Faculty Of Information Technology

IN 1901 - Micro-controlled Based Application Development Project

Self-Training Running Track System

Group No:48

Index No:	Name
224111U	**Kulasekara Y.B.M**
224094R	Jayaweera W.G.H.M.P.S.C
224097E	Kahaduwa T.S
224085P	Jayaprabha P.H.J
224076N	Himaya K.S

Contents

1. Introduction

We are pleased to present an innovative solution designed to enhance the training experience for athletes on running tracks. Our system aims to provide an accurate, user-friendly, and efficient timing solution for runners of all skill levels. Whether you're a competitive athlete aiming to break personal records or a recreational runner seeking improvement, our advanced timing system will revolutionize your training by making sessions more productive and focused on performance improvement.

2. Literature Survey

A Literature Study involves diving into existing writings about a subject to understand what experts have discovered or discussed. For our project, it means examining research papers, articles, and books about sprinting timing systems, sensor technologies, and athlete performance analysis. By delving into this literature, we gain insights into how timing systems work, which sensors are effective, and how athlete performances are evaluated. This process helps us learn from past experiences, avoid repeating mistakes, and incorporate proven methods into our design. Ultimately, it ensures that our project is built on a solid foundation of knowledge and addresses the specific needs of athletes, coaches, and event organizers. The literature study guides us in making informed decisions, refining our approach, and creating a more effective and efficient sprinting timing system.

3. Problem in brief

Runners frequently encounter difficulties when training without a coach, especially when it comes to accurately timing their runs. Manual timing methods often result in inaccuracies, making it challenging to track progress and refine running techniques. To address this, there is a need for a reliable self-training system that offers precise time tracking, instant feedback and helps athletes improve their performance independently.

4. Aim & Objective

Aim:

We aim to assist running athletes in overcoming the challenges they face when training alone by providing innovative solutions that enable accurate timing, immediate feedback, and comprehensive data analysis. Through the development of self-training running track timers, we seek to empower athletes to train autonomously and optimize their performance effectively, ultimately enhancing their overall training experiences and enabling them to reach their full potential.

Objectives:

- Integrate a camera module to accurately identify the starting and finishing positions of the runner, ensuring precise timing of each run.
- Implement a chip module to record video footage of the runner's performance, allowing for playback and analysis of running sessions.

- Utilize a sound counter to provide auditory cues for the runner, indicating the start time and enhancing the user experience.
- Incorporate an LDR sensor to detect the runner's movement at the finish line, automatically capturing the finishing time.
- Deploy two ESP32 modules with Wi-Fi capability to enable communication between components and facilitate data transfer to external devices.

- Utilize a display module to present the timing data of the runner in real-time, providing instant feedback on performance.
- Implement a database system to store and organize the timing data of each runner, allowing for easy retrieval and analysis of historical performance records.

5. Proposed Solution

The proposed solution for our project incorporates several features aimed at enhancing the precision, usability, and analysis capabilities of the sprinting timing system.

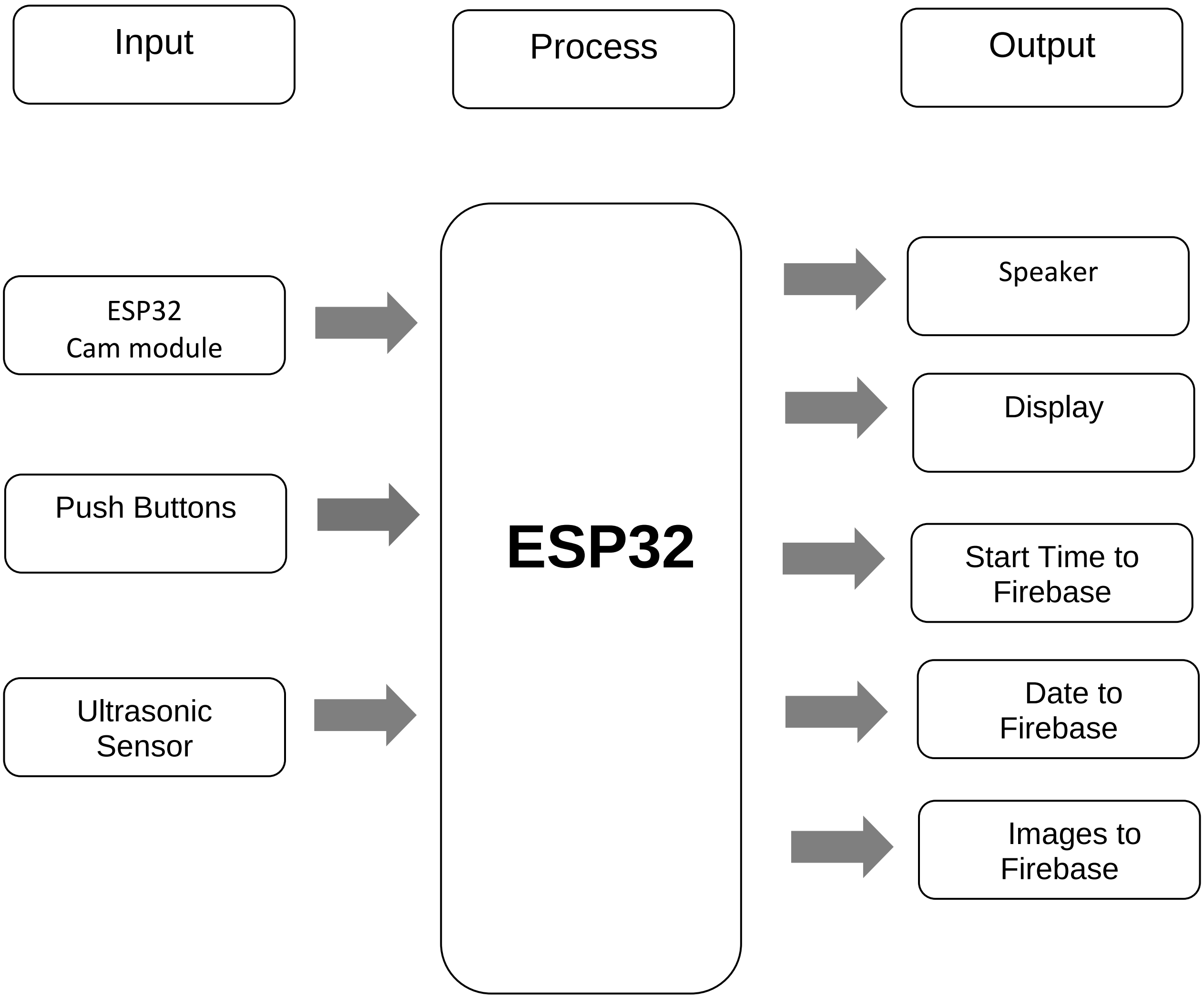
- High Precision Timing: Integration of laser technology, LDR sensors, and camera modules ensures accurate detection of start and finish times, minimizing errors in timing measurements.
- Video Recording and Playback: The chip module enables video recording of runner performances, allowing for detailed playback and analysis to identify areas for improvement.
- Auditory Cues: Use of a sound counter provides runners with audible signals for start times, enhancing user experience and ensuring synchronized starts.
- Automated Finish Line Detection: LDR sensors automatically capture the moment a runner crosses the finish line, eliminating the need for manual intervention and ensuring precise finish times.
- Wireless Communication: ESP32 modules with Wi-Fi capability facilitate seamless communication between components and enable data transfer to external devices, enhancing system flexibility and accessibility.
- User Interaction: The system features a simplified input method using two buttons — green and red — offering a user-friendly interface for easy operation. The green button for ‘YES’ actions, while the red button for ‘NO’, streamlines the process for users.
- Real-time Feedback: The display module presents timing data in real-time, providing instant feedback to runners and coaches on performance metrics.

- Web Interface: The development of a user-friendly web interface with Firebase allows users to access timing data and watch runner videos remotely, enhancing accessibility and convenience.
- Secure Data Storage: Firebase ensures secure storage of timing data and runner information, safeguarding sensitive data and providing peace of mind to users.
- Integration with image Capture: Integration of ESP32 chip modules allows for direct capture and storage of images onto Firebase, simplifying data management and analysis processes.
- Seamless Connectivity: Utilization of pocket Wi-Fi modules ensures seamless wireless connectivity between ESP32 modules and the web interface, enabling efficient data transmission and access.

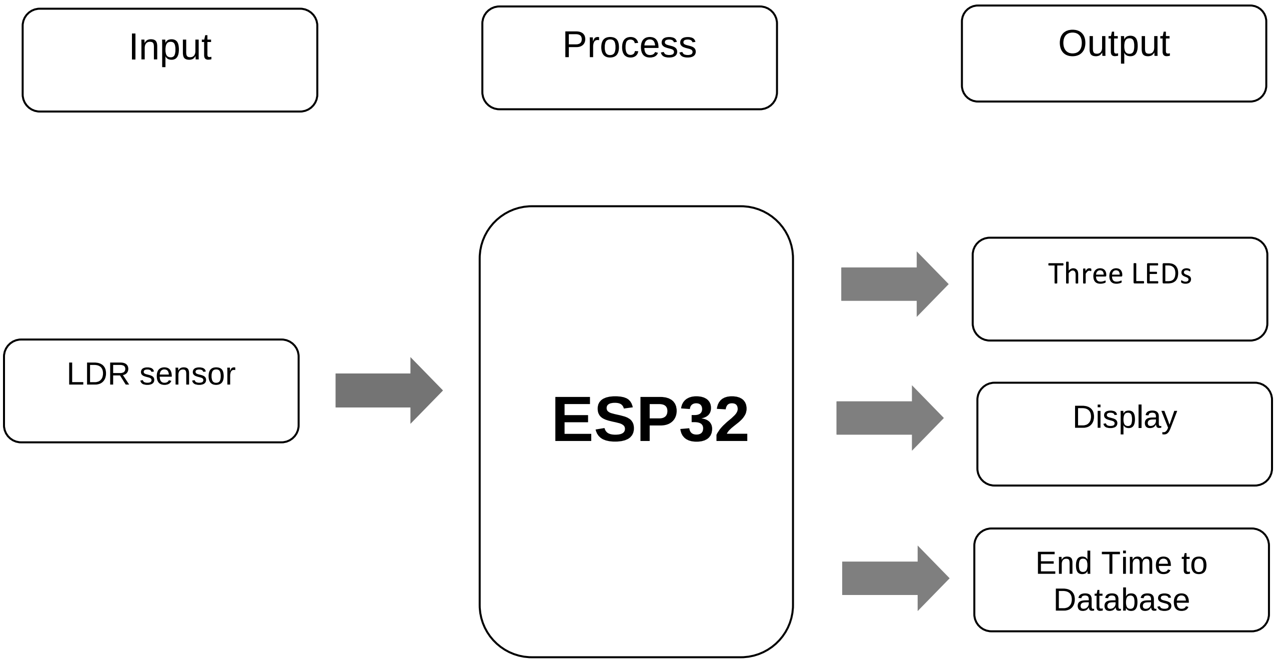
6. Analysis & Design

6.1 Block Diagram

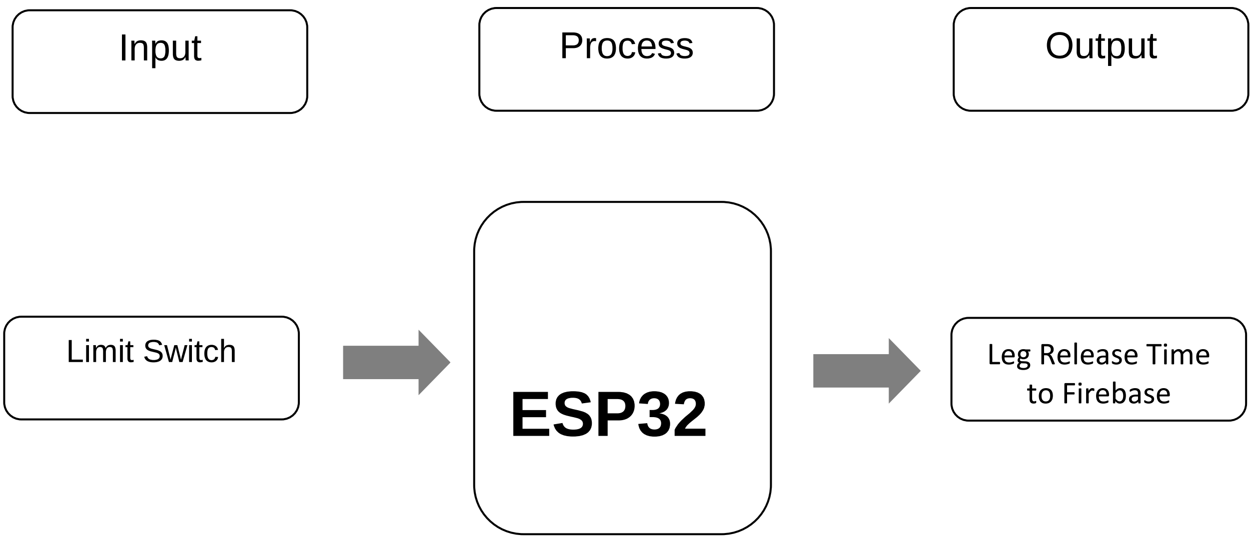
<u>Start</u>



End

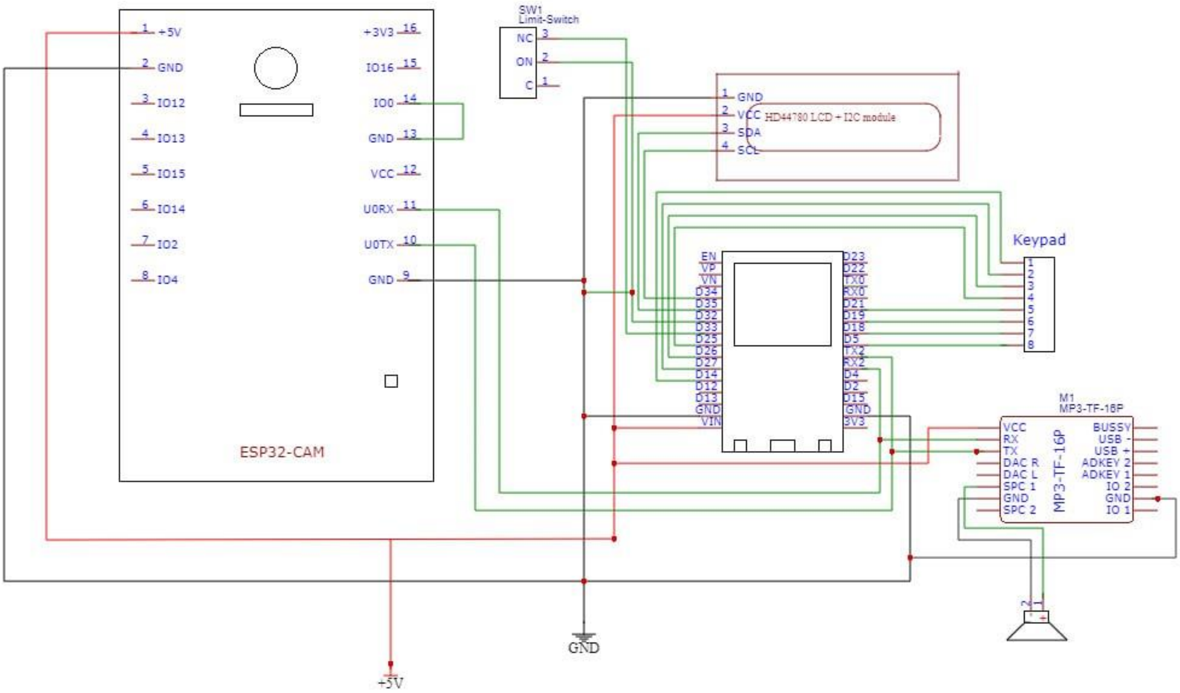


Starting Block

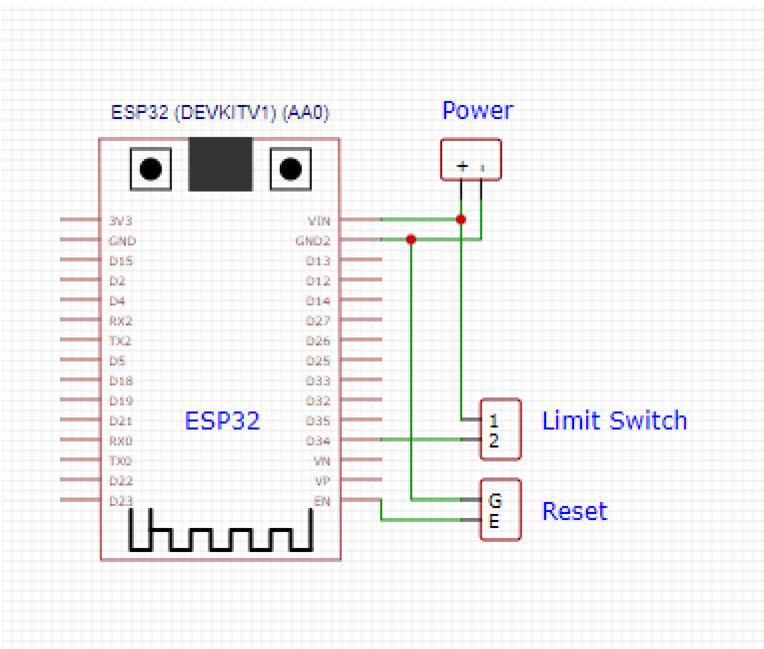


6.2 Schematic Diagram

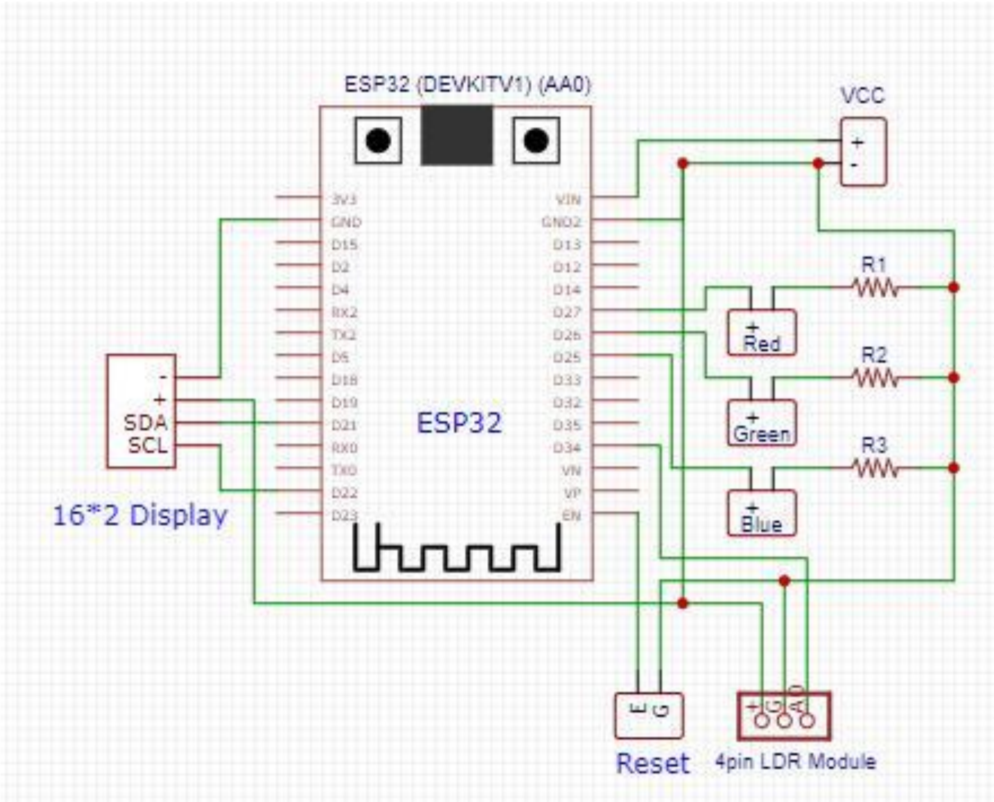
Start



Starting Block



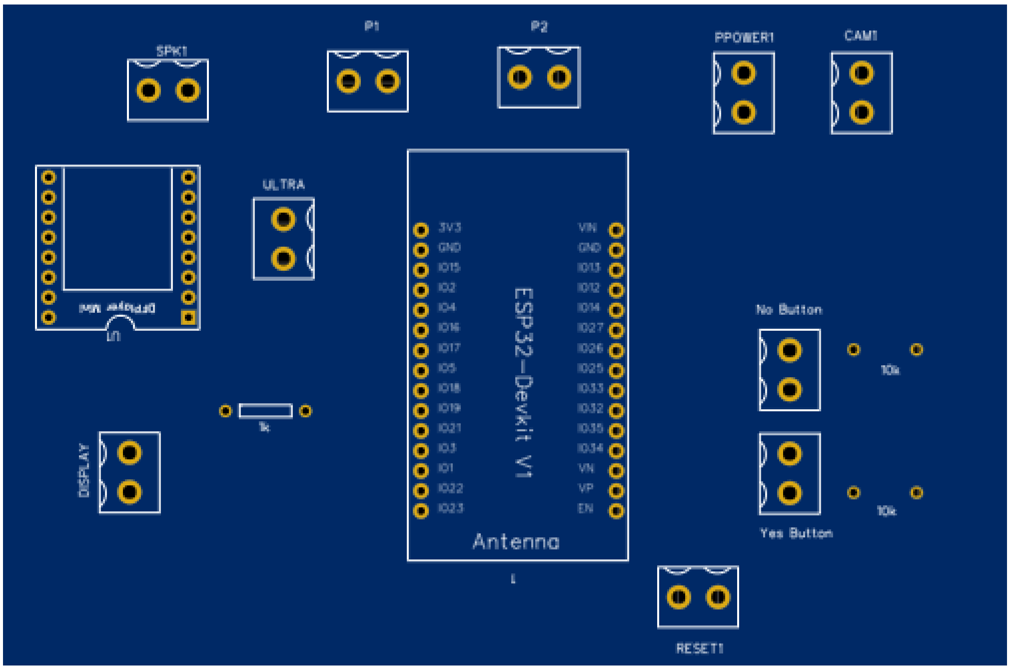
End



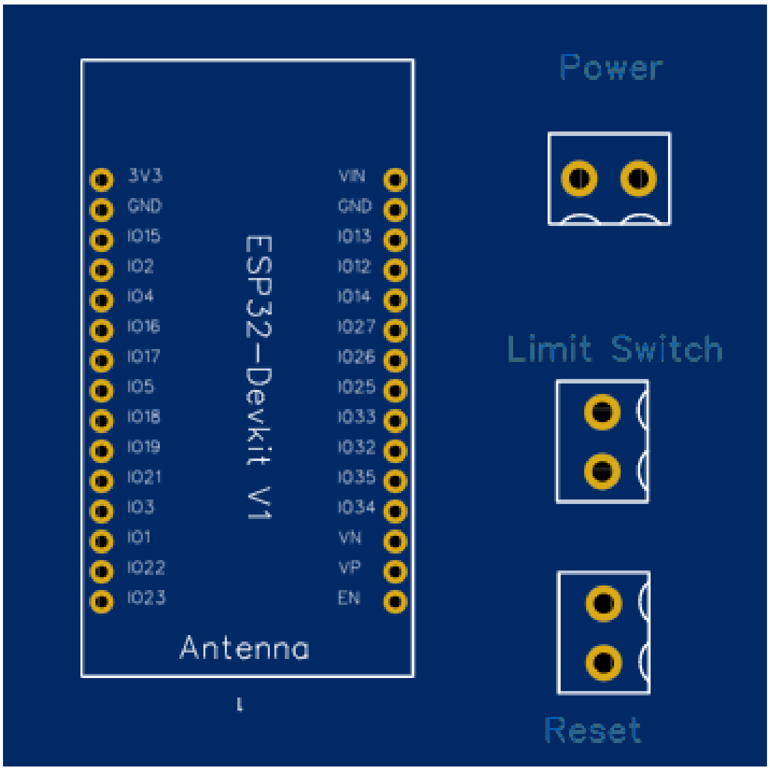
ESP32 (DEVKITV1) (AA0)
VCC
+
-
3V3
GND
D15
D2
D4
RX2
TX2
D5
D18
D19
D21
RX0
TX0
D22
D23
VIN
GND2
D13
D12
D14
D27
D26
D25
D33
D32
D35
D34
VN
VP
EN
ESP32
R1
R2
R3
Red
Green
Blue
SDA
SCL
16*2 Display
Reset
4pin LDR Module

6.3 PCB Design

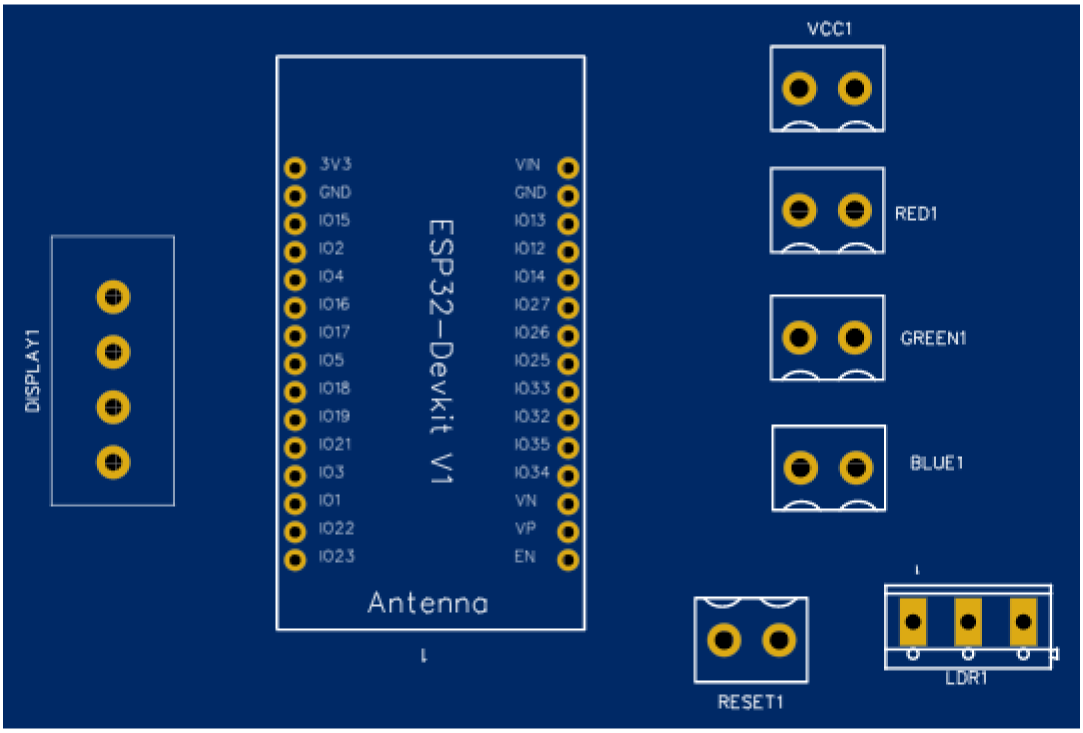
Start



Starting Block



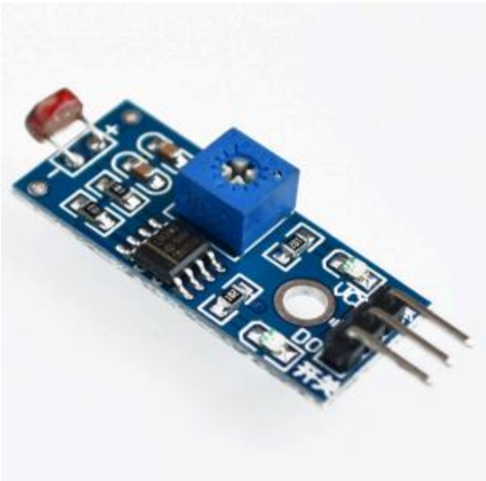
<u>End</u>


VCC1
RED1
GREEN1
BLUE1
DISPLAY1
3V3
GND
IO15
IO2
IO4
IO16
IO17
IO5
IO18
IO19
IO21
IO3
IO1
IO22
IO23
VIN
GND
IO13
IO12
IO14
IO27
IO26
IO25
IO33
IO32
IO35
IO34
VN
VP
EN
ESP32-Devkit V1
Antenna
RESET1
LDR1

7. Modules in the System

1. LDR Sensor

A Light Dependent Resistor (LDR), also known as a photoresistor, is a resistor whose resistance varies based on the amount of light that strikes its surface. In our system, the LDR sensor is used to detect when the runner crosses the finish line, signaling the end of the race and recording the final time. LDRs are commonly used in light-sensing circuits for various applications such as light meters, street lighting, and alarm systems. The sensitivity of the LDR to changes in light makes it a reliable component for detecting the runner's completion of the race.



2. 16*2 LCD Display

The 16x2 LCD is a widely used component in electronic projects for displaying textual information. This display can show up to 16 characters per line on 2 lines, providing a total of 32 characters. In our system, it is used to display timing information at the finish line, offering clear and real-time feedback to the runner. The display also features an optional backlight for better visibility in various lighting conditions. It supports both 4-bit and 8-bit data interfaces, giving flexibility in design and implementation, and making it suitable for a range of applications.

3. 20*4 LCD Display

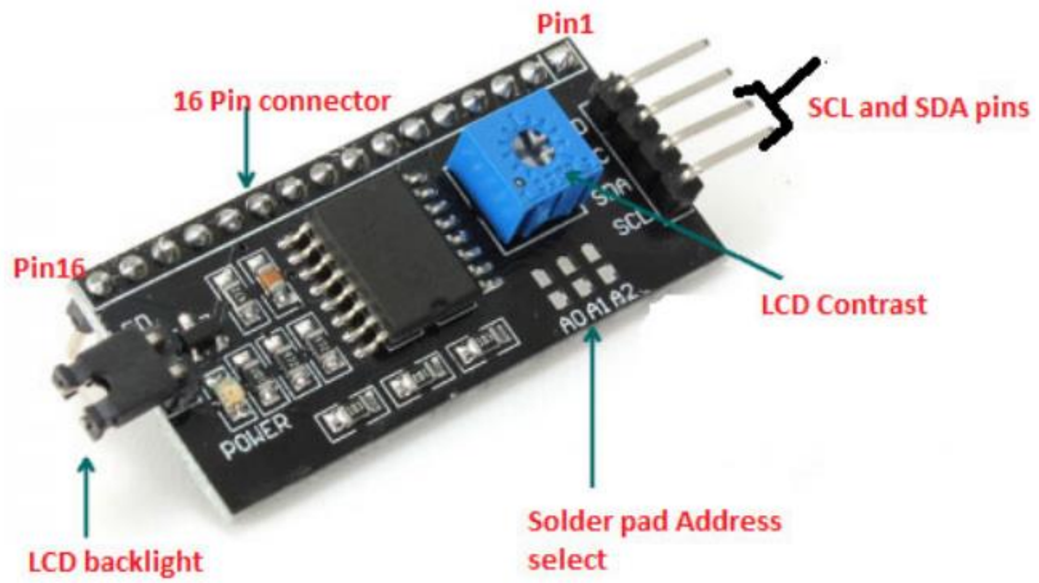
The 20x4 LCD display is a popular alphanumeric display used in various electronic and embedded systems. It can display up to 20 characters per line across 4 lines, providing a total of 80 characters. In our system, this display is used at the starting point to show commands to the runner, ensuring clear and easy-to-read instructions. The display features an optional backlight for enhanced visibility in different lighting conditions and supports both 4-bit and 8-bit data interfaces, allowing for flexibility in system design and implementation

4. I2C Module

In the context of LCDs, an I2C module can be used to control a 20x4 or 16x2 LCD, significantly reducing the number of GPIO pins needed from the microcontroller. Instead of using multiple data lines, the display can be controlled with just two lines (SDA and SCL), freeing up resources for other components and simplifying the wiring**.**

<u>Device Specification</u>

- Input Voltage: 2.5V to 6V DC (typically 5V)
- Communication Protocol: I2C/TWI (Two-Wire Interface)
- I2C Address: 0x27 or 0x3F (modifiable via jumpers)
- Current Consumption: < 20mA
- Data Rate: Up to 100kHz (standard mode) or 400kHz (fast mode)
- Compatible LCD Types: 16x2, 20x4 character LCDs
- Operating Temperature: -40°C to +85°C
- Number of Pins: 4 pins (VCC, GND, SDA, SCL)
- Dimensions: 41.5mm x 19mm
- Weight: Approximately 4g

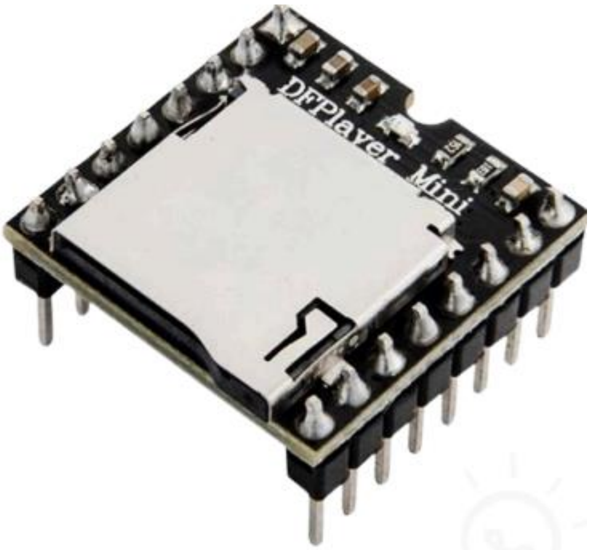


5. Amplifier

The MP3-TF-16P is a compact audio amplifier module featuring a built-in MP3 decoder, TF card slot, USB port, and Bluetooth connectivity. It supports multiple audio formats, has a high-quality stereo output, and easy integration, and is ideal for DIY audio projects and portable speaker systems.

Device Specification

- Input Voltage: 3.2V to 5V DC
- Power Consumption: 20mA
- Storage Type: Supports microSD card (up to 32GB)
- File Format Support: MP3, WAV, WMA
- Supported File System: FAT16/FAT32
- Speaker Support: 4Ω or 8Ω speaker (directly connected)
- Audio Bit Rate: 24kbps to 320kbps
- Output Power: 3W
- Dimensions: 22mm x 30mm
- Weight: Approximately 5g



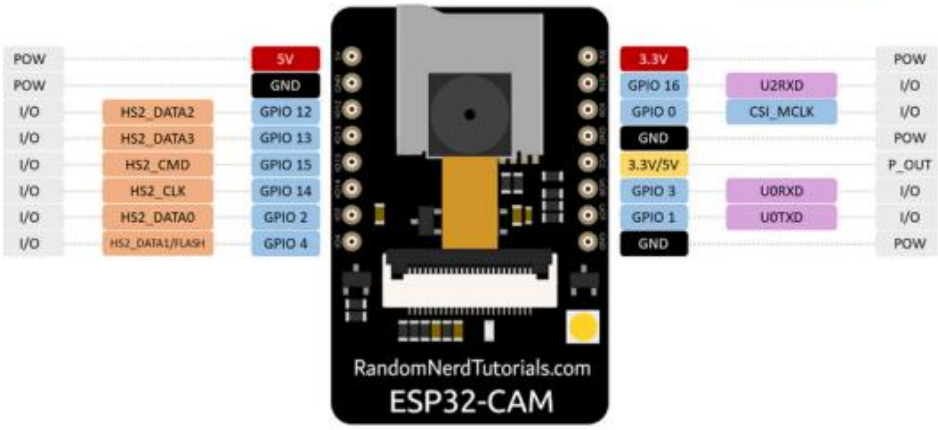
MP3-TF-16P Module

6. ESP 32 Camera

The ESP32 Camera is a versatile module that integrates the ESP32 microcontroller with a camera module, enabling the capture and processing of images or video streams. It leverages the ESP32's Wi-Fi and Bluetooth capabilities to transmit captured media over the network or to a connected device. The module typically includes built-in camera, such as the OV2640 or OV7670, and supports various image resolutions and formats. Here we are capturing the start position of the sprinter for checking the accuracy.

<u>Device Specification</u>

- **Microcontroller:** ESP32 Dual-core Ten silica LX6 processor
- **Clock Speed:** Up to 240 MHz
- **Flash Memory:** 4MB
- **RAM:** 520KB SRAM
- **Storage:** Supports microSD card up to 4GB
- **Camera:** OV2640 2MP (1600x1200 resolution) with 60°, 90°, or 120° lens options
- **Wi-Fi:** 802.11 b/g/n (2.4 GHz)
- **Input Voltage:** 5V (via USB) or 3.3V (via GPIO pins)
- **Operating Voltage:** 3.3V
- **GPIO Pins:** 9 (multipurpose)
- **Image Output Format:** JPEG, BMP, and Grayscale
- **Interface:** UART, SPI, I2C, PWM, ADC, DAC
- **Camera Interface:** SCCB
- **Power Consumption:**
    - Deep Sleep: 6mA
    - Active Mode: 160-260mA
- **Antenna:** Built-in PCB antenna (with an option to add an external antenna)
- **Dimensions:** 27mm x 40.5mm x 4.5mm
- **Weight:** 9.6g


POW
POW
I/O
I/O
I/O
I/O
I/O
I/O
HS2_DATA2
HS2_DATA3
HS2_CMD
HS2_CLK
HS2_DATA0
HS2_DATA1/FLASH
5V
GND
GPIO 12
GPIO 13
GPIO 15
GPIO 14
GPIO 2
GPIO 4
3.3V
GPIO 16
GPIO 0
GND
3.3V/5V
GPIO 3
GPIO 1
GND
U2RXD
CSI_MCLK
U0RXD
U0TXD
POW
I/O
I/O
POW
P_OUT
I/O
I/O
POW
RandomNerdTutorials.com
ESP32-CAM

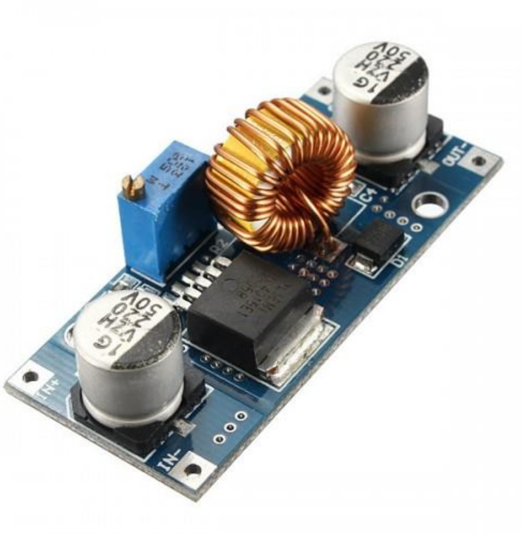

120°
90°

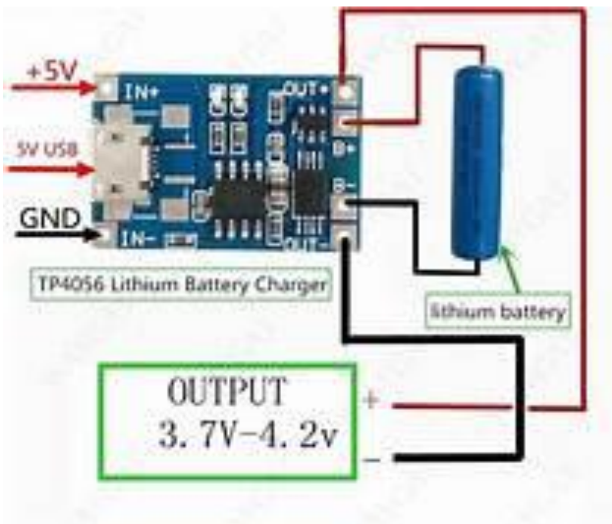

45°

7. Power &Charging module

The 18650 rechargeable battery is a type of lithium-ion cell named for its dimensions: 18mm in diameter and 65mm in length. These batteries are commonly used in a variety of applications, including laptops, flashlights, and electric vehicles, due to their high energy density, long cycle life, and relatively low self-discharge rate.

A charger for 18650 batteries is designed to safely replenish the battery's power. These chargers come in various types, from simple single-slot models to advanced multi-slot chargers with features like overcharge protection, temperature monitoring, and the ability to charge multiple batteries simultaneously. They ensure the battery is charged efficiently while preventing damage and prolonging the battery's lifespan.





8. Limit Switch

A limit switch is an electromechanical device used to detect the presence or absence of an object by physically contacting it. In our system, we utilize a lever limit switch to identify when a runner has started or is about to start running. The switch is triggered when the runner releases their leg from the starting position. If the runner releases their leg 7 seconds after pressing the lever, it is considered a false start. The limit switch plays a crucial role in ensuring accurate detection of the runner’s movement, contributing to the timing and fault detection mechanisms of the system.

<u>Device Specifications</u>

- Input Voltage: 250VAC, 125VDC
- Rated Current: 15A (250VAC), 0.6A (125VDC), 0.3A (250VDC)
- Contact Configuration: SPDT (1 NO + 1 NC)
- Operating Force (OF): 200gf max
- Mechanical Life: >1,000,000 operations
- Electrical Life: >50,000 operations
- Insulation Resistance: ≥100 MΩ (500VDC)
- Contact Resistance: ≤50 mΩ
- Dielectric Strength: 1000VAC for 1 minute (between terminals)
- Operating Temperature: -25°C to +85°C
- Dimensions: 27.8mm x 10.3mm x 15.9mm
- Mounting Hole Pitch: 22.2mm
- Weight: 6g

9. Ultra Sonic Sensor

An ultrasonic sensor operates on the principle of emitting and detecting high-frequency sound waves. It consists of a transmitter component responsible for generating these ultrasonic waves, typically at a frequency of around 40 kHz, which is beyond the range of human hearing. These sound waves are then emitted in a specific direction, propagating through the air in a spherical pattern. When these sound waves encounter an object in their path, they bounce off the object's surface, resulting in a process known as reflection.
The sensor also contains a receiver that detects the reflected sound waves. By measuring the time, it takes for the sound waves to travel to the object and back, the sensor can calculate the distance to the object using the speed of sound in the air. This distance measurement capability makes ultrasonic sensors valuable for tasks such as object detection, distance measurement, and obstacle avoidance in various applications.

<u>Device Specifications</u>

- Input Voltage: +5V DC
- Quiescent Current <2mA
- Working current: 15mA
- Effectual Angle <15°
- Ranging Distance: 2 – 400 cm
- Measuring Angle: 30°
- Trigger Input Pulse width: 10uS
- Dimension: 45mm x 20mm x 15mm
- Weight: 10g

10. LED Bulbs

Our system utilizes three LED bulbs—red, green, and blue—to provide visual cues during the runner's performance. The red LED illuminates when the laser beam is not detected by the LDR sensor, indicating that the finish line has not yet been crossed. Once the laser beam is detected by the LDR sensor, the green LED lights up. After 10 seconds, the green LED turns off, and the blue LED illuminates, indicating that the system is now in a valid run-checking phase. The run is only considered valid if the runner crosses the laser beam while the blue LED is lit. This sequence of LEDs ensures clear communication of the system’s status during the race.

11. Chip Module

MicroSD cards are versatile storage devices commonly used in electronic projects to store various types of data. In our project, two microSD card modules are used: one to store a pre-recorded voice command and the other to store images captured by the ESP32-CAM. These cards offer significant storage capacity in a compact form, making them ideal for embedded systems. They support fast read and write speeds, ensuring quick access to stored data. Their reliability and ability to manage large amounts of data make them an essential component for handling both voice commands and image files in our system.

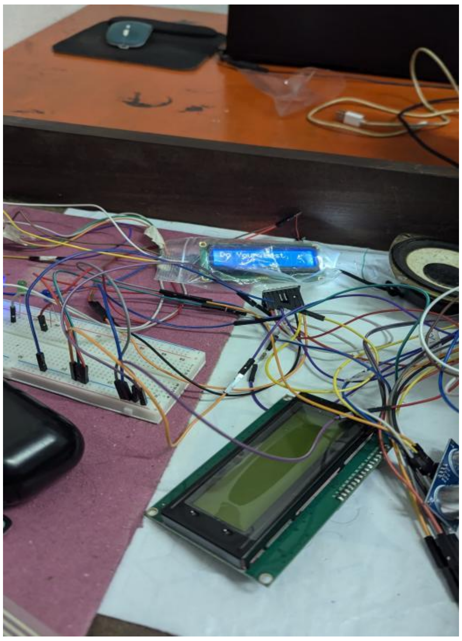
12. Speaker

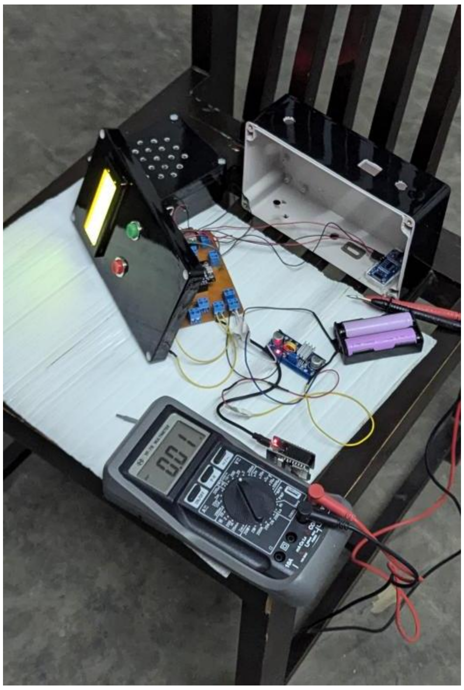
The system includes a 3W speaker to provide audio feedback during the training process. This speaker delivers clear and audible sound, ensuring that runners receive immediate voice or sound notifications. With its compact design and power output of 3 watts, the speaker is well-suited for our application, providing sufficient volume for both indoor and outdoor environments. It operates efficiently with low power consumption, making it ideal for integration into the overall system without significantly affecting power requirements.

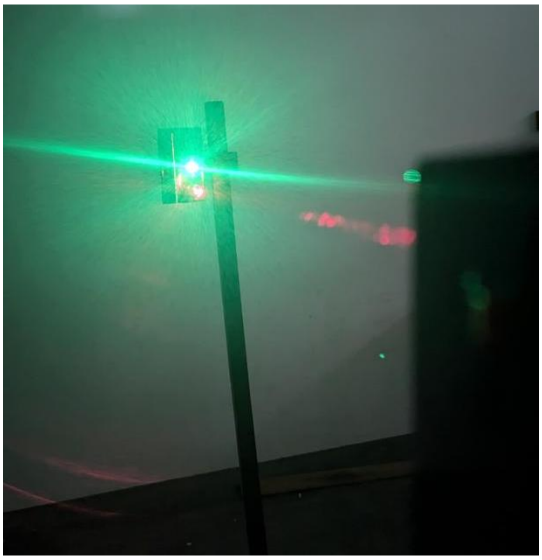
7. Testing and Implementation

The Self-Training Running Track System is an innovative solution designed to automate and enhance the training experience for runners. This system incorporates various hardware components, including sensors, switches, displays, and ESP32 boards, to create an intelligent and interactive training environment. The system utilizes a combination of components such as the HC-SR04 ultrasonic sensor, limit switches, LDR sensor, 16x2 and 20x4 LCD displays, and ESP32-CAM for capturing real-time data and providing immediate feedback. The ESP32 boards serve as the core controllers, managing data collection from the sensors, processing inputs, and controlling the outputs such as LEDs and displays. The system is designed to provide precise timing, accurate detection of the runner's position, and reliable fault monitoring, ensuring an efficient and user-friendly training process.

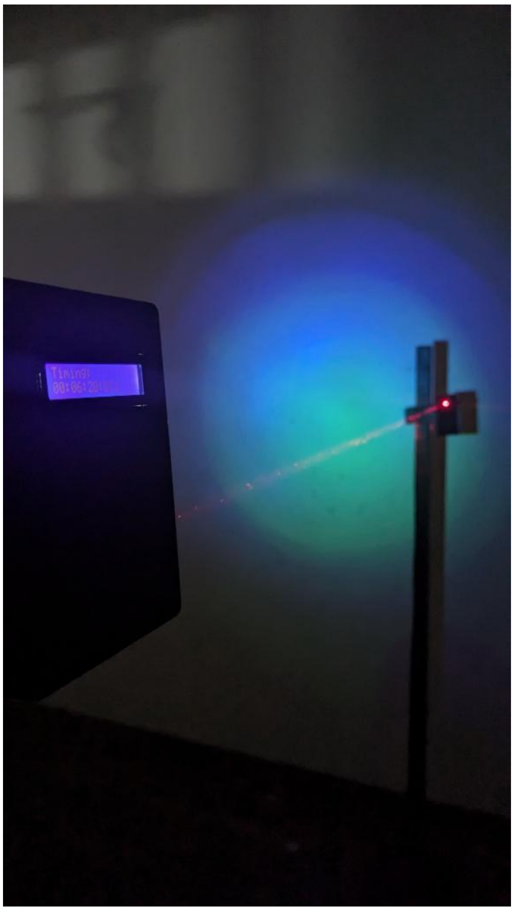
Implementation








PRACTICE MAKES SKILL
Laser Not Detected



Further Work

In future developments of the timing and detection system, several enhancements can be considered. First, integrating wireless communication capabilities could allow real-time data transmission to a central monitoring system, improving accessibility and analysis of performance metrics. Additionally, expanding the functionality of the 16x2 LCD to include graphical representations of timing data could enhance user experience and provide more detailed feedback to runners. Furthermore, exploring the use of more advanced sensors for detecting runner positions could increase the accuracy of fault detection, particularly in identifying false starts. Implementing machine learning algorithms to analyse collected data could also provide insights into runner performance trends and help in optimizing training regimens. Moreover, conducting user testing with a larger group of runners would provide valuable feedback for refining the system's usability and effectiveness. Lastly, considering the development of a mobile application to complement the hardware could facilitate easier access to timing data and performance analytics for users.

8. Individual Contribution

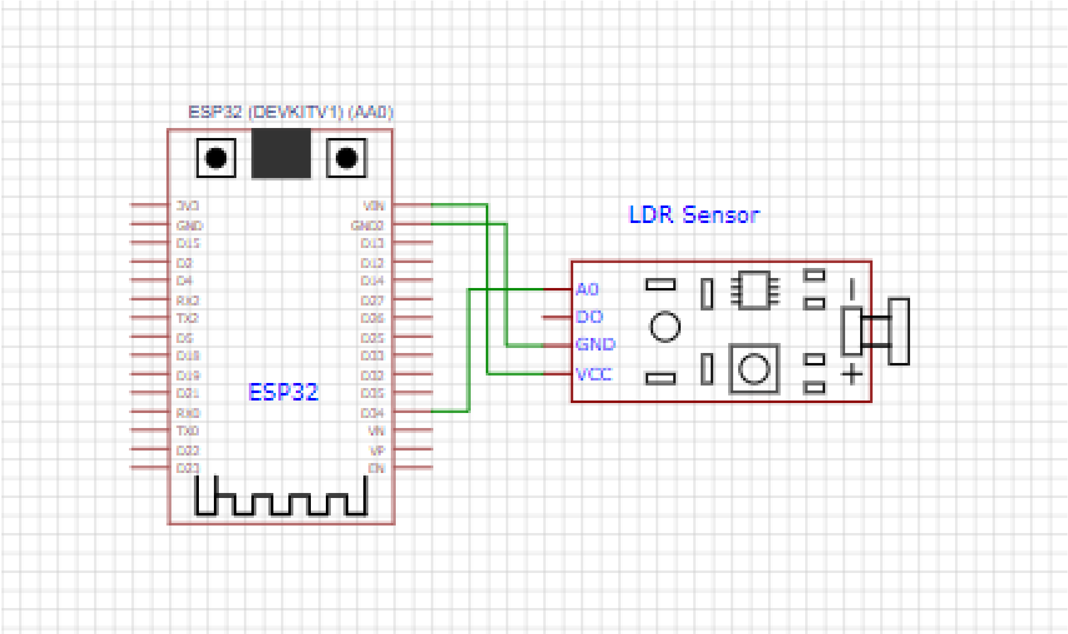
8.1. Kulasekara Y.B.M - 224111U(Leader)

Studying LDR Sensor and implementing the Modules into Box

As part of my contribution to the project, I was responsible for studying the functionality of the LDR sensor and implementing it into the system's hardware. The LDR sensor plays a crucial role in detecting when the runner crosses the finish line by identifying the laser beam interruption, which helps to capture the end time. I integrated the sensor into the system and designed the module box to securely house the components, ensuring accurate detection during operation. Additionally, I worked on configuring the system's visual feedback using LED bulbs in conjunction with the LDR, providing clear and reliable output for the runners. To enhance user interaction and system feedback, I configured a visual feedback system using three LED indicators in conjunction with the LDR sensor. The LEDs serve the following functions:

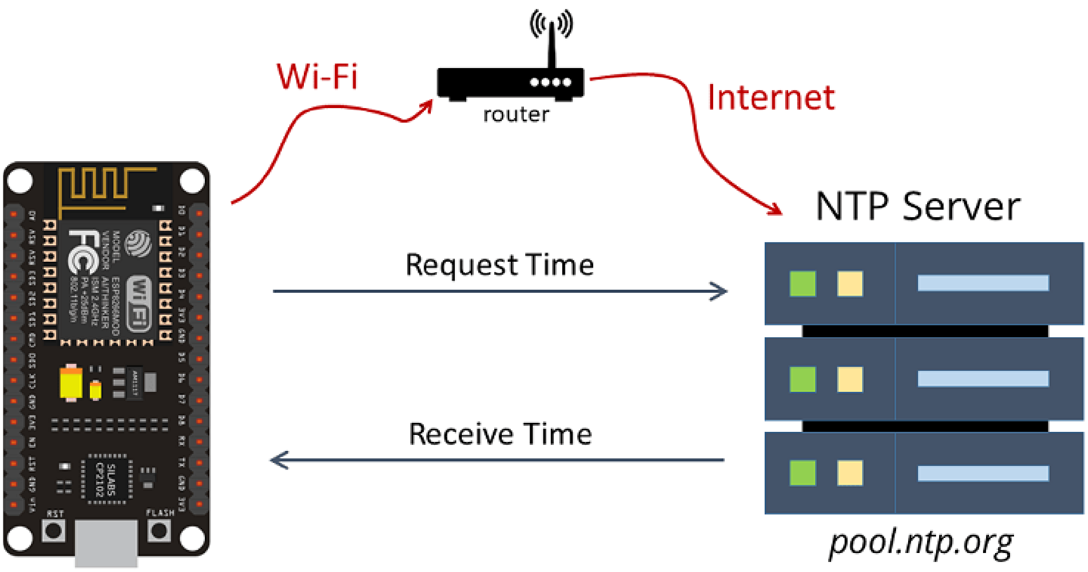
- **Red LED:** Indicates that the laser beam is not detected. This serves as a visual alert for potential issues with the laser or sensor alignment.
- **Green LED:** Illuminates when the laser beam is correctly detected by the LDR sensor. This provides confirmation that the sensor is functioning properly.
- **Blue LED:** Activates after the green LED has been continuously on for 10 seconds. This signals that the LDR sensor is fully operational and ready for accurate time calculation.

This integration was essential for ensuring the system's accuracy and smooth functioning.



Programming the time-calculating process

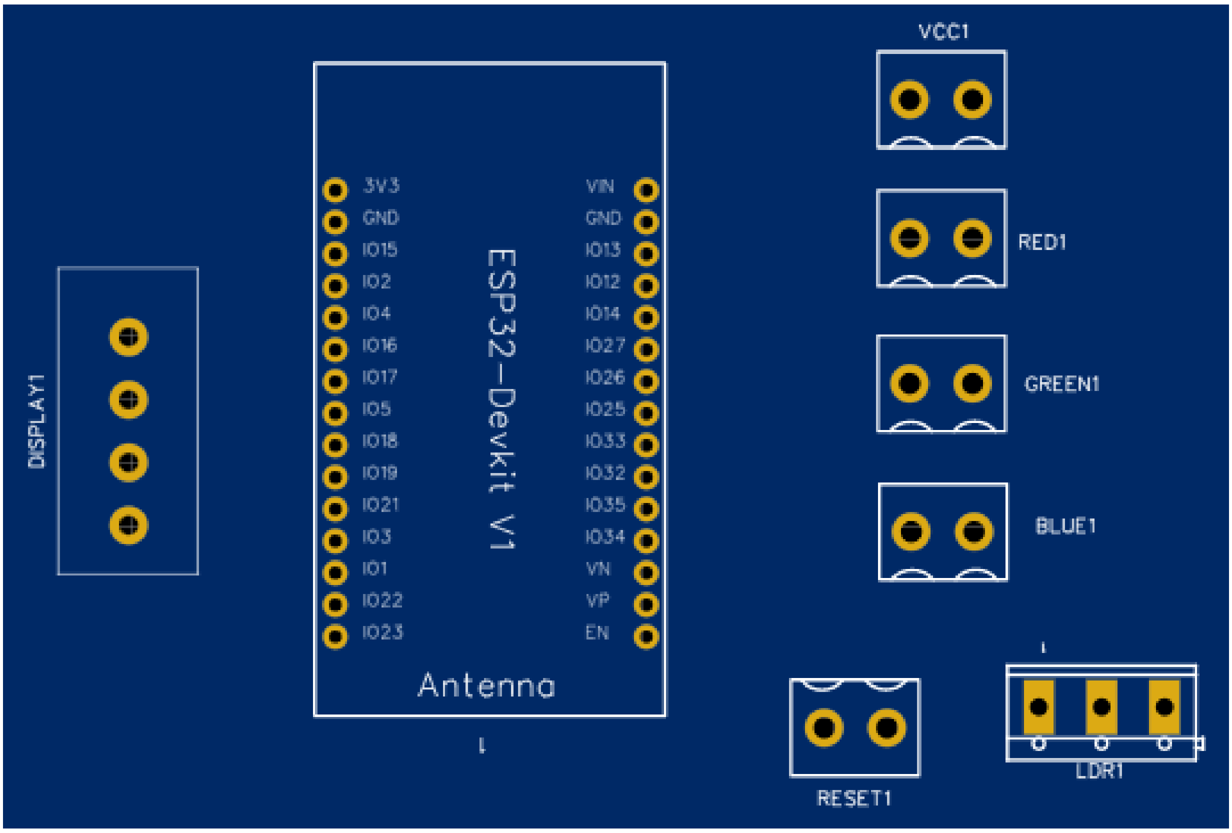
I was responsible for programming the time calculation in the system, which ensures precise timing during the runner's performance. To achieve this, I used the NTPClient library to get the current time from an NTP server, ensuring accurate timekeeping. The code I implemented monitors when the runner starts and finishes the race using sensor input. When the runner crosses the starting point, the system records the initial time (start time) from the NTP server, and as the race progresses, the code continuously checks the elapsed time using the Millis() function. Once the runner reaches the finish line and interrupts the laser beam, the system calculates the total race time. The logic ensures accurate time tracking by only triggering actions after specific intervals and resetting the timing if no detection occurs, providing reliable timing for the runner's performance.

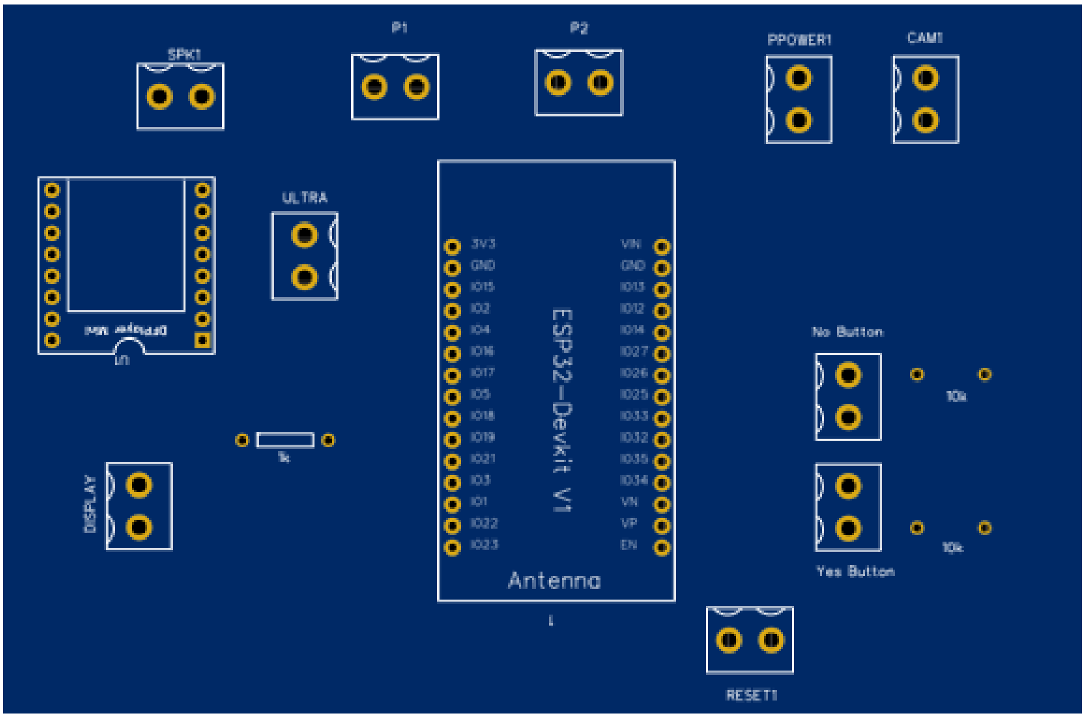

Wi-Fi
router
Internet
NTP Server
Request Time
Receive Time
pool.ntp.org

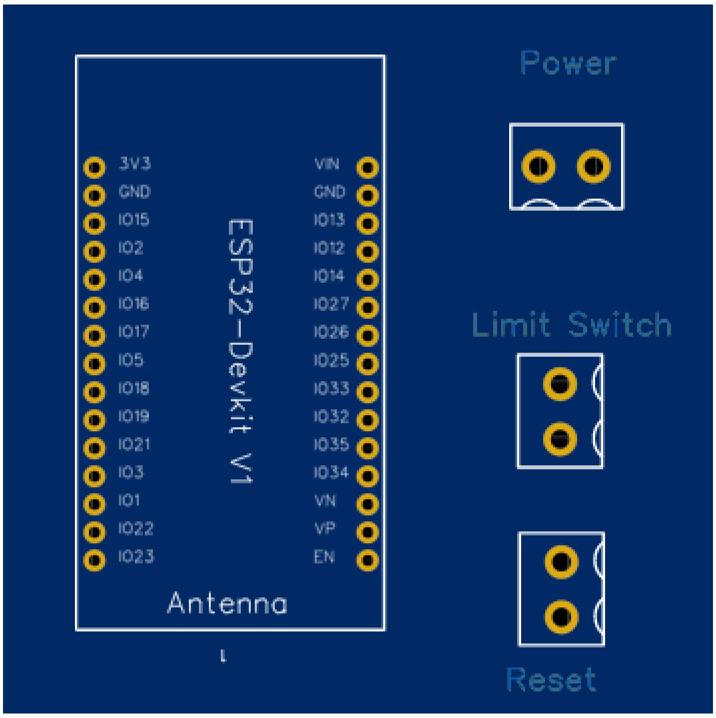
8.2. Jayaweera W.G.H.M.P.S.C - 224094R

Designing PCBs

I was responsible for designing the PCBs for the system using EasyEDA. The project required three ESP32 boards, and I designed three separate PCBs: one for the start point, another for the limit switch, and a final one for the endpoint. These custom PCBs were essential for ensuring proper connectivity and integration of the system components, allowing for smooth operation throughout the training track.

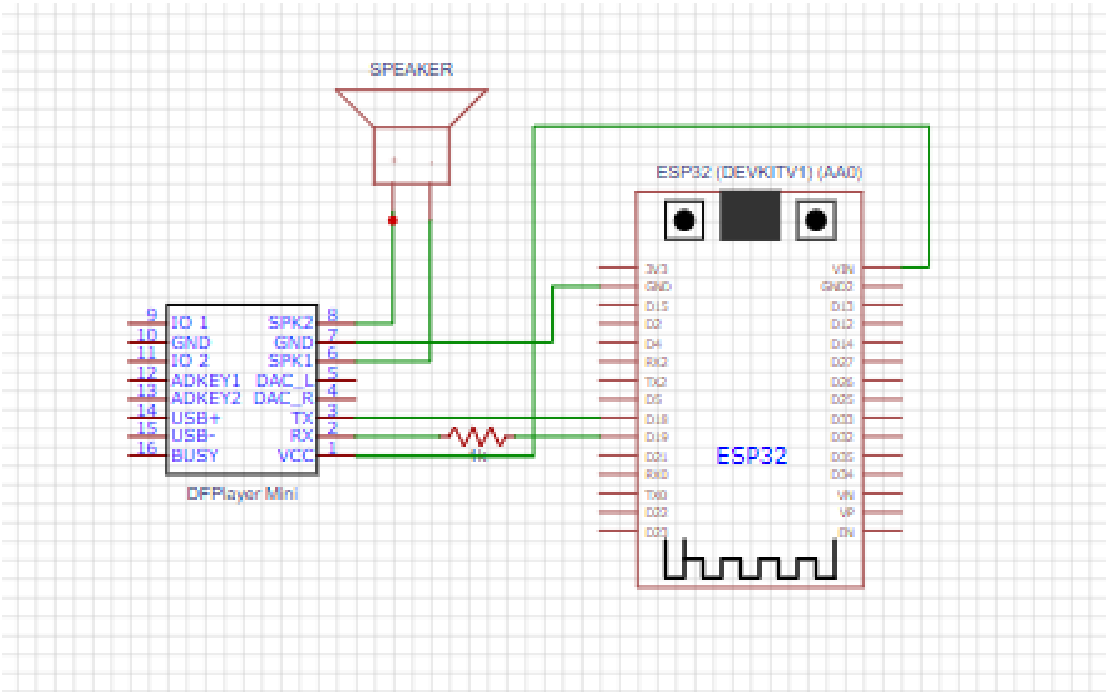



SPK1
P1
P2
PPOWER1
CAM1
ULTRA
DFPlayer Mini
U1
1k
DISPLAY
3V3
GND
IO15
IO2
IO4
IO16
IO17
IO5
IO18
IO19
IO21
IO3
IO1
IO22
IO23
VIN
GND
IO13
IO12
IO14
IO27
IO26
IO25
IO33
IO32
IO35
IO34
VN
VP
EN
ESP32–Devkit V1
Antenna
No Button
10k
Yes Button
10k
RESET1


3V3
GND
IO15
IO2
IO4
IO16
IO17
IO5
IO18
IO19
IO21
IO3
IO1
IO22
IO23
VIN
GND
IO13
IO12
IO14
IO27
IO26
IO25
IO33
IO32
IO35
IO34
VN
VP
EN
ESP32–Devkit V1
Antenna
Power
Limit Switch
Reset

Studying MP3-TF-16P Amplifier

This module allows audio playback and provides real-time audio feedback to runners through a 3W speaker. I focused on understanding its functionality, including file format support, storage capabilities, and playback control using UART. I also ensured its seamless integration with the microcontroller, enabling voice prompts and alerts during the runner's performance, which is crucial for enhancing the overall user experience.



Developing the Web Site

I was responsible for studying and programming the ESP32-CAM module, which is used to capture images of the runner at the starting point to ensure that the runner's position is correct before the race begins. I developed the system to automatically detect the runner's presence and capture photos at the right moment. This involved a deep understanding of the module's camera functions, including image capture and storage. Additionally, I programmed the ESP32-CAM using ESP-NOW to automatically take photos when the runner is detected in the starting position. These images are then stored and uploaded to Firebase for further review. To implement the accompanying website for viewing and analyzing the images, I used HTML, CSS, and JavaScript. This feature adds an extra layer of accuracy to the training system, allowing for more detailed analysis of the runner.

SoloRunner Dashboard Progress

Daily Training Overview ......

Date	Start Time	Leg Release Time	End Time	Timing	Block Position
2024-07-31	19:37:43:095	19:37:45:204	19:37:55:545	00:00:12:450	Click Here
2024-07-31	19:38:57:470	19:38:58:729	19:39:15:735	00:00:18:265	Click Here
2024-07-31	19:40:41:179	19:40:40:920	19:51:40:727	Faul Run	Click Here
2024-07-31	19:54:57:515	19:55:37:771	19:55:45:557	00:00:48:042	Click Here
2024-07-31	20:09:13:475	20:09:13:128	20:09:25:028	Faul Run	Click Here
2024-07-31	20:13:21:847	20:13:24:599	20:13:36:763	00:00:14:916	Click Here
2024-07-31	20:18:10:603	20:18:12:181	20:18:49:942	00:00:39:339	Click Here
2024-07-31	20:56:52:620	20:56:53:451	20:58:08:779	00:01:16:159	Click Here

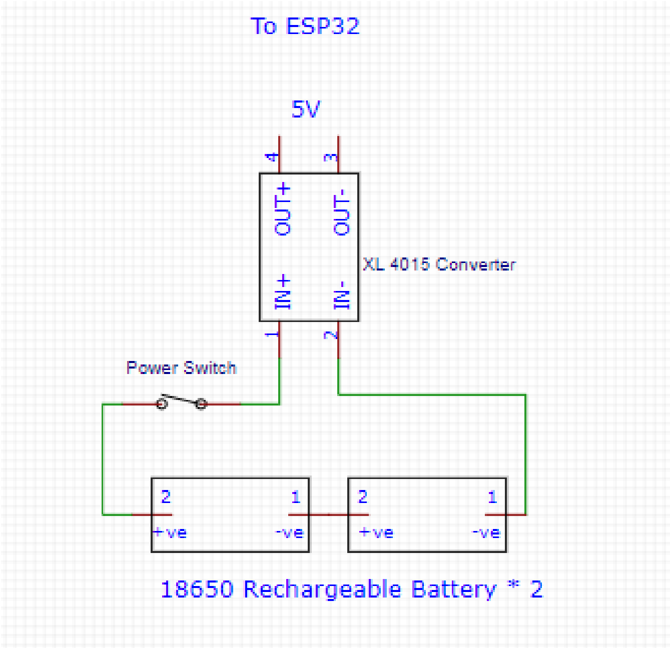
8.3. Kahaduwa T.S - 224097E

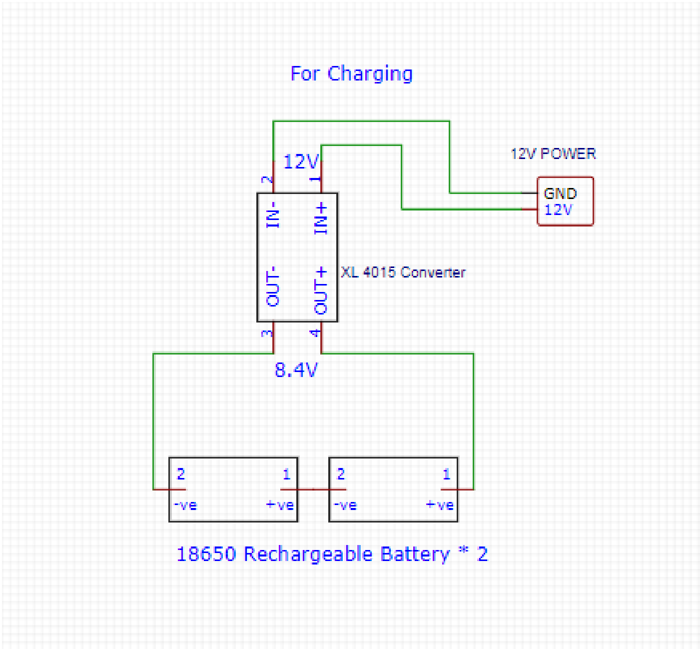
Studying the ESP 32 Camera and Programming

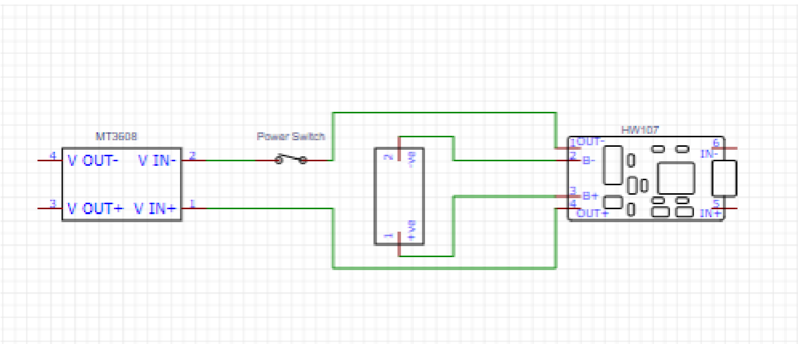
I was responsible for studying and programming the ESP32-CAM module, which is used to capture images of the runner at the starting point. This ensures that the runner's position is correct before the race begins. Responsible for programming the ESP32-CAM, I developed the system to automatically detect the runner's presence and capture photos at the right moment. I focused on understanding the module's camera functions, including image capture and storage. Additionally, I programmed the ESP32-CAM using ESP-NOW to automatically take photos when the runner is detected in the starting position. These images are stored and uploaded to Firebase for further review. This feature adds an extra layer of accuracy to the training system, allowing for more detailed analysis of the runner.

Studying and Implementing Power & Charging Modules

This involved selecting appropriate components to manage power distribution and ensure the reliable operation of all system elements. I focused on integrating a charging circuit for the 18650 rechargeable batteries, ensuring efficient and safe charging cycles. Additionally, I designed the power management system to regulate voltage levels and provide stable power supply to the ESP32 boards, sensors, and other peripherals. My work ensured that the system operated reliably and efficiently, with effective power management and charging capabilities.


To ESP32
5V
4
3
OUT+
OUT-
XL 4015 Converter
IN+
IN-
1
2
Power Switch
2
1
2
1
+ve
-ve
+ve
-ve
18650 Rechargeable Battery * 2

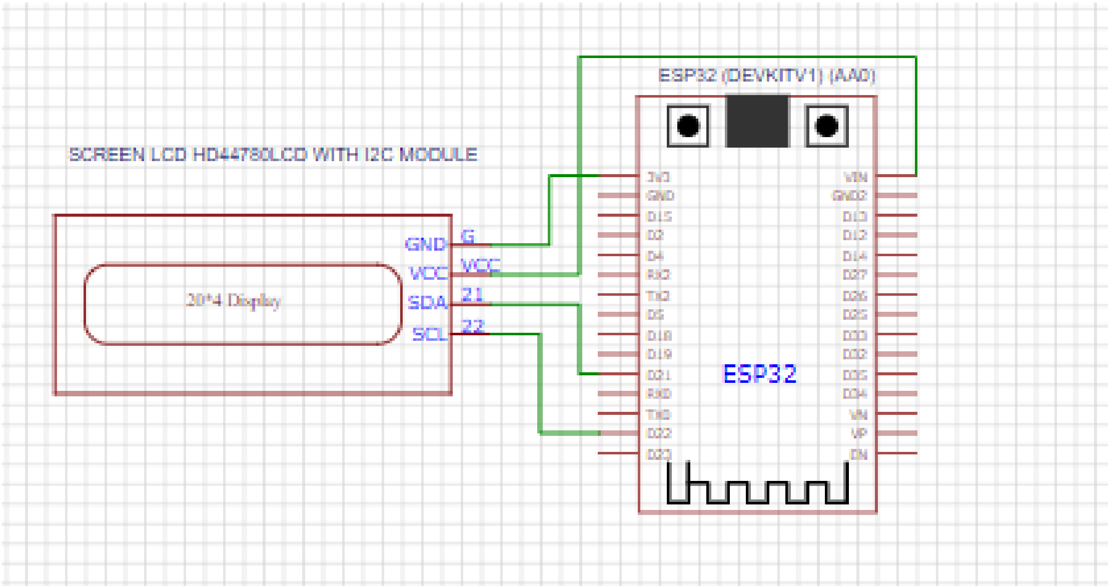

For Charging
12V POWER
12V
2
1
GND
12V
IN-
IN+
XL 4015 Converter
OUT-
OUT+
3
4
8.4V
2
1
2
1
-ve
+ve
-ve
+ve
18650 Rechargeable Battery * 2


MT3608
Power Switch
HW107
4
V OUT-
V IN-
2
3
V OUT+
V IN+
1
1 OUT-
2 B-
3 B+
4 OUT+
6 IN-
5 IN+

8.4 Jayaprabha P.H.J- 224085P

Studying 20*4 LCD Display at the start & Programming into ESP32 to provide a friendly user interface

I studied the 20x4 LCD and integrated it with the ESP32 to provide a user-friendly interface for the system. The 20x4 LCD offers ample space to display detailed information, such as the runner's start time, end time, and system status messages. I programmed the ESP32 to manage the data displayed on the LCD, ensuring clear and effective communication of critical system information. This integration enhances user interaction by providing a comprehensive view of the system's operational status and performance metrics.



Studying 16*2 LCD Display at the end & Programming into ESP32 to display the timing

The 16x2 LCD, with its dual-line, 16-character format, was used to provide clear and concise timing details, such as the runner's start time, end time, and total duration of the run. I integrated the display with the ESP32 to ensure that the timing data is accurately shown in real-time, enhancing the system's usability by providing immediate feedback to the runner and facilitating performance analysis.

Studying the I2C Module and implementing it with LCD

I studied the I2C module and implemented it to interface with both the 16x2 and 20x4 LCDs, optimizing the communication between the ESP32 and the displays. The I2C module simplifies wiring and reduces GPIO pin usage, facilitating efficient integration. I configured the module to handle data transfer for both display types, enabling the clear presentation of critical information such as timing, system status, and performance metrics. This implementation enhanced the system's functionality and user interface, providing flexible and readable output across different display sizes.

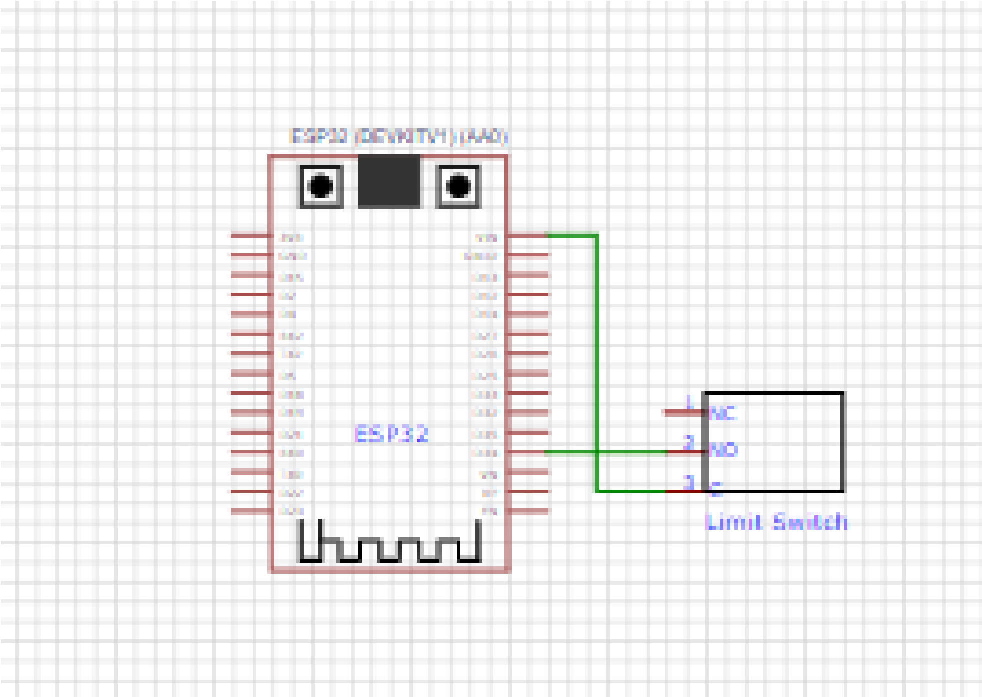
Implementing Firebase

I was responsible for implementing Firebase integration into the project. This involved configuring Firebase to handle data storage and retrieval for the system. I set up Firebase to manage and synchronize critical information such as runner performance data, timings, and fault statuses. This integration allows for real-time updates and seamless data management, enhancing the system's capability to track and store performance metrics efficiently. By leveraging Firebase, I ensured that the data is securely stored and easily accessible for further analysis and review.

8.5 Himaya K.S - 224076N

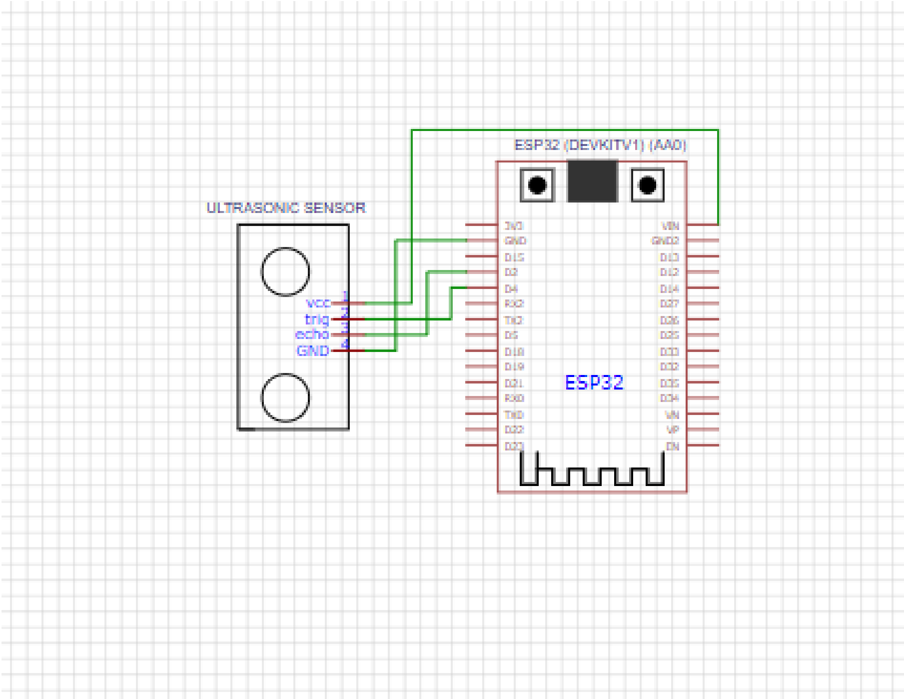
Studying the Limit Switch & programming into ESP32

I studied the limit switch and programmed it into the ESP32 to accurately detect the runner position and movement. The limit switch, used to monitor when the runner starts or leaves the designated position, required precise integration with the ESP32 to ensure reliable performance. I developed the code to read the switch's state and trigger appropriate actions based on the runner's interaction with the switch. This setup allows the system to accurately detect faults, such as early leg release, and ensure that the timing and positional data are correctly recorded and processed**.**



Studying the Ultrasonic Sensor & programming into ESP32

I studied the ultrasonic sensor and integrated it with the ESP32 to enhance the system's ability to detect and measure distances accurately. The ultrasonic sensor is crucial for detecting the runner's position and ensuring they are correctly aligned before starting. I developed the programming logic to interface with the sensor, read distance measurements, and trigger relevant system responses based on these measurements. This integration ensures precise detection of the runner’s presence at various points in the training track, contributing to accurate timing and performance monitoring.



9. References

- Daniels' Running Formula - 2nd Edition Paperback – October 1, 2005
- Serverless Web Applications with React and Firebase by Harmeet Singh – Senior
- Associate at Synchrony with expertise in UI development

- Sprinting Workouts by ATHLETE.X(Sprinting Workouts | The Sprinting Website)