INTERACTIVE SELF-STANDING TRAINING BAG

UNIVERSITY OF CENTRAL FLORIDA

COLLEGE OF ENGINEERING AND COMPUTER SCIENCE

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Dr. Samuel Richie

EEL4914: Senior Design I

60 Page Draft Checkpoint Report

Group 22

Hannah Clarke: Electrical Engineering

Joseph De La Pascua: Electrical Engineering

Nicole Karam Pannaci: Electrical Engineering

Natesha Ramdhani: Electrical Engineering

Table of Contents

Table of Figures

Table of Tables

Introduction: Project Statement/Narrative

For our Senior Design project, we are creating an interactive boxing/martial arts stand up workout bag, where the athlete can choose different types of workout patterns, and receive results to view progress. Our motivation behind this project stems from the idea of creating an easy yet engaging way for a trainee of any level, to exercise, train, and track their progress in various perspectives of fight training without the need for a training partner. The Coronavirus pandemic has made this even more relevant since most training locations closed for an extended time, leaving many people without a method or physical partner to train at home with.

Another aspect of our project is that our design will be at a lower cost to those on the market, which can do some, but not all, of the actions as our interactive design. Currently on the market, interactive standup punching bags usually sell for about $25,000. Because our design can provide feedback/results after a session as well as provide interesting/fun workout patterns, the user can practice their skills in an engaging manner without paying the $40 to $70 average hourly cost of a personal trainer. Additionally, our design can be used at home which saves the user the monthly cost of a gym membership necessary for the correct equipment: a cost that ranges from $10 to $100 per month.

In order to capture as many different aspects of training as possible, the bag will have a dual-sided design that accommodates various modes that the user can train with. "Side A" will have various zones with binary touch sensors that will detect whether a certain area was hit. These zones will be loosely based on different practical locations that a fighter would be making contact with in a boxing, kickboxing, or martial arts environment against a real opponent, so that the user can have the most immersive and accurate training possible. On "Side A", the device should be able to generate random sequences of techniques landing in various areas; this can be extended to using multi-color indicators to signal which side or technique type should be used to attack the opponent. "Side B" will contain a target zone with a sensor/cluster of multiple sensors arranged within this target so that more focused training can also take place and create a complete, rounded training session. These sensors will gather data on the training session and report it back to the user so that the user can quantify their performance in different types of workouts and track their progress. Some modes that can be implemented by "Side B" include a cardio mode where users try to make as many hits as possible within a time limit, a reaction mode where users aim to land their hits based on a stimulus by the indicator, and an accuracy mode where the user practices hitting a specific area as precisely as possible.

A regular standup punching bag alone does not have the capability to improve a user's performance, since it does not provide feedback or results without a boxing/martial arts trainer. With an interactive design, the user would be able to improve performance in precision, reaction time and speed. This design should be easy to use for first time athletes as well as experienced athletes because the user is able to select different mode levels that are most suitable for them, and increment their levels as needed to continue to receive better training.

Project Goals and Objectives

General Specifications

This device should:

- Be able to implement downtime modes
    - Idle Mode
    - Display Mode
    - Off Mode
- Be able to implement exercise modes
    - Combination Generator (Side A)
        - Uses set zones on opponent
        - Binary sensors count correct hits and save data for user
        - Practice training against an "opponent" to improve the user's skill
    - Cardio Mode (Side B)
        - Uses sensor target to count hits
        - User goal: make as many hits as possible in a certain time period
        - more hits = higher score
    - Reaction Mode (Side B)
        - Uses sensor to detect time for user to hit target after illuminating indicator
        - User goal: hit target as soon as possible after it lights up
    - Accuracy Mode (Side B)
        - Uses sensor to detect how close to center mark target is hit
        - User goal: hit center as accurately as possible to increase score

Marketing Requirements

- **Durability:** Device will be able to withstand impact from regular use. (+)
- **Cost:** Device must be cost efficient for a typical consumer; priced reasonably compared to other multimode home-gym fitness devices. (-)
- **Portability:** Device must be able to be moved if necessary without large risk of damage, and must be able to be used in a variety of different environments. (+)
- **Intuitive:** The user interface will be easy to understand for and welcoming to experts and enthusiasts alike. (+)
- **Versatile:** The device will have multiple modes of operation utilizing both sides and orientation of sensors for a well-rounded user experience. (+)
- **Device life:** The device should last as long as possible; at least 5 years for the user. (+)

Technical Requirements

- **Durability:** Components will be able to stand up to impact at a rate of over 120 hits/minute. (+)
- **Power:** Components will be power efficient; no more than 1.1kW. (-)
- **Compatibility:** Components will integrate well with other components used in the device. (+)
- **Component Costs:** Total cost of type of components of good quality will be within our budget, but as low as possible. (-)
- **Implementation Time:** The time taken for the design and implementation of modes using the sensors should not be excessive. Our goal is 12 weeks or less to build and develop. (-)
- **Longevity:** Components will not die out quickly, providing approximately 5 of years of use. (+)

House of Quality

Double Symbol = **Strong** Correlation | Single Symbol = **Weak** Correlation

Technical Requirements vs Technical Requirements (roof):

	Durability (+)	Power (-)	Compatibility (+)	Component Cost (-)	Implementation Time (-)	Longevity (+)
Implementation Time (-)						
Component Cost (-)						↓
Compatibility (+)				↑	↓	
Power (-)						↑↑
Durability (+)				↓		↑↑

Technical Requirements (Green) vs Marketing Requirements (Blue):

	Durability (+)	Power (-)	Compatibility (+)	Component Cost (-)	Implementation Time (-)	Longevity (+)
Durability (+)	↑↑			↓		↑
Cost (-)	↓		↓	↑↑		↓
Portability (+)	↑			↓		↑
Easy to Use (+)			↑		↓	
Versatile (+)		↓	↓	↓	↓	↑
Device Life (+)	↑↑			↑		↑↑

Table 1: House of Quality Specification Matrix

Realistic Design Constraints

Design constraints are involved in every engineering project. The following content provides certain constraints that will apply to the manufacturing and operation of the punching bag. These constraints include: Economic, Time, Environmental, Health, Safety, Social, Ethical, Political, and Manufacturability and Sustainability constraints. These constraints provide limits to engineering designs based on standard scarcity that exists in the material world. Some constraints, such as the health and safety, are less focused on scarcity and more focused on protecting both the consumer and the commons that the consumer is free to operate in. Each constraint will be considered both individually and with regard to each other in the full scope of the project. This is necessary to be a valid constraint.

Economic and Time Constraints

The budget for our project is $500. To cover the costs of our project, our group has decided to self-fund the budget. The primary costs of the project are the punching bag itself and the electronics that will be used. The punching bag itself is the largest direct cost for this project, to lower this cost a used punching bag will be acquired. A used bag is preferred as it will be heavily modified to accommodate the electronics used. Although our selection of MCU has lowered the costs associated with the bag, multiple different sensors will be acquired and used throughout the bag, adding more cost. Although sensors are generally inexpensive, multiple sensors are needed to cover all of the sections of the bag required to bring full functionality to the bag. Some of the components will be bought twice to cover any component failure that could happen along development of the project. The Economic constraints are necessary to consider as most items need to be purchased to finish and manufacture the product. This will be finalized until the end of the project.

Another constraint to consider for the running cost of this project is the cost of electricity. This constraint should be considered not in the manufacturing, but in the final product operating cost. This product will be required to take in AC power from the wall, and therefore will require standard energy costs to operate the machine. The design of power supply and operations will factor in to how much power is needed to run this machine and therefore how expensive the operating costs will be. This constraint will differ depending on which region and power supplier are used to power the machine. Standard Duke Energy operating costs in Florida is an average of $0.12/kWh. This does not take into account any monthly fixed costs for power.

One large constraint for this project is the overall time limit of the project. The project will be done in April 2021. The research and testing of the project will be complete during the first semester of Senior Design, by the beginning of December 2020. After research and testing is complete, the manufacturing of the punching bag will begin in the second semester of Senior Design in Spring 2021. Product testing will be done during and after the manufacturing of the punching bag. Testing and manufacturing will be required to incorporate the design of both the code and the electronics in the punching bag. After extensive testing, the completed product will be presented to the judges panel at the end of the Spring 2021 semester. The time constraints are important to consider as they dictate most of the schedule this project has to follow.

Environmental, Health, and Safety Constraints

To protect the commons for the consumers and producers of different items, certain environmental, health, and safety constraints need to be considered. The main environmental constraint restricting the product is the cost and availability of electricity. Electricity in the surrounding area is supplied by Duke Energy, who owns a coal plant that provides energy and electricity to much of the Orlando/UCF area. The direct cost of the electricity, in terms of dollars, was given in the previous economic constraint section, but environmental considerations have to be taken in the context of 21st century environmental degradation. Due to power being supplied by a coal power plant, running machinery in the Orlando area contributes to further carbon emissions in the atmosphere, exacerbating climate change. To limit the effects of runaway climate change, measures should be taken in the design steps to use as little power as possible for the operation of the machine. This constraint is part of a collective effort to limit personal contributions to greenhouse emissions and should be evaluated in all future engineering projects and designs. This personal responsibility cannot replace certain standards set for power efficiency and regulations on limits of power and emissions usage.

Due to consumer use, health and safety constraints are required for the design of the machine. The most pressing issue is the electrical wiring of the machine. The sensors on the punching bag will need to be wired to the MCU to perform the calculations necessary to the project. These wires should be out of the way of the punching surfaces for the safety of the user and the components. Another safety consideration is the overall weight of the machine. With the added components, the weight of the machine should not go over a threshold that would make it unsuitable to be carried without the water or sand stabilizer. This is important both for the usability and for the safety of the consumer. Due to the relatively low weight of the MCU, wires, and sensors when compared to the bag itself, this constraint should not be too limiting. One final constraint is the heat of the circuitry and the MCU. The MCU should be housed in heat resistant plastic in order to shield it from the rest of the machine, which could be damaged when exposed to any heat from the MCU.

Social, Ethical, and Political Constraints

Surrounding the manufacturing of this project there are several social, ethical, and political constraints for our design. The social and ethical responsibility to lower power use, and therefore emissions use, was already mentioned in the section above. Another ethical constraint applying to this project is the responsibility to report the correct calculations for the workout modes. This product will be used by athletes who want to complete standard workout regiments many times a week. To aid this strict schedule, our machine needs to both work every time it is powered on to be used and to report accurate information at the end of each section for the athlete to assess. In order for athletes to rely on our product the trust should be gained by effective and habitual use. Sticking to this constraint will also fulfill political constraints including ethical manufacturing of the product. In other words, if the product does not report information correctly, customers have grounds to stop using the product and even grounds to sue depending on the accuracy of our product's ending information and assessment. Avoiding false advertising of our product in the future would work to stop this particular concern, especially around the assessment of the workout our product performs.

Manufacturability and Sustainability Constraints

Sustainability is a constraint with a lot of focus for this project. Not only should sustainability of power be taken into account for but also sustainability of building materials. Electronics especially do not fare well once thrown out, in fact many electronics need to be specifically recycled to be disposed of safely. Our component disposal will follow all electronic recycling guidelines outlined by the Environmental Protection Agency (EPA). Only the regulations from the EPA have to be followed as the State of Florida has no additional E-waste laws or regulations. An E-waste recycler with a third-party certification is recommended for use by the EPA and the Florida Department of Environmental Protection. Another constraint for manufacturing sustainability is the punching bag itself. To meet sustainability (and cost) constraints, a used bag will be acquired to use for the prototype. This assures that new material and products are not wasted in the construction of the machine. Buying used components is one major consideration we made to meet both the economic and sustainability constraints. Two constraints were effectively applied during our buying habits.

Standards

There are not many standards that apply to the interactive workout bag product. The few that do apply include power supply standards, standards for fitness equipment, communication standards and PCB standards. Each type of applicable standard is examined, as well as its correlation to the design.

Power Supply Standards

IEC 60906-2:2011

Also known as the IEC system of plugs and socket-outlets for household and similar purposes - Part 2: Plugs and socket-outlets 15 A 125 V a.c. and 20 A 125 V a.c., it is a standard that applies any to the NEMA 5-15-P , also known as the "3-prong grounded plug" connection. This standard explains that this connection will provide protective earthling to the equipment that is connected and to the conductive parts of the socket. It also is electronically separated from the protective earthing circuit to provide electrical noise inmunity.

This standard applies to this project since we will need this type of plug to be compatible with the standard american wall outlet. Our design will also need the three prong connector (positive, neutral and ground) for our circuit design. It is also critical that our system has no electrical noise from the power supply, which the IEC 60906-2:2011 standard defines.

Regulation Workout Bag Standards (weight, size, etc)

The American Society for Testing and Materials (ASTM) is an international organization that standardizes the safety and testing of materials. The standard ASTM F2276-10(2015), also known as the Standard Specification for Fitness Equipment, is a standard that all designers and manufacturers for a fitness equipment must follow. The standard lays out specific criteria which is applicable to the workout bag design. In the standard, it is defined that only individuals age 12 or older can use this equipment, that the product must include all the documentation on how to assemble/build the product, include a list of parts and instructions, and provide adequate warnings. This standard also defines which exercise equipment is required to be used in an indoor setting or environment.

This standard applies to our project design because the workout bag falls under the classification of fitness equipment. Therefore, our project design will need to include a parts list, instructions and applicable warnings that include an age restriction.

PCB Standards

IPC-221B

The workout bag system implements two printed circuit boards, one in the base system and one in the UI system (remote). Association Connecting Electronics Industries, previously known as the Institute of Printed Circuits (IPC) sets the standards for all types of printed circuit boards (PCBs). In conjunction with the IPC standards.

The standards that apply to our PCB designs from the IPC are in IPC-2220 series. The IPC-221B standard is the standard that defines the fundamental design requirements (also called the generic design requirements) for designing the printed circuit boards, component mounting, and interconnecting structures. Because our PCBs will include a design and mounted components, it is important to follow the guidelines set by these standards.

Communication Protocol Standards
IEEE standard 802.15.4

The UI system and the base system communicate low rate wireless personal area network (LR-WPAN) which is defined by the IEEE 802.15.4 standard. The IEEE 802.15.4 standard is designed to create a WPAN that is low power. The IEEE 802.15.4 standard defines the physical layer (the bottom layer of the protocol layers) and affects the MAC layer, a sub layer of the network layer.

Zigbee, which is used to control the communication between the UI system and the base, builds on the IEEE 802.15.4 standard. The physical layer controls the interface of transmission, which include the transceiver and the transmission frequencies. The MAC (medium access control) layer controls the MAC frames that are then sent using the physical layer. The IEEE 802.15.4 standard also has three types of nodes on the network, a coordinator node, router node and device node. These nodes are able to interact as peer to peer networks or in spoke-hub networks, pictured below.
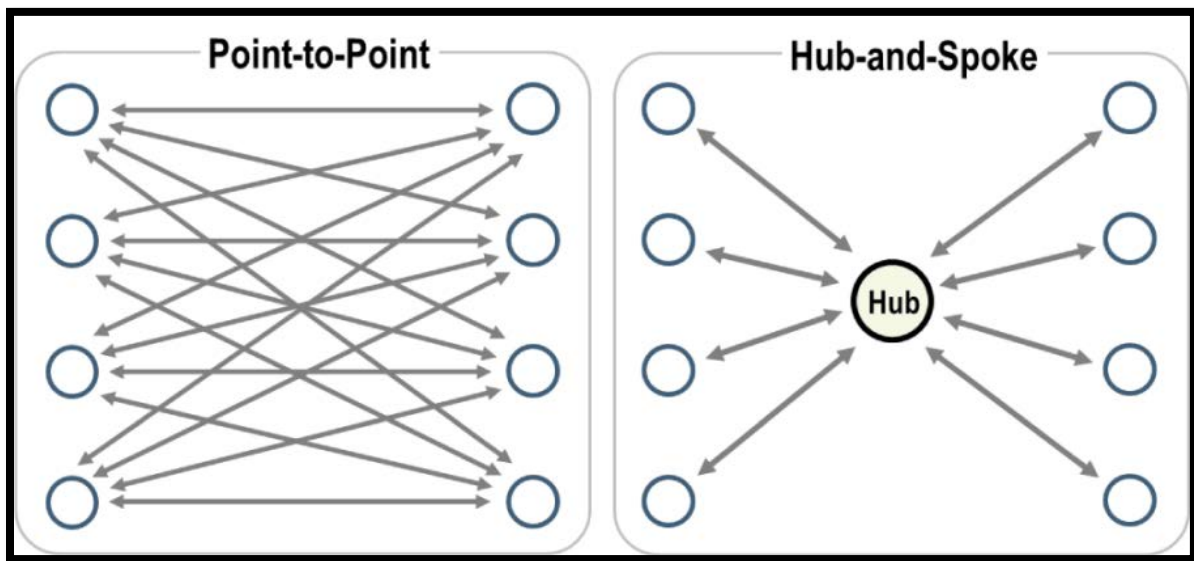
Figure 1: Point to Point (Peer to Peer) vs Hub and Spoke networks [14]

The IEEE 802.15.4 standard also defines the RF parameters for the communication. In North America, the IEEE 802.15.4 standard uses 902 to 928 MHz frequency bands. There are a maximum of 10 supported channels with a bandwidth of 2MHz and a 0.6MHz guard band. The data rate is also defined at 40kbps. Although the data rate is low, this communication standard is perfect for our design, since only a few bytes need to be sent across the network, and the communication system needs to be low power.

Market Research

The idea of a dynamic interactive punching bag has existed for a while. The first time it was patented, a device like the one we are designing was back in 2010 [1] in which a patent was submitted for a martial arts bag with one or more sensors for measuring a workout performance. In which, they also show many pictures of many portrayals of this type of design in which our design is like a combination of the Figure 1 and figure 3A. In figure 1, it will have a coach that will show on the screen to help users to hit the bag. In the 3A case, it would impact areas around for the user to hit but it still does not mention how the results are recorded and shown to the user.
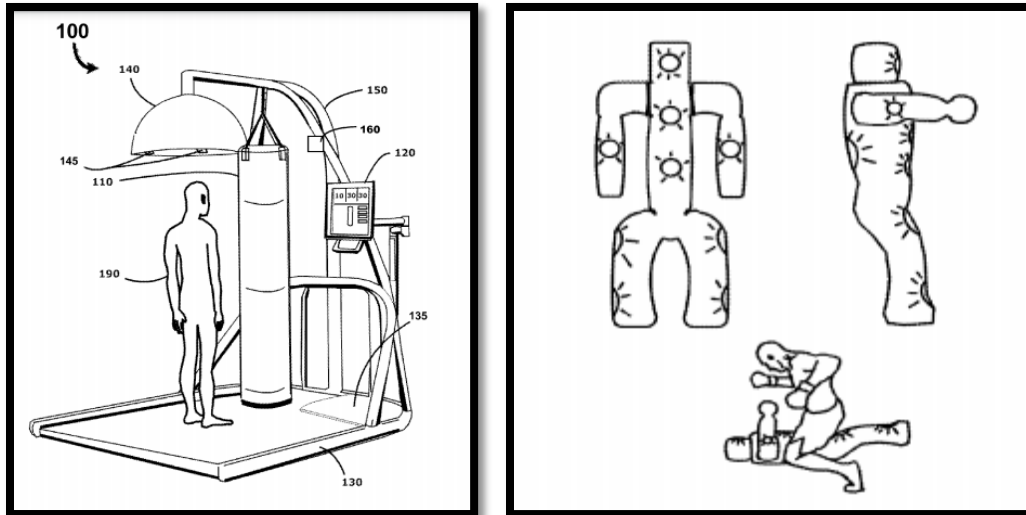
Figure 2 (Left): The main idea for the US Patent US20110172060A1
Figure 3 (Right): A variation of the US Patent US20110172060A1

We also found two research articles from the Czech Republic in which they used [7]and [8] In which, both measure a direct punch applied to a martial arts bag using strain gauges inside of the martial arts bag. In regards to the type of strain gauge used, both [7] and [8] use a strain gauge type SRK-3/V , and in [8] also uses a strain gauge L6E-C3-300Kg in their measurements. It was found interesting since in [8] came to the conclusion that L6E-C3-300Kg strain gauge has a better precision, speed, and simpler to use than the strain gauge type SRK-3/V, but in [8] that was made with the same people used the latter instead to further intensify the research about measurement direct punch force.

PADIPATA™ created an interactive punching bag, but it has all its sensors in the middle of the punching bag and it only leaves the user to use punching instead of using the user's legs as well. It is quite like the Figure 2 in the [1] reference. It comes in three sizes, and it is made by alloy-steel frame for easier assembly. It also has surround sound with an interactive screen inside of the band, and there are no iron chains in the structure of the device. This product was created in Changsha City, People's Republic of China and retails for a price of $25,000 (USD); our intent is to create a device that is much lower in cost than that, so that it is accessible to a wider audience than the Padipata bag. The device our team will create should also employ different interactive modes to make it more efficient for the user.

Technical Component Research and Standards Comparison
Sensors (Nicole)
Technical Standards
Sensors used in this device should:

1.1	Be durable enough to withstand the force of a punch/kick.
1.2a	Be able to detect distinct, rapid, successive hits in different locations (Side A).
1.2b	Be able to detect distinct, rapid, successive hits in the same location (Side B).
1.3	Be able to detect the location of a hit relative to a target point (Side B).
1.4	Consume a low amount of power.
1.5	Cover a zone with no more than four individual sensor units.

Table 2: Technical Standards for Sensors

Candidates for Use in Device
We will need sensors for our project, and these are the research done to find values of them. The original concept for the sensors in this device were inspired by the Dance Dance Revolution(DDR) home mats since these mats can support the weight of people stepping to it with extra force due to the excitement of the people when they play this game. These DDR mats use buttons to signal that an arrow was pressed, so no sensors are used. We could incorporate the following alongside/ instead of the buttons to enhance the how much force someone should apply to the self-standing kicking bag and/or the precision of their pressing to the area they have to punch. These components will be under a tarp to protect the electrical components.

Piezoresistance Sensors
The piezoresistance is the change of electrical resistance when a force is applied to it. These use a Wheatstone bridge to measure the change of resistance. It is very sensitive to changes, low cost, and resistant to many diverse changes. The drawback to this type of sensor is that it requires a lot of power for it to work.

Piezoelectric Sensors
Piezoelectrics use materials, such as quartz crystals or specially formulated ceramics, which generate a charge across the faces when pressure is applied. A charge amplifier converts this to an output voltage proportional to the pressure. A given force results in a corresponding charge across the sensing element. However, this charge will leak away over time meaning that the sensor cannot be used to measure static pressure. Low power and robustness. The drawback is that they are very complex, and they can only be used for dynamic pressure measurement.

Strain Gauge
A strain gauge is a mechanical way to measure the way the material changes when force is applied to an area. The possibility is using four long linear strain gauges to cover vertical, horizontal, and each diagonal sides, sharing the middle as the accuracy measurement of the hit, of an area to help the user to improve their precision and accuracy in each hit, as well when the user is hitting the area, this could tell the user to use more force to keep the same amount of force in each hit. Strain Gauges are a possibility since these types of sensors are robust and their whole purpose is to feel the change of movement of an area. This sensor will be attached to the tarp and due to the force and change of strain this component will be able to sense it and send this change to the software to count it as a hit. If used, it would be ideal to have a 45 degree stack plus a horizontal in each area or a Membrane Rosette Strain Gauge since this one covers a more circular area than having four linear straight gauges.

Inertial Measurement Unit (IMU) Sensors

IMUs will allow us to know the direction and force of every hit on the target on side B of the design. An Inertial Measurement Unit (IMU) is a type of device that includes gyroscopes and accelerometers to measure angular rate, force, and force. An IMU works by collecting the raw data of what surrounds the device, and tells its user about the X,Y,Z surroundings of the device. Some uses of IMUs include robots, drones, video game remotes, aircrafts, antennas, GPS, sports applications, and virtual and augmented reality. IMUs are beneficial for our project since it will help the user tell the force and direction in each hit. The IMU will sense to move in consequence of the hit from the user and it will be recorded each time and then it will be processed by the MCU to make each hit clear and different from each other. The type of sensor that would be beneficial to our project would need to have the capabilities to sense sports movements and/or motion tracking and gesture location.

Accelerometers

In case that the IMUs on side B do not work as desired, accelerometers shall be used instead. Accelerometers measures the acceleration forces applied. In this case, it shall tell us the force that is applied to the hit area. Uses of accelerometers include cars, machines, buildings, process control systems, safety installations, and airbags. They will be satisfactory backup options for the devices if IMUs do not work; however, they only report force magnitude of the impact, not the direction. This information will be able to be manipulated to calculate desired results, regardless.

Binary Sensors

For Side A, we shall need five binary sensors. Binary sensor is a sensor that returns a 0 when it is not sensed and returns a 1 when it is sensed. Every switch and button can be considered a binary sensor since it gives a 1 when is pressed and 0 when it is not pressed. Some applications of binary sensors include buildings, cars, electronics, and robots. Binary sensors are the best option for this part of the device since it is basic but sufficient for the desired use and user input.

Design Matrix and Analysis

Standard:	1.1	1.2a	1.2b	1.3	1.4	1.5	Total
Accelerometers	2	5	4	1	5	5	22
IMU	1	4	5	5	4	4	23
MIMU	3	3	3	4	1	1	15
Strain Gauge	5	1	2	3	2	2	15
Rosette Strain Gauge	4	2	1	2	3	3	15

Table 3: Design Matrix for Sensors on Side B

We will be discussing accelerometers and IMU since these had the highest ranking among all the sensors. In **1.1**, these have low ranking since without any protective layer, these sensors will not survive any damages done by any hits/kicks by the user. In **1.2b**, IMU is ranked higher since it can differentiate each force of each hit/kick with the direction of the impact, and Accelerometers only will measure the force of the hit/kick of the impact. In **1.2a** if it is just hitting in one area, accelerometers will do the job fine since it only measures the force applied to the hit/kick, while the IMU will collect the direction applied to the location as well. In **1.3**, IMU will measure the direction of the hit, so it will say how close it was to the target point. The accelerometer is the least desirable here since it only measures the force than according to the hit from the target area. In **1.4,** Accelerometers only measures the force and we can find one that can consume a lower amount of power for it to work. The IMU will not consume as much power as the accelerometer but since it measures more than one sensor, it could consume more power than the

accelerometer. In **1.5**, The Accelerometer contains just one sensor and that one sole sensor measures the force of the hit and IMU possess different types of sensors, so it might just be one device that measures everything, but, in this ranking, it was considered as well all the sensors that are inside of the device.

Indicators (Natesha)

Technical Standards

The indicators used on this device should:

2.1	produce light with high enough intensity to be visible behind a translucent cover
2.2a	(each individual unit should) be durable enough to withstand repeated hits
2.2b	(connections between each unit should) be durable enough to withstand repeated hits
2.3	be able to easily shape as needed without excessive strain
2.4	sufficiently indicate zone in multiple colors with as few physical units as possible
2.5	have a lifetime comparable to that of the device itself.

Table 4: Technical Standards for Indicators

Candidates for Use in Device

There are three main options for the indicators used: addressable LED strip lights, individual LEDs, and electroluminescent (EL wire). More detail about each will be provided in the next sections, before being compared directly to each other in a decision matrix to pick the best candidate for indicators in this device.

Addressable LED Strip

One indicator option is to use strips of addressable LED lights. These have become commonplace in recent years due to their popularity for indoor and outdoor decorative applications. Each segment of the LED strip requires an input voltage for power and three connectors containing the needed value of each primary color (red, green, blue) to create the desired color. The segment itself contains RGB LED light units and a controller to determine the values for each primary LED color. The strip can be cut in between segments and attached together by connecting the input of one segment to the output of another; each segment can work individually as long as it is a part of a complete closed circuit.

Since each LED is individually addressable, this approach would allow for a large range of visual effects. Additionally, the popularity of these lights means we will have a large set of features to choose from, such as mounting color, input voltage, and differing color schemes. Some of these features come at the cost of others; a strip with a waterproof sheath will be more durable, a feature we will likely need. However, these sheathes may make our such strips significantly more difficult to shape and cut to our design. Other features will challenge us to balance their worth with their cost; strips with higher pixel density can display higher resolution effects, but at increased costs. Pricing is heavily dependent on the features offered as well as the length of the strip, so we would need a preliminary design and lighting patterns ready in order to make a choice regarding this option.

LEDs

Individual LEDs are another option for indicators: using individual LED bulbs with some kind of material within which the LEDs are scattered will allow the light to diffuse through it so that the entire panel is illuminated. This would likely be the cheapest option since each individual component has the lowest cost and the group has the largest control of exactly how they would be used. Although LEDs themselves are fairly durable due to their plastic, rounded shell that is unlikely to break when faced with blunt force, their connections to each other may end up compromised if hit incorrectly. They can be controlled directly by a microcontroller and a DC power source, so they will be fairly straightforward to program for different effects and they require little power to function (meaning that they will not heat up easily). LEDs can burn out if left on for too long, though, making the life of the indicator unpredictable.

Electroluminescent Wire

Electroluminescent wire (EL Wire) is made of a stiff, phosphor coated core, wrapped by a corona wire acting as a capacitive plate around this core and an outer PVC covering protecting the wire. EL wire, as well as derivatives like EL tape and panels, are frequently used in clothing applications due to the fact that they produce no heat, consume less power than LEDs, and can retain their shape while still being flexible enough to curl around a finger. This makes them a good fit for our project, able to withstand accidental hits by the user better than an LED based approach might be.

Since EL wire is capacitive, it cannot be controlled with a typical PWM signal. Instead, they are powered by an AC inverter, with the frequency and voltage of the AC signal dictating the wire's brightness. The wire's brightness is therefore dependent on the load it places on the AC source; using more or longer wires will dim them. Additionally, the wire dims gradually over time as the core's phosphor degrades, with a typical lifespan of 1-2 years if outdoors or exposed to direct UV light, or about 3 years if in normal indoor conditions and maybe even longer in lower light conditions. Their lifespans are also dictated by the voltage being applied to the wire, so by regulating the voltage applied, we may be able to make the wire last longer. Ideally, making the EL wire segments modular would be the best option for both profit and allowing the user to keep the device functioning as long as they would like to use it (with the bonus benefit of allowing the user to buy whichever different colors they would like to use).

To use EL wire in our design, we will need to account for either a DC to AC inverter with sufficient voltage/frequency. Alternatively, we can also connect the wire directly to an AC source and eliminate a potentially noisy inverter, designing separate circuitry to control these AC components.

Design Matrix and Analysis

Standard:	2.1	2.2a	2.2b	2.3	2.4	2.5	Total
Addressable LED Strip	3	3	2	1	3	3	15
Electroluminescent Wire	2	1	3	3	2	2	13
Individual LEDs	1	2	1	2	1	1	8

Table 5: Design Matrix for Indicators

The best scoring option for our project's indicators are the addressable LED strips. These are strips of many LEDs, which a controller is able to individually address to control brightness and color. They offer more than enough brightness **(2.1)** and lighting control **(2.4)** for our project, and are relatively more durable **(2.2)** and long-lasting **(2.5)** than our other options. One potential shortcoming of these strips may be their stiffness; the thickness of some LED strips can make them difficult to bend into tight shapes **(2.3)**, but the benefits of using this indicator far outstrip this inconvenience.

We can easily rule out individual LEDs, our lowest scoring option. While this option can offer the same potential as LED strips, we would need to hand-solder and wire each individual indicator, making this choice significantly more fragile and difficult to shape (2.2, 2.3) than other indicators. Their brightness (2.1) and ability to indicate in different colors (2.4) is limited by this as well, since we would like to avoid using too many physical units, and their individual brightness cannot be pushed too high without risk of burnout (2.5).

Electroluminescent wires came close in score to addressable LED strips. These devices are made of a phosphor-covered core wire and two thin outer wires, and produce light (the color is chosen by its plastic sheath) when AC voltage is applied. Ultimately, they are limited in their brightness (2.1), especially over long periods of time, as they lose 50% of their brightness in 3000 hours (2.5). They are quite durable (2.2) and easy to shape to our needs (2.3), but we would need to pack multiple runs of EL wire in an area to use multiple colors (2.4), while LED strips are able to show multiple colors and effects with one unit.

Processor & Memory (Joseph)

Technical Standards

The processor used by the device should:

3.1	Have sufficient input lines for all sensors and communication necessary.
3.2	Have sufficient output lines for all indicators and communication necessary.
3.3	Communicate with the UI system to provide raw data.
3.4	Require internal ROM and RAM.
3.5	Work with a volatile memory large enough to store data from multiple sensors.

Table 6: Technical Standards for Processor and Memory

Considerations for Components

Four main contenders for our main processor were considered. The processor is required to take in inputs from the sensors and run calculations according to the data to assess the workout after it is completed. Our processor will also be the component that starts and controls all aspects of the workout modes. The main considerations taken to choose what type of processor was most appropriate for this design are: sufficient input lines, sufficient output lines, able to communicate with the sensors, and sufficient ROM and RAM. These four considerations were evaluated individually for each processor design and points were assigned to each selection considering how well it applied to that consideration. The product with the most points at the end of this analysis was chosen. Ultimately the processor chosen was the standard MCU.

FPGA

A field-programmable gate array (FPGA) is an integrated circuit that can be configured to have certain integrated circuitry configured by the designer after manufacturing. The main difference between an FPGA and a standard processor is the hardware of the FPGA can be configured as well as the software. To encode a FPGA, a hardware description language (HDL), like Verilog, must be used. The HDL is used to change the internal circuitry of the FPGA by assigning different components to the circuitry of the board. Once configured, SW can be loaded onto the FPGA and it can be used as a flexible MCU to perform calculations.

When evaluating the design considerations with the FPGA, the FPGA can be standardized by its compatibility with our final project. The first consideration to be evaluated is the number of input lines for the processor. The FPGA has User I/O pins that can be programmed to be input or output pins. Given the flexible nature of the FPGA, the pins themselves are designed to operate multi directionally. Although the pins have this flexibility, there are not as many total I/O pins on the standard FPGAs, certainly not as much as other options considered. The next design consideration is the ability to connect to and communicate with the sensors. FPGAs are compatible with connections such as UART, I2C, and SPI, making it compatible with the sensors used in the machine. The last design consideration evaluated is the availability of sufficient RAM and ROM to store the data from the sensors. FPGA boards have RAM and ROM of sufficient size, at around 256MB of RAM. Although many of the considerations are met with the FPGA, the overall cost and lack of I/O make this a lower contender to be included in the final design.

DSP

A digital signal processor (DSP) is a specialized microcontroller chip that is optimized for the needs of digital signal processing (DSPg). These processors are especially and widely used in fields such as audio signal processing, telecommunications, and in consumer electronics such as mobile phones. The main use for these processors is to measure, filter, and compress analog and digital signals. It does this by first converting the analog signal to digital, then applying processing filters, then converting the output digital signal to analog signal. Although DSPg algorithms can be run on any microprocessor, the DSP offers advantages in areas such as data compression that make DSP favored to use for this operation when it is the main operation for a system.

The first consideration to be evaluated is the number of input and output lines. There has to be sufficient I/O to collect the data from all of the sensors simultaneously. DSP processors have sufficient GPIO pins and 3 full UART interfaces, four SPI connections, and three I2C interfaces. The DSP has sufficient I/O to incorporate all of the sensors into the microcontroller. Given that the sensors communicate using UART, I2C, and SPI, there is enough I/O to healthily support the data collection. The last consideration is the amount of RAM and ROM that the processor is able to support. The DSP has up to 4GB of expansion, with an onboard amount of about 1024KB. This is enough expansion to store the data from the sensors. The DSP meets all the requirements for the project, however because it is specialized for signal processing, the DSP does all of these functions while being more expensive than other, non-specialized processors.

Raspberry Pi

The Raspberry Pi is the most all-in-one processor that was selected for consideration. The Raspberry Pi is essentially a small computer included with a chip, I/O, and a specialized power input. The Pi acts as a desktop computer on a chip, even being able to load an operating system on the chip. The Pi is usually used for consumer grade projects where an operating system can be beneficial to add more complexity and control to an electrical project. The raspberry pi is the second most expensive processor at $50.

The PI has 40 standard GPIO pins used for I/O. There are also standard USB 2.0 and 3.0 ports for I/O. Even though there is an adequate amount of I/O. The key functionality of SPI, UART, and I2C is missing from this processor. This is enough to disqualify the raspberry pi from consideration, as the sensors communicate using the UART, SPI, or I2C protocols. The Raspberry Pi is not the component to be used.

MCU

A microcontroller is an integrated circuit that is designed to perform specific operations for a closed system. The standard components included with a microcontroller are the processor, memory, and general I/O. These are typically all included together on a single chip. Microcontrollers are primarily designed for embedded tasks that are automatically controlled. These tasks include anything from

automobile engine control systems to implantable medical devices. Any time an automatic aspect of control is needed a microcontroller is generally an acceptable processor to use.

To evaluate the MCU, the first consideration of I/O lines needs to be addressed. There is a large amount of variability of I/O pins for different microcontrollers. The amount of GPIO pins that a standard MCU has is around 50 pins. This is a sufficient amount of I/O pins for the assigned tasks. The next consideration is the ability to communicate with the sensors. With 2 I2C ports, 4 UART ports, and 2 SPI interfaces, there is plenty of room to include all of the sensors for the machine. The final consideration is the amount and availability of RAM and ROM. With a ROM of 128KB and RAM of 8KB there should be sufficient memory to store the data from the sensors. Expansion of RAM and ROM is also available with the microcontroller. Due to the relatively low cost and the satisfaction of all of the considerations, the MCU will be chosen as the processor for the project.

Design Matrix and Analysis

Standard:	3.1	3.2	3.3	3.4	3.5	Total
FPGA	1	4	1	3	9	9
DSP	4	1	4	1	10	10
MCU (MSP)	3	2	4	4	13	13
Raspberry Pi	2	3	2	4	11	11

Table 7: Design Matrix for Processor and Memory

3.1 is the power analysis. For the point selection, the highest level was given to the processor with the lowest input power necessary to run. This is the DSP MCU. The MSP MCU was the second lowest input voltage requirement. This makes the MCU and DSP more attractive for total consumption of power and therefore cheaper operating costs. The raspberry pi runs at 5V making it a contender for second place with the MSP. The FPGA board requires the most input voltage at 7V-15V, making it the most expensive operating cost.

3.2 is the compatibility analysis. The main components to compatibility for this project is speed of the processors and the I/O options to connect to the sensors and screens. FPGA has several features that make it very compatible with this project. It has the second fastest speed, it has ample RAM (256MB) and flash memory (16MB), and has SPI and UART interface for connection. Another unique feature of the FPGA is the ability to change both the software circuitry and the hardware circuitry on the same board. This will allow us to code a different hardware implementation for each mode of operation, allowing for much greater flexibility of operation. The Raspberry pi is another contender for this mode due to its fastest speed and the amount of I/O that is available both on the baseboard and with extensions to the controller. The I/O is USB and standard GIPO headers so it has greater compatibility with both USB and GPIO sensors. It has 4GB of SDRAM. The MSP MCU has many options in terms of I/O, particularly UART, I2C, and SPI are the three main connections. But the total processing speed is only up to 25MHz, making it the slowest processor, but it edges out the DSP due to many different options of I/O. It also comes with 128KB of flash memory and 8KB of RAM. The DSP controller has middle road speed, but only works with I2C connection, making the connection type limited. The DSP has 2MB of flash memory.

3.3 is the cost analysis. At $120-$200 the FPGA is the most expensive board, followed closely by the raspberry pi at $100. The MSP and DSP MCUs are each relatively the same price at around $15.

3.4 is the implementation time analysis. The implementation time for the MCU and Rasp. Pi are very similar due to both being already compatible microcontrollers with set hardware circuitry. This allows us for a more "plug and play" method for these processors. The FPGA board has programmable hardware circuits, which increases the implementation time because we have to code both hardware and software through the board. The FPGA does have I/O already part of the controller however which decreases the implementation time. The DSP MCU does not come with the same I/O board as the other three options, soldering would be necessary to implement the DSP MCU, making it have the worst implementation time. After considering all these options in the table above, the MSP MCU was determined to have the highest total score after the analysis.

UI System (Hannah)

Technical Standards
The device's user interface should:

4.1	Provide completed results to the user including number of total targets, number of total successful hits, total time per mode run, and average time per hit.
4.2	Receive user inputs such as turn on/off, select mode, and select level of difficulty.
4.3	Be capable of communication between itself and the base.

Table 8: Technical Standards for UI System

Considerations and Candidates for Components

Displaying Results and Messages to User
One of the main concepts that makes the workout bag an interactive design is the ability to show results of the workout session to the user in a simple and effective way. To do this, several types of ways to display results were examined and compared.

A simple option to provide the user with results would be to incorporate a USB port that the user can download the results from. This option would require the user to acquire a flash drive and a machine to view the results. Because we want our design to be self contained, meaning the user shouldn't need any other devices to use all the functions of the workout bag, this option is less favorable as the only method to provide the user with results. If there is extra time, our team can quickly implement this method as an additional function.

Another approach to provide results to the user, would be to incorporate a screen that displays results. There are several different types of screens that are comparable. The two types considered were a LED monitor and an LCD display. The LED monitor supports inputs of VGA, DVD or HDMI. This would require a processor that is fast enough along with a framer chip to support HDMI output. The LED monitor would be viable, if the processor in the handheld remote is a beaglebone (or similar), however, the cost of the LED monitors is far above our budget for displays, roughly $200 to $300. The LCD display (2004) is a compatible display with the microcontroller (by using I2C or SPI) and is within our budget. The LCD displays also come with a display driver, therefore no additional driver chips are necessary. Although the display can only support 4 rows with 20 columns, it is sufficient for the data that is being displayed.

Receiving User Input
In order for the workout bag to be an interactive design, there must be a way for the UI system to receive user inputs. User inputs include selecting the workout mode for the session, selecting a difficulty level, and a way to begin, cancel, or exit the workout session. To reduce the complexity of the design, the types

of user input options considered are limited to switches, buttons and LED lights, instead of considering any touchscreen inputs.

The best option to receive the user input would be a push button for each type of user input, instead of toggle switches. If toggle switches were used, each type of mode and difficulty setting, resulting in almost twice as many toggle switches as push buttons. In order to reduce the amount of push buttons, the options will need to be cycled through each time the button is pushed.

The user input also requires a way to indicate to the user which selections they have made, since the push buttons have to be pressed to cycle through the options. Two methods to complete this would be to have an LED light that is enabled by each option or have the option displayed on the LCD screen. Having an LED enabled for each option would require several extra components that use up several of the GPIO pins, therefore displaying the option on the LCD screen is the best choice.

Communication between the Base System and the User Interface

The UI system of the workout bag requires a way to communicate with the base system of the design. One way for the base to communicate to the UI system would be a wired connection, however, due to the amount of physical movement around/near the base system, there is a high possibility of the user tripping over the wires and causing injury to the user or damage to the device. Several different approaches were examined and compared.

There are several types of wired connections that are comparable. The two compared for this project were SPI and UART. UART stands for universal asynchronous reception and transmission. The advantages of using UART are the simplicity of implementation and operation, the lack of need for a clock, and the use of error detection (parity bit). However, UART's frame sizes are limited to 9 bits and transmissions speeds are relatively low. Another disadvantage is that each chip's baud rate must be within a certain percentage of each other to avoid data loss.

SPI stands for serial peripheral interface. The advantages to using SPI is that the implementation and operation are also very simple. SPI is also able to send data continuously due to not requiring a start and stop bit for data transmission. SPI also has a faster data transmission than UART. However, there are downsides to using SPI as well. Such as the lack of flow control and receiving acknowledgments, and the use of more pins which limits the number of devices the chip can support.

There are also several types of wireless connections that are comparable as well. Three types of wireless communication types that were compared are WIFI, Bluetooth and Zigbee. The advantages to using WiFi allow each MCU to connect to the home's router, which allows the UI system to communicate with the base without line of sight. However, WIFI modules consume more power than the other options, which is critical when the UI system is battery operated and they are usually more expensive. The advantage to using Bluetooth is its relatively low power and inexpensive, unlike WiFi. However, the range of the communication is limited to relatively close range. The advantage to using Zigbee is that there is no need for a router and can operate in larger distances than Bluetooth. Zigbee also consumes a low amount of power, which is suitable for battery operated devices. The disadvantage to using Zigbee is the communication requires line of sight to the other device, and has a relatively low data rate.

The best fit for our design would be using Zigbee for the communication between the UI system and the base system. The UI system is battery powered, therefore, Zigbee's low power consumption has a high impact on the design decision. Although the data transmission rate is relatively low, our design only needs to transmit a couple bytes of data. Because this will be our teams first exposure to using Zigbee, an alternate choice of using SPI for a wired connection, as our team is familiar with this interface.

Design Matrix and Analysis

Category/Standard	Options		Ranking		
4.1: **Display Results**	LCD Screen (LCD2004)	3			
	USB Port	2			
	LED Monitor	1			
4.2: **Receive Input**	Buttons with LCD		3		
	Buttons with LED		2		
	Toggle Switch		1		
4.3: **Communication** **Base ↔ UI**	Zigbee				5
	WiFi				4
	Bluetooth				3
	SPI				2
	UART				1

Table 9: Design Matrix for UI System

Standard **4.1** (provide results to user)'s highest rank was given to the LCD screen, as it is compatible with many processors, and can immediately display results to the user. However, this is not a finalized parameter of the design, as the display will be the last component that is configured as its dependent on the processor selection and type of communication used.

Standard **4.2** (receive user inputs) was won by the push buttons w/ LCD Screen to show selection option. Push buttons are relatively easy to implement and to program a debounce method. As push buttons and LCD display units are relatively inexpensive, it is an easy and cheap way to be able to cycle through modes, while showing which sections have been made. Since push buttons can be easily programmed to cycle through modes with repetitive pushes, few buttons are necessary for proper execution.

Standard **4.3** (communication between base and UI system) is best matched in our design by using Zigbee for the communication between the UI system and the base system. The UI system is battery powered, therefore, Zigbee's low power consumption has a high impact on the design decision. Although the data transmission rate is relatively low, our design only needs to transmit a couple bytes of data. Because this will be our teams first exposure to using Zigbee, an alternate choice of using SPI for a wired connection, as our team is familiar with this interface.

Power Supply (Joseph)

Technical Standards

The device's power supply should:

5.1	Connect to a standard US AC 110-120V input.
5.2	Use an AC-DC voltage converter (such as a 20V 60Hz AC to 12V DC converter) to provide power to the system's DC components.
5.3	Provide AC voltage of necessary magnitude to any components that require an AC input.

Table 10: Technical Standards for Power Supply

Considerations for Components

The components that require power are the MCU, the sensors, and the LED strips. These can be powered in combination with a standard AC/DC power supply along with the output power pin with the MCU. The AC/DC power supply will need to take in a 120V/60Hz signal and convert it to a 12V DC signal. This signal will be used to power the LED strips, which require 12V of DC voltage. To power the MCU, an input voltage of 1.8 to 3.6V DC is required. The 12V output of the AC/DC converter will be lowered to this level with a DC/DC 12V-3.3V converter. After the conversion, the voltage will be the sufficient level to power the MCU. The power to the sensors will come from the MCU output power.

AC/DC and DC/DC Conversion

There are several different conversion designs that could be used for the AC/DC and the DC/DC converter. The design for the AC/DC conversion is the flyback AC/DC converter. The schematic of a flyback converter is shown below:
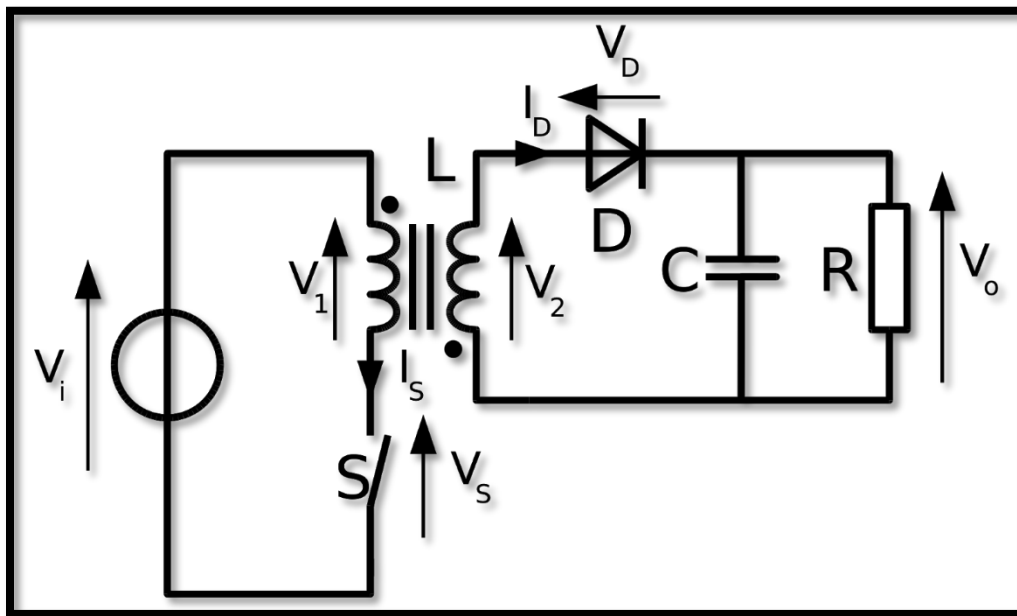

Figure 4: Flyback Converter schematic

This converter uses an inductor as an auto-transformer to "step down" the input voltage. Once the voltage signal is transferred across the transformer the energy in the inductor is slowly built up. Once the switch opens the input signal is interrupted and the energy is transferred to the secondary side of the converter. The diode on the secondary side acts as a rectifier and the capacitor smooths the rectified voltage, allowing for the initial AC voltage to become DC. The resistor in the diagram acts as the output load. Generally, a feedback loop is used to monitor the output voltage to make sure it stays within an acceptable range.

The standard DC/DC converter that will be used is based on the buck-boost converter topology. A model of a buck-boost converter is shown below:
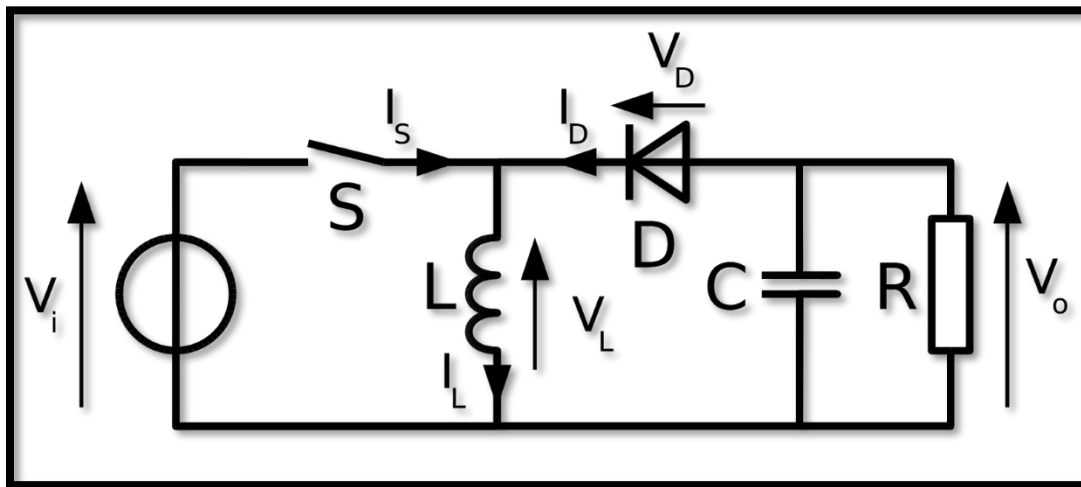
Figure 5: Buck-Boost Converter schematic

There are two operating states with this converter. The first takes place when the switch is closed. While closed the input voltage is allowed to supply the current to the inductor, filling it with energy, during this stage the capacitor supplies current to the output load as well. When the switch is closed, the inductor releases the energy to the output through the diode. In practice, the output current starts low then gradually increases until it reaches a steady state value.

Design Matrix and Analysis

The power supply selection is highly dependent on which processor is ultimately chosen. If the raspberry pi is chosen, the included power supply is what will be used. The Multi output wall adapter has a wide selection of different output voltages and can be used with the FPGA and MCU. The 3.3V wall adapter will be used if the DSP or MCU is chosen to be the processor. The total is the same for all three power supplies as they all have very similar implementations. They all are wall outlet adapters that will plug directly into the processor unit. The Multi output PS has a micro USB outlet. Due to several of the processors use of USB-A input, no matter which PS is chosen (apart from the raspberry pi PS) a micro USB to USB-A adapter will be needed.

Software (Hannah)

Technical Standards and Analysis

The software implemented by this device should:

6.1	Work with both the base system and the UI system.
6.2	Incorporate receiving user inputs, communications, and computational mathematics.
6.3	Include low power modes while the system waits for input.
6.4	Utilize multiple timers to control the indicators and the sequence.
6.5	Detect when a sensor registers a hit and continue the sequence.

Table 11: Technical Standards for Software

Considerations for Components

The code for the entire project would be split up into two groups, the base and the UI system. For the UI system, the code will need to incorporate user inputs, communication, and simple math. It will also need to be able output the results to a downloadable file (USB connection) or to a LED screen. The UI code will require reading inputs from the buttons, and sending the proper data to the base processor. Next, the code will require to wait in a low power mode until it has received data back from the base, and process the data to provide results such as, number targets, number of hits, hit ratio, selected mode and total time. Lastly, the UI code will send the processed data to either be downloaded by the USB port, or printed to an LED screen.

For the base systems code, it will need to incorporate ways to handle input from the sensors, communications to the UI system, and timers. The base code will need to use a low power mode, until it has received data from the UI system, which holds the mode of the run, followed by the start signal. The code will then process which LED lights go on, and use a timer to keep the LED lit for a certain period of time. Next the code will need to read the sensors that correspond to the segment of where the LED target is. Next, the code will need to determine, from the sensor's data, if the target was hit during the allotted time, and change a register to hold the number total hits, as well as total of lit up targets. Lastly, the code will need to send the data to the UI system receiver, and return to low power mode.

As shown by the set of software standards, the programming language selection is highly dependent on which processor is ultimately chosen. The language in the FPGA is Verilog. The IDE for the MCU can compile embedded C. DSP programming uses C/C++. The language used for a Raspberry Pi is Java/Python/C/C++/C#.

Device Design Concept

This design is largely based on the experience that Natesha has had throughout years of martial arts training and teaching to students of all ages and experience levels. In order to build the necessary skills to be a well-rounded fighter, a martial artist must be able to execute their moves well, and be able to apply it in practice during live sparring matches. This concept is reflected in the dual-side design of the bag: Side A aids application in practice while Side B is geared toward refinement of the move being used.
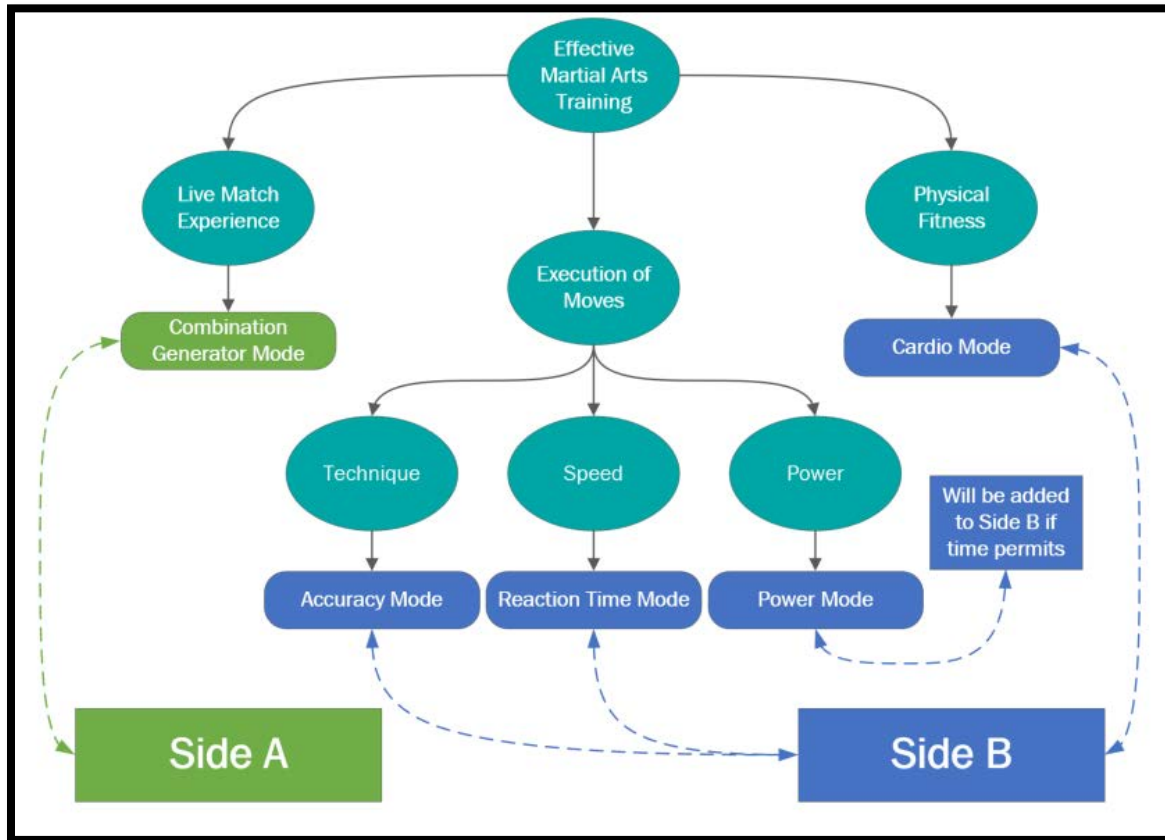
Figure 6: Association map of device active modes

Functional Modes of Device's Side B

In order to refine a move, such as a punch, a martial artist must first refine the technique (including how the move is executed and where the move is supposed to land on an opponent), then work on speed or timing. Power mode will be added to the device only if there is time to work on it without compromising the other determined aspects of the project. The hardware is available without adding extra budget, but it will be somewhat of a time drain, taking time that could be used to further improve existing features. This method builds on a move from the base up, so that more essential parts of execution in technique are not lost when aspects such as power and speed are added. Side B was designed with these principles in mind. In the center of this side of the bag, there is a target that will be fitted with sensors and indicators to implement training modes inspired by the process of refining a move. Figure _A_ shows the original concept of the target on Side B. The physical target's shape will be largely dependent on the orientation and size of the sensors, and the ability to shape the indicators around the target to sufficiently prompt the user to attack. Addressable LED strips are less conducive to round shapes, so a shape with relatively straight lines, such as a parallelogram, square, or rectangle, would likely be the most suitable for

these indicators. Since this zone will include modes where the aim is to hit as close to the center of a target as possible, the best option will likely be a square, where all 4 sides are at an equal radius from the center, even if all points on each side are equidistant from the center.

Accuracy Mode

One mode will test the accuracy of the move, so that the user can work on the technique aspect of landing a move where they intend to on an opponent. Using the higher performance mode of the IMU sensor, the device will be able to detect the location of a technique that lands on a target marked on the outside of the bag. Using the output registers from the high performance mode, a numerical score will be calculated for each impact. After a specific amount of time, the session ends and the numerical scores will be averaged and displayed to the user so they can save this value and may use it to track their progress as they train.

Reaction Time Mode

Another mode will test the user's reaction time using the sensors and indicators to allow the user to refine the speed of an attack on their opponent while reacting to a live stimulus as they would be in a real sparring match. The target's indicators will light as a timer is set to test the user's reaction time executing a technique. Once the target has sensed a hit, the timer will stop and register the reaction time of the user's technique. In one session, the user will have several reaction times stored that will be averaged and displayed to them. The user can save this value and may use it to track their progress as they continue their training over time.

Functional Mode of Device's Side A

While Side B focuses on refinement of the user's move execution, it lacks in getting the user in the mindset of using their skills directly against an opponent. Side A was designed to supplement Side B's refinement with the ability to target these techniques at practical locations as they align with the physical figure of an opponent. Key locations that a martial artist aims for on an opponent are highlighted by the indicators, and fitted with sensors that will tell whether or not the zone was hit during an allotted time. As with Side B, the shape of the zones will be dependent on the equipment needed for their proper function. The only requirement of the sensors on this side will be to detect when a hit has been successfully landed; therefore only a binary touch function of a sensor will be required, as long as the sensor is able to detect contact at any location within the zone. Indicators in Side A have the same constraints on the design of the zones as they did on Side B: zones made of straight lines, such as parallelograms, rectangles, and squares will be optimal.

Combination Generator Mode

This setup on Side A will be used for a "combinations" mode which will prompt the user to hit them while the indicator is on so as to simulate target locations that open for contact on an opponent. Each successful hit will gain the user one point towards their total. After the zone is hit or timer runs out, the indicator will turn off and the next zone will soon be illuminated. Although it will operate similarly to the reaction time mode on Side B, more zones for contact provide an opportunity to expand on the skills used and further allow for growth on the skills practiced on a stationary target. The zones will light up in a random pattern, simulating inconsistently opening and closing target options while in a live match with an opponent. After a specific amount of time, the session ends and the user's score will be recorded. The user will have the option to save the score so that they can track their progress over time.
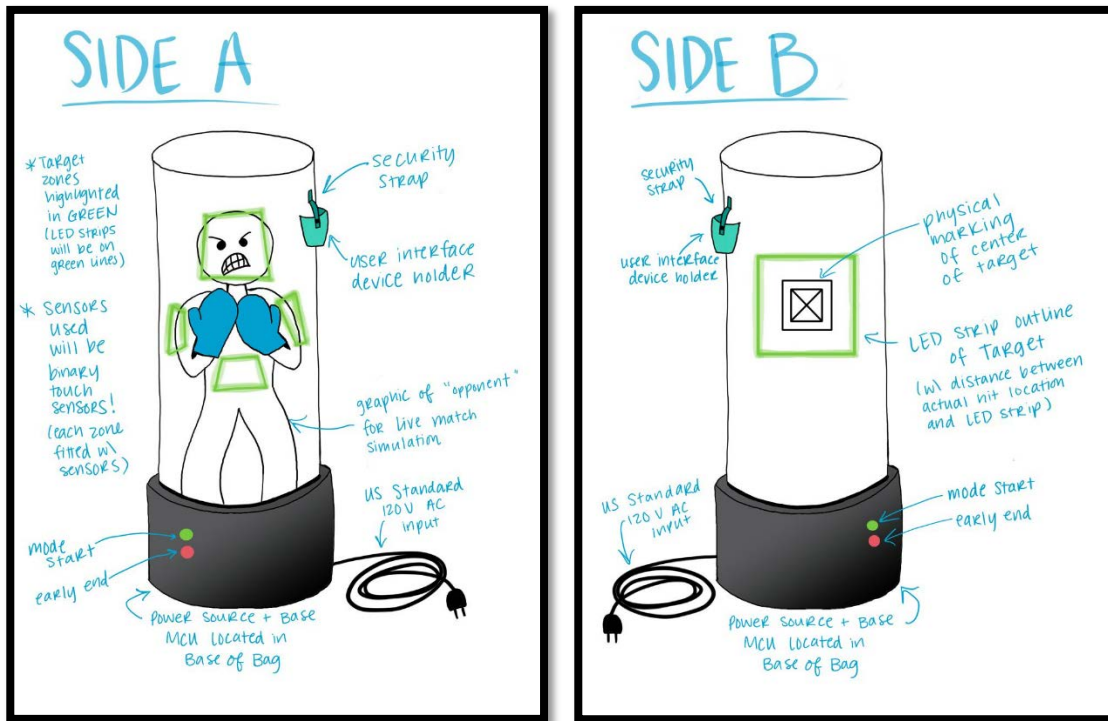

Figure 7 (Left): "Side A" of the device, Annotated
Figure 8 (Right): "Side B" of the device, Annotated

General Performance

As mentioned in the descriptions, all modes will gather data to "score" the user on their performance quantitatively. The user will have the option to save this data and add it to the collection of their performance data. All data calculations will be performed on the base unit, and the UI system will display the mode specific results on the screen. In the event that the USB port is implemented, the user will have the option of downloading their session results to the flash drive. Otherwise, the results are cleared, and the user can select a new mode.

In addition to the active modes, the device should also employ passive modes for the time in between the active modes. One such mode will be used when the user is starting up the device and selecting a mode to use. The UI system will enter a low power mode, while the user is actively training with the base, and will exit the low power mode as once the base transmits the results of the session to the UI.

Alternatively, the base system will implement its low power modes while the user is using the remote to select the mode and difficulty. Once the base receives the mode and difficulty setting, it will exit the low power mode. The base returns to the low power mode once the results have been sent back to the UI.

Display/Ambient Mode

The option to add a display mode is also open, and would add a fun, homely touch to this device intended for home workouts. Within the typical home, a large standalone punching bag may be considered an eyesore in all places except the garage or home gym area. To make this device more attractive to consumers, we can program the lights on the device to create patterns with the LED lights that are fun for the user to look at or have running while the bag is not in use. This option is not necessary, but it may help the marketability of the product to consumers who may be on the fence regarding whether or not such an item belongs in their place of living.

Base System Hardware Design

This device will have two components that communicate with each other to make the user experience interactive and immersive. There will be a main device/Base System and a User Interface device. The base system will be based on the standing bag, fitted with indicators and sensors that will gather data on the user's performance. The UI device will be smaller and handheld with a space for safekeeping while the bag is in use. This will be where the user can give input (such as mode of exercise) and receive outputs (such as their exercise statistics).
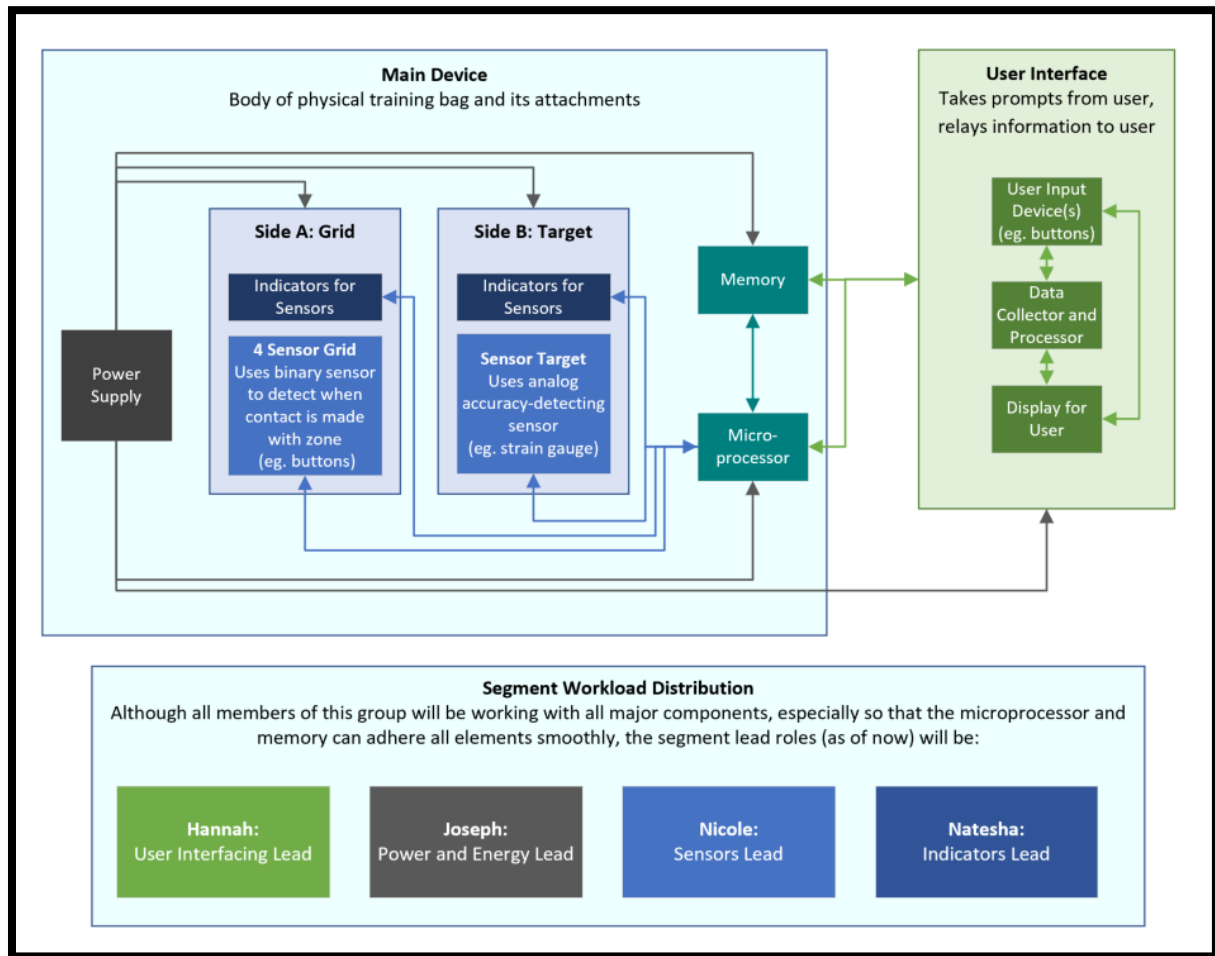
Figure 9: Hardware connection block diagram; description of current distribution of group roles

Device Body Build and Design Plan

The body of this device will be a used Wavemaster branded bag by Century Martial Arts. These bags come in various sizes, but the most likely candidates for use in this project are the far left and far right positioned bags in the image below.

Figure 10: Century Martial Arts Wavemaster options for body base [24]

On this bag, there will be LED indicators designating the zones to hit as well as physical markings to guide the user. Side A's grid of sensors will be laid out to simulate hitting those areas on a physical opponent. Side B has a target showing the center that the user will be aiming for when using modes on this side. For each of these zones, it is required of the markers used to endure large amounts of physical contact without much deterioration of the design. These markings can be seen on the bag designs below.

Figure 11 (Left): "Side A" of the device
Figure 12 (Right): "Side B" of the device

To create this, some mediums we could use include permanent marker, paint, tape, and vinyl decals. Permanent marker is cost effective, easy to use, and takes little dry time. Similarly, paint is inexpensive, has an intense pigment even over a black background, and simple to implement. However, extended periods o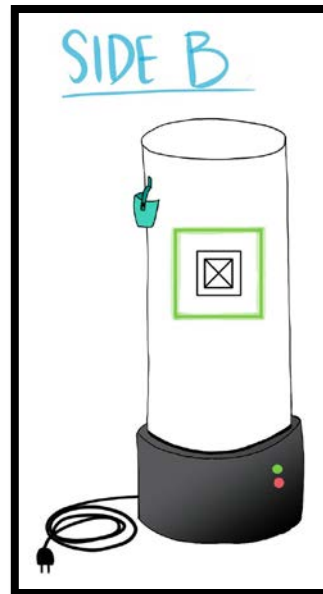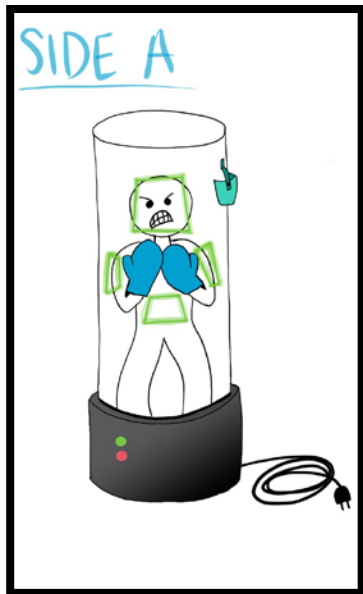f physical contact will cause the markings to erode and possibly stain the hands of the user. Tape (particularly electrical tape) can cover large zones and make many different shapes. However, the edges may start to raise after lots of contact, and bumps in the tape where the pattern curves may be uncomfortable to hit. Vinyl decal stickers may stand up to prolonged contact the best out of these options, but do not easily cover large areas. This would be ideal to use for small detail, especially in areas that will be frequently touched.

Sensor Design

Based on the technical research presented in the sensors section of this report, the design of this device will make use of binary sensors as well as either IMUs or accelerometers.

Side B Primary Sensor Choice: IMU

The following is a comparison of IMUs we found fitting our requirements: the FIS1000, the LSM6DSRTR, and the MPU-9250. We want an IMU which supports SPI, with at least a 6-axis sensor, and a high resolution. Minimizing cost and shipping time is desirable as well.

Out of these three, the best ones are the LSM6DSRTR and MPU-9250. The major distinguishing characteristic is mounting. The LSM6DSRTR will require careful soldering as a surface mount unit without pins, whereas MPU-9250 already comes mounted with pins for us to use.

	FIS1100	LSM6DSRTR	MPU-9250
Sports Application?	Yes	Yes	Yes
Obsolete?	Yes	No	Yes
LGA Package	3.3x3.3x1mm	2.5x3x0.86mm	3x3x1mm
Current	2.75mA	1.2 mA	3.4mA
Voltage(Min/Mx)	2.4V/3V	1.71V/3.6V	2.4V/3.6V
Output	Digital	Digital	Digital
Interface	I2C/SPI	I2C/SPI & MIPI I3C	I2C or SPI
Sensor type	6-Axis	6-Axis	9-Axis
Temp(Min/Max)	-40C/85C	-45C/85C	-45C/85C
Resolution	16 bit	16 bit	16 bit
Lowest Price per Unit	$4.02 (Digikey)	$4.77 (Digikey)	$8.99 (Amazon)
Stock at Above Source	3000	293	Undisclosed
Time to ship	5 days	Immediately	2 days

Table 12: IMU Candidate Comparison

Side B Backup Sensor Choice: Accelerometer

The following is a comparison of accelerometers we found fitting our requirements. We want an accelerometer which supports SPI, with at least a 3-axis sensor, and a high resolution. Minimizing cost and shipping time is desirable as well.

All of these accelerometers are sufficient backup options, but have the same limitations as the surface mount LSM6DSRTR IMU that requires precise and careful soldering on a small scale.

	MC3672	MC3630	BMA490L
LGA Package	1.29x1.09x0.74 mm 8-ball	2.0x2.0x0.94 mm 12-pin	2.0x2.0x0.95 mm 12-pin
Current	-10uA/10uA	-10uA/10uA	13.5uA/150uA
Voltage(Min/Mx)	-0.3V/3.6V	-0.3V/3.6V	1.62V / 3.6V
Output	Digital	digital	Digital
Interface	I2C (1MHz) SPI (4MHz)	I2C (1MHz) SPI (4MHz)	SPI,I2C
Sensor type	3-Axis	3-Axis	3-Axis
Temp(Min/Max)	-40C/85C	-40C/85C	-40C/85C
Resolution	8,10 or 12 bits with FIFO	8,10 or 12 bits with FIFO	16 bits
Price (Mouser)	$3.30	$1.21	$3.80
Stock (Mouser)	321	7448	5333
Time to ship	Immediately	Immediately	Immediately

Table 13: Accelerometer Candidate Comparison

Side A: Binary Sensor

Binary sensors can be made with a similar method as a mechanical switch. To make a binary sensor, we can put two pieces of metal attached by their backs to a piece of wire. One side will be connected to ground and the other one to power. There will be a third metal piece that is connected to the other end of a wire. In between the pieces of metal, there will be a small square foam space in the middle to help them avoid connecting). This method is based on the design of the buttons on a Dance Dance Revolution dance pad. We will be looking for either pieces of conductive metal to be able to cut to shape the switches needed, or for some sort of soft conductive material that will let current flow when contact is made. These materials can be found at stores such as Lowe's, Home Depot, or any local electronics store.

Indicator Design

The indicator that was decided upon after completing technical research is an addressable LED strip.

Choosing an LED Strip

When deciding on an LED light strip for the indicators in this device, the most important requirements are addressable LED cells, RGB LED capability, and the ability to cut and connect these strips in any way that we must manipulate (including short lengths). The strip should be durable, so descriptors such as waterproof and weatherproof are positives as well. We will likely need at least 12 feet of the LED strip to split between the five zones on the device, but we must be mindful of the cost so that it suits our budget.

In the product comparison research completed so far, the most viable option for use in this device is we have found for our needs so far is an ALITOVE 5 m (16.4 ft) RGB WS2811Addressable LED Strip reel from Amazon. This product provides more than enough length for our needs. Its LEDs are organized in groups of 3, with each group able to be fully individually controlled. The strip can be cut as needed and resoldered together to make smaller sections of LED lighting, as our project requires. The price for this is $26.99 (about $1.65/Ft) which is an ideal cost compared to many other strips we have seen, some of which have exceeded $90. This strip requires a 12 V input and a ground connection. Information is sent to each of the addressable LED cell clusters on the data line of the strip. It should be able to work with the MSP430 that has been selected for this device.

Figure 13: WS2811 LED Strip with wire ribbon configuration [2]

Attaching the LED Strip

Most LED strips come with an adhesive backing, but in experience working with these strips, this adhesive is not always as strong as the companies claim it to be. For this reason, we will likely need to find a different way to physically fit the indicators onto the bag, and make sure that they are secure enough that they will not move when the bag is being repeatedly hit with high speed and/or force.

Processor Design

TI MSP430F5528

This processor was chosen because it meets all of the technical requirements that were outlined in the research section. This processor, of the original contenders, has the best price to features ratio and meets the technical requirements without going overboard on features. The relevant specifications for this component are listed in the table below. The pinout diagram for this microcontroller is also pictured in the appendix of this document.

RAM	8KB		SPI Interfaces	4
ROM	128KB		UART Interfaces	2
Clock Speed	25MHz		I2C Interfaces	2
GPIO pins	47		Timers	16
Input Power	1.8V - 3.6V		Extended Memory	Available

Table 14: MSP430F5528 Specifications

The processor will be used to collect data from the sensors and to use that data to calculate results according to the data and power mode. The sensors will communicate with the processor using UART, SPI, and I2C. Most of these interfaces will be used to connect to all the sensors on the punching bag. To power the MCU, the DC/DC converter will be used to convert the 12V DC signal to a 3.3V DC signal. This signal will be adequate to power the MCU.

Power Supply Design

The power supply will originate from the AC outlet and will be converted to a 12V DC signal with an AC/DC converter. This 12V DC signal will be used to power the LEDs directly. This is necessary as the LEDs require a 12V signal to power on. The sensors will be powered using the output power pin from the MCU. This will allow us to both power and read from the sensors using the same device. This will simplify the power design and allow us to use less wires. The MCU will be powered using both the AC/DC converter and the DC/DC converter. First the AC/DC converter is used to convert to a DC input, then the DC/DC converter is used to convert the 12V DC signal to a 3.3V DC signal suitable for the MCU.
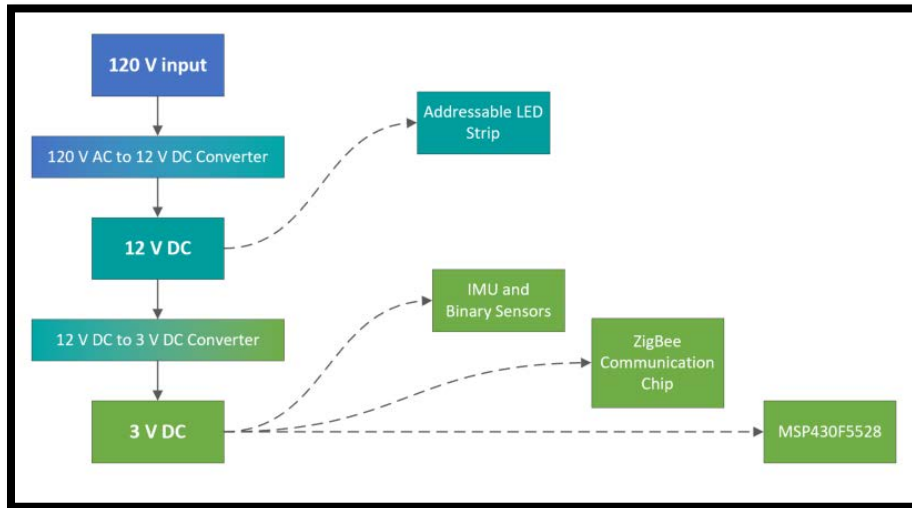

Figure 15: Flow chart for Power Supply in Base System

Printed Circuit Board Design

Since there are two main components of this device that each require their own printed circuit boards (PCB). These PCBs will each hold one microcontroller, as well as sufficient connection pins for all power, sensor, and indicator inputs and outputs. The base system will house one of the PCBs, as this is the location on the bag least likely to get damaged by a hit. The UI system handheld component will house the second PCB.

UI Hardware Design

The UI system that will be used to control the punching bag will be controlled by a handheld, wireless remote. Since the UI component of this overall device is essentially its own individual device with independent hardware requirements from the base, the details of its hardware design will be discussed in this section. The remote will have all of the component sections detailed in the Base hardware section. Each part should be evaluated and picked to best meet the requirement specifications. A sketch of the first draft of the UI Remote can be seen below.
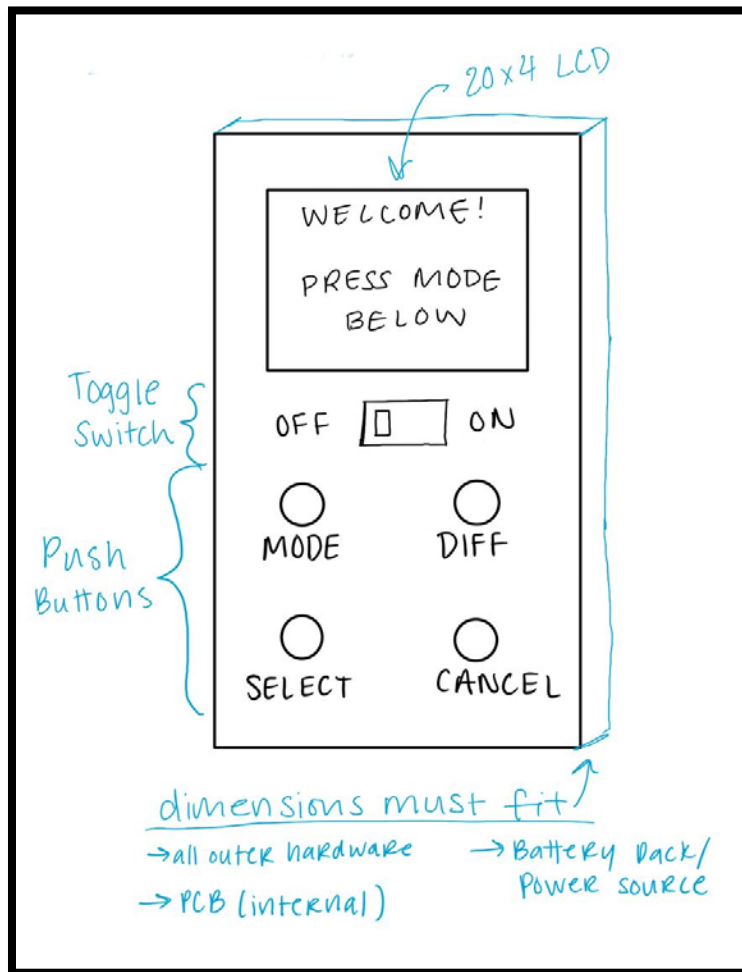

Figure 16: Draft sketch of the UI system for the device

Processor (PCB1)

To control the remote, a processor is required to perform the main calculations and operations based off the inputs for the UI hardware system. The processor needs standard GPIO pins to connect I/O to the MCU. The processor should also be capable of I2C communication. The I2C communication is required to talk to the LCD display on the remote. With these specifications a processor can be chosen to put in the remote. Due to the low expense and the meeting of the technical specifications, the same processor (MSP430F5528) can be used for the machine and the wireless remote.

Display

The LCD display is a 20 column by 4 row screen. The MSP chip communicates with the display controller (driver) to configure the input and output pins. The LCD driver uses I2C and requires two GPIO ports.

Buttons & Switches

Push buttons will be used to control different functions for the remote. The push buttons require GPIO pins to connect to the MCU. One pin is needed for each push button, for a total of four GPIO pins necessary for the four buttons. To read a high value from the push buttons, the push buttons should be pushed in. To set this up in the correct way, the resistors for these configurations should be set to pull down mode.

Another input method needed for the remote is a sliding switch. This switch will act as the power on/off for the remote. To connect the switch to the MCU one GPIO pin is required. The resistor can be set to pull down or pull up as each would have the same functionality on a switch.

Power Supply

The processor requires an input power from 1.8V to 3.6V. The LCD display requires 5V for full functionality and the Zigbee chip is 1.8V to 3.8V. This means that a 5V DC signal is the lowest possible voltage required to power on all of the electronics inside the remote. To generate this amount of voltage using wireless methods four AA batteries will be included in the remote. The four batteries together will supply 6V, which will be enough voltage to power all the essential electronics inside of the remote.

Protective Case

The Case can be manufactured in multiple different ways. The most precise way to manufacture the case would be to 3D print the outside material. This method would give us the most flexible control when designing the case. It would allow us to plan for and include all the compartments that we need ahead of time. Another method of manufacturing the case would be to use plexiglass and glue. This would allow us to plan for and manufacture the case according to our plans, although the construction would be more involved than construction with a 3D printer. Finally one last alternative way to manufacture the case would be to use a plastic welding tool to shape plastic in the desired shape. Because of the ease of use and the increased flexibility of design, 3D printer manufacturing is the optimal way to construct the remote case.

For protection of the handheld unit while the bag is in use, a secure pocket will be attached to the bag.

Software Design

The software for the interactive workout bag is implemented using agile software development. This method emphasizes the use of adaptive design and development. The technologies integrated in our designs thus far are not set in stone, per say. During development, the technologies, and design choices may change, due to our team taking advantage of adapting our designs and development as new information is uncovered while the product is being built.

With this method of design, it allows us to implement specific components, test their performance, and make adjustments to each component along the way, it also allows our design to be flexible, and to take other design restrictions into account.

Design Environment / Tools

Design Method

The software for the interactive workout bag is implemented using agile software development. This method emphasizes the use of adaptive design and development. The technologies integrated in our designs thus far are not set in stone, per say. During development, the technologies, and design choices may change, due to our team taking advantage of adapting our designs and development as new information is uncovered while the product is being built.

With this method of design, it allows us to implement specific components, test their performance, and make adjustments along the way, it also allows our design to be flexible, and to take other design restrictions into account.

Design Environment / Tools

The software design environment depends greatly on the processor that is used. Because our design implements the use of MSP microcontrollers, Code Composer Studio (CCS) will be used as the IDE to write, push, and debug the program onto the microcontrollers. This choice is beneficial for the team because everyone has experience with programming microcontrollers to perform various tasks, which include the low power modes, interruptions, display drivers, and more. CCS uses embedded C as the program language, which all of our team members have experience in as well. CCS also already comes with the header files and interrupt vectors for the MSP chips, and multiple useful libraries used for complex math and communications, which makes it easier to use as well.

General Overview

The software for the workout bag will be split into two different areas: UI system and Base system. The UI system is a separate device from the base and handles the interface between the user and the system. The base system is the electronic components tasked with reading sensors, providing light sequence patterns, and communicating with the UI system (also called the remote).

In general, the user turns on the device with the UI system. In the UI system, the user selects the mode and difficulty on the remote. The UI system then transmits the users' selections to the base. Once the base receives the mode and difficulty, the workout bag's targets blink to indicate to the user that they can press the start button. The base system then executes the desired mode and difficulty by enabling various timers, sensors and target area LEDs. The base system will then complete the sequence, and compute the results related to the mode, and send them back to the UI system for the user to see. The UI system then returns to the beginning, allowing the user to select another mode and difficulty or to shut down.

UI System

The first action to happen, is when the user turns on the system by using the on/off switch. The system then turns on the screen, and displays its welcome message, and prompts the user for their selection. There are two branches based on the user selection, system off or mode. If the OFF button is pushed, the system calls an interrupt to terminate the program and shut down. If the Mode button is pushed, the user is able to select a mode. Once the Select button is pressed, the user then can choose a difficulty setting, based on the mode. The user then presses select a second time. The mode type and difficulty type are sent to the base system MCU and the UI system enters a low power mode, while the rest of the sequence happens at the base. In order for the user to be in place when the session begins, the workout bag's target areas will blink to indicate to the user that they can begin the session by pressing the start button on the base.

Low Power Modes

The UI system will not be used the majority of the time, since the user's main interaction is with the workout bag. Since the UI system is battery powered, the low power modes on the MCU will be used while the user is completing a session. Once the UI system transmits the necessary data to the base system, the UI system will enter a low power mode to conserve power.

As the UI system receives data, the MCU is interrupted, and begins processing the display sequence for the data that is given based on the mode selected.

LCD Display

Welcome Screen

The LCD display will have several different types of messages printed to the screen. The first screen displayed is the Welcome Screen. On the first row, the display shows "<name of product>". On the third row, the display shows "Press Mode". Because the buttons do not have individual LEDs, the display will show the mode and difficulty selection on the screen, as the user presses the buttons to cycle through the options.

User Input

The user inputs their workout selection by using the buttons under the display. Each button is labeled with their function. The functions for the buttons are Mode, Difficulty, Select and Cancel. The buttons used for the user inputs are susceptible to the bouncing effect, where the user may press the button only once, but the MCU receives several phantom pushes. To correct this, a de-bouncing method is used, that incorporates a small delay that disables the button to avoid the processor receiving multiple button pushes.
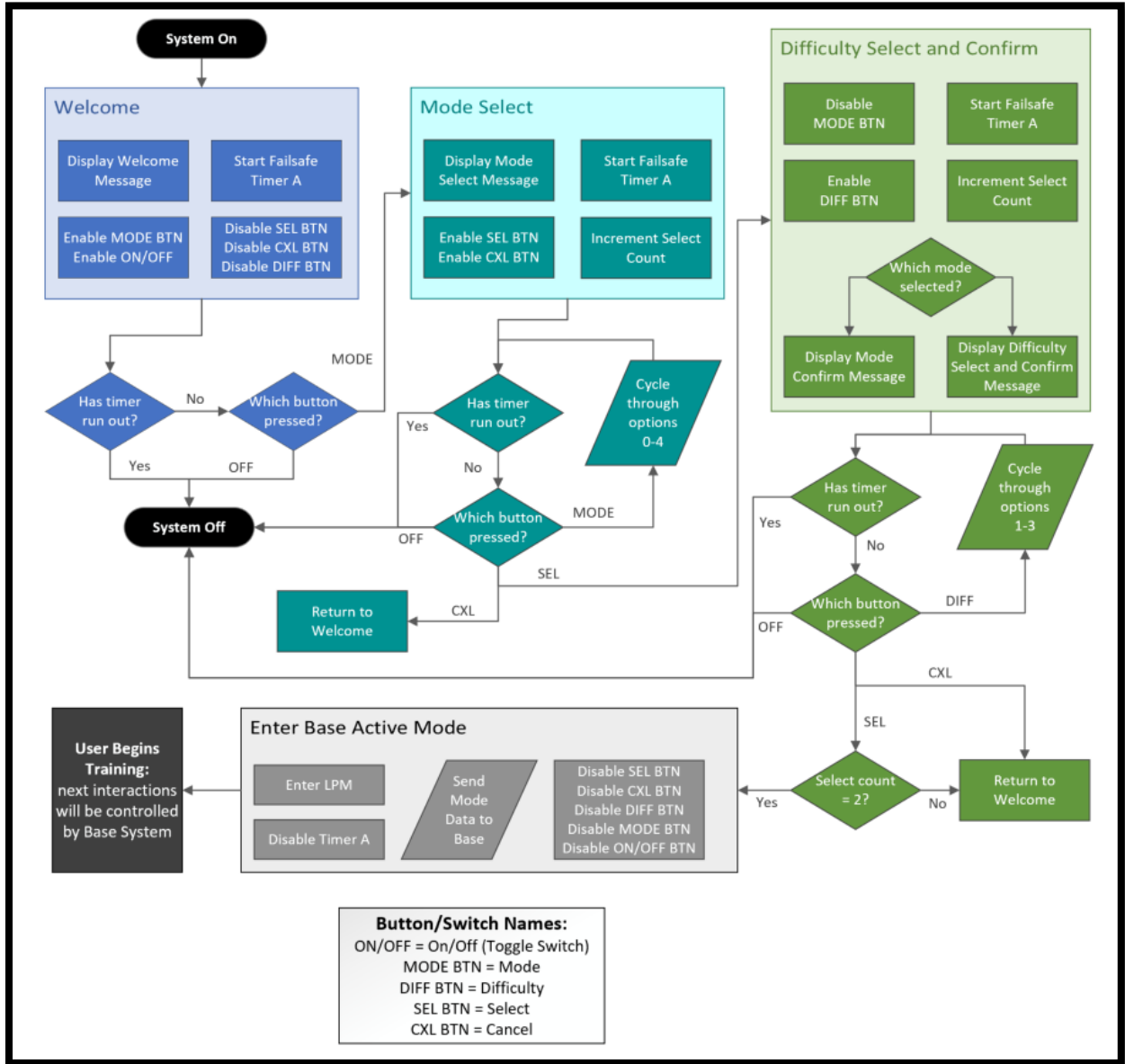
Figure 17: Software flow chart for UI and Initialization

MODE (MODE BTN)

The user is to press the MODE button to cycle through the available modes (5). On the display in the first row, the first six columns display "MODE: ". Columns 7 through 20 display the mode options, where each press of the mode button, changes the mode option that is displayed. Internally, the first time the mode button is pressed, the mode register is set to 0. Each time the button is pressed, it increments the register, until it holds number 4 (last mode), and then returns to 0 and keeps going. To select a mode and continue, the user presses the select button. The mode option and the register values are shown in Table X, below.

Reg Val	Mode Option
0	Combination Generator Mode
1	Reaction Time Mode
2	Accuracy Mode
3	Cardio Mode
4	Ambient Mode

Table 14: Mode options and register values

SELECT (SEL BTN)

When the user has the mode they wish to use on the screen, the select button is used to store this mode, and move on to the next function. When the select button is pushed 1 time, the mode button is disabled and the difficulty button is enabled based on the mode type. Then on the display, the second row reads "DIFFIC: ". The second time the select button is pressed, the mode and difficulty buttons are disabled and the UI system sends the mode and difficulty setting to the base.

DIFFICULTY (DIFF BTN)

Based on the mode selection, different options of difficulty will be available. The options in total are: Easy, Medium and Hard. Modes 0,1,2,and 3 are able to have different difficulty settings (Easy, Med, Hard), whereas mode 4 does not have a difficulty setting. Only in modes 0, 1, 2 and 3 can cycle through the modes, meaning the difficulty button is enabled for those modes only.

When mode 0 is selected, the difficulty setting is adjusting the amount of time the allotted time to hit a target area, where the allotted time gets shorter as the difficulty setting is increased. When mode 1 is selected, the delay between targets is randomized with higher variances. When mode 2 is selected, the difficulty setting is adjusting the threshold distance for accurate hits. As the difficulty setting increases, the threshold distance reduces, effectively making it harder to complete accurate hits. When mode 3 is selected, the difficulty setting is controlling how long the session lasts. As the difficulty setting increases, the length of the session increases.

When mode is set to 4, the difficulty setting is not applicable. In mode 4, the workout bag is in the ambient mode and is not used for training or measuring the user's ability.

Reg Val	Difficulty Option
0	Easy
1	Medium
2	Hard
3	X

Table 15: Difficulty options and register values

CANCEL (CXL BTN)

The cancel button is used to start over the selections. When pressed, it resets the mode, difficulty, and select registers, and returns to the welcome screen. This can be pressed at any time until the user presses the select button for the second time. To cancel a workout after the select button has been pressed twice, the user will need to press the exit early button on the base.

The cancel button is also used to return to the welcome screen after results have been displayed to the user. When pressed after the results are displayed, the function remains the same, which returns to the welcome screen to make selections.

UI Failsafe Timer 0

The UI system is highly dependent on the user performing an action (making selections and pressing buttons), which leaves the UI system to be potentially stuck in an infinite loop of waiting for input response. In the event that the user walks away from the UI system, the remote requires a way to shut off to avoid remaining on and wasting battery. To do this, a timer is used that resets every time an active button is pressed. If the timer is able to reach its end, an ISR is called to shut down the UI system.

Results Transmission

After the base system has transmitted the user's results to the UI system, the UI system is interrupted to exit low power mode. The display is turned on, and the result categories and their values are displayed to the user. The results are displayed until the user presses cancel or until the failsafe timer goes off. The display then returns to the welcome screen, allowing the user to make another selection, or shuts off, depending on if the user presses mode, uses the OFF button or the failsafe timer goes off.

Because the LCD display has 4 lines that can hold up to 20 characters, the results category names and values must be able to be displayed within those limits. This means if the category abbreviations are within 5 characters and the results values are 3 characters (0-999), then a maximum of 6 results can be displayed (two columns of 3 rows). This is because the first row of the display is needed to display the users' selections, in the manner of: MODE / DIFFICULTY.

Communication with Base

The base and the UI system will need to communicate in order to send the user inputs to the base and send the results back to the user. To do this, the MCU in the base and the UI system will require a transceiver. The transceiver is connected to the MCU with SPI (4 wire) connections, and uses the IEEE standard 802.15.4 for creating the wireless personal area network (WPAN). To integrate this with the MSP chips, a CC2XXX transceiver chip can be used to support the frame handling, buffering, and transmission. The chip contains the zstack software developers kit to configure the communications between the two devices on the WPAN. This works with IEEE Standard 802.15.4 as shown in Figure 18.
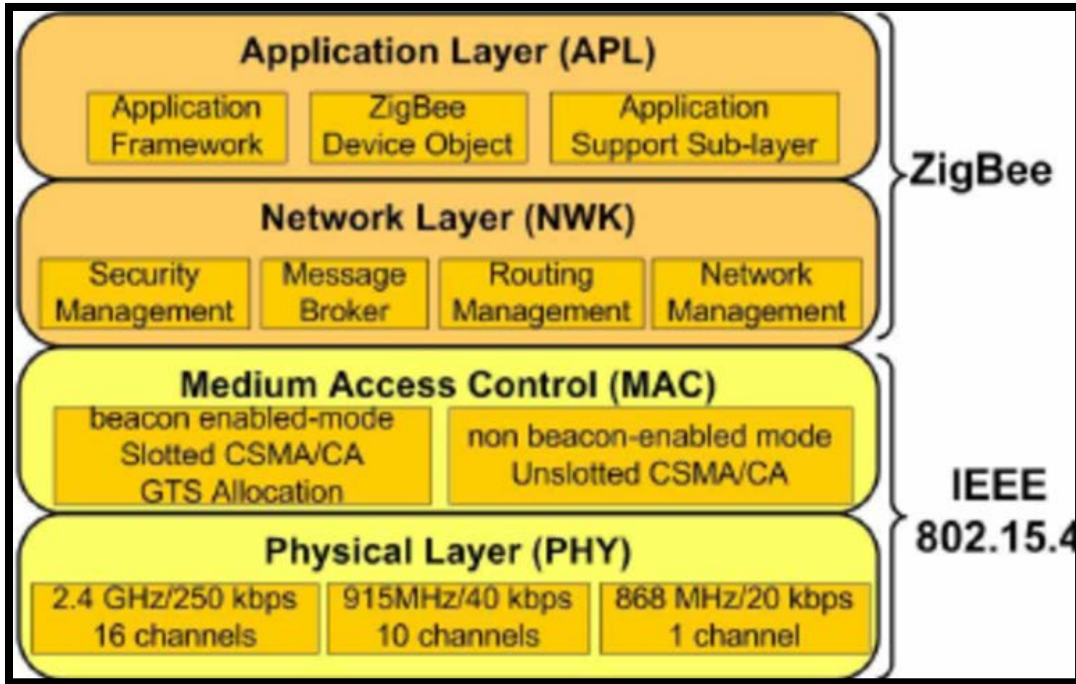

Figure 18: Relation of ZigBee to IEEE Standard 802.15.4 [17]

Our team has not had the opportunity to work with Zigbee to create wireless communications between two devices. Therefore, the program's design is not solely dependent on the use of wireless communication. In the event our team is unable to successfully transmit the required data between the two MCU's, a wired connection is able to be used instead (SPI, I2C, UART). The remote's pouch is on the side of the bag, reducing the probability of the user tripping on any exposed wires.

Base System

When the base system receives power, the MCU initializes it's devices (communication, LEDs and Sensors) and then enters low power mode. Once the base system receives the user's mode selection and difficulty selection from the UI system, the base system will enable its LEDs to cycle through a color sequence. This is to indicate to the user that the base system is ready for the user to press the Start Button. Once the user presses Start, the base MCU will then load the premade sequences based on the mode selected. The base system cannot enter low power modes here, as the processor is actively turning on/off target indicators and capturing sensor data from the target areas that are light up. Once the main sequence timer ends, the processor can begin to compile the data into the results which are provided to the user, and then sends the results to the UI system and returns to low power mode.

Low Power Mode

While the base system is not in use, the MCU will enter a low power mode, to reduce the power consumed during inactive moments. When the UI system transmits data to the base system with Zigbee, the base system's receive flag is raised and an interrupt is called to "wake" the MCU and to begin processing. When the base system has completed its tasks and has sent the results to the UI system, the MCU will return to the specific low power mode. The MCU is able to enter/leave low power modes in the event of transmission or reception through Zigbee.

Target Areas (LEDs)

The workout bag has five different modes: Combination Generator mode, Reaction Time mode, Accuracy mode, Cardio mode and Ambient mode. Each mode will have unique target areas that the base MCU will need to enable to indicate to the user where the hits should take place.
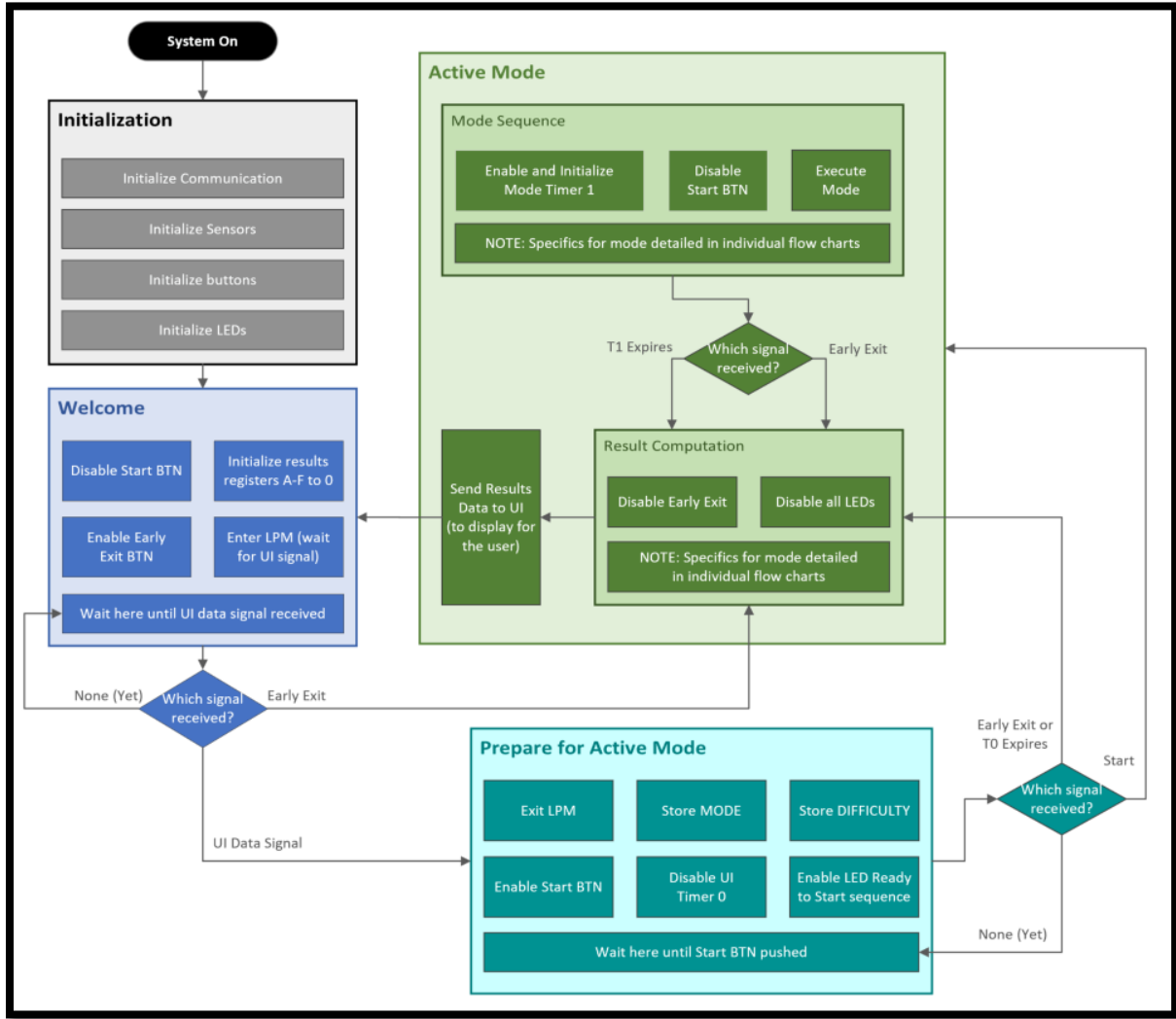
Figure 19: Software flow chart for Base Active Mode Initialization

Combination Generator Mode

The Combination Generator mode is on side A of the workout bag. The addressable LED strips outline four target areas that light up to indicate where the user should hit. Each target area is identified with a numerical value, 0 through 3. In this mode, the MCU will begin its first timer, which limits the session to a specific number of seconds.

Next, the MCU will generate a random number between 0 and 3, and turn on the corresponding target area. Simultaneously, the MCU will start its second timer that controls how long the target area is on. When the second timer ends, an interrupt is called, and the program returns to the random number generator, and the second timer is started. The length of the second timer is variable, as the user is able to select different difficulty settings, where the harder settings result in shorter secondary timers, meaning the user has less time to hit the target area.

The user shouldn't have to wait for the secondary timer to end if they have successfully hit the target area. Therefore, if the user hits the target area before the secondary timer ends, the MCU calls an interrupt to return to the random number generator to continue with the session. Lastly, once the first timer ends another interruption is called, and it ends the session by turning off all LEDs and sensors.
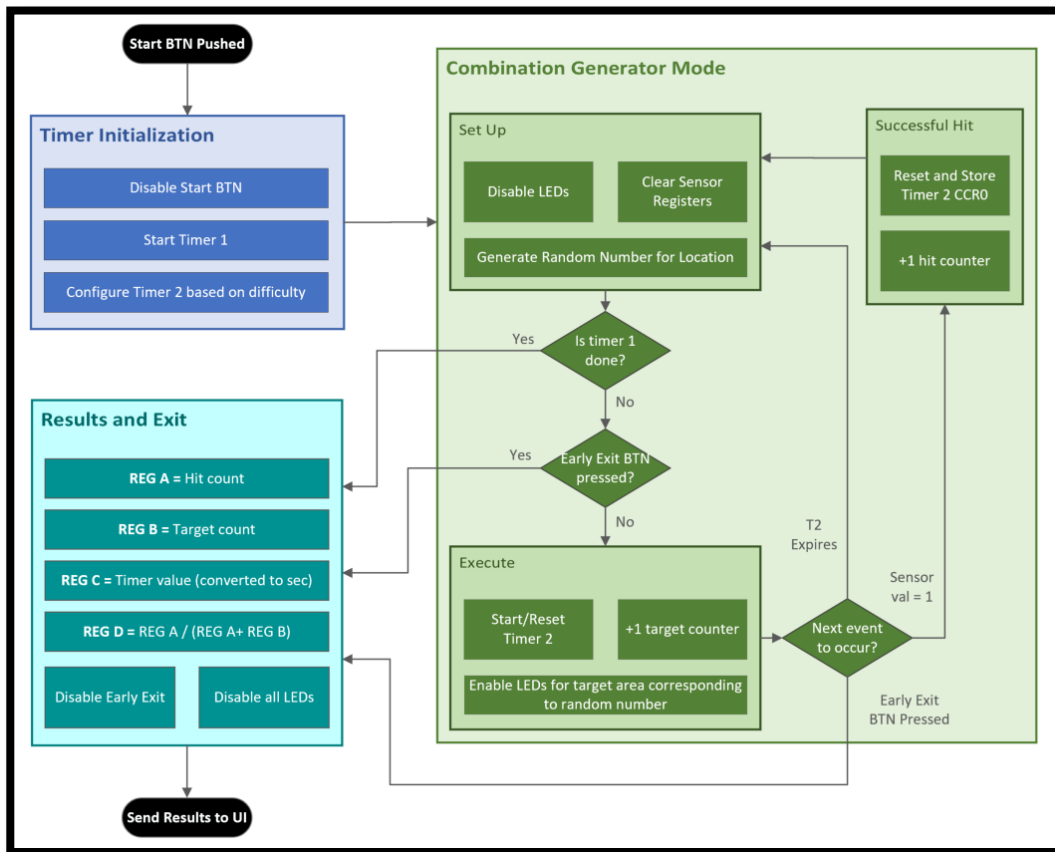

Figure 20: Software flow chart for the Combination Generator Mode on Side A

Reaction Time Mode

The Reaction Time mode is on side B of the workout bag. Side B has one target area and one IMU sensor. For the reaction mode, the MCU starts its first timer, which controls how long the session lasts. This time is set by the developers, and not adjustable. Next, a second timer begins as the target area is enabled. Once the sensor detects a hit, the second timer ends and the target area is turned off. The MCU waits in a delay loop for a short period of time and the process repeats. This delay is adjustable, with the difficulty setting. The time from when the target area is enabled (second timer starts) and when the second timer ends is the amount of time the user reacts. Each hit's reaction time is stored, and averaged for the final result, until the first timer ends.

Since this mode is calculating reaction time, the program is dependent on the user's hit to move forward with the sequence. However, the entire sequence is based on the first timer. Therefore, when the first timer expires, the program is able to move on to the results computation, without getting stuck in an infinite loop, however, this will affect the end results.
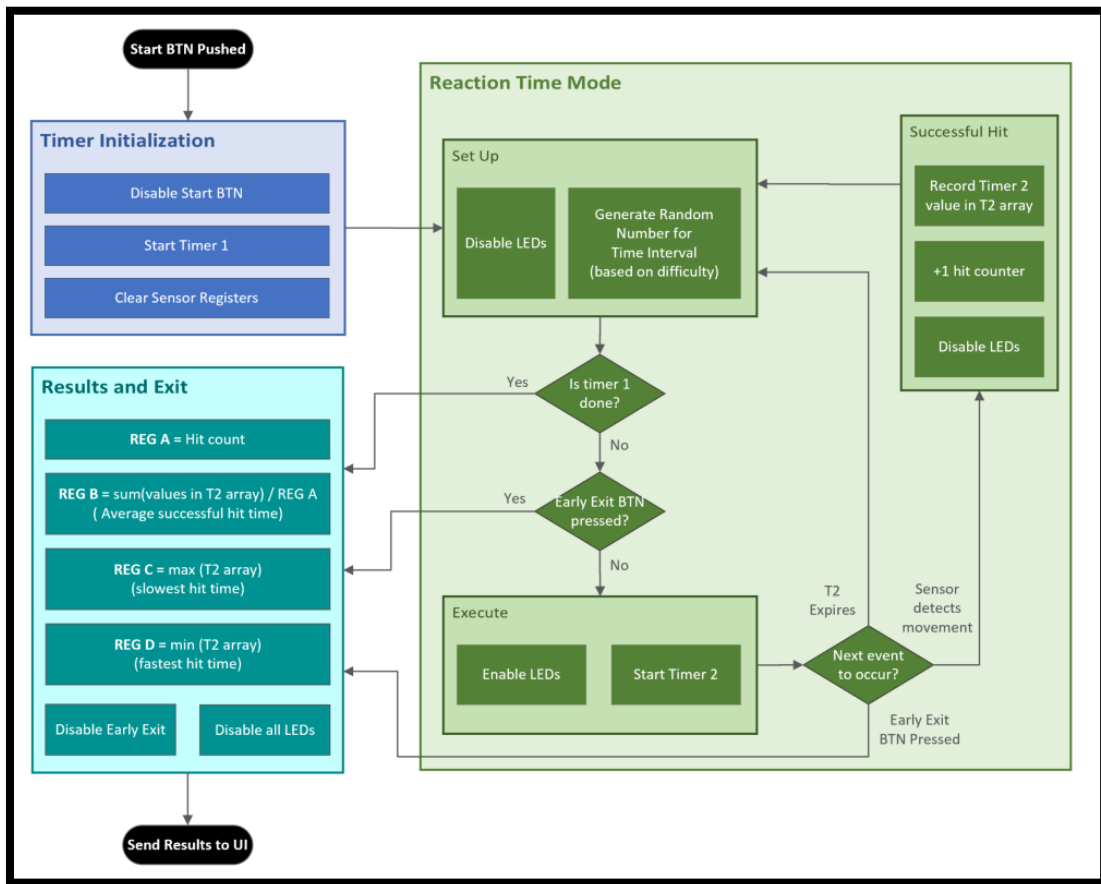

Figure 21: Software flow chart for the Reaction Time Mode on Side B

Accuracy Mode

The Accuracy mode is on side B of the workout bag. Similar to the reaction time mode, the accuracy mode has one target area that lights up. The MCU starts its first timer, which controls the length of the session. Next, the second timer starts when the target area is lit. Different to the reaction time mode, the Inertial Measurement Unit (IMU) sensor is used to locate the hit's distance from the center of the target area, which is marked on the outside of the workout bag. Once a hit is detected, the process repeats itself in the same way as the reaction time mode, however the LEDs do not turn off. The difficulty setting changes the threshold distance for an accurate hit, instead of the delay, as in the Reaction Time mode. When the first timer reaches its limit, an ISR is called to exit the session and display the results.

The MCU will have predetermined thresholds of distance to measure the level of accuracy of each hit. The user can choose the difficulty setting, whereas the difficulty settings get harder, the threshold distance becomes smaller. Hits with a distance greater than the threshold are considered a "hit", while hits with a distance smaller than the threshold are considered an "accurate hit".
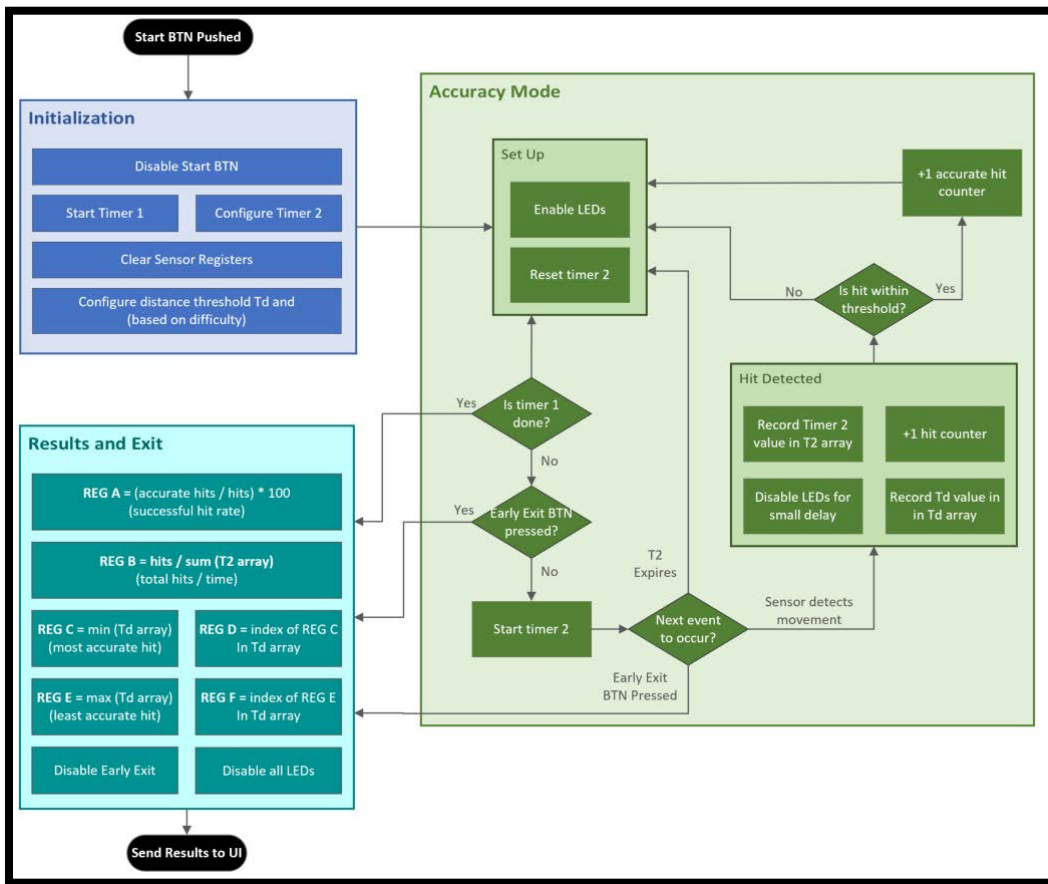

Figure 22: Software flow chart for the Accuracy Mode on Side B

Cardio Mode

The cardio mode is also on side B of the workout bag. In the cardio mode, one target area is used and remains lit for the entire session. The goal is to have as many hits as possible during the allotted time. The MCU will start its timer and the target area will be lit. Because the cardio mode is only counting the number of hits during the session, only one timer is used. Each time the user hits the target area successfully, a register holding the number of hits increments by one. Once the timer ends, the session is over, and the program moves on to calculating and displaying results.

The cardio mode is not dependent on receiving the user's hits to move forward, therefore, if the user decides to not hit the workout bag during the time frame, the score will be affected at the end. If the user would like to end the session without affecting their score at the time (by a significant amount), they will need to use the exit early button.

In the cardio mode, the length of the session is variable with the difficulty setting. Therefore, to make the workout more challenging, the length of the session is increased for each difficulty setting.
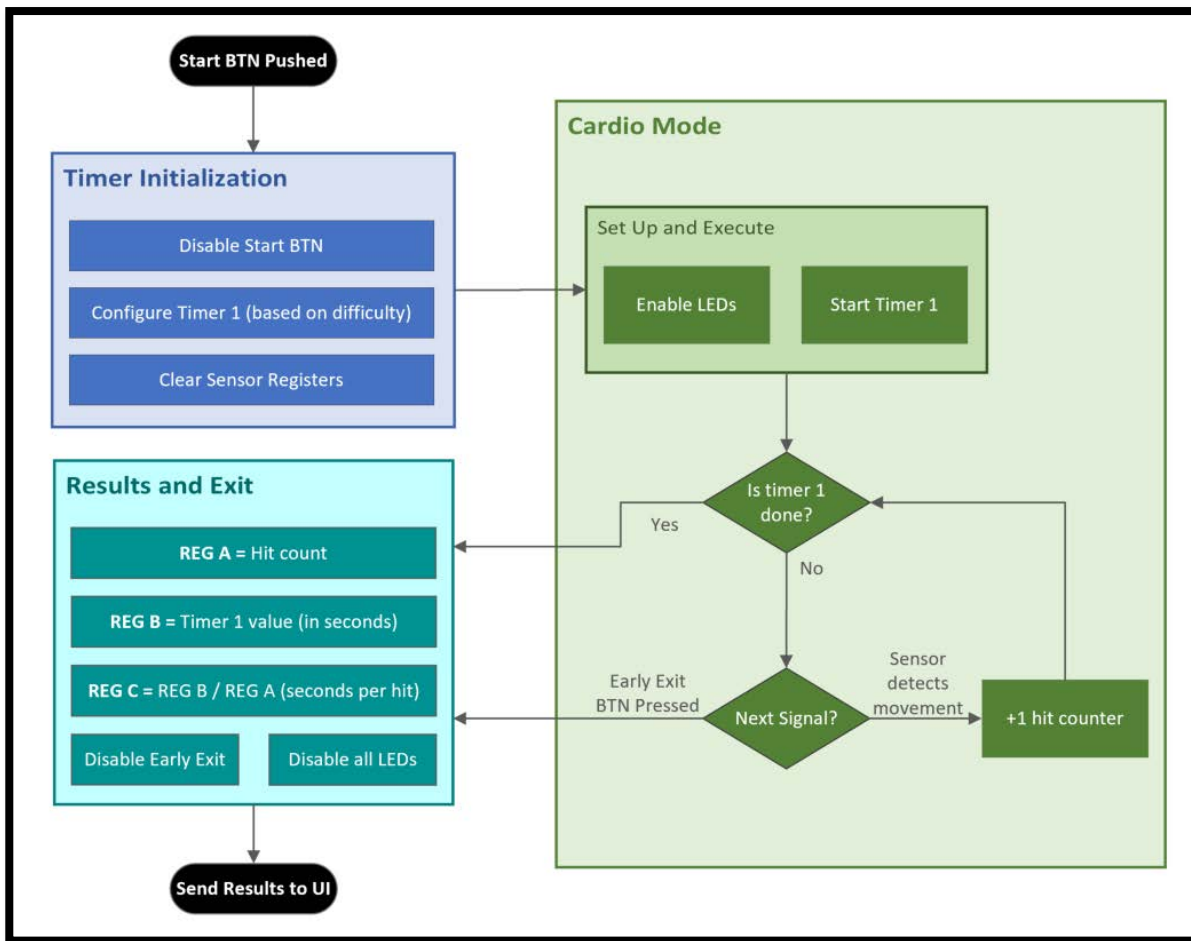

Figure 23: Software flow chart for the Cardio Mode on Side B

Ambient Mode

The last mode that the workout bag includes is the Ambient mode. This mode uses side A and B. In the ambient mode, all the LEDs are enabled, and cycled through the display colors. Because this mode does not require any user interaction, the mode will require an automatic way to shut off after a given amount of time. This prevents the workout from being in an infinite loop with all the LEDs on. This requires the use of one timer, that repeats itself for a given number of times, to allow the Ambient mode to remain on for a decent length of time before shutting off.

The ambient mode can also be disabled by using the exit early button to return to the welcome screen/mode selection.
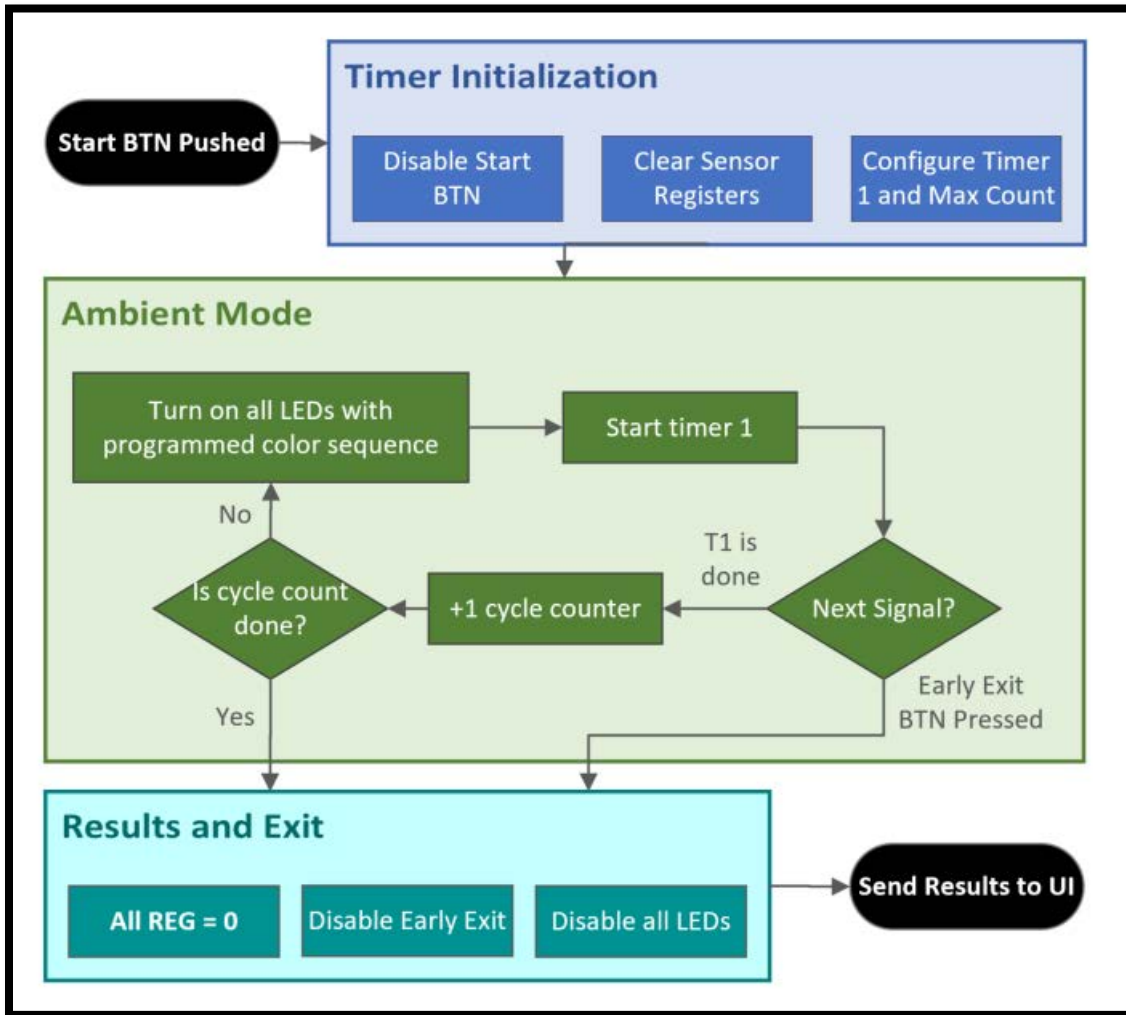

Figure 24: Software flow chart for the Ambient Mode using both sides A and B

Results Computation

The workout bag will have several different methods of computing results based on the mode of the session. In general, the MCU will collect the relevant data according to the mode in easily accessible memory. Once a session is complete, the MCU will perform basic arithmetic to find the desired output values. These are the values that the base system transmits to the UI system.

Combination Generator Mode

In this mode, the user is tasked with hitting different target areas in a specific amount of time. The overall goal of this mode, is to measure the number of successful hits within the given time allotment. Therefore, the results sent to the UI system include the total time of the session, the number of successful target hits, the number of total target areas, and the average time for a hit.

The total time of the session comes directly from the first timer. Because the first timer can be interrupted with the exit button, the program uses the value of the timers compare register to calculate the time by knowing the clock type's frequency. This is done by dividing the total number of cycles (this also needs to be calculated, as the total number of cycles can be higher than the maximum value a register can hold) by the clock's frequency. For example, if the total number of cycles is 32,768 (0 to 32,767) and the clock frequency is 32kHz, then the elapsed time is 1 second.

The total number of successful hits is calculated in real time. As a sensor detects a hit, the MCU increments the register that holds this value. Similarly, the total number of targets is also calculated in real time. Because this mode is timer based, the total number of targets depends on how fast the user is able to hit them. Therefore, this value is also calculated in real time, by incrementing another register every time the target area is updated, meaning when the secondary timer's interrupt is called.

Lastly, the average time for a hit is calculated in two stages. First, each time a hit is registered, the secondary timer value is added into a specific register. This creates a sum of the total amount of time of every hit. In the second stage, the total time of every hit is divided by the number of successful hits.

Reaction Time Mode

In this mode, the user is training their reaction speed, therefore the results that are most useful are the average time for a hit, the number of hits, the longest time for a hit, and the shortest time for hit. These are the four results that will be sent to the UI system.

Similar to the combination generator, the average time for a hit is calculated the same exact way. In the event the user discontinues the session without pressing the exit early button, the time it takes for the system to cycle through will not be counted, since the sensors did not detect a hit, allowing the results to not be affected. The total number of hits are also calculated the same way as the combination generator's successful hits.

The longest time for a hit and shortest time for a hit are found by storing each hit's time in an array. At the end of the session, the maximum and minimum value of the array are found, and stored in separate registers. The array holding all the times is then cleared to avoid keeping unnecessary data. If the sensors do not detect a hit, meaning the user has allowed the secondary timer to expire and the exit ISR is called, the time is not stored into the array, allowing the results to not be affected.

Accuracy Mode

In the accuracy mode, the user is training their ability to produce accurate hits to a given target. The results that are most useful are: most accurate hit (best) distance, most accurate hit (best) reaction time, least accurate hit (worst) distance, least accurate hit (worst) reaction time, accuracy rate, and hit rate. These are the 6 results sent to the UI system.

The best and worst hit distances are found by storing each hit distance into a distance array. At the end of the session, the maximum and minimum value of the array are found, and stored into separate registers. The best and worst hit times are found the same way. Similar to the reaction mode, when the second timer's ISR is called, it means the bag did not receive a hit, and the time is not stored. This keeps the index of both distance and time arrays to be consistent, where the index relates the time of the hit to the distance of the hit.

The accuracy rate is found by using the distance array, where each index stores that hits distance. Each hits' distance is then compared to the threshold value, which is set depending on the difficulty setting. If the hits' distance is less than the threshold distance, the hit is considered accurate and a register that holds the number of accurate hits is incremented by one. At the end of the array, the total number of accurate hits is divided by the total number of hits (length of distance array) to give the accuracy rate (a decimal ratio).

The hit rate is found by dividing the total time of the session by the total number of hits. This allows the user to identify their progress of their accuracy rate with the speed at which they are training.

Cardio Mode

In the cardio mode, the user is performing as many hits as possible in a given amount of time. The most useful results for this mode are total number of hits, total time, average time per hit and hit rate.

The total number of hits is found in real time, by incrementing a register each time a sensor detects a hit in the target area. The total time is directly from the timer. Because the timer can be interrupted with the exit button, the program uses the value of the timers compare register to calculate the time by knowing the clock type's frequency. This is done the exact same way as in the combination mode generator.

The average time per hit is calculated by dividing the total time value by the total number of hits. This allows the user to see the average time it takes them to complete a hit. Lastly, the hit rate is found by dividing the total number of hits by the total time value, or inverting the average time per hit value. This allows the user to see the average amount of hits per second they are able to complete.

Early Exit

The system has another way to end the sequence early, if desired by the user. A button located on the side, near the bottom of the base, is an exit button, that allows the user to terminate the current sequence with an interrupt and skip to computing the results and sending them to the UI system. Basically, this button allows the user to call the first timers (session timer) interruption and end the program early. Since the user may want the data from their progress so far, the current results are still processed and sent to the UI system.

Administrative Content: Design Considerations and Decisions
Initial Milestones and Timing for Senior Design 1 and 2
Components: Decide Processor, Sensors, Indicators, UI System
The prototype should be done by the end of December, and the prototype UI System should be finished by the end of January, to give us enough time to debug the project, and move our components over to the final design. Temperature sensors are optional to the very end to measure the temperature of the sensor grid and to make sure it is not overheating.

Construction: Complete Sensor Grid (Side A) and Target (Side B)
This project will need multiple areas with sensors, so these will need to be completed as soon as possible to be able to incorporate the electrical components, indicators, and the software which depend on the sensors. The deadline for the first sensor grid prototype is beginning of January, and the other three are no later than the end of January.

Final Touches: PCB and Interface Completion
The PCB components should be completed no later than the beginning of March. This will give us time to finalize the project and be sure that all modes that are included are functioning as intended. Once this has been verified, we would like to begin working on additions that will make this device something that will be worth the cost for a customer. One possibility of this is making an extra display on the body of the bag to interface with the user and add another element of interactivity. This will be challenging due to the frequent impacts the bag will face that could damage the screen, but it would allow more elements to be added to diversify the training possibilities from this device alone.

Group Member Familiarity with Project Elements
- **Hannah** is familiar with software since she works in a company with software engineers. She also likes writing code and programming in her free time.
- **Joseph** is familiar with processors and microcontrollers. He has knowledge of the process of ensuring a system is powered sufficiently.
- **Nicole** took an in-depth sensors class, so she is familiar with various types of sensors that we could use for this project.
- In addition to knowledge of typical electrical engineering curriculum, **Natesha** has many years of experience with and regularly uses a self-standing kicking bag since she is a fourth-degree black belt in karate and works as a sensei in her dojo.

With the group's combination of knowledge of electrical engineering principles and real-world experience with all aspects of this project, we should be able to successfully create and develop this device.

Project Challenges

Most challenges that our group will be facing are a result of the CoVID-19 pandemic that we are currently living through. Due to social distancing, we will need to work on our sections separately; as of now we will not be able to get together unless it is to test the device. As of now we are doing most of our meetings and communication via Discord, but once sensor/indicator testing begins, we will likely face difficulty in being sure everyone has what they need to individually test components.

In general, being able to optimize the parts used to cater to all specifications as well as being low cost, accessible, and realistic for the scope of this design prototype will be a challenge. The pandemic has also slowed electrical component production around the world, meaning that we will have to be sure about our designs before ordering many items and begin attaining and assembling components as soon as possible in attempts to avoid roadblocks.

One non-pandemic related obstacle is the size and scale of this project. A self-standing training bag is a large and heavy piece of boxing equipment, so this component should be in a location accessible to all members with sufficient space to work on device assembly and construction.

Budget and Financing

Our group will self-fund this project. We are budgeting to contribute a maximum of $500 USD, divided equally among our teammates. As of now, no equipment aside from standard circuit components (eg. wires, resistors) has been acquired. All parts of this project must be attained at some later point. Table 1 below depicts estimated costs of each component.

The individual component that would cost the most money is a self-standing kicking bag since it is a big and technical instrument aimed at people who are experienced in this area. However, finding a used bag will lower the cost of this item, and will not make a difference in the quality of the bag as it will need to be changed to incorporate our components regardless of its condition.

Although sensors are generally inexpensive, we would need enough components to detect contact in the large target area so that the device can test accuracy in addition multiple sensors to detect contact in a binary "hit or no hit" manner in all areas of the sensor grid.

Component	Price (USD)	Quantity	Total (USD)
Sensors	$0.4 to $5.00	20	$6.40 to $100.00
Processor	$10.00 to $15.00	2	$20.00 to $30.00
RAM	$50.00	2	$100.00
Miscellaneous Buttons	$1.00 to $5.00	4	$4.00 to $20.00
Standup Punching Bag	$130.00	1	$130.00
Indicators	$10.00 to $30.00	1	$10.00 to $30.00
AC to DC converter	$2.36 to $20.00	1	$2.36 to $20.00
Miscellaneous Hardware	$20.00 to $30.00	n/a	$20.00 to $30.00
Total			$292.76 to $430.00

Table 16: Expected costs for this project

References

[1]   A. D. Morales, M. C. Morales and L. C. Eldridge, Jr., "Interactive systems and methods for reactive martial arts fitness training". US Patent US20110172060A1, 18 11 2010.

[2]   ALITOVE Store, "ALITOVE RGB Addressable LED Strip WS2811 12V LED Strip Lights," Amazon, [Online]. Available: amazon.com.

[3]   American Heart Association, "Warm Up, Cool Down," American Heart Association, 1 09 2014. [Online]. Available: https://www.heart.org.

[4]   Association Connecting Electronics Industries, "IPC 2221B," Techstreet, 01 11 2012. [Online]. Available: https://www.techstreet.com.

[5]   ASTM International, "ASTM F2276 - 10(2015)," ASMT International, 2015. [Online]. Available: https://www.astm.org.

[6]   D. Hunt, "MSP430F5529LP example code: LCD Counting Example Using a 4-bit Interface," davesmotleyprojects, 2019. [Online]. Available: http://www.davesmotleyprojects.com.

[7]   D. Lapkova and M. Adamek, "Using Strain Gauge for Measuring of Direct Punch Force," *XXI IMEKO World Congress "Measurement in Research and Industry",* 2015.

[8]   D. Lapkova, L. Kralik and M. Adamek, "Possibilities Of Force Measuring In Professional Defense," *XXI IMEKO World Congress "Measurement in Research and Industry",* 2015.

[9]   Duke Energy Indiana, "Average Duke Energy Electric Bill," Duke Energy, 08 05 2020. [Online]. Available: https://www.citact.org.

[10]   Enviorment and Protection Agency, "Regulations, Initiatives and Research on Electronics Stewardship," Enviorment and Protection Agency, [Online]. Available: epa.gov.

[11]   F. D. o. E. Protection, "Electronic Waste," Florida Deparment of Enviormental Protection, [Online]. Available: https://floridadep.gov/waste/permitting-compliance-assistance/content/electronics-waste.

[12]   International Electrotechnical Commission, "IEC 60906-2:2011," Webstore, 12 05 2011. [Online]. Available: https://webstore.iec.ch.

[13]   J. T. Adams, "An Introduction to IEEE STD 802.15.4," IEEE, 24 07 2006. [Online]. Available: https://ieeexplore.ieee.org.

[14]   J.-P. Rodrigue, "Point-to-Point versus Hub-and-Spoke Networks," Hofstra University, 2020. [Online]. Available: https://transportgeography.org.

[15]   Lady Ada, "Overview |EL Wire," Adafruit, 29 07 2012. [Online]. Available: https://learn.adafruit.com.

[16]   Longrunner, "for ArduinoIDE, Longruner 20x4 LCD Display Module IIC/I2C/TWI Serial 2004 with Screen Panel Expansion Board White on Blue, 4 pin Jump Cables Wire Included," Amazon, [Online]. Available: https://www.amazon.com.

[17]   M. Alves, "The IEEE 802.15.4/ZigBee protocol stack architecture," ResearchGate, [Online]. Available: https://www.researchgate.net.

[18]   Raspberry Pi Foundation, "Raspberry Pi 4 Tech Specs," Raspberry Pi Foundation, [Online]. Available: https://www.raspberrypi.org.

[19]   T. K. Hareendran, "EL Wire Experiments & A Minor Hack," Electro Schemantics, 02 05 2019. [Online]. Available: https://www.electroschematics.com.

[20]   Texas Instruments Incorporated, "66AK2G1x Multicore DSP+Arm KeyStone II System-on-Chip (SoC) (Rev. F)," Texas Instruments Incorporated, [Online]. Available: https://www.ti.com.

[21]   Texas Instruments Incorporated, "MSP430F5528," Texas Instruments Incorporated, 11 9 2020. [Online]. Available: https://www.ti.com.

[22]   Texas Instruments Incorporated, "ZigBee RF Transceiver Datasheet," Texas Instruments Incorporated, 12 2007. [Online]. Available: ti.com.

[23]   The Hook Up, "The COMPLETE guide to selecting individually addressable LED strips," The Hook Up, 5 08 2019. [Online]. Available: http://www.thesmarthomehookup.com.

[24]   "Wavemaster," Century, [Online]. Available: https://www.centurymartialarts.com.

[25]   yida, "UART vs I2C vs SPI – Communication Protocols and Uses," Seed Studio, 25 09 2019. [Online]. Available: https://www.seeedstudio.com/blog/2019/09/25/uart-vs-i2c-vs-spi-communication-protocols-and-uses/.

[26]   yida, "UART vs I2C vs SPI – Communication Protocols and Uses," Seed Studio, 25 09 2019. [Online]. Available: https://www.seeedstudio.com/blog/2019/09/25/uart-vs-i2c-vs-spi-communication-protocols-and-uses/.

Appendix

A) Pinout Diagram of MSP430F5528

Figure 4-2 shows the pinout for the MSP430F5528, MSP430F5526, MSP430F5524, and MSP430F5522 devices in the RGC package.

NOTE: TI recommends connecting the exposed thermal pad to V$_{SS}$.