Smart Sports Scoreboard

ECE 445 Design Document

Michael Manning, Max Mitchell, and Matthew Rosenbaum

Group 49

TA: Daniel Vargas

2/26/2021

1 Introduction

1.1 Objective

Smartphones have become a central part of our society in America over the past decade. As of last year, 96% of Americans owned a cell phone, and 81% of the population owned a smartphone [1]. As a result, we have become nearly dependent on using smartphones to obtain information, stay connected to social circles, and communicate. As of 2019, people have been found to check their phones an average of 96 times a day, a number that has increased by 20% over two years prior [2]. Among these occurrences, sports apps are among the most common - a Capgemini study found that 69% of sports fans utilize smartphone technology to enhance their sports following [3]. Although, this great ability has brought with it a few negatives. The first finding is that adults are making less and less eye contact, between only 30% and 60% of a whole conversation. Yet, between 60% and 70% is needed to establish an emotional connection [4]. These findings also assert that in this smartphone era, it has become culturally acceptable to check a sports score during a dinner. It has also been noted that checking a smartphone while with others gives them an impression that the phone is more important. It also gives off a feeling of rejection to the others in the group [5]. On top of the social issues, constantly checking a phone, for even a tenth of a second, can damper one's ability to focus. An average of 23 minutes are required to become fully focused again [6]. Unless a fan is watching their team play on television, the only way to follow a score is to constantly refresh an app or website.

Our team strives to provide fans with a fast and real-time sports experience without use of a single screen. In order to do this, we will develop a wall-mounted LED display, interfaced with Bluetooth and Wi-Fi, that allows users to specify their favorite sports teams and display information about the team's ongoing games. The device will use a Wi-Fi chip to connect to the home's internet and scrape the score and game information from the web. There will also be a Bluetooth application that allows users to easily choose their favorite teams as soon as the scoreboard system is taken out of the box.

1.2 Background

Many media companies, such as ESPN, Yahoo!, and FOX, have developed their own sports apps within the last 10 years. These apps allow the user to check the scores or even stream the game directly within the given app. As stated according to [3], 69% of sports followers utilize these platforms to follow their teams. The ESPN app itself has 70 million downloads and 2 million daily active users [7]. Although, outside of apps, websites, and television, there are no viable products on the market that enable a user to follow their teams sans the use of a screen.

With the ability to scrape the web or utilize an API to grab sports scores, there is an excellent opportunity to give fans a new method for following their favorite teams and scores. We plan to test our system modularly, checking that information from an actual sports game is being delivered. Accomplishing these feats will give every sports fan a unique and high-quality fandom experience without needing to pick up their phone.
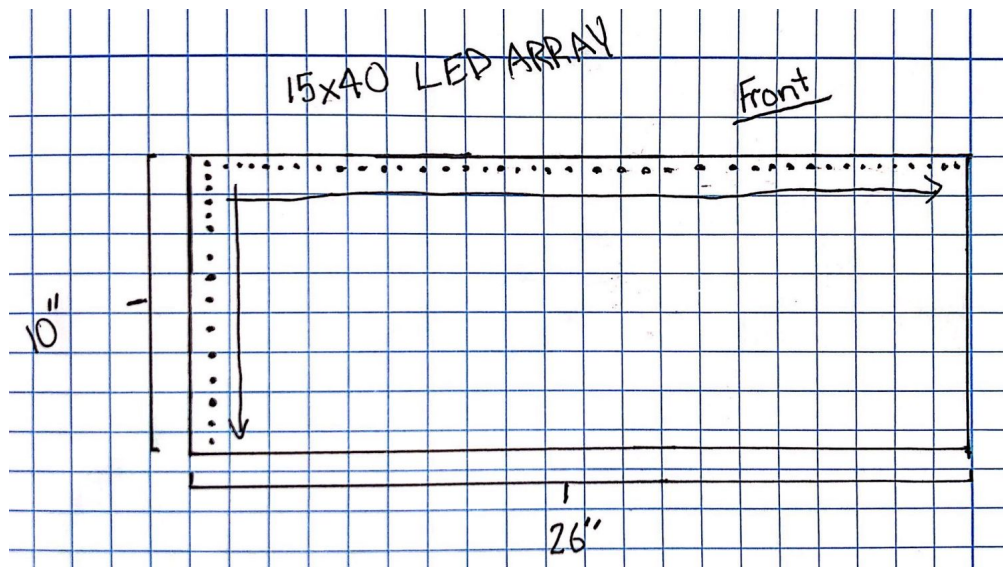
1.3 Physical Design
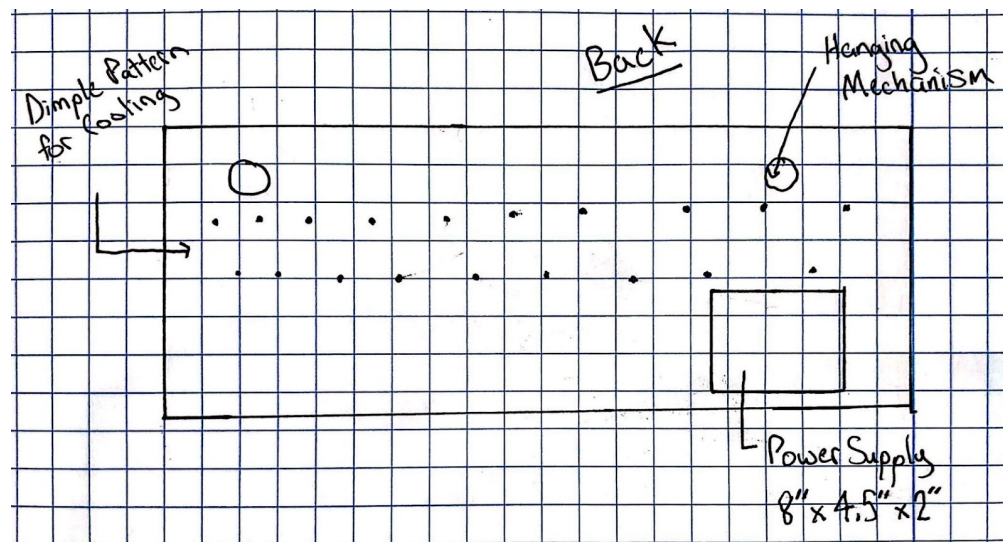

Fig. 1. Physical Design of Front of Scoreboard
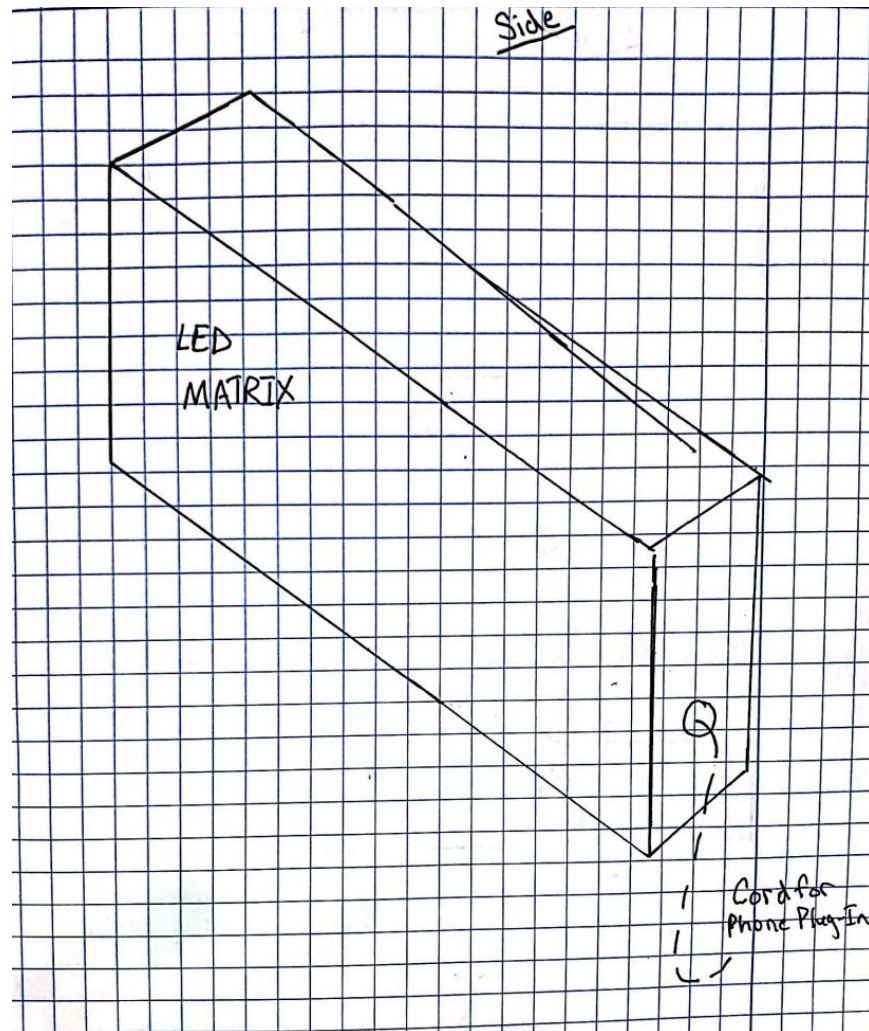

Fig. 2. Physical Design of Back of Scoreboard

Fig. 3. Physical Design of Side of Scoreboard

After talking with the machine shop, the physical design of our scoreboard is going to be constructed of wood. Our LED Display is going to have dimensions of 26"x10" and there will be a 1" wood barrier on each side of the scoreboard to allow for access to the LED's in order to replace or perform work on them. The power supply we plan on using for the device has dimensions of 8"x4.5"x2" and is going to be connected through the back of the device. All components within the device are going to be accessible through the back. The device is going to be hung with angle brackets, or another mechanism that may work more properly. With the current design we will have a workspace inside of the scoreboard with dimensions 3 ¾"x10"x26". Within these dimensions we will have to store the microphone, the PCB, and other components. The machine shop stated that they need to visualize our parts before they can come up with a definitive drawing of how the device will look.
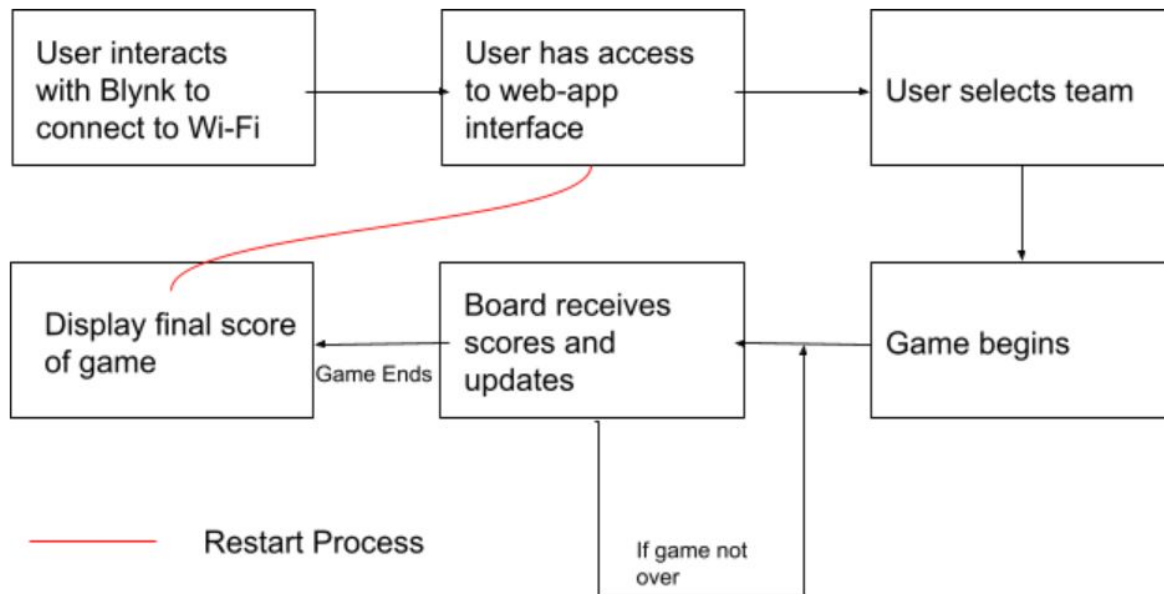
1.4 User Interface Workflow

Fig. 4. User Interface Workflow Diagram

Above is a diagram for the user experience. This is the ideal workflow for the user and how the device will operate starting from the first time they interact with it until the restart of the process.

1.5 High-Level Requirements

- The scoreboard system must be fast, accurate, and reliable. This means that it operates at a latency no greater than 4-6ms with a 5 GHz band, and it displays the correct score of the game at run-time.

- The scoreboard system must operate at 15 amps to fit standard electrical outlets in the U.S. The Bluetooth LE protocol must transmit data at a speed of at least 1 Mb/s.

- The scoreboard system must have an intuitive UI that enables the user to select their favorite team and customize LED interfaces using team themes.

2 Design

In our design, the ESP32 acts as the brains of the design, enabling a web interface through Wi-Fi, a Bluetooth connection to send over Wi-Fi password information, enough processing power to drive our LEDs and react to noise from the microphone in real time, and a small form factor to fit within our enclosure. Our power system is sufficient enough to power all of our LEDs and the sensors. Our enclosure, while not displayed in the block diagram, is another crucial component. It will have to look clean while also being light enough to be mounted on a wall.
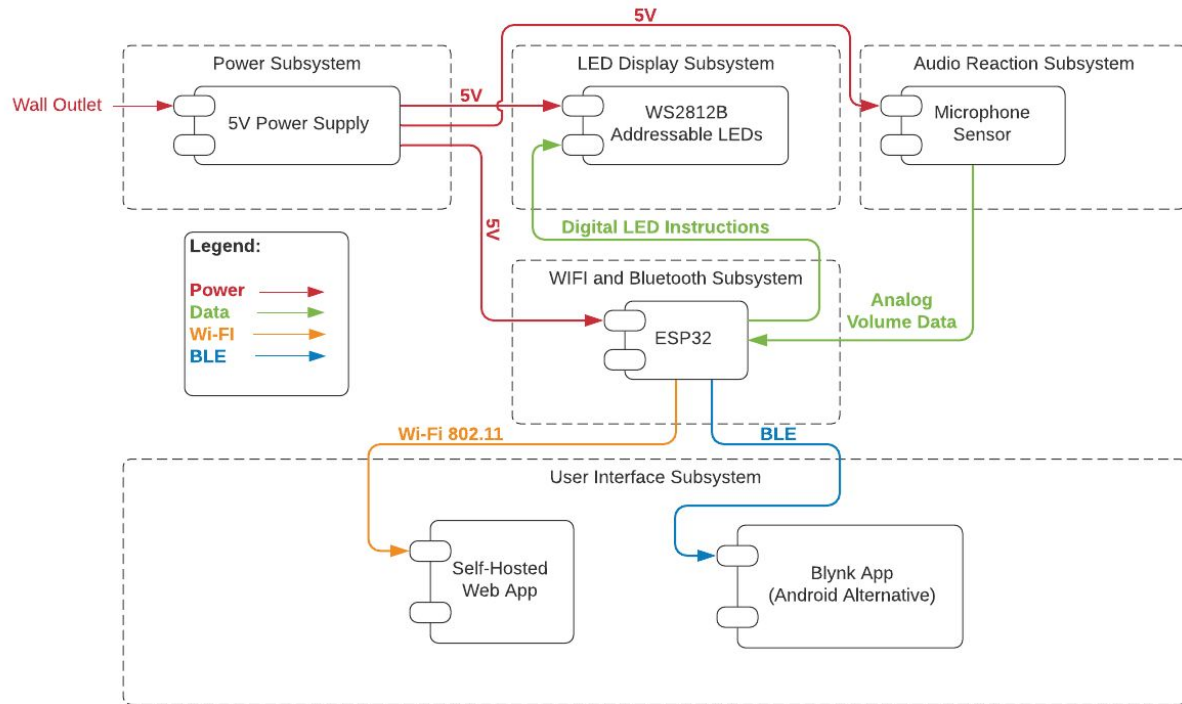

Fig. 5. Block Diagram

2.1 Power Subsystem

A power supply is required to power the LEDs, the ESP32 chip, and the microphone sensor. Both the WS2812B and ESP32 can be powered with 5V, and we will have 600 LEDs with a maximum power draw of 60mA per LED, for a total power draw of 36A. This assumes that all of our LEDs are on at max intensity, which would never be the case in our project. The ESP32 and microphone draw a negligible amount in comparison. Therefore a 15A power supply should be suitable, with the note that only half of our LEDs can be on at maximum power at any given time. This subsystem must interface with the LED display subsystem, audio reaction subsystem, and Wi-Fi and Bluetooth subsystem, providing constant power at 5V +/- 0.1V. If the power supply were to malfunction, the entire device would fail.

2.1.1 5V Power Supply

We will be using a power supply with a built-in AC to DC converter that plugs into a standard US wall outlet. This will allow for easy set-up for the users, who only have to plug the device into the wall. It will interface with the LEDs, a microphone sensor, and ESP32 to provide power.

| Requirements	Verification

1. Supply voltage output of 5V +/- 2% power
2. Can support load between .036A to 15A
	1. Probe both terminals using an oscilloscope, ensuring output voltage is within 2% of 5V.
	2. Connect output of 5V power supply to VDD of constant-current test circuit from Figure 5.
	    a. Change R_s from Figure 5 to deliver a max of 15A. Measure using a multimeter.
	    b. Measure output voltage with an oscilloscope, ensure output voltage is within 2% of 5V.
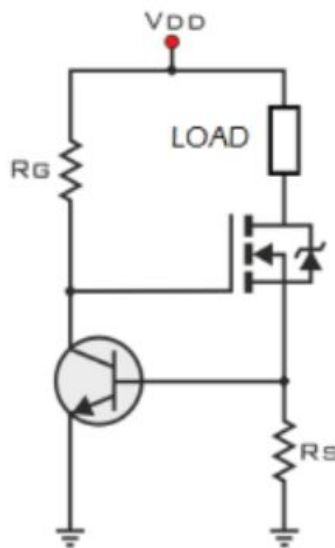
Fig. 6. Circuit Schematic for Constant Current Test Circuit where $I_L = .07/R_S$

2.2 LED Display Subsystem

The LED Display will act as the main visual interface that displays the current score of the game or record of the user's favorite team. This subsystem will interface with the power supply to receive 5V power, and with the ESP32 to receive digital lighting instructions through its GPIO pins. If a portion of the LED strip were to fail, the rest of the device would function normally, but with some LEDs non-operational.

2.2.1 WS2812B Addressable LEDs

We will be using two rolls of 5 meter, 300 pixel WS2812B LEDs. This gives us 600 LEDs to work with at a resolution of about 1.5 LEDs per inch. If we do an array of 40x15, we have a dimension of ~26 inches by ~9 inches. This is the perfect size for a wall mounted device such as ours. Each roll will be cut into 7 strips of 40 and one strip of 20, giving us 14 strips of 40 and 2 strips of 20. These will be arranged horizontally to the front of the enclosure, with power connected to each horizontal strip to reduce loss from resistance. The data pins will be serial, giving us an array we can easily map to control the lights as if they are a 40x15 display.

Requirements	Verification

1. Each of the 600 LEDs can operate at 36mA and are visible from 5m away.
2. Each of the 600 LEDs are programmable by the ESP32 microcontroller, and can be set to 10 different colors (red, yellow, orange, green, navy blue, baby blue, purple, pink, white, neon green).
	1.
	   a. Deliver 36mA to load by changing R_s from the constant current circuit of Figure 5. 5V must be supplied to VDD from a 5V power source.
	   b. Use a meter stick to measure 5m.
	2. Write a test script that iterates over 600 LEDs. At each LED, set color to each of the ten colors listed in requirements. If all ten colors are shown, then all LEDs are programmable. Results will be kept on a grid of 600 rows and 11 columns. The first column is the LED #, and the next 10 columns correspond to the color.

2.3 Audio Reaction Subsystem

The audio reaction subsystem makes the scoreboard an interactive highlight of the room in group settings. It is quite common for sports fans to become vocal during close times or big plays, and the scoreboard will listen to such moments and react accordingly. There may be a "sound level" setting similar to that used at some stadiums, or an animation that occurs after the volume hits a certain threshold. This subsystem interacts with the ESP32 to send the analog microphone volume information via a GPIO pin, and with the power subsystem to receive 5V power. If this subsystem were to fail, the rest of the device would function normally, just without audio reaction in the lights.

2.3.1 Microphone Sensor

We will use a SparkFun Sound Detector or create our own microphone sensor to detect the room's noise level. The sensor takes 5V power so it can be powered by the power supply. It will transmit the sensor data through one of the ESP32's GPIO pins.

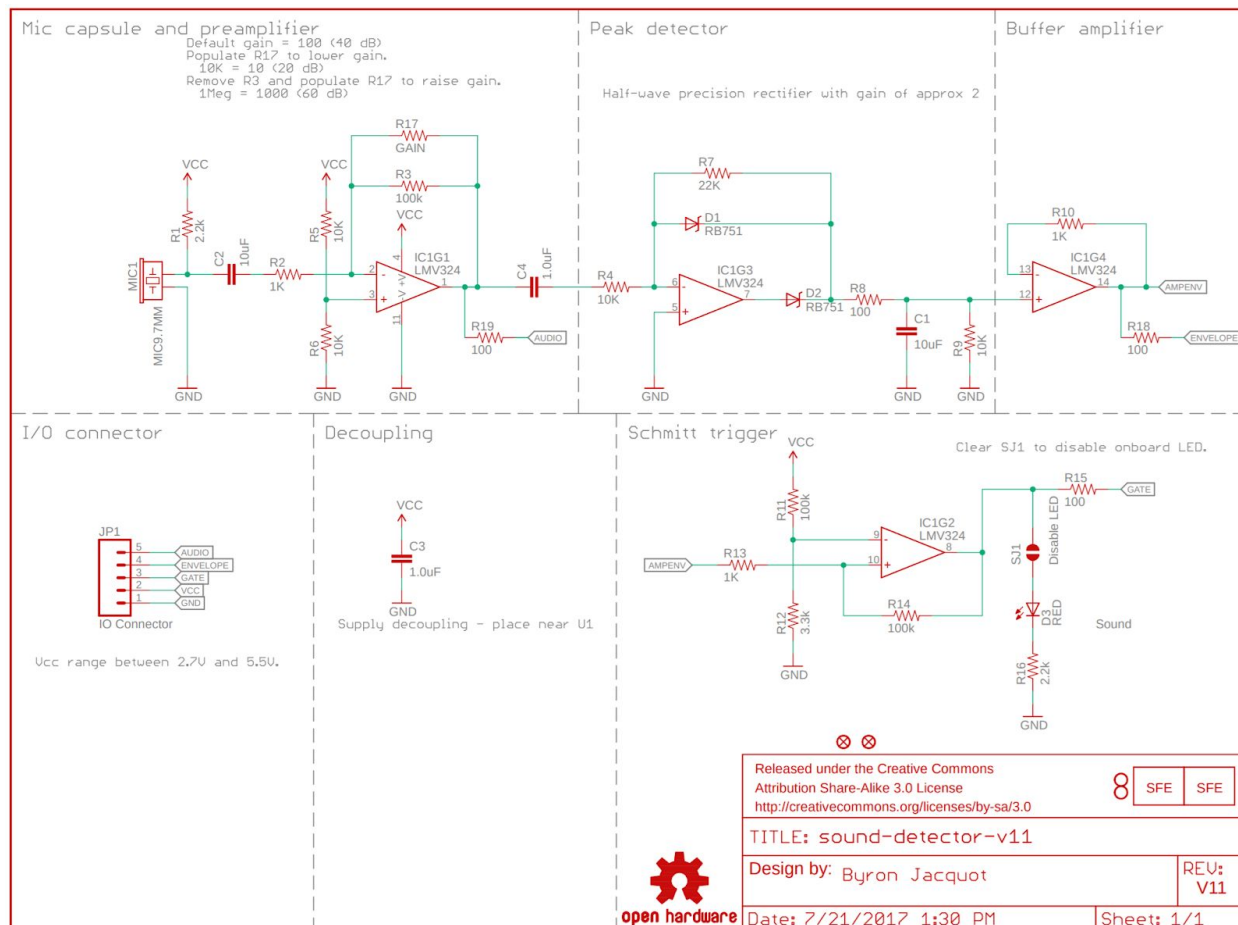| Requirements	Verification

1. Must be able to differentiate between normal room talking volumes and yelling volumes.	1.
	   a. Test the sensor at a baseline volume level and record the sensor output.
	   b. Test the sensor at a yelling volume level and record the sensor output.
	   c. Ensure that there is an observable (+/- 10%) difference in the recorded sensor output.

Fig. 7. Circuit Schematic for SparkFun Sound Detector.

2.4 Wi-Fi and Bluetooth Subsystem

While called the Wi-Fi and Bluetooth subsystem, this subsystem also acts as the driving microcontroller to control the lights, interpret microphone data, and interact with the user through the web application and bluetooth connection. This subsystem interfaces with every other subsystem. It interfaces with the power subsystem to receive 5V power, with the LED display subsystem to send lighting information over the GPIO pins, with the audio reaction subsystem to receive analog volume data over a GPIO pin, and with the user interface subsystem through Wi-Fi 802.11 and Bluetooth Low Energy. This is the most critical subcomponent, and any part of the ESP32 malfunctioning would likely cause the entire system to fail.

2.4.1 ESP32

We are using an ESP32 chip as it has support for both Wi-Fi and Bluetooth, and it has a dual core processor to handle both the connectivity and the application code. It is limited to 520 kB of on-chip SRAM, but has support for external Flash and SRAM should our code be larger. Most

modules contain an external Flash built in. This interfaces with the power supply for 5V power, the lights to control them via a GPIO pin, and the microphone sensor to read input through a GPIO pin. It also interacts with the user interface subsystem by hosting a local web application through Wi-FI, and retrieving Wi-Fi password information through Bluetooth.

Requirements	Verification
3.	3.
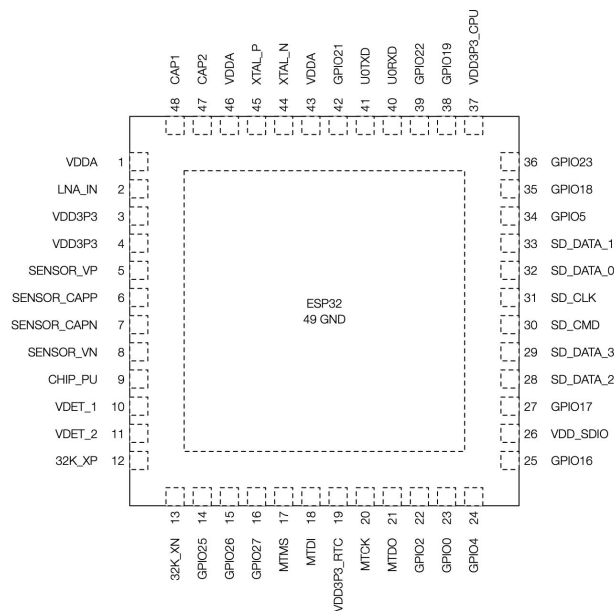
Fig. 8. ESP32 Pin Layout [10]

2.5 User Interface Subsystem

This subsystem is fully in software, but will be the only way the user interacts with the device to change their favorite team or control the visual settings. There will be two user interfaces, the main one is a web application connected to the ESP32 over Wi-Fi to control the settings, and the second one is through a phone app called Blynk connected to the ESP32 over Bluetooth Low

Energy to retrieve the Wi-Fi password for the network it will be connected to. If this subsystem fails, the user would no longer be able to connect to new Wi-Fi networks or change their favorite team, but if the password is already entered the rest of the device would continue to function properly.

2.5.1 Web Application

We will create an intuitive web interface for users to communicate with the device. This is where users will tell the device who their favorite teams are, and what visual configuration they want. Some options could include a moving gradient of lights, a scoreboard for when games are in progress, or the current record of the team. This block interfaces with the ESP32 chip's Wi-Fi 802.11 protocol and the user's phone or computer browser.

Requirements	Verification
4.	4.

2.5.2 Bluetooth Connection

We will utilize a Bluetooth connection using the ESP32's Bluetooth Low Energy (BLE) to receive the user's Wi-Fi password information. We will utilize Blynk, an iOS app that allows users to create custom UI's and easily connect to Bluetooth enabled microcontrollers suchs as the ESP32. Blynk will allow our group to overcome Android limitations.

Requirements	Verification
5.	5.

2.7 Schematics?

2.8 Board Layout?

2.9 Software

2.9.1 Sports Data Scraper

2.9.2 LED Driver

2.10 Tolerance Analysis

The most important tolerance we must maintain is the ESP32 chips processing speeds to parse the API data and drive the 600 LEDs in real time. The ESP32 utilizes a dual-core Xtensa LX6 processor, with both cores running at 240MHz. The ES2812B LEDs do not depend on processing speeds however, they instead depend on the length of the led array. Each LED requires 3 Bytes of data (1 for each color channel), giving us a total memory requirement of:

$$3 \text{ (Color channels) } * 600 \text{ (LEDs)} = 1800 \text{ Bytes} = 14400 \text{ Bits} \qquad \text{Eq. 1}$$

The ESP32 has 520kB of SRAM, so this is not an issue. Looking at the datasheet, the transmission time takes 1.25 us per bit, and 50us for the reset, for a total transmission time of:

$$14400 \text{ (bits) x } 1.25\mu s + 50 \text{ } \mu s = 18.05 \text{ ms} = 55.4 \text{ fps} \qquad \text{Eq. 2}$$

Most modern monitors use either 30fps or 60fps, so this falls within the acceptable range of real time response rates. If needed, we could use multiple GPIO ports and split the data to shorter segments, but that will not be necessary with this number of LEDs.

As part of the ESP32 documentation, one of the processors manages the Wi-Fi and Bluetooth while the other manages the application code.

3 Costs and Schedule

3.1 Costs

As shown below, our costs are broken down into two categories, labor and components. This is very common when managing a project. The grand total is composed of the sum of the labor costs and the component costs. The grand total for the costs of our project is **INSERT GRAND TOTAL.**

3.1.1 Labor Costs

The labor costs of are project are broken down on a person by person basis and summed up at the end to give us the total cost of labor for the group. As professional engineers, about to enter the working world, we used a rate of $30/hour. We assumed that for the rest of the 10 weeks, we will all be contributing about 10 hours/week to the project. Using these statistics we came up with our total labor costs to be $22,500.

Partner	Hourly Rate	Hours/Week	Weeks	2.5x Multiplier	Total/Person
Michael	30	10	10	2.5	$7,500
Max	30	10	10	2.5	$7,500
Matthew	30	10	10	2.5	$7,500

Total Labor Costs: $22,500

3.1.2 Component Costs

3.2 Schedule

4 Ethics and Safety

4.1 Safety

The Smart Sports Scoreboard is intended to be used in the homes of sports fans. Indoor use mitigates some risks, but safety issues are still capable of arising due to a number of unforeseen circumstances that could occur within a user's home. This device has been designed with OSHA standards in mind [8]. All energized electrical components will be properly enclosed or isolated such that the user can interact with the device and have no harm to themself. In creating an enclosure, insulators will be used so that electric current can not flow through the device and into the user. Failure to account for this could, although improbably, result in shock, electrocution, or burns. Although the device is intended for in-home use, there is still a possibility that the Smart Sports Scoreboard could be exposed to wet conditions. Wet conditions lower skin resistance and allow current to flow through the body of the user much easier. This device is constructed such that a user can have slightly wet hands and still safely operate the device. The web interface also allows for the user to operate the device while not having direct contact with the Smart Sports Scoreboard. This component adds another level of safety for the user and their experience.

4.2 IEEE Ethics Accordance

We as a group, in accordance with the IEEE Code of Ethics, understand that it is our responsibility to commit ourselves to the highest ethical and professional standards in creating this device. In particular, our device is responsible for "hold(ing) paramount the safety, health, and welfare of the public" [11]. As a Smart Sports Scoreboard, further versions of the product may include sports gambling features so that users may stay up to date with betting odds on games that they are following. Any sports betting features do not have the ability to make wagers. These features are specifically for entertainment purposes. We as a group only condone sports gambling where it is done legally in states that allow it, and only when it is done responsibly by the individuals making the wagers.
Our device also strives "to treat all persons fairly and with respect" [11]. The purpose of our device is to provide real-time information in a visually appealing way to the user, so that they may spend less time on their smart devices and more time honed in on what is happening around them. We, in no way, intend to alter the information the user receives, and only hope to provide the user with true information. Therefore, we strive to treat all persons, teams, and players being represented on our device fairly and with the utmost respect.

It is our responsibility as a group to hold each other accountable in "striv(ing) to ensure this code is upheld by colleagues" [11]. We, in accordance with the code of ethics, will support our teammates and continually follow up with each other to ensure that we uphold conduct of the

highest standard. If a member of the group is to behave unethically, no retaliation will be present against individuals who report a violation. It is our responsibility in creating a device to make sure that positive value is provided to the world, and this is what we plan on doing.

References

[1]   Pew Research Center, *Mobile Fact Sheet*, Pew Research Center, June 12, 2019. Accessed on February 15, 2021. [Online]. Available: https://www.pewresearch.org/internet/fact-sheet/mobile/

[2]   Asurion, *Americans Check Their Phones 96 Times a Day*, Asurion, November 21, 2019. Accessed on February 14, 2021. [Online]. Available: https://www.asurion.com/about/press-releases/americans-check-their-phones-96-times-a-day/

[3]   C. Gough, *Sports fans using mobile apps worldwide 2019, by age*, Statista, March 20, 2020. Accessed on February 14, 2021. [Online]. Available: https://www.statista.com/statistics/1100567/sports-content-mobile-apps/

[4]   S. Shellenbarger, *Just Look Me in the Eye Already*, Wall Street Journal, May 28, 2013. Accessed on February 15, 2021. [Online]. Available: https://www.wsj.com/articles/SB10001424127887324809804578511290822228174

[5]   T. Nakamura, *The action of looking at a mobile phone display as nonverbal behavior/communication: A theoretical perspective*, ScienceDirect, November 19, 2014. Accessed on February 15, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0747563214005652

[6]   Harvard Business Review, *How to Stop Constantly Checking Your Phone*, Harvard Business Review, January 13, 2020. Accessed on February 14, 2021. [Online]. Available: https://hbr.org/tip/2020/01/how-to-stop-constantly-checking-your-phone

[7]   K. Draper, *ESPN Tries to Get With a Mobile, App-Driven World*, The New York Times, April 12, 2018. Accessed on February 15, 2021. [Online]. Available: https://www.nytimes.com/2018/04/12/sports/espn-app.html

[8]   *Construction Safety & Health*, SH-16586-07-06-F-36, 2018.

[9]   IEEE.org, *IEEE Code of Ethics,* IEEE, 2021. Accessed on February 13, 2021. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html

[10]   Espressif Systems, "ESP32 Series," ESP32-D0WD-V3 datasheet, Jan 2021. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf