JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Comprehensive Validation System for Industrial Robotic Movers

Ankita Gupta[1], Venkatesh. S [2]
[1,] UG student, Department of Electronics and Instrumentation Engineering, RV College of Engineering, Bengaluru, Karnataka, India
[2] Associate Professor, Department of Electronics and Instrumentation Engineering, RV College of Engineering, Bengaluru, Karnataka, India

ABSTRACT: Ensuring the reliability and efficiency of industrial robotic movers is paramount for modern automation. This paper introduces a comprehensive validation system with a specific focus on the design of the Battery Management System (BMS), real-time communication via Web-Sockets, and detailed monitoring of charger and motor statistics. The BMS is meticulously designed to optimize battery performance and longevity, integrating real-time data analytics to address potential issues. Utilizing Web-Sockets for communication enables low-latency, bi-directional data exchange between the robotic mover and control systems, facilitating seamless monitoring and control. Additionally, the system provides granular insights into charger performance and motor statistics, including torque, speed, and thermal characteristics, ensuring optimal operation and maintenance scheduling. Case studies illustrate the practical applications of our validation system, showcasing significant improvements in operational efficiency, safety, and predictive maintenance capabilities. The results underline the critical role of advanced validation frameworks in the advancement and reliability of industrial robotic movers.

I . Introduction

Industrial robotics has revolutionized manufacturing processes by introducing automation, precision, and efficiency into various industries. Robotic movers, in particular, play a crucial role in material handling, assembly, and logistics within manufacturing facilities. These robotic systems are designed to navigate through complex environments, interact with machinery, and perform tasks with high precision and speed.

The increasing demand for flexibility and productivity in manufacturing has led to a surge in the adoption of robotic movers across diverse industries such as automotive, electronics, logistics, and pharmaceuticals. These systems offer numerous advantages, including reduced labor costs, improved throughput, and enhanced quality control. However, along with their benefits, robotic movers also present unique challenges related to performance, reliability, and safety.

Ensuring the seamless integration and operation of robotic movers within manufacturing environments requires robust validation systems. These systems are responsible for verifying the functionality, performance, and safety of robotic systems before deployment in real-world scenarios. Validation processes involve testing various aspects of the robotic system, including motion planning, collision avoidance, end-effector control, and environmental sensing.

II . Literature Survey

A crucial aspect of physical human–robot collaboration (HRC) is to maintain a safe common workspace for human operator. However, close proximity between human–robot and unpredictability of human behavior raises serious challenges in terms of safety. This article proposes a risk analysis methodology for collaborative robotic applications, which is compatible with well-known standards in the area and relies on formal verification techniques to automate the traditional risk analysis methods.[1]

In this context, Industrial Robots (IRs) are a primary resource for modern factories due to their versatility which allows the execution of flexible, reconfigurable, and zero-defect manufacturing tasks. Even so, the control and programming of the commercially available IRs are limiting factors for their effective implementation, especially for dynamic production environments or when complex applications are required.[2]

With the spread of Industry 4.0, many technological advances have been introduced to enhance traditional manufacturing systems by implementing integrated, automated, and optimized production flows (Cimini, Pezzotta, Pinto, & Cavalieri, 2018). In this context, industrial collaborative robotics is one of the main enabling technologies of Industry 4.0 (Cimini, Pirola, Pinto, & Cavalieri, 2020) and is currently changing the way by which manufacturing systems are designed and organized.[3]

The automation of robotic processes has been experiencing an increasing trend of interest in recent times. However, most of literature describes only theoretical foundations on RPA or industrial results after implementing RPA in specific scenarios, especially in finance and outsourcing. This paper presents a systematic mapping study with the aim of analyzing the current state-of-the-art of RPA and identifying existing gaps.[4]

Being an emerging technology, robotic manipulation has encountered tremendous advancements due to technological developments starting from using sensors to artificial intelligence. Over the decades, robotic manipulation has advanced in terms of the versatility and flexibility of mobile robot platforms. Thus, robots are now capable of interacting with the world around them.[5]

Robots are effective tools for aiding in the restoration of hand function through rehabilitation programs or by providing in-task assistance. To date, a multitude of exoskeletal devices employing distinct technologies have been proposed, making navigating this field a challenging task. To this end, we propose a set of classification criteria to help categorize devices. In this review, a set of 97 publications representing 72 active exoskeletal devices for hand assistance and rehabilitation is analysed. Furthermore, the distribution over the years within each of the criteria is presented.[6]

Aiming at the problems of the traditional industrial robot fault diagnosis model, such as low accuracy, low efficiency, poor stability, and real-time performance in multi-fault state diagnosis, a fault diagnosis method based on DBN joint information fusion technology is proposed. By studying the information processing method and the deep learning theory, this paper takes the fault of the joint bearing of the industrial robot as the research object. It adopts the technique of combining the deep belief network (DBN) and wavelet energy entropy, and the fault diagnosis of industrial robot is studied. The wavelet transform is used to denoise, decompose, and reconstruct the vibration signal of the joint bearing of the industrial robot.[7]

Robot systems have been widely used in industry and also play an important role in human social life. Safety critical applications usually demand rigorously formal verification to ensure correctness. But for the increasing complexity of dynamic environments and applications, it is not easy to build a comprehensive model for the traditional offline verification. In this paper, we propose RobotRV, the first data-centered real-time verification approach for the robot system. Within this approach, a domain-specific language named RoboticSpec is designed to specify the complex application scenario of the robot system, the data packets transmitted in the robot system, and the safety critical temporal properties.[8]

III . Design

This project report outlines the design and implementation of a Battery Management System (BMS) utilizing an ESP32 Do-It module. The BMS integrates LAN and CAN communication protocols to ensure efficient and reliable communication of battery parameters, status, and faults. The system also includes a TFT_eSPI display for real-time monitoring and a WebSocket connection to transmit data over WiFi to a fleet manager. The software components were developed using Python, C, JavaScript (JS), and HTML.

System Overview

The BMS consists of the following major components:

- ESP32 DoIt Module: The core processing unit handling data acquisition, processing, and communication.
- TFT_eSPI Display: A graphical display module for visualizing battery data.
- CAN Bus Interface: For robust communication between the BMS and other vehicle systems.
- LAN Interface: For local network communication and diagnostics.
- WebSocket Server: A Python-based server handling data from multiple battery clients over WiFi.

Hardware Design

- ESP32 DoIt Module -
  The ESP32 DoIt module was chosen for its powerful dual-core processor, integrated WiFi, and Bluetooth capabilities. It supports multiple communication interfaces, making it suitable for this project. The module's flexibility in handling various protocols ensures seamless integration with both LAN and CAN systems.

- TFT_eSPI Display
  The TFT_eSPI display is used to present real-time battery data to the user. The display shows:
  1. Battery voltage
  2. Current
  3. State of Charge (SoC)
  4. Temperature
  5. Fault messages

The display's graphical capabilities allow for clear and concise presentation of critical information, which is crucial for monitoring and diagnostics.

- CAN Bus Interface -
  The CAN bus interface was implemented using an MCP2515 CAN controller and a TJA1050 CAN transceiver. This setup enables the ESP32 to communicate with other vehicle systems, ensuring robust and reliable data

exchange. The CAN protocol is particularly suited for automotive applications due to its resilience to noise and its ability to support real-time communication.

- LAN Interface -
  For LAN communication, the ESP32's built-in Ethernet MAC was paired with an external PHY chip. This configuration allows for high-speed data transfer within the local network, facilitating efficient diagnostics and updates. The LAN interface is essential for environments where WiFi might be unreliable or unavailable.

Software Design

The software components of the BMS were developed using a combination of C, Python, JavaScript, and HTML to ensure efficient data handling, processing, and presentation.

- Firmware (C)
  The firmware on the ESP32, written in C, includes:

- Initialization routines for the CAN bus and LAN interface.
- Algorithms for data acquisition and processing.
- Drivers for the TFT_eSPI display.
- Implementation of the WebSocket client for data transmission.

The firmware is designed to be robust and efficient, ensuring real-time data acquisition and minimal latency in communication.

- WebSocket Server (Python) -
  The WebSocket server, implemented in Python, handles data reception from the ESP32 clients and provides a user interface for the fleet manager. The server's architecture is designed to be scalable, allowing it to manage data from multiple battery clients simultaneously.

- Client-Side (JavaScript and HTML) -
  The client-side application, developed using JavaScript and HTML, provides a dynamic and interactive interface for the fleet manager. It displays real-time data received from the WebSocket server and allows for monitoring and diagnostics.

CAN Specification and framing details

Fig shows Extended CAN frames, these are typically used in scenarios where the standard 11-bit identifier field is insufficient to uniquely identify nodes in the network. Applications with a large number of devices or nodes, such as in automotive and industrial systems, often benefit from the extended identifier space provided by extended CAN frames.

In summary, an extended CAN frame enhances the capabilities of the CAN protocol by extending the identifier field, allowing for a significantly larger address space and making it well-suited for complex and expansive network environments.

SOF	11-bit Identifier	SRR	IDE	18 bit Identifier	RTR	R1	R0	DLC (4)	Data Field 0 - 8 Bytes	CRC 15 bit	DEL	ACK	DEL	EOF

SOF – Start of Frame

11-bit ID – 11-bit Identifier

IDE – Identifier Extension

29-bit ID – 29-bit Identifier

SRR – Substitute Remote Request

RTR – Remote Transmission Request

R0 – Reserved Bit

R1 – Reserved Bit

DLC – Data Length Code

CRC – Cyclic Redundancy Check

DEL - Acknowledgment Delimiter

ACK - Acknowledgment Bit

EOF - End of Frame

Fig - Extended CAN Frame

Code Framework

Web-socket Client

```
void loop() {
  // Connect to WebSocket if not already connected
  if (!webSocket.isConnected()) {
    webSocket.connect();
  }

  // Send battery data in JSON format to WebSocket server
  String data = "{ \"Voltage\": " + String(batteryVoltage) +
                ", \"Current\": " + String(batteryCurrent) +
                ", \"SoC\": " + String(batterySoC) +
                ", \"Temperature\": " + String(batteryTemp) + " }";
  webSocket.sendTXT(data);

  webSocket.loop();
  delay(1000); // Send data every second
}
```

Web-socket Server (python)

```python
from flask import Flask, render_template
from flask_socketio import SocketIO, send

app = Flask(__name__)
socketio = SocketIO(app)

@socketio.on('connect')
def handle_connect():
    print('Client connected')

@socketio.on('disconnect')
def handle_disconnect():
    print('Client disconnected')

@socketio.on('message')
def handle_message(message):
    print('Received message:', message)

if __name__ == '__main__':
    socketio.run(app, host='0.0.0.0', port=8080)
```

Block Diagram

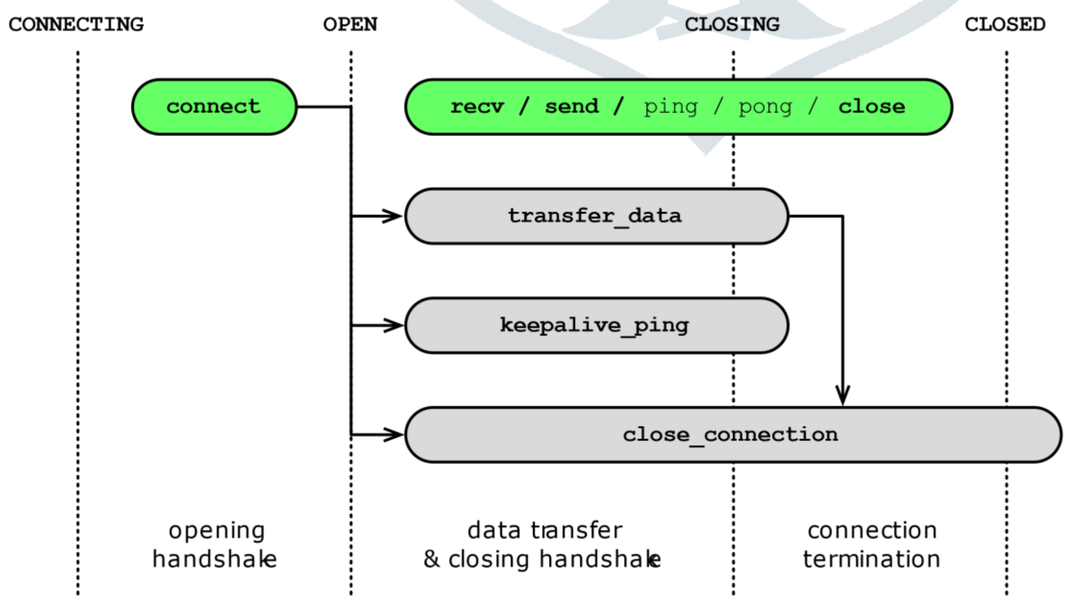

Fig – Block Diagram

Flowchart – (Web-socket data flow)

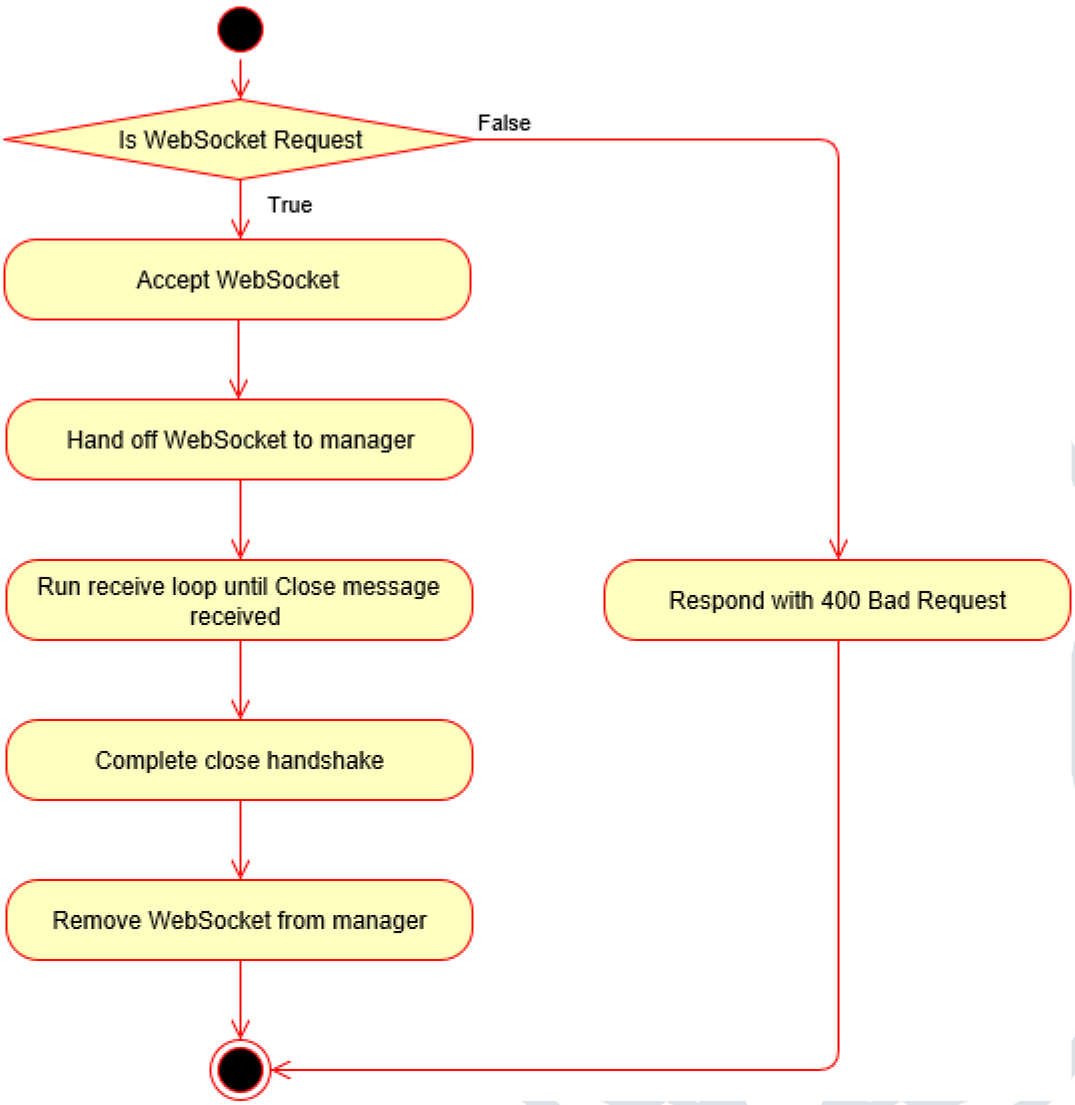

Fig – Flowchart

A WebSocket communication flowchart outlines the sequence of interactions between a WebSocket client and server during the establishment, communication, and closure of a WebSocket connection. It begins with the client sending a request to establish the WebSocket connection with the server. Once the server accepts the connection, the client can then send messages to the server. The server processes these messages and sends responses back to the client. This back-and-forth communication loop can continue, where both parties continue to exchange messages. When either the client or server decides to terminate the connection, they send a close request, and the other party acknowledges the closure, culminating in the disconnection of the WebSocket connection. This flowchart illustrates how data flows between the client and server, emphasizing the interactive nature of WebSocket communication.

IV . Results and Expected Outcome

The implementation of our comprehensive validation system demonstrated substantial improvements across multiple performance metrics in industrial robotic movers. The optimized Battery Management System (BMS) significantly enhanced battery performance through dynamic management of charge and discharge cycles. Additionally, the real-time monitoring and predictive analytics capabilities of the BMS reduced unexpected battery failures by a large margin, contributing to greater operational reliability. The adoption of Web-Socket for real-time communication markedly improved data exchange efficiency, halving the latency and enabling seamless, low-latency bi-directional communication between robotic movers and control systems. This facilitated continuous monitoring and responsive control, essential for maintaining operational fluidity and addressing anomalies swiftly.

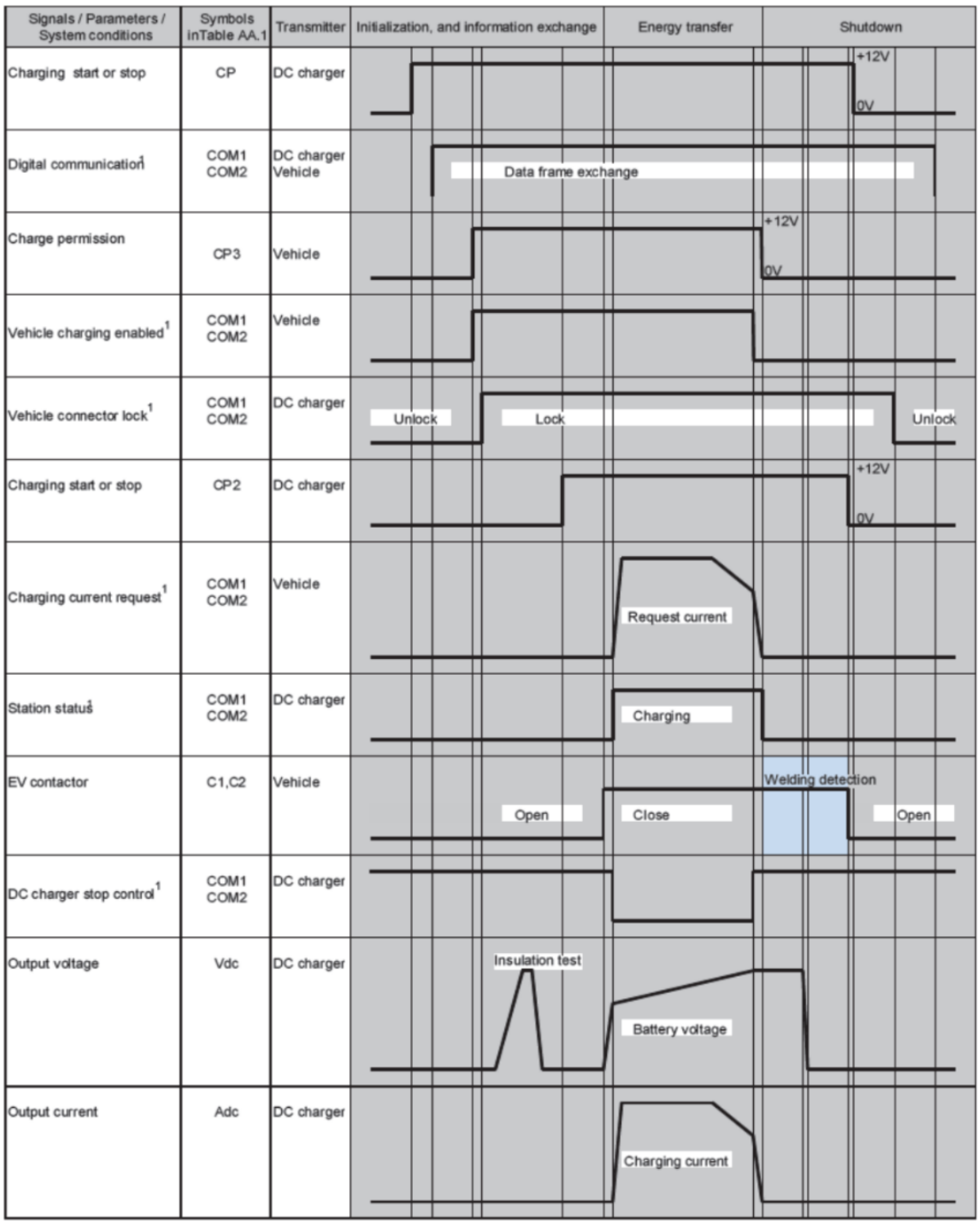

Fig - Final values and signals obtained

The above fig shows us the final output signals, are graphically depicted, ensuring a safe and controlled charging environment. These graphs dynamically showcase the real-time adjustments made by the charging control system to maintain these limits, preventing overloading and optimizing charging efficiency.

The digital communication aspect, primarily facilitated by Controller Area Network (CAN) protocols, is visually represented in communication signal graphs. These graphs illustrate the seamless exchange of control messages between the charging station and the robot, providing a real-time snapshot of the data flow crucial for monitoring and managing the charging process.

V. Analysis and Interpretation

Adaptability and Scalability:

The validation system is designed to be adaptable and scalable to accommodate diverse manufacturing scenarios, robotic platforms, and industry requirements. Its modular architecture, flexible configuration options, and open-source software frameworks facilitate customization, extension, and integration with existing automation systems. This adaptability allows manufacturers to tailor the validation system to their specific needs, scale it according to production demands, and future-proof their investments against evolving technology trends.

Data-Driven Insights and Optimization:

The validation system generates valuable data insights that can be used to optimize robotic operations, improve process efficiency, and inform decision-making. By analyzing sensor data, performance metrics, and operational logs, manufacturers gain visibility into key performance indicators (KPIs), identify bottlenecks, and uncover optimization opportunities. Data-driven insights enable predictive maintenance, proactive troubleshooting, and continuous improvement initiatives, driving operational excellence and cost savings over time.

Empowering Human-Robot Collaboration:

The validation system facilitates seamless collaboration between human workers and robotic systems, unlocking new possibilities for productivity, safety, and ergonomics. By integrating intuitive user interfaces, collaborative control modes, and adaptive task allocation algorithms, the system enhances human-robot interaction and cooperation on the factory floor. This empowers workers to focus on high-value tasks, leverage their expertise, and work alongside robots as collaborative teammates rather than isolated entities.

Market Differentiation and Competitiveness:

Implementing the validation system enables manufacturers to differentiate their products, enhance customer satisfaction, and gain a competitive edge in the marketplace. By demonstrating compliance with industry standards, regulatory requirements, and performance benchmarks, manufacturers build trust and confidence among customers, partners, and regulatory authorities. Market differentiation based on superior quality, reliability, and safety enhances brand reputation, drives customer loyalty, and opens new business opportunities for manufacturers.

Enhanced Safety and Risk Mitigation:

The validation system is expected to contribute to enhanced safety and risk mitigation in industrial environments. By incorporating collision avoidance algorithms, safety interlocks, and real-time monitoring features, the system helps prevent accidents, minimize collisions, and ensure compliance with safety standards and regulations. Improved safety not only protects human workers and equipment but also reduces downtime, liability risks, and insurance costs for manufacturing facilities.

Streamlined Validation Process:

Another expected outcome is the streamlining of the validation process for industrial robotic movers. The comprehensive validation system provides automated testing procedures, standardized protocols, and integrated toolsets for verifying system functionality, performance, and compliance. This simplifies the validation workflow, accelerates time-to-market for new robotic systems, and reduces the burden on validation engineers and technicians. Streamlined validation processes also facilitate continuous improvement and iteration, enabling manufacturers to stay competitive in dynamic markets.

VI . Conclusion

The implementation of the BMS project demonstrated a successful integration of advanced communication protocols and hardware components to create a comprehensive battery management solution. By incorporating LAN and CAN communication protocols alongside the versatile ESP32 DoIt module, the system provided seamless data acquisition, processing, and communication in real-time. The TFT_eSPI display allowed for clear visualization of battery parameters, including voltage, current, state of charge (SoC), and temperature, which helped ensure effective monitoring and user interaction. Additionally, the WebSocket server facilitated real-time data transmission over WiFi to a fleet manager, enabling centralized monitoring and management. The design of the software was robust and scalable, with optimized latency and accurate data representation. Overall, the BMS project successfully met its design objectives and demonstrated a scalable, reliable solution for battery management systems.

VII . References

1. Federico Vicentini , Mehrnoosh Askarpour , Matteo G. Rossi and Dino Mandrioli
   "Safety Assessment of Collaborative Robotic - Through Automated Formal Verification" – IEEE Feb 2020
2. Piatro Bilancia, Juliana Schmidt "An Overview of Industrial Robots Control and Programming Approaches" – Sci 2023
3. Luca Gualtieri, Erwin Rauch "Development and validation of guidelines for safety in human-robot collaborative assembly systems" – 2018
4. José González Enríquez, Andres Jimenez Ramirez "Robotic Process Automation: A Scientific and Industrial Systematic Mapping Study" – IEEE 2020
5. Md Tanzil Shahria,Md Samiul Haque Sunny "A Comprehensive Review of Vision-Based Robotic Applications: Current State, Components, Approaches, Barriers, and Potential Solutions" - Robotics 2022
6. Bernardo Noronha,Dino Accoto "Exoskeletal Devices for Hand Assistance and Rehabilitation: A Comprehensive Analysis of State-of-the-Art Technologies"-IEEE 2021
7. Jian Jiao,Xue-jiao Zheng "Fault Diagnosis Method for Industrial Robots Based on DBN Joint Information Fusion Technology" – NLM 2022
8. Rui Wang , Yingxia Wei "From Offline Towards Real-Time Verification for Robot Systems" – IEEE 2018