Soccer Team Gameplay Metrics

Team 54 — Regis Lato, Shixing Ma, Yi Rui Zhao

ECE 445 Design Document — Spring 2019

TA: Dongwei Shi

1 Introduction

1.1 Objective

In recent years, the use of data analytics is sports have become more and more widespread. To get an edge in games, players and coaches use data to make decisions in training and gameplay [1]. However, in the sport of soccer, there isn't an easy way of collecting data for player-ball interactions. Currently, if someone wanted ball possession data over the course of a game, they would have to record it by hand. It's a tedious task, and mistakes could be made during the data collection process, so it would be more effective to automate the entire process instead.

We aim to build a system that is able to measure and calculate metrics for individual players over the duration of a soccer game. For this project, we will be focusing on metrics between the player and the ball rather than more personal metrics. We will collect data using a sensor system integrated into a ball, and present it to the user on an application after analyzing it. The metrics that we aim to gather include: passes between player A and B, bad passes, longest string of dribbles, time of possession, shots on goal, and so on.

1.2 Background

Currently, there are companies that make smart soccer balls like the Adidas or DribbleUp, but their balls are primarily used for personal training. For example, the Adidas miCoach smart ball only records speed and spin of a shot from a stationary ball position [4]. You have to press a button to let the ball know that you are about to shoot it. This wouldn't work as we require touch recognition is a live game environment. The DribbleUp ball focuses on AR tracking of the ball [3]. This works with amount of touches and juggles for a player but making sure that nothing comes in between the camera and the ball and that the player is not in motion and focused in the middle of the camera frame. This again would not work in a live game environment. These smart balls lack the ability to be used in a multiple player drill session or in an actual soccer match. Thus we need to come up with a new solution that would fit this criteria.

1.3 High-level requirements list

- Each touch of the ball by an individual player must be registered and captured by the system.
- The system cannot be wired and must all work on battery power for at least 22.5 minutes.
- An application must be able to process the data and display the metrics such as passes between player A and B, bad passes, longest string of dribbles, time of possession, and shots on goal.

2 Design

2.1 Block Diagram

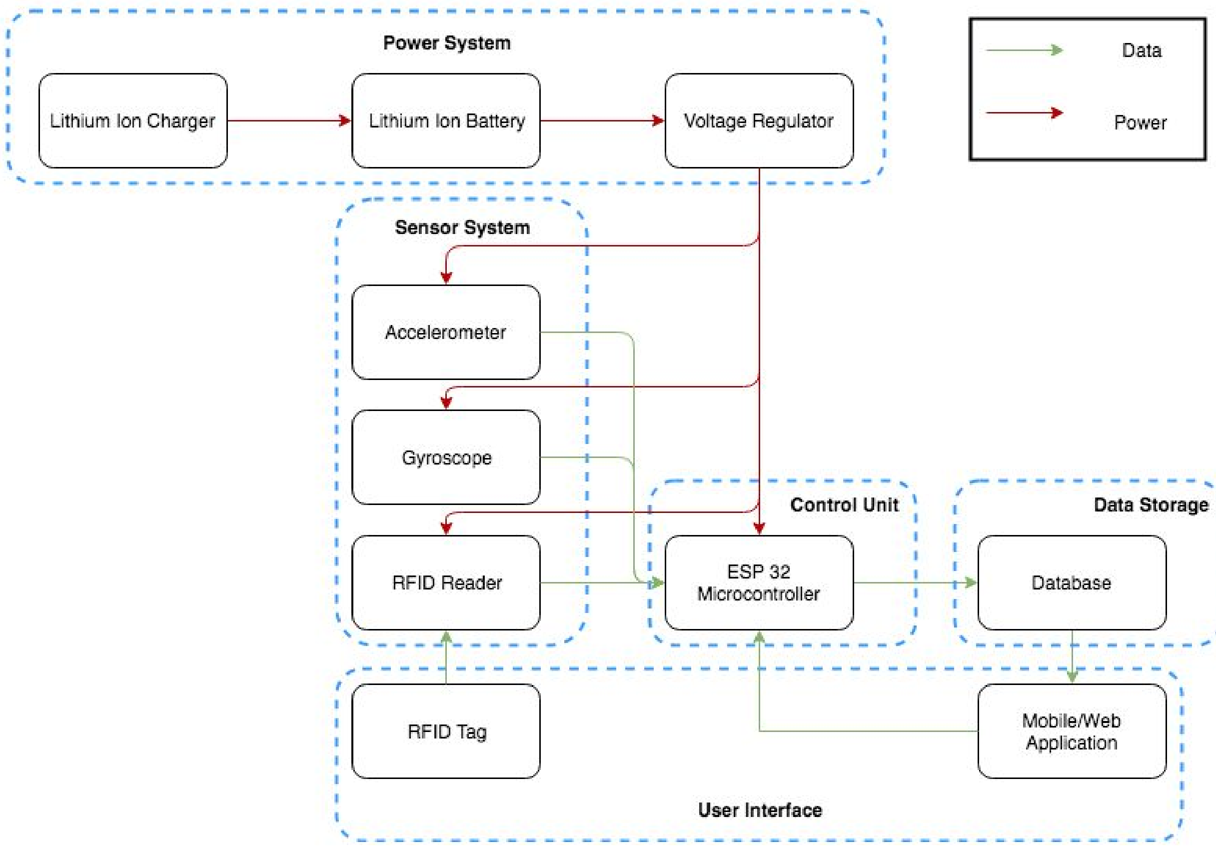


Figure 1. Block Diagram

2.2 Physical Design

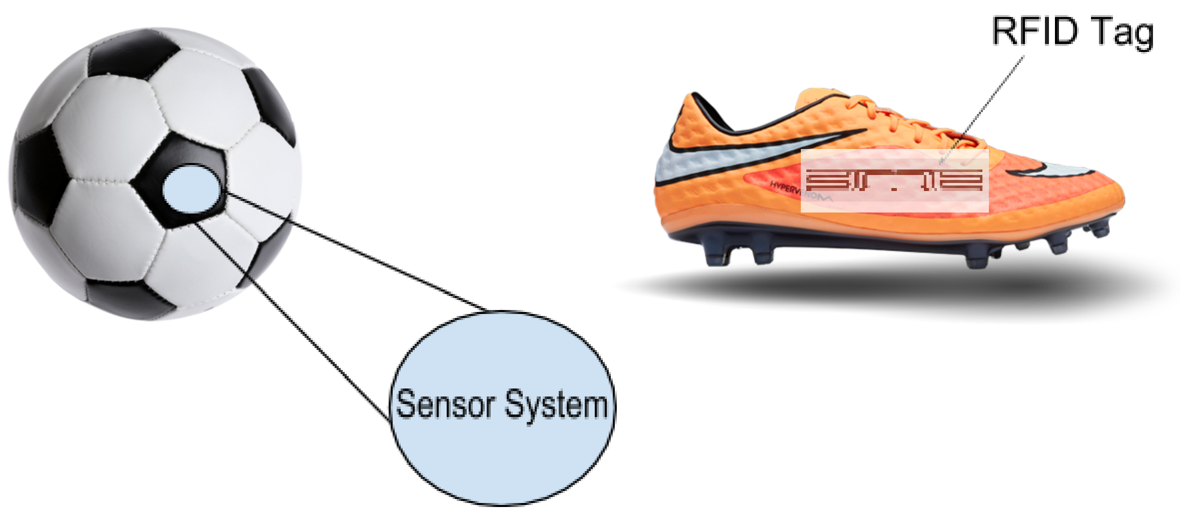


Figure 2. Physical Design

We will be integrating the power system, sensor system, and control unit inside a high density foam soccer ball. Thin RFID tags attached to sticker paper will be stuck on player's soccer cleats. For this project, we are more interested in collecting and analyzing data metrics than creating a robust product. Manufacturing processes already exist for shock absorption for electronics in soccer balls, so we will not be focusing on that aspect of the design at all.

2.3 Functional Overview & Requirements

2.3.1 Power System

The power system powers all the electronics that will go inside the ball. It will need to be able to support the sensors and microcontroller described in the next part. Typically, the voltage required for sensors are very similar, so the system will be relatively simple. For the parts that we have chosen, the most restricting component is the ESP32, which normally uses 3.3V and has a maximum voltage of 3.6V, while the other components can use any input voltage from 2.5 to 5 volts. Therefore we will build our power system to have a nominal 3.3V output.

- **Lithium Ion Battery**
  A soccer ball cannot be plugged in during a game, so we will have to use a portable power source to power the electronics inside the ball. Since none of the components we plan to use require high voltage or current, we will use a lithium ion battery as the power source. Since all of the sensor components can run off 3.3V, a LiPo battery with a 3.7V nominal voltage will be used.

Requirements	*Verification*
1. The battery must be able to sustain up to 250 milliamps of current draw at 3.3 volts for a quarter of a soccer match length (~22.5 minutes)	1. A. Fully charge the lithium ion battery to 4.2V. B. Connect it to a load that draws 250mA of current. C. Leave the load connected for 22.5 minutes. D. Disconnect it from the load and use a voltmeter to check that the voltage is above 3.3V.

- **Lithium Ion Charger**
  The charger will regulate voltage and current going into the battery during the charging process to ensure that the battery doesn't overcharge and break down. To accomplish this, we have chosen the MCP73833 Linear Li-Ion / Li-Polymer Charge Management Controller from Microchip. It's able to charge at 4.2V with a maximum current of 1A.

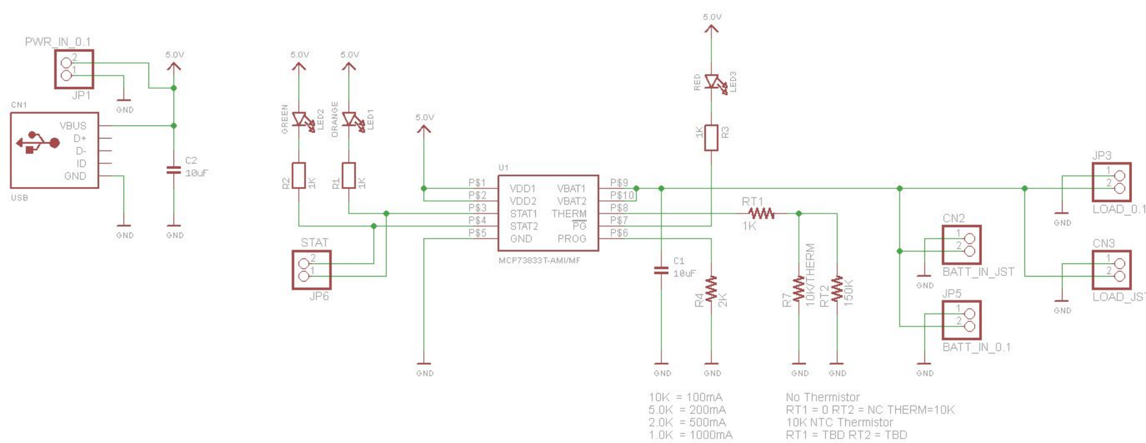


Figure 3. Charging Circuit Layout [8]

Requirements	*Verification*
1. The charger must be able to charge the battery to 4.2 volts with a continuous current of >100 mA from a 5V USB power source.	1. A. Discharge the lithium ion battery to 3.3V. B. Wire the charging module in figure 3. C. Connect the charger to a 5V

	USB source with the battery connected. C. Use a voltmeter and an ammeter to monitor the voltage and current of the charging IC to make sure that it charges with a current of at least 100mA, and stops at 4.2V.

- **Voltage Regulator**
  The voltage of a lithium ion battery varies depending on how much charge is left. Some sensors require a stable voltage input in order to operate properly, so a voltage regulator is required to accomplish this. The LD1117V33 low drop linear voltage regulator meets our requirements. It outputs a fixed voltage at 3.3V with a maximum current of 800 mA.

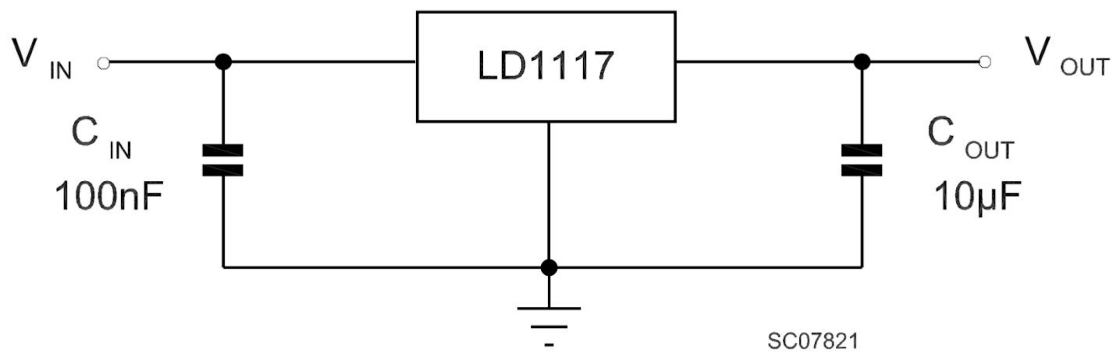


Figure 4. Voltage Regulator Circuit Layout [9]

Requirements	***Verification***
1. Able to step down battery voltage to 3.3V ± 5% while maintaining a 250 mA peak current.	1. A. Connect the regulator input to a 5V lab supply and wire the circuit as shown in figure 4. B. Connect the output of the regulator to a 250 mA load. C. Verify that the voltage is within 5% of 3.3V using a voltmeter.

2.3.2 Sensor System

- **Accelerometer and Gyroscope**
  The accelerometer provides data on the current speed of the ball. This will enable additional metrics to be computed, and will also help determine when the ball goes out of play.
  The gyroscope will provide data on the current spin and orientation of the ball. This will enable us to determine the spin on a shot or pass as well as provide data to help us determine other factors, like if the ball is stationary or not.
  We chose the BNO055 IMU that integrates both the accelerometer and gyroscope. It will communicate via I2C to send data to the microcontroller.

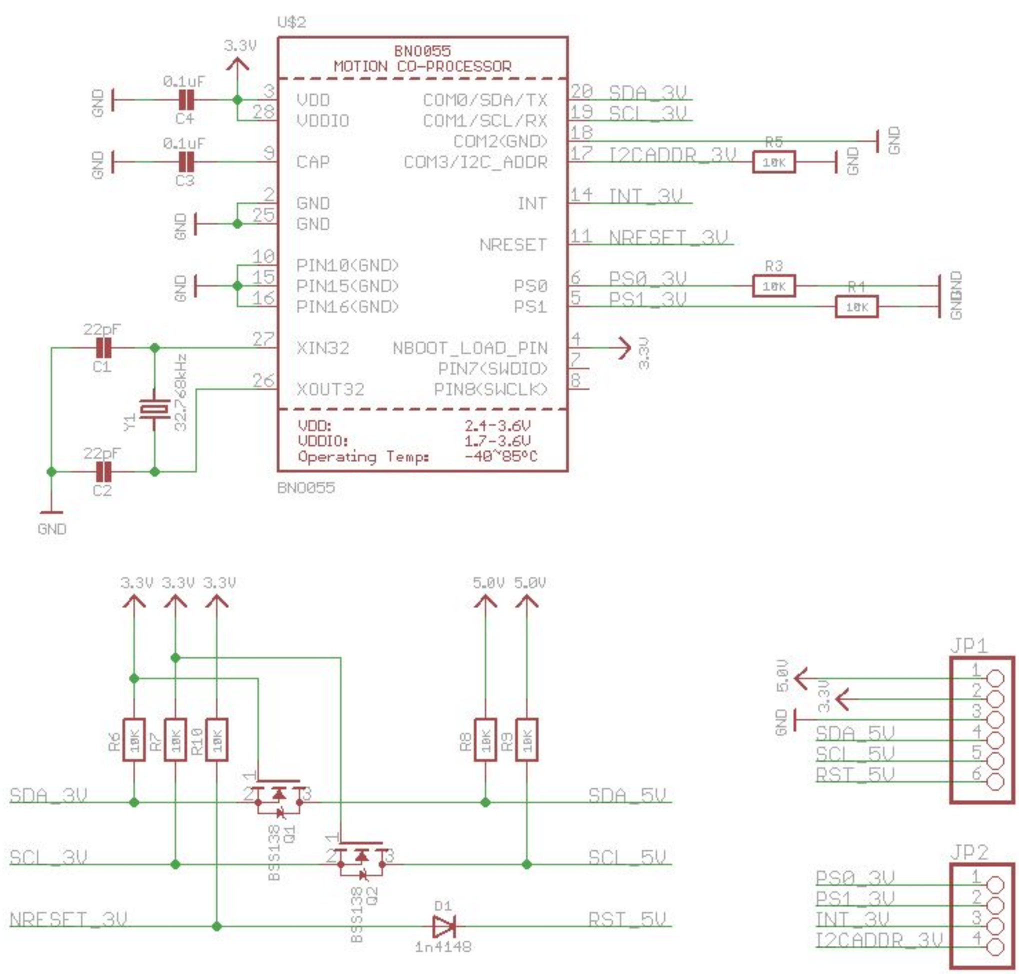
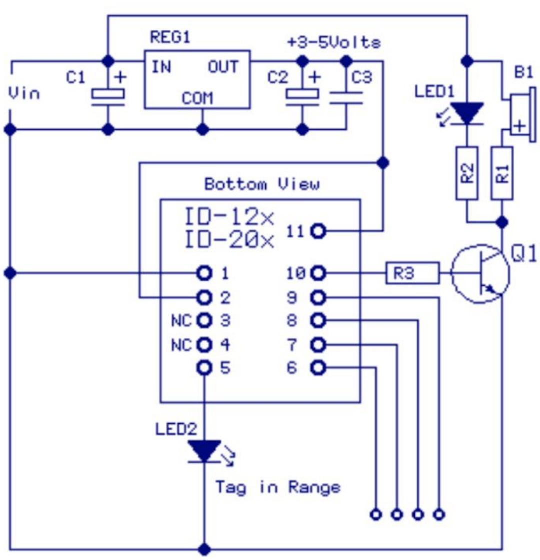


Figure 5. IMU Circuit Layout [10]

Requirements	***Verification***

Accelerometer: Measure a minimum of ±4g of acceleration, and at least has 3 degrees of freedom. *Gyroscope*: The gyroscope must be able to measure 2000 to 4000 degrees per second in 3-axis.	We will compare the accelerometer and gyroscope to an already working accelerometer sensor on iPhone XR. 1. A. Wire the IMU according to figure 5. B. Attach the sensor on the phone and test with different forces on movement and angular movement. C. Compare the change in the data on both sensors

- **RFID Reader**
  The RFID reader provides the primary method of detecting the person that is in contact with the ball. It will read the RFID tag of the player in possession and send that information to the microcontroller.



Parts List	
Part #	**Value**
R1	100R
R2	4K7
R3	2K2
C1	10uF 25v electrolytic
C2	1000uF 10v electrolytic
C3	100nF
Q1	BC457 or similar
LED1	Read LED
LED2	Tag In Range LED
B1	2.7khz – 3kHz 5v PKPK AC

Figure 6. ID-20LA Circuit Layout [11]

Requirements	***Verification***
1. Must at least have a detection distance of the radius of a soccer ball (~11 cm). 2. Read time should be faster than the smallest contact time of a typical kick (~8 ms)[2]	1. Place the RFID tag at a distance of 11cm and check if the sensor can read it. 2. Move the RFID tag in the field of the RFID reader at a speed that would make it in contact for about 8ms and verify if there is a read.

2.3.3 User Interface

- **RFID Tag**
  Each player will have an unique RFID tag attached to their cleats to be identified with. The RFID tags should be about as thin as possible so that it hardly affects the player and is easily attached inside or on the cleat.

Requirements	***Verification***
1. Each tag should have a unique identifier/data string.	1. A. Move the tag to where the RFID reader can read and see if the tag reads. B. Read each tag multiple times to see if the tag is unique and that the tag reads remain the same for the same tag.

- **Web/Mobile Application**
  The application is the processing and visualization of the data that shows player's stat and analysis.

Requirements	***Verification***
1. It must be able to correctly compute and display stats in a table/list and have interactive features that allow the users to select the items to display.	1. A. Enact a scenario where we personally calculate the completed passes, dribbles, misplaced passes etc. B. Check if the algorithms and application calculate and display the same values

The algorithm processes the IMU data collected inside a certain time window with some filters (e.g. a low pass filter to remove noise and gaussian filter to detect kicking impulse) to determine the last state of the ball's movement.

If the ball was stationary for some time over the threshold (e.g. 2 seconds) then the ball was out of play (e.g. free kick, corner,etc.) and no stat computation will be done.

If a kick action was detected then combined with RFID data the algorithm should determine whether the ball was dribbled or passed. The RFID data is interpreted as the possession of the ball.

If the change of possession was between the players in the same team then the pass is good. If there was no change in possession (RFID reading didn't change) then it was a dribble.

If the change happened between two players from different team then it was a interception or bad pass.

Each action is time stamped and stored back to the data container throughout the game. Given the time interval an analysis can be computed such as average pass rate, rate of successful pass, and most consecutive successful pass, etc.

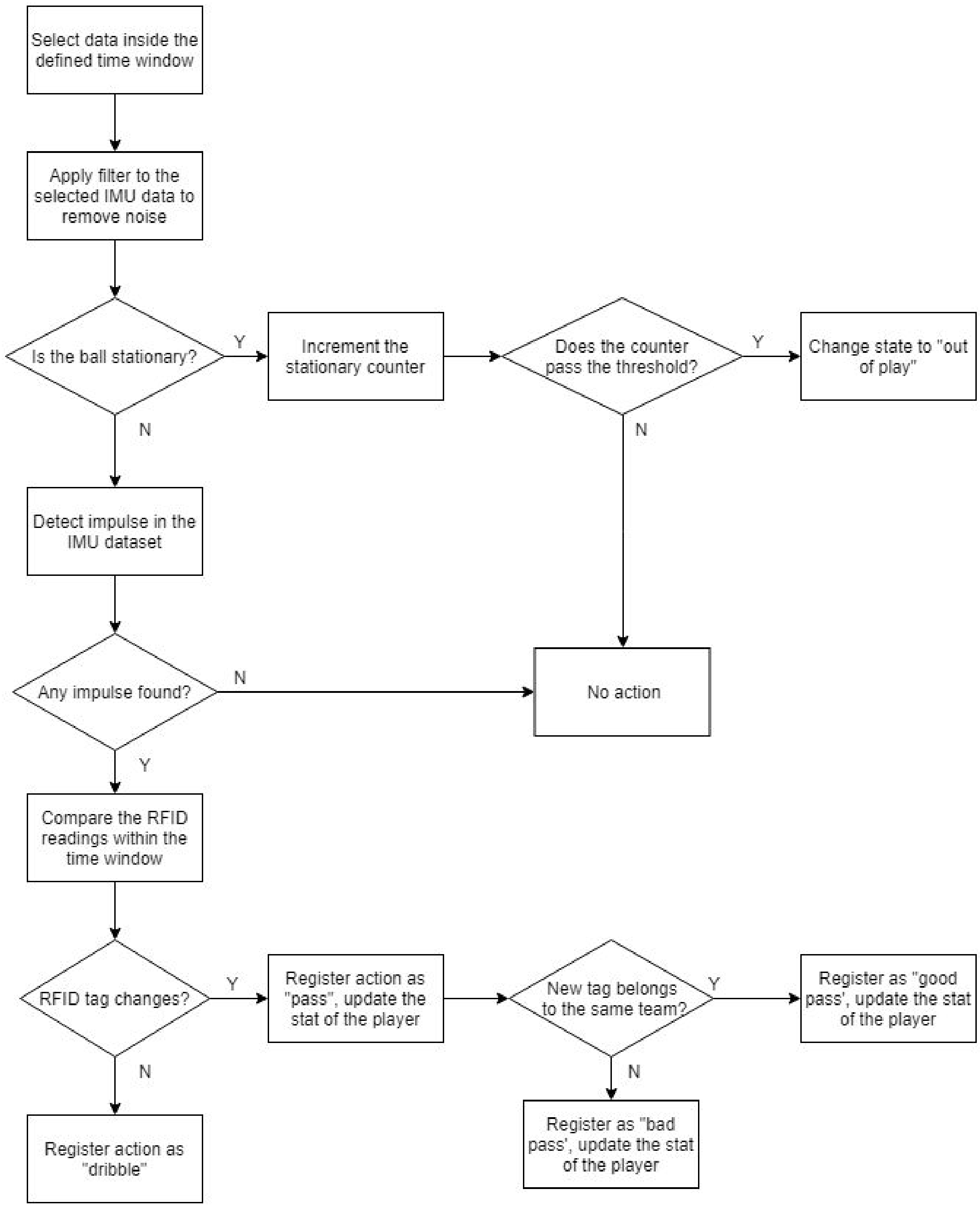


Figure 7. Processing Algorithm per Time Window

2.3.3 Control Unit

- **Microcontroller**
  A microcontroller unit is needed to pull data from the sensor systems and push it to an external data storage system. The ESP32 implements a TCP/IP and full 802.11 b/g/n WIFI MAC protocol and supports up to 150 Mbps of data rate which is sufficient for the data transmission.

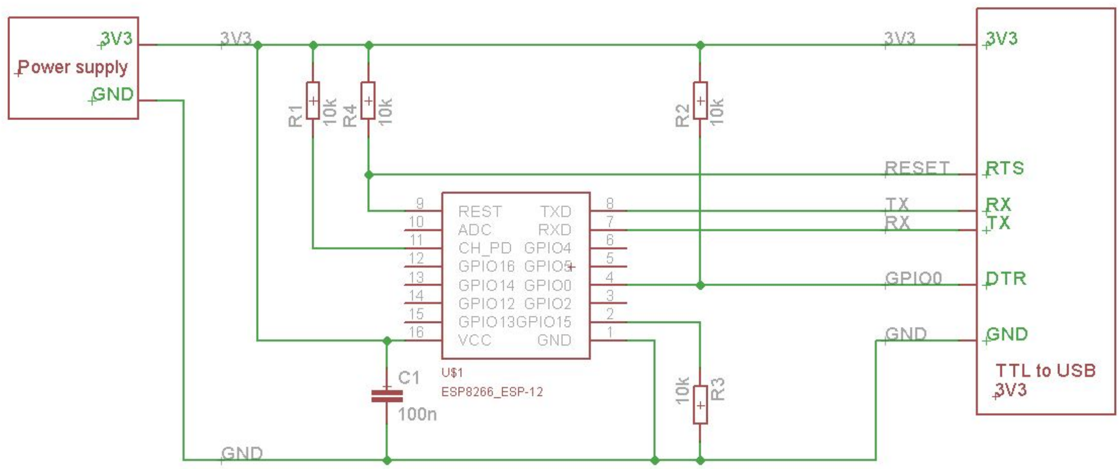


Figure 8. ESP32 Minimal Hardware [13]

Requirements	***Verification***
1. The microcontroller must have enough pins to support the sensor system and run off a 3.7V to 4.2V lithium ion battery. Ideally, it will also have wifi+bluetooth capabilities to utilize cloud storage	1. A. Wire the microcontroller according to figure 8. B. Program a simple blink script and verify that it works.

- **Data Storage and Transmission**
  The receiving server establishes the wireless TCP connection with the device controller via WIFI network. The IMU data will be time stamped and continuously logged throughout the duration of the game. The RFID data will be logged only if the RFID reader has the reading. Otherwise, the RFID data should be empty or meaningless. The RFID data will also be time stamped. The data needs to be stored inside a structured container (either a database or csv format file) so that it can be pulled out and analyzed for the metrics we want.

  Data format:
  RFID data: 16 bytes

STX (02h)	DATA (10 ASCII)	CHECK SUM (2 ASCII)	CR	LF	ETX (03h)

IMU data:

Parameter	Data type	bytes
Accel_Data_X	signed	2
Accel_Data_Y	signed	2
Accel_Data_Z	signed	2

Parameter	Data type	bytes
Gyr_Data_X	signed	2
Gyr_Data_Y	signed	2
Gyr_Data_Z	signed	2

Requirements	***Verification***
1. The storage system must have enough capacity enough to store sensor data recorded every 0.2 seconds over the length of a soccer match (90 minutes).	1. A. Calculate and/or sample the amount of data the sensors generate per minute and multiply it by 90 to get total data size over 90 minutes. B. Verify that the size of the storage is greater than the amount calculated in part A.

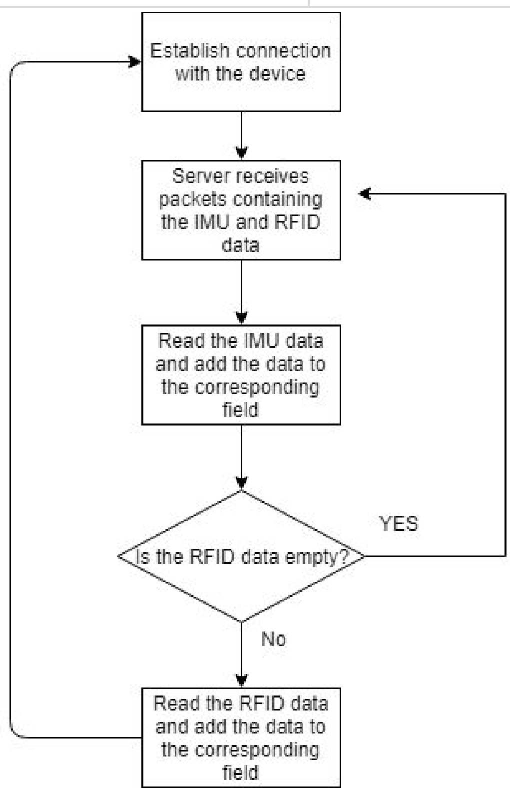


Figure 9. Receiver Actions

2.4 Tolerance Analysis

For our entire system to function properly, it's essential that the power system supplies the other onboard electronic components with the correct voltage and enough current. Our whole system and project is based around the success of receiving the data from the sensors. Without a correctly functioning power supply, the sensors will not be able to read data, and the microcontroller will not be able to send data. Therefore, we believe that the power subsystem is the part of our design that has the highest risk.

From the various datasheets, we note that the BNO055 IMU takes an input voltage of 2.4 - 3.6V, the ESP takes 2.3 - 3.6V, the ID-20LA takes 3 - 5V. Thus, our on-board system must be supplied with a voltage between 3.0 and 3.6 volts. We can calculate the tolerance of our input voltage with equation 1:

(Eq. 1)

$$tolerance = \frac{V\,max - V\,min}{V\,expected} \times 100\%$$

Plugging in the numbers with an expected voltage of 3.3V, we get a tolerance of 18.18%. In other words, our system has a tolerance of $3.3V \pm 9.09\%$. The voltage regulator requirement is to have an output of 3.3V with an error of at most $\pm 5\%$. From the results we derived, we can conclude that variations in the voltage regulator will not affect the functionality of our system.

In the circuit simulation in figure 10, the voltage regulator from figure 4 in loaded with a load varies from 100 mA to 250mA. From the results in figure 11, we can see that the regulator output jumps during changes in load current, from 3.264 - 3.336V. These output voltages are well within our operating range, and the regulator should be able to handle our expected loads.

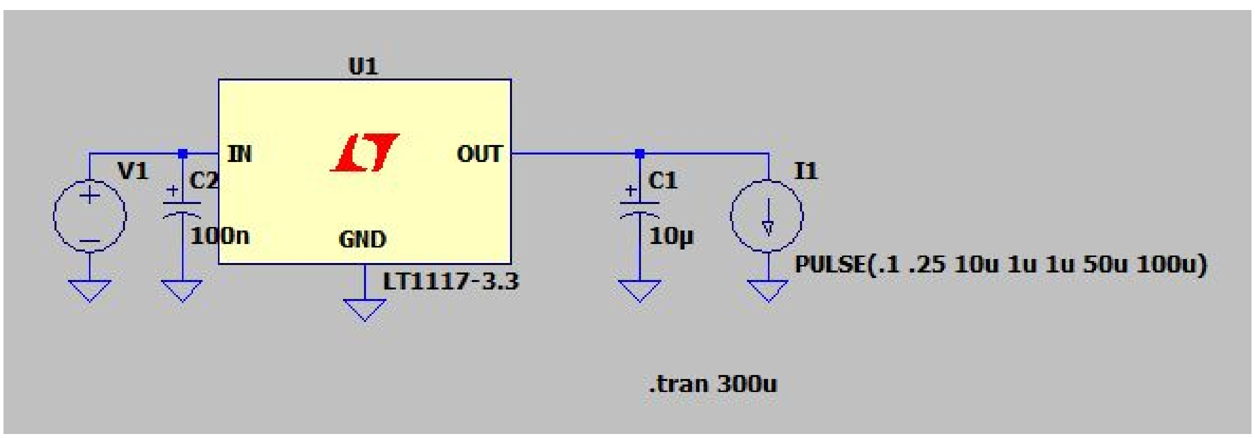


Figure 10. Voltage Regulator Varying Load Circuit

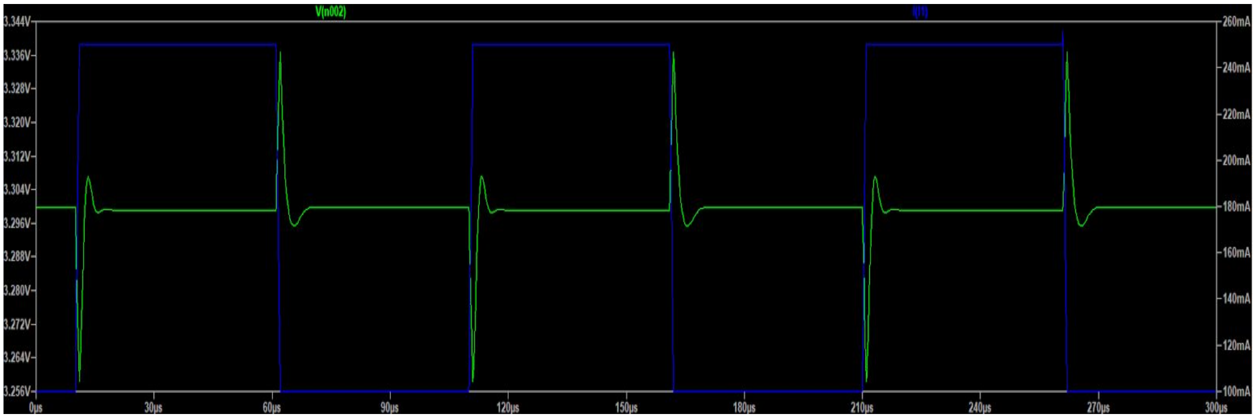


Figure 11. Voltage Regulator Varying Load Waveform

3 Cost and Schedule

3.1 Cost Analysis

We consider the 11 weeks of the semester of work on the project. The estimation for hourly work pay is $40/hour. Considering that for the 11 weeks of work we put in 12 hours/week per person. The total labor costs add up to:

$$3\ (people) \cdot 11\ weeks \cdot 12\ \frac{hours}{week} \cdot 40\ \frac{\$}{hour} = \$15,840$$

Part	***Quantity***	***Cost per Unit***
3.7V Lithium Ion Battery - 1200mAh (Generic, Adafruit)	1	$9.95
Li-Ion Charger (MCP73833, Microchip)	1	$12.50
Voltage Regulator (STMicroelectronics, LD1117V33)	1	$1.25
IMU ACCEL/GYRO/MAG I2C 28LGA (Bosch sensortec, BNO055)	1	$12.10
RFID Reader (ID-Innovations, ID-20LA)	1	$34.95
RFID Tags (Generic, 125 kHz)	10	$1
Microcontroller (Espressif, ESP32)	1	$15
Foam Ball (Generic)	1	$10.90
TOTAL COST		$97.65

3.2 Schedule

Week	***Regis***	***Shixing***	***Yi Rui***
2/18	Finish design document	Finish design document	Finish design document
2/25	Order Parts Begin component testing	Order Parts Begin component testing	Order Parts Begin component testing
3/4	Component verification and testing	Component verification and testing	Component verification and testing

3/11	Building and prototyping	On board software design	Building and prototyping
3/18	Design PCB layouts	Data analysis algorithm development	Design PCB layout
3/25	RFID specific testing/ Make changes to RFID to achieve a as low of a read time as possible ~10ms.	Verify data analysis algorithm correctness with test data	Design CAD model for enclosure 3D print enclosure
4/1	Soldering and assembly	Application interface development	Soldering and assembly
4/8	Begin tests and fixes	Begin tests and fixes	Begin tests and fixes
4/15	Final tests and fixes	Final tests and fixes	Final tests and fixes
4/22	Demo	Demo	Demo

4 Ethics and Safety

4.1 Ethical Issues

The testing and demo of our project involves data collection and processing. IEEE code of ethics, #3 [5] and ACM code of ethics 1.3 [6] address the issues of honesty, and mention that fabricating or falsifying data is strictly prohibited. We promise we will never manipulate data and forge results to make our product look like it works when it doesn't in our development and testing.

The working environment consists of people from different backgrounds with different roles such as our TA, teammates, and instructors. According to IEEE code of ethics #7 and #8 [5], ACM code of ethics 1.4 [6], we should treat everyone equally. We will respect and accept others' criticism and advice, and treat all individuals involved in this project equally and professionally and not engage in acts of discrimination.

In the software development part in this project, we will possibly use libraries and frameworks published by other people or organizations. According to ACM code of ethics 1.5 [6], we should respect and follow the permission of usage and license agreements of any outsourced software involved. We will give proper credit to authors of open source code that we use.

IEEE code of ethics #9 [5] states that we should avoid injuring others, their property. To ensure this, we will evaluate the system stability and potential risk during the development and before testing. Actions will be taken immediately and accordingly when incident occur to prevent or minimize the damage to individuals and surroundings.

4.2 Safety Issues

The main safety risk in our project lies with our usage of a lithium ion battery.

The ball we plan on prototyping with is made of foam, which is flammable. The battery and hardware generates heat during operation, which may lead to overheating and can cause battery failure and fire. To prevent this, we will examine the thermal condition of the system during our development and decide whether to add a cooling design to the project. In addition, we will monitor the temperature inside the foam ball using a temperature sensor during the testing.

The battery may explode or burn when overcharged. During the building process, we will test the charging circuitry and make sure the charging voltage is correct before attaching the battery to the charging circuit.

Since lithium ion batteries have a tendency to catch on fire when damaged or pierced [7], we will try to isolate the hardware as much as possible by 3D printing an enclosure for it. The soccer ball should also have the capability to operate outdoor and moisture could damage the hardware leading to short circuit. The enclosure will help for this problem as well.

References

[1] Forbes.com. (2019). *CHANGING THE GAME: The Rise of Sports Analytics*. [Online] Available at: https://www.forbes.com/sites/leighsteinberg/2015/08/18/changing-the-game-the-rise-of-sports-analytics/#6a8e6b9c4c1f [Accessed 7 Feb. 2019].

[2]Ojs.ub.uni-konstanz.de. (2019). *BASIC MECHANICAL ANALYSIS OF SOCCER BALL IMPACT* [Online] Available at: https://ojs.ub.uni-konstanz.de/cpa/article/viewFile/5637/5130 [Accessed 27 Jan. 2019].

[3]Kickstarter. (2019). *DribbleUp Smart Soccer Ball*. [Online] Available at: https://www.kickstarter.com/projects/dribbleup/dribbleup-smart-soccer-ball [Accessed 5 Feb. 2019].

[4]Digital Trends. (2019). *Adidas Smart Ball Uses Sensors, Bluetooth to Measure Impact* **|** *Digital Trends*. [Online] Available at: https://www.digitaltrends.com/health-fitness/adidas-new-bluetooth-soccer-ball-analyzes-kicks-help-improve-game/ [Accessed 7 Feb. 2019].

[5]Ieee.org. (2019). *IEEE Code of Ethics*. [Online] Available at: https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed 6 Feb. 2019].

[6]Acm.org. (2019). *ACM Code of Ethics and Professional Conduct*. [Online] Available at: https://www.acm.org/code-of-ethics [Accessed 6 Feb. 2019].

[7] Call2Recycle. (2019). *Damaged, Defective and Recalled Batteries.* [Online] Available at: https://www.call2recycle.org/safety/damaged-defective-and-recalled-batteries [Accessed 6 Feb. 2019].

[8] Adafruit. (2019). *Li-Ion and Lipoly Batteries.* [Online] Available at: https://learn.adafruit.com/li-ion-and-lipoly-batteries [Accessed 18 Feb. 2019].

[9] STMicroelectronics, “Adjustable and fixed low drop positive voltage regulator”, LD1117 Datasheet, Rev 33 [Revised Nov. 2013].

[10] Adafruit. (2019). *BNO055 Absolute Orientation Sensor.* [Online] Available at: https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor/overview [Accessed 19 Feb. 2019].

[11] ID-Innovation, “ID-2LA, ID-12LA, ID-20LA Low Voltage Series Reader Modules Datasheet Version1.0 Date 09/01/13”, ID-2/12/20 LA Series Datasheet X1, Sept. 2013 [Revised Nov. 2013].

[12] Bosch Sensortec, “BNO055 Intelligent 9-axis absolute orientation sensor”, BST-BNO055-DS000-14 datasheet, June 2016 [Revision 1.4].

[13] Espressif Systems,“ESP32 Series Datasheet”, ESP32 Datasheet V2.8, Jan 2019.