Workshop Report on Industry 4.0 Technologies

M.Masuod

June 9, 2024

Abstract

Executive Summary: This report summarizes the activities and outcomes of the workshop on Industry 4.0 Technologies. The workshop focused on data collection, visualization, and integration using Node-RED and MQTT protocols. The objectives were to understand and apply these technologies to create interactive dashboards for monitoring and controlling industrial processes.

Contents

1. Industry 4.0 Technologies

1.1 Introduction

The workshop on Industry 4.0 Technologies aimed to provide practical experience with modern industrial tools and protocols. Using Node-RED and MQTT. We were able to collect and visualize data from virtual sensors and control systems. This report documents the steps taken, results obtained, and insights gained during the workshop.

1.2 Aim/Objectives

The main objectives of the workshop were:

- To collect data from a virtual DHT11 sensor using the MQTT protocol.
- To visualize data using the Node-RED Dashboard.
- To integrate various Industry 4.0 technologies for effective data monitoring and control.

1.3 Methodology

The workshop involved setting up a virtual environment using Ubuntu Desktop, Node.js, Node-RED. We created Node-RED flows to read data from sensors and visualize it on a dashboard.

1.3.1 Setting Up the Environment

1. Install Ubuntu Desktop in a virtual environment (Virtual Box).
2. Install Node.js and Node-RED on the Ubuntu Desktop.
3. Start Node-RED and MongoDB by entering the following commands in the terminal:

```
node-red
sudo docker start mongo-database
```

4. Access the Node-RED programming interface via a browser at `http://127.0.0.1:1880`.
5. Install the Node-RED Dashboard module through the Manage Palette option.

1.3.2 Creating the Node-RED Flow for DHT11 Sensor

1. Delete any existing nodes in the flow and click Deploy.
2. Add an "inject" node to trigger the flow every second.
3. Add a "dht-emu" node to simulate the DHT11 sensor, set to read temperature and humidity.
4. Add a "gauge" node to display the temperature reading. Configure it to show the value in Celsius.
5. Add a "change" node to set the message payload to humidity.
6. Add another "gauge" node to display the humidity reading in percentage.
7. Deploy the flow and visualize the results on the dashboard at `http://127.0.0.1:1880/ui`.

1.4 Results

1.4.1 DHT11 Sensor Data Visualization

The dashboard successfully displayed real-time temperature and humidity data from the virtual DHT11 sensor.

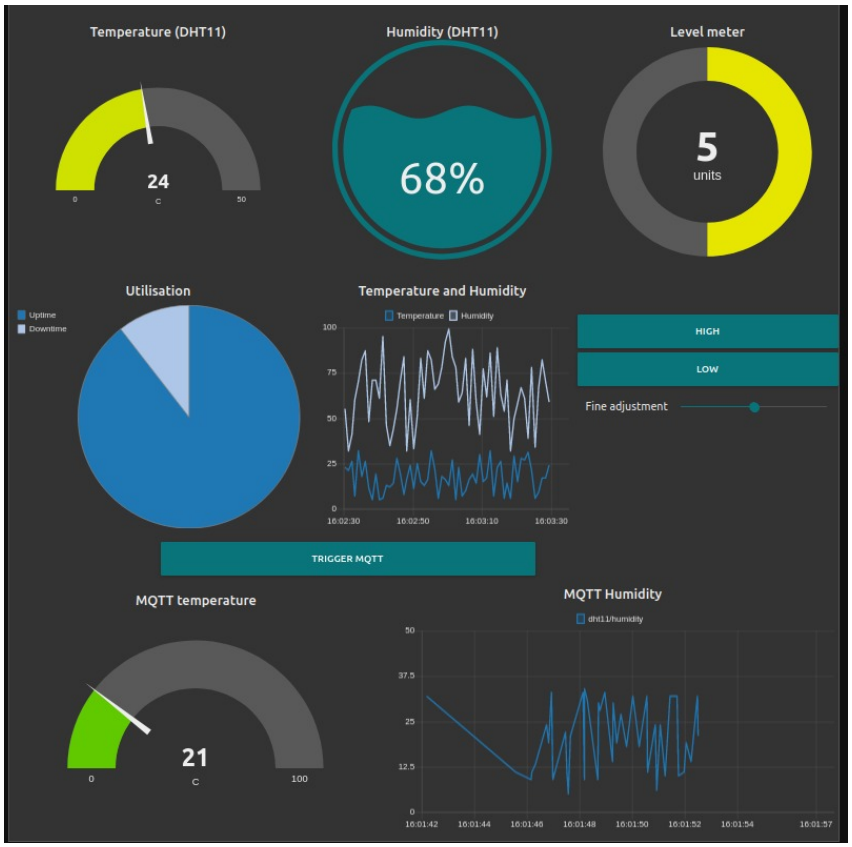

Figure 1.1: Node-RED Dashboard for DHT11 Sensor

1.4.2 Node-RED Flow for DHT11 Sensor

The flow comprised inject, dht-emu, change, and gauge nodes, enabling real-time data visualization.

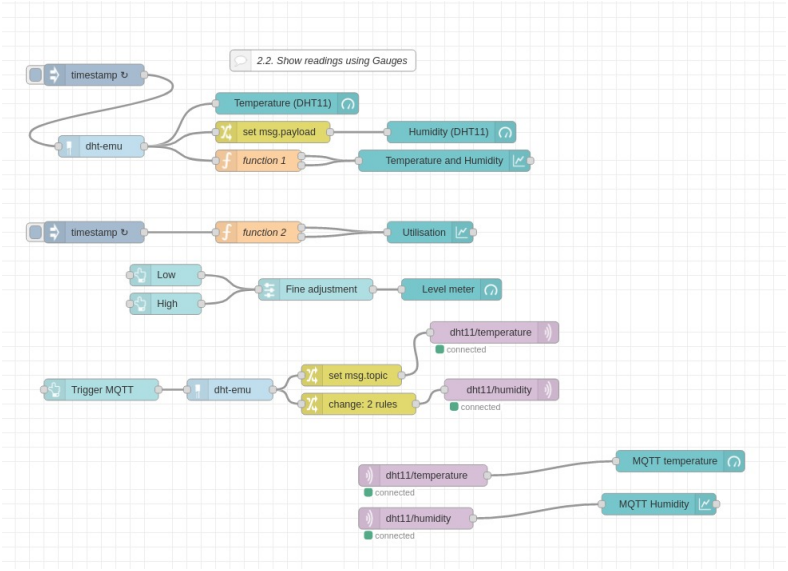

Figure 1.2: Node-RED Flow for DHT11 Sensor

1.4.3 Paper Machine Dashboard Visualization

The dashboard provided insights into the paper machine's operational parameters, including paper thickness, feed rate, and glue level status.

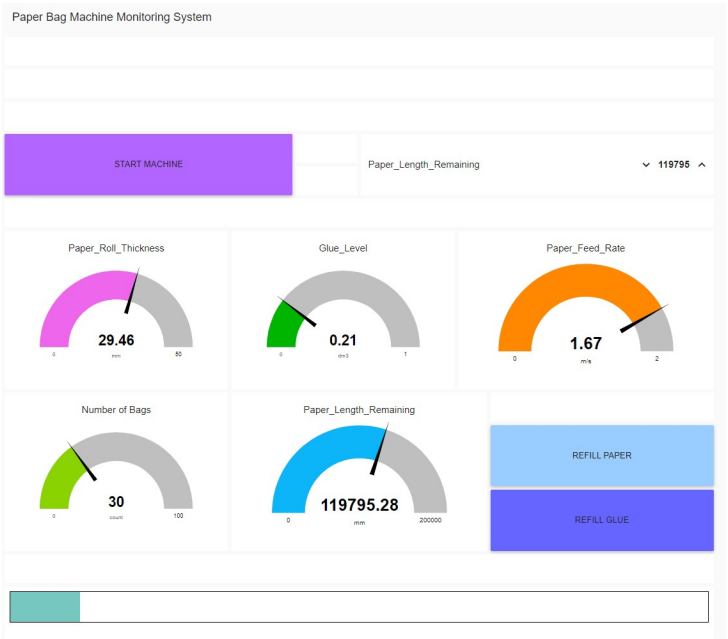

Figure 1.3: Paper Machine Monitoring Dashboard

1.4.4 Node-RED Flow for Paper Machine

The flow integrated various sensor data to monitor the paper machine's status effectively.

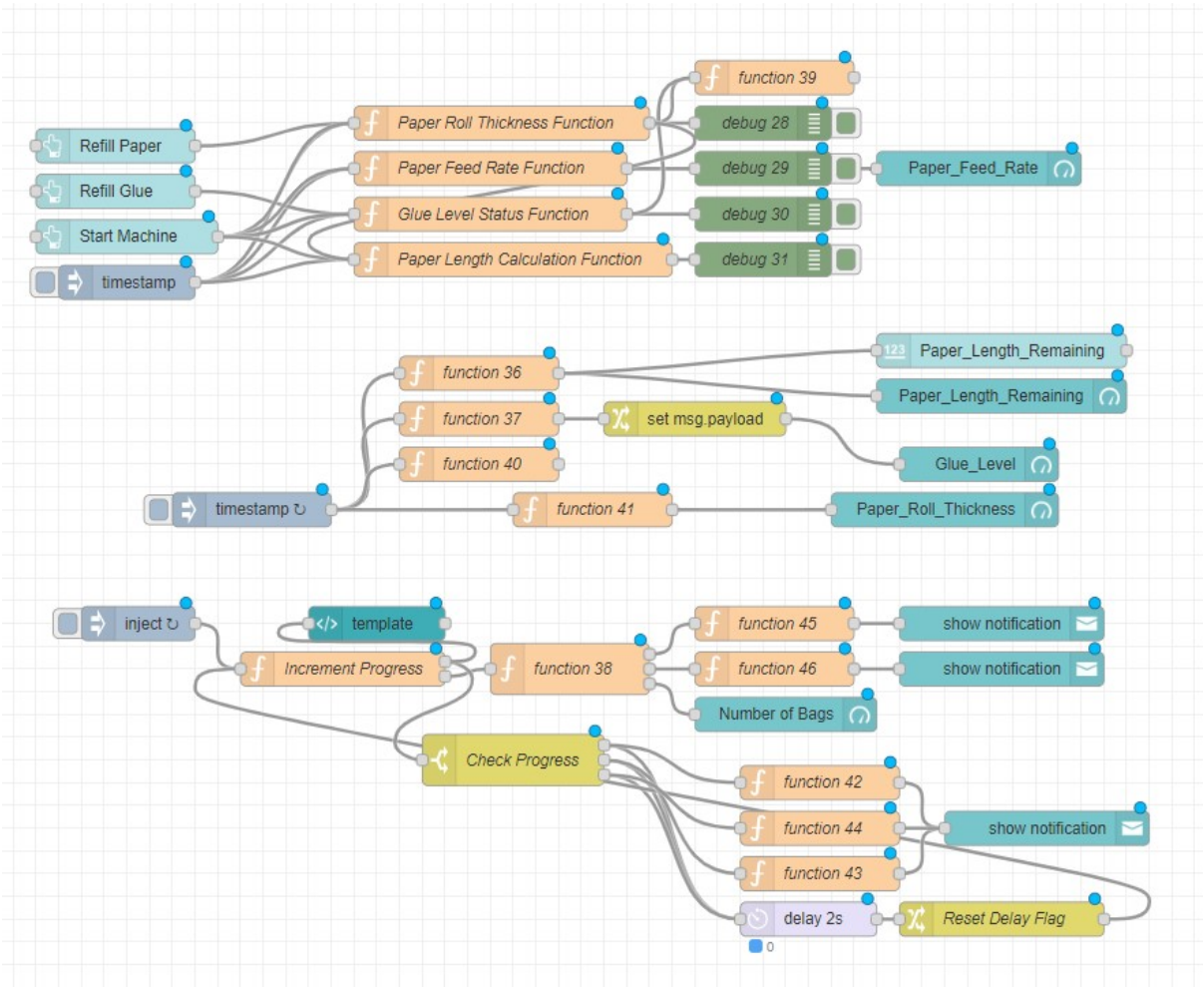

Figure 1.4: Node-RED Flow for Paper Machine Monitoring

1.5 Analysis

The workshop demonstrated the effectiveness of Node-RED for real-time data monitoring and control. The dashboards provided clear and actionable insights, enhancing our understanding of industrial process visualization.

1.6 Discussion

1.6.1 Node-RED and MQTT Integration

The integration of Node-RED with MQTT allowed seamless data transmission and visualization. By setting up flows to collect data from virtual sensors and display it on dashboards, participants gained hands-on experience with real-time monitoring systems.

1.6.2 Challenge 1: MQTT Button Configuration

The first challenge involved configuring a button to trigger three MQTT publishers. These publishers sent temperature and humidity readings from a DHT11 sensor and temperature readings from an OpenPLC runtime to three different MQTT topics. The setup required configuring an MQTT broker and three subscribers to display the data on the dashboard.

Steps:

1. Add a button node to the dashboard to trigger the flow.
2. Configure three MQTT out nodes to publish data to different topics.
3. Set up three MQTT in nodes to subscribe to the topics and process the data.
4. Use gauge and chart nodes to display the temperature, humidity, and PLC data on the dashboard.

This setup enabled real-time monitoring of multiple data points from different sources, illustrating the versatility of MQTT and Node-RED.

1.6.3 Challenge 2: Paper Machine Monitoring Dashboard

The second challenge involved designing a dashboard for a paper bag-making machine. The dashboard needed to display the paper roll thickness, feed rate, and glue level status. Function nodes generated random values for these parameters to simulate real-time monitoring.

Steps:

1. Add function nodes to generate random values for paper thickness and feed rate.
2. Use a boolean value to simulate the glue level status.
3. Configure gauge and chart nodes to display these values on the dashboard.

The dashboard design aimed to help workers make informed decisions about refilling the machine. By monitoring the paper roll thickness and feed rate, workers could estimate when the paper would run out. The glue level status indicated when it was time to refill the glue tank, balancing between avoiding downtime and minimizing waste.

Dashboard Design Explanation

To achieve this goal, the dashboard was designed to display real-time data on the paper roll thickness, feed rate, and glue level status. Here's how the design helps workers make informed decisions about refilling the machine with paper and glue:

1. **Paper Roll Thickness Monitoring:** By continuously monitoring the thickness of the paper roll using a depth sensor, workers can anticipate when the paper will run out. As the thickness decreases over time, the dashboard provides a visual representation of the remaining paper, allowing workers to estimate the remaining length of the paper roll. This estimation helps in planning timely refills to avoid downtime.

2. **Feed Rate Monitoring:** The dashboard also displays the feed rate of the paper entering the machine, measured by an optical flow sensor. By observing the feed rate trend over time, workers can predict the rate at which the paper is being consumed. Combining this information with the estimated remaining length of the paper roll, workers can better anticipate when the paper will run out and plan refills accordingly.

3. **Glue Level Status Indication:** The glue level status is represented as a boolean value on the dashboard, indicating whether the glue tank is full or needs refilling. A proximity sensor installed at a specific height in the glue tank detects the presence of glue. When the glue level falls below the sensor, the boolean value changes, signaling that it's time to refill the glue tank. This ensures that workers refill the glue tank promptly to avoid interruptions in the bag-making process. I have also added a level measurement for glue to give more precise results for the usage.

Overall, the dashboard provides a comprehensive overview of the machine's status, allowing workers to make informed decisions about refilling the machine with paper and glue. By continuously monitoring the paper roll thickness, feed rate, and glue level status, workers can optimize the refill timing, minimizing both downtime and waste in the production process.

1.7 Recommendations & Action Plan

Based on the workshop experience, the following recommendations are made:

- Enhance the dashboard with additional sensors for more comprehensive monitoring.
- Implement data logging features to analyze historical data trends.
- Explore advanced data visualization techniques for better insights.
- Integrate alarm and notification systems for critical thresholds.

1.8 Conclusion

The workshop provided valuable insights into the use of Industry 4.0 technologies for data collection and visualization. The successful implementation of Node-RED dashboards demonstrated the potential of these tools in modern industrial applications. Continued exploration and enhancement of these systems will lead to more efficient and intelligent industrial operations.

2. Appendices

2.1 Node-RED Flows

Below are detailed Node-RED flow configurations and code snippets used during the workshop:

```
[Node-RED Flow for DHT11 Sensor]
[Inject] --> [dht-emu] --> [gauge (Temperature)] --> [change] --> [gauge (Humidity)]
```

2.2 Sensor and System Specifications

Here are the specifications and data sheets for the virtual DHT11 sensor.

2.2.1 Appendix A: Virtual DHT11 Sensor

- **Model:** Virtual DHT11 Sensor Emulator
- **Type:** Digital Temperature and Humidity Sensor
- **Interface:** MQTT
- **Temperature Range:** 0°C to 50°C
- **Humidity Range:** 20
- **Accuracy:** ±1°C (Temperature), ±5
- **Communication Protocol:** MQTT (Message Queuing Telemetry Transport)

2.3 References

- https://flows.nodered.org/node/node-red-dashboard
- https://github.com/ctch3ng/VBox_Ubuntu_NodeRED_MongoDB_OpenPLC