CanDrive
Smart Dashboard

Submitted By:

Ajay Singh (240840130002)

Astitva Srivastava (240840130006)

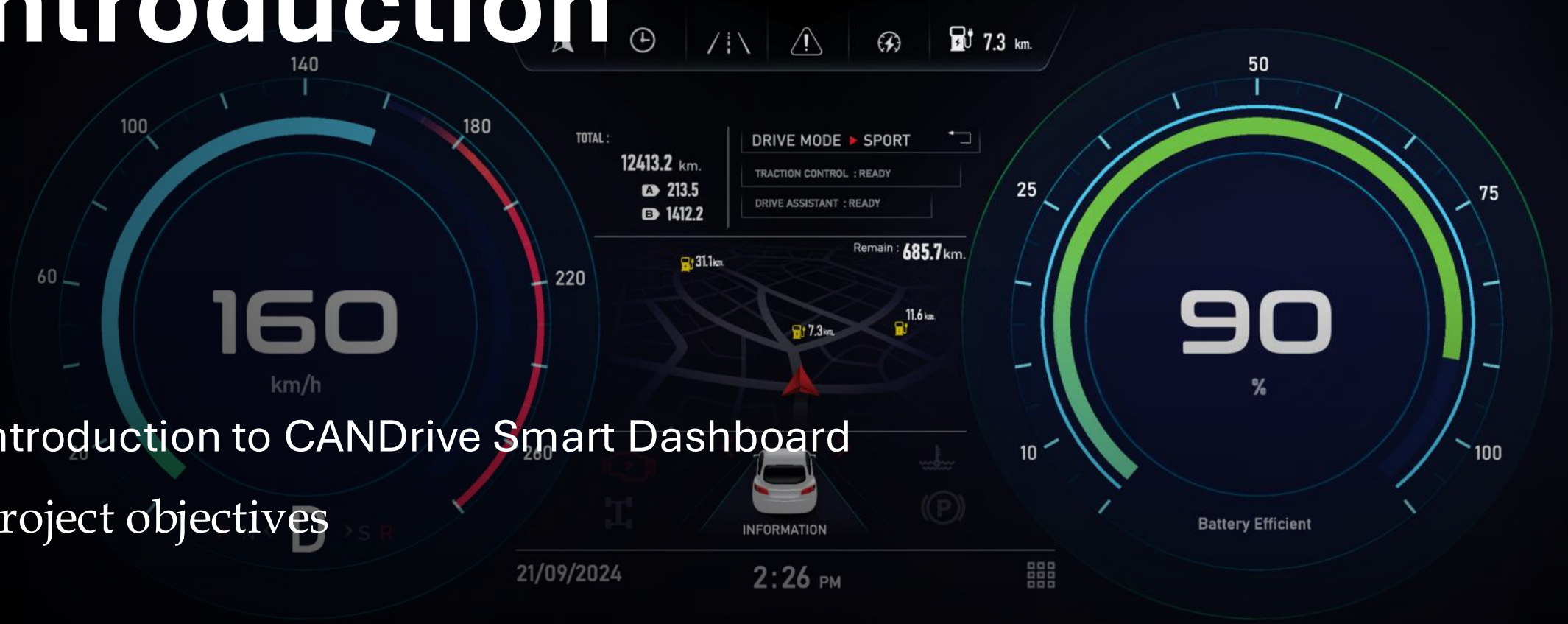
Hansraj Sahani (240840130009)

Rohan Jagtap (240840130011)

Nivya Ashokkumar (240840130023)

Content Outline

Introduction



- Introduction to CANDrive Smart Dashboard
- Project objectives

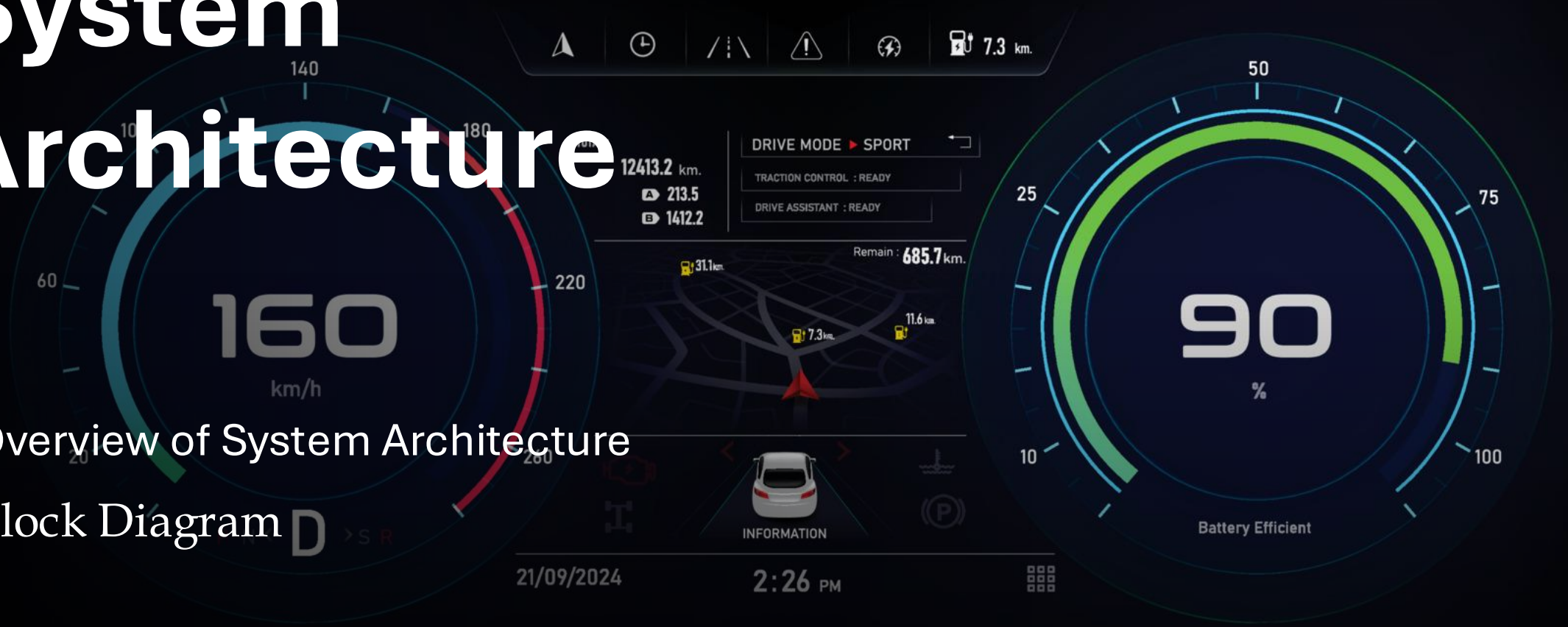
Introduction to CANDrive Smart Dashboard

- Modern vehicles require **real-time monitoring** for **performance optimization, predictive maintenance, and safety enhancements**.
- Traditional systems rely on **manual diagnostics** or **basic onboard diagnostics (OBD)**, which may lack real-time insights.
- The **CANDrive Smart Dashboard** is an **IoT-based vehicle monitoring system** that provides **real-time data acquisition, processing, and cloud visualization**.
- The system integrates:
  - **FreeRTOS** – Ensures efficient multitasking for sensor data processing.
  - **CAN Bus Communication** – Facilitates high-speed data transfer between STM32 microcontrollers.
  - **MQTT & Thingsboard** – Enables cloud-based remote monitoring and analytics.
- By leveraging **IoT and embedded systems**, the project aims to enhance **predictive maintenance, operational efficiency, and vehicle safety**.

Project Objectives

- **Develop** a **real-time vehicle monitoring system** with efficient data acquisition and cloud-based visualization.
- **Utilize FreeRTOS** to manage multiple concurrent tasks efficiently.
- **Implement CAN communication** for seamless data transfer between microcontrollers.
- **Integrate ESP32 with MQTT** for cloud connectivity and dashboard visualization via **Thingsboard**.
- **Improve predictive maintenance** by monitoring key vehicle parameters such as **motor temperature, speed, distance, and door status**.

System Architecture

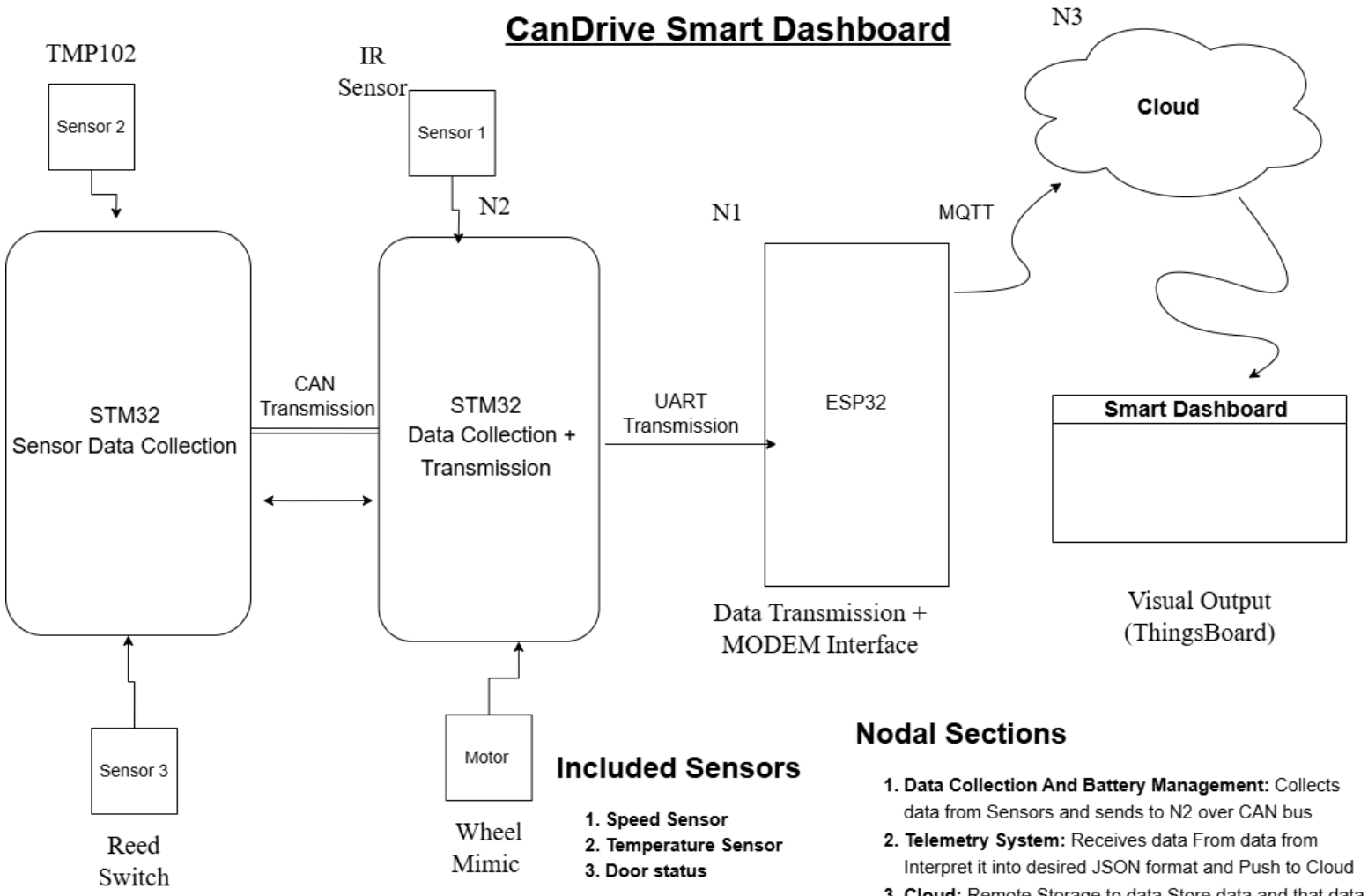


- Overview of System Architecture
- Block Diagram

Overview of System Architecture

- The **CANDrive Smart Dashboard** consists of **three main processing units**:
  - **STM32 Discovery Board 1** – Collects sensor data.
  - **STM32 Discovery Board 2** – Processes and relays data.
  - **ESP32** – Transmits data to the cloud.
- **Sensors Used:**
  - **TMP103** – Monitors motor temperature.
  - **IR Sensor** – Tracks wheel revolutions for speed and distance calculation.
  - **Reed Switch** – Detects door status.
- **Communication Protocols:**
  - **CAN Bus** – Facilitates fast and reliable communication between STM32 boards.
  - **UART** – Enables data exchange between STM32 and ESP32.
  - **MQTT** – Transfers sensor data to the cloud for visualization.

System Block Diagram



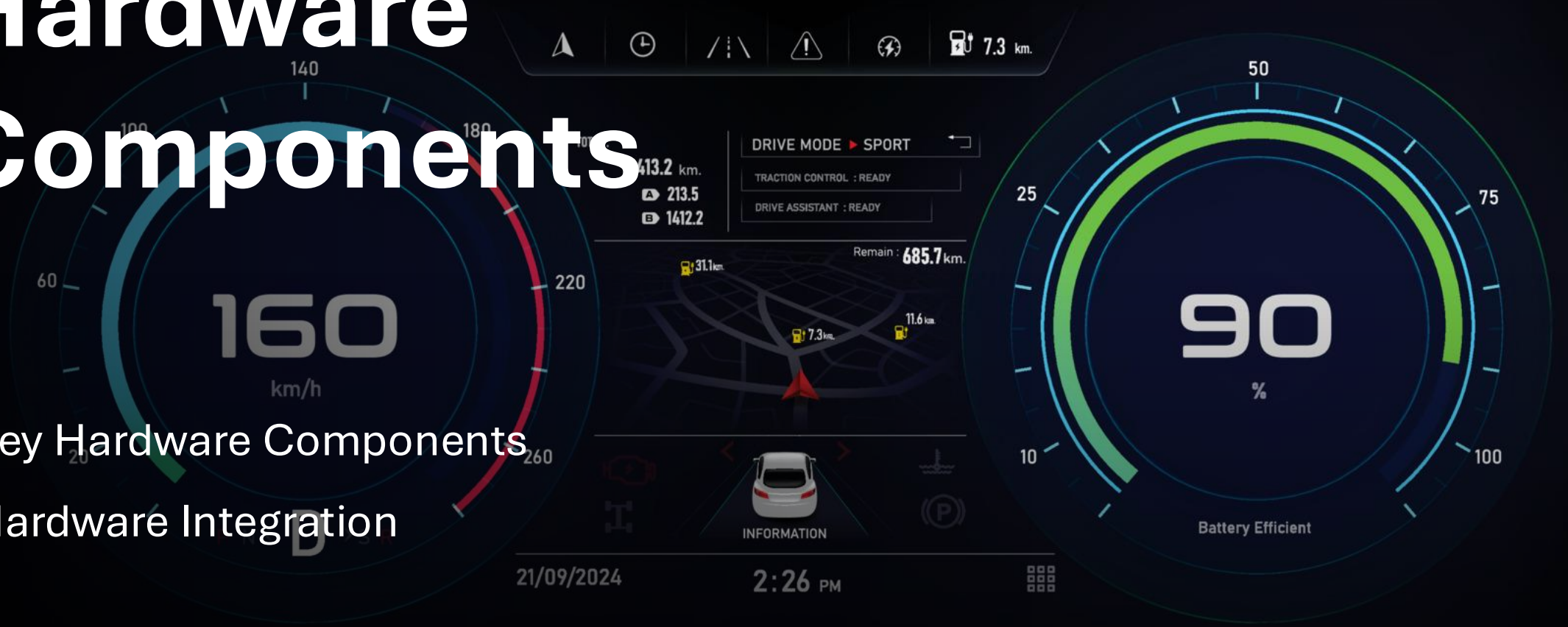
Included Sensors

1. **Speed Sensor**
2. **Temperature Sensor**
3. **Door status**

Nodal Sections

1. **Data Collection And Battery Management:** Collects data from Sensors and sends to N2 over CAN bus
2. **Telemetry System:** Receives data From data from Interpret it into desired JSON format and Push to Cloud
3. **Cloud:** Remote Storage to data Store data and that data will be represented on the dashboard

Hardware Components



- Key Hardware Components
- Hardware Integration

Key Hardware Components

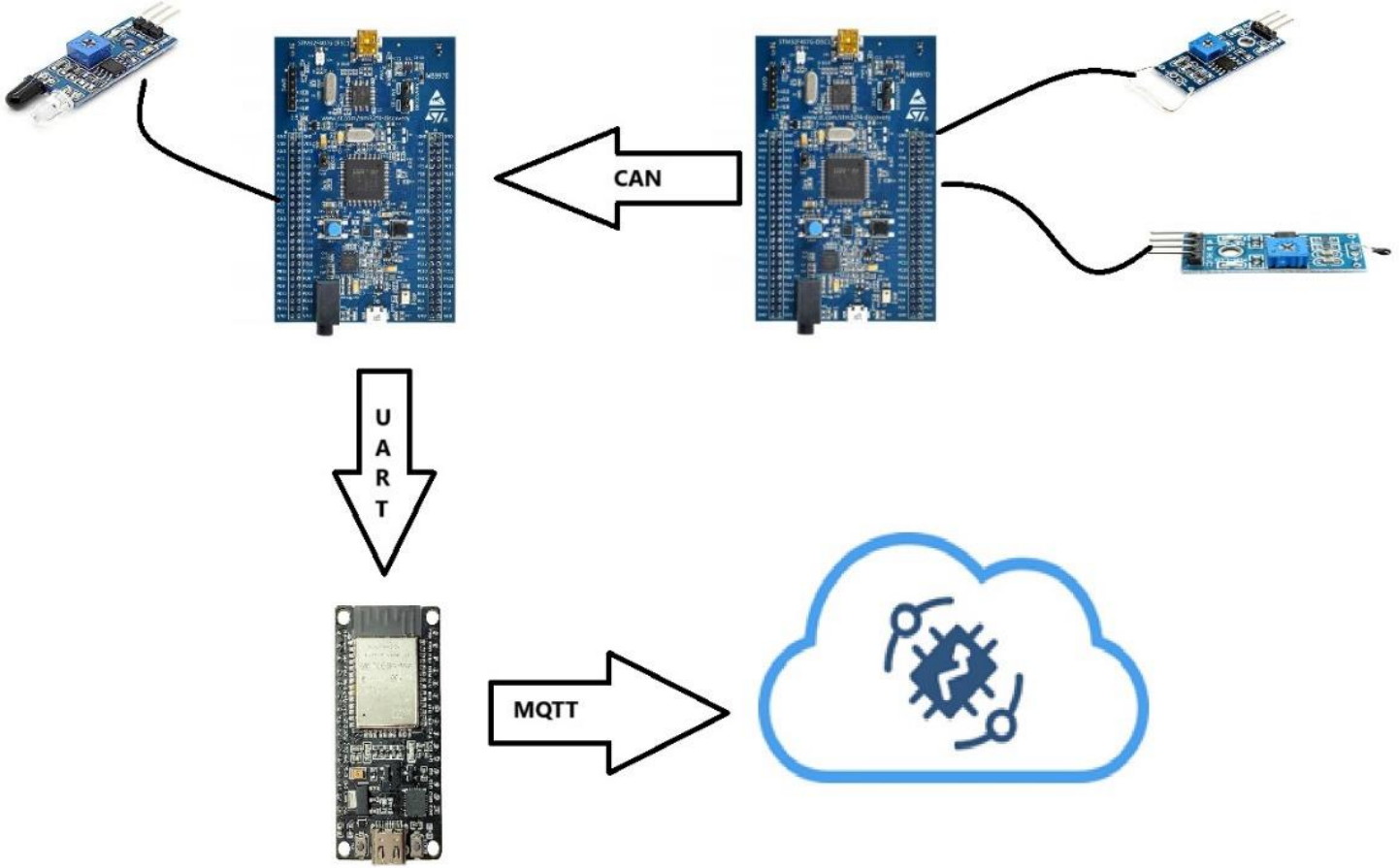
- The **CANDrive Smart Dashboard** consists of multiple hardware components working together for **real-time data acquisition and communication**.
- **Core Processing Units:**
  - **STM32 Discovery Board 2** – Collects sensor data and transmits it over **CAN Bus**.
  - **STM32 Discovery Board 1** – Processes and relays data to ESP32 via **UART**.
  - **ESP32** – Formats data and sends it to the cloud using **MQTT**.
- **Sensors Used:**
  - **TMP102 (Temperature Sensor):**

  Measures motor temperature for **overheating prevention**.
  - **IR Sensor (Speed & Distance Calculation):**

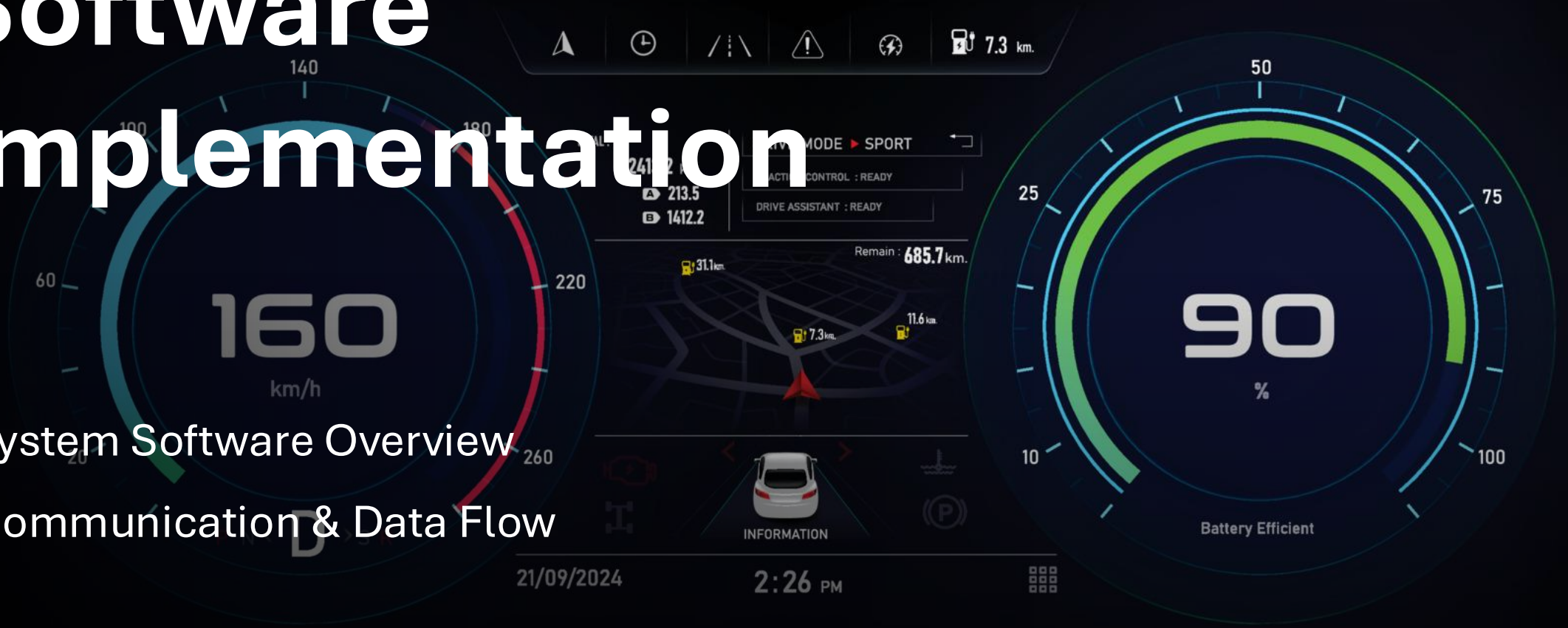
  Tracks wheel revolutions to determine **vehicle speed and distance traveled**.
  - **Reed Switch (Door Status Monitoring):**

  Detects whether the vehicle door is **open or closed**.

Hardware Integration



Software Implementation



- System Software Overview
- Communication & Data Flow

System Software Overview

- The **CANDrive Smart Dashboard** software is built on **FreeRTOS**, facilitating **multitasking** and **real-time data processing**.
- Major components include:
  - **FreeRTOS Task Management** – Schedules tasks efficiently for sensor data processing and transmission.
  - **Communication Protocols – CAN Bus (STM32 ↔ STM32), UART (STM32 ↔ ESP32), and MQTT (ESP32 ↔ Cloud)** for seamless data flow.
  - **Cloud Integration** – ESP32 transmits data to **Thingsboard** via **MQTT** for real-time visualization.
- **Task Execution in FreeRTOS:**
  - **Sensor Task** – Reads values from **TMP102, IR sensor, and Reed switch**.
  - **CAN Communication Task** – Transfers data between **STM32 boards**.
  - **UART Processing Task** – Formats and transmits data from STM32 to **ESP32**.
  - **MQTT Publishing Task** – Sends JSON-formatted data to **Thingsboard Cloud**.

CAN Node 1 (Software Architecture)

Application Layer

- Task 1 - Processing speed knob
- Task 2 - Speed & distance measurement
- Task 3 - CAN packet reception & processing
- Task 4 - Transmitting vehicle parameter packet to ESP via UART

System service calls

- GPIO (IR sensor, Motor driver control)
- UART (STM-ESP32 communication)
- CAN (STM node 1 – STM node 2 communication)
- Timers (PWM, periodic)

Embedded OS (FreeRTOS)

Hardware Abstraction Layer (HAL)

CAN Node 2 (Software Architecture)

Application Layer

- Task 1 - Read temperature
- Task 2 - CAN packet transmission to Node 1

System service calls

- GPIO (Reed switch)
- I2C (TMP102)
- CAN (STM node 1 – STM node 2 communication)

Embedded OS (FreeRTOS)

Hardware Abstraction Layer (HAL)

Vehicle Paramater Payload

MSB		LSB
Byte 0	Byte 1-2	Byte 3-4
speed	distance travelled	temperature+Door
(data from CAN node 1)		(data from CAN node 2)

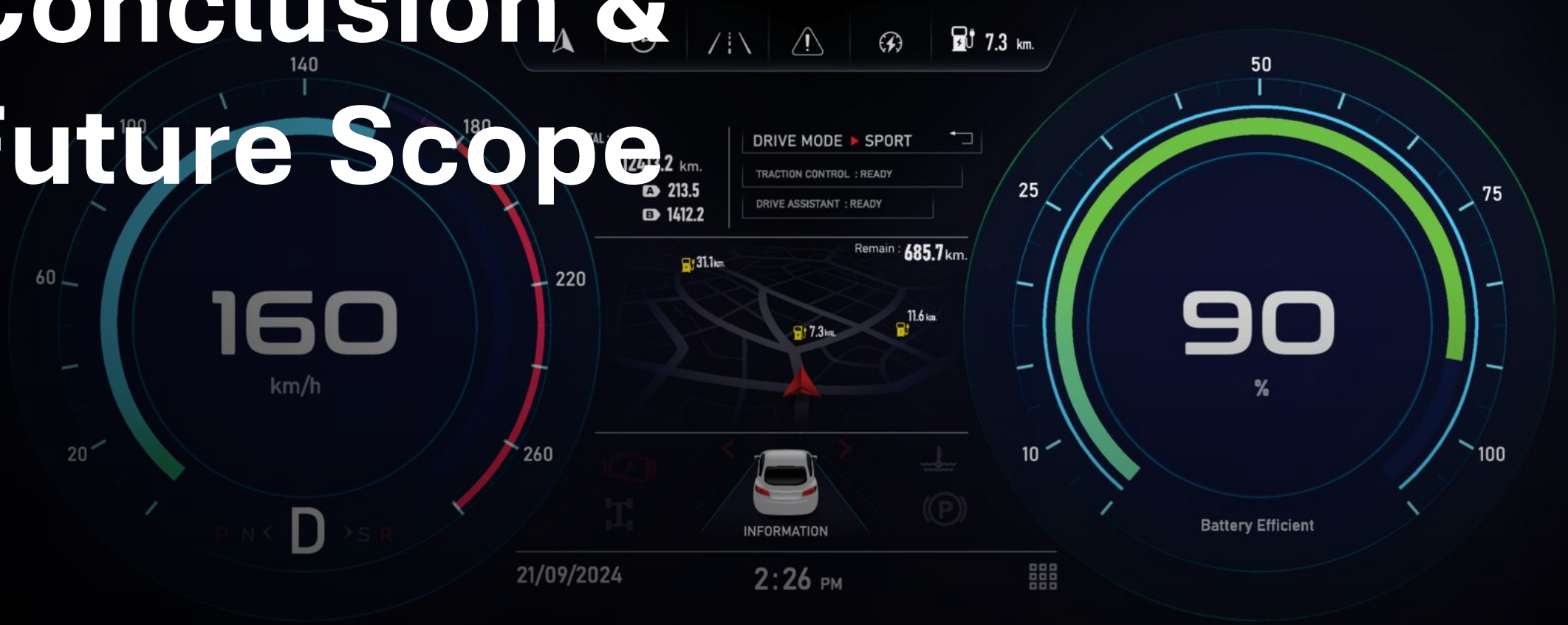
last bit of byte 4 is door status

byte3 and most significant 4 bits of byte 4 is temperature

End-to-End System Workflow

- The system follows a structured **data flow from sensors to cloud visualization**:
- **Sensor Data Acquisition:**
  - TMP102 (Temperature), IR Sensor (Speed & Distance), Reed Switch (Door Status).
- **STM32 Board 2:**
  - Reads sensor values and transmits data via **CAN Bus**.
- **STM32 Board 1:**
  - Processes received data and relays it to ESP32 via **UART**.
- **ESP32 (Wi-Fi Enabled IoT Node):**
  - Formats data into **JSON** and sends it via **MQTT** to **Thingsboard Cloud**.
- **Real-Time Dashboard:**
  - Data is visualized in a **graphical interface** for monitoring and analysis.
- **System Reliability:**
  - Uses **FreeRTOS for efficient task scheduling**, ensuring smooth real-time operation.
  - CAN Bus and UART ensure **secure and low-latency data transfer**.

Conclusion & Future Scope



Conclusion

•**CANDrive Smart Dashboard** provides a **robust and real-time vehicle monitoring solution**.

•The system **enhances predictive maintenance** by **detecting anomalies** early.

Key Achievements:

- **Accurate sensor data acquisition** and seamless cloud integration.
- **Multi-protocol communication** (CAN, UART, MQTT) ensuring efficient data flow.
- **Remote vehicle monitoring** with **Thingsboard-based IoT visualization**.

Future Scope & Enhancements

- **Advanced AI Analytics:**
  - Implement **ML models** to analyze sensor trends for **predictive failure detection**.
- **GPS Integration:**
  - Add **real-time vehicle tracking** for fleet management applications.
- **Mobile App Development:**
  - Create a **user-friendly smartphone dashboard** for remote access to vehicle status.
- **Edge Computing Capabilities:**
  - Implement **on-device processing** in ESP32 to reduce cloud dependency.