TAS Journal, vol. 7, n. 3, p. 85-94
ISSN 2595-1521
SEPTEMBER 2023
pelegrino.jv@gmail.com

Simulation and control of manufacturing processes applying Industry 4.0 in the CIM-B Kit

Pelegrino JV*, Souza CMP†, Silva Junior WC*, Dantas RD*

** Federal Institute of Education, Science, and Technology of São Paulo, Guarulhos, Brazil. † Mackenzie Presbiterian University, São Paulo, Brazil.*

Abstract

Industry 4.0 is a technological revolution transforming how industries produce and manage manufacturing processes. In this context, this work proposes the study and updating of a didactic kit (CIM-B) that simulates and controls a manufacturing process (composed of PLC, pneumatic actuators, various discrete sensors, and a load cell), applying some of the concepts of Industry 4.0. To control the manufacturing process in CIM-B, a production control and monitoring dashboard will be developed using Node-RED and a read/write system via RFID tags inserted into each piece to be manufactured. These pieces will receive the registration of the processes to be executed according to the product selection defined in the dashboard and a database for real-time activity records. All integration and data exchange will be done via Node-RED, using the communication protocols Modbus and MQTT and the MySQL database. Four types of pieces (variations of the manufacturing process and type of material of the work) were used for simulation. As a result, it was possible to verify that the system integration can be done using Node-RED (which acted as the integrator of the various automation equipment used in the kit) and low-cost devices such as the ESP32 microcontroller and the MRC522 RFID reader/writer.

Keywords: Industry 4.0; CIM-B kit; Node-RED; Database; RFID.

Introduction. In the mid-15th century, during the medieval era, all production work was done manually by artisans who passed down their knowledge to their apprentices. However, after many years, the need to increase production efficiency and meet the growing demands of the population led to the first industrial revolution, characterized by the use of the steam engine and advancements in weaving (Sacomano, 2018).

Subsequently, with the advancement of steel, the second industrial revolution took place, driven by Taylorism and Fordism, which promoted mass production and specialization of labor. Japan was a significant highlight during this period as it was the first to understand the importance of quality over quantity, contrasting with the approach championed by Henry Ford in the United States (Noldin, 2002).

doi.org/10.32640/tasj.2023.3.85

This concept of lean production became more appealing to customers. It resulted in the third industrial revolution, which promoted effective communication between companies and customers through communication networks and computers, commonly called global globalization. This revolution led to the origin of networks and the interconnection of various technologies worldwide (Acatech, 2013).

Finally, the connectivity of everything, from small devices and microcontrollers to industries and management panels, emerged with the significant advancement of technology in the fourth industrial revolution, also known as Industry 4.0 (Drath, 2014). This revolution is characterized by automated systems that exchange data with each other and with humans, optimizing all stages of product production material acquisition, whether physical or not, to managerial analysis and post-sales. In this revolution, customer feedback regarding product quality can be collected and fed back into the process, creating a cycle (Oliff. 2017).

In today's world, we are experiencing the transition to the fourth industrial revolution, which also promotes the integration of advanced technologies such as Artificial Intelligence, the Internet of Things, robotics, and 3D printing, among others. In this context, this work aims to study and update the didactic kit (CIM-B) to simulate and control a manufacturing process in real time and individualize by product. Through programming in Ladder via Twido Suite, realign the Twido Suite connection with the PLC, install RFID readers and writers, create a database in MySQL Workbench, and integrate all components via Node-Red.

Materials and Methods. A Programmable Logic Controller (PLC) from the Schneider brand was used for the equipment activations. It received information from connected sensors and input devices and performed programmed actions on the connected outputs. In this project, an ESP32-WROOM microcontroller with internal programming in C++ enabled Wi-Fi, an essential connection for interconnecting other devices. Another item used was RFID tags, which allow sending of specific information about a product through electromagnetic readers, facilitating data management and transmission/acquisition within the process, including without requiring it to stop (6). In addition to the physical equipment and resources, the software is needed to handle all the gear simultaneously, as mentioned in one of the foundations of Industry 4.0. One of the leading software used in this project is Node-RED, a platform that enables a direct connection between hardware devices, APIs, and online services simply and intuitively (7). With all the data that can be originated in various forms, a database was created, another non-physical platform that allows storing a large amount of information, assisting processes such as cloud computing and controlling Big Data variables (8). In this work, the MySQL database from the company was chosen for use.

For all these equipment and software to communicate, the use of communication protocols is necessary. Three protocols were used (ModBus, Transmission Control Protocol/Internet Protocol (TCP/IP), and Message Queuing Telemetry Transport (MQTT)).

The work was divided into the development of the dashboard, analysis of dashboard data, writing of information to the RFID tag, pneumatic activations, reading of information from the RFID tag, manufacturing and weighing execution, and storage of information in the database. It is important to note that it was necessary to attach the RFID tags to the top of the parts before the start of the process to capture them at precise stages. Due to the lack of metal materials, aluminum foil was introduced around some features to simulate the sensor detection of a metallic object, increasing the number of examples in terms of material and mass.

Five blocks were created with outputs that automatically retrieve the logic signals from the PLC and two more blocks with logic-level inputs that send the call to the PLC for communication via Modbus. For communication via MQTT, an output block was created that automatically retrieved the data coming from the broker that reads the RFID, which is combined with the data coming from the broker and passes them on, and another block that receives information from the Node-RED and sends the information to the broker. To control inputs and outputs, function blocks were created in Javascript. Interface blocks were included for monitoring PLC signals, where information was taken from the broker and sent to the database. The necessary connections structure for the development of the project was defined, the connection between the PLC and the Router took place via network cable, the router connection with the TwidoSuite was carried out, allowing the plant to receive all the programming in Ladder, the relationship of the router with Node-Red was through a link to an open Node server with the computer, through a command prompt, in the case of this project the code for the server number was 127.0.0.1:1880. The connection between the RFID and the Node-RED takes place through the router; the ESP connects to the broker via Wifi, thus communicating with the MQTT, which is carried out satisfactorily with the Node-RED, which also has a connection to the MQTT network, thereby programming the ESP into the RFID antenna. The relationship of Node-RED with the database was made virtually and was done directly by blocks and codes internal to Node-Red,

The connections of each ESP32 with the RFID tag reader antenna are as follows. The MOSI pin represents the communication for sending data from the primary to the secondary device (Master Output Slave Input). The SDA (Serial Data) pin is responsible for data communication. The SCK (Serial Clock) pin defines the system clock. The MISO pin represents the communication from the secondary to the primary device (Master Input Slave Output). The GND (Ground) pin represents the ground connection. The 3V3 pin supplies the system with 3.3 Volts. The RST (Reset) pin is used for resetting the communication. The links are shown in Figure 1.

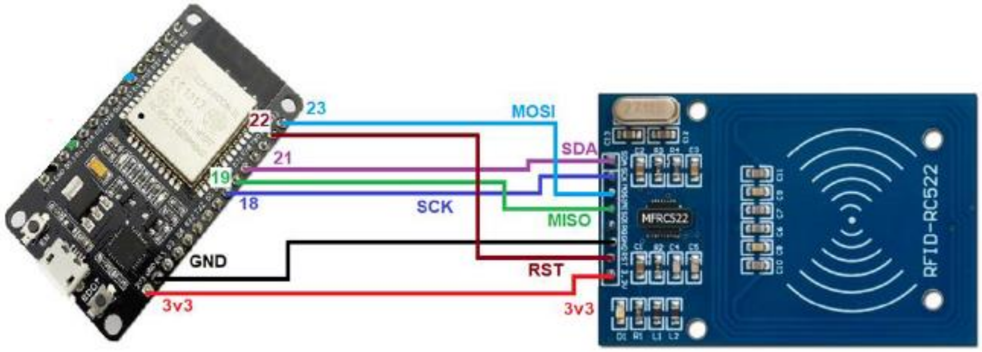


Figure 1 – Circuit ESP/RFID.

To obtain all the information from the executions during the procedure, a database was developed using MySQL Workbench software hosted on a local machine. This database provides information about the part's material, processing time, and completed steps. Each step represents a part of the kit, where Step 1 involves identifying the material and recording the information on the RFID tag. After that, in Step 2, the previously registered information is read, and in Step 3, the drilling, weighing, and allocation of the part to the correct container are performed. All this data is transferred from the kit to Node-RED, making it available in the database in a tabular format.

After all the connections were protected and tried separately, some tests of the complete process were carried out using different configurations and materials, with other parts, metallic and non-metallic, all with different masses, customized in Figure 2.

Figure 2 – Cylindrical parts.

The tests were divided into a few steps. First, the user provided data for machining. The treadmill was activated, and the part was positioned under the RFID recorder, fixed to the support after receiving a pulse from the inductive or capacitive sensors to record information provided by the user on the RFID tag attached to the part. In the third step, the information recording on the material was completed, shown in Figure 3, the conveyor belt was activated again, and the piece was transported to the turntable.

Figure 3 – Transporting the part to the turntable.

The part was positioned on the turntable; then, the turntable was activated, performing a rotation and placing the region under the RFID reader, allowing decision-making about the drilling time. In the fifth step, the user read the options, and the turntable performed one more rotation, positioning the part under the drill, as shown in Figure 4. Long, short, or no drilling can be performed at this moment.

Figure 4 – Drilling step.

In the sixth step, the rotary table rotates again, taking the part to the gripper responsible for taking the amount to the load cell; with the positioning of the interest on the load cell, the weighing is carried out, and the reservoir for which the machined part will receive is defined. In the next step, the gripper is activated after weighing, transporting the piece from the load cell to the container specified by its mass. At the end of the process in the plant, we can see all the information reflected in the database, such as part material, completed steps, individual identifier code for each part (UID), mass, and start time.

Results and Discussions. To initiate the manufacturing process of the kit and analyze how production would be carried out, the development of a dashboard in Node-RED was necessary.

The blocks for constructing the dashboard, except for Form, receive the logical levels from the sensors through Modbus communication, defined by the red blocks. Each block has a window similar to the example of the first sensor, where "FC" indicates whether it is a retrieval or sending of the sensor's logical level, "address" is the address in the PLC's memory for the sensor, "quantity" is the number of sensors being read, and "poll rate" is how often Node-RED reads this value. Once each required field is filled in and the data is sent, an object is generated with the names of each selected item, followed by either "true" or "false," indicating whether it has been established.

After the form submission, the data must be processed to be sent via MQTT to the ESP32, which enables the reading of RFID tags. The data is passed to a function block with JavaScript code, which converts all true or false values to 0 and 1. In the same block, these numbers are combined to form a single word or string. This transformation reduces the output from a large object to a series of size six, allowing for the subsequent writing of the information to the RFID tag.

After the data conversion, a final modification is required from string to buffer. This is necessary because when the information is sent to the broker for writing to the tag, it must be stored in buffer format. This conversion is performed by a block that handles the transformation from string to buffer format.

With the conversion of the information into buffer format, it is possible to send the data from Node-RED to the microcontroller using the MQTT protocol. In Node-RED, the blocks responsible for communication are used. After sending the information, the union that carries out the data forwarding is called "DATA." In this block, you must input the name of the broker used in the project, the port for information communication, and the topic to which the information will be sent to the broker.

After completing the Node-RED connection step with the broker, the next step is to connect the microcontroller to it. The entire implementation of the ESP32 code was done in the Arduino interface with lines of code in C++. In the code, it is necessary to import the libraries for Wi-Fi connection, communication between the microcontroller and the RFID antenna for reading, and relationship with the broker. Once the Wi-Fi security information and network name are implemented in the code, it is possible to establish the microcontroller connection via the MQTT protocol. To capture the information sent by Node-RED, two constants need to be indicated in the code: the communication ID (which can be any value) and the topic (which should be the same as the topic of the data sent by Node-RED in this case, "DATA"). With both the ESP32 and Node-RED stages ready, communication can be established. The conveyor belt is turned on to start the process, and the form information is manually sent to the broker. When the piece reaches the stage of the first RFID square in Figure 5, the antenna detects the presence of the tag and writes the information received from Node-RED on it, thus completing the step of writing the material type and the manufacturing process to be executed.

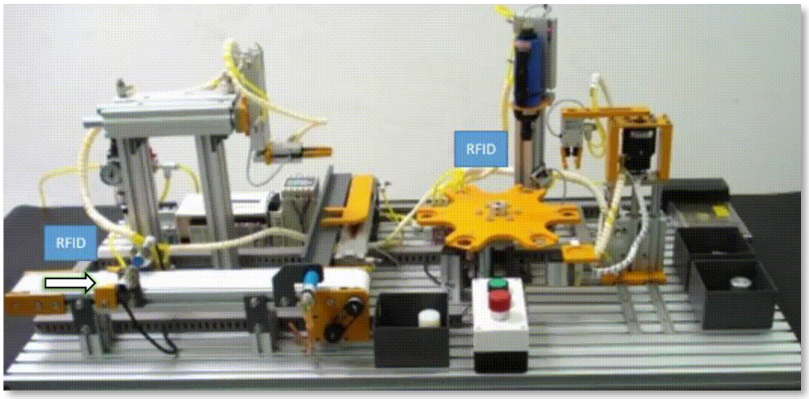


Figure 5 – KIT CIM-B

The conveyor belt starts automatically after the system begins with the belt being activated for the first time and reaches the writing stage with successful communication between the microcontroller and Node-RED. The piece goes through phases of pneumatic activations, defined by the ladder programming inserted in the PLC. Figure 5 shows the writing stage in the first blue square labeled RFID on the left. The piece goes through all the transportation processes, reaches the rotating table, stops at the second RFID communication point in the blue square closer to the center of the image, undergoes machining action by the drill, is grabbed by the gripper, its mass is measured, and it is allocated to one of the containers at the end of the kit on the far right, according to its group.

With the piece in the correct position for reading on the rotating table, the MFRC522 antenna exchanges electromagnetic signals with the RFID tag of the work and reads the information previously written during the initial reading stage. The read information is sent via MQTT using the same connection method as during the writing process, except for the difference in the ID and topic. This time, the case is named "RFID" instead of "DATA," as the rationale explains, to avoid issues with crossed or overlapping information.

The received information is the same, converted from the form into a six-digit binary sequence string. The first two digits determine the piece's material (metal or plastic), and the last two specify the manufacturing process type (long or short machining). For this step, only the last two digits are essential. This output is passed to two JavaScript programming blocks that analyze the value of the penultimate digit (if it's 1, it indicates long machining) and the last number (if it's 1, it means short machining). Therefore, these two blocks allow the logical level to pass only if they receive a value of 1. This value of 1 is then passed to two blocks that communicate via Modbus to activate the drill. They send a high logical level to one or the other memory of the PLC, which is connected to different timers to control the manufacturing period. After the manufacturing stage, the piece is rotated again by the rotating table until it is positioned for the unloading arm to pick it up and place it on the scale, as shown in Figure 6. Its mass is measured, and based on the measurement, the rotating arm identifies which batch to dump it into, as shown in Figure 7.

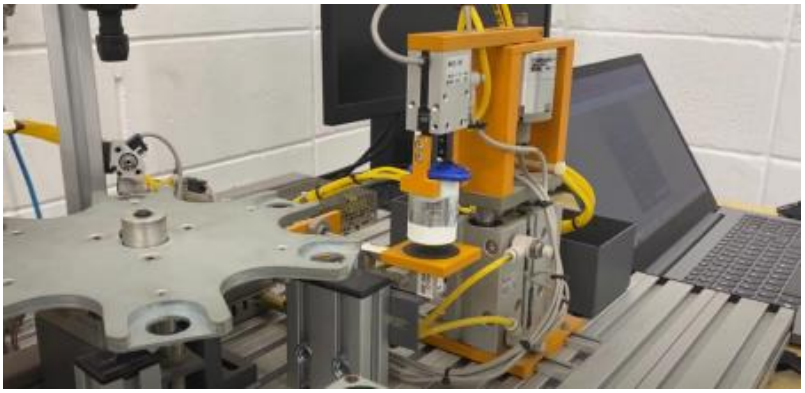
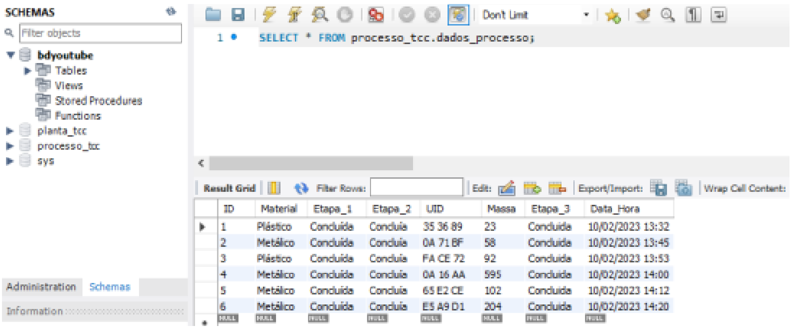


Figure 6 – Part dumped into the load cell.

Figure 7 – Pouring the part into the container

Alongside the entire process, some information is stored to build the data in the MySQL database. The initial values are retrieved from the capacitive, inductive, and optical sensors and the rotating table. These values help determine whether the piece has gone through the main stages of the kit.

Throughout the process, some information is stored to build the data in the MySQL database. Initial values are retrieved from capacitive, inductive, and optical sensors and the turntable. These values help to determine whether the part went through the steps requested by the user, the tests performed were practical, all possible conditions were verified, such as variation in the mass of the part and variation in drilling, and the equipment proved to be functional, as well as programming and storage data as shown in Figure 8.



```
SELECT * FROM processo_tcc.dados_processo;
```

ID	Material	Etapa_1	Etapa_2	UID	Massa	Etapa_3	Data_Hora
1	Plástico	Concluida	Concluia	35 36 89	23	Concluida	10/02/2023 13:32
2	Metálico	Concluida	Concluia	0A 71 BF	58	Concluida	10/02/2023 13:45
3	Plástico	Concluida	Concluia	FA CE 72	92	Concluida	10/02/2023 13:53
4	Metálico	Concluida	Concluia	0A 16 AA	595	Concluida	10/02/2023 14:00
5	Metálico	Concluida	Concluia	65 E2 CE	102	Concluida	10/02/2023 14:12
6	Metálico	Concluida	Concluia	E5 A9 D1	204	Concluida	10/02/2023 14:20
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 8 – Database.

Conclusion. At the end of the project development, achieving satisfactory results based on expectations was possible. The communication and utilization of Node-RED to make the kit closer to Industry 4.0 concepts were executed and concluded in a manner that ensured success in the CIM-B kit manufacturing process. Throughout this work, the importance and advantages of applying Industry 4.0 concepts in the CIM-B kit were identified, and these benefits can be replicated in more robust systems within the industrial sector. This application can significantly reduce the cost of production, as seen in the average price of RFID readers and ESP32, which, when combined, are relatively inexpensive compared to equipment used in large-scale industrial production lines. The use of advanced technologies, which were extensively studied, allowed for increased flexibility in the kit's manufacturing process. Previously, programming was limited to a single language without additional protocols that could expand the functionality of the proposed equipment. With the new implementations, different working methods for the CIM-B kit can now be developed, thanks to the concepts of Industry 4.0. The success of this implementation in the kit opens up opportunities for other students and teachers to give more prominence to the subject of Industry 4.0 within the course. It also stimulates further discussion around the topic, fostering improvements to the kit through additional implementations based on this project. For example, the integration of Artificial Intelligence could be explored by capturing images during the kit's stages, utilizing the recorded data in the database for statistical analysis, and incorporating it into classroom activities. These advancements will enhance the kit's capabilities and provide a solid foundation for future developments and innovations.

Acknowledgment. I want to thank my advisor, Rogério Daniel Dantas, for his valuable knowledge and continuous support throughout the project. I also thank my classmate, Bruno Carrilho, for assisting me throughout the project. I am grateful to the Guarulhos campus of IFSP for providing the necessary infrastructure. Lastly, I thank my family and friends for their support during this exhaustive study period.

Disclosure. The authors report no conflicts of interest in this work.

References.

(1) SACOMANO, J.B. et al. Industria 4.0: Conceito e Fundamentos. São Paulo: Editora Edgard Blücher Ltda., 2018.

(2) NOLDIN, Jr., Contribuição ao estudo da cinética de redução de briquetes auto-redutores. Rio de Janeiro: PUC-Rio, 2002.

(3) ACATECH. Recommendations for implementing the strategic initiative Industry 4.0. Acatech—National Academy of Science and Engineering, 2013.

(4) DRATH, R. et al. Industry 4.0: Hit or hype? IEEE industrial electronics magazine, v. 8, n. 2, p. 56–58, 2014.

(5) OLIFF, H., & Liu, Y. Towards Industry 4.0 Utilizing Data-Mining Techniques: A Case Study on Quality Improvement. Procedia CIRP, 63, 167–172, 2017.

(6) POSADA, J. et al. Visual Computing as a Key Enabling Technology for Industrie 4.0 and Industrial Internet. Computer Graphics and Applications, IEEE. 2015.

(7) BODGAN M. David Irwin. Industrial Communication Systems. CRC Press, 2011.

Authors ORCID

Pelegrino JV - https://orcid.org/0009-0002-5362-9753.
Souza CMP - https://orcid.org/0000-0002-3741-1036.
Silva WCS - https://orcid.org/0000-0001-8128-281X.
Dantas RD - https://orcid.org/0009-0002-9606-7244