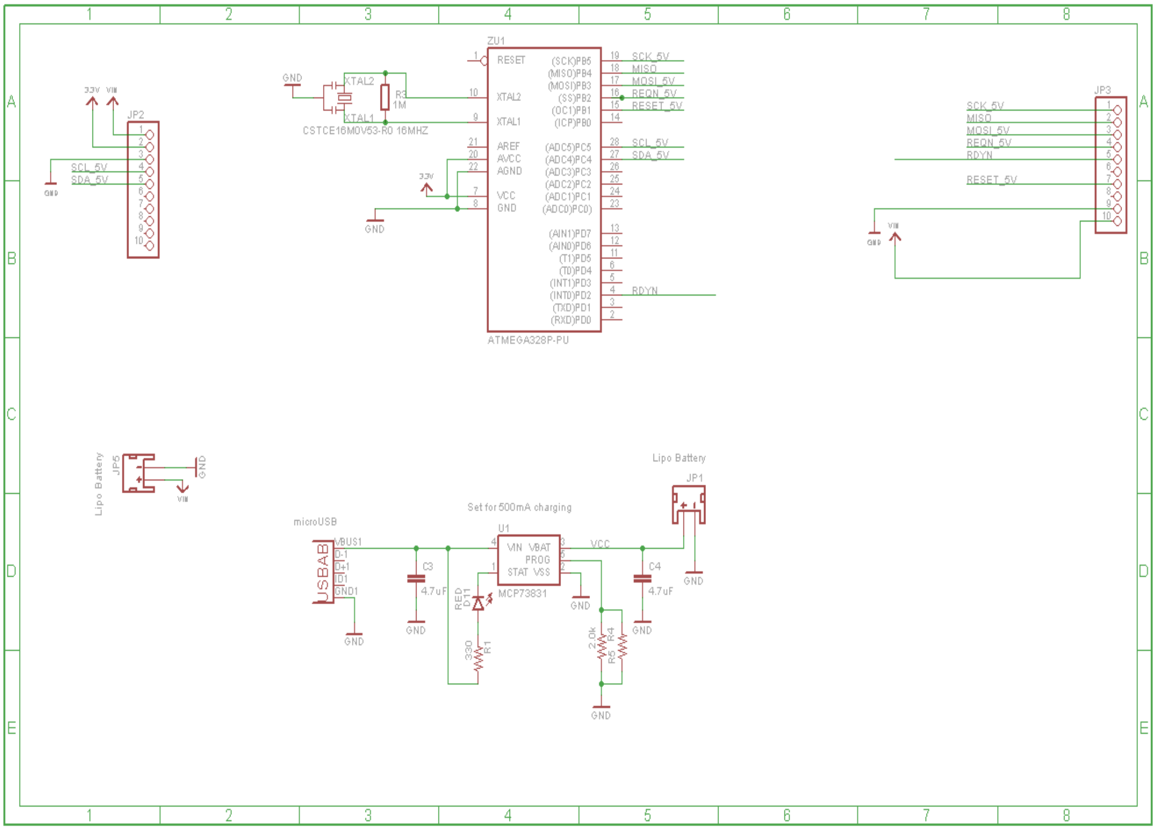
The final objective of Senior Design I was to create a document outlining the project idea, the distribution of responsibilities, the relevant research, the design methodology, and any other subsequent data that is paramount towards the development of the SMART Skateboard. This document is meant to be frequently referred to for guidance during Senior Design II, where we will actually develop the final version of the SMART Skateboard.

Senior Design II:

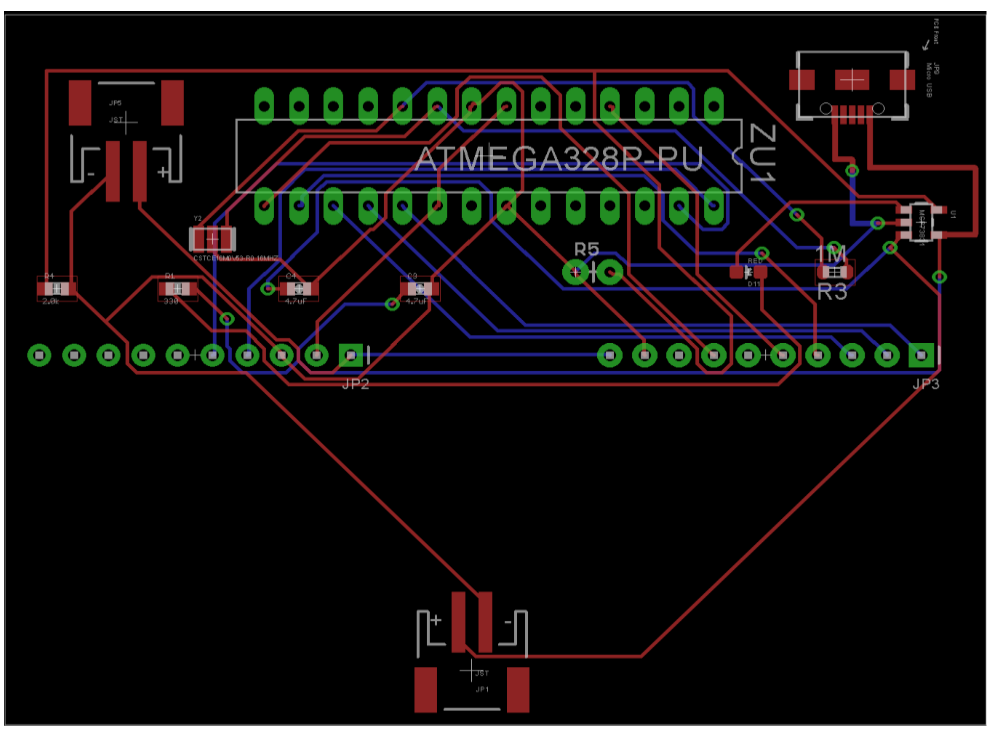
As we wrap up the year, this project has been a very reflective process, and one of the most challenging two semesters of our academic careers. We not only learned how to transform an idea into a physical, functioning product, but we also accomplished to work as a true professional team. The SMART Skateboard is a simple concept with a number of critical details. We took on the challenge of this project and followed through to completion. Altogether, making this an eye-opening experience to get the SMART Skateboard off the ground.

The following covers all of the implemented updates to our SMART Skateboard design, to which we ensured a successful working project.

Updated PCB schematic/design:

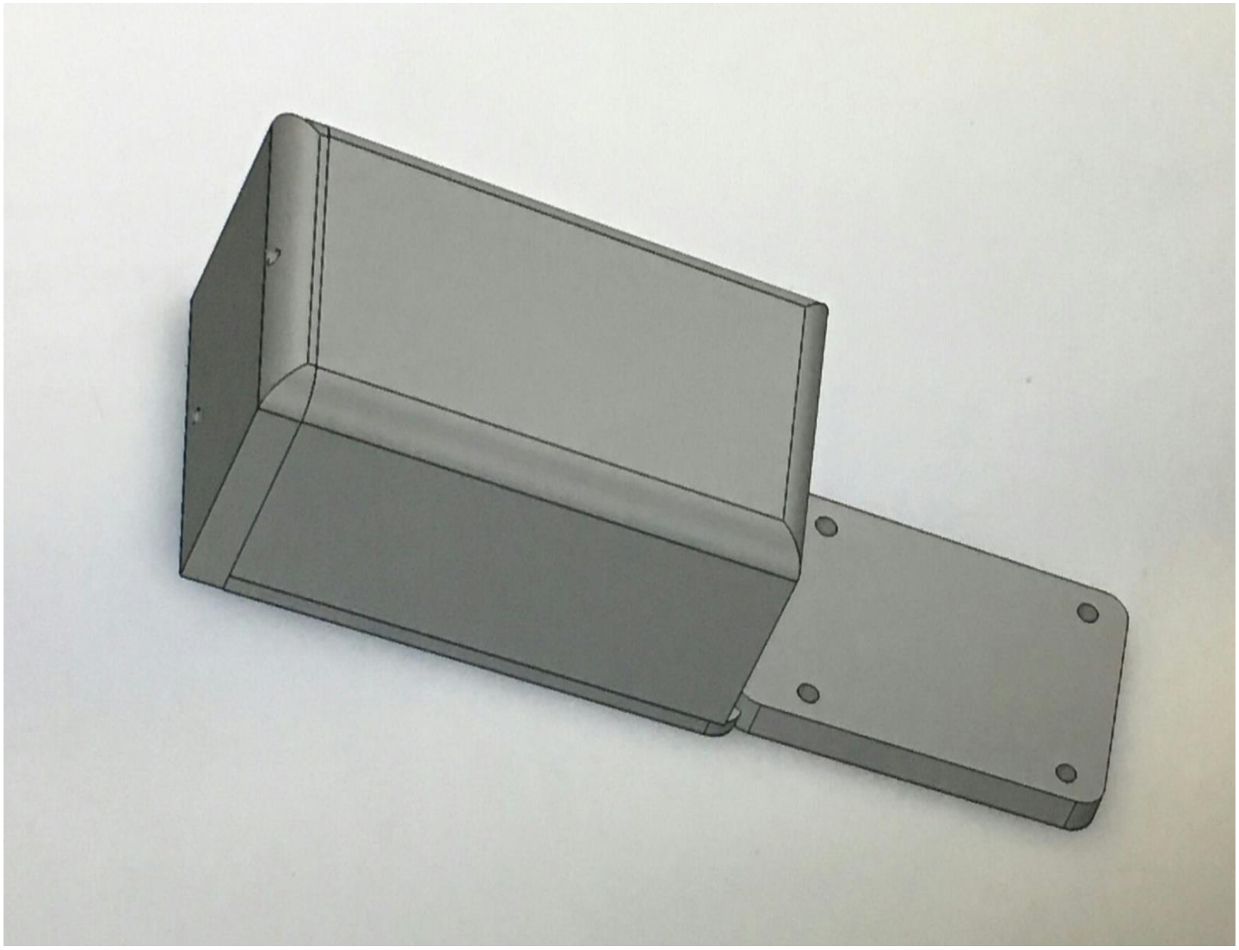


Updated PCB Schematic



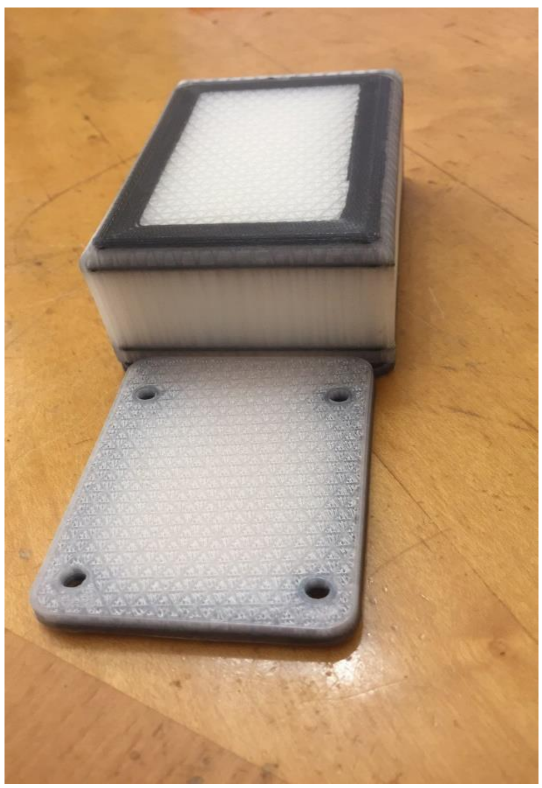
Updated PCB Design

Final holster design/prototype:

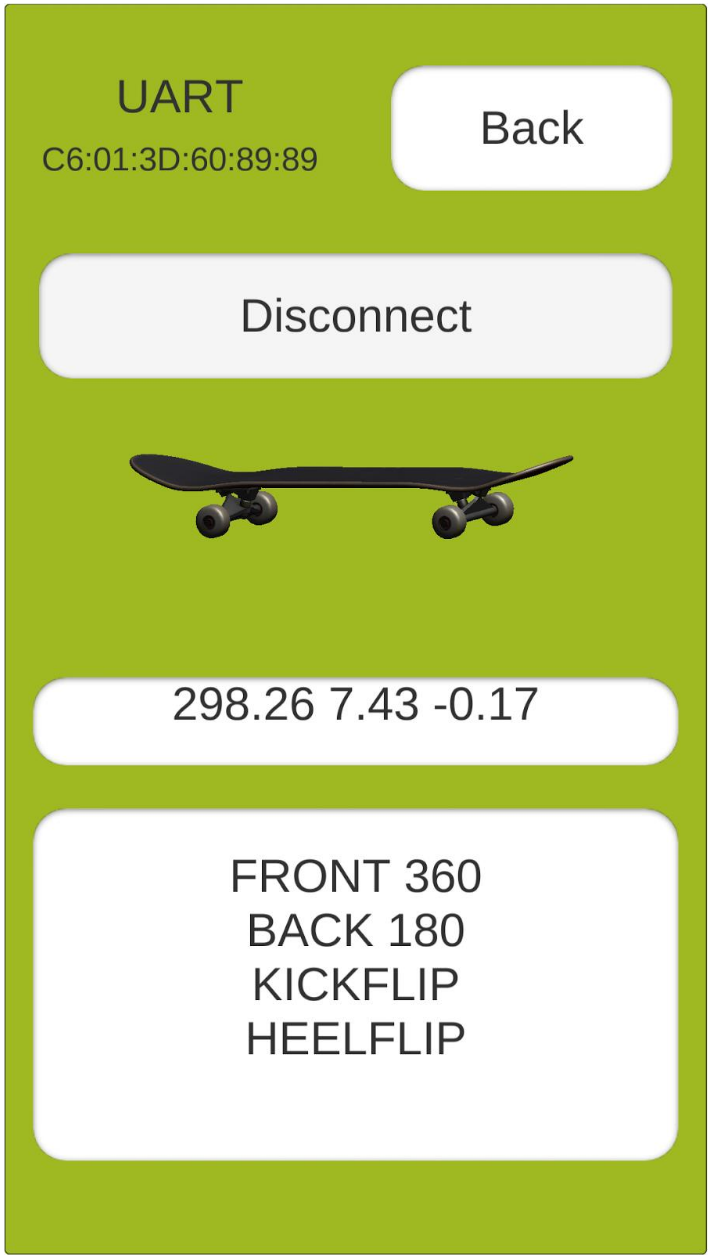


CAD Design of Holster


3D Designed Prototype of Holster (A)


3D Designed Prototype of Holster (B)

Mobile application interface:



Updated Mobile Application

Updated budget:

UPDATED BUDGET	
Arduino Uno	$25.95
nRF8001 Bluetooth LE	$19.95
Gyroscope, Accelerometer & barometer	$29.95
Resistors/Capacitors	$5.00
PCB Manufacturing + Shipping	$41.60
Holster Manufacturing + Shipping	$82.52
Total	*$204.97*

9.2 Future Work

Multi-Platform compatibility:

In the future there are some upgrades the team has thought about adding to the project. Some of the main functions we have thought about is compatibility with different platforms in various operating systems. At the moment the project would only support android, however, in the future we are planning on making it compatible with IOS and Windows. Adding such features would attract more users to our product, as it would facilitate the usability of it. Another interesting feature we have discussed before is being able to use the sensors as a game, where the users would gain points for landing different tricks. Once the user finishes his/her sessions, they would be able to see how many points they have accumulated and they can try to beat their own score, or compete with friends or other users.

Challenge Game:

Adding the challenge game to our app would be one of the greatest improvements, it would be fun for our the team as well as for the users. After implementing the point system game we also discussed we would like to add a social media platform to our app specialized for skaters who use our product. In the social media part of the app the users would be able to schedule meetups and talk about how to improve their techniques in order to land better tricks, we believe creating a community for a product is definitely a good idea, because it is not just about using an item but now it becomes about sharing knowledge between people who are alike.

Individual Challenge:

The challenges do not even have to be necessarily multiplayer. Since the SMART skateboard has a collection of tricks that it knows, it can prompt the user with a random trick that they need to perform but they will be limited to a certain amount of tries. If the user lands the trick within the limit of tries, then they will accumulate points. If they do not land then they will lose that round. This will allow the user to constantly improve themselves by using the app to generate random tricks for them to attempt.

GPS Integration:

Also, adding GPS to our device would be one of the major improvements. By adding GPS, we would give our users the ability to share their location with they people they choose to. Having GPS would also allow us to measure the total distance traveled in a certain amount of time by the user, and by using this information we could also estimate the amount of calories burned by using the distance and the type of tricks performed.

At such level, our device would not just be a device that users purchase just to have fun, but to measure their overall health and improvements over time. At this point this device is only designed to be implemented on a skateboard, but in the future it could very well be modified to the point where the user chooses what the device is being attached to, whether it is a bicycle, a skateboard, roller skates or whatever else it might be, there is definitely room for improvement.

Other Sports:

The previous section brings up an interesting point that once we successfully dominate the skateboard wearable sensor market, the logical thing to do would be to expand into other sports. The same needs that drove us to bring data analysis and trick classification to skateboarding is present in other sports. In football, our device could be implemented to track helmet impact. In track or soccer, players can use a device such as ours to track their speed during an event or game. In snowboarding it is also very similar and the user could use a device such as ours for spins and flips. Also in biking or roller blading speed is also very important. In gymnastics it would be helpful to see how fast you are spinning.

Appendix A: Sources

"Accelerometer." *Wikipedia*. Wikimedia Foundation, n.d. Web. 01 Aug. 2016. <https://en.wikipedia.org/wiki/Accelerometer>

"Android N Developer Preview." *Android Developers*. N.p., n.d. Web. 01 Aug. 2016. <https://developer.android.com/index.html>.

"Arduino - ArduinoBoardMega2560." *Arduino - ArduinoBoardMega2560*. N.p., n.d. Web. 01 Aug. 2016. <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>.

"Arduino - ArduinoBoardUno." *Arduino - ArduinoBoardUno*. N.p., n.d. Web. 01 Aug. 2016. <https://www.arduino.cc/en/Main/ArduinoBoardUno>.

Bader, D. L., and M. J. Pearcy. "Prosthetics and Orthotics International." *Material Properties of Velcro Fastenings*(n.d.): n. pag. Web. 20 July 2016. <http://www.oandplibrary.org/poi/pdf/1982_02_093.pdf>.

"Barometric Pressure Sensor - BMP180." *- COM-11823*. N.p., n.d. Web. 01 Aug. 2016. <https://www.sparkfun.com/products/11823>

"Barometric Pressure Sensor - MPL115A1." *- SEN-09602*. N.p., n.d. Web. 31 July 2016. <https://www.sparkfun.com/products/9602>

"Battery Technologies." *Sparkfun*. N.p., n.d. Web. 15 July 2016. <https%3A%2F%2Flearn.sparkfun.com%2Ftutorials%2Fbattery-technologies>.

"CC2541 (ACTIVE)." CC2541. Texas Instruments, Apr. 2014. Web. 01 July 2016. <http://www.ti.com/product/CC2541>.

"Featured Platforms & Tools." *Microsoft Developers – Any Developer. Any App. Any Platform.* N.p., n.d. Web. 01 Aug. 2016. <https://developer.microsoft.com/en-us/>.

"Gyro Breakout Board - MLX90609 - 300°/s." *- SEN-08372*. N.p., n.d. Web. 31 July 2016. <https://www.sparkfun.com/products/retired/8372>

"Gyroscope." *Wikipedia*. Wikimedia Foundation, 13 July 2016. Web. 31 July 2016. <https://en.wikipedia.org/wiki/Gyroscope>

"How Do I Power My Arduino?" *Cases for Your Raspberry Pi*. N.p., 18 Sept. 2015. Web. 31 July 2016. <http://www.modmypi.com/blog/how-do-i-power-my-arduino>.

"Institute of Electrical and Electronics Engineers." IEEE Std 802.11-2007, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 12 June 2007.

"Institute of Electrical and Electronics Engineers." IEEE Std 802.15.1-2005, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs). Web. 14 June 2005. <http://standards.ieee.org/getieee802/download/802.15.1-2005.pdf>

"J.S.T. Mfg. Co., Ltd." *J.S.T. Mfg. Co., Ltd.* N.p., n.d. Web. 31 July 2016. <http://www.jst-mfg.com/product/detail_e.php?series=199>.

"LiPo Battery." *Honcell*. N.p., n.d. Web. 12 July 2016. <http://www.honcell.com/product/lipo-battery/25-1-000mah/>.

"NRF8001." / Bluetooth Low Energy / Products / Home. Nordic Semiconductor, 12 Mar. 2015. Web. 5 July 2016. <https://www.nordicsemi.com/eng/Products/Bluetooth-low-energy/nRF8001>.

"MAX21000+ Maxim Integrated | Mouser." *Mouser Electronics*. N.p., n.d. Web. 31 July 2016. <http://www.mouser.com/ProductDetail/Maxim-Integrated/MAX21000+/?qs=%2fha2pyFaduhyACdIJ1C9lHrwFaI7kOnVOUXnqq2VyaU%3d>

Menniti, Michele. "Feeding Power to Arduino: The Ultimate Guide." *Open Electronics*. N.p., 9 July 2015. Web. <http://www.open-electronics.org/the-power-of-arduino-this-unknown/>.

"MSP430™ Ultra-low-power Microcontrollers." *16-bit 32-bit MCU*. N.p., n.d. Web. 01 Aug. 2016. <https://www.ti.com/lsds/ti/microcontrollers_16-bit_32-bit/msp/overview.page>.

"RN4020." - Bluetooth Module. Microchip, 18 Feb. 2015. Web. 5 July 2016. <http://www.microchip.com/wwwproducts/en/RN4020>.

Sicilian, Tony. "Bluetooth Data Transfer with Android." Java Code Geeks. N.p., 10 Sept. 2013. Web. 20 July 2016. <https://www.javacodegeeks.com/2013/09/bluetooth-data-transfer-with-android.html>.

"SparkFun Triple Axis Accelerometer Breakout - ADXL362." - *SEN-11446*. N.p., n.d. Web. 01 Aug. 2016.

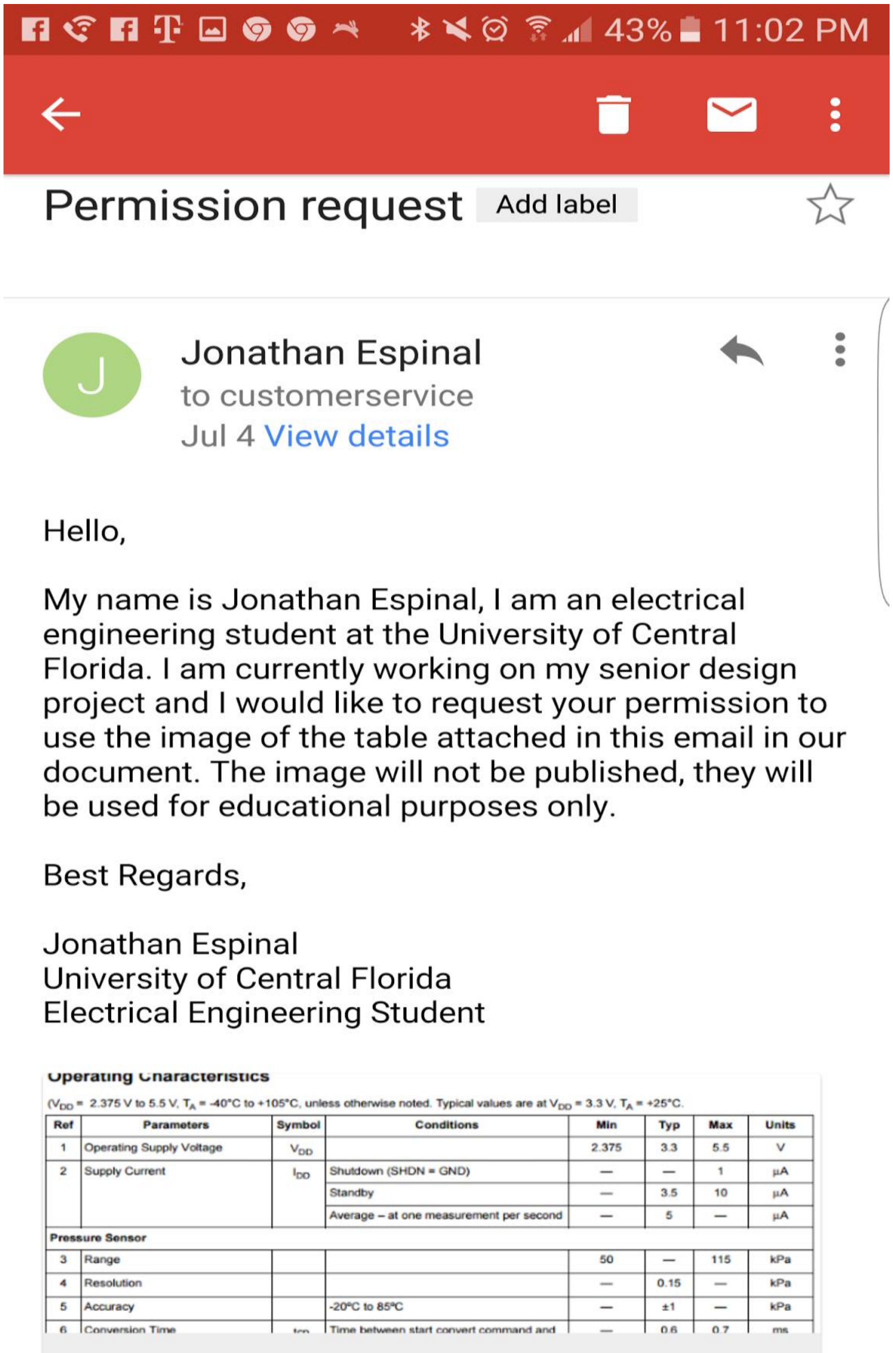
<https://www.sparkfun.com/products/11446>

"Texas Instruments CC2541 2.4-GHz Bluetooth® SoC." TI CC2541 2.4-GHz Bluetooth® SoC. Mouser Electronics, Apr.-May 2014. Web. 01 Aug. 2016. <http://www.mouser.com/new/Texas-Instruments/ti-cc2541/>.

"Triple-Axis Digital-Output Gyroscope - ITG-3200." - *SEN-09793*. N.p., n.d. Web. 31 July 2016.
<https://www.sparkfun.com/products/9793>

Appendix B: Permissions

Permission to use images from SparkFun:



43% 11:02 PM

Permission request Add label

Jonathan Espinal
to customerservice
Jul 4 View details

Hello,

My name is Jonathan Espinal, I am an electrical engineering student at the University of Central Florida. I am currently working on my senior design project and I would like to request your permission to use the image of the table attached in this email in our document. The image will not be published, they will be used for educational purposes only.

Best Regards,

Jonathan Espinal
University of Central Florida
Electrical Engineering Student

Operating Characteristics

(V_{DD} = 2.375 V to 5.5 V, T_A = -40°C to +105°C, unless otherwise noted. Typical values are at V_{DD} = 3.3 V, T_A = +25°C.

Ref	Parameters	Symbol	Conditions	Min	Typ	Max	Units
1	Operating Supply Voltage	V_{DD}		2.375	3.3	5.5	V
2	Supply Current	I_{DD}	Shutdown (SHDN = GND)	—	—	1	µA
			Standby	—	3.5	10	µA
			Average – at one measurement per second	—	5	—	µA
Pressure Sensor							
3	Range			50	—	115	kPa
4	Resolution			—	0.15	—	kPa
5	Accuracy		-20°C to 85°C	—	±1	—	kPa
6	Conversion Time		Time between start convert command and	—	0.6	0.7	ms

43% 11:03 PM

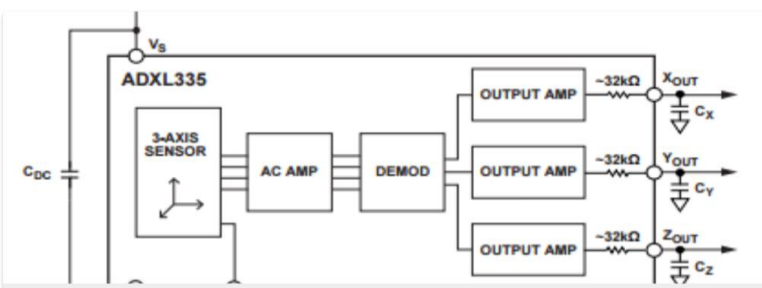
Jonathan Espinal
to customerservice
Jul 4 View details

Hello,

My name is Jonathan Espinal, I am an electrical engineering student at the University of Central Florida. I am currently working on my senior design project and I would like to request your permission to use the images attached in this email in our document. The images will not be published, they will be used for educational purposes only.

Best Regards,

Jonathan Espinal
University of Central Florida
Electrical Engineering Student



adxl335 Acc...meter.PNG

43% 11:03 PM

SparkFun Customer Service
to me
Jul 5 View details

Type your response ABOVE THIS LINE to reply

Jonathan Espinal
Subject: Image use permission request

JUL 05, 2016 | 12:43PM MDT
Kris H replied:

Hi Jonathan,

Thank you for checking in! Please feel free to use our product photos in your project documentation or reports. Please let me know if I can be of further assistance, and have a lovely day!

Kind Regards,

Kris Hamilton
Customer Support & Volume Sales Lead
303-945-2984

Request for permission to use STMicroelectronics images:

43% 11:02 PM

Jonathan Espinal
to tom.gable
Yesterday View details

Hello,

My name is Jonathan Espinal, I am an electrical engineering student at the University of Central Florida. I am currently working on my senior design project and I would like to request your permission to use the image of the gyroscope table attached in this email in our document. The image will not be published, they will be used for educational purposes only.

Best Regards,

Jonathan Espinal
University of Central Florida
Electrical Engineering Student

OffDr	Zero-rate level change vs temperature(3)	FS = 2000 dps		±0.04		dps/°C
NL	Non linearity(3)	Best fit straight line		0.2		% FS
Rn	Rate noise density(3)	BW = 50 Hz		0.011		dps/($\sqrt{Hz}$)
ODR	Digital output data rate(3)			11.9/23.7/ 47.3/94.7/ 189.4/ 378.8/ 757.6		Hz
Top	Operating temperature range		-40		+85	°C

Gyroscope selection.PNG

43% 11:02 PM

Image usage permission request

Add label

Jonathan Espinal
to tom.gable
Yesterday View details

Hello,

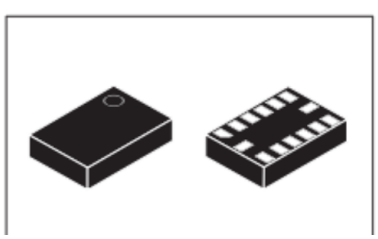
My name is Jonathan Espinal, I am an electrical engineering student at the University of Central Florida. I am currently working on my senior design project and I would like to request your permission to use the image of the table attached in this email in our document. The image will not be published, they will be used for educational purposes only.

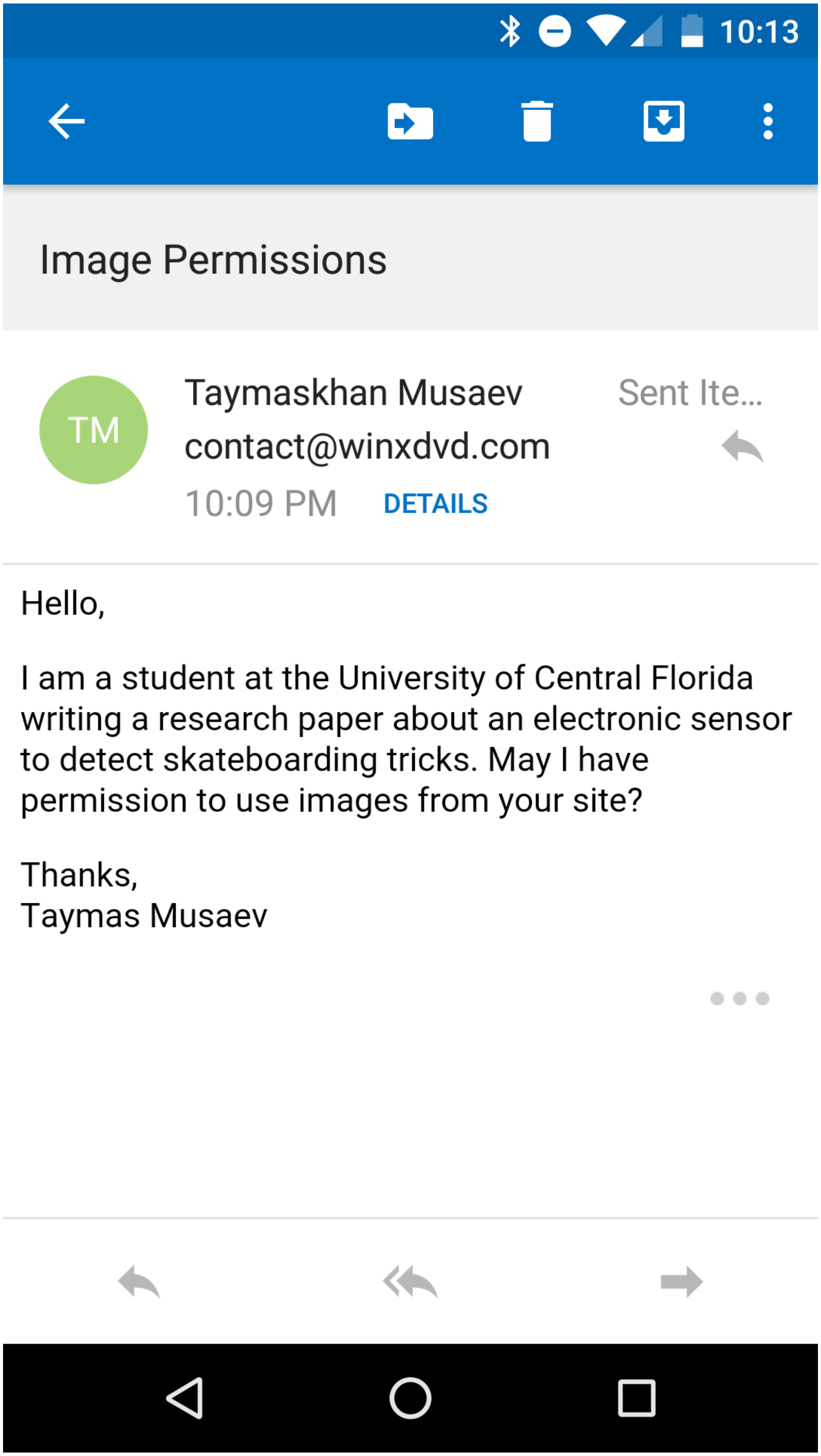
Best Regards,

Jonathan Espinal
University of Central Florida
Electrical Engineering Student

Ref.	Min.	Typ.	Max.
A1			1
A2		0.785	
A3	0.16	0.2	0.24
D1	2.85	3	3.15
E1	4.85	5	5.15
N1		0.8	
L1		4	
T1		0.8	
T2		0.5	

mechanical data





Hello,

I am a student at the University of Central Florida writing a research paper about an electronic sensor to detect skateboarding tricks. May I have permission to use images from your site?

Thanks,
Taymas Musaev

10:13

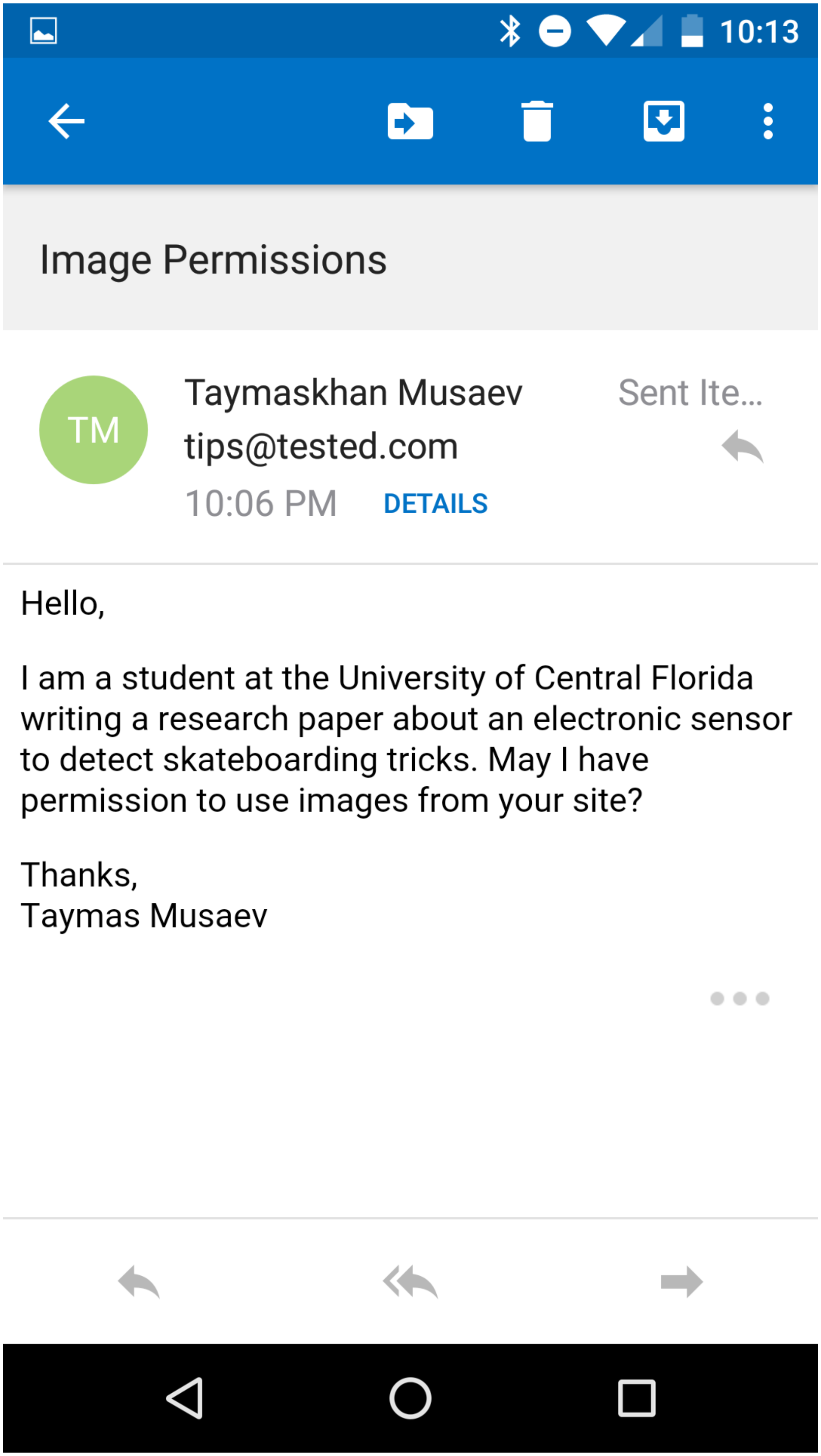
Image Permissions

TM Taymaskhan Musaev
liz@raspberrypi.org
10:08 PM DETAILS
Sent Ite...

Hello,

I am a student at the University of Central Florida writing a research paper about an electronic sensor to detect skateboarding tricks. May I have permission to use images from your site?

Thanks,
Taymas Musaev



10:13

Image Permissions

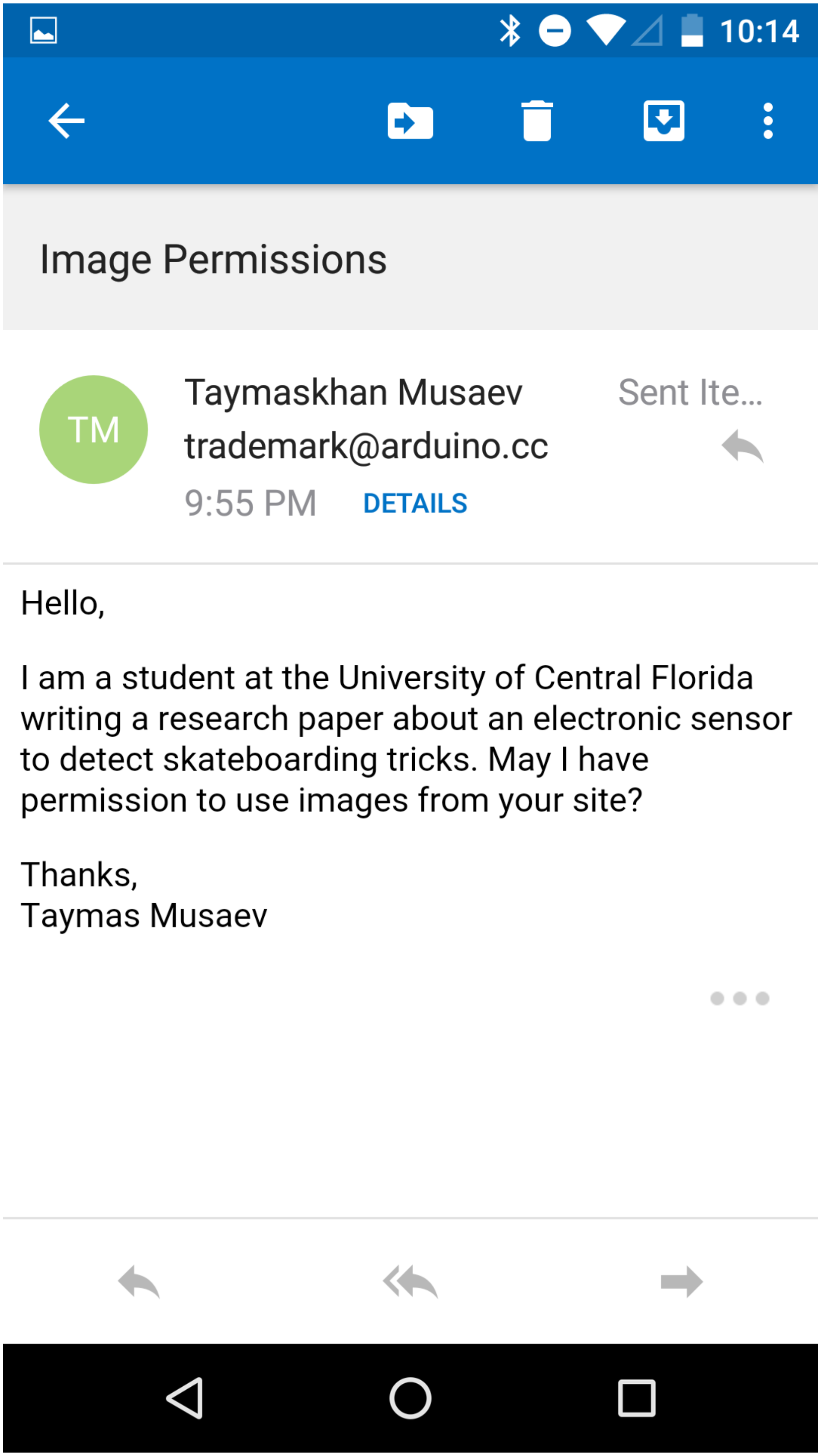
Taymaskhan Musaev
tips@tested.com
10:06 PM DETAILS

Sent Ite...

Hello,

I am a student at the University of Central Florida writing a research paper about an electronic sensor to detect skateboarding tricks. May I have permission to use images from your site?

Thanks,
Taymas Musaev



10:14

Image Permissions

Taymaskhan Musaev
trademark@arduino.cc
9:55 PM DETAILS
Sent Ite...

Hello,

I am a student at the University of Central Florida writing a research paper about an electronic sensor to detect skateboarding tricks. May I have permission to use images from your site?

Thanks,
Taymas Musaev

Hello,

I am a student at the University of Central Florida writing a research paper about an electronic sensor to detect skateboarding tricks. May I have permission to use images from your site?

Thanks,
Taymas Musaev

Wed 7/13/2016 7:41 PM
Taymaskhan Musaev
Permission to use images
To 'info@skateparkoftampa.com'

Hello,

I'm a college student making a sensor that will detect skateboarding tricks based on rotation for my senior design course. Can I use images from your website, specifically: http://skateparkoftampa.com/spot/images/tex2010knibbskfseq.jpg, to help illustrate the tricks in my research paper?

Thanks,
Taymas

Message*

Hello,

I'm currently a UCF engineering student working on my research paper and would like to request permission to use the image attached within this message. The image will not be published. Only used for educational purposes. Please get back to me at your best convenience.

Regards,
Brandon

File upload

Hello,

I'm currently an engineering student at UCF working on my research paper and would like to request permission to use the image attached within this link: http://www.mastercardbiz.com/blog/2015/09/21/considering-nfc-pros-and-cons-of-near-field-communication-for-payments/

The image will not be published. Only used for educational purposes. Please get back to me at your best convenience.

Regards,
Brandon