Soccer Team Gameplay Metrics

By

Regis Lato

Shixing Ma

Yi Rui Zhao

Final Report for ECE 445, Senior Design, Spring 2019

TA: Dongwei Shi

01 May 2019

Project No. 54

Abstract

Soccer Gameplay Metric is a project aimed at gathering different metrics from a soccer gameplay automatically and wirelessly from a soccer ready to be viewed live as the gameplay goes on into a user interface. The system is composed of different sensors embedded into a soccer ball like an RFID reader, accelerometer and gyroscope. An RFID tag is embedded in the players' shoes which will be unique to each player so that every time a player touches the ball it can be sensed by the RFID reader. The accelerometer and gyroscope work in conjunction with the reader to eliminate false positives and provide accurate readings. The accelerometer and gyroscope also provide data to be able to calculate acceleration and spin of ball. Our findings show that this concept is possible and with some future work it could be designed to be a consumer ready product.

Contents

1 Introduction

1.1 Purpose

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, so it would be more effective to automate the entire process instead.

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.

We built a system that is able to measure and calculate metrics for individual players over the duration of a soccer game. For this project, we focused on metrics such as touches of the ball per player, good passes, bad passes, possession time and power of each touch. Countless more metrics can be derived just by the data collected by the sensors used in this project. We collected data using a sensor system integrated into a ball, and present it to the user on an application after analyzing it.

1.2 Functionality

- Each touch of the ball by an individual player is registered and captured by the system, and so the actions of different players are distinguished from each other and players' performance metrics can be calculated.
- The system is wireless and works on battery power for over 10 hours.
- The application is able to process the data and display the metrics such as passes between players, bad passes, good passes, each touch of the ball by each player and time of possession.

1.3 Subsystem Overview

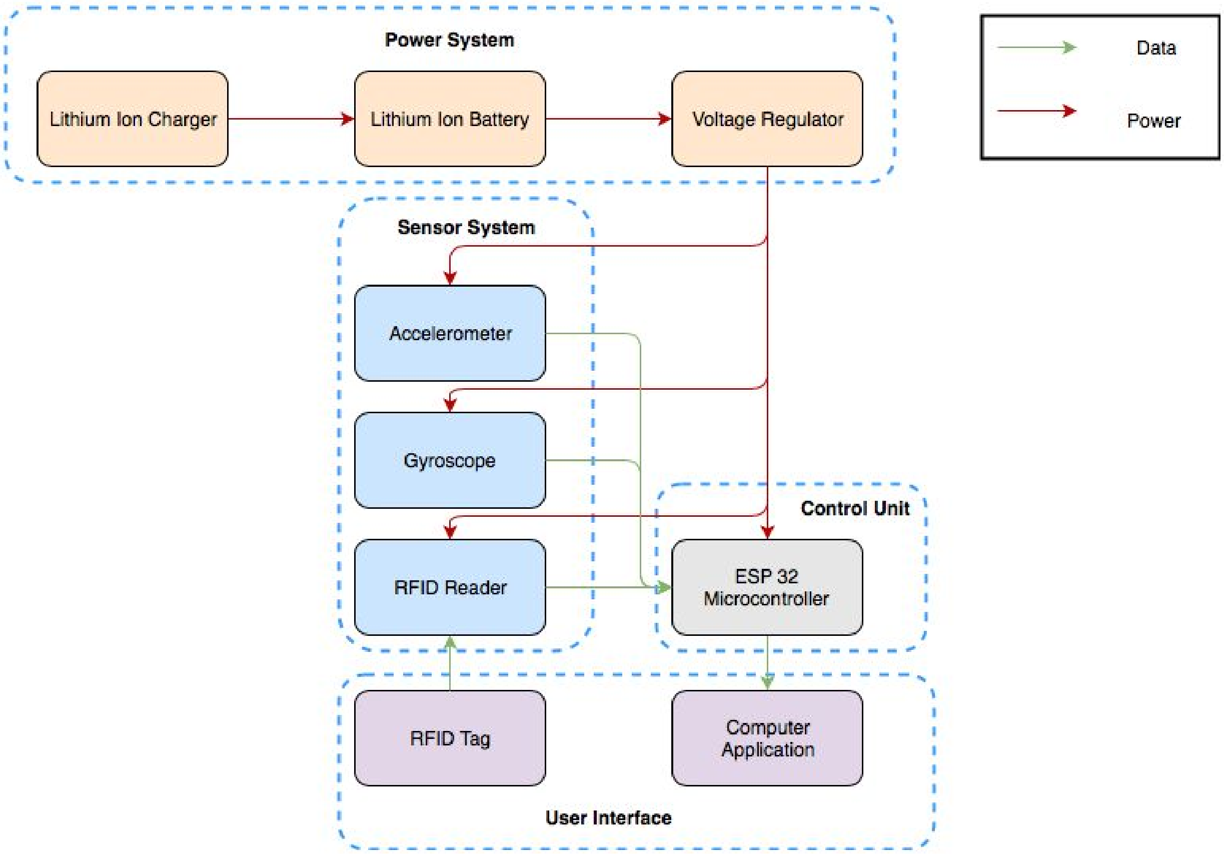


Figure 1. Top-Level Block Diagram

As shown in Figure 1 above, the overall system is divided into four sub-subsystems: Power System, Sensor System, Control Unit and User Interface.

The power system consists of a rechargeable Lithium ion battery, a voltage regulator, and a charger. The battery is the only power source of the whole system. The voltage regulator is used to regulate the output power of the battery and so it can meet the power requirements of other components in the system. The charger of the battery is included because it's important to only charge the battery under the safe condition.

The sensor system contains a inertial measurement unit (IMU), which integrates accelerometer and gyroscope, and an RFID reader. The IMU is used to measure the motion of the soccer ball and the RFID reader reads the RFID tags of the players, which is used to register player's touches of the ball.
Control unit contains the microcontroller and the on-ship software running on that. The microcontroller is responsible for communication between the sensor system and the computer

application, including collecting data from the sensors and transmit those data through wifi to the processing application.

Lastly, the user interface contains the RFID tag and the computer application. The RFID tags are associated with the player roster information such as the players' identities and the team they are on. The application is responsible for all the data processing and analysis, and displaying the result.

1.4 Physical Design

We integrated the power system, sensor system, and control unit inside a high density foam soccer ball. Thin RFID tags attached to sticker paper are 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 that are up to par with FIFA regulations for a match ready ball, so we will not be focusing on that aspect of the design.

2. Design

2.1 Power System

The power system powers all the electronics that go inside the ball. It is able to support the power requirements of the IMU, RFID reader, and microcontroller. The voltage required for sensors is similar, so the system is 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 3 to 5 volts. Therefore we built our power system to have a nominal 3.3V output. The only exception is the RFID reader that gets 3.7V directly from the battery.

2.1.1 Lithium Ion Battery

The soccer ball cannot be wired in during a game, so we use a portable power source to power the electronics inside the ball.

We decided to use a rechargeable lithium ion battery because of the relatively high power consumption of our system. Using a primary cell battery that would need to be thrown out after a few hours of use would be inefficient and wasteful.

We chose the battery based on two categories: output voltage and capacity.
The system runs on 3.3V, so a 3.7V battery output is optimal.
We calculated the capacity required using equation 1. The system draws a current of 100 mA on average. Therefore theoretically we should get around 12 hours of usage out of our 1200 mAh battery.

$$Battery\ Life\ (hours) = \frac{C\ (mAh)}{I\ (mA)} \tag{1}$$

2.1.2 Battery Charging IC

Since lithium ion batteries are prone to catching on fire when supplied with the wrong voltage or current, a charging IC is required to control the voltage and current to safely charge the battery.

We chose the MCP73833 Linear Li-Ion / Li-Polymer Charge Management Controller from Microchip which is able to charge a battery to 4.2V with a maximum current of 1A. We designed a board to with the IC to charge the battery using a 5V USB power source. This way, there’s no need to reopen the ball between usage sessions.

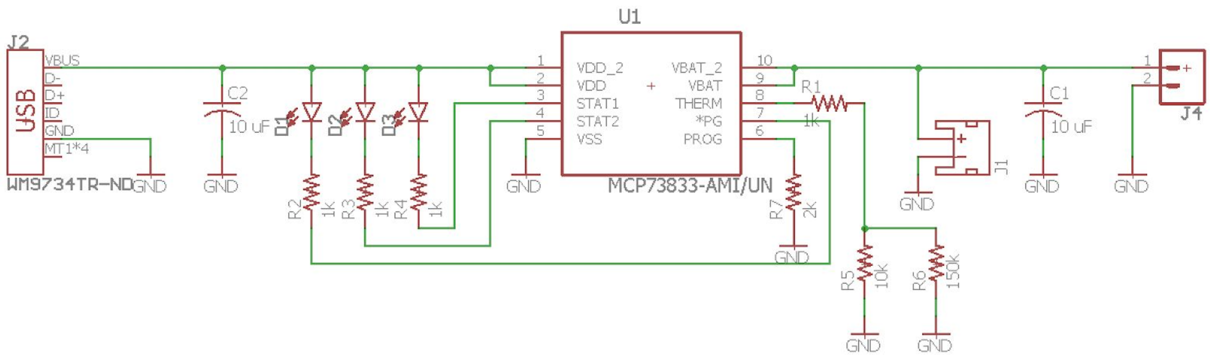


Figure 2. Charging Circuit Schematic

2.1.3 Voltage Regulator

The sensors and microcontroller in our system require a stable voltage input to operate correctly. The voltage of a lithium ion battery varies depending on how much charge is left, so it's not ideal to have the battery directly powering the system. In addition, the battery we're using outputs up to 4.2 volts at max charge, while the maximum input voltage of the ESP32 is 3.6V, so a voltage regulator is required.

The AP2112-3.3 is low dropout linear voltage regulator meets our preliminary requirements. It outputs a fixed voltage at 3.3V with a maximum current of 600 mA. Two 1 uF decoupling capacitors are required at the input and output to ensure stable operation.

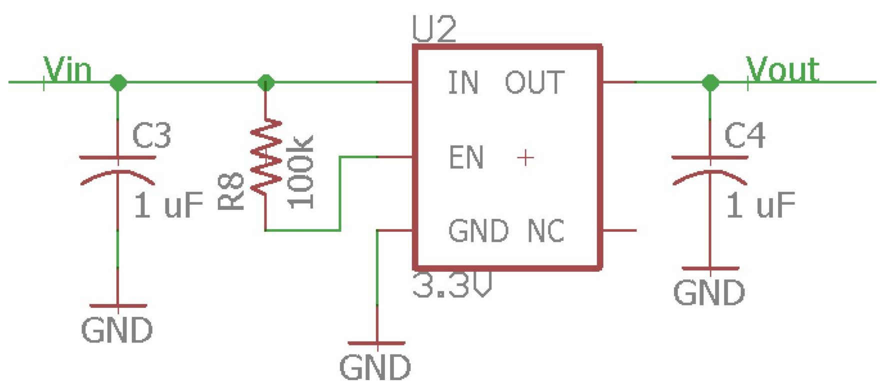


Figure 3. Voltage Regulator Circuit Layout

2.2 Sensor System

2.2.1 IMU (Accelerometer and Gyroscope)

The inertial measurement unit (IMU) is an IC unit that has three different sensors integrated inside: an accelerometer, a gyroscope, and a geomagnetic sensor. We determined that it would be more suitable to IMU instead of separate sensor modules due to its compactness and simplicity.

The accelerometer provides data on the acceleration that the ball experiences in the x, y, and z directions. In our system, we mainly use the accelerometer to detect peaks that correspond to kicks or bounces. The accelerometer has to detect up to ±4g of acceleration for high power kicks. The gyroscope provides data on the spin and orientation of the ball. This enables 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.

The BNO055 IMU sensor used on this project can measure up to ±16g of acceleration and 2000 degrees of spin per second so it is well suited for the specified requirement.

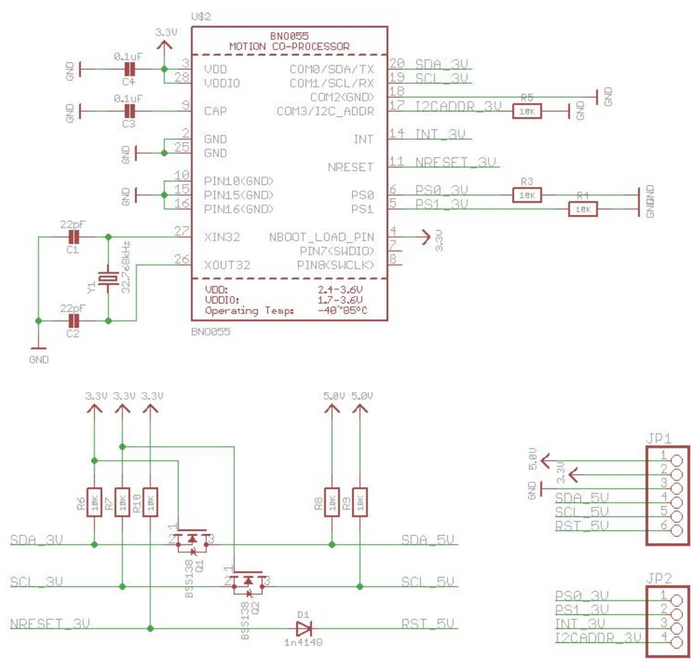


Figure 4. IMU Circuit Layout [10]

2.2.2 RFID Reader

The RFID reader is our primary way of tracking the person that is currently in contact with the ball. When the player kicks the ball, the ID is read from the RFID tag on the player’s shoes. Each tag ID is mapped to a specific player on the field in the software, and multiple tags can be used to increase the number of points of contact on the player.

Read speed is an important aspect of the RFID reader. If the read speed is too slow, we won’t be able to detect fast kick with low contact times. Since read speed is intrinsic to the hardware, we can only alleviate this issue by increasing the available read time.

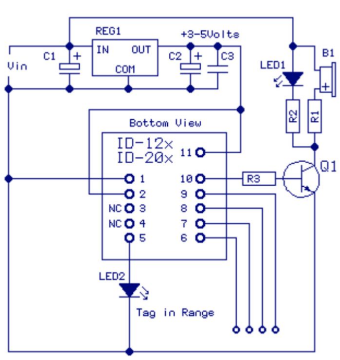
To increase available read time, we have to increase the in-range travel distance. As the RFID tag enters the read range, we get extra time to read the tag as it travels to and away from the ball. This relation is shown in equation 2, where Rt is available read time, V is velocity of the ball, ro is the in-range travel distance, and Ct is the kick contact time.

$$R_t = \frac{1}{V} \cdot r_o \, x \, 2 + C_t \tag{2}$$

In general there are two ways to increase the in-range travel distance: decrease the size of the ball, or increase the read range of the RFID reader. Equation 3 (derived from Friis transmission equations) shows that RFID read range is a function of the transmitted power Pt. By increasing the input voltage of the reader, we increase the read range in a linear manner.

$$R_{\max} = \frac{\lambda}{4\pi}\sqrt{\frac{P_t G_t G_r}{P_{\min}}} \tag{3}$$

When we tested out the RFID reader, we measured a read range of 10.4 cm at 5V, which is almost of the desired 18 cm stated on the datasheet. We had to make changes to our design to increase the effective range. We reduced the size of our ball from a radius of 10 cm to a radius of 7.5 cm. We also had to move voltage input of the RFID reader from the 3.3V output of the regulator to the 3.7V output of the battery.



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 5. ID-20LA Circuit Layout [11]

2.3 Control Unit

2.3.1 Microcontroller

A microcontroller is necessary to pull data from the sensor systems and push it to an external storage system. The microcontroller communicates via I2C with the IMU and via serial with the RFID reader to get data.

The ESP32 is chosen because of its WIFI and Bluetooth capabilities. This way, we are able continuously send data and update the metrics in real time. It 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.

A large capacitor is placed near the input power pin for voltage smoothing. The current draw of the ESP32 can spike up to 400 mA during WIFI operation, so without a capacitor, brownouts will occur. The RST and EN pins need to be toggled by the two buttons to put the ESP32 into flashing mode for programming.

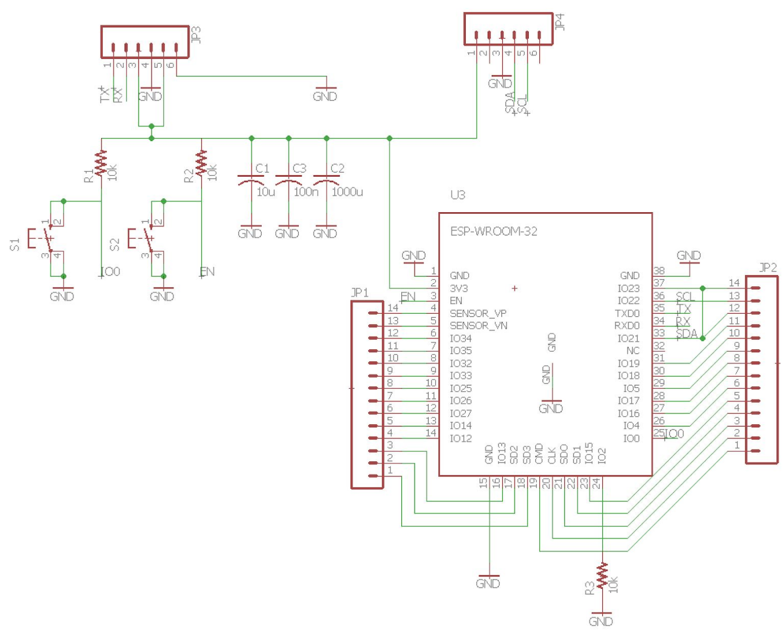


Figure 6. ESP32 Minimal Hardware

2.3.2 On-Chip software

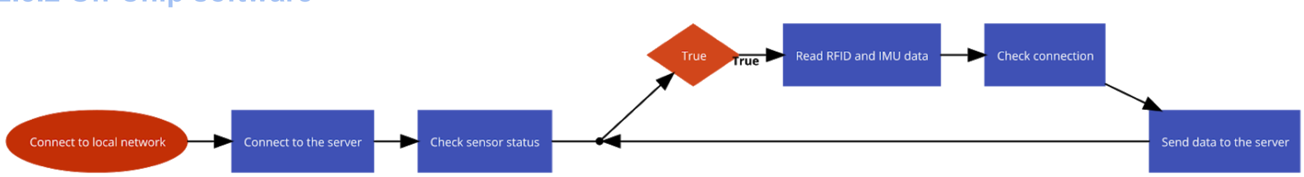


Figure 7. On-chip Software Flowchart

The software running on the ESP32 microcontroller has several requirements: collecting data generated by the sensors, connecting the device to the server, and sending the data to the server via WIFI network. The program is designed using Arduino and some libraries including some hardware drivers provided by the hardware companies.

The general procedure of the on-chip software is shown in the flowchart above. In practice, the IMU data is a collection of 8 bytes long double precision float numbers in the x,y,z directions and the RFID data is a byte string of 5 bytes. In the program, the two types of data are packed into a 29 bytes long data structure and sent out together. The sample/sending rate is about 10*29 = 290 bytes/sec which is far less than most of today's wireless network's bandwidth limit.

2.4 User Interface

2.4.1 RFID Tag

Each player will has an unique RFID tag attached to their cleats to be identified with. In our initial design, we wanted to use RFID tags that were as small and thin as possible so that it hardly affects the player and can be easily attached inside or on the cleat. However, we found out that in exchange, we would be sacrificing read range, which was an issue during testing. Consequently, we had to trade them out for ISO cards that were larger, but gave us a longer read range.

The RFID data on each tag is associated with the player's information and the correspondence is stored inside the database.

2.4.2 Computer Application

We wrote an application that processed and visualized the data for each player.

The application has two parts: the receiver and the data processing program. The two programs communicate via a local file I/O. The reason to divide the two functionality is to reduce the effect of delay caused by processing the data. The application is written in Python and is running on the server machine.

The receiver's job is receiving the data sent from the microcontroller and preprocessing the data for the ease of further analysis and storing the data on local disk. The data received is 29 bytes long raw byte string and is needed to be unpacked into the corresponding fields and converted to usable data types. Finally, the preprocessed data is time stamped and written to a local csv file to be read by the processing program.

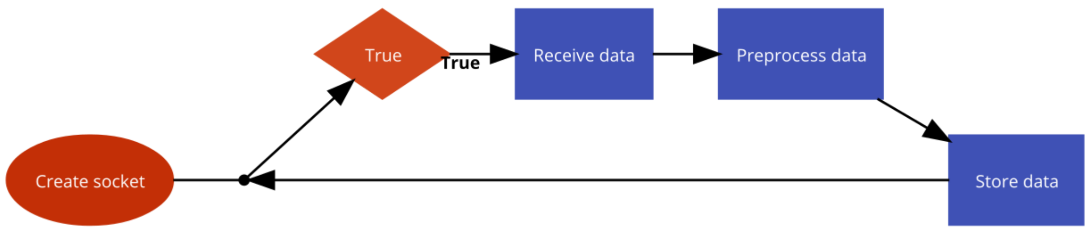


Figure 8. Receiver actions

In the processing program there are two major implementations: bounce detection and player metrics calculation.

The bounce detection algorithm detects the bounce of of the soccer ball, either kicked by the player or bounced by other objects, using solely IMU data. It's done by detecting the peak of the IMU data. This feature can be further developed by correlating with the RFID data to distinguish the player's kick from the object bounce.

The player's metrics includes number of touches, possession time, total passes count and good passes count. The metrics are calculated using the RFID data. During the processing the player's information such as identity and the team he/she played for is looked up in the database and the program uses those information to update the player's metric.

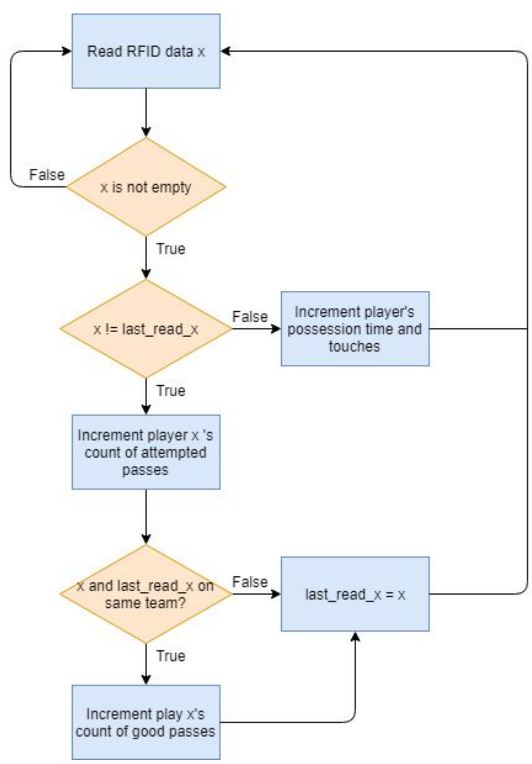


Figure 9. Player metrics calculation

Finally, the calculated player's metrics are displayed using a histogram.

3. Design Verification

3.1 Power System

3.1.1 Battery

The battery must be able to sustain up to the current draw of the system at 3.3 volts for the length of a soccer match length (90 minutes). This is to ensure that we have comprehensive data for the entire match and no delays during mid-game.

We put together the system and powered it using an external power source at 3.7V. The current draw of the system with the software running was measured to be 100 mA on average. Using equation 1, we calculated that with the measured power consumption of the system, we will be to get about 12 hours of continuous usage from it, which is much more than the requirement of 90 minutes.

3.1.2 Battery Charging IC

The charging IC has to be able to fully charge the battery in a reasonable amount of time from a 5V USB power source. In our design review, we determined that this would mean that the IC would need to output 4.2V with a minimum current of 100 mA.

We discharged the battery to 3.5V at a current of 100 mA (less than 0.2C), which means that the battery has less than 10% charge left according to figure 10. We plugged the battery and a 5V power source to the charging board shown in figure 2, and measured a voltage of 4.2V from the output of the MCP73833 with a current of 500 mA. This is consistent with the total charge time of 2 hours and 10 minutes to charge ~90% of a 1200 mAh battery.

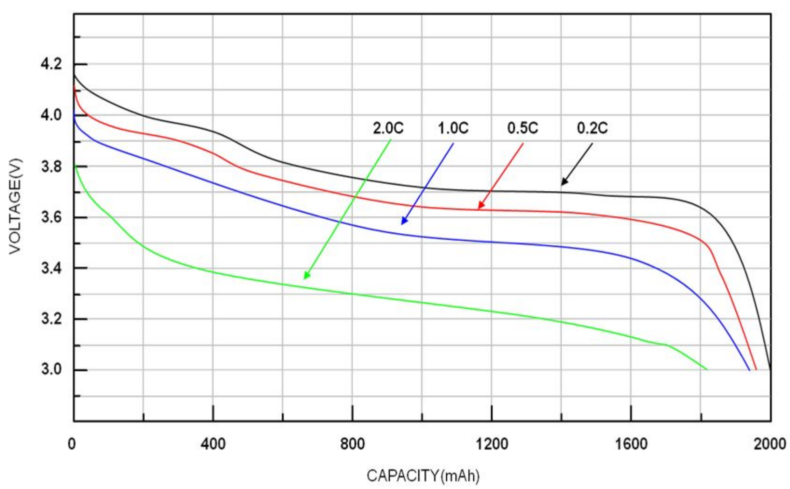


Figure 10. Discharge Curve for a 2000 mAh Battery

3.1.3 Voltage Regulator

The regulator needs to be able to convert a 3.7 to 4.2 input voltage to 3.3V and be usable under the current draw of the system (100 mA). The battery outputs 3.7 to 4.2 volts depending on the remaining charge left, and our system uses 3.3V.

We built the board as shown in figure 3 and connected the power supply to Vin and our system (100 mA load) to Vout. Figure 11 shows the voltage at Vout as we swept Vin from 0 to 5 volts. Between an input of 3.7V and 4.2V, the output remained very close to 3.3V (±0.01%).

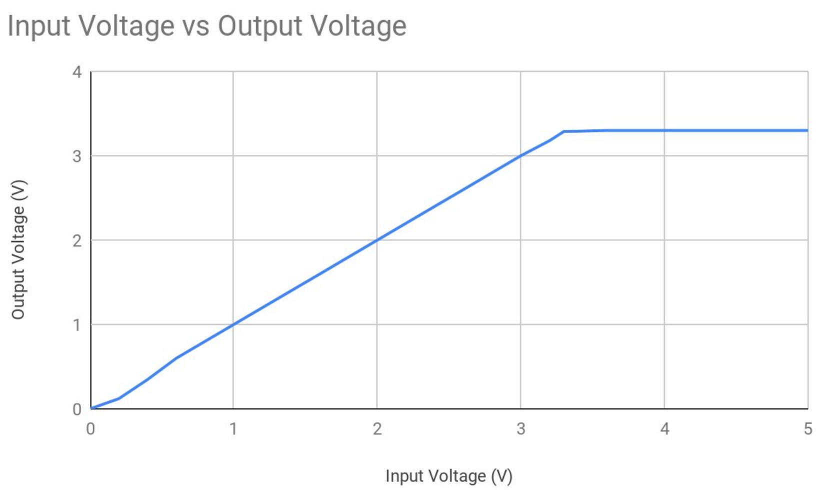


Figure 11. Input vs Output Voltage of Regulator Under 100 mA Load

3.2 Sensor System

3.2.1 IMU (Accelerometer and Gyroscope)

The accelerometer has to detect up to ±4g of acceleration for high power kicks, and the gyroscope has to detect up to 2000 degrees/second for fast spin kicks.

To test accuracy, we looked at the gravity vector of the accelerometer in different orientations. We verified that the gravity vector pointed in the expected x, y, z directions. If gravity is positive in the z direction, we made sure that the vector is negative in the z direction after we flipped it over. We verified the gyroscope accuracy by attaching it to an iPhone XR and comparing the data from the phone and the IMU gyroscope.

We then tested the detection limit of the accelerometer by placing it inside the foam ball and kicking the ball around as hard as possible. We measured a maximum acceleration of 50 m/s^2 from one of our kicks, which corresponds to over 5gs of acceleration. Figure 12 shows the magnitude of acceleration from the kicks.

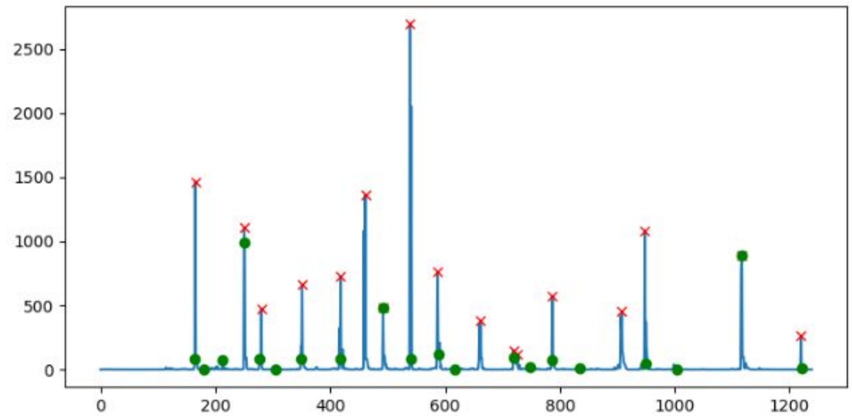


Figure 12. Accelerometer, Kicking Power over Time

3.2.2 RFID Reader

The RFID reader must at least have a detection distance of the radius of the ball (~7.5 cm) and the read time should be faster than the smallest contact time of a typical kick (~15 ms) to ensure all touches of the ball are detected.

We characterized the RFID read range as a function of input voltage. We connected the reader to a power supply and measured the maximum read distance as the input voltage is swept from 2.5 to 5 volts.

Our results show that read range linearly increases with voltage, which is consistent with equation 3. The read range is 9.4 cm at 3.7V, which is larger than the radius of the ball. From equation 2, we calculate an additional available read time of 10ms, which puts the effective read time of the RFID reader (20ms - 10ms = 10ms) below the smallest contact time of a typical kick.

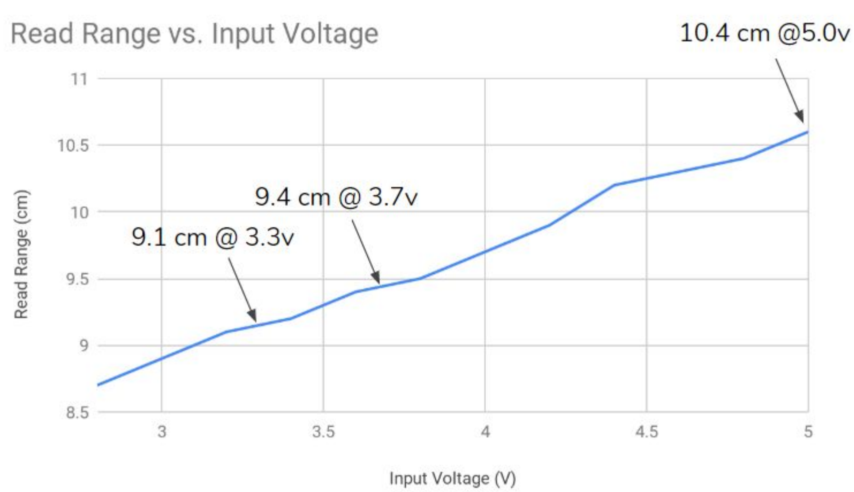


Figure 13. RFID Read Range vs. Input Voltage

4. Cost and Schedule

4.1 Parts

Table 1. Parts List

Part	Manufacturer	Retail Cost	Bulk Cost	Actual Cost
3.7V Lithium Ion Battery - 1200mAh	Generic	1	$9.95	$9.95
Li-Ion Charger (MCP73833)	Microchip	1	$0.67	$0.66
Voltage Regulator (AP2112-3.3)	Diodes Incorporated	1	$0.13	$0.49
IMU ACCEL/GYRO/MAG I2C (BNO055)	Bosch sensortec	1	$5.92	$12.10
RFID Reader (ID-20LA)	ID-Innovations	1	$34.95	$34.95
RFID Tags (125 kHz)	Generic	10	$0.20	$1
Microcontroller (ESP32)	Espressif	1	$3.80	$3.80
Foam Ball (Generic)	Generic	1	$10.90	$10.90
TOTAL				$82.85

4.2 Labor

We worked 12 weeks of the semester on the project. The estimation for ideal hourly work pay is $45/hour. Considering that for the 12 weeks of work, we put in, on average 2 hours a day, thus 14 hours/week per person. This means in total the amount of time we spent is 168 hours per person on the project. The total labor costs add up to:

$$3\ (people) \cdot 168\ hours\ \cdot 45\ \frac{\$}{hour}\ \cdot\ 2.5 =\ \$56,700$$

4.3 Schedule

Table 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	Establish communication with ESP32	Establish wifi connection microcontroller	Design PCB layout
3/18	Make ESP32 communicate with IMU sensor through I2C	Work on on-board software	Design PCB layout
3/25	Work on RFID communication and data transfer through serial to ESP32	Work on initial data transfer through wifi from microcontroller side to server side	Soldering and assembly, initial testing of parts and components
4/1	Soldering and assembly, testing of sensors combined on dev board	Complete data wifi data transfer of the sensors	Second round improved PCB layout design
4/8	Soldering and assembly, work on physical design and untethered testing of the system	Start working on algorithms to interpret the received data on server side	Soldering and assembly, Test and verify all connections on PCB boards
4/15	Begin final tests and fixes and troubleshooting of the whole system	Finish working on algorithms and displaying the data on a visual user interface	Begin final tests and fixes and troubleshooting of the whole system
4/22	Demo	Demo	Demo

5. Conclusion

5.1 Accomplishments

One of the biggest challenges was to make it possible for the reader to detect the tag when a fast moving ball hit a player's shoe. Using a read range that exceeded the ball radius resulted into making this possible by giving the RFID reader more time to read the RFID tag. The RFID reader works in conjunction with the IMU sensor to eliminate any false positives and give accurate readings for every touch of the ball. Achieving this was one of the biggest accomplishments of this project as it was the biggest requirement to prove this concept on a working prototype.

5.2 Uncertainties

The main issue that we faced while working on this project was when we tested the RFID reader and the results of the real life test data did not match the ones on the data sheet. The problem was the read range of the RFID reader. The data sheet of the ID-20LA RFID reader tells us that the read range at 5V for the RFID reader is supposed to be 18 - 20 cm and depending on the type of RFID tag used it could be up to 24 cm. This would be ideal for our project. But after testing the RFID reader we only got a maximum read range of 10.4 cm. This about half of what the data sheet said. This read range would not work with our initial design. To counteract this huge lack of a read range we decided to use a smaller ball than initially decided. We used a ball of 7.62 cm in radius with was enough to make our project be able to prove the concept in a working prototype.

5.3 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.

The main safety risk in our project lies with our usage of a lithium ion battery.
The ball we used is made of foam, which is flammable. The battery and hardware generates heat during operation, which could lead to overheating and cause battery failure and fire. To prevent this, we will monitored the temperature inside the foam ball and made sure that nothing was running too hot.
The battery may explode or burn when overcharged. During the building process, we tested the charging circuitry and made 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 tried to isolate the hardware as much as possible by layering pieces of foam in between. This way, there is some leeway for the ball to be compressed without damaging anything inside.

5.4 Future Work

The prospective of this project is to ultimately be a consumer ready product that would consist of a soccer ball and tags that can be easily embedded in soccer cleats. To get to that we would first have to take into consideration the challenges that we faced in building our current prototype. The RFID reader would require a bigger antenna than the one currently used to provide a bigger read range for a size 5 soccer ball which is 11 cm in radius. We would aim for a RFID read range of about 20 cm. This would make possible for very accurate tag reads.

Miniaturizing the electronics is then the next step so that the weight of the electronics does not affect the weight of the ball. Adidas is able to manufacture a ball with electronics inside and the ball is within FIFA regulations for an official match ball which means that it feels and performs exactly like any other official match ball. A relationship with Adidas or similar company would make possible for this proof of concept to make the transition to a consumer product.

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.

Appendix A   Requirement and Verification

Table 3. RV Table

Part	***Requirements***	***Verification***	***Results (Y/N)***
Battery	The battery must be able to sustain up to 100 milliamps of current draw at 3.3 volts for the length of a soccer match length (90 minutes)	A. Fully charge the lithium ion battery to 4.2V. B. Connect it to a load that draws 100mA of current. C. Leave the load connected for 90 minutes. D. Disconnect it from the load and use a voltmeter to check that the voltage is above 3.7V.	Y
Charger	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.	A. Discharge the lithium ion battery to 3.3V. B. Wire the charging module. C. Connect the charger to a 5V USB source with the battery connected. D. 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.	Y
Voltage Regulator	Able to step down battery voltage to 3.3V ± 5% while maintaining a 100 mA current.	A. Connect the regulator input to a 5V lab supply and wire the circuit as shown in figure 3. B. Connect the output of the regulator to a 100 mA load. C. Verify that the voltage is within 5% of 3.3V using a voltmeter.	Y
Accelerometer	*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 degrees per second in 3-axis.	We will compare the accelerometer and gyroscope to an already working accelerometer sensor on iPhone XR. A. Wire the IMU 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	Y
RFID Reader	Must at least have a detection distance of the radius of a soccer ball (~7.5 cm). Read time should be faster than the smallest contact time of a typical kick (~15 ms)	Place the RFID tag at a distance of 10cm and check if the sensor can read it. Move the RFID tag in the field of the RFID reader at a speed that would make it in contact for about 15ms and verify if there is a read.	Y
RFID Tag	Each tag should have a unique identifier/data string.	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.	Y
Application	It must be able to correctly compute and display stats in a table/list/graph.	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	Y
Microcontroller	The microcontroller must have enough pins to support the sensor system and have wifi+bluetooth capabilities to utilize cloud storage	A. Wire the microcontroller according to figure 6. B. Program a simple blink script and verify that it works.	Y