

Article

Not peer-reviewed version

---

# Optimizing Productivity Indicators through the Implementation of Industry 4.0 Technologies

---

Juan Emigdio Soto Osornio , [Armida González Lorence](#) , Jaime Navarrete Damian ,  
Cornelio Morales Morales , [José Gabriel Ayala Landeros](#) \*

Posted Date: 2 February 2024

doi: 10.20944/preprints202402.0131.v1

Keywords: Industry 4.0; Internet of Things; Programming Algorithms; Data Analysis; Digital Twin; Productivity Indicators



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

# Optimizing Productivity Indicators through the Implementation of Industry 4.0 Technologies.

Juan Emigdio Soto Osornio <sup>1</sup>, Armida González Lorence <sup>2</sup>, Jaime Navarrete Damian <sup>3</sup>, Cornelio Morales Morales <sup>1</sup> and Jose Gabriel Ayala Landeros <sup>4,\*</sup>

<sup>1</sup> Departamento de Ingeniería Electrónica, Instituto Tecnológico de San Juan del Río, TecNM, San Juan del Río, Querétaro, México, C.P. 76800; emigdio.so@sjuanrio.tecnm.mx; cornelio.mm@sjuanrio.tecnm.mx

<sup>2</sup> Departamento de Ingeniería en Sistemas Computacionales, Instituto Tecnológico de San Juan del Río, TecNM, San Juan del Río, Querétaro, México, C.P. 76800; armida.gl@sjuanrio.tecnm.mx

<sup>3</sup> Subdirección Académica, Instituto Tecnológico de San Juan del Río, TecNM, San Juan del Río, Querétaro, México, C.P. 76800; jaime.nd@sjuanrio.tecnm.mx

<sup>4</sup> Departamento de Ingeniería Industrial, Instituto Tecnológico de San Juan del Río, TecNM, San Juan del Río, Querétaro, México, C.P. 76800; gabriel.al@sjuanrio.tecnm.mx

\* Correspondence: gabriel.al@sjuanrio.tecnm.mx; Tel.: +52 4274271478

**Abstract:** Key performance indicators are essential for any company to enhance its competitiveness. However, a common issue is the manual collection of production data using record sheets, which are later digitized on desktop computers and processed in Excel sheets. Shockingly, 41% of companies in Mexico have no plans for digitalization. This work aims to develop a Real-time Productivity Indicator Monitoring System using Industry 4.0 technologies for a medium-sized manufacturing firm specializing in die-cutting automotive parts. The Advanced Quality Planning methodology was applied to develop the Conceptual, Product, and Process Engineering and its validation. The experimental tests demonstrated a reliability level of 99.98%, which exceeded initial expectations. The primary contributions presented in this work are related to the development of algorithms, communication protocols, data analysis, and general optimization of the calculation process. The study's only limitation was the short experimentation time.

**Keywords:** industry 4.0; internet of things; programming algorithms; data analysis; digital twin; productivity indicators

## 1. Introduction

Based on the Economic Censuses conducted by INEGI in 2019, the manufacturing sector in Mexico was responsible for generating 48.2% of the national total gross production, making it the most significant contributor. This sector comprises 579,828 economic units, which accounts for 12.1% of the country's total. Microenterprises comprise most of these units, representing 93.7% of all economic units, while small, medium, and large companies account for 4.2%, 1.3%, and 0.8%, respectively. Mexico's small and medium-sized business markets have 24,353 and 7,538 economic units, respectively. In the state of Querétaro, according to DENUÉ [1], there are 1,086 small and medium-sized manufacturing companies, representing the study's potential market. Additionally, the San Juan del Río municipality has 157 companies operating within this.

Companies use various performance indicators to measure their operations. These include PPk and CPk for process ability, different types of control charts, Mean Time Between Failures (MTBF) and Mean Time to Repair (MTTR) for Total Productive Maintenance (TPM), and others. One important indicator is Overall Equipment Efficiency (OEE), which depends on three factors: Availability (D), Speed (R), and Quality (C). To calculate these indicators, it is necessary to collect production data across the entire supply chain, both internal and external.

Calculating performance indicators (KPIs), such as Overall Equipment Efficiency (OEE), is a common issue by manually collecting data from record sheets and performing calculations in Excel

sheets. This method can lead to uncertainty regarding the accuracy of the information due to human errors. Additionally, many companies still do not have access to these valuable indicators, which are crucial for decision-making [2].

It is crucial to promote the digitization of established Mexican companies. According to the International Data Corporation, 57% of Mexican companies are significantly involved with the Internet of Things. However, 41% have no plans for future digitization, indicating that the process is not advancing evenly across the country. Small and medium-sized enterprises lag the most behind [3].

This project aims to create a real-time productivity monitoring system for a medium-sized manufacturing company specializing in die-cutting automotive parts. The company is based in the municipality of San Juan del Río, Querétaro. Production data is collected using record sheets (as shown in Figure 1) and then digitized on desktop computers. The data is then processed in Excel sheets, which can take between 1 to 5 days to analyze depending on the availability of staff. This delay in analyzing the data has been identified as a problem that needs to be addressed by implementing a real-time monitoring system.

“There is a lack of reliable real-time productivity indicators for decision-making”.

Some of the potential causes found in the calculation process are:

- Incomplete or Altered Data
- Confusing and illegible writing
- Lack of use of information technologies
- Long data capture times
- Long processing times
- Standardization of reports
- Late decision making

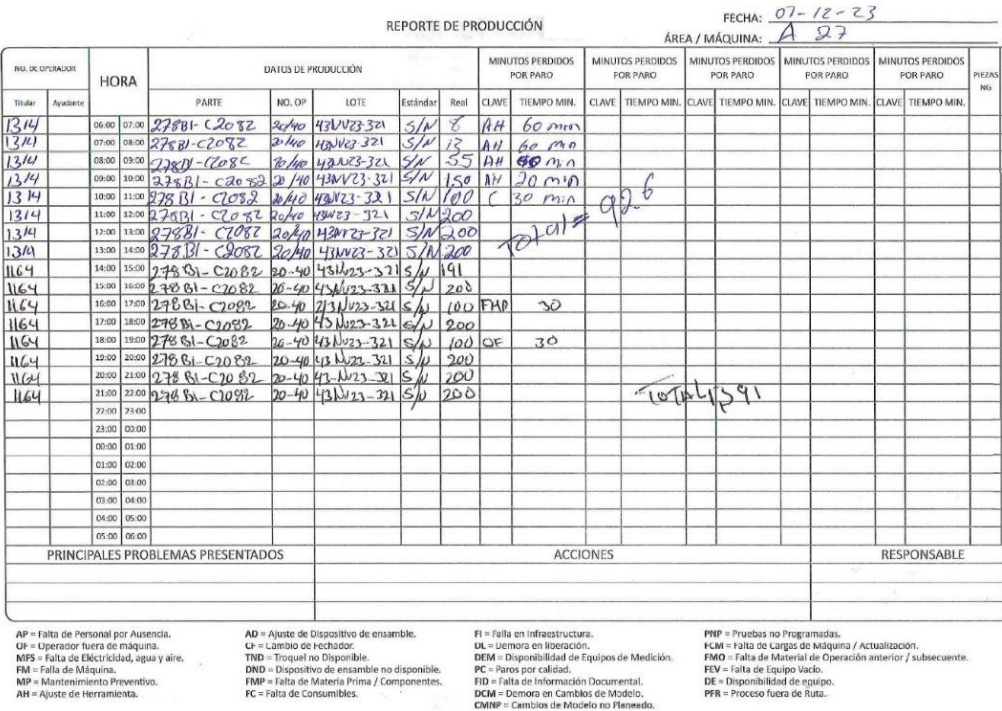


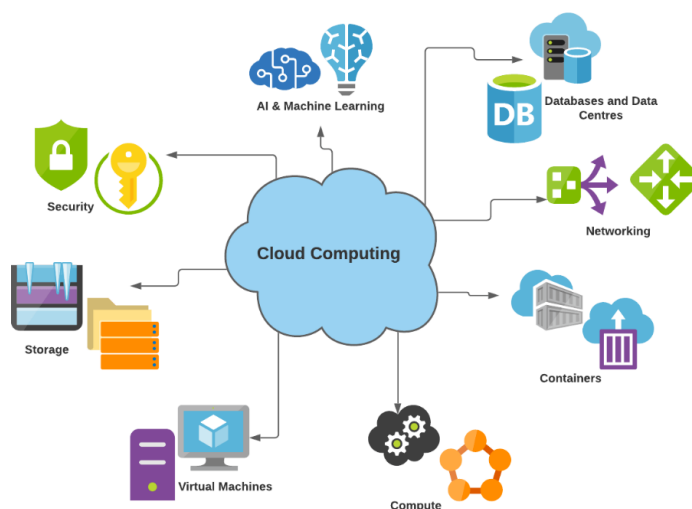
Figure 1. Production Report. Source: Tyrsa S.A. de C.V.

To design the proposal, we reviewed recent publications on Industry 4.0 and its related technologies, including Cloud Computing, Big Data, Industrial Internet of Things, Cybersecurity, and Artificial Intelligence. Additionally, we analyzed the tools used in these areas to ensure we comprehensively understand the project requirements.

According to the National Survey on Availability and Use of Information Technologies in 2021, 75.6% of the Mexican population, equivalent to 88.6 million people, used the Internet in Homes. This indicates that the number of internet users worldwide has reached 5.16 billion people, which accounts for 64.4% of the world's population, according to [4] statistics.

According to INEGI's report [5], 66.4% of homes in Mexico have internet access, totaling 24.3 million homes. Out of all the users, 96.8% prefer using a smartphone as their primary device to access the internet. These statistics clearly indicate the significance of cloud services in various social sectors, especially in the most fundamental areas like data storage services.

The major players in the cloud services industry are Amazon Web Services (AWS), Microsoft Azure (MA), and Google Cloud. IBM and Alibaba follow them, along with smaller "clouds" offered by companies like Adobe, Salesforce, Cloudflare, and Snowflake. Figure 2 illustrates the range of services these companies provide [6].



**Figure 2.** Main Cloud Computing Services. Source: [6].

Most proposals for implementing Architecture in Industry 4.0 acknowledge at least three layers in the smart factory environment. These layers are the cyber-physical System (CPS), which monitors processes, creates virtual copies, and makes decentralized decisions; the Internet of Things (IoT), which enables the CPS to communicate with its environment in real-time; and the Internet of Services (IoS), which provides users with improved and faster services.

According to Mejía et al. [3], Industry 4.0 comprises four main components: physical devices and workforce responsible for production functions, real-time communication networks, cloud resources for data storage and analysis, and applications that are the final products or services accessed by the consumer.

According to Shafique et al., the IoT architecture comprises three main components: hardware, middleware, and the presentation layer. The hardware component comprises the sensor nodes, their integrated communication, and interface circuits. The middleware component includes storage, analysis, and data management resources. Finally, the presentation layer comprises visualization tools compatible with various platforms to present the data to the end-user in an understandable form [7].

In order for a real system to communicate with a digital system, the devices need to be interconnected using Internet of Things (IoT) or Industrial Internet of Things (IIoT) technologies. Data from various sources, such as sensors, are sent through wired or wireless to a communication interface, which may be through a PLC or any other microcontroller that communicates with the physical-digital world. A digital model of the cyber-physical system can be created by connecting different objects, which provides real-time process data [8].

Implementing Industry 4.0 presents a challenge in standardizing communication and information management protocols. These protocols enable intelligent and connected production systems to predict and support production in real-time, leading to improved sustainability and organizational performance. In Slovakia's manufacturing industry, the adoption of I4.0 technologies is recognized as a means to increase productivity, sustainability, and energy efficiency while also enhancing competitiveness [9].

Artificial Intelligence (AI), for its part, uses tools such as Data Science, Data Mining, Machine Learning and Deep Learning. With these technologies, it is possible to develop applications for presenting key performance indicators and decision-making in real-time. The data may be stored in a public or private cloud database for processing and transformation into useful information. The fifth industrial revolution, Industry 5.0 (I5.0) and artificial intelligence (AI) offer the overwhelming potential to build an inclusive digital future by ensuring supply chain resilience and sustainability [10].

Globally, there is a growing interest in adopting new Industry 4.0 technologies, and different countries have adopted diverse strategies to achieve this. For example, China invested \$100 million in studying IoT industrial standards and technologies in Shanghai as part of its twelfth five-year plan. Significant efforts are being made to merge cross-domain research activities spanning machine-to-machine (M2M) communication, wireless sensor network (WSN), and RFID into a unified IoT framework. Experts predict that by 2025, every object will have an Internet node, leading to a surge in the number of devices connected to the Internet. According to INEGI [5], 500 billion devices will be connected to the Internet by 2030.

Mexico is working towards adopting Industry 4.0 technologies to advance towards the fourth industrial revolution. The Ministry of Economy has identified four fundamental pillars to work on: Development of Human Capital, Innovation, Creation of Clusters, and Technology Adoption. Siemens AG states that if Mexico continues to adopt these technologies and related policies and programs, it could become the world's fifth most influential economy by 2025, increasing the income of the Mexican industry by approximately three and a half billion pesos. Additionally, studies and surveys suggest that production costs could be reduced by 30%. Some of the Mexican government's initiatives in this area include México Conectado, the National Digital Strategy, the National Digital Literacy and Inclusion Program, and the Software and Innovation Industry Development Program [3].

Investment in science and technology has a crucial role in advancing society. In the state of Querétaro, there are seven clusters working in various sectors, including Automotive, Aerospace, Energy, Logistics Innovation, Medical and Health, Information Technologies, and Plastics. These clusters have existed for over a decade and have been actively developing new technologies to improve their respective industries [11].

The state of Querétaro is ranked second in the State Competitiveness Index, just below Mexico City. The state boasts 67 industrial parks and zones. In 2022, 55 investment projects were completed, creating 23,894 jobs, with an investment of 95,666 million pesos. From January to September 2023, 47 investment projects were finalized, generating 13,996 jobs and 26 billion pesos, mainly in the automotive and aeronautical industries. According to INEGI, in 2021, the state's Gross Domestic Product (GDP) increased by 6.2%, with secondary and tertiary activities showing the highest growth rate of 9.1% and 4.6%, respectively [12].

There are several research questions that can be posed regarding the implementation of a new system. For example, does the system implementation reduce delivery time? Does it enable a better understanding of the processes? Does it help to standardize the processes? Does it facilitate the measurement of different wastes? Does it allow for OEE calculation automation without errors? However, in our case, we will be focusing on measuring the system's reliability.

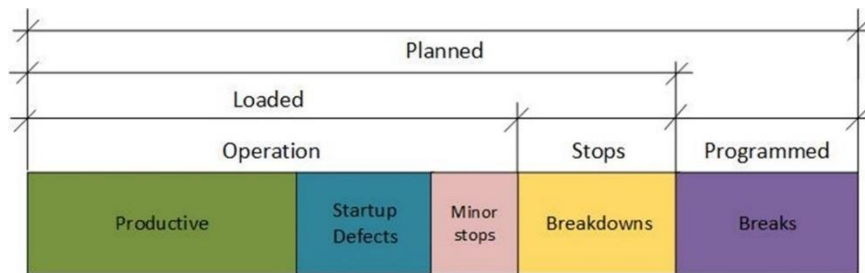
The aim of the project is to create a system that can register and monitor production variables in real-time using Industry 4.0 technologies. The system will calculate productivity indicators to help make timely decisions. To achieve this goal, the following research hypothesis has been proposed:

Null hypothesis: The system's reliability is at most 99%, and the alternative hypothesis is that it exceeds 99%.

Real-time access to productivity indicators through I4.0 technologies is valuable for facilitating work in various fields, especially production systems that utilize agile thinking. Agile methodologies are centered around Level Production (Heijunka), Visual Management, Long-Term Thinking, and Standardized Processes. Measuring these indicators is crucial to implementing continuous improvement techniques. However, the lack of automated measurement systems makes it difficult to apply these methodologies effectively.

The application of lean manufacturing is based on calculating OEE, which depends on product quality, equipment availability, and performance [13]. When implementing improvement strategies like Total Productive Maintenance (TPM), measurement techniques are necessary to evaluate process efficiency and effectiveness. This is where the need for the OEE indicator arises. The OEE helps prevent the underutilization of machines and production lines, providing a systematic method for measuring production objectives [14]. It's important to note that achieving an OEE between 85-95% is the goal for every organization to be considered a world-class company. The manufacturing industry is one of the economic sectors where Global Equipment Efficiency calculation has the most significant application.

Calculating the Overall Equipment Effectiveness (OEE) requires understanding the 6 major time losses that can occur in a production system, affecting this indicator. The total available machine time should take into account scheduled stops, such as preventive maintenance and breaks. Stoppages due to preparation and breakdowns that affect availability should also be considered. Performance is affected by minor stops and speed reduction. The start-up time should also be considered, which may lead to defective products due to machine adjustment (refer to Figure 3 for more information).



**Figure 3.** The 6 big losses. Source: Authors' compilation.

According to Nakajima [15], the OEE calculation requires three indicators: Availability (A), Performance (P) and Quality (Q), all expressed as percentages.

**Availability (D):** Refers to the time during which the machine is ready to produce, in relation to the planned production time (tpl). Availability reduces the planned time due to stops (start-up times, downtime, breakdowns, waiting). It is calculated using the following formula 2:

$$D = \frac{\text{Planned Time} - \text{Unproductive Time}}{\text{Planned Time}}$$

$$D = \frac{\text{Productive Time}}{\text{Planned Time}} \quad (1)$$

Taking into account that the productive time (tp) can be directly taken into account, it is easier to calculate the availability as follows:

$$D = \frac{tp}{tpl} \quad (2)$$

**Performance (R):** It is the ratio of the total units produced (TUP) to the total units expected (Te). The total units expected are calculated by multiplying the operation time (to) by the nominal capacity (cn). Microstops and reduced machine speed can affect performance. The operation time is calculated by adding the productive time to the start-up time and the time of minor stops (refer to formula 3).

$$R = \frac{\text{Total Units Produced}}{\text{Operation Time} \times \text{Nominal Capacity}}$$

$$R = \frac{TUP}{to \times cn} \quad (3)$$

**Quality (C):** It is the proportion of products that meet the quality requirements with respect to the total production achieved, excluding bad and reworked parts (Formula 4).

$$C = \frac{\text{Total de Unidades Buenas}}{\text{Total de Unidades Producidas}}$$

$$C = \frac{TUB}{TUP} \quad (4)$$

Formula 5 is used to calculate the OEE once the aforementioned parameters have been estimated. The OEE is used to evaluate the performance of equipment or an entire production line.

$$OEE = D \times R \times C \times 100\% \quad (5)$$

Another important concept for testing the quality of a product is reliability, which is defined as the probability of a physical asset performing the desired/required functions (operating without failures) under established operating conditions for a specified period of time. It is calculated using the Weibull distribution (Equation 6).

$$C = e^{-\left(\frac{MTTR}{MTBF}\right)} \quad (6)$$

Where:

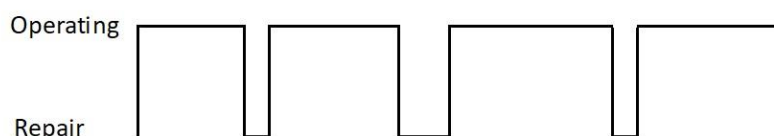
MTBF = Mean Time Between Failures

MTTR = Mean Time To Repair

$$MTBF = \frac{\text{Available Time} - \text{Dead Time}}{\text{No. Failures}} \quad (7)$$

$$MTTR = \frac{\text{Dead Time}}{\text{No. Failures}} \quad (8)$$

To calculate the Mean Time Between Failures (MTBF) and Mean Time To Repair (MTTR) of any equipment, it is necessary to conduct tests over a significant period of time. At least two failures must occur during the testing process to calculate the averages, as shown in Figure 4. Otherwise, the reliability of the equipment would be considered 100%, and there would be no need to calculate MTBF and MTTR [16].



**Figure 4.** Operation and maintenance times.

## 2. Materials and Methods

For the proposed solution, the Advanced Quality Planning (APQP) methodology used in the Automotive Industry was used, which considers four stages before mass production, namely: Conceptual Engineering, Product Engineering, Product, Process Engineering and Validation and Testing. It is based on advance planning of the Requirements, Needs and Expectations of clients, using specific analytical techniques according to the type of product. The methodology suggests working collaboratively and simultaneously through the different stages and achieving continuous improvement (Figure 5). It is important to mention that this methodology coincides with several requirements and techniques used in Project Management established in the PMBOK of the Project Management Institute (PMI). The stages considered in the APQP that are used in the development of the project are described below [17].

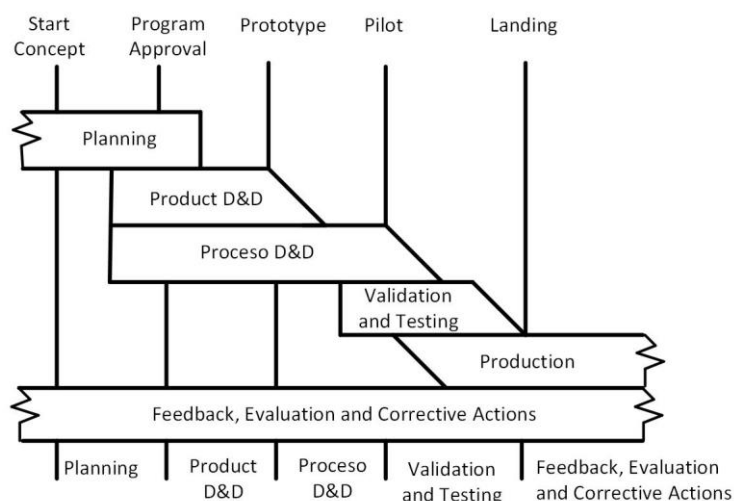


Figure 5. APQP.

### First stage. Conceptual Engineering

Design objectives were defined based on customer feedback, organizational strategy, benchmarking, product assumptions, and project management techniques such as the Business Case and the Project Charter. The defined objectives include quality and reliability targets, a preliminary list of materials, a preliminary process flow diagram, and a preliminary list of special product characteristics. Quality Function Deployment (QFD) is a widely used tool at this stage.

**The Voice of the Client:** To know the voice of the client, 3 techniques were used: Surveys, Interviews, and Review of cases in the literature. Each of these techniques is described below:

**Surveys:** A group of students from the Open Industrial Engineering program at the Instituto Tecnológico de San Juan del Río, who work in different companies, were surveyed to determine how many companies use Overall Equipment Effectiveness (OEE) as a measure of productivity. Initially, 30 students were sent the survey via Google Forms. Based on the responses, the probability of success ( $p$ ) was determined to be 0.39, and the probability of failure ( $q$ ) was 0.61. Formula 9 was then used to calculate the appropriate sample size, taking into account the number of manufacturing companies in the municipality of San Juan del Río ( $N = 157$ ), an error margin of 0.1, and a confidence level of 90% ( $\alpha = 0.1$ ). The resulting sample size was 47 companies.

$$n = \frac{N * Z_{\alpha/2}^2 * p * q}{E^2 * (N - 1) + Z_{\alpha/2}^2 * p * q} \quad (9)$$

Supplementing the required surveys resulted in a new value for  $p$ , 43%, and  $q$ , 57%. This indicates that a higher percentage of companies still don't use OEE as a productivity measure.

**Interviews:** A potential client was found from a medium-sized manufacturing company dedicated to die-cutting parts. A series of meetings were scheduled with the Engineering, Production, and Continuous Improvement departments to understand the company's situation regarding measuring productivity. During these meetings, it was discovered that the calculation process is not standardized, there is no standardized classification of minor stoppages, records are not filled in real-time, results are delayed, and decisions are not made based on the indicators. It was recommended that the process be automated to ensure better efficiency.

**Review of cases in the literature:** Reports from professional residencies, theses, research articles, and data provided by companies in the region were considered to determine the level of OEE at which companies in the region operate. The aim of this review was to assess the OEE level of companies in the region. The results of this review are in Table 1.

Table 1. OEE levels of companies in the region.

Company	OEE	Business	Source	Observation
Tyrsa SA de CV	68%	Die cutting of automotive parts	Engineering department	Punching machine No. 43
Inplarsa	7.8%	Polymer recycling and formulation	Production departament	Machine under remodeling
RR Donnelley Servicios, S.A. de C. V.	28.17%	Printing Services	[18]	Machine standards unknown
not specified	31.68%	Not available	[19]	Production line
Silos y Camiones S.A. de C.V.	67.63%	plant automation and maintenance	[20]	Based on weighing systems
not specified	63.53%	Pharmacist	[8]	Average value of one week

Source: Authors' compilation.

Table 1 shows different OEE values that vary depending on the operating conditions of each plant. In some cases, outdated technology requires more maintenance, while in others, there may be different ways of measuring that are not standardized. Calculation errors are also common due to a lack of knowledge of the methodology.

**Organizational strategy:** The proposed strategy is a collaborative effort between the ITSJR working group and the sponsoring company. Their shared interest is in digitalizing information and automating processes, creating a joint strategy called "Innovation Applied to Industry 4.0".

**Benchmarking:** There are various platforms available for developing OEE calculation applications. These platforms enable the collection, storage, analysis, visualization, and delivery of data in the form of Performance Indicators (KPIs) in real-time. For instance, PI System is a software from OSIsoft that offers such capabilities [8]. Other platforms include the Smart OEE System from ZEO, a Spanish company, PRODAIS SPC Consulting Group, and VersaCall System. These platforms collect data from different sources in the industrial plant and process it to present it to the final user [21].

One major challenge small and medium-sized businesses face is the high cost of implementing such solutions. Table 2 outlines the key features of the aforementioned solutions.

Table 2. Main Characteristics of Commercial Platforms.

Characterística	Sistema Smart OEE	AVEVA PI System	VersaCall	PRODAIS SPC
Data source	PLC's and Manual Inputs	HMI, Databases, EAM, Devices, IIoT, SCADA, DCS, Controllers, Cloud Systems, PLC	PLC's, Manual Inputs, Sensors, Contact Switches, Software Switches	PLC's, Barcode

Applications	Analysis of Productivity Indicators	Data Analytics, Time Series Analysis, Process Charts, Machine Learning	ANDON Systems, OEE Calculation, Machine Monitoring, Visual Management	ANDON Systems, OEE Calculation, Production Dashboards
Display	OEE Automated Reporting Dashboards	Standardized or Customized Reporting Dashboards	Production line visualization boards, Indicator Reports.	Standardized or Customized Reporting Dashboards
Data integration	Does not have information	Standardized data supported by AI platforms	Does not have information	Does not have information
User Control	Credentials by user type	Credentials by user type	Predefined users	Unlimited users
Accessibility	From anywhere and device	From anywhere and device	Cell phones, Tablets, 2-way radios, PC	Cell phones, Tablets, PCs, e-mail
Notifications	Custom Alerts	Custom Alerts	Audible and visual alarms	e-mail
Connectivity	Universal data capture devices	300 drivers	IIoT and Connectivity 4.0 equipment, as well as manual inputs	Barcode
Server	Local/Cloud/Virtual	Local/Cloud/Virtual	Local/Cloud/Virtual	Local
Storage capacity	Scalable for large volumes of data	Scalable for large volumes of data	Scalable for large volumes of data	Does not have information
Origin of Technology	Spain	UK	EEUU	México
Platform	Software OEE	PI System	Versa Vision	XL Lean Tools
Web page	<a href="http://www.sistemasoe.com">www.sistemasoe.com</a>	<a href="http://www.aveva.com">www.aveva.com</a>	<a href="https://www.versacall.com">https://www.versacall.com</a>	<a href="https://prodasis.com">https://prodasis.com</a>

Source: Authors' compilation.

Table 3 presents a list of technologies commonly used in process automation as part of supply chain management. The table includes details about the application areas, types of technologies used, sensors, and devices. This information serves as a representative sample of the various technologies used [22].

**Table 3.** IoT technologies used in the supply chain. Source: [22].

Application Domain	Application áreas	Technologies	Sensors and Devices
<ul style="list-style-type: none"> <li>• Supply chain management</li> <li>• Agricultural and food supply chain</li> <li>• Medical and healthcare supply chain</li> <li>• Reverse chain management</li> <li>• Asset supply chain</li> <li>• Additive manufacturing supply chain</li> <li>• School uniform supply chain</li> <li>• Electric vehicle supply chain</li> <li>• E-commerce supply chain</li> </ul>	<ul style="list-style-type: none"> <li>• Follow-up</li> <li>• Tracking and monitoring</li> <li>• Quality and Occupational Safety Risks</li> <li>• Information exchange</li> <li>• Traceability</li> <li>• Logistics</li> <li>• Inventory monitoring</li> <li>• Control and Monitoring</li> <li>• Quality tracking and monitoring</li> <li>• Warehouse management</li> <li>• Inventory management</li> <li>• Risk management</li> <li>• Security analysis</li> <li>• Manufacturing</li> <li>• Smart shipping</li> <li>• Decision making</li> </ul>	<ul style="list-style-type: none"> <li>• RFID</li> <li>• NFC</li> <li>• IoT</li> <li>• Cloud</li> <li>• Artificial intelligence (AI)</li> <li>• Blockchain</li> <li>• Smart tags</li> <li>• QR code</li> <li>• Agent based system</li> </ul>	<ul style="list-style-type: none"> <li>• GPS,</li> <li>• Humidity sensor</li> <li>• Temperature sensor</li> <li>• Pressure sensor</li> <li>• Gyroscopic sensor</li> <li>• Motion sensor</li> <li>• Weight sensor</li> <li>• Accelerometer</li> <li>• Hall sensor</li> <li>• Vision camera</li> <li>• Smoke sensor</li> <li>• occupancy sensor</li> <li>• Door sensor</li> <li>• Gas sensors</li> <li>• Level sensor</li> <li>• Light exposure sensor</li> <li>• Color sensor</li> </ul>

Source: Authors' compilation.

**Quality Function Deployment (QFD):** Figure 6 shows a matrix that relates the customer's demands to the quality characteristics of a product. The matrix assigns a relative importance score between 0 to 10 to each of the client's requirements. For example, if the requirement includes several machines, it is given the highest weightage. Additionally, the matrix assigns a score of 9 for a strong relationship between a customer requirement and a quality characteristic, 3 for a moderate relationship, and 1 for a weak relationship.

When evaluating the matrix, consider the degree of difficulty, where zero corresponds to easy and ten to extremely difficult. This method assigns weight to the product's different quality characteristics.

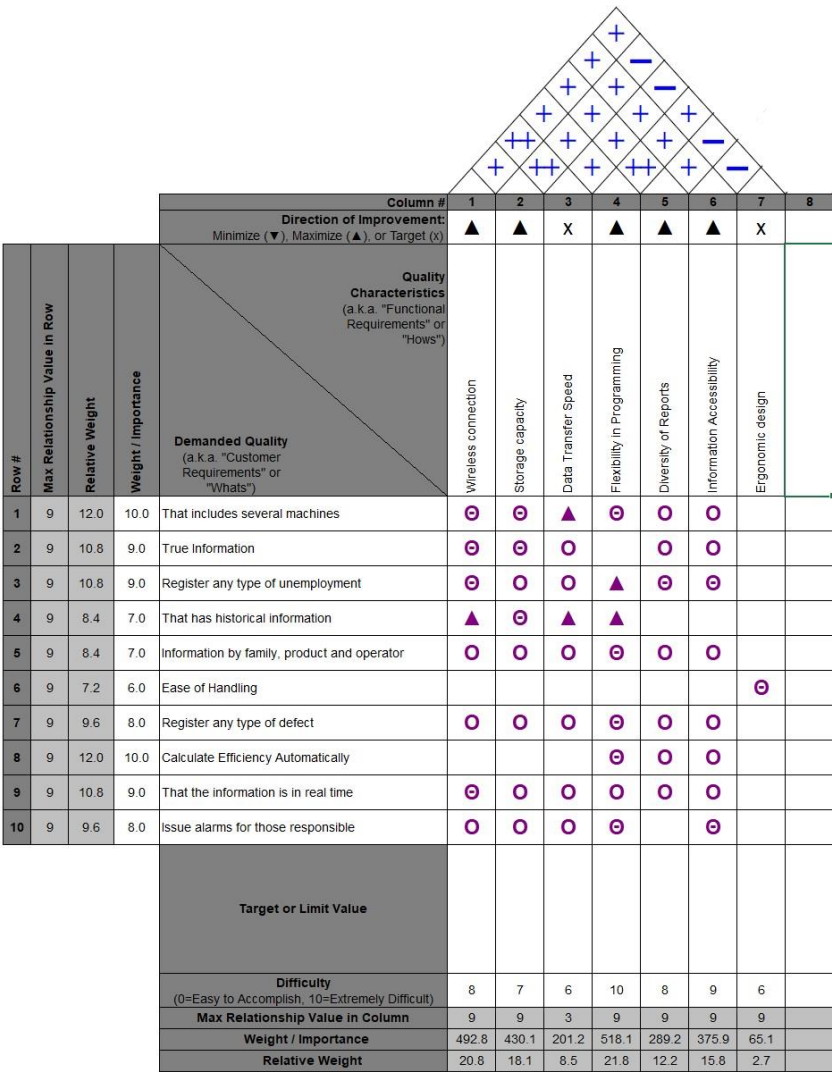


Figure 6. Product QFD. Source: Authors’ compilation.

The result of the quality Deployment Function is a list of the product's special characteristics that appear in Table 4.

Table 4. Main product features.

ID	Features	Weighing
1	Flexibility in Programming	21.8
2	Wireless connection	20.8
3	Storage capacity	18.1
4	Information Accessibility	15.8
5	Diversity of Reports	12.2
6	Data Transfer Speed	8.5
7	Ergonomic design	2.7
Source: Authors’ compilation		Total
		99.9

Furthermore, information security and user administration were regarded as significant features.

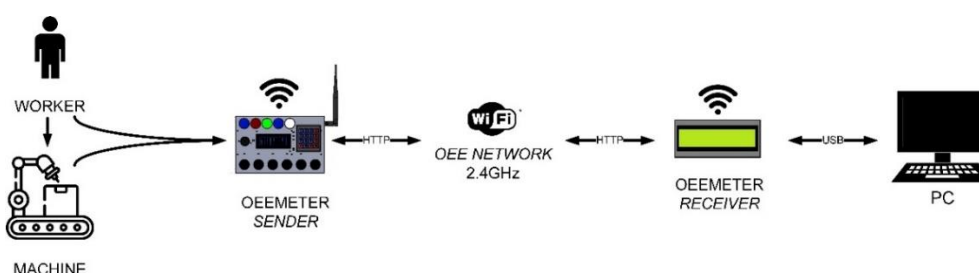
**Product assumptions:** Free software that allows for an affordable cost will be used.

**Design objectives:** Design the data Sender and Reception cards and the database and control panels with free software to reduce product costs.

**Quality and reliability objectives:** The system must work continuously and without errors with at least 99% reliability.

A preliminary list of materials and a process flow diagram are prepared at this stage. However, due to space constraints, they are not included in this section. These tools are presented in greater detail in the following sections. This stage's outcome is the product's conceptual design, consisting of four components (Figure 7).

1. Sender
2. Receiver
3. Local server
4. Query terminals



**Figure 7.** Conceptual Design of the OEE System. Source: self-made.

**Hardware Selection:** The OEE (Overall Equipment Effectiveness) indicator must be measured in real-time by continuously capturing availability, performance, and quality information. To achieve this, a micro processing unit (microcontroller) must interact with physical ports and manipulate census variables at the software level. This component will execute the main algorithm for calculating the indicator and managing the environmental variables.

The microcontroller is the calculation unit but cannot interact directly with the environment. To facilitate this interaction, peripherals that act as information translators, both for output and input, are implemented. When measuring the OEE, the microcontroller must interact directly with both the operator and the machine to capture each indicator. Therefore, the following aspects have been defined:

- Determination of the state of the machine using push buttons and typing of codes that define a particular state of loss, which will be considered for the calculation of Availability.
- An electrical signal is acquired from the machine to represent the production of a piece. The source can be an electronic or mechanical sensor or an AC/DC signal. This signal is used to accurately count the number of cycles completed and accumulated operating time, to obtain the Performance parameter.
- A mechanical button with two possible states- On and Off- determines product quality. If the pieces produced meet quality standards, the button is on; otherwise, It Is off.

The real-time measurement section was reviewed, and the communication section was examined for the various frequencies available, such as 433 MHz, 720 MHz, 2.4 GHz, and 5 GHz. The appropriate communication protocol, speed, bandwidth, technology, and infrastructure to establish a suitable network were considered before choosing 2.4 GHz, which is standardized by the Institute of Electrical and Electronics Engineers (IEEE). There are chips available that have both a microcontroller and a wireless communication module, such as the Expressif family with the ESP32 and ESP8266 and Raspberry Pi which comes with a module that can connect to 802.11 networks, opting for the ESP32 due to its processing features, memory, and libraries developed for a greater number of applications.

This system is designed to be used in an industrial environment that follows safety standards and has nominal working voltages. The devices mounted in control cabinets require protection and

a proper architecture. To meet these requirements, the circuit must have the capacity to be powered with voltages between 24 VDC and 120 to 240 VAC. The signal input should also be isolated from the complete system to protect it against faults and discharges. To safeguard against electrical failures, the system is based on the architecture of a UPS device. A UPS provides backup electrical energy to connected devices in case of power outages.

Our system will use a fan to regulate the voltage to the required level for each component using DC-DC voltage converters. These converters generate heat, so it is essential to confine them in a cabinet and extract the heat using a simple fan.

A set of sturdy 22 mm buttons are designed to swiftly capture information and an alphanumeric keyboard is provided to input pre-established codes, both of which are placed on the exterior. To display detailed information about digitization, there are 22mm LED light indicators and a screen that can be an LCD, OLED, TOUCHSCREEN, or any other type. In order to navigate through different information screens, the concept of using a pointer is introduced, which can be a rotary encoder, button crosshead or a joystick. A buzzer is also included as an audio component that emits a 2 kHz sound when activated and is easy to use.

The data will be stored on a server that has access to a database. Information will be downloaded from each machine wirelessly using WiFi. To facilitate communication, we will use protocols such as HTTP, ModBus, MySQL, TCP-IP, etc. In our case, we have chosen to use the HTTP protocol for GET requests.

The protocol in place will ensure the security of the information. Additionally, a local VPN will isolate the network infrastructure, creating a local area that does not require internet access. This will keep the data traffic completely private. To access the internet while maintaining data security between systems, a device named a receiver has been incorporated into the design. The receiver acts as a bridge between the storage server and the network with all the systems that monitor the indicators, making them emitters. All these systems are connected to a wireless network, and the receiver is connected to the server using a USB 2.0 cable. It manages the information flow for security reasons.

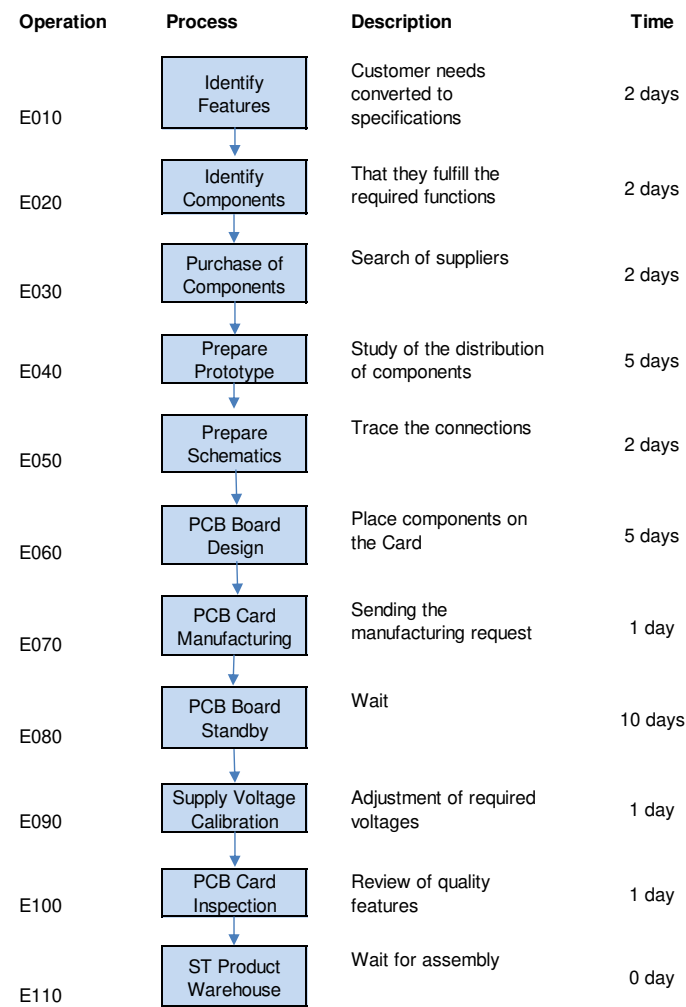
It is important to note that the electronic system has a multitude of components that it must control, which will be embedded in an electronic board. To manage peripherals such as buttons, keyboard, JoyStick pointer, fan, sound device, etc., a lower-power microcontroller will be used as a slave. This will free up processing power, allowing better management of tasks carried out by the ESP32. The ESP32 will perform the OEE calculation and transmit all the information about this indicator in real-time. The communication between these two microcontrollers will be through their UART (Universal Asynchronous Receiver-Transmitter) ports, which are serial ports used to send and receive information of any type, with the only characteristic being the transmission speed given in bauds per second.

**Software Development:** Open-source platforms like Visual Studio Code version 1.84.2 and Arduino IDE version 2.2.1 were utilized to develop the software. Both are development environments that enable editing, compiling, and debugging code. These platforms allow the implementation of algorithms on the ESP32 and ATmega 328 microcontrollers responsible for information management and the interface program between the receiver and storage server. Programming languages such as C++, Python, and JavaScript were employed for web development, and the SQL database manager was used to manage the database.

### **Second stage. Product Engineering**

This stage involves several preparatory steps to build a prototype. It starts with creating a Process Flow Diagram (Figure 7) and designing Programming Algorithms (Figures 8 and 9). The Selection of Components (Table 5) is done based on Conceptual Engineering. The prototype (Figure 10) is built to understand the design's nature. Techniques such as Analysis of Design Failure Modes and Effects (AMEFDs) and Design for Ease of Manufacturing and Assembly are used. Throughout this stage, the Drawings must contain Engineering specifications and materials. Finally, a Control Plan is concluded. To ensure the quality of the design, Design Reviews and Verifications must be carried out with the departments involved at every step of the process.

**Diagrama de Flujo del Proceso:** The process for designing and manufacturing the Emitter card is depicted in Figure 7. After manufacturing the prototypes and testing their functions, the card is sent to an external supplier who integrates all the components into a system.



**Figure 7.** Sender Card Process Diagram. Source: Self-Made.

**Algorithm Design:** The algorithms used to calculate the Overall Equipment Effectiveness (OEE) are divided into different stages such as initialization, calculation of Availability, Performance, and Quality. There is also an algorithm for monitoring power supply and wireless connection. Figure 8 shows the OEE initiation, monitoring diagram, and Availability calculation. On the other hand, Figure 9 contains the diagrams for calculating Performance and Quality.

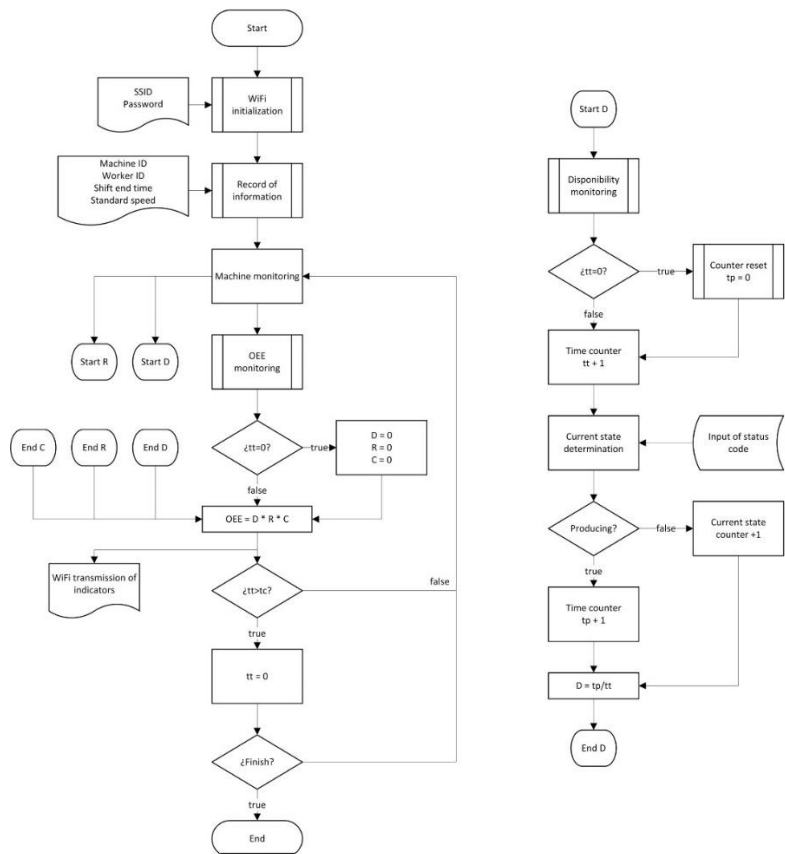


Figure 8. OEE initiation, monitoring diagram and availability calculation. Source: Self-Made.

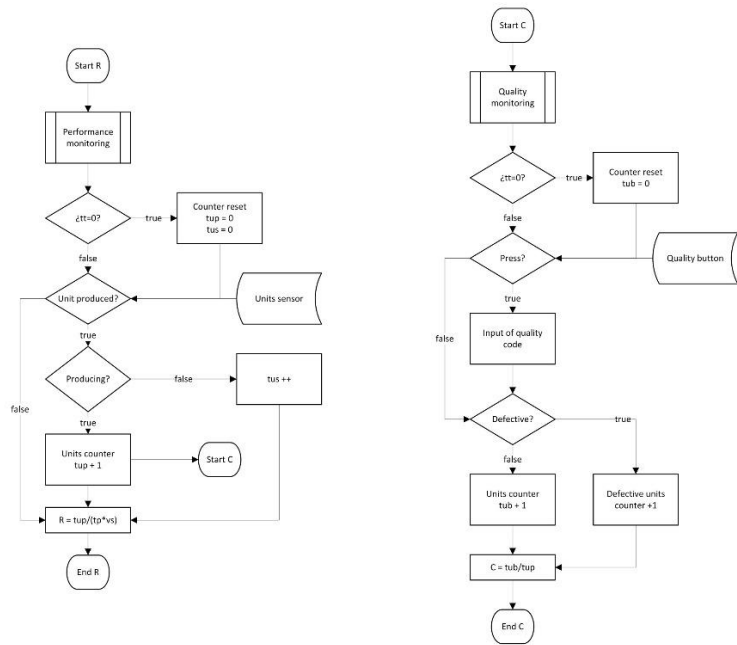


Figure 9. Algorithm for calculating Performance and Quality. Source: Self-Made.

**Component Selection:** To comply with the specifications of the Conceptual Engineering, the different components were selected among the alternatives existing on the market. Table 5 shows the selected key elements and the function they fulfill.

**Table 5.** Components of the Electronic System.

ID	Name	Description
1	Microcontroller ESP32 WiFi 2.4 GHz	The ESP32 module is an all-in-one, integrated, and certified WiFi/Bluetooth solution that provides not only the wireless radio but also an integrated processor with interfaces to connect with various peripherals. Processor with two cores, with operating frequencies between 80-240 MHz.
2	Microcontroller ATmega 328	The ATmega 328 microcontroller contains 13 digital I/O and 5 analog I, with UART and I2C communication capacity, it contains sufficient processing RAM for tasks carried out with a processing speed of 16 MHz
3	Switching power supply DCDC type Step-Down LM2596	Switches from direct voltage to step-down type direct voltage have an efficiency of close to 90%
4	Charger Module HX-2s-JH20	Battery protective charger module with BMS (Battery Management System) allows simultaneous charging of 2 18650 type Lithium batteries
5	Lithium battery 18650, 3.7 V, 2000 mAh	Energy storage, rechargeable, useful for portable applications
6	4 x 4 membrane matrix keyboard	A device that groups several buttons and allows them to be controlled using a smaller number of conductors than would be needed individually
7	UPS circuit based on LM358 amplifier, MosFET IRF44N, TDS-1202 mechanical relay	A device that allows the flow of electrical energy through batteries when the electrical supply fails. Performs analog sensing through an OpAmp that controls a relay, directing the source to be used
8	JoyStick	Auxiliary input device for applications that provide alternatives to keyboard and mouse use. The joystick provides positional information within a coordinate system that has absolute maximum and minimum values in each axis of motion
9	LED indicators	An LED indicator emits light when an electric current is applied to it, the color depends on the meaning you want to give it in the system.
10	Buttons	It is an electromechanical device that is used to make a momentary electrical connection.
11	BJT NPN Transistor Power Conversion Circuits	Process of using a low-power signal, typically generated by a control device such as a microcontroller or integrated circuit, to control and manipulate a high-power load
12	Optoisolated circuit for electrical signal acquisition	It is Commonly used to acquire electrical signals when it is necessary to provide electrical isolation between the control and acquisition parts to ensure safety and reduce the risk of interference.
13	Display LCD	Proporciona una interfaz visual para la presentación de información en una amplia variedad de dispositivos electrónicos, ofreciendo eficiencia energética, delgadez y versatilidad en la visualización de datos

14    Active buzzer

An electroacoustic device is a tool that produces audible sounds.

Source: Authors' compilation.

**Block Diagram of the Electrical System:** Once each component for a particular task was identified, a block diagram was created to correlate the functions of each component, as shown in the figure below:

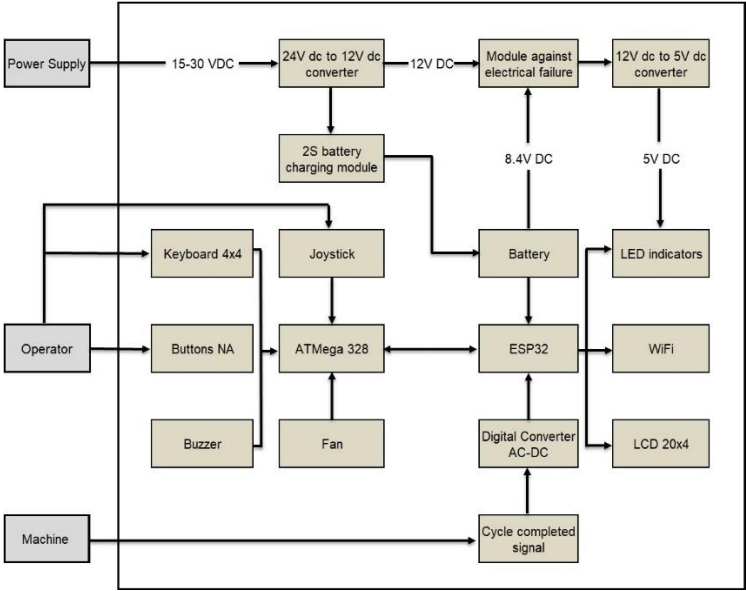


Figure 10. Block Diagram. Source: Self-Made.

**Initial Prototype:** Once the relationship between each component was established, the prototype was assembled using test boards (Protoboards). This was done to identify any design errors or necessary adaptations and to obtain electrical diagrams based on connectivity tests.

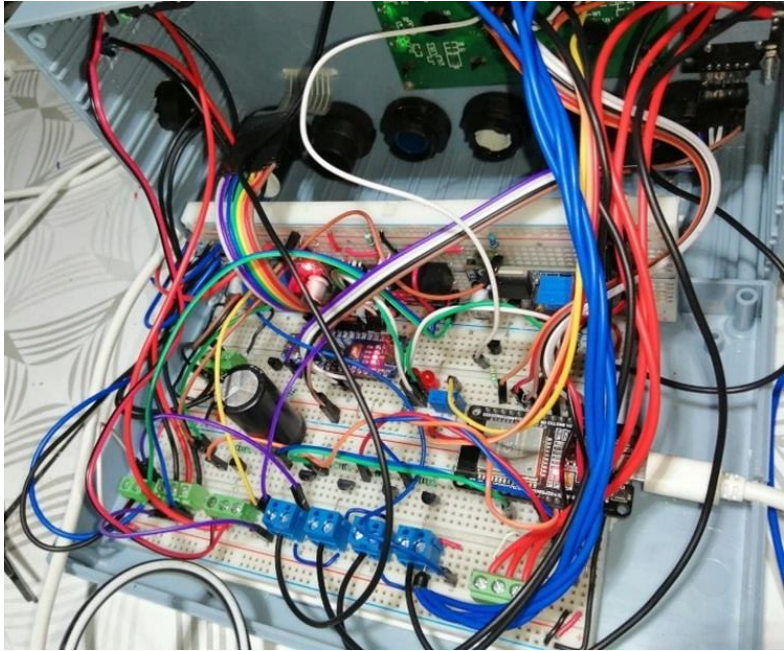


Figure 11. Prototype Assembly.

**Electronic Circuit Design:** After defining the ports and connections between the components, we proceeded to draw the schematics for printing the test PCB board. The Protoboard assembly uses a development board that is specifically focused on testing, and the system intends to have all the

components embedded, except for the charging module and external components that will be mounted on the cabinet. All the remaining components must be integrated into a single plate. We followed the recommended circuit for the use of the ESP 32 microcontroller, as well as the reference circuit to be able to program and power it, based on the documentation on the Espressif page. We repeated the same process for the following components: ATmega 328P Microcontroller, Peripherals, Power Supply, Battery Bank, UPS Power Failure Control, Main Power Supply, and AC/DC signal converter.

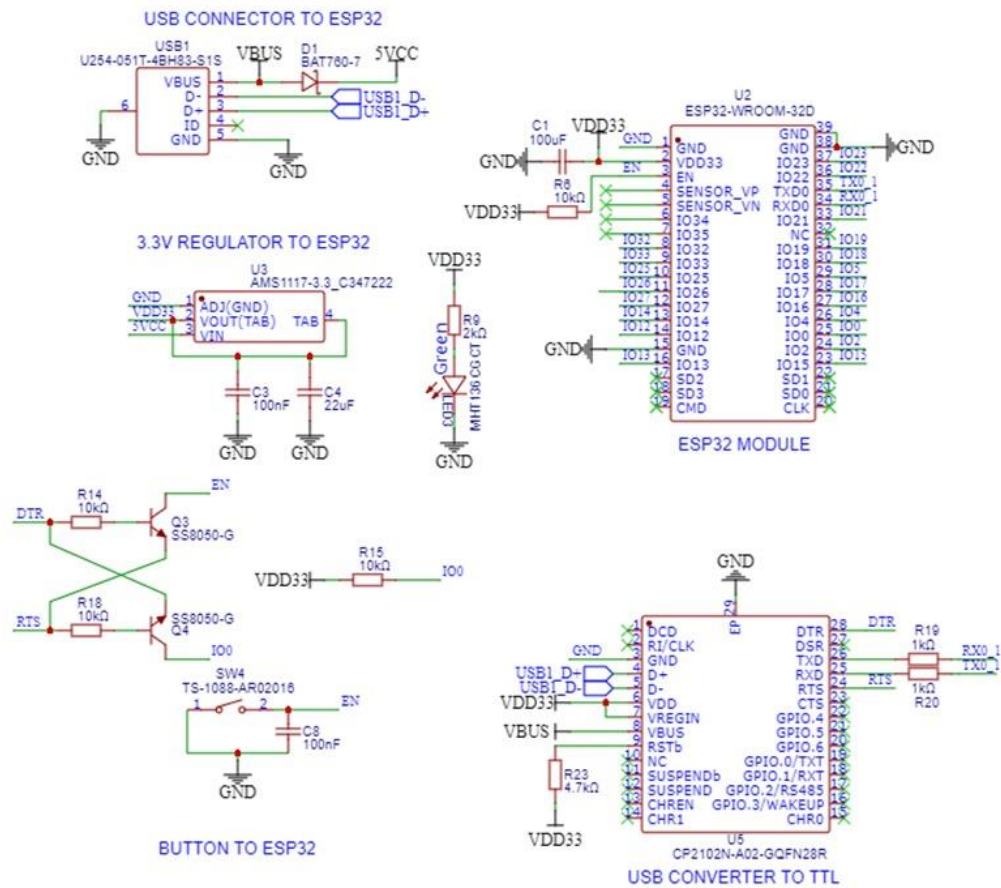


Figure 12. Circuito del Microcontrolador ESP 32.

**Integrated Circuit Design:** The routing process begins once the schematics have been created in EasyEDA software version 6.5.37. This involves physically drawing the board and placing each component with its respective electrical connection (using IPC electrical standards). The same process is carried out for designing the receiver board, resulting in the final integrated circuit on a single board (Figure 13).



Figure 13. 3D view of the transmitter's PCB (printed circuit board).

**Analysis of Design Failure Modes and Effects (AMEFDs):** The AMEF is a risk analysis tool that helps identify potential problems and their impacts in product or process design. Its aim is to prevent

problems and their negative effects and facilitate strategies to mitigate them. Tables 6 and 7 display the Design of AMEFs for the Issuer. Table 6 evaluates the most critical failure modes, while Table 7 outlines the actions taken to address them.

Table 6. AMEFD for the Sender.

PROCESS FUNCTION	POTENTIAL FAILURE MODE	EFFECTS OF POTENTIAL FAILURE	S	C	O	CURRENT PROCESS CONTROLS	D	NPR
Identification of Characteristics	Lack of specifications	of Dissatisfied customer	8		6	Requirements of the client	6	288
Component Identification	Components out of Specification	Purchases cannot be made	8		4	Component specifications	4	128
Purchase Components	Components out of Specification	Cannot build prototype	4		4	Component Specifications	4	64
Develop prototype	Lack of components	The prototype does not work	4		4	Material's list	2	32
Prepare Schematics	Wrong strokes	The card does not work	8		4	Prototype operation	4	128
PCB Board Design	Wrong strokes	The card does not work	8		4	Prototype operation	4	128

Source: Self-Made.

Table 7. Actions taken in the design of the Sender.

RESULTS OF THE ACTIONS				
Actions taken	S	O	D	NPR
Conduct interviews with users	4	2	4	32
Conduct reviews with a multidisciplinary team	4	2	2	16
Apply supplier development methodology	2	2	2	8
Prepare checklist	2	2	2	8
Create pattern templates	2	2	2	8
Create pattern templates	2	2	2	8

Source: Self-Made.

Third Stage. Process Engineering

During this stage, we receive the Emitter Card from our supplier, PCBLAB. We then conduct a reception inspection based on the Process Flow Diagram, the Analysis of Process Failure Modes and Effects (AMEFPs - Table 8), and the Pre-release Control Plan (CP - Table 9). Additionally, we focus on developing different programs necessary for device communication and product operation. These programs are based on the algorithms that were mentioned in the previous stage.

Table 8 shows the main failure modes for the emitter during the card reception and inspection stages before assembly. On the other hand, Table 9 presents some actions taken to mitigate the possible failure modes.

Table 8. AMEFP for the Emitter.

PROCESS FUNCTION		POTENTIAL FAILURE MODE		EFFECTS OF POTENTIAL FAILURE		CURRENT PROCESS CONTROLS		S	C	O	D	NPR
PCB Manufacturing	Card Capture platform	Capture errors	Defective card	8	4	Card Design	4	128				
Voltage Calibration		Voltage out of range	Card Damage	6	2	Testing bench	4	48				
PCB Card Inspection		Shorts between tracks	Component Damage	6	4	Testing bench	4	96				
PCB Card Inspection		Cold welding	There is no continuity	4	4	Testing bench	4	64				
PCB Card Inspection		Dirt between pins	Component Damage	4	2	Visual control	2	16				
PCB Card Inspection		Card Dimensions	Bad assembly	4	2	Vernier	2	16				
PCB Card Inspection		Component orientation	Assembly problems	4	2	Visual control	2	16				
PCB Card Inspection		Power unit heating	Component Damage	6	2	Thermocouple	2	24				
PCB Card Inspection		UPS Response Time	Loss of information	6	2	Oscilloscope	2	24				

Source: Self-Made.

Table 9. Actions taken in the design of the Emitter.

RESULTS OF THE ACTIONS				
Actions taken	S	O	D	NPR
Prepare procedure	2	2	2	8
Place position guides	2	2	2	8
Prepare testing guide	2	2	2	8
Prepare testing guide	2	2	2	8
Perform cleaning	2	2	2	8
Give design flexibility	2	2	2	8
Give instructions to the supplier	2	2	2	8
Include heatsink and fan	2	2	2	8
Define operation parameters	4	2	2	16

Source: Self-Made.

Based on failure modes, the Pre-Launch Control Plan is presented, which includes process specifications and evaluation methods, among other things (Table 10).

Table 10. Pre-launch Control Plan.

Process Flow		Machine	Characteristics			C l a s s	Methods				
Op.	Process Name	Device/Tool	I T E M	Process Characteristic	Product Characteristic		Product/ Process Specification	Evaluation method	Frequency and sample size	Control method	Reaction plan
E090	Voltage Calibration	Test Bench			Output voltage		12 V $\pm$ 0.5	Multimeter	100%	Record	Recalibrate
E090	Voltage Calibration	Test Bench			Output voltage		12 V $\pm$ 0.5	Multimeter	100%	Record	Recalibrate
E100	PCB Card Inspection	Test Bench			Track Continuity		No shorts	Multimeter	100%	Record	Rework
E100	PCB Card Inspection	Test Bench		Component welding			continuity in connections	Multimeter	100%	Record	Rework
E100	PCB Card Inspection	Test Bench		Cleaning			continuity in connections	Multimeter	100%	Record	Cleaning
E100	PCB Card Inspection	Test Bench			Card Dimensions		a = 70 $\pm$ 1 mm l = 240 $\pm$ 1 mm e = 2 $\pm$ 0.1 mm	Vernier	100%	Record	Rework
E100	PCB Card Inspection	Test Bench			Component orientation		Correct Orientation	Template	100%	Record	Decline
E100	PCB Card Inspection	Test Bench		Power Unit Heating			Maximum temperature 50 oC	Thermo-couple	100%	Temporal Graph	Check System
E100	PCB Card Inspection	Test Bench		UPS Response Time			Maximum 50ms	Oscilloscope	3 pcs at the beginning	Response Graph	Change brand

Source: Self-Made.

**Program Development:** The programs were developed using the Arduino platform and some of its libraries. Visual Studio Code was used as the code editor and Python was used as the programming language. The programs were based on the algorithms presented in Figures 8 and 9, which are described below:

**Receiver Source Code:** This device is designed to connect to a PC via the USB protocol. In order to communicate with devices using serial interfaces like UART (Universal Asynchronous Receiver/Transmitter), a set of AT (Attention) commands needed to be established. These commands are commonly used in communication devices, modems, and GSM modules. You can find a list of these commands in Table 11.

Table 11. Communication Commands for the Receiver.

ID	Comand	Function
1	inf?	Request information from the PC to identify the Receiver as a compatible device.
2	inf=	Respond to the request for information: inf = OEEMETER model: Terminal versión: 1
3	rtc?	The recipient requests real-time information from the PC to synchronize the internal clock.
4	rtc=	Serves real-time request: rtc=15:45:25 04/09/2023
5	net?	The PC receives a request for information about the network it needs to connect to, which is already saved in memory.
6	net=	Serves the network request: net=0 (default OEE NETWORK)
7	net:	Command to be able to transmit information via WiFi to a machine, the syntax is: net(machine number)(message) e.g. Net:1Hello
8	dat?	Command to request information from a particular machine, accompanied by the net: command, e.g. Net:2dat?
9	dat=	Handles the reception of information from the machine dat=18925189251892518925

Source: Self-Made.

After establishing these commands, the next step was to design the instruction flow in the source code to ensure proper communication.

```
// Declaration of global variables, objects, and libraries to be used..
//DUAL CORE
//WIFI NETWORK
//NETWORK
//IP CONFIGURATION
//WEBSERVER
//WIFI COMMUNICATION
//GENERAL
//TRANSFER RX/TX
//IP FROM EACH STATION
//WIFI FUNCTION READ AND WRITE
//UART COMMUNICATION
//LCD DISPLAY
//REAL TIME CLOCK RTC
//GPIO's
//LEDS
//GENERAL VARIABLES
```

Firstly, the SETUP program is created, which is only executed once when starting the program (**Appendix A1**). This routine initializes the communication ports. After the Setup has been developed, the program will use dedicated functions for communication management and port management (**Appendix A2**). Finally, we have the main program (**Appendix A3**). The ESP32 microcontroller has two cores, which come in handy for running multiple processes at the same time, making the program faster and the code lighter for each core.

The communication software used to exchange data between the Receiver and the PC and store it in a database was developed using Python language (**Appendix A4**). The Visual Studio Code text editor was used to create this source code. When executed, this code scans the USB ports and filters compatible devices using the `inf?` command. Once a compatible device is found, the code establishes communication with it and proceeds to synchronize the data efficiently. The code then manages the incoming data from each transmitter and stores it in the database.

*Sender Source Code:* The same methodology was followed as for the Receiver, considering that the Transmitter has two microcontrollers, the ATmega 328P, which is responsible for controlling peripherals (**Appendix A5a**) and the ESP32, which is responsible for calculating indicators and wireless communication (**Appendix A5b**). Communication between these two devices is also through AT commands (Table 12).

**Table 12.** Communication commands for the ESP32 microcontroller.

ID	Command	Function
1	inf?	Requests ESP32 for information to identify established connectivity
2	inf=	Respond to the connectivity request: inf=Station,Model:Slave,V:1.0
3	sta?	Request for information on each peripheral status
4	sta=	Respond to the request for states: sta=joy;0;but;0,out{0,0,0},fan=75,pws=12,bat=8
5	joy=	Returns the value of the pointer joystick: joy=5 (arriba)
6	key=	Returns the keyboard value: key=A (key A pressed)
7	but=	Returns the value of the buttons: but=3 (button 3 pressed)
8	fan=	Sets the fan speed (in percentage): fan=75
9	buzzer	Intermittently activates the sound element
10	out_1	Set output 1 status: out_1=1 (turn on output 1)

11	out_2	Set output 2 status: out_2=1 (turn on output 2)
12	out_3	Set output 3 status: out_3=1 (turn on output 3)
13	reset	Reset the microcontroller to default values for each peripheral
14	pws?	Request the voltage of the external power supply
15	bat?	Request battery voltage
16	pws=	Returns the power supply voltage value: pws=11.98
17	bat=	Returns the battery voltage value: bat=8.32

Source: Self-Made.

The ESP32 Microcontroller, like the terminal, must have wireless communication sections, but it will also be in charge of presenting the data through an LCD screen, so the structure of the program is based on screens that, as the program progresses, initialization and configuration, it moves and generates the appropriate interface.

**Data Base Desing:** The database management system for our project involved using two tools - HeidiSQL and MySQL. HeidiSQL is a user-friendly SQL editor that provides an intuitive interface for managing the MySQL database. It simplifies database management and optimizes development efficiency. MySQL was used as the primary DBMS for the project.

The MySQL Database architecture efficiently manages dynamic and static data, providing a comprehensive platform for machine operation monitoring and analysis.

This system is composed of five tables. The first table is named "Dynamic Data" and it serves as the heart of real-time monitoring for the machines. This table stores essential information such as the current status of the machine, the duration spent in that state, and its activation (referred to as "Table 13").

Table 13. Dynamic Data.

IND	I	ENA	STA	ESTADO_	ESTADO_A	TIEMPO_ESTA	REGISTRO	VELOCIDA
ICE	D	BLE	TUS	ACTUAL	NTERIOR	DO_ACTUAL	_EVENTO	D_ACTUAL
1	5			Produccion			2023-12-25	
	1	1	1	do	Paro	19:45:41	19:45:45	18
2	4						2023-12-27	
	7	1	1	Paro	Arranque	19:04:30	19:04:32	20
3	5			Produccion			2024-01-02	
	0	0	1	do	Falla	15:46:47	15:46:49	20
4	4			Produccion			2024-01-01	
	8	0	1	do	Arranque	19:02:42	19:02:45	20

Source: Self-Made.

The second table, designated "Static Data," holds unchanging information about each machine. This data includes the machine number, IP address, and assigned operator (Table 14).

Table 14. Static Data.

INDI	I	PROYECT	NO_PA	FAMIL	LO	OPERACI	OPERAD	VELOCID	PZA_GO
CE	D	O	RTE	IA	TE	ÓN	OR	AD	LPE
1	5	HIROTEC	TIR_425I	A0024	112	60	JGAL	120	44

	4	FAURECI	TIR_258						
2	7	A	D	A0146	86	20	GTV	80	38
	5		TIR_346						
3	0	YKM J59W	A	B214	44	48	CCM	60	42
	4		TIR_542						
4	8	SUSR 35A	D	B0002	128	54	JESO	90	68

Source: Self-Made.

The third table, named "Indicators", is exclusively dedicated to tracking key performance parameters such as Overall Equipment Efficiency (OEE), Availability, Performance, and Quality. The fourth table, called "Times", focuses on collecting and storing various machine states provided by the company. Lastly, the fifth table specializes in monitoring the start and stop times of the machines.

The Python language is used to manage the efficient transmission of data from the sender to the database and its subsequent extraction for the website's interface. This bidirectional interaction between the data source, the database and the interface, enables flexibility in various operations such as inserting new data or updating existing information, responding to specific data storage needs.

Python code provides us with the ability to implement robust security protocols, which ensures safe transmission of data between the sender, database, and website. It safeguards the confidentiality and integrity of the information at all times. Additionally, this language enables the extraction of dynamic data hosted in the database and improves the response time of the web application (refer to Figure 14).

```
@route('/DATOS_DINAMICOS/<Numero:int>')
def DATOS_DINAMICOS(Numero):
    machine_data = {}
    global maquinasRegistradas
    try:
        # Conexión a la base de datos
        mydb = mysql.connector.connect(
            host="127.0.0.1",
            user="root",
            password="12345678",
            database="oeometer"
        )
        # Crear un cursor
        mycursor = mydb.cursor()

        if 1 <= Numero <= maquinasRegistradas:
            # Consulta SQL para obtener datos de la máquina especificada
            sql_query = """
            SELECT i.INDICE, i.ID, i.ENABLE, i.STATUS, i.ESTADO_ACTUAL, i.ESTADO_ANTERIOR, i.TIEMPO_DE_ESTADO_ACTUAL,
            DATE_FORMAT(i.REGISTRO_EVENTO, '%Y-%m-%d %H:%i:%s') AS REGISTRO_EVENTO,
            i.VELOCIDAD_ACTUAL, i.PIEZAS_PRODUCIDAS, i.PIEZAS_TOTALES, i.PIEZAS_ACEPTADAS, i.PIEZAS_RECHAZADAS, i.PIEZAS_RETRABAJO, i.IP_ADDRESS,
            t.DISPONIBILIDAD, t.RENDIMIENTO, t.CALIDAD, t.OEE, DATE_FORMAT(t.HORA, '%H:%i:%s %Y-%m-%d') AS HORA,
            t1.TIEMPO_TOTAL, t1.TIEMPO_PRODUCIENDO,
            t2.VELOCIDAD_STD

            FROM datos_dinamicos i
            INNER JOIN datos_indicadores t ON i.INDICE = t.INDICE
            INNER JOIN datos_tiempos t1 ON i.INDICE = t1.INDICE
            INNER JOIN datos_estaticos t2 ON i.INDICE = t2.INDICE
            WHERE i.INDICE = {Numero}
            ORDER BY DATE_FORMAT(t.HORA, '%Y-%m-%d %H:%i:%s') DESC
            LIMIT 1
            """
            # Ejecuta la consulta SQL
            mycursor.execute(sql_query)
```

Figure 14. Code for data extraction.

A form using HTML was designed to gather data from a website's user interface. When the form is submitted by the user, a route is triggered on our Python server (Bottle) via a POST request. This route leverages pymysql to establish a connection with the MySQL database and then proceeds to insert the collected data from the form into the scoreboard. Finally, a response is sent back to the website as confirmation of the transaction, as shown in Figure 15.

```
app = Bottle()

# Configuración de la conexión a la base de datos
conexion = pymysql.connect(host='127.0.0.1', user='root', password='12345678', database='oemeter')

# Ruta para recibir los datos y agregarlos a la tabla de indicadores
@app.post('/agregar_indicadores')
def agregar_indicadores():
    # Obtener los datos del formulario
    oee = request.forms.get('oee')
    disponibilidad = request.forms.get('disponibilidad')
    rendimiento = request.forms.get('rendimiento')
    calidad = request.forms.get('calidad')

    try:
        with conexion.cursor() as cursor:
            # Definir la consulta SQL para agregar indicadores a la tabla
            consulta_sql = "INSERT INTO indicadores (oee, disponibilidad, rendimiento, calidad) VALUES (%s, %s, %s, %s)"

            # Ejecutar la consulta con los datos proporcionados
            cursor.execute(consulta_sql, (oee, disponibilidad, rendimiento, calidad))

            # Confirmar la transacción
            conexion.commit()

            # Devolver una respuesta exitosa
            return template("Datos agregados correctamente a la tabla de indicadores.")
    except Exception as e:
        # En caso de error, imprimir el mensaje de error
        print(f"Error al agregar indicadores: {e}")
        # Devolver una respuesta de error
        return template("Error al procesar la solicitud. Consulta los registros para más detalles.")
```

Figure 15. Code to insert data into the DB.

Fourth Stage. Product/Process Validation

The manufacturing of this product is done through a process that allows for limited production of devices according to the client's specific needs, rather than mass production. The Sender and Receiver cards, along with the peripherals, are sourced from external suppliers and only require inspection based on the control plan. As a result, the product validation process is carried out during the Issuer's assembly. The Assembly Operations Diagram is displayed in Figure 16, while the final assembly is shown in Figure 14. Quality tests, connectivity tests, and reliability calculations for the system are performed.

*Production Run:* Once all the parts of the Transmitter were ready, which include the Cabinet, Embedded Board, and Components, a sequence of assembly was defined. The operations diagram is shown in Figure 16, and the final assembly is shown in Figure 17.

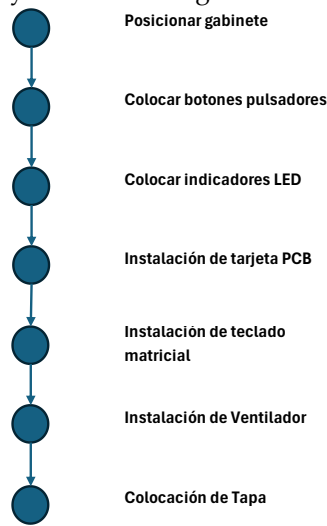


Figure 16. Assembly Operations Diagram. Source: Self-Made.



Figure 17. Assembled cabinet.

**Connectivity Tests:** The connectivity tests were carried out in the field. Firstly, the system was configured with the startup sequence and then connected to a Local Area Network via its IP address. Once connected, a tour was made through the different machines located at an average distance of 100 m. The test lasted for 30 minutes with the die-cutting machines in operation to detect any interference or vibration that could affect the system. During the test, the signal was received correctly on the receiver without any interference (see Figures 18 and 19).

```
/////Script started/////
Detecting devices...
Devices detected 2 in:
Port: COM4
Port: COM2
Searching compatible devices...
Error to connect port COM4: could not open port 'COM4': PermissionError(13, 'Acceso denegado.', None, 5)
Connecting port: COM2
Device information: OEE METER; Model:Terminal; Version:1.0
Device compatible.
Send and Received data by port COM2
Terminal: rtc?
RTC synchronizing: 04:45:14 04/12/2023
Synchronized
Terminal: net?
Terminal: net=1
Network connected: 1
TX: ping
Terminal: pong
TX: net:hola
Terminal: HTTP Response code: 200      Connection succesfull
TX: net:inf?
Terminal: HTTP Response code: 200      Connection succesfull
Terminal: HTTP Response code: 200      Station:inf=OEE METER, Model:Station, Version:1.0
```

Figure 18. Terminal connectivity test.



Figure 19. Field Connectivity Tests.

**System Reliability Calculation:** To measure the reliability of the system, tests were conducted during the week of January 15-19 of this year, with an average of 6.4 hours per day. Table 15 shows some test conditions, while Table 16 provides the testing period. The test results are summarized in Table 17, where a total of 37705 information packets were requested and 60 requests were rejected with error code 5, indicating that the response time exceeded the limit. The accumulated downtime is 20.4 seconds.

**Table 15.** Test conditions.

ID	Description	Amount	Unit
1	Waiting time between each request	2.5	s
2	Time spent for requests	94262.5	s
3	Delay time for each request	0.56	s
4	Average package size	200	Byte
5	Average transfer speed	2.86	KB/s

**Table 16.** Trial period.

ID	Event	Valor	Unit
1	Start test	15/01/2024	-
2	End of test	19/01/2024	-
3	Total test time	115215	s
4	Total time connected to WiFi network	115215	s
5	Disconnections	0	s
6	Total disconnected time	0	s

**Table 17.** Registered Event.

ID	Event	Valor
1	Requested HTTP packets	37705
2	Received on first attempt	37645
3	Received on second attempt	60
4	Received on third attempt	0

With this data we can calculate the MTBF and MTTR as follows

$$MTBF = \frac{115215 - 20.4}{60} = 1919.9$$

$$MTTR = \frac{20.4}{60} = 0.34$$

Applying Equation 6, we have:

$$C = e^{-\left(\frac{0.34}{1919.9}\right)}$$

$$C = e^{-0.00018} = 98.98\%$$

### 3. Results and Discussion

According to the Advanced Quality Planning methodology, we can highlight the following results:

**Product specification:** It is crucial to have the product specifications to determine the project's scale and guide its direction. Defining the customer's requirements in terms of specifications is important, as well as considering some other technical features that the product should have. Below are described the benefits that each of these characteristics provides:

*Programming Flexibility:* Open source software and editing platforms that do not require licensing, as well as access to any part of the code to make adjustments requested by the client.

*Wireless connection:* The system enables data transmission and access to information on-site through a local area network (LAN), without requiring an internet connection. It also features a reconnection code that ensures uninterrupted connectivity in case of any unforeseen events

*Storage capacity:* There are two options available for storing the information - either storing it on a local disk or hiring the services of a cloud provider, based on the customer's preference and after analyzing the costs involved

*Information Accessibility:* As mentioned earlier, we can use a Local Area Network internally and switch to a Global Network for external queries

*Diversity of Reports:* The application has been designed to enable users to access the indicators related to Overall Equipment Effectiveness (OEE), such as Availability, Performance, and Quality. Additionally, users can also view various events recorded in the database, including preventive and corrective maintenance times, preparation times, and more.

*Ergonomic design:* For this case, an appropriate size was chosen for the operators, however, a strategy is still needed for the arrangement of workstations.

*Users Management:* The system has a database of users and their privileges based on their role within the organization.

**Programming Algorithm:** Programming algorithms were developed to meet the functionality requirements specified in the product features and different communication protocols. Flowcharts were developed for the initialization of the Transmitter, for the calculation of Availability, Performance, and Quality. With this input information, the calculation logic is determined for the Overall Efficiency of the Equipment. Additionally, algorithms were developed for monitoring the power supply and wireless connection that allow for two important functions, the activation of electrical power through batteries and automatic reconnection to the network.

**Computer programs:** One of the important outcomes is the computer programs developed for the various functions of the system. As mentioned earlier, the main programs can be summarized as follows: Receiver Initialization, Transmitter Initialization, Communication Management Functions and Port Handling, Main Program, Communication Program between Receiver and PC, Programming of ATmega 328p Microcontroller, Programming of ESP32 Microcontroller, as well as the functions derived from them (**Appendix A**).

**Measuring system:** A productivity measurement system has been successfully developed based on the construction of prototypes, which allowed Process Diagrams to be elaborated, Failure Modes and Effects Analysis (FMEA) to be evaluated for both design and process, and Control Plans to be built (Figure 14). It is important to mention that after designing the Transmitter and Receiver cards, they were sent to an external supplier to be manufactured as an embedded system that is finally assembled with the other components

**Display Screens:** In this case, a main screen was developed for each machine, which allows navigating between the different machines that have been registered. Additionally, for each machine, it is possible to consult each of the indicators that make up the OEE by accessing the Availability, Performance, and Quality menus (Figure 20). In Figure 21, the screen showing the production target and the percentage of compliance is displayed, which serves as a visual aid for the operator.



Figure 20. Productivity indicators screen.



Figure 21. Production Target Screen.

During the reliability tests, only 60 delays were experienced among the 37705 test requests. These delays do not pose any risks for data loss since the information gets recorded in the database just 0.34 seconds after the request. In the future, the testing period will be extended to either confirm or adjust the reliability calculation

### 5. Conclusions

The Advanced Quality Planning methodology can be customized for the development of Productivity Indicator Measurement Systems using Industry 4.0 technologies, including the Industrial Internet of Things, Cloud Computing, Big Data, Systems Integration, and Cybersecurity. These technologies play a critical role in ensuring good product performance. The objectives achieved through this approach include:

- Tools such as questionnaires, interviews, literature reviews, and Quality Function Deployment methodology are used to determine specifications by understanding the Voice of the Customer.
- The components of the product are designed with a special emphasis on the Emissor Card, where all the components are integrated into an embedded system. Prototypes are created to construct Process Diagrams, Mode and Effect Analysis Models, and Control Plans.
- Manufacturing processes are designed to produce components that meet product specifications. Control plans are created to inspect critical product characteristics during assembly.

- Our database design includes information about machines, operators, products, and events. This data is necessary for calculating productivity indicators and presenting them on dashboards that can be accessed across all devices.

To sum up, the Productivity Indicator Monitoring System was designed and manufactured using free software. This software enables the digitization and real-time processing of production data to present them accurately. Based on the tests conducted so far, the system has an exceptional reliability rate of 99.98%.

It's important to note that we were limited by time constraints for this project. However, we plan to expand reliability testing over longer periods.

**Author Contributions:** Conceptualization, S.O. and A.L.; methodology, A.L.; software, G.L. and M.M.; validation, S.O. and A.L.; formal analysis, A.L.; investigation, S.O.; resources, G.L.; writing—original draft preparation, S.O.; writing—review and editing, N.D.; S.O.; M.M.; A.L.; supervision, N.D.; project administration, A.L.; funding acquisition, S.O. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by Tecnológico Nacional de México (TecNM), with registration number 18828 of the 2023 call (Scientific Research Projects, Technological Development, and Innovation).

**Data Availability Statement:** Annex A contains the main programs corresponding to the system's functionality. The database cannot be shared due to confidentiality.

**Acknowledgments:** I would like to extend a special thank you to Gian Carlo Trejo Valerio and Cesar Cruz Martínez, two students from the Electronic Engineering major at the Tecnológico Nacional de México (TecNM), Instituto Tecnológico de San Juan del Río campus, for their invaluable contributions and dedication to the project.

**Conflicts of Interest:** The authors have no conflicts of interest to declare.

## Appendix A

### Appendix A1. Setup source code

```
void setup () {
  //DUAL CORE
  //GPIO CONFIG
  //OUTPUTS
  //LEDS
  //GPIO INIT
  //LEDS
  //LCD INIT AND SETTINGS
  //COMMUNICATION INIT
  //CONECTING WITH PC SOFTWARE
  //RTC SYNCRONIZATION
  //WIFI INITIALIZATION
  //REQUEST INITIALIZATION
  //MACHINE 1
  //MACHINE 2
  //MACHINE 3
  //MACHINE 4
  //MACHINE 5
  //MACHINE 6
  //SERVER INITIALIZATION
  //SETUP FINISHED
}
```

### Appendix A2. Communication management functions and port management.

```
//GENERAL FUNCTIONS
void blink_led_green(int times, int wait) { }
void blink_led_blue(int times, int wait) { }
void blink_led_red(int times, int wait) { }
void rtc_upgrade () { }
//LCD FUNCTIONS
void lcd_write(int col, int row, String message, bool erase, bool clear) { }
//UART COMMUNICATION
void UART_received () { }
void wait_ok () { }
void send_ok (int times, int pause) { }
//WIFI NETWORK
void connect_wifi (bool starting) { }
void status_wifi () { }
//WIFI COMMUNICATION
void Wifi_received () { }
String httpGETRequest(const char* emisor) { }
```

### Appendix A3. Main Program

```
//CORE 1 TASKS
void loop () {
}
//CORE 0 TASKS
void loop2 (void * parameter) {
}
void loop3 (void * parameter) {
}
```

### Appendix A4. Communication program between the Receiver and the PC

```
#####LIBRARIES#####
import serial
import serial.tools.list_ports
import time
from datetime import datetime
#####GLOBAL VARIABLES#####
starting = True
system_on = True
uart_write = False
uart_TX = "null"
uart_RX = "null"
ready = False
network = 0
terminal_connected = False
terminal_port = any
uart = any
terminal_device = 0
device_connected = False
#####
#####FUNCTIONS TO READ AND WRITE DATA ON TXT FILES#####
def save_data (data_to_write, data):
def read_data(data_to_read):
def reset_data ():
```

```

#####FUNCTIONS TO DETECT AND CONNECT USB PORT#####
def scan_serial_ports():
def connect_to_serial_port(port_test):
def device_connection():
#####FUNCTIONS TO READ AND WRITE DATA IN USB PORT#####
def read_write():
def uart_received():
def close_por():
def read_number(number):
def read_boolean(boolean):
#####
reset_data()
print ("#####Script started#####")
#####/MAIN#####
while system_on:
reset_data()
print ("#####Script finished#####")

```

## Appendix A5. Sender Source Code

```

a)    ATMega 328P Microcontroller Programming
//COMMUNICATION
//KEYBOARD
//GPIO's
//BUZZER
//FAN
//OUTPUTS
//BUTTONS
//JOYSTICK
//POWER SUPPLY
void setup () {
//GPIO CONFIG
//OUTPUTS
//BUZZER
//COOLER
//OUTS
//INPUTS
//JOYSTICK
//GPIO INIT
//BUZZER
//COOLER
//OUTS
//COMMUNICATION INIT
//POWER SUPPLY
}
void loop () {}
//GET NEW STATE DEVICES
char get_button () {}
char get_joystick () {}
char get_key () {}

//READ VOLTAGES FROM BATTERY AND POWER SUPPLY
float get_battery () {}

```

```

float get_psstate () { }
//POWER SUPPLY MONITORING
void read_PS () { }
//SET DEVICES LEVELS
void buzzer_beep (int wait) { }
void fan_velocity (int percent) { }
//COMMUNICATION
void PC_Receive () { }

```

b) *ESP32 Microcontroller Programming*

```

//MEMORY
//DUAL CORE
//WIFI NETWORK
//NETWORK
//IP CONFIGURATION
//WEBSERVER
//WIFI COMMUNICATION
//IP
//DATA TO GET
//FUNCTION FROM REQUEST
//LCD DISPLAY
//VARIABLES
//REAL TIME CLOCK RTC
//CURRENT TIME VARIABLES
//COMMUNICATION
//ATMEGA
//GPIO's
//LEDS
//SENSORS
//POWER SUPPLY
//COOLER VELOCITY
//KEYBOARD
//JOYSTICK
//BUTTONS
//GENERAL VARIABLES OF SYSTEM
//OEE VARIABLES
//GENERAL DATA
//DISPONIBILITY
// WORK SCHEDULES
//EFFICIENCY
//PRODUCTION COUNTERS
//QUALITY
//ALARMS
//STATE TIMES
//BOOLEANS
//SENSORS
//SIMULATION

void setup () {
//DUAL CORE
//GPIO CONFIG

```

```

//OUTPUTS
//LEDS
//INPUTS
//SENSORS
//GPIO INIT
//LEDS
//LCD INIT AND CONFIG
//COMMUNICATION INIT
//REQUEST INITIALIZATION
//INTERRUPT INIT
//MEMORY
}
//CORE 1 TASK
void loop () {
  switch(display_n) {
    case 1://DISPLAY STARTING
    case 2://SELECT MODE
    case 3://RTC CONFIGURATION DATE
    case 4://RTC CONFIGURATION TIME
    case 5://DATA CONFIGURATION MACHINE
    case 6://DATA CONFIGURATION USER
    case 7://DATA CONFIGURATION PIECE
    case 8://ENTER WORK SHIFTS
    case 9://CONFIGURATION OF WORK SCHEDULES
    case 10://SAVING DATA
    case 11://WIFI CONNECTION
    case 12://TERMINAL CONNECTION
    case 13://TERMINAL SYNCHRONIZATION
    case 100://PRINCIPAL DISPLAY
    case 200://DISPONIBILITY DISPLAY
    case 300://EFFICIENCY DISPLAY
    case 400://QUALITY DISPLAY
    case 255://SCREEN TEST
  }
}
//CORE 0 TASK
void loop2 (void * parameter) { }
void loop3 (void * parameter) { }
void loop4 (void * parameter) { }
//GENERAL FUNCTIONS
void led_power (int times, int wait, bool state) { }
void power_supply () { }
void battery_level () { }
void buzzer () { }
void cooler_velocity (int percent) { }
void save_data () { }
void read_data () { }
void simulation () { }
//REAL TIME CLOCK
void rtc_upgrade () { }
void rtc_config (String data) { }
unsigned int HHMMSS_SS (char hours, char minutes, char seconds) { }
void SS_HHMMSS (unsigned int seconds) { }

```

```

    void time_elapsed (char H_1, char M_1, char S_1, char H_2, char M_2, char S_2) { }
//LCD FUNCTIONS
    void lcd_write (int col, int row, String message, bool erase) { }
    String print_float (float number) { }
    String print_int (unsigned int number, int length, bool zero) { }
//PHISIC INPUTS
    bool QUA_PORT () { }
//COMMUNICATION FUNCTIONS
    void ATmega_Received () { }
    void PC_received () { }
    void ATmega_Actions (int option, bool enable) { }
    void joystick_actions (int value) { }
//WIFI NETWORK
    void terminal_connect () { }
    void terminal_rtc () { }
    void wifi_connect () { }
    void wifi_reconnect () { }
    void wifi_status () { }
    void wifi_received () { }
    void wifi_send (String value) { }
    String httpGETRequest(const char* serverName) { }

```

## References

1. INEGI, «Directorio Estadístico Nacional de Unidades Económicas,» Instituto Nacional de Estadística y Geografía, México, 2021.
2. O. Mörtz, C. Emmanouilidis, N. Hafner y M. Schadler, «Cyber-physical systems for performance monitoring in production intralogistics,» *Computers & Industrial Engineering*, vol. 142, pp. 1-10, 2020.
3. M. A. Mejía Huidobro, A. D. Camacho Vera y M. Marcelino Aranda, «Estrategias del sector público y privado para la implementación de la Industria 4.0 en México,» *RUII*, vol. 6, nº 1, pp. 13-31, 2020.
4. INEGI, «Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en los Hogares (ENDUTIH) 2022,» INEGI, México, 2022.
5. INEGI, «ESTADÍSTICAS A PROPÓSITO DEL DÍA MUNDIAL DEL INTERNET,» INEGI, México, 2023.
6. xtb, «Computación en la Nube,» 23 december 2023. [En línea]. Available: <https://www.xtb.com/lat/analisis-y-noticias/analisis-demercado/computacion-en-la-nube-el-futuro-de-la-digitalizacion>.
7. K. Shafique, B. A. Khawaja, F. Sabir, S. Qazi y M. Mustaqim, «Internet of Things (IoT) for Next-Generation Smart Systems: A Review of Current Challenges, Future Trends and Prospects for Emerging 5G-IoT Scenarios,» *IEEE Access*, vol. 8, pp. 23022-23040, 2020.
8. J. M. Rodríguez Sierra, «Nuevo sistema de Gestión de Eficiencia Global (OEE) en tiempo real para Industria,» Universidad Politécnica de Valencia, Valencia, España, 2019.
9. K. Valaskova, M. Nagy, S. Zabochnik y G. Lăzăroiu, «Industry 4.0 Wireless Networks and Cyber-Physical Smart Manufacturing Systems as Accelerators of Value-Added Growth in Slovak Exports,» *Mathematics*, vol. 10, pp. 1-21, 2022.
10. T. Ahmed, C. Lekha Karmaker, S. Benta Nasir, A. Moktadir y S. Kumar Paul, «Modeling the artificial intelligence-based imperatives of industry 5.0 towards resilient supply chains: A post-COVID-19 pandemic perspective,» *Computers & Industrial Engineering*, vol. 177, pp. 1-12, 2023.
11. J. Contreras, «Estos son los Clústers Empresariales de Querétaro,» 24 february 2021. [En línea]. Available: <https://www.liderempresarial.com/estos-son-los-clusters-empresariales-de-queretaro/>. [Último acceso: 08 december 2023].
12. IMCO, «Resultados Generales del Índice de Competitividad Estatal 2023,» IMCO, Cd. México, 2023.
13. S. F. Fam, N. Isamil, H. Yanto, D. D. Prastyo y B. P. Lau, «LEAN MANUFACTURING,» *Journal of Advanced Manufacturing Technology*, vol. Special Issue, 2018.
14. C. A. Díaz-Contreras, D. A. Catari-Vargas, C. D. Murga-Villanueva, G. A. Díaz-Vidal y V. F. Uezada-Lara, «EFECTIVIDAD GENERAL DE EQUIPOS (OEE) AJUSTADO POR COSTOS,» *Interciencia*, pp. 158-163, 2020.
15. S. NAKAJIMA, *Introduction to TPM Total Productive Maintenance*, Eleventh ed., Minesota: Productivity Pr, 1988, p. 129.

16. J. M. Tamborero del Pino, «NTP 331. Fiabilidad: la distribución de Weibull,» Ministerio del Trabajo y Asuntos Sociales, Madrid, 1994.
17. Chrysler LLC, Ford & General Motors, «Planeación Avanzada para la Calidad de Productos y Planes de Control,» AIAG, NY, 2008.
18. J. I. Legorreta Rodríguez, «Aplicación de Lean – Six Sigma para aumentar la eficiencia global del equipo de impresión rotativa offset,» Instituto Tecnológico de San Juan del Río, San Juan del Río, 2021.
19. A. GONZÁLEZ-TORRES, A. RAMÍREZ-CASTAÑEDA, E. POBLANO-OJINAGA y F. MENDOZA-MONTERO, «Implementación del OEE como herramienta de mejora continua aplicada a una línea de producción,» Revista de Docencia e Investigación Educativa, vol. 2, nº 6, pp. 1-7, 2016.
20. F. E. Soto Ramírez, «Medición Automatizada de la Eficiencia por medio de Sistemas de Pesaje,» CIATEQ, Pachuca, Hidalgo, 2014.
21. IMPI, «Reporte de Búsqueda de Estado de la Técnica,» IMPI, Cd. México, 2023.
22. S. Taj, A. S. Imran, Z. Kastrati, S. M. Daudpota, R. A. Memon y J. Ahmed, «IoT-based supply chain management: A systematic literature review.,» Internet of Things, vol. 24, pp. 1-25, 2023.
23. INEGI, «Censos Económicos,» Instituto Nacional de Estadística y Geografía, México, 2019.
24. JIMP, «Total Productive Maintenance Training Textbook,» JIMP.
25. S. Borris, «RCM-Reliability Centered Maintenance,» de Total Productive Maintenance, NY, McGraw-Hill, 2006, p. 414.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.