International Journal
Information Technology Applications (ITA)
Volume 13, Number 2, December 2024

VISUALIZATION OF PRODUCTION DATA USING NODE-RED

Ján Cigánek, Dávid Červenka

Abstract:

The paper effectively addresses the critical integration of SCADA systems in modern industrial automation, specifically focusing on the visualization and control of production processes. By examining a real-world application — using a school model production line controlled by a PLC device and visualized through the Node-RED platform — the paper offers valuable insights into the practical implementation of such systems. The proposed work not only highlights the importance of data visualization in streamlining operations, but also underscores the growing need for efficient control and monitoring of production processes. Through both theoretical analysis and hands-on implementation, the thesis demonstrates how combining SCADA technologies with user-friendly environments can significantly improve operational oversight, ensuring smoother automation workflows and long-term data management.

Keywords:

SCADA, PLC, visualization, production process, control algorithm, Node-Red, simulation.

Introduction

The development, implementation, and design of SCADA (Supervisory Control And Data Acquisition) systems have gained increasing importance in recent years, driven by the rapid advancements in information and communication technologies.

The primary role of SCADA systems is to control technological processes, representing industrial automation. These systems not only manage and monitor processes but also handle and archive vast amounts of data over extended periods. Process control, based on the evaluation and analysis of results, leads to improvements in technological processes within a company, such as enhanced product quality, increased reliability, higher efficiency, and reduced energy consumption.

In this work, we focus specifically on the design and implementation of a SCADA system for a real laboratory model of a production line, controlled by a PLC (Programmable Logic Controller) device and visualized within the Node-RED environment. This study aims to explore the integration of SCADA systems with educational models, offering insights into how industrial automation principles can be applied in smaller-scale settings.

Through this practical application, we demonstrate the potential for SCADA systems to optimize production processes, improve data management, and foster more efficient operations in both industrial and educational contexts.

1 Production

Production processes refer to all activities involved in a series of steps or actions dedicated to transforming raw materials or components into finished products. These processes often utilize machinery, tools, equipment, and other resources to perform tasks such as production, assembly, packaging, and transportation. At the core, manufacturing processes involve the interaction of manufacturing forces, aimed at creating a specific product. Every manufacturing process requires human involvement and labor resources.

Production can be characterized as a method in which economic resources—such as labor, capital goods, and land—are utilized to provide goods and services to consumers. The primary objective of manufacturing is to ensure the efficient and productive production of goods for sale, aiming for quick delivery to customers without compromising product quality.

There are numerous types of production processes that companies can use, depending on their production goals, manufacturing volumes, and technological tools or software systems in place. These processes can vary significantly depending on the type of product being produced, the industry, and the specific manufacturing techniques applied.

This paper aims to explore the diversity of production processes and their adaptation to different sectors, as well as the technological advancements that shape them. By examining the interconnected nature of production methods, it provides insights into optimizing manufacturing efficiency while maintaining high-quality standards [1].

Efficient production processes are essential for companies to maintain their competitiveness and meet customer demand while upholding high quality standards and controlling costs. In recent years, automation of production processes has become increasingly popular due to its ability to enhance efficiency and consistency, while simultaneously reducing labor costs.

a) Automation of production processes

Automation allows companies to streamline production workflows, minimize human error, and improve product quality by ensuring precise and repeatable operations. The integration of automated systems and robotics into production lines has revolutionized industries, enabling faster production times, lower operational costs, and higher output volumes. This trend is not only beneficial for large-scale manufacturers but is also becoming more accessible to small and medium-sized enterprises (SMEs), who can leverage automation technologies to remain competitive in an evolving marketplace.

Automation refers to the implementation of technology that reduces the need for manual labor and enables systems to make automated decisions based on computer algorithms. It involves the application of machines to tasks that were previously carried out by humans or, increasingly, to tasks that would otherwise be impossible to perform [2].

The key advantage for businesses is the reduction of operational costs, as automation streamlines workflows and reduces the reliance on human labor for repetitive or complex tasks. Automation also enhances workplace safety by performing hazardous or physically demanding tasks that may pose risks to employees.

The impact of automation has been transformative in industries where it has been adopted. It has revolutionized manufacturing, logistics, and various other sectors, and it is rare to find an aspect of modern life that has not been affected by its advancements. The widespread use of automation has not only increased productivity but also contributed to more consistent product quality and quicker response times to customer demands [3].

Automation can also be defined as a technology related to the execution of a process using programmed instructions combined with automated feedback control to ensure that they are executed correctly. The resulting system is able to operate without human intervention.

The very development of this technology has become increasingly dependent on the use of computers and computer-related technologies. As a result, automated systems are increasingly sophisticated and complex. Today's advanced systems represent a level of capability of control and performance that in many ways exceed the ability of humans to perform the same activities. Thus, if a controlled system behaves in a way that meets our needs, it means that control is not necessary in this case. Control is associated with certain behaviours of the system, that is directed towards some goal. For example, the field of PLC automation is the field of system control with goal-directed behaviour (from the steam engine to modern digital control systems) [2, 4].

b) Programmable logic controller (PLC)

PLC is a programmable computing device used to control electromechanical processes, especially in the industrial sector. It is a user-programmable digital automatic machine which has some specific features compared with conventional computers. The role of PLC is to control the functions of the system using internal programmed logic. Companies around the world use PLCs to automate their most important processes [6].

PLCs are widely used in various industries due to their speed, ease of use and programmability. PLCs can be programmed in a variety of ways, from ladder logic based on electromechanical relays to special modified programming languages such as BASIC and C. Today, most PLC devices use one of the following five programming languages: Ladder Diagram (LD), Structured Text, Function Block Diagram, Instruction List, Flowchart [5].

Today, PLC systems are used in a variety of industrial applications, including manufacturing, energy, transportation and many other areas. Their functions are constantly expanding and improving to handle increasingly complex processes. Examples of PLC device applications in mechatronics include controlling motor speeds, monitoring the position and attitude of equipment, controlling pneumatic and hydraulic systems, controlling measurement and sensor equipment, and more. In general, it can be said that PLC systems are very important for the efficient control and automation of industrial processes in the field of mechatronics and for improving the overall performance and efficiency of machinery and equipment [6].

Over time, PLCs have evolved and adapted to the specific needs of the industrial environment. Due to their flexibility, PLCs can be used in a wide range of industries. PLC devices can be individually wired to control and regulate the performance of manufacturing machines that can meet the high demands of modern industry. PLC can usually be installed directly on the production machine, which implicitly saves the necessary space capacity. In addition to the ability to remotely control the PLC, one of its greatest advantages is the ability to communicate with other peripherals. However, depending on the area of application, it is essential that the PLC operator must have expert knowledge [7].

The components of a PLC device are divided into three main categories: inputs, outputs and CPU (Central Processing Unit). PLCs collect data from production by monitoring the inputs to which machines and devices are connected. The input data is then processed by the CPU, which applies logic to the data based on the state of the input. The CPU then executes the logic of the user-created program and sends data or commands to the machines and devices to which it is connected.

Inputs can include on/off states for elements such as mechanical switches, buttons and encoders. High/low states for elements such as temperature gauges, pressure sensors, liquid level detectors, or open/closed states for elements such as pumps. The second type of input, human-assisted inputs, includes pushbuttons, switches, sensors from devices such as keyboards, touch screens, remote controls or card readers. Outputs can be characterised as physical actions or visual results based on PLC logic in response to the corresponding inputs. To give an idea of the physical outputs and also for an idea of what they contain, it is for example starting a motor, turning on lights, a valve tripping or a pump shutting down. Visual outputs are sent to devices, such as printers, projectors and monitors [5].

The functionality of PLC devices is based on cycles. First, the PLC recognizes the state of all connected input devices. The PLC device runs a user-created program that uses the state of the inputs to determine the state to which the outputs should be switched. The PLC then adjusts the output signals for each corresponding device. Once all of these steps, the PLC performs a cleanup (maintenance step) that includes an internal safety diagnostic to ensure normal operating status. The PLC device restarts the cycle after completion of each process. After the restart, the inputs are checked again. Thanks to the wide range of available communication devices, it is possible to connect the control device to almost any PLC [5, 6].

c) Visualization systems

System visualization is the process of mapping the flow and function of a system, server or data flow. Visualizations help to diagnose problems faster, communicate between departments and efficiently build or update the system. The concept of a visualization system is the consistency of all parts of the system, where information retrieval affects, for example, perception (brightness, colour contrast, character size), coding (icons, codes) and organization (structure) [8].

Visualization is a field that deals with the processing and presentation of data. In the broader sense, it compares all data from different sources. It represents a technique for creating images, diagrams or animations to communicate a message. At present, visualisation has increasingly expanding applications in science, engineering, medicine and so on. A typical visualisation application is the field of computer graphics. The invention of computer graphics may be the most important development in visualisation since the invention of central perspective in the Renaissance. The development of animation also has helped to advance visualization [9].

The field of computer graphics is a large and diverse area, which is the intersection of computer science and design. It deals with the whole process of creating computer generated images, from the creation of 3D digital models through the texturing process, rendering and lighting these models, to the digital display of these representations on screen. This process presents simple techniques for rendering objects on converting mathematical representations of three-dimensional objects into a two-dimensional image on screen. Computer graphics can thus be understood broadly and generally as any graphical representation created by a computer.

Information visualization is the practice of presenting data in a meaningful and visual way that users can easily interpret and understand. This includes data visualizations and dashboards. Information visualization is an effective way to share knowledge in a format that is digestible for non-experts. It typically shows relevant context in the data, allowing decision makers to more easily draw conclusions and take informed action. Information visualisations are often created with the audience in mind and are designed to present important information that needs to be understood. By having an idea of how the visualizations will be used, the visual designer can determine the best format for arranging the information and visuals to be used [6].

Contemporary information visualization is becoming increasingly popular as advanced tools and technologies enable interactive features that allow users to manipulate data in real time. This allows non-technical users to explore topics from different perspectives and easily gain insight into the data.

There are many software tools available for process visualization, which help in mapping, analyzing, and optimizing various manufacturing and business processes. These tools play a crucial role in improving efficiency, communication, and decision-making within organizations. There are some of the most well-known and widely used software tools for process visualization. Each of these tools is valuable for process visualization and automation, and their use depends on the specific needs of the industry or application. WinCC and AVEVA InTouch are more tailored for traditional SCADA applications in industrial environments [11,12], while Promotic offers greater flexibility and customization [10].

Node-Red (Fig.1) is a graphical development tool (application) that is gradually becoming the industry standard for the development of industrial IoT applications that collect, process, and share data over the Internet. It is used to connect events and data from, for example, physical devices (sensor), social networks and services such as email, SMS or notifications. It also serves as a programming tool to connect hardware devices (e.g. interfacing with PLCs and Arduino), APIs and online services in new and interesting ways [13].

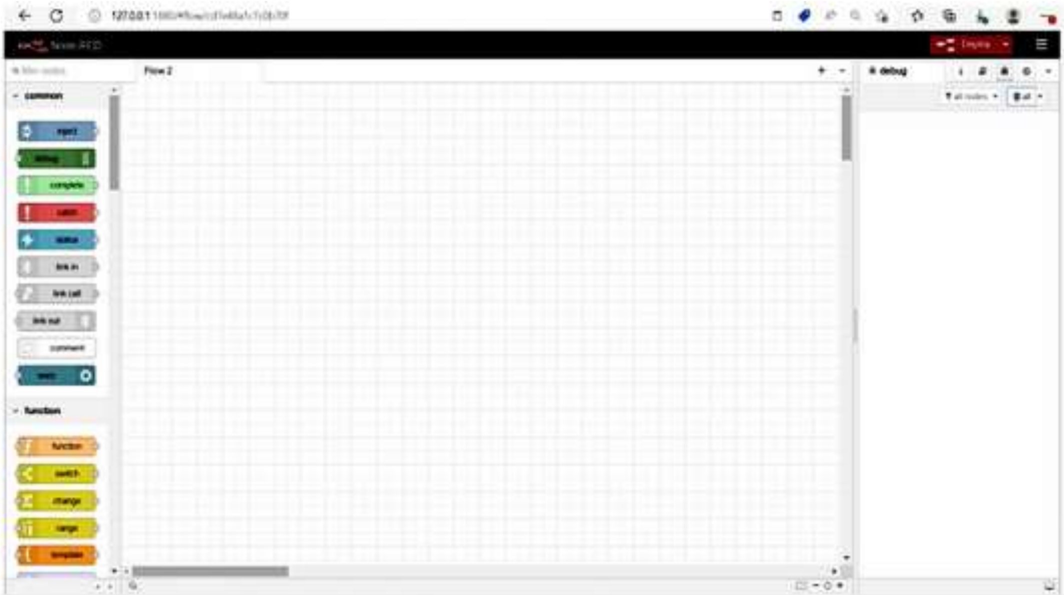

Fig.1. Node-Red environment.

Node-Red is an open-source tool originally developed by IBM's Emerging Technology Services team, now part of the OpenJS Foundation. It is supported by Windows, Linux and Mac OS platforms. It is also possible to use the nodes application, but this requires knowledge of the JavaScript programming language [14].

Flow-based programming was invented by J. Paul Morrison in the 1970s. It is a way to describe the behavior of an application as a network of nodes. Each node has a clearly defined purpose - it receives, transmits or manipulates data. The network is responsible for the flow of data between nodes. Due to its simple logic, this model allows a wider range of users to understand the whole process.

Thus, it is possible to get an idea of its operation from the model without having to understand the individual lines of code of each node. A flow is created by combining any number of nodes. They are connected by wires formed by dragging the grey output of one node to the grey input of another node [14].

Node-Red consists of a web-based runtime environment based on Node.js. In the browser, it is possible to create an application by dragging nodes from the palette onto the canvas and start connecting them. Node-Red provides the aforementioned web-based flow editor, which makes it easy to link flows using a wide variety of nodes in the palette. Flows can then be deployed to runtime with a single click, at which point the application is deployed to the runtime environment in which it subsequently runs [14,15].

There is a wide range of libraries available, and a short selection of the default libraries available in the main panel of the Node-Red tool can be seen in (Fig.1).

The Node-red-dashboard is a collection of nodes that can be used to create a web dashboard that interacts directly with a given flow [16].

Node-red-contrib-s7 is a collection of Node-Red nodes serving as an interface to the Siemens S7-PLC. Each single link to a PLC device represents an S7 endpoint configuration node. It is possible to configure the PLC address, also the available variables and their addresses, and the cycle time for reading the variables.

2 Case Study

The primary piece of equipment used at this researsch was Fishertechnik model of a production line with two machining stations from. This is an ideal training, simulation and demonstration model for industrial automation training and demonstration. The production line model is mounted on a solid wooden board. Voltage is optionally available in standard 9V as well as the world industry standard 24V. 24V voltage is required for unconditional operation with PLC equipment.

The production line model (Fig.2) contains 2 machines (milling and drilling station) and 4 conveyor belts arranged in a U-shape. The conveyor line uses 8 type XS DC motors, 2 sliders, 4 pushbuttons serving as limit switches, 5 phototransistors and 5 LED light barriers. The model has a circuit board with relays for reversing the direction of rotation of the motors. All inputs and outputs can be connected to a jack plug (26-pin, 2.54 mm pitch) or to series terminals with push-in terminals [17].

Fig.2. Laboratory model of production line [17].

SIMATIC S7-1200 (Fig.3) is a modular, compact, versatile and comprehensive control system for automation applications. A flexible design, a communication interface that meets the highest communication standards in the industry, and a wide range of built-in functions make this PLC an entry-level model that is part of automation tasks.

The digital inputs on the PLC device operate at 24V, with a voltage level in the 0-5V range considered a logic zero and a level in the 15-24V range considered a logic one. The analog inputs have a range of 0 to 10V. The digital outputs have a voltage level of 24V [19].

Memory The PLC 1200 has up to 50 KB of integrated main memory with a seamless interface between program and data memory on the SIMATIC S7-1200. The flexible boundary between data and program ensures efficient use of all disk space.

In addition to the main memory, the SIMATIC S7-1200 includes 2 MB of integrated memory and 2 KB of backup memory in case of a power failure. For transferring the necessary project to multiple base units and updating the firmware, the SIMATIC memory card is available in two sizes: 2 MB or 24 MB.

Fig.3. SIMATIC S7-1200 [18].

The control algorithm was created in TIA Portal environment using the Ladder diagram language. In the tables, IO tags were created with the relevant type and address according to the individual sensors (inputs) and actuators (outputs) connected to the PLC.

The implementation of the algorithm consisted of controlling the conveyors and sliders on the production line so that the product was transferred through the entire line. When the product is detected at the first light sensor, conveyor 1 is started and stopped at the second sensor. The slider 1 will move backwards until the limit switch is pressed, causing conveyor 1 to restart for 2 seconds. The motion of the slider and conveyor 2 is then activated, which moves the product to light sensor 3. This sensor activates machine tool 1 for 5 seconds.

In the next phase, the product is moved to conveyor 3, where it is detected by sensor 4, activating machine tool 2 for 7 seconds. After the product has been processed, it is moved from conveyor 3 by slider 2 to conveyor 4, where it is dropped off the line. The program also includes the treatment of product loss using timers and alarms.

The program is divided into function blocks in TIA Portal, such as Alarms, Counters and Conveyors. In addition, the motor hours of the actuators and the number of times the motors are switched on are measured. An algorithm for counting and resetting the number of finished products at the exit of the line has been developed based on the detection by the last light sensor #5, as well as an algorithm for counting the number of products on the line. The laboratory model can be controlled in both automatic and manual mode.

The Node-Red environment was used to implement and design the visualization environment used to control the production line model. The Node-Red program is written in JavaScript running on the NodeJS platform. The communication between the PLC devices and the personal laptop was realized through an Ethernet cable. A library called node-red-contrib-s7 was needed to communicate between the Node-Red and TIA Portal (PLC device).

For the visualization, it was necessary to download the libraries that offered us the relevant parts to control the correct operation of the production line, for example the most used library for communication and for visualization node-red-dashboard. It was necessary to implement individual functions, e.g. buttons, warps and switches, to control the functionality of the production line. We used the JavaScript programming language to implement the backlighting of the buttons in manual mode and for "moto-hours" (listing seconds, minutes, hours), number of products on the line and number of finished products.

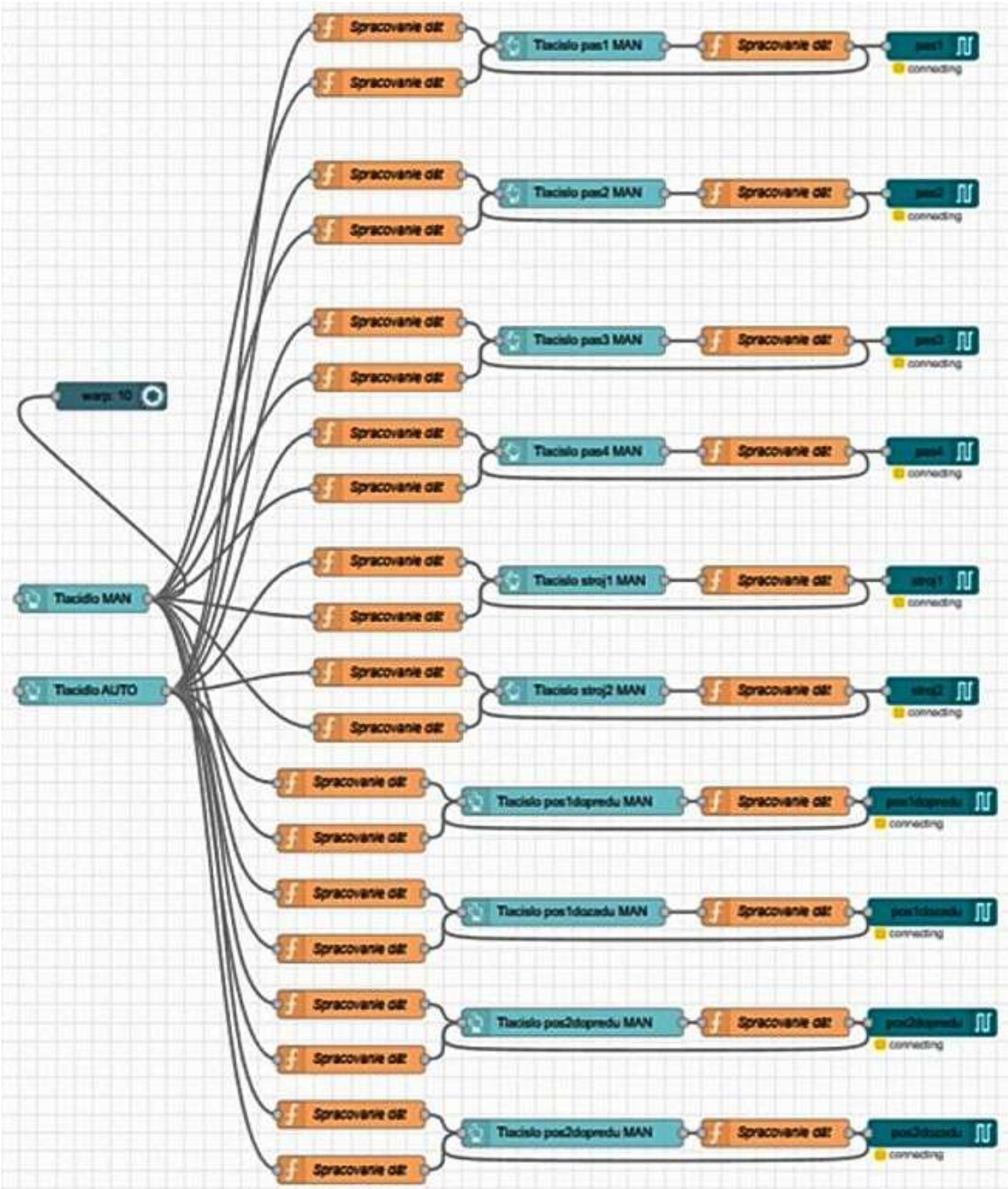

Fig.4. Automatic and manual control in Node-Red environment.

The production line starts when the start button is pressed (Fig.5), at that moment the automatic mode is running. To switch from the automatic mode to manual mode, there is the button called MAN, where the backlight of the buttons is set to green and red. If all the buttons used for manual mode are illuminated in red, this indicates that the entire production line operation is stationary and waiting for the operating personnel who decide to operate the line component manually with a button. The illumination of the button to green will not occur until it is pressed by the operating personnel. The AUTO button is used to switch back to automatic mode from manual mode.

Fig.5. HMI in Node-Red environment during manual mode (MAN, AUTO buttons on left side). Four columns: Control, Sliders, Machines+Conveyors, Moto-hours.

It was necessary to implement a graphical representation of "moto-hours" in the visualization, in seconds, minutes and hours (Fig.6). The visualization will show how long the product was on the conveyor belt and how long it was under the machine tools for that section. A count is also taken of how many products are on the line and how many products have been produced.

We have set a reset option for each associated function to list the "moto-hours" of the individual components. If the operating personnel chooses to reset the running time counting of a given product via a component, they have the option to use the reset button and the given values of seconds, minutes and hours will be counted from the beginning (from 0). The option to reset the count of products present on the line and the number of products produced is available by pressing the reset button.

Conclusion

The aim of the paper was to learn about the design and implementation of a SCADA system for a real school model of a production line, which is controlled by a PLC device and visualized in the Node-Red environment. We provided new insights and details about the PLC device as well as the Node-Red visualization environment.

After analyzing and familiarizing ourselves with the current state of the art, we focused on their implementation. The first step was the implementation of the production line control algorithm on the PLC device, which we implemented in the TIA Portal environment. Subsequently, we described the correct functionality of the production line algorithm. After this step, we focused on communication between the PLC device and the visualization system, which was implemented and designed in the Node-Red environment.

We made a visualization model through which it was possible to control the operation of the production line, where we described the implementation of the control of the visualization model. From the achieved results, was achieved the functionality of the SCADA system algorithm for the real school model of the production line, as well as the possibility of controlling the visualization model of the production line through the Node-Red application.

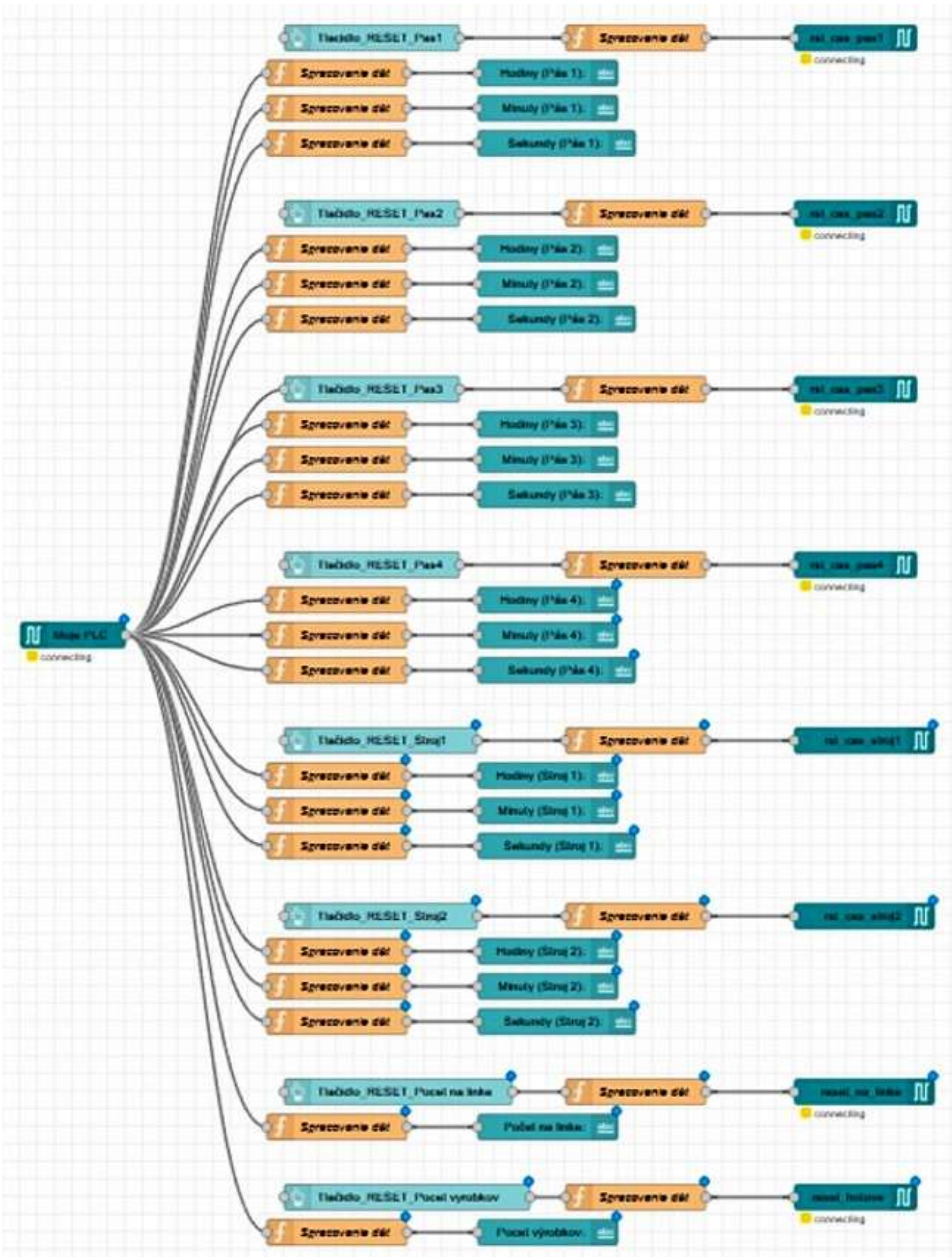

Fig.6. Moto-hours in Node-Red environment.

Acknowledgement

This paper was supported by the Slovak Grant Agency VEGA 1/0637/23, KEGA 021STU-4/2024 and 010STU-4/2023, and by the Scientific Grant APVV-21-0125.

References

[1] Indeed Editorial Team: Production Process: Definition and Types for Businesses To Use [Online]. September 30, 2024.
Available: https://www.indeed.com/career-advice/career-development/production-process

[2] T. Pettinger: Automation meaning and examples [Online].
Available: https://www.economicshelp.org/blog/glossary/automation-meaning-and-examples/

[3] M. P. Groover: automation [Online]. December 15, 2024.
Available: https://www.britannica.com/technology/automation

[4] P. Kuna: Fundamentals of automation (In Slovak). Nitra: UKF, 2015. 57 p.
ISBN 978-80-558-0841-3

[5] Inductive automation: PLC: Programmable Logic Controller. PLC Basics, Operation, Programming and the Future of PLCs [Online]. February 24, 2020.
Available: https://inductiveautomation.com/resources/article/what-is-a-PLC

[6] J. Cigánek, F. Žemla: PLC systems in mechatronics (In Slovak).
Bratislava: Chemical library STU, 2023. 88 p. ISBN 978-80-8208-097-4

[7] Paessler: IT Explained: PLC [Online]. Available: https://www.paessler.com/it-explained/plc

[8] Lucidchart: Why visualizing systems is essential in the Next Normal [Online]. Available: https://www.lucidchart.com/blog/why-visualizing-systems-is-essential-in-the-next-normal

[9] Spotfire: What is information visualization? [Online].
Available: https://www.spotfire.com/glossary/what-is-information-visualization

[10] Microsys: PROMOTIC SCADA visualization software [Online].
Available: https://www.promotic.eu/en/index.htm

[11] Siemens: SIMATIC WinCC - Process visualization with Plant Intelligence [Online], 2010.
Available: https://www.pcne.eu/uploads/tx_etim/siemens_wincc.pdf

[12] AVEVA: AVEVA InTouch HMI [Online].
Available: https://www.aveva.com/en/products/intouch-hmi/

[13] K. Bourke: What is Node-RED? How Can I Use it to Create IoT Applications? December 6, 2022.
Available: https://www.realpars.com/blog/node-red

[14] OpenJS Foundation & Contributors: Node-RED [Online]. Available: https://nodered.org

[15] P. Král: Node-RED application (In Czech) [Online]. Available: https://docplayer.cz/113829298-Aplikace-node-red-zakladni-seznameni-co-je-to-node-red.html

[16] OpenJS Foundation & Contributors: Node-RED dashboard [Online].
Available: https://flows.nodered.org/node/node-red-dashboard

[17] Fischertechnik: Indexed line with 2 Machining Stations 24V DC [Online].
Available: https://www.fischertechnikwebshop.com/en-gb/fischertechnik-indexed-line-with-2-machining-stations-24v-dc-96790-en-gb

[18] Elettronew: Siemens Simatic S7-1200 1212C 8DI/6DO/2AI DC/DC/DC CPU 6ES72121AE400XB0 [Online].
Available: https://www.elettronew.com/en/automation-siemens-simatic/siemens-simatic-s7-1200-1212c-8di-6do-2ai-dc-dc-dc-cpu-6es72121ae400xb0-28941.html

[19] Siemens: Simatic S7-1200 [Online]. Available: https://www.siemens.com/global/en/products/automation/systems/industrial/plc/s7-1200.html

Authors

Ing. Ján Cigánek, PhD.
Faculty of Electrical Engineering and Information Technology,
Slovak University of Technology in Bratislava, Slovakia
jan.ciganek@stuba.sk
He was born in 1981 in Malacky, Slovakia. He received the diploma and PhD. degree in Automatic Control from the Faculty of Electrical Engineering and Information Technology, Slovak University of Technology (FEI STU) in Bratislava, in 2005 and 2010, respectively. He is now Assistant Professor at Institute of Automotive Mechatronics FEI STU in Bratislava. His research interests include optimization, robust control design, computational tools, SCADA systems, big data, and hybrid systems.

Bc. Dávid Červenka
Faculty of Electrical Engineering and Information Technology,
Slovak University of Technology in Bratislava, Slovakia
xcervenka@stuba.sk
He was born in 2000 in Malacky, Slovakia. He received the Bc. degree in Automotive Mechatronics from the Faculty of Electrical Engineering and Information Technology, Slovak University of Technology (FEI STU) in Bratislava, in 2023. He is now Student of master degree at Institute of Automotive Mechatronics FEI STU in Bratislava. His research interests include computational tools, SCADA systems, database systems and cloud services.