Directional Impact Sensing Helmet (DISH)
ECE 445 Project Proposal

Team 5: Patrick Sear, Saathvik Narra, Ryan Josephson
Professor: Viktor Gruev
TA: Ugur Akcal
Spring 2023

1.   Introduction

1.1   Problem

In the NFL, many athletes suffer from concussions or other conditions as a result of repetitive, severe head trauma. These events can lead to long-term effects on the athlete's health and significantly contribute to the reduced lifespan of professional football players. The average professional football player dies younger than age 60 [1], no doubt accelerated by frequent, intense head trauma. This problem can be helped by making accurate collision data immediately available to medical personnel for making game-time decisions and helmet manufacturers so that they can make informed design choices based on real, in-game data.

1.2   Solution

The Directional Impact Sensing Helmet (DISH) is an electronic system that can be attached inside any existing football helmet. It can determine where on an athlete's head a collision occurs, as well as how hard the hit is. The data collected will be quickly available to personnel on the sidelines so that they can make real-time decisions to ensure the safety of their players. The information will be visualized for ease of use of the team medical staff so that they can best inspect the player.

The data will also be useful for designing the next generation of helmets. Over the course of a season, all hits can be tracked and analyzed so that newer models can be tailored precisely to protect from hits that occur in a real, game-time environment. The helmets could even be designed differently for each position, prioritizing the most common hits each role receives.

For the DISH to work properly, we can group our project into three modules: power, control, and data reception and visualization.

1.3 Visual Aid
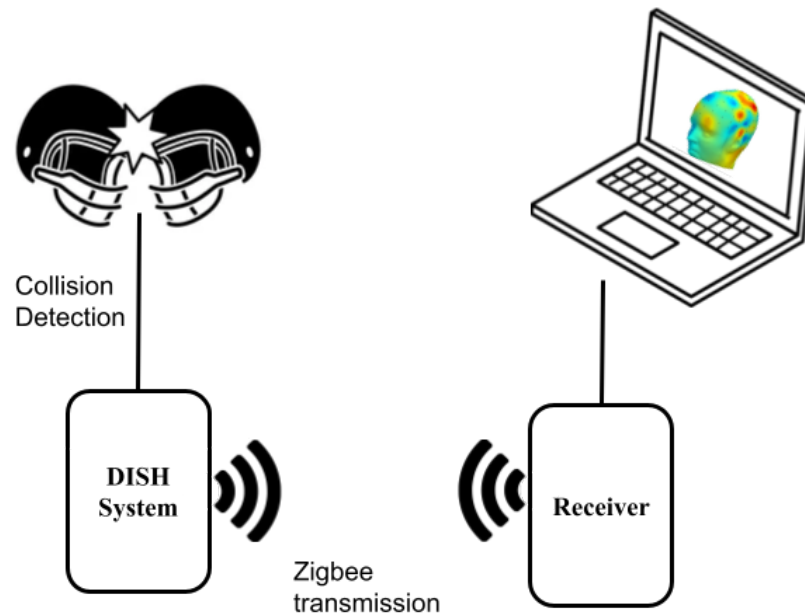
Fig. 1: High-Level Overview of DISH System

1.4 High-Level Requirements

- The helmet must track the location and severity of each collision within <20% error.
- The data must be transmitted to the receiver and visualized in the software before the start of the next play (< 30s), such that immediate substitutions can be made.
- The in-helmet system must be able to comfortably fit within a standard football helmet.
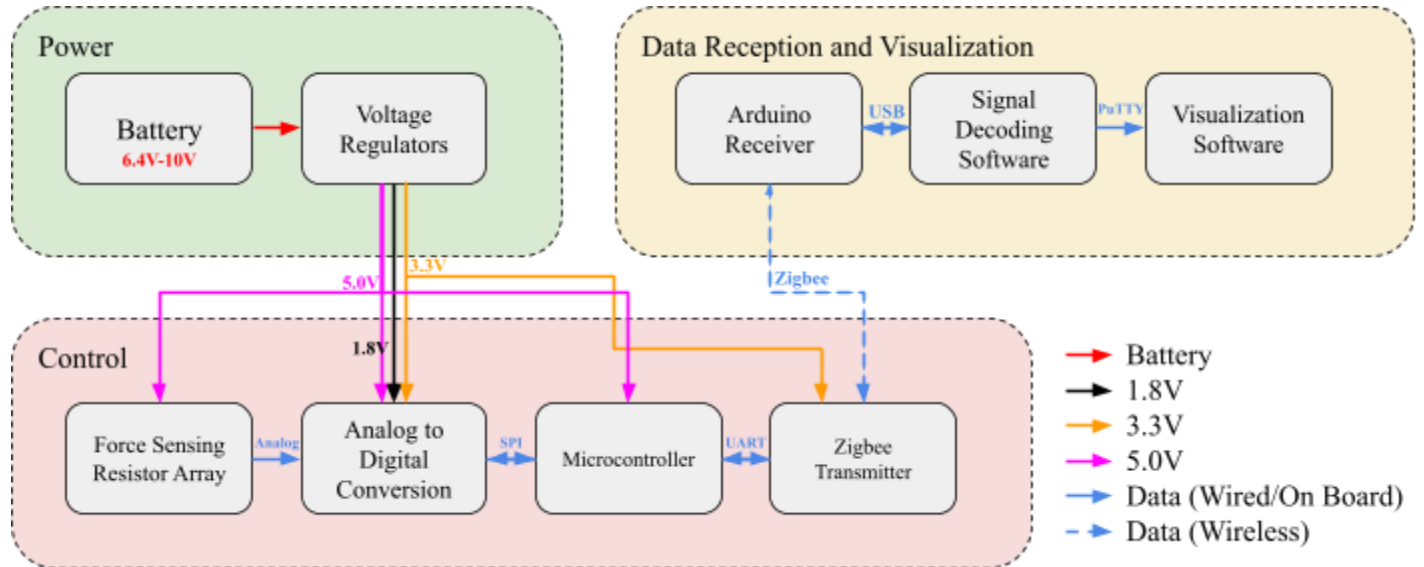
2. Design

2.1 Block Diagram

Fig. 2: Block diagram

2.2 Block Descriptions

2.2.1 Power
The electronics located within the helmet need to receive power. For this reason, the Power Subsystem will contain batteries and voltage regulators.

2.2.1.1 Batteries
The power subsystem will be powered using batteries. These batteries are the only source of power for the entire helmet. For this reason, they are small enough and durable enough such that they don't pose any safety hazards to the player. We plan on using LiFePO4 or NiMH batteries, as these tend to be more durable and would be safer for our purposes.

Requirement 1: The battery must output a voltage compatible with the overlapping input voltage ranges [2] of the voltage regulators.
Requirement 2: The battery must be capable of providing enough power for the entire system over an elongated period of time.
Requirement 3: The battery must be small enough to fit within the helmet without posing a safety hazard.

2.2.1.2 Voltage Regulators

To ensure we have constant voltages throughout our system, we plan on using SPX1117 voltage regulators to attain constant 1.8V, 3.3V, and 5V rails. The 1.8V rail will be used for the off-chip simultaneous sampling ADC. The 3.3V rail will be used to power the ADC and the Zigbee transmitters. The 5V rail will be used to power the microcontroller, force sensing resistors, and the ADC.

Requirement 1: Each voltage regulator must be able to supply enough power to the components they are supporting.

Requirement 2: Each voltage regulator must be maintained within voltage ranges that are required by their supported components.

2.2.2   Control

This subsystem is responsible for all control signals within the helmet and the communication between the helmet and the receiver.

2.2.2.1 Microcontroller

The microcontroller we plan on using is an ATmega32U4, or a similar chip. This is because we should be able to program it with the Arduino bootloader, and then use the Arduino IDE to implement software. This microcontroller will be in direct communication with the off-chip simultaneous sampling ADC and the Zigbee transmitter. Using the microcontroller, we will make a trigger that, if triggered, would pass data from the force sensing resistor array to the receiver.

Requirement 1: The microcontroller must be able to communicate with the analog to digital converter via SPI.

Requirement 2: The microcontroller must be able to correctly identify when a collision occurs.

Requirement 3: The microcontroller must be able to send and receive signals to and from the remote receiver via the Zigbee transmitter.

2.2.2.2 Force Sensing Resistor Array

There will be an array of strategically placed force sensing resistors placed within the helmet. They will be used to measure how much force is received to the head. To be able to measure relevant data from the force sensing resistors, we will construct voltage divider circuits for each force sensing resistor such that we can measure voltage changes when a force is applied to the sensor.

Force sensitive resistors rated for the intensity of collision seen in a football game can be quite expensive. For the sake of this project, we are going to use force sensing resistors that are rated for a lower amount of force. If we were to change these sensors for the more expensive alternatives, very few changes would be required. Effectively, the entire system would be the same, save for some resistor values and the calibration of some measurements. For this reason, we believe it is acceptable to use force sensing resistors with a lower force rating.

Requirement 1: The placement of each force sensing resistor must provide useful data that another force sensing resistor couldn't collect.
Requirement 2: Each force sensing resistor must be measured in such a way that changes to the force applied can be measured.

2.2.2.3 Analog to Digital Conversion

Due to how the ATmega32U4 MUXes its analog inputs, we are going to use an off-chip simultaneous sampling ADC. We plan on using the ADS1178IPAPR chip, since it can take up to eight simultaneous analog inputs and communicate via SPI, so we can use fewer pins in the microcontroller.

Requirement 1: The off-chip simultaneous sampling ADC must be able to measure all signals from the force sensing resistor array concurrently.
Requirement 2: The off-chip simultaneous sampling ADC must be able to communicate with the microcontroller.
Requirement 3: The off-chip simultaneous sampling ADC must not lose any meaningful data.

2.2.2.4 Zigbee Transmitter

The microcontroller in the helmet will receive the data from the ADC and will transmit the data to the receiver via a Zigbee transmitter. Zigbee is an optimal communication protocol because it is cheap and has a range that is usable for our purposes. To ensure the wireless connection between the microcontroller and the receiver is intact, the microcontroller will send a verification signal periodically to the receiver. The specific module we are going to use is most likely the ZB Series S2C XBee Module.

Requirement 1: The Zigbee transmitter must be able to properly send a signal up to 40 meters away.
Requirement 2: The signal must be able to be sent through the helmet and through adverse conditions.

2.2.3   Data Reception and Visualization

This subsystem is responsible for digesting the data collected from the helmet and displaying it meaningfully to the user.

2.2.3.1 Arduino Receiver

The receiver will consist of an Arduino Uno Rev3 and a Zigbee receiver, likely the ZB Series S2C XBee Module, connected to a personal laptop designed to receive the packet transmitted from the Zigbee. The serial port data will be processed via PuTTY or a similar client into a file which can be analyzed by the visualization software.

Requirement 1: The receiver must be able to receive all packets without error in a timely manner.
Requirement 2: The microcontroller on the remote's Arduino needs to be able to read data received by the Zigbee.
Requirement 3: The receiver must be able to convert the data from a Zigbee signal into data accessible to outside software.

2.2.3.2 Signal Decoding Software

The software must be able to decode the Zigbee signal, which would entail identifying which player the signal is from and the force data from each individual sensor.

Requirement 1: The decoder must be able to determine exactly which player the data is from.
Requirement 2: The decoder must be able to decipher the force data meaningfully.

2.2.3.3 Visualization Software

Upon receiving new collision data, the software will update to make it available to view. It will likely use React and Javascript to generate a heat map we can place over a 3D model to display the location and severity of the collision.

Requirement 1: The software must be able to translate received data into a heatmap which can be overlaid onto a 3D model.
Requirement 2: The user must be able to view and edit the raw data received.

2.3   Tolerance Analysis

We'll need to ensure that we can obtain a sufficient digital resolution when converting the analog signals for use in the microcontrollers. This problem is exacerbated by the inherent nonlinearity of the force sensing voltage divider and force sensitive resistor. The voltage divider is planned to be constructed as shown below in *Figure 3*.
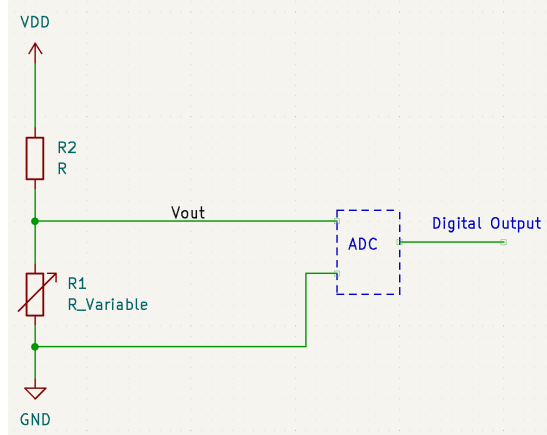
Fig. 3: Voltage divider schematic

With this schematic, the analog output will be governed by a nonlinear relationship to R1, which is itself already a nonlinear dependance on force. The output voltage potential is as seen below in *Equation 1*.

$$V_{out} = V_{DD} * \frac{R1}{R1+R2} \quad (Eq.\ 1)$$

Simulating the output of the voltage divider using a standard force sensitive resistor dependency highlights the potential problems with this setup. For the expression shown in *Equation 2*, R_f is the resistance of the force sensing resistor, R_0 is an initial resistance value with no force applied, and $F_{applied}$ is the applied force. The -0.77 exponential is an estimation based on the operation of a force sensing resistor that is on the market [3]. A force vs output voltage plot can be seen below in *Figure 4*.
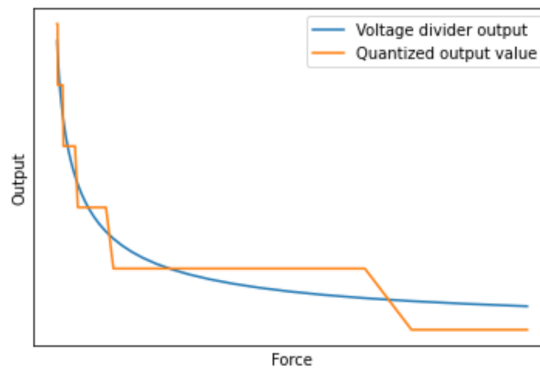
$$R_f = R_0 * F_{applied}^{-0.77} \quad (Eq.\ 2)$$

Fig. 4: Risk of Quantization at 5 Level Resolution

Clearly, this system has too few voltage steps, and will not be able to digitally differentiate analog signals at high force levels. We plan to use a 16-bit ADC, so we do not expect this to be a problem. But, it is still a concern to be aware of during the design process. The simulated 16-bit resolution overlaid with analog voltage output as a function of force can be seen below in *Figure 5*.
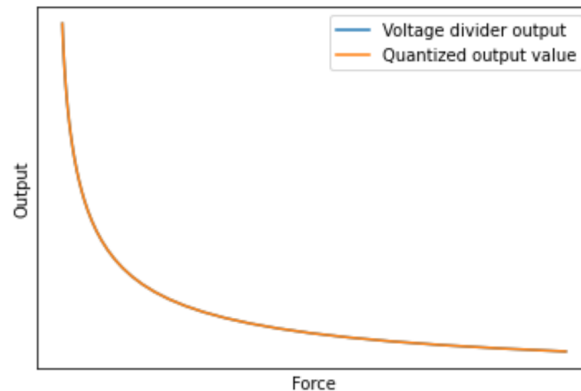
Fig. 5: Risk of quantization at 16-bit resolution

Further concerns are raised when we actually implement the calculation to determine the force form the output voltage. Because the resistance of the force sensing resistor is exponential, we may encounter some trouble with precision of the force calculated. Using the voltage divider circuit as seen in *Figure 3*, we can strategically select our R2. By doing this, we can experience more noticeable changes from the force sensing resistor when experiencing forces of magnitude that are of interest. This can help with linearization of data to minimize other calculations.

3. Ethics and Safety

Given that this device is designed to be used in a high-impact environment, it is critical that it is designed with only the highest standard of safety in mind. The IEEE Code of Ethics states that it is our responsibility "to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design" [4]. This is especially relevant since our device will be located within a helmet, very close to the head of the user.

It is important that the physical footprint of the device is as small as possible so that it will not pose any extra risk to the wearer. Existing in-helmet devices like quarterback transceiver modules can serve as a model of safe physical design choices. A small, plastic encasement with additional padding will help to mitigate any risk presented by the device.

We also must consider the inherent risk of a battery system. We must consider both battery placement, as well as the durability of the battery. There is a risk of placing a battery so close to

the head and in a heavy contact environment. The batteries must be well encased and have no possibility of failing in a dangerous way due to a collision. Per OSHA guidelines [5], potential hazards to lithium batteries include "dropping, crushing, and puncturing," each of which are possible within a football game. The device will also be located in the bottom corner of the helmet, somewhere collisions are less frequent, so the circuit and battery will be under reduced stress. We must also ensure there is minimal risk of our battery exploding, as certain lithium batteries have this possibility. According to the OSHA [5], to reduce risk to the user, the batteries must be inspected "for signs of damage, such as bulging/cracking, hissing, leaking, rising temperature, and smoking before use" in case thermal runoff is imminent.

Another area for concern is the network of force sensing resistors placed within the helmet. We must ensure the addition of our device does not reduce the safety the football helmet already provides to the user. For this reason, we must ensure all padding is maintained within the helmet. With this, we must also make sure all cables placed within the helmet pose no safety hazards, including extra pressure felt on the head and possibly loose cables.

References

[1]   Nguyen VT, Zafonte RD, Chen JT, et al. Mortality Among Professional American-Style Football Players and Professional American Baseball Players. JAMA Netw Open. 2019;2(5):e194223. doi:10.1001/jamanetworkopen.2019.4223

[2]   MaxLinear, "800mA Low Dropout Voltage Regulator," SPX1117 datasheet, Sept. 2018, https://assets.maxlinear.com/web/documents/sipex/datasheets/spx1117.pdf.

[3]   Adafruit Industries, "Force Sensitive Resistor (FSR)," Force Sensitive Resistor (FSR) datasheet, Dec. 2021, https://cdn-learn.adafruit.com/downloads/pdf/force-sensitive-resistor-fsr.pdf.

[4]   "IEEE Code of Ethics," IEEE.org, 2023, https://www.ieee.org/about/corporate/governance/p7-8.html.

[5]   Occupational Safety and Health Administration, "Preventing Fire and/or Explosion Injury from Small and Wearable Lithium Battery Powered Devices," June 2019, https://www.osha.gov/sites/default/files/publications/shib011819.pdf.