National University of Sciences & Technology
School of Electrical Engineering and Computer Science
Department of Computing

# Semester Project
# Internet of Things (IoT)

Instructor: Ms. Aimen Javed

| Submitted By | CMS ID |
|---|---|
| **Abdullah Hussain** | **429345** |
| **Jaweria Manahil** | **419118** |
| **Saif Sheikh** | **429592** |
| **Sara Adnan** | **411228** |
| **Zainab Athar** | **405094** |

# Contents

# PROJECT DESCRIPTION: SMART STREET MANAGEMENT SYSTEM

## Introduction

This report elucidates the implementation of Smart Street Management System, which includes an automated Smart Street Lighting System followed by implementation of an innovative Vehicle Speed Detection system. Part A of the proposed system is *"Smart Street Lighting"* which is a power-saving substitute for the manual light switching system. It involves lights that adapt to surrounding illumination and operate by motion detection. The circuit is based on light intensity detection, under low illumination, the resistance of the LDR falls allowing current to flow through the circuit and whenever motion is detected, the LEDs light up. During daytime, the resistance of the LDR rises restricting the flow of current through the circuit, therefore, the LEDs will be turned off. This allows for reduction in power consumption and an efficient automated control over street lights requiring no manual handling.

Second part of this project deals with a "*Vehicle Speed Detection*" System. Over-speeding of vehicles has become one of the major causes for accidents and loss of many lives these days as people, especially the younger generation, have an incautious and reckless attitude towards road safety. Part-B of this report presents a system, developed for over-speeding detection and alerting offenders as well as authorities by turning on the buzzer automatically. Presently, RADAR guns or LIDARS are extensively used for overspeed detection, but it requires a person to pull the trigger for detecting the speed. In this work, it has been proved that automation provides efficient performance compared to a human handled system.

Github Link: [https://github.com/Saraadnan-sa/Smart-street-management.git](https://github.com/Saraadnan-sa/Smart-street-management.git)

## Objectives

We aim at designing a system that supersedes the existing street management systems and rectifies the shortcomings of these systems. The objectives of the proposed system are:

- Increase energy savings and reduce maintenance expenses due to automation.
- Turn on the street lights for a desirable amount of time based on ambient light levels and motion detection, so they are not worn out and have increased lifespan.
- Reduce the need for maintenance checks by minimal usage of street lights.
- Stamp out dependence on the labour force to operate the lights, human negligence would no longer affect the system.
- Contribute to environmental sustainability by lowering carbon footprint through reduced energy consumption and efficient lighting management.

- Minimise road accidents by providing an effective method for detecting speed violations by vehicles and alert passersby.
- Completely automate the speed detection process to reduce inefficiency in manual speed detection systems due to personal errors.
- Provide valuable data for law enforcement agencies to enforce speed limits and improve traffic management.
- Efficiently report speed violations via buzzer alarm to alert authorities which can be used for issuing fines and taking legal action.
- Data obtained from speed observations and traffic patterns can aid in urban planning and infrastructure development.

# Expected Outcomes

Currently, streetlights are switched on for the whole night and during the day, they are switched off. They are actuated by manual operators whose negligence at times leads to the lights being turned on for long time periods. Saving this energy is a primary outcome we tend to achieve, as these days energy resources are getting reduced day by day. Our project aims at being an energy-saving alternative to the existing conventional streetlights control system. We expect to achieve automation of the light switching by automatically turning the lights on at night based on vehicular motion, and turning them off during the day. The outcome will be reduction in costs and environmental impact as well as improved energy efficiency and safety in the area of implementation.

We also aim to develop a sub-system that detects over-speeding of vehicles on streets and alerts passersby in case of any speed limit violation. This is to discourage speeding behaviour through real-time monitoring of over-speeding incidents. It intends to replace the present conventional system that involves handheld speed detection guns to record a vehicle's speed and then manually inform authorities about the vehicle which is an ineffective method. By means of automated vehicle speed detection, we expect a reduction in traffic accidents and provision of road safety due to improved compliance with speed regulations by drivers.

# APPLICATION REQUIREMENTS

## Software

**1. WOKWI Platform:** Utilise WOKWI simulation platform for building and running circuit simulations, and testing the circuit to ensure proper functionality and interaction of components in a virtual environment.

**2. WiFi Connectivity:** Ensure the availability of a stable WiFi connection for running simulations of both the smart street lighting and vehicle speed detection systems.

**3. ThingSpeak:** ThingSpeak - a cloud-based repository and analytics platform - is required for storing, analysing, and visualising data.

**4. Git Version Control:** Implement Git for version control to track changes in the project.

**5. GitHub:** Required for hosting of git repositories, collaborative development, and project documentation.

**6. Virtual WiFi Configuration:** Set up virtual WiFi configurations in both systems to enable communication with ThingSpeak for data transmission.

**7. Libraries:**

- WiFi.h: Enables ESP32 to utilise its integrated Wi-Fi module.
- LiquidCrystal_I2C.h: Initialises and simplifies communication with LCD displays.
- ThingSpeak.h: Facilitates interaction with the ThingSpeak cloud platform for data storage and analysis.

**8. IDE (VS Code):** Required for code development, debugging, and maintenance of project files.

## Hardware

Hardware requirements will be covered in the next section for both sub-systems of the Street Management System.

*Note: Hardware is not actually implemented this section is just for information purpose. The system is simulated on Wokwi.*

| **SOFTWARE** | **HARDWARE** |
|---|---|
| **WOKWI Platform:** For building and running circuit simulations | **PIR Sensors:** For motion detection through infrared radiation |
| **WiFi Connectivity:** Stable WiFi connection for running simulations | **LDR**: To measure ambient light levels |
| **ThingSpeak**: For storing, analyzing, and visualizing data from both systems | **LCD:** To provide a visual interface to display info |

| | |
|---|---|
| **Git**: For version control to track project changes | **ESP32 Microcontroller:** To control various components based on program logic |
| **GitHub:** For hosting of Git repositories and collaborative development | **LEDs:** Emit light when powered |
| **Virtual WiFi:** To enable communication with ThingSpeak for data transmission | **Resistors:** Limit current through LEDs |
| **IDE**: For code development and debugging | **Push Button:** Simulates the behavior of a motion sensor |
| **Python**: Required for Data Cleaning & Data Summarization | **Piezoelectric Buzzer:** For generating alerts in case of overspeeding. |
| **Power BI:** For creating a dynamic dashboard to display infographics & valuable insights | |

# Part A - Smart Street Lighting System

## 1. Sensors and related components detail:

The project document should briefly describe the sensors and related electronic components required for the IoT application.

### Components Used:

**PIR Sensors (Passive Infrared Sensors):**
Function: Detects motion by the infrared radiation emitted by objects, including moving vehicles or people.

Quantity Used: 5

Usage in Project: The PIR sensors used in the project will trigger the street lights to turn on when motion is detected. We have strategically placed 5 to cover the desired area around the street lights. Each PIR sensor is allocated 3 out of 7 LEDs, with an overlap of 2 common LEDs among them.

## LDR (Light Dependent Resistor):

Function: Measures ambient light levels and uses illumination unit “lux” to represent surrounding light intensity.

Quantity Used: 1

Usage in Project: In our system, LDR will determine whether the ambient light levels are low or high indicating night and day times respectively. The street lights will remain off during the day and turn on at night.

## 20x4 LCD (Liquid Crystal Display):

Function: Provides a visual interface to display information.

Quantity Used: 1

Usage in Project: A 20x4 LCD has been utilised to display status information, including day and night times, LDR values, as well as motion detection information from the sensors

## ESP32 Microcontroller:

Function: Central processing unit for the IoT system, capable of wireless communication. It controls various components based on program logic.

Quantity Used: 1

Usage in Project: It processes information from the LDR and PIR sensors and controls the operation of the street lights based on program logic. It also communicates wirelessly (using Wi-Fi) for data transmission to cloud.

## LEDs:

Function: Emit light when powered and illuminate the surrounding area.

Quantity Used: 7

Usage in Project: In the circuit, we have 7 LEDs grouped into 3 overlapping sets corresponding to each of the 5 PIR sensors. These LEDs represent street lights, whenever a sensor detects motion, the corresponding 3 LEDs will turn on and illuminate the street for the passing vehicle/person.

## Resistors:

Function: Limit current flow through LEDs.

Quantity Used: 7

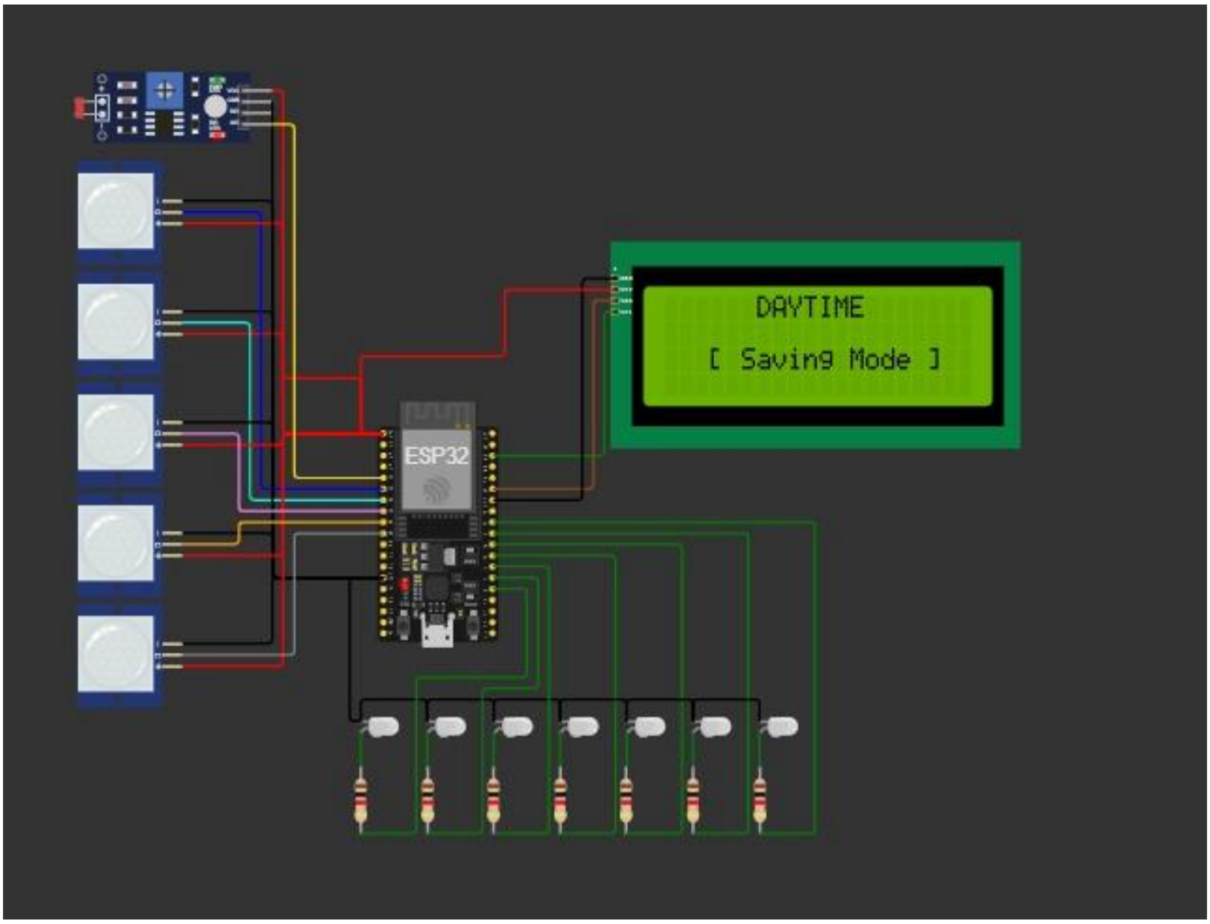
Usage in Project: We have connected 7 resistors in series with 7 LEDs to prevent excessive current through the LEDs and protect them.

## 2. Design and Development board and system Circuit Diagram:

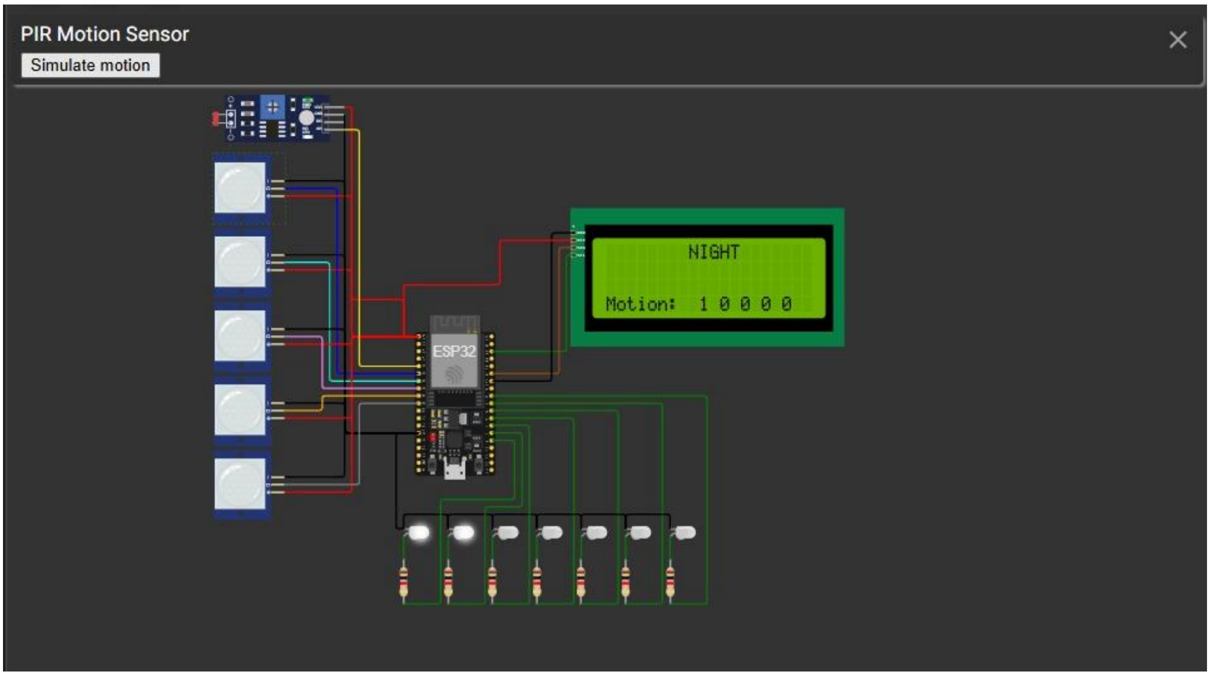
Provide a brief justification for using a particular development board and the complete system circuit diagram.

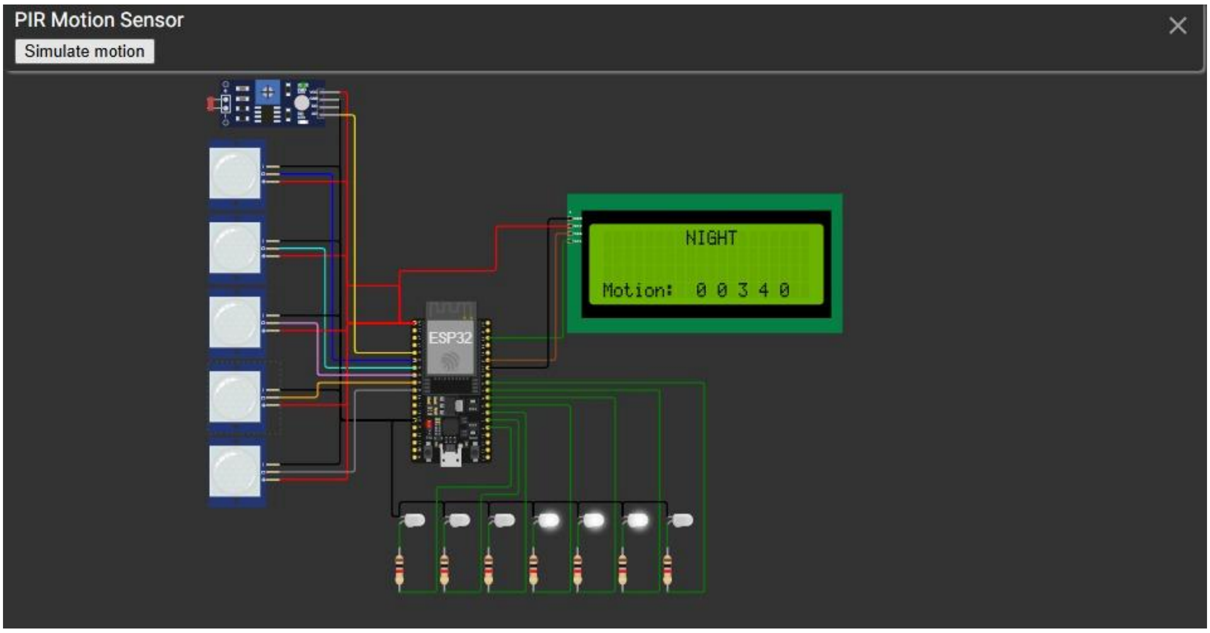
### Circuit Diagram


**During Daytime**



## During Night



## Development Board Justification (ESP32):

1. **Wireless Capabilities:**

ESP32 has built-in Wi-Fi connectivity, empowering our smart street lights system by connecting to the internet and sending data to the IoT platform for remote monitoring and control, allowing further integration and analysis.

2. **Power and Memory:**

Processing Power: ESP32 offers a dual-core processor running at up to 240MHz with sufficient processing power for handling PIR sensor inputs, making decisions based on programmed logic, and controlling multiple LEDs. This ensures efficient operation of our IoT system.

Memory: It also offers 520 KB of SRAM and 4-16MB of flash memory, allowing us to store our program code and any necessary data efficiently.

3. **Cost-Effectiveness:**

ESP32 is known for its relatively low cost compared to other microcontrollers with similar capabilities. This makes it a cost-effective choice for our project which incorporates multiple components.

4. **Community Support and Documentation:**

ESP32 has a large and active community, providing ample support and a wealth of libraries, tutorials and examples of ESP32 circuits. This makes it easier for us to troubleshoot issues, find resources, and enhance the functionality of our project.

5. **Peripheral Options:**

ESP32 offers a plethora of built-in peripherals that simplify the connection and control of various sensors and actuators. It reduces the need for additional external components and simplifies the overall circuit design.

6. **Energy Efficiency:**

ESP32 features various power-saving modes that can significantly reduce power consumption when not actively processing data or communicating. This is imperative for battery-powered applications like our smart street lights system, potentially extending the system's operational lifespan.

## Design Justification:

1. **Multiple Sensors:**

The use of multiple PIR sensors enhances the coverage area for motion detection, ensuring that the street lights respond effectively to vehicle movement.

**2. LDR:**

Incorporating an LDR provides a reliable way to distinguish between day and night, enabling the system to activate the street lights based on motion only when ambient light levels are low.

**3. LED Design:**

The design of using sets of three LEDs per PIR sensor ensures ample illumination for passing vehicles, providing a clear line of sight at the front with the consecutive lighting of three LEDs.

**4. LCD Display 20x4:**

The inclusion of an LCD provides a user-friendly interface for displaying important information about the system's status like the time of day, motion status at sensors, and the LDR illumination value. It enhances the project's usability and makes it more comprehensible.

**5. Energy Efficiency:**

The overall design, with the combination of sensors and conditional LED activation, contributes to energy efficiency. The system only activates when necessary, reducing unnecessary power consumption and saving costs.

## Circuit operation:

**Daytime:** LDR detects high ambient light. Since light intensity is greater than the threshold value of 100, ESP32 keeps all LEDs off.

**Nighttime:** LDR detects low ambient light. Since light intensity is lesser than the threshold value of 100, ESP32 enables motion detection via PIR sensors and corresponding lighting up of LEDs.

**Motion detection:** When a car passes by a PIR sensor, the associated set of 3 LEDs switches to full brightness. As the car keeps moving along the street, the LEDs would keep turning on due to motion detection by the PIR sensors.

**Timeout:** After the preset delay duration (default: 5 seconds for PIR sensor), the triggered LEDs turn off if no further motion is detected within a specific timeframe.

# 3. Details of communication technology and protocols used at physical, internet and application layers.

Please also discuss possible network topologies if the system needs to be scaled up.

## Communication Technology and Protocols:

**Physical Layer:**
WiFi is being used in order to transmit the data from the Wokwi ESP32 to ThingSpeak.

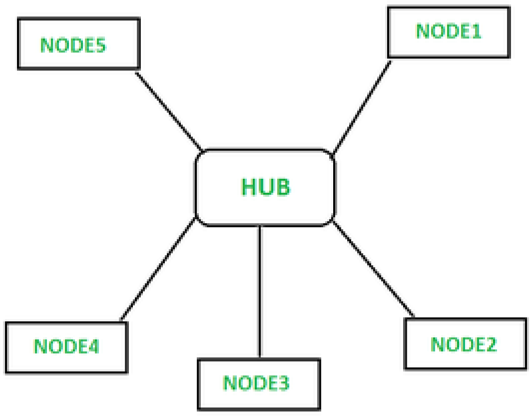
**Internet Layer:**
At the internet layer, the Internet Protocol (IP) is fundamental for addressing and routing data packets between devices connected to the internet. In our project, the ESP32 uses IP to communicate over the internet.

**Application Layer:**
HTTP (Hypertext Transfer Protocol) is used in the application layer for communication of data between the Wokwi ESP32 and ThingSpeak. The communication flow involves the ESP32 sending an HTTP request to send data to ThingSpeak servers, where the data is displayed.

## Network Topology of Current System:

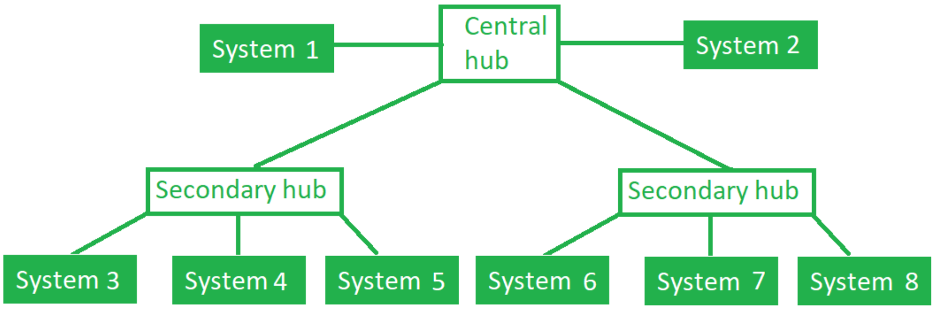
Our system follows the star topology as there is a central hub or control unit - ESP32 microcontroller in our case - and all other sensors, actuators and external platforms are connected to this control hub. ESP32 contains the program logic for managing communication with all other components.

This is a representation of star topology:



## Network Topology of Scaled up System:

The scaled up system will have implementation of our street management system in most streets of the city. Every street will have a local control hub or intermediary with storage and processing capabilities, and this LCH will be coordinating communication with all sensors and actuators in our system. It is further connected to the central hub or cloud platform. This hierarchy will be followed in all streets where our system will be implemented. After connecting all the hierarchies, we will have a tree topology.




# 4. Analysis of Power consumption:

The detailed description of power consumption of every sensor should be included and the approach that can be used for the optimal consumption of power should be described.

## PIR Motion Sensor:

**Power Consumption**

Idle State = 40 µA

Active State = 400 µA

**Sensor Details:** Passive infrared motion detector PIR-STD-LP

| Movement alarm unit PIR-STD-LP | |
|---|---|
| Power supply | 3...5 V DC |
| Input current | Idle output "H" 40 µA<br>Active output "L" 400 µA |
| Coverage | Approx. 4 to 12 m, 5-step adjustable<br>Adjustable in steps through 4 soldered joints |
| Signal bandwidth | 0,4...10 Hz |
| Coverage angle | Horizontal ±50°<br>Vertical ±30° |
| Digital output | Open Collector<br>max 5 V, 20 mA |
| Analog output | 0V...Vcc -0,5 V |
| Operating temperature range | -20... 60 °C |
| Ambient humidity | 0..90% RH<br>Dew formation not allowed |
| Dimensions | 25 x 25 x 26 mm<br>Mounting diameter Ø 24 mm |
| CE-Conformance | 2014/30/EU |
| EMV-noise emission | EN 61000-6-3:2011 |
| EMV-noise withstanding | EN 61000-6-1:2007 |
| Ordering No. | PIR-STD-LP |
| Rights reserved for change in technical data due to technological advance-ments! | |

**Description:**

When an object moves within the sensor's field of view, it causes a temporary imbalance in the infrared radiation reaching the sensor. The PIR sensor detects this change and produces an output signal. This signal is often in the form of an increase in electrical current.

**Reference**:
https://shop.bb-sensors.com/en/Motion-Brightness/PIR-motion-sensor-power-consumption-40-A.html

# LDR Sensor:

## Power Consumption

Active State: 100mA

| Test Conditions |
|---|
| Reflecting object – 73 mm × 83 mm Kodak 90% grey card, 100 mm distance, $V_{LEDA}$ = 3 V, LDRIVE = 100 mA, PPULSE = 8, PGAIN = 4x, PPLEN = 8 µs, LED_BOOST = 100%, open view (no glass) above the module. |

**Description:**

During the day the resistance through LDR is high due to which the current consumption is low, however, at night the LDR resistance is low and current starts flowing through it due to which current consumption is high at night.

**Reference**:https://cdn.sparkfun.com/assets/learn_tutorials/3/2/1/Avago-APDS-9960-datasheet.pdf

# Part B - Vehicle Speed Detection System

## 1. Sensors and related components detail:

The project document should briefly describe the sensors and related electronic components required for the IoT application.

### Components Used:

#### Push Button:

Function: Simulating the behaviour of a motion sensor.

Quantity: 2

Usage: Push buttons are used in the system to simulate the movement of vehicles on the road, and consequently calculate their speeds to check overspeeding vehicles.

#### Piezoelectric Buzzer:

Function: A piezoelectric buzzer converts electrical energy into mechanical vibrations, generating audible sound waves for alarms, alerts, and feedback in electronic devices.

Quantity: 1

Usage: The buzzer is used to sound an alarm if the speed of a vehicle exceeds the defined limit.

#### Resistor:

Function: Limit the flow of current through the push buttons.

Quantity: 2

Usage: 2 resistors are used in the circuit in order to control the flow of current through the push buttons.

### ESP32 Microcontroller:

Function: The ESP32-DevKit-C-V4 serves as the brain of the system, functioning as a microcontroller.

Quantity: 1

Usage: The ESP32 manages the entire speed detection system, ensuring accurate data processing and control over the associated components. It processes the data collected from the push buttons, calculates the speed, and triggers the piezoelectric buzzer if an overspeeding event is detected.

### LCD Display 20x4:

Function: Provides a visual interface to display information

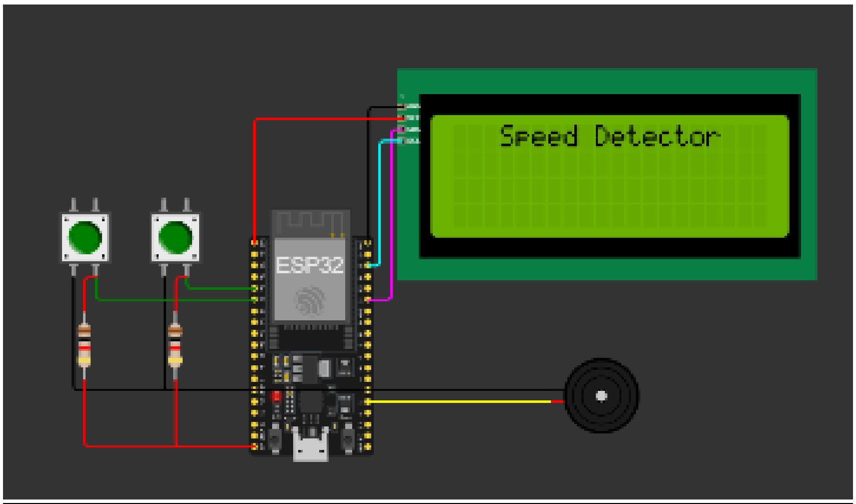
Quantity: 1

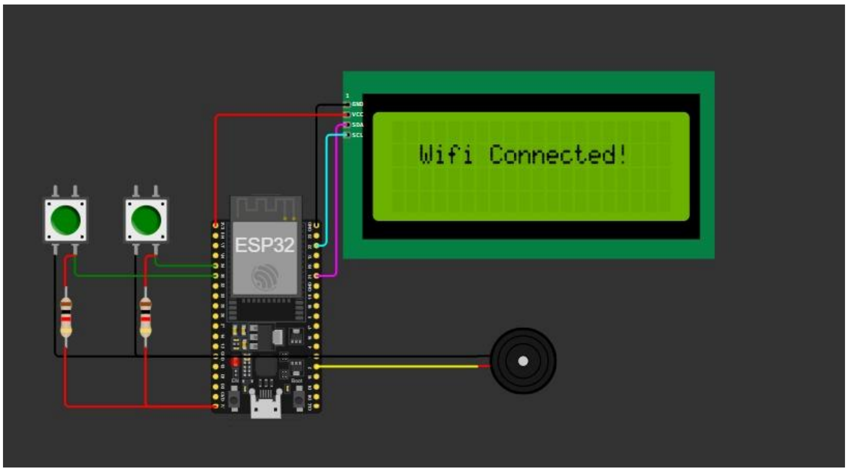
Usage: The LCD2004 serves as the user interface, displaying relevant information such as the speed of the vehicle and indicating whether it is overspeeding or not.

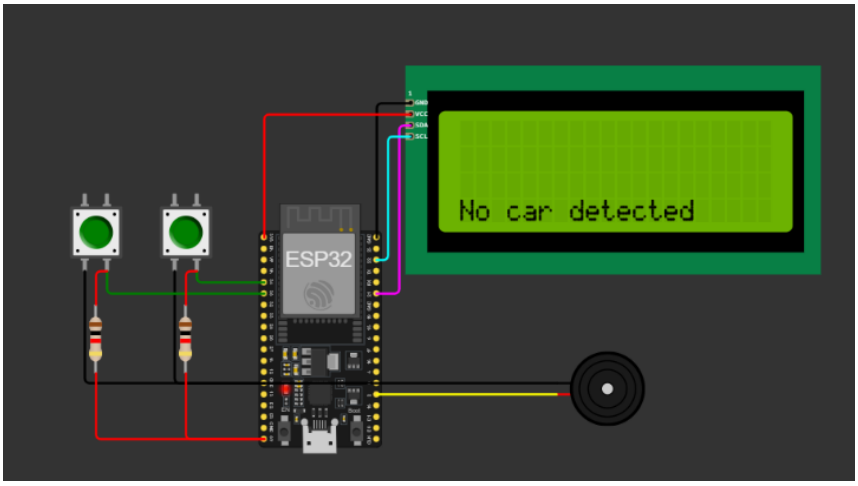
## 2. Design and Development board and system Circuit Diagram:

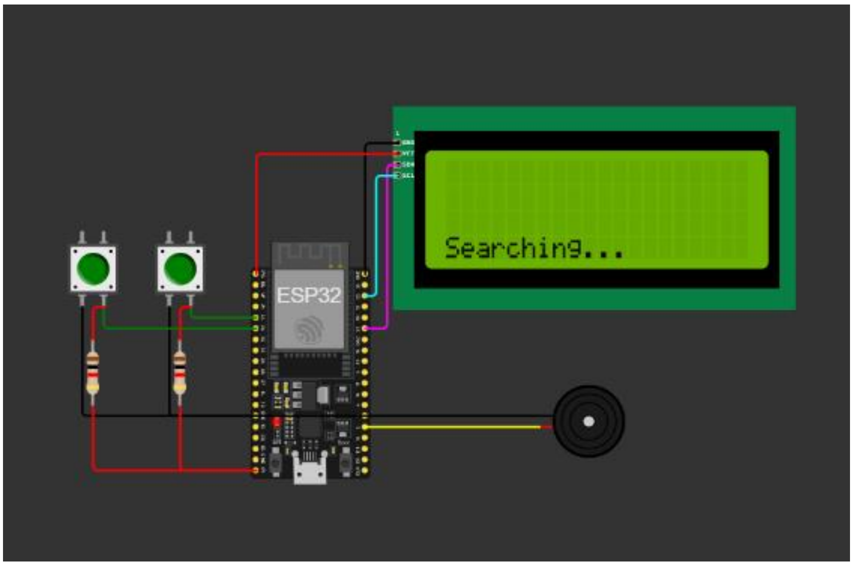
Provide a brief justification for using a particular development board and the complete system circuit diagram.

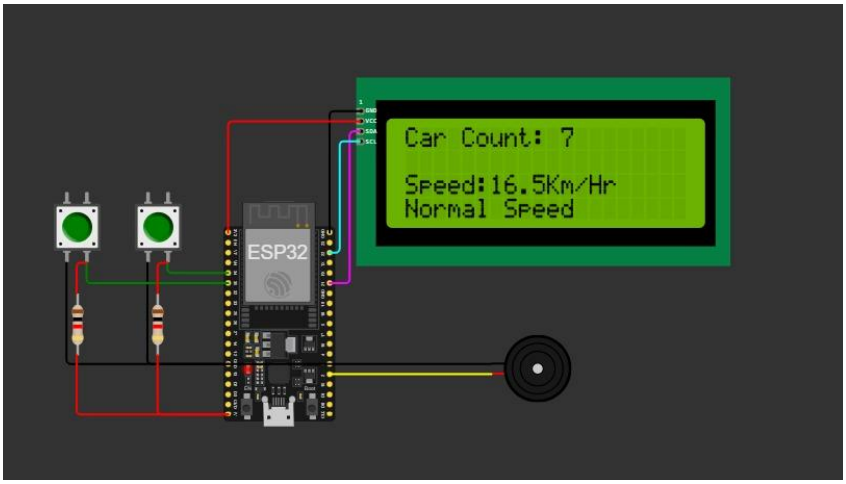
## Circuit Diagram

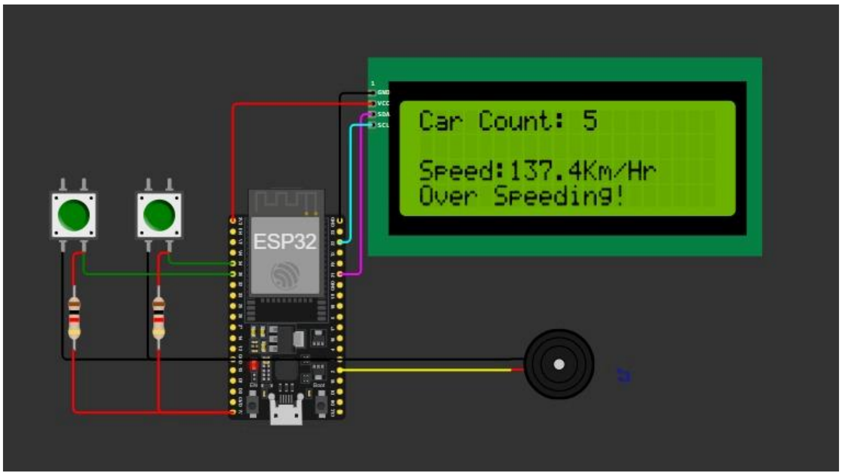



ESP32
No car detected


ESP32
Searching...






## Development Board Justification (ESP32):

**1. Wireless Capabilities:**

The ESP32, equipped with integrated Wi-Fi connectivity, enhances our Vehicle Speed Detection system by enabling the system to connect to the internet, facilitating the transmission of data to the IoT platform for remote monitoring and control.

**2. Power and Memory:**

Processing Power: ESP32 offers a dual-core processor running at up to 240MHz with sufficient processing power helping us make decisions based on programmed logic.

Memory: It also offers 520 KB of SRAM and 4-16MB of flash memory, allowing us to store our program code and any necessary data efficiently.

**3. Cost-Effectiveness:**

The cost-effectiveness of the ESP32 stems from its affordability in comparison to other microcontrollers offering similar capabilities. This characteristic makes it a budget-friendly selection for our project.

4. **Community Support and Documentation:**

ESP32 has a large and active community, providing ample support and a wealth of libraries, tutorials and examples of ESP32 circuits. This makes it easier for us to troubleshoot issues, find resources, and enhance the functionality of our project.

5. **Peripheral Options:**

ESP32 offers a plethora of built-in peripherals that simplify the connection and control of various sensors and actuators. It reduces the need for additional external components and simplifies the overall circuit design.

6. **Energy Efficiency:**

The ESP32 incorporates diverse power-saving modes that can effectively diminish power consumption during idle periods or when not engaged in data processing or communication. This aspect is crucial for applications powered by batteries, such as our vehicle speed detection system, potentially prolonging the operational lifespan of the system.

## Design Justification:

1. **Push Buttons:**

Push buttons are used to effectively simulate the behaviour of a motion sensor. The push button is pressed whenever a vehicle crosses the button - this allows us to calculate the time interval in which the vehicle crossed the two measurement points, thereby allowing us to determine the speed of the vehicle.

2. **Piezoelectric Buzzer:**

A piezoelectric buzzer is used to generate an alert whenever a vehicle is found to be overspeeding. This allows us to give instantaneous feedback to both the driver and the authorities that a vehicle is overspeeding.

3. **LCD Display:**

The inclusion of an LCD provides a user-friendly interface for displaying important information about the system's status like the time of day, motion status at sensors, and the LDR illumination value. It enhances the project's usability and makes it more comprehensible.

5. **Resource Efficiency:**

Our vehicle speed detection system completely automates the speed detection process to reduce inefficiency in manual speed detection systems. Using an automated IoT based system helps us reduce cost and increase efficiency by replacing manual labour with sensors and automated alerts.

## Circuit operation:

**Speed Detection:** When a car crosses a sensor - push button in our case - the time is recorded. When the car crosses the second sensor, the time difference between the two recorded times is calculated. The speed of the vehicle is then calculated by dividing the distance between the two sensors by the time difference.

**Overspeeding alert:** If the speed of a vehicle crosses the speed limit of the street - arbitrary value of 50 km/h in our case - the piezoelectric buzzer goes off, generating an alert to both the violator as well as the authorities of the overspeeding in the area.

# 3. Details of communication technology and protocols used at physical, internet and application layers.

Please also discuss possible network topologies if the system needs to be scaled up.

## Communication Technology and Protocols:

**Physical Layer:**
WiFi is being used in order to transmit the data from the Wokwi ESP32 to ThingSpeak.

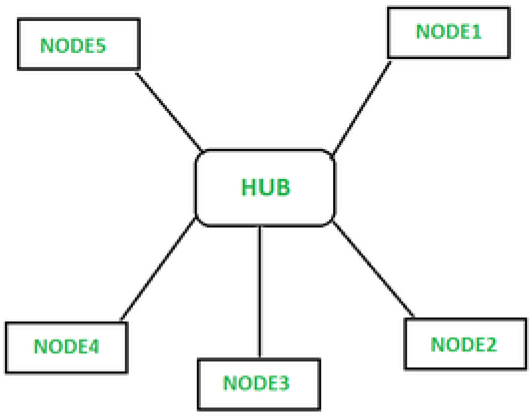
**Internet Layer:**
At the internet layer, the Internet Protocol (IP) is fundamental for addressing and routing data packets between devices connected to the internet. In our project, the ESP32 uses IP to communicate over the internet.

**Application Layer:**
HTTP (Hypertext Transfer Protocol) is used in the application layer for communication of data between the Wokwi ESP32 and ThingSpeak. The communication flow involves the ESP32 sending an HTTP request to send data to ThingSpeak servers, where the data is then processed and displayed.

## Network Topology of Current System:

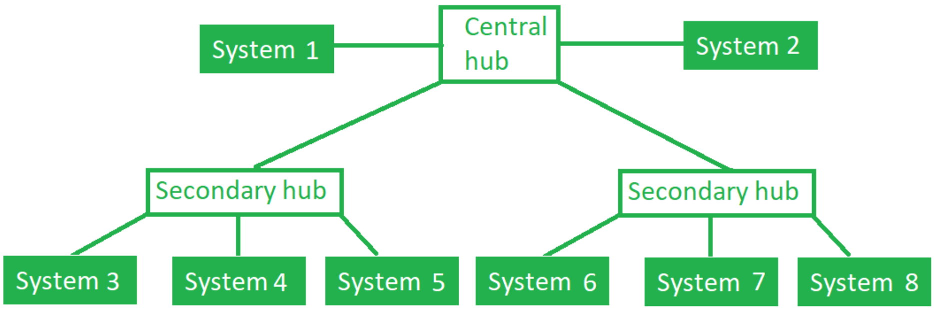
Our system follows the star topology as there is a central hub or control unit - ESP32 microcontroller in our case - and all other sensors, actuators and external platforms are connected to this control hub. ESP32 contains the program logic for managing communication with all other components.

This is a representation of star topology:




## Network Topology of Scaled up System:

The scaled up system will have implementation of our street management system in most streets of the city. Every street will have a local control hub or intermediary with storage and processing capabilities, and this LCH will be coordinating communication with all sensors and actuators in our system. It is further connected to the central hub or cloud platform. This hierarchy will be followed in all streets where our system will be implemented. After connecting all the hierarchies, we will have a tree topology.




# 4. Analysis of Power consumption

Detailed description of power consumption of every sensor should be included and the approach that can be used for the optimal consumption of power should be described.

## Sensors used:

### 1) PIR Motion Sensor:

Power Consumption
Idle State = 40 µA
Active State = 400 µA

**Sensor Details:**
Passive infrared motion detector PIR-STD-LP.

| Movement alarm unit PIR-STD-LP | |
|---|---|
| Power supply | 3...5 V DC |
| Input current | Idle output "H" 40 µA<br>Active output "L" 400 µA |
| Coverage | Approx. 4 to 12 m, 5-step adjustable<br>Adjustable in steps through 4 soldered joints |
| Signal bandwidth | 0,4...10 Hz |
| Coverage angle | Horizontal ±50°<br>Vertical ±30° |
| Digital output | Open Collector<br>max 5 V, 20 mA |
| Analog output | 0V...Vcc -0,5 V |
| Operating temperature range | -20... 60 °C |
| Ambient humidity | 0..90% RH<br>Dew formation not allowed |
| Dimensions | 25 x 25 x 26 mm<br>Mounting diameter Ø 24 mm |
| CE-Conformance | 2014/30/EU |
| EMV-noise emission | EN 61000-6-3:2011 |
| EMV-noise withstanding | EN 61000-6-1:2007 |
| Ordering No. | PIR-STD-LP |
| Rights reserved for change in technical data due to technological advance-ments! | |

**Reference**:
**https://shop.bb-sensors.com/en/Motion-Brightness/PIR-motion-sensor-power-consumption-40-A.html**

# Tools and Technologies

- Wokwi
- ThingSpeak
- Vscode
- Git and Github
- C++ Programming Language in WOKWI IDE

# Impact on Society and Environment

The integration of our Smart Street Lighting System with Vehicle Speed Detection holds the promise of ushering in numerous positive changes for both society and the environment. Some key impacts of our project include:

**Efficient Energy Utilization:**
A key advantage of our project lies in its emphasis on energy conservation. By ensuring street lights remain inactive during daylight hours, we actively contribute to energy preservation - which not only curtails electricity consumption but also aligns with our commitment to fostering sustainable practices.

**Mitigating Light Pollution:**
Our project's intelligent design aims to minimize light pollution by activating street lights only when necessary, providing ample illumination for safety without the unnecessary lighting of other street lights. This approach helps preserve the natural environment and prevents disturbances to both people and wildlife by exposure to excessive artificial light.

**Advancing Safety on the Streets:**
The Smart Street Lighting System plays a crucial role in enhancing road safety. As a vehicle approaches, the system intelligently illuminates the surroundings, benefiting both drivers and pedestrians. This initiative underscores our dedication to reduced energy consumption and creating safer streets for everyone.

**Traffic Speed Management:**
The integration of the Vehicle Speed Detection System contributes significantly to regulating traffic speed. By issuing alerts when a vehicle exceeds the defined speed limit, we actively participate in making roads safer and reducing the likelihood of accidents.

**Data-Driven Urban Planning:**
Beyond its immediate benefits, our system generates valuable data on traffic patterns and vehicle speeds. This data can be instrumental in urban planning, providing authorities with insights for informed decision-making regarding road infrastructure and traffic management strategies.

**Setting a Sustainable Eco-friendly System:**
Through the implementation of smart and energy-efficient systems, our project sets an example for other municipalities and communities to follow suit. We envision a broader cultural shift towards sustainability, where eco-friendly practices become integral to urban development.