

**UNIVERSIDAD NACIONAL DE CAAGUAZÚ
FACULTAD DE CIENCIAS Y TECNOLOGÍAS
CARRERA DE INGENIERÍA EN
ELECTRÓNICA**



PROYECTO FINAL DE GRADO

**Diseño de un sistema SCADA con
instrumentación electrónica integrada para la
supervisión de la máquina blistera de la planta
farmacéutica QUIMFA S.A.**

Autor:

Pastor Miguel Galeano Aquino

Tutor: Ing. Federico Javier Cabrera Sanabria

CORONEL OVIEDO, DICIEMBRE DE 2025



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.

VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.



Usted es libre de:

- **Compartir** — copiar y redistribuir el material en cualquier medio o formato
- **Adaptar** — remezclar, transformar y construir a partir del material

Bajo los siguientes términos:

- **Atribución** — Usted debe dar [crédito de manera adecuada](#), brindar un enlace a la licencia, e [indicar si se han realizado cambios](#). Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante.
- **No Comercial** — Usted no puede hacer uso del material con [propósitos comerciales](#).



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.
VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

DERECHO DE AUTOR

Quienes suscriben, **Pastor Miguel Galeano Aquino**, autor del trabajo de investigación titulado **“Diseño de un sistema SCADA con instrumentación electrónica integrada para la supervisión de la máquina blistera de la planta farmacéutica QUIMFA S.A.”**, declaran que voluntariamente ceden a título gratuito en forma pura y simple ilimitada e irrevocablemente a favor de la Facultad de Ciencias y Tecnologías – UNCA, el derecho de autor de contenido patrimonial, que le corresponde sobre el trabajo de referencia. Conforme a lo anteriormente expresado, esta sesión le otorga a la FCyT la Facultad de comunicar la obra divulgarla, publicarla y reproducirla en soportes analógicos o digitales en la oportunidad que así lo estime conveniente. La FCyT deberá indicar qué autoría o creación del trabajo corresponde a mi persona y hará referencia al autor y a las personas que hayan colaborado en la realización del presente trabajo de investigación.

En la ciudad de Coronel Oviedo a los, del mes de del 2025

.....

Firma



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.

VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

PÁGINA DE APROBACIÓN

Trabajo de fin de grado para la obtención del Título de Ingeniero Electrónico, aprobado en representación de la Facultad Ciencias y Tecnología de la Universidad Nacional de Caaguazú, por el Tribunal Examinador constituido por los siguientes profesores y con la siguiente nota final:

CALIFICACIÓN FINAL: _____

ACTA N. °: _____

FECHA: _____

Prof. Ing.

Prof. Ing.

Prof. Ing.



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.
VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

Dedicatoria

A Dios, por estar siempre presente, guiando mis pasos y dándome la fortaleza necesaria para llegar hasta aquí.

Dedico este trabajo a mi madre Gladys Aquino, quien ha sido mi apoyo incondicional, mi aliento constante y el motor de mi perseverancia en cada etapa de este proceso. Su amor y dedicación son la razón por la que hoy puedo celebrar este logro.

A mis abuelos y familiares, quienes siempre se interesaron por mi bienestar, preguntándome con cariño sobre mis avances y alentándome a seguir adelante.

A mi grupo de estudios "Yo y los Cabros", que ha sido mi apoyo en los momentos más difíciles de la carrera, compartiendo risas, consejos y experiencias que hicieron que los días de estudio y trabajo fueran más llevaderos.

A los amigos que conocí a lo largo de esta etapa, tanto en la facultad como fuera de ella, por su aliento y compañía, que siempre me motivaron a seguir con mis sueños.

Y, a aquellos pequeños detalles cotidianos que me dieron energía en los momentos de cansancio, como el café que me mantuvo despierto durante largas jornadas de estudio, y esos breves momentos de relajación con amigos y familia que me recordaron la importancia de equilibrar esfuerzo y descanso.



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.

VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

Agradecimientos

Quiero expresar mi agradecimiento a mi tutor el Ing. Ing. Federico Javier Cabrera Sanabria, por su constante orientación, paciencia y dedicación en la elaboración de esta tesis. Su valiosa guía académica ha sido fundamental para llevar a cabo este trabajo de manera rigurosa y eficiente.

Agradezco profundamente al jefe de mantenimiento de Quimfa S.A. el Sr. Antonio Acuña, quien me proporcionó información crucial y orientación práctica para el desarrollo de mi proyecto final de grado. Su apoyo y experiencia fueron esenciales para comprender y aplicar los conocimientos adquiridos.

Mi agradecimiento especial a los profesores que, con su excelente enseñanza y compromiso, se tomaron el tiempo y esfuerzo de formar bien a sus estudiantes, y cuya dedicación me permitió crecer como profesional.

Finalmente, agradezco a mis compañeros de carrera, quienes de manera intelectual me brindaron su apoyo durante el proceso de elaboración de esta tesis. Sus aportes, ideas y sugerencias han sido enriquecedores y fundamentales para la mejora de este trabajo.



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.
VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

Resumen

El presente Proyecto Final de Grado tuvo como objetivo diseñar un sistema de supervisión remota basado en SCADA e integrado a una base de datos SQL para la máquina blistera Romaco de la planta farmacéutica QUIMFA S.A., con el fin de centralizar la información operativa hoy dispersa y reducir las “islas de automatización” existentes. Para ello se desarrolló un prototipo piloto completamente simulado en la plataforma Ignition, donde se implementó un gemelo digital simplificado de la blistera mediante scripts en Python, tags de memoria y lógica de eventos que replican la producción, los rechazos y los paros de máquina en distintos escenarios de operación. Sobre esta infraestructura se diseñó una base de datos relacional en PostgreSQL para el registro histórico de estados, contadores y eventos, y se implementó un algoritmo de cálculo de OEE (disponibilidad, rendimiento y calidad) que procesa automáticamente los datos recopilados para generar indicadores de desempeño en tiempo real. La interfaz se desarrolló en el módulo Perspective de Ignition, permitiendo la visualización web responsiva de los parámetros críticos desde navegadores de escritorio y dispositivos móviles. Para garantizar un acceso remoto seguro sin exponer puertos a Internet pública se configuró una red privada virtual mediante Tailscale, que habilita la supervisión cifrada del sistema desde ubicaciones externas. Se añadió un enclavamiento de arranque no intrusivo validado con OpenPLC/Modbus. Los resultados de la validación demostraron la coherencia del flujo de datos, la consistencia del cálculo de OEE y la estabilidad de la visualización local y remota, sentando las bases para una futura migración del prototipo a la máquina física de planta. El trabajo se enmarca en la línea de Control y Automatización Industrial y contribuye al cumplimiento de los ODS 9 (Industria, innovación e infraestructura) y 12 (Producción y consumo responsables).

Palabras clave: SCADA, OEE, Ignition, supervisión remota, base de datos SQL, Modbus TCP/IP, Control y Automatización Industrial, ODS 9, ODS 12.



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.
VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

Abstract

This Final Year Project aimed to design a remote supervision system based on SCADA and integrated with an SQL database for the Romaco blister machine at the pharmaceutical plant QUIMFA S.A., in order to centralize operational information that is currently dispersed and reduce existing “automation islands.” To achieve this, a fully simulated pilot prototype was developed on the Ignition platform, where a simplified digital twin of the blister machine was implemented using Python scripts, memory tags, and event logic that replicate production, rejects, and machine stops under different operating scenarios. On top of this infrastructure, a relational database in PostgreSQL was designed for the historical logging of states, counters, and events, and an OEE (availability, performance, and quality) calculation algorithm was implemented to automatically process the collected data and generate real-time performance indicators. The interface was developed in Ignition’s Perspective module, enabling responsive web visualization of critical parameters from desktop browsers and mobile devices. To ensure secure remote access without exposing ports to the public Internet, a virtual private network was configured using Tailscale, which enables encrypted system supervision from external locations. A non-intrusive start interlock validated with OpenPLC/Modbus was added. The validation results demonstrated the coherence of the data flow, the consistency of the OEE calculation, and the stability of local and remote visualization, laying the groundwork for a future migration of the prototype to the physical machine on the plant floor. The project falls under the Industrial Control and Automation research line and contributes to achieving UN SDGs 9 (Industry, Innovation and Infrastructure) and 12 (Responsible Consumption and Production).

Keywords: SCADA, OEE, Ignition, remote supervision, SQL database, Modbus TCP/IP, Control and Industrial Automation, SDG 9, SDG 12.



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.

VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

ÍNDICE

INTRODUCCIÓN	1
1. METODOLOGÍA	3
1.1. DISEÑO DEL ESTUDIO.....	3
1.2. RECOLECCIÓN DE DATOS.....	3
1.3. CONTEXTO DEL PROYECTO	4
1.4. DEFINICIÓN DEL PROBLEMA.....	4
1.5. FUNCIONAMIENTO DE LA MÁQUINA DE BLÍSTERES.....	5
1.6. IDENTIFICACIÓN DE PARÁMETROS DE INTERÉS DEL PROYECTO	5
1.7. MÓDULO DE INSTRUMENTACIÓN ELECTRÓNICA PARA VARIABLES CRÍTICAS	8
1.7.1. Selección de variables críticas.....	8
1.7.2. Requisitos funcionales del enclavamiento.....	9
1.7.3. Arquitectura de hardware propuesta	9
1.7.4. Lógica de enclavamiento.....	10
1.7.5. Integración con SCADA y base de datos	11
1.8. CÁLCULO DEL OEE Y PARÁMETROS DE EFICIENCIA	12
1.9. DISEÑO Y CONFIGURACIÓN DE LA BASE DE DATOS SQL.....	13
1.10. EVALUACIÓN Y SELECCIÓN DE LA PLATAFORMA TECNOLÓGICA.....	14
1.11. DESARROLLO DEL SCADA EN IGNITION	15
1.11.1. Módulos de Ignition utilizados en el proyecto.....	16
1.11.1.1. OPC-UA Server	16
1.11.1.2. Perspective Module	16
1.11.1.3. Reporting Module.....	16
1.12. PROGRAMACIÓN DE LA LÓGICA DE SIMULACIÓN.....	16
1.12.1. Definición de Variables.....	16
1.12.2. Desarrollo del simulador de proceso	18
1.12.3. Implementación del algoritmo de cálculo de OEE.....	19
1.13. DESARROLLO DE LA INTERFAZ GRÁFICA	20
1.14. CONFIGURACIÓN DE LA RED VPN Y ACCESO REMOTO	21
1.15. REPORTES.....	22
1.16. VALIDACIÓN DEL SISTEMA IMPLEMENTADO.....	22
1.16.1. Validación de la lógica de simulación	22
1.16.2. Verificación de integridad de datos en la base de datos SQL.....	22
1.16.3. Certificación del algoritmo de cálculo de OEE.....	23
1.16.4. Pruebas de experiencia de usuario en Perspective y acceso remoto mediante Tailscale.....	23
2. RESULTADOS Y ANÁLISIS	25
2.1. RESULTADOS DE LA IMPLEMENTACIÓN TÉCNICA	25
2.2. RESULTADOS OPERATIVOS DE LA SIMULACIÓN	25
2.3. ANÁLISIS ECONÓMICO Y PRESUPUESTO	25
2.4. CONSIDERACIONES FINALES DEL DISEÑO	26
3. CONCLUSIONES Y RECOMENDACIONES	27
3.1. CONCLUSIONES	27
3.2. RECOMENDACIONES	28



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.
VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

4. BIBLIOGRAFÍA	29
5. ANEXOS.....	1
ANEXO A. ENTORNO REAL EN LA PLANTA.....	1
ANEXO B. ARQUITECTURA LÓGICA DE NAVEGACIÓN Y CONTROL DE ACCESO BASADO EN ROLES	1
ANEXO C. CÓDIGO FUENTE DEL SIMULADOR DE PROCESO SIMROMACO	2
ANEXO D. LÓGICA DE SIMULACIÓN DE LA BLISTERA.....	6
ANEXO E. CÓDIGO FUENTE DE CÁLCULO DE OEE	6
ANEXO F. LÓGICA DEL CÁLCULO DE OEE	10
ANEXO G. SCRIPT DE CREACIÓN DE BASE DE DATOS SQL	11
ANEXO H. 1.ARQUITECTURA GENERAL DEL SISTEMA	18
ANEXO I. HMI DEL OPERADOR.....	18
ANEXO J. VISTA DE SUPERVISIÓN SCADA DE LA BLISTERA	19
ANEXO K. VISTA GENERAL ESQUEMÁTICA DE LA BLISTERA ROMACO Y UBICACIÓN DE COMPONENTES PRINCIPALES.....	19
ANEXO L. DESGLOSE VISUAL DEL OEE (DISPONIBILIDAD, RENDIMIENTO Y CALIDAD) Y TIEMPOS ASOCIADOS DE OPERACIÓN Y PÉRDIDA	20
ANEXO M. CONEXIONES DE AGUA DE REFRIGERACIÓN.....	20
ANEXO N. CONEXIONES DE AIRE COMPRIMIDO.....	21
ANEXO O. DIAGRAMA DE FLUJO DEL ENCLAVAMIENTO DE ARRANQUE (START PERMISSIVE) POR PRESIÓN DE AIRE Y TEMPERATURA DE AGUA.....	21
ANEXO P. ARQUITECTURA DE HARDWARE PROPUESTA DEL ENCLAVAMIENTO DE ARRANQUE E INTEGRACIÓN CON SCADA/SQL.....	22
ANEXO Q. ESQUEMA DE CONEXIONADO DEL MÓDULO DE INSTRUMENTACIÓN ELECTRÓNICA.....	22
ANEXO R. LISTA DE COMPONENTES Y CARACTERÍSTICAS	23
ANEXO R.1. CPU LOGO! 8.4 24RCE	23
ANEXO R.2. MÓDULO DE EXPANSIÓN LOGO! AM2.....	23
ANEXO R.3. SENSOR DE PRESIÓN WIKA A-10.....	24
ANEXO R.4. SENSOR RTD PT100 RS PRO DE 4 HILOS.....	24
ANEXO R.5. FUENTE DE ALIMENTACIÓN 24VDC.....	25
ANEXO R.6. RELE TRS 24VDC 1CO	25



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.
VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

ÍNDICE DE FIGURAS

FIGURA 1.1 DIAGRAM DE CAUSA-EFECTO	4
FIGURA 1.2. CONCEPTO GENERAL DE MONITOREO	8
FIGURA 1.3. MODELO ERD QUE REPRESENTA LA ORGANIZACIÓN DE LA BASE DE DATOS	14
FIGURA 5.1. BLISTERA EN PLANTA EN AREA LIMPIA	1
FIGURA 5.2.. ESQUEMA DEL FUNCIONAMIENTO DE LAS VENTANAS DEL SISTEMA	1
FIGURA 5.3.. ESQUEMA LOGICO DE LA SIMULACION DE LA ROMACO	6
FIGURA 5.4. ESQUEMA LOGICO DEL CALCULO DE OEE	10
FIGURA 5.5.. REPRESENTACIÓN ESQUEMÁTICA DE LOS COMPONENTES DEL SISTEMA	18
FIGURA 5.6. VENTANA DE JUSTIFICACIÓN DE PARO	18
FIGURA 5.7.. INTERFAZ PRINCIPAL DEL ADMINISRADOR DEL SISTEMA SCADA	19
FIGURA 5.8. VISTA GENERAL ESQUEMÁTICA DE LA MÁQUINA	19
FIGURA 5.9.. DESGLOSE VISUAL DE LOS TRES FACTORES DETERMINANTES DE OEE	20
FIGURA 5.10.. PUNTOS DE CONEXIÓN DEL AGUA DE REFRIGERACIÓN PARA INSTRUMENTACIÓN DEL SERVICIO.	20
FIGURA 5.11. PUNTOS DE CONEXIÓN DE AIRE COMPRIMIDO	21
FIGURA 5.12. DIAGRAMA DE FLUJO DEL ENCLAVAMIENTO DE ARRANQUE	21
FIGURA 5.13. ARQUITECTURA DEL ENCLAVAMIENTO DE ARRANQUE	22
FIGURA 5.14. ESQUEMA DE CONEXIONADO: SENSORES 4–20 MA HACIA MÓDULO LOGO! AM2 Y SALIDA POR RELÉ	22



MISIÓN: Formar profesionales excelentes con conocimientos científicos y tecnológicos, competentes, con sentidos crítico, ético y responsabilidad Social.

VISIÓN: Ser una Facultad líder, con excelencia en la formación de profesionales que contribuya al desarrollo del País.

ÍNDICE DE TABLAS

TABLA 1.1. DETERMINACION DE SEÑALES DE INTERES	7
TABLA 1.2. COMPONENTES MÍNIMOS DEL MODULO PROPUESTO.....	10
TABLA 1.3. MAPA DE VARIABLES PARA INTEGRACIÓN MODBUS TCP	12
TABLA 1.4. MATRIZ DE DECISIÓN PONDERADA	15
TABLA 1.5. TABLA DE CONTADORES Y PRODUCCIÓN	17
TABLA 1.6. TABLA DE ESTADO DE MÁQUINA Y FORMATO	17
TABLA 1.7. TABLA DE SISTEMA DE MENSAJES	17
TABLA 1.8. TABLA DE VARIABLES OEE	17
TABLA 1.9. TABLA DE VARIABLES DE TIEMPO (PARA CÁLCULOS OEE)	18
TABLA 1.10. TABLA DE SISTEMAS DE PAROS	18
TABLA 1.11. TABLA DE PARÁMETROS AUXILIARES	18
TABLA 1.12. PRESUPUESTO CONSOLIDADO DE LA INVERSIÓN INICIAL (PROTOTIPO SIMULADO; HARDWARE DE ENCLAVAMIENTO POR COTIZAR)	26

Introducción

La industria farmacéutica es un sector clave en la economía global, especialmente en tiempos de crisis sanitaria y constante innovación. A medida que las demandas de calidad y eficiencia aumentan, las empresas del sector deben integrar tecnologías avanzadas para mantenerse competitivas en el mercado internacional. En este sentido, la automatización industrial y la adopción de tecnologías de Industria 4.0 se han convertido en factores decisivos para optimizar los procesos de producción y mejorar la trazabilidad y calidad de los productos. De acuerdo con [1] y [2] las empresas que tardan demasiado en migrar a estructuras conectadas corren el riesgo de perder competitividad, por lo que es imperativo comenzar a recolectar y analizar información lo más inmediato posible. La transición hacia fábricas inteligentes, en las cuales los sistemas y máquinas están interconectados para compartir datos en tiempo real, es crucial para mantener la eficiencia y adaptabilidad en un entorno cada vez más exigente.

En QUIMFA S.A., específicamente en el área de blísteras, se utiliza maquinaria moderna con protocolo OPC-UA. Sin embargo, estas máquinas operan de manera aislada, lo que genera lo que se denomina como "islas de automatización" [3]. A pesar de contar con máquinas potentes y con alta capacidad de producción, el sistema actual presenta varias falencias operativas que afectan directamente la eficiencia y calidad del proceso productivo. Según relevamientos realizados con la Jefatura de Ingeniería de la planta, se ha constatado que la información generada por las máquinas no se centraliza ni se utiliza de manera eficaz, ya que los datos se registran manualmente en papel o a través de hojas de Excel, lo que aumenta el riesgo de errores humanos y dificulta la trazabilidad de la información. Además, la falta de un sistema automatizado para el cálculo en tiempo real del OEE (Overall Equipment Efficiency) definido por Nakajima [4] impide una visión clara del rendimiento de la máquina. Esta carencia también afecta la respuesta ante fallos o paradas no programadas, ya que no se cuenta con un sistema que permita detectar y analizar los problemas de manera proactiva. Se ha identificado que la planta también enfrenta dificultades para acceder a las zonas limpias de Grado B de manera inmediata para brindar soporte técnico, lo que agrava aún más la situación.

Con el fin de abordar estas limitaciones, el presente proyecto propone la implementación de un sistema SCADA basado en Ignition, que permitirá centralizar la información de todas las máquinas en una plataforma común. Este sistema proporcionará monitoreo en tiempo real de las blísteras, permitiendo la visualización continua de los parámetros operativos, como tiempos de funcionamiento, tiempos de paro y eficiencia de la máquina. Además, se implementará un sistema

de bases de datos SQL para almacenar y gestionar toda la información generada por la máquina, lo que facilitará el análisis y la toma de decisiones. Este enfoque permitirá no solo optimizar el uso de recursos, sino también mejorar la eficiencia operativa y reducir las mermas en el proceso de producción.

Existen tecnologías habilitadoras para la interconexión de máquinas y sistemas, como el OPC-UA, que es ya un estándar industrial para la comunicación de datos en tiempo real [5]. Sin embargo, QUIMFA S.A. aún no ha implementado completamente estas capacidades, lo que limita el potencial de sus equipos. La integración de plataformas SCADA como Ignition es una de las soluciones más efectivas y ampliamente adoptadas para superar este tipo de limitaciones. Estas plataformas permiten integrar diversas fuentes de datos, automatizar el control y monitoreo de procesos y proporcionar visualización en tiempo real, lo que contribuye significativamente a la mejora de la productividad y eficiencia en la planta.

El desarrollo del proyecto consistió en la configuración de un sistema SCADA para la adquisición de datos en tiempo real de la máquina blíster mediante el protocolo OPC-UA, centralizando la información operativa en una base de datos SQL para asegurar su trazabilidad histórica. Sobre esta infraestructura, se implementaron algoritmos para el cálculo automático del OEE y se diseñó una interfaz gráfica (HMI) accesible desde plataformas de escritorio y móviles, lo que habilitó la supervisión remota. Adicionalmente, se definió la arquitectura de un enclavamiento de arranque basado en PLC (Siemens LOGO! 8.4) y señales industriales (4–20 mA), y su validación funcional se realizó mediante un SoftPLC (OpenPLC) comunicándose localmente con el Gateway de Ignition a través de Modbus TCP/IP, manteniendo la lógica prevista para la implementación física. Finalmente, el sistema fue validado mediante pruebas para garantizar la precisión de los indicadores y la optimización de los procesos productivos en la planta.

En este marco, el trabajo se encuadra en la línea de investigación de Control y Automatización Industrial, dado que integra protocolos de comunicación y sistemas de monitoreo propios de la digitalización industrial. Asimismo, el proyecto se alinea con la Agenda 2030 contribuyendo al ODS 9 (Industria, Innovación e Infraestructura) mediante la modernización tecnológica de la planta, y al ODS 12 (Producción y Consumo Responsables), al utilizar el cálculo de eficiencia (OEE) para minimizar mermas y promover el uso sostenible de los recursos productivos.

1. Metodología

1.1. Diseño del estudio

Este proyecto se sitúa en un enfoque técnico y aplicado, orientado a optimizar la eficiencia operativa en la planta QUIMFA S.A., particularmente en el sector de blísteras. El propósito fundamental es crear un sistema SCADA de supervisión remota para la máquina blisteadora Romaco, empleando una base de datos SQL para calcular parámetros esenciales de eficiencia (como el OEE). El sistema facilita la supervisión en tiempo real de la máquina y la toma de decisiones fundamentadas en datos, optimizando de esta manera la gestión operativa de la planta.

La estructura del estudio utiliza una metodología incremental y piloto, que permite evaluar la efectividad de la solución antes de una implementación más amplia en otras áreas de producción. Así, el proyecto se organiza en fases de desarrollo que abarcan la recopilación de datos, la integración de sistemas, la creación de la interfaz gráfica (HMI) y la validación. Este método asegura que cada fase del sistema se evalúe en condiciones reales de operación, facilitando ajustes y una optimización constante antes de su ampliación.

La investigación se fundamenta asimismo en la utilización de tecnologías establecidas en el sector, como OPC-UA para la integración de maquinaria y SCADA para la visualización y análisis de información. Estas tecnologías son esenciales para optimizar la interoperabilidad de los dispositivos y ofrecer una solución escalable que se adapte a las necesidades operativas futuras de la instalación.

1.2. Recolección de datos

Se realizó una entrevista semiestructurada con el gerente de ingeniería y el jefe de mantenimiento de la planta para recolectar datos, con el objetivo de obtener información relevante sobre el estado actual del sistema de monitoreo y las limitaciones operativas de la planta. La entrevista se desarrolló utilizando una guía básica de preguntas, lo que permitió explorar aspectos específicos según avanzaba la conversación. Asimismo, se llevaron a cabo observaciones directas no estructuradas en la zona de blísteras, lo que permitió identificar de manera práctica las falencias en el manejo de datos operativos y los desafíos asociados al seguimiento de la máquina. Finalmente, se consultó el manual de usuario de la máquina y se revisaron diversas fuentes secundarias como libros especializados, estudios científicos y páginas web técnicas para fundamentar teóricamente los contenidos necesarios para el desarrollo del estudio.

1.3. Contexto del proyecto

QUIMFA S.A. es una empresa con más de 65 años de presencia en el mercado paraguayo, dedicada al desarrollo, producción y comercialización de productos farmacéuticos de alta calidad. La compañía cuenta con certificaciones internacionales y presencia en Bolivia, Uruguay y Perú, destacándose por su compromiso con la innovación y la calidad de sus productos.

En la planta ubicada en Asunción, QUIMFA S.A. utiliza maquinaria avanzada como la blíster Romaco, con una capacidad de 500 blísters por minuto. Sin embargo, la planta enfrenta desafíos operativos debido a la falta de un sistema de monitoreo remoto y la dependencia de registros manuales, lo que limita la eficiencia y la toma de decisiones en tiempo real.

El problema principal radica en la falta de integración de datos y en la ausencia de un cálculo en tiempo real del OEE, lo que dificulta la optimización de los procesos productivos. Este proyecto propone la implementación de un sistema SCADA para centralizar y visualizar los datos operativos en tiempo real, mejorando la gestión de la producción y la eficiencia operativa en QUIMFA S.A..

1.4. Definición del problema

A continuación, en la Figura 1.1 se presenta un diagrama de causa-efecto que da paso a la definición del problema por resolver en este proyecto.

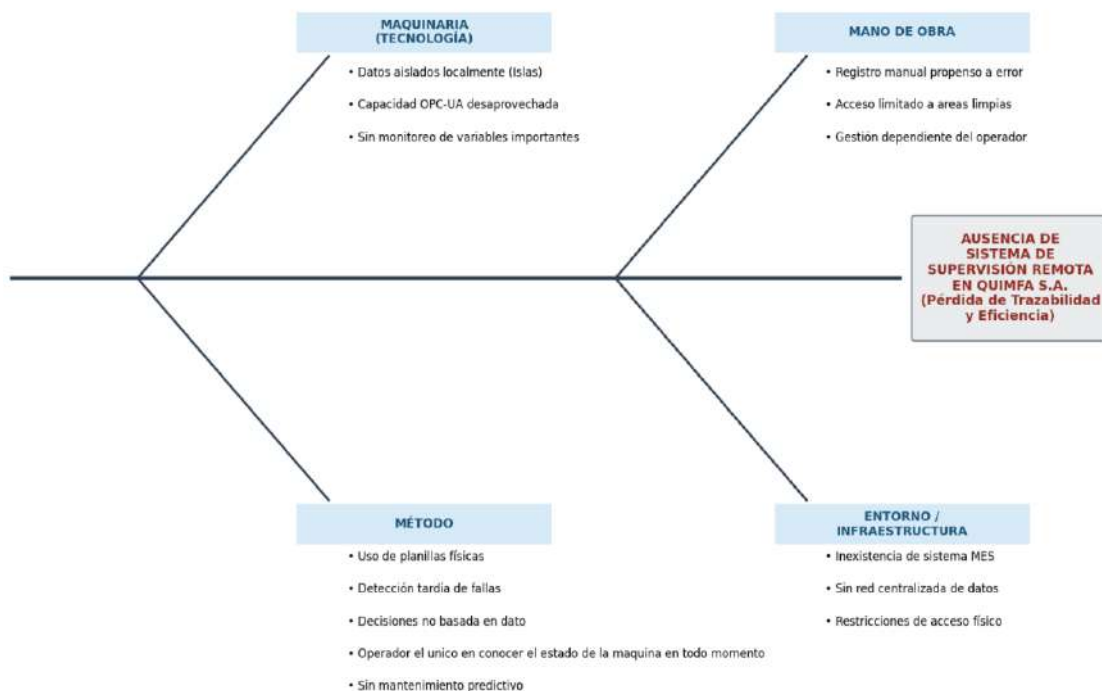


Figura 1.1 Diagrama de Causa-Efecto

En la planta QUIMFA S.A., existe una falta de integración de datos operativos de la blístera Romaco, lo que genera un desconocimiento sobre información clave relacionada con el rendimiento de la máquina. Este problema afecta la capacidad de tomar decisiones informadas y proactivas, ya que no se cuenta con un monitoreo en tiempo real que permita un análisis integral de los parámetros críticos de la máquina, como tiempos de paro y eficiencia operativa. La falta de un sistema centralizado de supervisión remota y cálculo de OEE impide optimizar los procesos productivos, reduciendo la eficiencia global y dificultando la mejora continua.

1.5. Funcionamiento de la máquina de blísteres

El proceso productivo de la blístera Romaco se estructura en módulos sincronizados que transforman el material de empaque y las tabletas en un blíster terminado. El ciclo comienza con el desenrollado de la lámina base (componente 1) que se alimenta hacia el sistema de calentamiento (componente 6) y moldeo (componente 7), donde se conforman las cavidades según el formato requerido. La lámina moldeada avanza hacia la zona de alimentación (componente 14), donde las tabletas se distribuyen automáticamente. En esta etapa, se verifica el llenado (componente 16) para asegurar que cada blíster esté correctamente formado antes de pasar al sellado (componente 28), donde ambas láminas se unen a presión y temperatura.

Luego, el material sellado pasa por las estaciones de compactado y corte (componentes 32 y 35), donde se ajustan las dimensiones finales del blíster. Los blísteres defectuosos son separados automáticamente del flujo principal, mientras que los conformes se transfieren a la cinta de descarga (componente 40) para su empaque final.

Este proceso modular mejora la productividad y genera datos operativos (como tiempos, rechazos y alarmas) esenciales para la supervisión remota y el cálculo de indicadores de eficiencia dentro del sistema SCADA. Para facilitar la identificación de los módulos y componentes mencionados, en el anexo se incluye una vista general esquemática de la máquina (véase Anexo K).

1.6. Identificación de parámetros de interés del proyecto

Una parte clave o esencial de todo proyecto es conocer que es lo que realmente se busca y desea controlar/monitorear (en otras palabras, la información necesaria para el proyecto). El objetivo es identificar las señales y variables que serán fundamentales para el desarrollo del sistema SCADA que monitoreará la Blístera Romaco en la planta de QUIMFA S.A. Esta etapa es crucial, ya que el éxito del sistema depende directamente de la calidad y pertinencia de los datos que se recopilen.

Sin una correcta identificación de las variables, cualquier sistema diseñado sería ineficiente e incapaz de proporcionar soluciones precisas, independientemente de su capacidad técnica.

El proceso de identificación de parámetros comenzó con una investigación exhaustiva sobre el equipo involucrado, específicamente la Blíster Romaco, y su funcionamiento dentro del entorno de producción de QUIMFA. Para ello, se consultaron documentos técnicos y se realizó un análisis detallado con el apoyo del jefe de mantenimiento y el gerente de ingeniería de la planta. Estas conversaciones fueron fundamentales para entender las necesidades específicas del proyecto y las características de la maquinaria. Además, se realizó una observación directa en planta para comprender mejor las dinámicas operativas, lo que permitió identificar los parámetros más relevantes para el monitoreo.

De acuerdo con los registros recopilados manualmente, se identificaron diversas irregularidades y limitaciones en el método actualmente utilizado para medir la productividad de las máquinas. Este método se basa únicamente en sumar las horas programadas, sin considerar cambios de turno, ajustes en los horarios de producción ni las paradas por mantenimiento; en otras palabras, asume que las máquinas operan en jornadas continuas.

Según los porcentajes proporcionados por el jefe de mantenimiento y el gerente de ingeniería, aproximadamente un 15% de los paros registrados en la semana analizada se debieron a tareas de mantenimiento, mientras que el resto corresponde a causas aún no identificadas. Esto permite afirmar que no se cuenta con información precisa sobre los factores que realmente están provocando los paros productivos y reduciendo el rendimiento de la máquina.

Para el diseño del sistema se seleccionaron señales simples y fácilmente transmisibles, de modo que no sea necesario modificar la arquitectura cada vez que se realicen cambios y que, además, puedan aplicarse a futuras implementaciones en otras máquinas sin importar su tipo o marca. A partir de estas señales, el sistema prioriza la recolección automática de datos y la generación de reportes en tiempo real. La plataforma Ignition permitirá obtener información instantánea sobre el estado de la máquina, generar indicadores precisos de productividad e identificar paros no planificados, contribuyendo así a mejorar la eficiencia operativa.

Señales Físicas	
Nombre Señal	Descripción
Luz Roja Continua	Señal que indica un paro en la máquina sin errores asociados.
Luz Roja	Señal que indica un paro debido a un error en la máquina.

Luz Verde Continua	Señal que indica que la máquina está funcionando correctamente.
Luz Verde	Señal que indica que la máquina está esperando para reactivarse después de un paro.
Luz Amarilla	Señal que indica un paro en la máquina debido a problemas de alimentación.
Luz Roja Continua	Señal que indica un paro en la máquina sin errores asociados en una segunda unidad.
Luz Roja	Señal que indica un paro debido a un error en la máquina en la segunda unidad.
Luz Verde Continua	Señal que indica que la máquina está funcionando correctamente en la segunda unidad.
Luz Verde	Señal que indica que la máquina está esperando para reactivarse después de un paro en la segunda unidad.
Cantidad Blisters producidos	Cantidad total de blisters producidos
Cantidad Blisters en mal estado	Cantidad de blisters producidos con defectos por la máquina.
Cantidad Blisters en buen estado	Cantidad de blisters en buen estado (sin defectos) producidos por la máquina.
Presión de aire de entrada	Presión del aire comprimido suministrado desde la red neumática de planta hacia la blíster
Temperatura de avance de agua fría	Corresponde a la temperatura del agua de refrigeración en el tramo de entrada (impulsión) hacia el circuito de refrigeración de la blíster
Señales Generadas en la Plataforma	
Paro por Material	Justificación del motivo de paro, el sistema la pregunta al operario.
Paro por espera	Justificación del motivo de paro, el sistema la pregunta al operario.
Paro no planificado	Justificación del motivo de paro, el sistema la pregunta al operario.
Paro por ajustes	Justificación del motivo de paro, el sistema la pregunta al operario.
Paro por cambio formato	Justificación del motivo de paro, el sistema la pregunta al operario.
Paro por mantenimiento	Justificación del motivo de paro, el sistema la pregunta al operario.

Tabla 1.1. Determinación de señales de interés

A partir de la información recopilada se definió el diseño del sistema, cuyo objetivo es generar una línea de tiempo en tiempo real que muestre el estado de la máquina, diferenciando el tiempo activo del tiempo detenido. Este último incluye paros por mantenimiento, cambios de formato, ajustes del operario, esperas por máquinas conectadas, fallas en la alimentación y paros no planificados. Estas seis categorías, establecidas según criterios del departamento de Ingeniería, permiten

registrar con precisión los motivos de paro y comprender cómo afectan el tiempo activo y el rendimiento, facilitando la toma de decisiones para reducir tiempos muertos y mejorar la eficiencia global de la planta.



Figura 1.2. Concepto general de monitoreo

Se definieron varias variables que serán almacenadas en una base de datos SQL, accesible y modificable desde la plataforma SCADA. Esto permitirá almacenar un historial de los datos operativos y facilitar análisis comparativos a lo largo del tiempo. Además, la base de datos cumple una función crítica en el contexto farmacéutico, ya que permite registrar todas las modificaciones realizadas en la plataforma, garantizando la confiabilidad del sistema y asegurando el cumplimiento con las normativas relevantes, como la CFR-21 Parte 11 de la FDA. Esta base de datos también proporcionará funcionalidades adicionales, como el registro de nuevas máquinas y su información asociada, para la creación de indicadores de eficiencia y el cálculo de tiempos de producción en tiempo real para cada máquina.

1.7. Módulo de instrumentación electrónica para variables críticas

1.7.1. Selección de variables críticas

En este trabajo se seleccionaron como variables críticas de habilitación de arranque la presión del aire comprimido de alimentación y la temperatura del agua de refrigeración (avance), debido a que son medios indispensables para el funcionamiento estable de la blíster y cuentan con valores de referencia establecidos en la documentación del fabricante. Para el aire comprimido, se adoptó como condición de habilitación una presión de entrada dentro del rango mínimo–máximo indicado por el fabricante para la alimentación: mínimo 5 bar y máximo 8 bar. Adicionalmente, el sistema neumático establece como valor nominal de operación una presión de trabajo ajustada a 6 bar, lo cual se tomó como valor objetivo (setpoint) para la supervisión del estado del servicio. Anexo N

Para el sistema de refrigeración, la documentación técnica define para el agua de refrigeración una presión de servicio de mínimo 2 bar y máximo 3 bar, y una temperatura de avance entre 12 °C y 15 °C. En base a estos valores se plantea la instrumentación de temperatura en la línea de agua de avance, de modo que el sistema pueda validar que el servicio se encuentra en condiciones antes de habilitar el arranque (Ver Anexo M). La supervisión de la presión de agua se deja como mejora futura debido al alcance y al tiempo disponible para la implementación del piloto.

Como criterio de implementación, se propone un PLC independiente encargado exclusivamente de la lectura de estas variables y de generar una señal de permiso de arranque, evitando intervenir el controlador eléctrico original de la máquina. Esta decisión se alinea con la indicación del fabricante de que no se hace responsable de cambios, ampliaciones u otras intervenciones efectuadas por el operador

1.7.2. Requisitos funcionales del enclavamiento

El módulo electrónico propuesto debe implementar un enclavamiento de arranque (start permissive), cuyo objetivo es impedir el arranque de la máquina cuando las variables críticas de servicios (aire y refrigeración) se encuentren fuera de rango. Para ello, un PLC independiente adquiere las variables, evalúa condiciones y genera una señal de Permiso_Arranque (habilita o bloquea el inicio), manteniendo el control original de la máquina sin modificaciones en su controlador eléctrico. La lógica de decisión del enclavamiento se presenta en el diagrama de flujo del Anexo O

1.7.3. Arquitectura de hardware propuesta

Para implementar el enclavamiento de arranque sin intervenir el controlador original de la blíster, se propone un módulo de instrumentación independiente compuesto por: un transmisor de presión de aire (0–10 bar, salida 4–20 mA), un sensor de temperatura PT100 con transmisor 4–20 mA instalado en la línea de agua de avance del chiller, un PLC Siemens LOGO! 8.4 dedicado a la adquisición y evaluación de estas variables, un módulo de expansión analógica LOGO! AM2 para disponer de dos entradas analógicas configurables (0–10 V / 0–20 mA, aptas para 4–20 mA), una etapa de salida mediante relé/contacto de interfaz para generar la señal Permiso_Arranque, y la integración de señales hacia el SCADA para visualización y registro. La arquitectura general y el flujo de señales entre niveles de campo, control, supervisión y acción se presentan en el Anexo P. El PLC LOGO! 8.4 adquiere la presión de aire comprimido de alimentación y la temperatura del agua de refrigeración (avance) mediante señales analógicas provenientes de transmisores

industriales (4–20 mA) y evalúa su condición respecto a los rangos definidos en el apartado 1.7.1. En caso de cumplimiento simultáneo de las condiciones, se habilita la salida Permiso_Arranque; caso contrario, la salida permanece deshabilitada, bloqueando el arranque y generando el evento correspondiente.

La señal Permiso_Arranque se materializa como un contacto libre de potencial (relé de interfaz) que actúa como autorización externa de arranque (fail-safe por contacto normalmente abierto). Esta arquitectura se adopta como alternativa no invasiva, considerando que el fabricante no se responsabiliza por cambios o intervenciones en el controlador eléctrico efectuadas por el operador. El esquema de conexionado del módulo (sensores → LOGO! AM2 → LOGO! 8.4 y salida a relé) se presenta en el Anexo Q, y la lista de componentes con referencias típicas se presenta en el Anexo R.

Elemento	Especificación mínima	Nota
PLC auxiliar	Siemens LOGO! 8.4 24RCE (ref. típica 6ED1052-1MD08-0BA1)	PLC dedicado a presión/temperatura + Permiso_Arranque
Módulo analógico	LOGO! AM2 (ref. típica 6ED1055-1MA00-0BA2)	2 entradas analógicas; admite 4–20 mA
Sensor de presión	Transmisor 0–10 bar, salida 4–20 mA	WIKA A-10, 24 VDC
Sensor de temperatura	PT100 + transmisor 4–20 mA	PT100 (RS PRO 4 hilos) + Tx 4–20 mA
Relé interfaz	Relé 24 VDC, 1 contacto NA (contacto seco)	Ej.: TRS 24VDC 1CO, 6 A
Fuente 24 VDC	Industrial (100–240 VAC → 24 VDC)	24 VDC / 5 A

Tabla 1.2. Componentes mínimos del módulo propuesto

1.7.4. Lógica de enclavamiento

Una vez definidas las variables críticas y la arquitectura de instrumentación, el enclavamiento se implementa como una condición lógica de permiso de arranque. El PLC auxiliar genera la señal Permiso_Arranque, la cual habilita o bloquea el inicio de operación de la máquina según el cumplimiento simultáneo de las condiciones de servicio.

Para su implementación se definen dos señales binarias internas: Aire_OK y Temp_OK, obtenidas a partir de la comparación de las variables medidas con los rangos establecidos en los apartados anteriores. La salida del enclavamiento se calcula mediante una operación lógica:

$$\text{Permiso_Arranque} = \text{Aire_OK} \text{ AND } \text{Temp_OK}$$

Con el fin de mantener un comportamiento fail-safe, la salida de permiso acciona un relé de interfaz con contacto normalmente abierto: únicamente cuando el PLC confirma condiciones válidas se energiza la salida y se cierra el contacto, habilitando el arranque. Ante condición fuera de rango, pérdida de alimentación o falla del PLC auxiliar, la salida se desenergiza, el contacto permanece abierto y el arranque queda inhibido por defecto.

Adicionalmente, para evitar habilitaciones o bloqueos por fluctuaciones transitorias de lectura, se recomienda aplicar un retardo de validación (por ejemplo, de 2 a 5 s) previo a la activación del permiso, y registrar el motivo del bloqueo para su visualización y análisis posterior en el SCADA y la base de datos.

1.7.5. Integración con SCADA y base de datos

La integración del módulo de enclavamiento al sistema final se realiza mediante la publicación de variables de instrumentación (presión y temperatura) y del estado lógico Permiso_Arranque hacia el entorno SCADA, con el objetivo de habilitar supervisión en tiempo real, alarmas, y registro histórico para trazabilidad y análisis posterior. En la implementación prevista, dichas variables serán adquiridas y evaluadas por un PLC Siemens LOGO! 8.4, mientras que para la etapa de validación en entorno simulado se emplea un OpenPLC como controlador emulado, manteniendo la misma lógica de enclavamiento y exponiendo las variables mediante un servidor Modbus TCP/IP para su lectura desde Ignition.

En Ignition, las variables se gestionan como tags de proceso y se configuran alarmas asociadas al estado del permiso. Cuando Permiso_Arranque = 0, el sistema genera una alarma de “Bloqueo de arranque” y registra un evento con marca de tiempo, incluyendo el motivo del bloqueo (aire fuera de rango y/o temperatura fuera de rango) y los valores medidos al momento del evento. Esto permite disponer de evidencia objetiva del motivo que impide el arranque y vincularlo con eventos operativos para análisis posterior.

Para estandarizar el intercambio de datos entre el controlador (LOGO! 8.4 en implementación / OpenPLC en validación) y el SCADA, se propone un mapa Modbus mínimo que contemple registros para variables analógicas y bobinas para estados lógicos. En el SCADA se aplica el escalado correspondiente a unidades de ingeniería (Bar, °C) y se almacena el histórico de variables y eventos en la base de datos, lo que permite revisar tendencias previas a un bloqueo y soportar análisis de fallas y mejora continua.

Variable	Tipo Modbus	Dirección	Formato	Escalado en SCADA	Descripción
Presión aire	Holding Register	HR40001	Entero	HR/100 → bar	AirPressure_bar
Temp agua avance	Holding Register	HR40002	Entero	HR/10 → °C	ChillWaterTemp_C
Aire_OK	Coil	C00001	Bool	—	Condición binaria
Temp_OK	Coil	C00002	Bool	—	Condición binaria
Permiso_Arranque	Coil	C00003	Bool	—	1 habilita / 0 bloquea
Motivo_Bloqueo	Holding Register o Tag interno	HR40003 (opcional)	Código	1=AIRE,2=TEMP,3=AMBOS	Motivo del bloqueo

Tabla 1.3. Mapa de variables para integración Modbus TCP

1.8. Cálculo del OEE y Parámetros de Eficiencia

El cálculo de OEE implementado en el sistema para la Blíster Romaco en QUIMFA S.A. sigue un enfoque que considera tres factores clave: disponibilidad, rendimiento y calidad, los cuales son fundamentales para evaluar la efectividad total del equipo en el proceso productivo [4]. Para una representación esquemática del desglose del OEE y los tiempos asociados, véase Anexo L.

El sistema comienza por recolectar los datos necesarios mediante la lectura de las señales provenientes de los tags SCADA, los cuales incluyen el estado de la máquina (en funcionamiento o parada), la velocidad real de producción, y los conteos de productos buenos y defectuosos. Adicionalmente, se registran los tiempos relacionados con el funcionamiento de la máquina, los tiempos de paro planificados (por mantenimiento o cambio de formato) y los tiempos de paro no planificados (por fallos no programados). Estos datos se extraen automáticamente, eliminando la intervención manual y mejorando la precisión de la información.

El primer factor, la **disponibilidad** (A), según [6] se calcula como el ratio entre el tiempo de funcionamiento real de la máquina (t_o) y el tiempo de producción disponible teóricamente (t_r). El tiempo de producción disponible teóricamente se refiere al tiempo total de operación esperado, considerando las interrupciones planificadas (como paros programados para mantenimiento o descansos). El tiempo de funcionamiento real se obtiene cuando la máquina está operativa y produce a plena capacidad, excluyendo tiempos de paro no planificados. Si la máquina está parada, ya sea por mantenimiento o ajustes, este tiempo se resta del tiempo total disponible, lo que permite calcular de manera precisa la disponibilidad del equipo.

El segundo factor, el **rendimiento (P)**, según [6] mide cuán eficiente es la máquina en términos de producción real comparada con la producción teórica esperada, es decir, si está produciendo a la velocidad nominal esperada. Este cálculo se realiza tomando la velocidad nominal de producción del formato actual de la máquina, que se obtiene desde la base de datos del sistema. Luego, se calcula la producción teórica de acuerdo con esta velocidad y el tiempo de funcionamiento real de la máquina (t_o). El rendimiento se determina comparando el número total de unidades producidas con la producción teórica esperada, lo que da una medida de cuán cerca está la máquina de su capacidad óptima. Si el rendimiento excede el 120% de la producción esperada, se limita a ese valor para evitar distorsionar los cálculos.

El nivel de **calidad (Q)** según [6] se calcula como la proporción de productos buenos producidos respecto al total de productos fabricados, es decir, la relación entre el número de productos buenos (t_q) y el total producido. Este factor es crítico, ya que, aunque la máquina pueda estar operando a una buena velocidad, es fundamental que los productos generados sean de alta calidad. Si la máquina produce demasiados productos defectuosos, la eficiencia global de la producción se ve comprometida.

El **OEE** total se obtiene multiplicando los tres factores anteriores: $OEE = Q \times P \times A$ [4]. Esto proporciona una métrica única que refleja la eficiencia global del equipo, considerando no solo la cantidad de tiempo que la máquina está en funcionamiento, sino también su rendimiento y la calidad de los productos. Este cálculo es fundamental para identificar áreas de mejora, ya que muestra cómo el equipo está operando en términos de eficiencia total.

1.9. Diseño y Configuración de la Base de Datos SQL

En el proyecto se utilizó PostgreSQL como SGBD open source y gratuito, distribuido bajo la licencia PostgreSQL, lo que permite su uso en entornos comerciales y no comerciales sin costos de licenciamiento ni restricciones relevantes, siempre que no se presente como una versión modificada propietaria. Sobre esta plataforma se diseñó un esquema relacional para almacenar de forma organizada y confiable los datos del SCADA, registrando estados de la blíster simulada, contadores de producción, eventos de paro y mensajes de diagnóstico, con relaciones claras entre tablas para evitar redundancias y facilitar el cálculo de indicadores de eficiencia (OEE).

La base de datos funcionó como repositorio central del piloto, recibiendo continuamente datos desde Ignition (scripts de simulación y módulos de cálculo) mediante consultas e instrucciones SQL ejecutadas desde el SCADA para insertar, actualizar y consultar registros automáticamente ante cambios de estado o producción. Se priorizó una estructura mínima con tablas de

producción/OEE, eventos de paro y catálogos (tipos de paro y mensajes), suficiente para validar el sistema y escalable para añadir futuras máquinas, turnos o niveles de análisis sin cambiar la arquitectura general.

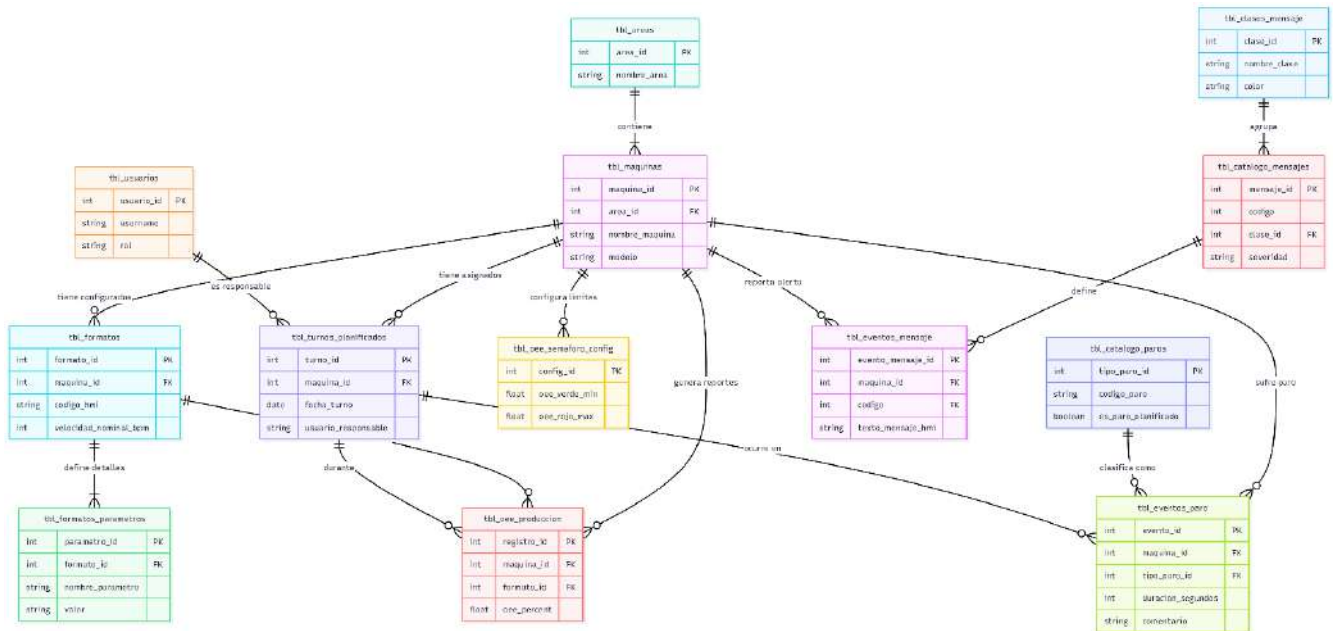


Figura 1.3. Modelo ERD que representa la organización de la base de datos

1.10. Evaluación y selección de la plataforma tecnológica

Tras analizar alternativas como Siemens WinCC y Wonderware InTouch, se eligió Ignition como plataforma SCADA debido a su arquitectura abierta, su integración sencilla con PostgreSQL y su modelo de licenciamiento más accesible. Ignition ofrece una única licencia por servidor sin límites de tags ni conexiones, lo que permite ampliar el sistema sin costos adicionales por cada variable añadida, a diferencia de los modelos basados en tags de WinCC o de los Historians propietarios requeridos por InTouch. Esta flexibilidad reduce costos a largo plazo y facilita la expansión futura del sistema.

Además, Ignition destaca por su arquitectura web, que permite acceso remoto desde cualquier dispositivo sin instalar clientes adicionales, y por su servidor OPC UA integrado, útil para futuras integraciones con PLCs industriales en planta. Su combinación de escalabilidad, interoperabilidad y bajo costo operativo la convierte en la opción más adecuada para el proyecto, ofreciendo una solución robusta y preparada para el crecimiento de QUIMFA S.A

Criterio de Selección	Peso (%)	Siemens WinCC	Wonderware InTouch	Ignition SCADA
-----------------------	----------	---------------	--------------------	----------------

Modelo de Licenciamiento	30%	Bajo (Por Tags/Limitado)	Bajo (Por Tags/Costoso)	Alto (Ilimitado)
Conectividad SQL Nativa	25%	Medio (Requiere Add-ons)	Medio (Complejo/Prop.)	Alto (Nativo/Directo)
Arquitectura Web/Móvil	20%	Medio (WebUX/Navigator)	Medio (Access Anywhere)	Alto (Nativo HTML5)
Interoperabilidad (OPC-UA)	15%	Alto (Nativo Siemens)	Alto (Vía Drivers)	Alto (Driver Incluido)
Costo Total de Propiedad	10%	Medio	Alto	Bajo (Óptimo)
PUNTUACIÓN FINAL (1-5)	100%	3.5	3.3	4.9 (GANADOR)

Tabla 1.4. Matriz de Decisión Ponderada

La elección de Ignition se justifica claramente al obtener la puntuación más alta en cada criterio relevante, particularmente en la flexibilidad de licenciamiento, conectividad nativa con SQL, y su arquitectura basada en Web, lo que ofrece una solución escalable, eficiente y económica para QUIMFA S.A.

1.11. Desarrollo del SCADA en Ignition

El sistema SCADA utilizado en este proyecto se desarrolló sobre la plataforma Ignition, seleccionada por su capacidad para integrarse con tecnologías modernas, trabajar bajo arquitecturas escalables y operar de manera nativa con protocolos industriales como OPC-UA. En el contexto del piloto, Ignition funcionó como núcleo del sistema de supervisión, permitiendo centralizar las variables simuladas de la blíster, gestionar la comunicación interna y estructurar la información necesaria para la visualización remota y el análisis de desempeño.

Una de las principales ventajas de su uso fue la posibilidad de construir interfaces accesibles desde cualquier dispositivo con navegador web, lo que facilitó la supervisión remota sin requerir instalaciones adicionales. Asimismo, sus módulos de diseño, simulación y registro de datos permitieron reproducir de manera fiel el flujo de información esperado en una futura integración real dentro de la planta farmacéutica QUIMFA, brindando así un entorno seguro y controlado para validar la viabilidad del sistema propuesto.

1.11.1. Módulos de Ignition utilizados en el proyecto

1.11.1.1. OPC-UA Server

Ignition incorpora un servidor OPC UA interno como mecanismo estándar de interoperabilidad industrial. No obstante, en esta tesis la comunicación de control se valida mediante Modbus TCP/IP, ya que el enclavamiento de arranque se ejecuta en un SoftPLC (OpenPLC) que expone registros y bobinas por dicho protocolo, replicando el esquema previsto para un PLC Siemens LOGO! 8.4 en una implementación física. Por esta razón, el servidor OPC UA interno no se utilizó en la etapa de validación; su empleo queda como alternativa para futuras ampliaciones e integración con otros equipos o líneas de producción.

1.11.1.2. Perspective Module

Se utilizó el módulo Perspective para desarrollar las pantallas de supervisión remota accesibles desde computadoras, tablets o teléfonos móviles. Perspective permitió construir dashboards responsivos, donde se visualizaron en tiempo real los estados de máquina, métricas de producción, tendencias y alarmas. Su diseño basado en web completó el objetivo de habilitar una supervisión remota desde cualquier dispositivo con navegador, sin necesidad de instalaciones adicionales.

1.11.1.3. Reporting Module

En el contexto del proyecto, el módulo Reporting de Ignition se utilizó para generar reportes automáticos de producción y OEE de la blíster Romaco, a partir de los datos almacenados en la base de datos SQL y de los tags del SCADA. Se diseñaron plantillas parametrizadas que consolidan tiempos de paro, unidades buenas/defectuosas y disponibilidad, presentadas en tablas y gráficos. Los informes se programaron para su ejecución periódica y exportación en formato PDF, permitiendo su distribución interna por correo o almacenamiento en un directorio de red para análisis histórico.

1.12. Programación de la Lógica de Simulación

1.12.1. Definición de Variables

Para el desarrollo del programa se crearon un total de 22 Tags/variables. Estas se clasificaron en distintos folders para facilitar su distinción y ordenamiento lógico dentro de la arquitectura del proyecto. Se agruparon según su funcionalidad operativa: contadores de producción, variables de

estado de máquina, variables OEE, variables de paro, mensajes y códigos de evento, simulación y parámetros auxiliares.

Nombre	Tipo	Carpeta	Dato	Descripción
TotalCount	Memory / OPC	NBP950	Integer	Contador acumulativo total de blisters producidos por la máquina. Incrementa en cada ciclo válido.
GoodCount	Memory / OPC	NBP950	Integer	Contador de unidades en buen estado, útil para el cálculo de Calidad en OEE.
BadCount	Memory / OPC	NBP950	Integer	Contador de unidades rechazadas; se utiliza para métricas de desperdicio y calidad del proceso.

Tabla 1.5. Tabla de contadores y producción

Nombre	Tipo	Carpeta	Dato	Descripción
EstadoMaquina	OPC	NBP950	Integer	Indica el estado operativo de la máquina. Ej.: 0 = STOP, 1 = RUN
FormatoID	Memory	NBP950	Integer	Identificador numérico del formato cargado (corresponde a tbl formatos).
SimVelocidadBPM	Memory	NBP950	Integer	Velocidad simulada en BPM para pruebas del SCADA sin producción real.

Tabla 1.6. Tabla de Estado de máquina y formato

Nombre	Tipo	Carpeta	Dato	Descripción
MensajeCodigo	OPC	NBP950	Integer	Código numérico asociado al mensaje de estado/alarma recibida desde el PLC.
MensajeTexto	OPC	NBP950	String	Texto descriptivo del estado o alarma actual, proveniente del HMI oficial.

Tabla 1.7. Tabla de sistema de mensajes

Nombre	Tipo	Carpeta	Dato	Descripción
OEE_Actual	Memory	NBP950	Double	Valor OEE global calculado en tiempo real.
OEE_Disponibilidad	Memory	NBP950	Double	Relación entre tiempo operando y tiempo total del turno.
OEE_Rendimiento	Memory	NBP950	Double	Relación entre velocidad real y velocidad teórica del formato.
OEE_Calidad	Memory	NBP950	Double	Porcentaje de unidades buenas respecto al total producido.
OEE_LastTS	Memory	NBP950	DateTime	Última marca de tiempo en la que se recalculó el OEE.

Tabla 1.8. Tabla de variables OEE

Nombre	Tipo	Carpeta	Dato	Descripción
OEE_TiempoOperando_s	Memory	NBP950	Double	Tiempo acumulado en segundos durante el cual la máquina estuvo en RUN.
OEE_TiempoParoNP_s	Memory	NBP950	Double	Tiempo total en segundos de paros no planificados.
OEE_TiempoParoPlanif_s	Memory	NBP950	Double	Tiempo total de paros planificados.
OEE_TiempoPlanificado_s	Memory	NBP950	Double	Duración completa del turno en segundos.

Tabla 1.9. Tabla de variables de tiempo (para cálculos OEE)

Nombre	Tipo	Carpeta	Dato	Descripción
ParoActivo	Memory	NBP950	Boolean	Indica si existe un paro activo en tiempo real.
ParoCodigo	OPC / Memory	NBP950	String	Código del tipo de paro (NP, AJ, CF, MAT).
ParoEventoID	Memory	NBP950	Integer	ID del evento de paro activo (se corresponde con <code>tbl_eventos_paro</code>).

Tabla 1.10. Tabla de sistemas de paros

Nombre	Tipo	Carpeta	Dato	Descripción
TurnoID	Memory	NBP950	Integer	Identificador del turno actual.
TimerPing	Memory / Gateway	Global	String	Señal de vida enviada desde el Gateway para verificar conectividad.

Tabla 1.11. Tabla de parámetros auxiliares

1.12.2. Desarrollo del simulador de proceso

Ante la imposibilidad de trabajar directamente sobre la blístera física, se implementó un gemelo digital simplificado de la Romaco dentro del servidor de Ignition, mediante un Gateway Timer Script en Jython denominado `SimRomaco` (código completo en Anexo C). Este script se ejecuta de forma periódica y actualiza un conjunto de tags de memoria bajo la ruta `[default]NBP950/`, emulando la producción, el rechazo de piezas y la generación de eventos de fallo.

En cada ciclo, el script lee los tags `EstadoMaquina`, `SimVelocidadBPM`, `TotalCount`, `GoodCount` y `BadCount` mediante `system.tag.readBlocking`. Si la máquina virtual está en RUN y la velocidad simulada es mayor a cero, se calcula la producción teórica del intervalo a partir de la velocidad en blisters por minuto, convirtiéndola a blisters por segundo y aplicando un factor de variación aleatorio para introducir ruido en la simulación. El resultado se redondea a un número entero de piezas producidas y, sobre ese valor, se aplica una tasa de rechazo del 2 %, pieza por

pieza, utilizando probabilidades aleatorias. Los contadores `TotalCount`, `GoodCount` y `BadCount` se actualizan en consecuencia y se escriben nuevamente con `system.tag.writeBlocking`.

Además, el simulador incorpora un motor de eventos aleatorios que reproduce mensajes informativos, advertencias y fallas críticas. Para ello, se definen grupos de códigos de mensaje (no críticos, de proceso y críticos) y, con una probabilidad acotada por ciclo, se selecciona uno al azar. El código se escribe en `MensajeCodigo` y su descripción se obtiene de la tabla `tbl_catalogo_mensajes` mediante `system.db.runScalarPrepQuery`, almacenándose en `MensajeTexto` y registrándose en `tbl_eventos_mensaje`. Cuando el código pertenece al grupo de errores críticos, el script detiene la máquina virtual (`EstadoMaquina = 0`). De este modo, el gemelo digital no solo genera producción y rechazo, sino que también alimenta la lógica de paros y mensajes del sistema SCADA de forma consistente, permitiendo probar la supervisión, el registro histórico y el cálculo de eficiencia sobre una base de datos de eventos simulados pero realistas.

1.12.3. Implementación del algoritmo de cálculo de OEE

El cálculo del OEE (Overall Equipment Effectiveness) se implementó como un Gateway Script independiente, `CalcOEE` (Anexo E), que se ejecuta de forma periódica sobre los tags asociados a la blíster simulada. El script lee, mediante `system.tag.readBlocking`, el estado de la máquina (`EstadoMaquina`), la velocidad real (`SimVelocidadBPM`), los contadores de producción (`TotalCount`, `GoodCount`, `BadCount`), el `FormatoID` y los acumuladores de tiempo `OEE_TiempoPlanificado_s`, `OEE_TiempoOperando_s`, `OEE_TiempoParoNP_s`, `OEE_TiempoParoPlanif_s`, junto con el timestamp `OEE_LastTS`. A partir de este último y de la hora actual, se calcula el intervalo de tiempo transcurrido en segundos, que se distribuye entre tiempo operando, paro planificado o paro no planificado según el valor de `EstadoMaquina`, `ParoActivo` y `ParoCodigo` (por ejemplo, "CF" y "M" se consideran paros planificados).

El tiempo planificado total se obtiene como la suma del tiempo operando y de los tiempos de paro planificado y no planificado. Con el `FormatoID` activo, el script consulta la tabla `tbl_formatos` mediante `system.db.runScalarPrepQuery` para obtener la velocidad nominal en blísters por minuto. Este valor permite calcular la producción teórica durante el tiempo en marcha y, a partir de ella, el rendimiento como la razón entre la producción real (`TotalCount`) y la producción teórica, limitando el resultado a un máximo del 120 %. La disponibilidad se calcula como el cociente entre tiempo operando y tiempo planificado, y la calidad como la razón entre piezas

buenas y totales ($\text{GoodCount}/\text{TotalCount}$). El OEE global se obtiene como el producto de estos tres factores.

Finalmente, los valores de disponibilidad, rendimiento, calidad y OEE se convierten a porcentajes con dos decimales y se escriben en los tags `OEE_Disponibilidad`, `OEE_Rendimiento`, `OEE_Calidad` y `OEE_Actual`, junto con la actualización de los acumuladores de tiempo y del timestamp `OEE_LastTS`, utilizando `system.tag.writeBlocking`. De esta forma, el sistema SCADA dispone de indicadores de OEE en tiempo real, calculados sobre los datos generados por la simulación.

1.13. Desarrollo de la Interfaz Gráfica

El desarrollo de la interfaz gráfica se realizó en Ignition, priorizando una estructura limpia y segmentada para evitar la sobrecarga de información en una sola pantalla. Se definió una arquitectura de navegación con pantallas diferenciadas para autenticación, operación, análisis y reportes, manteniendo siempre una jerarquía clara entre las funciones de operación en línea y las tareas de análisis y gestión.

Como primera capa se implementó una ventana de inicio de sesión, que valida credenciales contra una tabla de usuarios en la base de datos SQL y asigna permisos por rol (operador, supervisor o administrador). Solo los usuarios autenticados acceden a la supervisión y las funciones sensibles (modificación de parámetros, generación de reportes), reforzando la seguridad y la trazabilidad exigidas en el entorno farmacéutico.

Superada la autenticación, el supervisor accede a una pantalla principal de operación, concebida como vista de trabajo diario. En ella se visualizan el estado de la blíster Romaco, la velocidad instantánea en BPM, los contadores de unidades buenas y rechazadas, y los indicadores de OEE y sus componentes (Disponibilidad, Rendimiento y Calidad) en tiempo real. En la misma pantalla se incorporó la selección del formato de blíster desde la base de datos, de modo que el sistema recupere la velocidad nominal asociada al formato y pueda calcular la producción teórica y el rendimiento sin intervención manual. Esta concentración de funciones operativas en una única vista evita la dispersión de información y facilita el uso por parte del operario.

La gestión de paros se resolvió mediante una ventana emergente que se activa automáticamente cuando el sistema detecta el paso a estado detenido. En ella el operador clasifica el paro seleccionando el motivo entre categorías predefinidas vinculadas a la base de datos (mantenimiento, ajuste, cambio de formato, material, no planificado, entre otros). Cada registro

almacena motivo, marca temporal y usuario, alimentando tanto el cálculo de disponibilidad como una vista de análisis de paros donde se presentan los motivos acumulados por día o turno, preparada para futuros análisis tipo Pareto sin recargar la pantalla de operación.

En paralelo, la interfaz integra una zona de visualización de datos y OEE que combina indicadores numéricos con un historial dinámico de eventos. Este historial registra estados, alarmas y mensajes de la máquina, permitiendo relacionar visualmente variaciones en el OEE con secuencias de paros y cambios de formato. De este modo se mantiene una estructura intuitiva: la pantalla de operación se centra en el estado actual y el desempeño inmediato, mientras que el detalle de eventos y motivos se presenta de forma ordenada y sin saturación.

Finalmente, se desarrolló una función específica de reportes, orientada a usuarios de supervisión y gestión. Desde esta ventana, mediante un selector de fecha y un único botón, el sistema invoca el módulo Reporting de Ignition, consulta la base de datos PostgreSQL y genera automáticamente un informe de producción diaria en formato PDF o Excel, consolidando producción total, OEE por turno y tiempos de paro clasificados por motivo, garantiza que los reportes administrativos se generen de forma consistente y directamente a partir de los datos registrados en el sistema de supervisión.

1.14. Configuración de la Red VPN y Acceso Remoto

La arquitectura se basa en un Gateway de Ignition instalado en un PC/servidor dentro de la red local de la planta, donde corre la aplicación hecha en Perspective. Este módulo entrega interfaces web HTML5 responsivas accesibles desde cualquier navegador, sin clientes dedicados. En modo local, los usuarios de la LAN industrial ingresan por la URL del Gateway, por ejemplo <http://localhost:8088> (desde el servidor) o <http://192.168.x.x:8088> (desde otra PC usando la IP interna).

Para la supervisión remota segura se implementa una VPN tipo malla con Tailscale (WireGuard): cada equipo autorizado (Gateway, laptop, smartphone) entra a una red privada cifrada y recibe una IP virtual 100.x.x.x y un nombre interno con MagicDNS, permitiendo acceder como <http://100.xxx.xxx.xxx:8088> o <http://nombre-servidor:8088>. Así, no se expone el Gateway a Internet, no se requiere port forwarding del puerto 8088, y el acceso queda limitado a dispositivos autenticados, gestionado por identidades y ACL de Tailscale, reduciendo riesgos de escaneo y accesos no autorizados.

1.15. Reportes

La generación de reportes se implementó con el módulo Ignition Reporting, configurado para leer directamente de la base de datos SQL del sistema. Desde la ventana Reportes, el supervisor selecciona día o turno y, con un botón, se ejecutan consultas SQL sobre las tablas de producción, paros y OEE de la blistera Romaco, filtradas por fecha y máquina. Con esos resultados se genera automáticamente un informe en PDF que consolida producción total, OEE y tiempos de paro por motivo, listo para descarga, archivo o distribución interna para gestión y trazabilidad histórica.

1.16. Validación del Sistema Implementado

La validación del sistema se realizó con el objetivo de comprobar la coherencia del flujo de información dentro del entorno simulado, garantizando la consistencia de los datos almacenados, la correcta generación de reportes y la confiabilidad del cálculo de OEE. Adicionalmente, se validó la lógica del enclavamiento de arranque mediante un SoftPLC (OpenPLC) programado bajo el modelo IEC 61131-3, verificando su integración con el SCADA a través de Modbus TCP/IP como aproximación a la implementación física propuesta con PLC Siemens LOGO! 8.4.

1.16.1. Validación de la lógica de simulación

La primera etapa consistió en validar el comportamiento del simulador SimRomaco, encargado de emular la blistera Romaco mediante Gateway Timer Scripts y tags de memoria, reproduciendo cambios de estados, producción, rechazos y eventos de paro. En paralelo, se verificó el funcionamiento del enclavamiento ejecutando la lógica en OpenPLC, forzando valores de presión y temperatura fuera/dentro de rango y comprobando el cambio de estado de Permiso_Arranque y su correcta visualización en Ignition.

1.16.2. Verificación de integridad de datos en la base de datos SQL

La segunda etapa se centró en la verificación de la integridad de los datos registrados en la base de datos PostgreSQL empleada por el sistema. Cada transición de estado y cada evento de mensaje generado por la simulación debía almacenarse automáticamente en las tablas de eventos operativos, tales como `tbl_eventos` y `tbl_eventos_mensaje`, incluyendo la marca temporal, el identificador de máquina, el código de evento y el texto asociado. Se ejecutaron consultas directas sobre la base de datos, comparando los valores registrados con las secuencias de estados observadas en la HMI y con los logs generados por los scripts del Gateway. Este contraste permitió comprobar que no se producían pérdidas de eventos, que las marcas de tiempo correspondían con

el momento real de la transición y que la estructura de claves e índices permitía mantener un historial ordenado y trazable de la operación simulada. Adicionalmente, se verificó que los códigos de mensaje almacenados en base de datos coincidieran con las descripciones visualizadas en la interfaz, garantizando la consistencia entre la capa de supervisión y la capa de persistencia.

1.16.3. Certificación del algoritmo de cálculo de OEE

La tercera etapa correspondió a la validación del algoritmo de cálculo de OEE implementado en el script CalcOEE. Para ello se diseñaron distintos escenarios de prueba en la simulación, combinando periodos de operación continua, paros planificados (mantenimiento y cambio de formato), paros no planificados y variaciones de velocidad, así como distintos niveles de rechazo de producto. Para cada escenario se registraron los valores de tiempo operando, tiempos de paro planificado y no planificado, producción total y producción buena. También se comprobó que el algoritmo respetara las condiciones de saturación configuradas (por ejemplo, límite superior del rendimiento) y que el OEE acumulado se mantuviera consistente tanto en la visualización en tiempo real como en los valores históricos almacenados. Esta certificación permitió concluir que el módulo de cálculo de OEE implementado era matemáticamente correcto y robusto frente a diferentes patrones de operación simulada.

1.16.4. Pruebas de experiencia de usuario en Perspective y acceso remoto mediante Tailscale

La última etapa de validación se enfocó en la experiencia de usuario y en el acceso remoto seguro a la aplicación. En primer lugar, se evaluó la interfaz desarrollada en el módulo Perspective de Ignition, verificando que los dashboards de producción y OEE se actualizarán en tiempo real, que los indicadores de estado de máquina, contadores y tiempos respondieran de forma fluida a los eventos generados por la simulación y que el diseño responsivo se adaptara correctamente a diferentes resoluciones de pantalla (monitores de escritorio, portátiles y dispositivos móviles). En segundo lugar, se validó el acceso remoto mediante la VPN Tailscale, conectando clientes externos al dominio configurado y accediendo al Gateway a través de la dirección virtual proporcionada por la VPN. Durante estas pruebas se midieron tiempos de respuesta percibidos, estabilidad de la sesión y consistencia de los datos mostrados respecto a la visualización local, confirmando que el sistema podía ser supervisado desde ubicaciones externas sin degradación significativa de la experiencia de uso. El uso de Tailscale permitió además comprobar que el acceso remoto se

realizaba sin necesidad de abrir puertos en el router, manteniendo el Gateway oculto de Internet pública y restringiendo la supervisión a dispositivos autenticados.

2. RESULTADOS Y ANÁLISIS

2.1. Resultados de la implementación técnica

El sistema integró correctamente la simulación de la blistera, el servidor SCADA Ignition, la base de datos PostgreSQL y la HMI desarrollada en Perspective. En las pruebas sobre la red local, realizadas principalmente en la misma laptop que ejecutaba el Gateway, la actualización de variables en las pantallas fue percibida como inmediata, sin retardos apreciables ni caídas del servicio, aunque con un uso elevado de memoria RAM. Las consultas directas a la base de datos confirmaron que los eventos generados por los scripts se registraron con su marca temporal y código correspondiente, sin pérdidas ni inconsistencias entre lo visualizado en la HMI y lo almacenado en SQL. El acceso remoto mediante Tailscale, desde una red de datos móviles, mostró un comportamiento similar al de la LAN: las pantallas se actualizaron con fluidez y no se observaron cortes de sesión, validando la viabilidad de la supervisión remota sin exponer puertos del Gateway a Internet pública.

2.2. Resultados operativos de la simulación

Operativamente, la simulación reprodujo un escenario de operación normal con paros moderados, donde el script SimRomaco generó producción, rechazos y eventos de falla controlados. La tasa de rechazo se configuró en torno al 2 %, valor estimado por la jefatura de planta, lo que permitió probar la lógica de separación entre producto bueno y defectuoso y su impacto en el factor de calidad. No obstante, los valores de disponibilidad y OEE se vieron fuertemente afectados por la limitación de la licencia educativa de Ignition, que obliga a reiniciar el Gateway cada dos horas, interrumpiendo los scripts y añadiendo periodos artificiales de no disponibilidad. En consecuencia, los indicadores obtenidos son válidos como verificación funcional del algoritmo de OEE y del flujo de información, pero no pueden interpretarse como estimaciones representativas del desempeño real de la blistera de QUIMFA. Los paros y fallas se generaron de forma aleatoria con fines de ensayo, por lo que tampoco se construyó un análisis estadístico formal de causas.

2.3. Análisis económico y presupuesto

El desarrollo del prototipo y la validación funcional realizados en esta tesis se enfocaron en un entorno simulado, por lo que no implicaron inversiones directas en hardware de planta (intervención sobre el controlador de la blistera) ni en instrumentación instalada. En particular, la validación del enclavamiento de arranque se ejecutó con un SoftPLC (OpenPLC) y se integró al

SCADA por Modbus TCP/IP, utilizando herramientas de software disponibles. Sin embargo, el diseño físico propuesto contempla un módulo electrónico independiente compuesto por PLC Siemens LOGO! 8.4, transmisor de presión 0–10 bar (4–20 mA), sensor PT100 con transmisor 4–20 mA y relé de interfaz de contacto seco. Estos componentes se consideran como presupuesto referencial para una fase de implementación real en planta.

El desarrollo del prototipo requirió unas 200 a 250 horas de trabajo académico, equivalente al esfuerzo que en una empresa podría considerarse costo de desarrollo. Aun así, el gasto en licencias es bajo frente al impacto potencial de un SCADA en reducir tiempos muertos y mejorar la eficiencia, pudiendo recuperarse rápidamente incluso con mejoras moderadas en el OEE de la blistera.

Concepto	Cantidad / Alcance	Costo estimado (USD)
Licencia Ignition 8.3 Pro (Se vende por servidor)	1 servidor con tags y clientes ilimitados	1.200
Servidor/PC para Gateway (hardware existente)	Equipo ya disponible en planta	0
Sistema operativo Windows (licencia existente)	Licencia ya adquirida por la empresa	0
Base de datos PostgreSQL versión 18.1	Software libre	0
VPN Tailscale	Uso en modalidad gratuita	0
Módulo de enclavamiento (propuesto)	1 set: LOGO! 8.4 + 2 transmisores 4–20 mA + relé + fuente/cableado	500
Total		1.700

Tabla 1.12. Presupuesto consolidado de la inversión inicial (prototipo simulado; hardware de enclavamiento por cotizar)

2.4. Consideraciones finales del diseño

Los resultados confirman que la arquitectura propuesta integra de forma coherente la supervisión SCADA, la base de datos SQL y el cálculo de OEE, permitiendo trazabilidad, registro histórico y visualización web segura. Adicionalmente, se definió un módulo electrónico no intrusivo de enclavamiento de arranque (PLC auxiliar + sensores + relé de interfaz) para impedir el inicio de la máquina cuando los servicios críticos (aire comprimido y agua de refrigeración) se encuentran fuera de rango, reforzando el carácter de automatización industrial del proyecto.

Aunque en esta tesis no se realizó la conexión física a la blistera, el prototipo establece una base sólida para una futura implementación real, ya que la lógica de enclavamiento se validó mediante SoftPLC (OpenPLC) siguiendo el modelo IEC 61131-3 y se integró al SCADA a través de Modbus TCP/IP. Este enfoque permite migrar el control emulado al PLC Siemens LOGO! 8.4 propuesto,

y posteriormente ampliar la adquisición de datos de la máquina mediante protocolos industriales (p. ej., Modbus TCP u OPC UA) en una línea funcionando bajo condiciones reales.

3. Conclusiones Y Recomendaciones

3.1. Conclusiones

- La centralización de los datos operativos de la blistera en el SCADA y en la base de datos SQL permitió superar las islas de automatización previamente identificadas y disponer de información trazable y unificada para la toma de decisiones.
- El sistema SCADA diseñado en Ignition, con interfaz web en Perspective, cumplió el objetivo general de proporcionar una supervisión moderna y remota del estado y desempeño de la blistera Romaco en QUIMFA S.A.
- La implementación de la blistera como gemelo digital mediante scripts en Python sobre tags de memoria permitió validar la lógica operativa y de fallas sin requerir la conexión al equipo físico.
- La integración de la VPN Tailscale con el Gateway de Ignition garantizó un acceso remoto seguro y estable al sistema de supervisión sin necesidad de exponer puertos de la planta a Internet pública.
- El algoritmo de cálculo de OEE implementado en el script CalcOEE y soportado por consultas SQL demostró exactitud y consistencia al reproducir correctamente los indicadores de disponibilidad, rendimiento y calidad en los distintos escenarios simulados.

Se concluye además que el módulo electrónico propuesto (LOGO! 8.4 + transmisores 4–20 mA + relé de interfaz) constituye una alternativa simple y no intrusiva para mejorar la seguridad operativa, al garantizar que la blistera solo habilite el arranque cuando la presión de aire (5–8 bar) y la temperatura de avance del chiller (12–15 °C) se encuentran dentro de los límites definidos.

3.2. Recomendaciones

- Se recomienda que, en una siguiente fase, se reemplacen progresivamente los tags de memoria del gemelo digital por señales reales provenientes de la máquina y del módulo de instrumentación propuesto. En particular, se sugiere migrar la validación realizada con OpenPLC hacia el PLC Siemens LOGO! 8.4 instalando los transmisores 4–20 mA (presión y PT100 con transmisor) y el relé de interfaz, y conectando dichas variables al SCADA mediante un protocolo industrial robusto (Modbus TCP/IP, y/o OPC UA según la arquitectura disponible en planta), de modo a consolidar el paso del entorno simulado a operación real.
- Se recomienda que QUIMFA adquiriera una licencia de Ignition en modalidad de servidor y disponga de un equipo dedicado para el Gateway SCADA, siguiendo el escenario de inversión analizado, a fin de asegurar un funcionamiento continuo y soportado institucionalmente.
- Se recomienda estandarizar la ingeniería del sistema utilizando tipos de datos definidos por el usuario (UDT) y plantillas reutilizables, de manera que la arquitectura pueda replicarse rápidamente en otras blisteras y líneas de proceso de la planta.
- Se recomienda considerar la implementación del sistema propuesto como un proyecto viable económicamente, dado que el costo de licenciamiento y de infraestructura del SCADA puede escalarse por etapas y, adicionalmente, el módulo electrónico de enclavamiento requiere un conjunto acotado de componentes (PLC auxiliar, dos transmisores y relé), con impacto potencial en la reducción de paradas por condiciones de servicios fuera de especificación y en la mejora del OEE.
- Se recomienda que, en una futura implementación más robusta, QUIMFA incorpore especialistas en análisis avanzado de datos y técnicas de machine learning para explotar los históricos generados por el sistema en la predicción de fallas y la optimización del desempeño de la blistera.
- Se recomienda evaluar el enlace de la base de datos del sistema SCADA con sistemas corporativos como SAP, a fin de mejorar la redundancia de la información y alimentar la toma de decisiones administrativas con datos de planta en tiempo real.
- Se recomienda incorporar en la interfaz de supervisión mecanismos que permitan seleccionar la visualización de estadísticas e indicadores en horizontes diarios, semanales o mensuales, facilitando el análisis de tendencias a diferentes escalas temporales.

4. Bibliografía

- [1] M. E. P. y J. E. Heppelmann, «How Smart, Connected Products Are Transforming Competition,» *Harvard Business Review*, p. 23, 2014.
- [2] B. Ding, «Pharma Industry 4.0: Literature review and research opportunities in sustainable pharmaceutical supply chains,» *ELSEVIER*, pp. 115-130, 2018.
- [3] B. H. John Bessant, «Islands, archipelagoes and continents: Progress on the road to computer-integrated manufacturing,» *Research Policy*, Vols. %1 de %2Volume 17, Issue 6,, pp. 349-362, 1988.
- [4] S. Nakajima, *INTRODUCTION TO TPM (TOTAL PRODUCTIVE MAINTENANCE)*, 1988.
- [5] OPC FOUNDATION, [En línea]. Available: <https://opcfoundation.org/>.
- [6] INTERNATIONALSTANDARD, *ISO 22400-2*, 2014.

5. Anexos

Anexo A. Entorno real en la planta



Figura 5.1. Blistera en planta en area limpia

Anexo B. Arquitectura lógica de navegación y control de acceso basado en roles

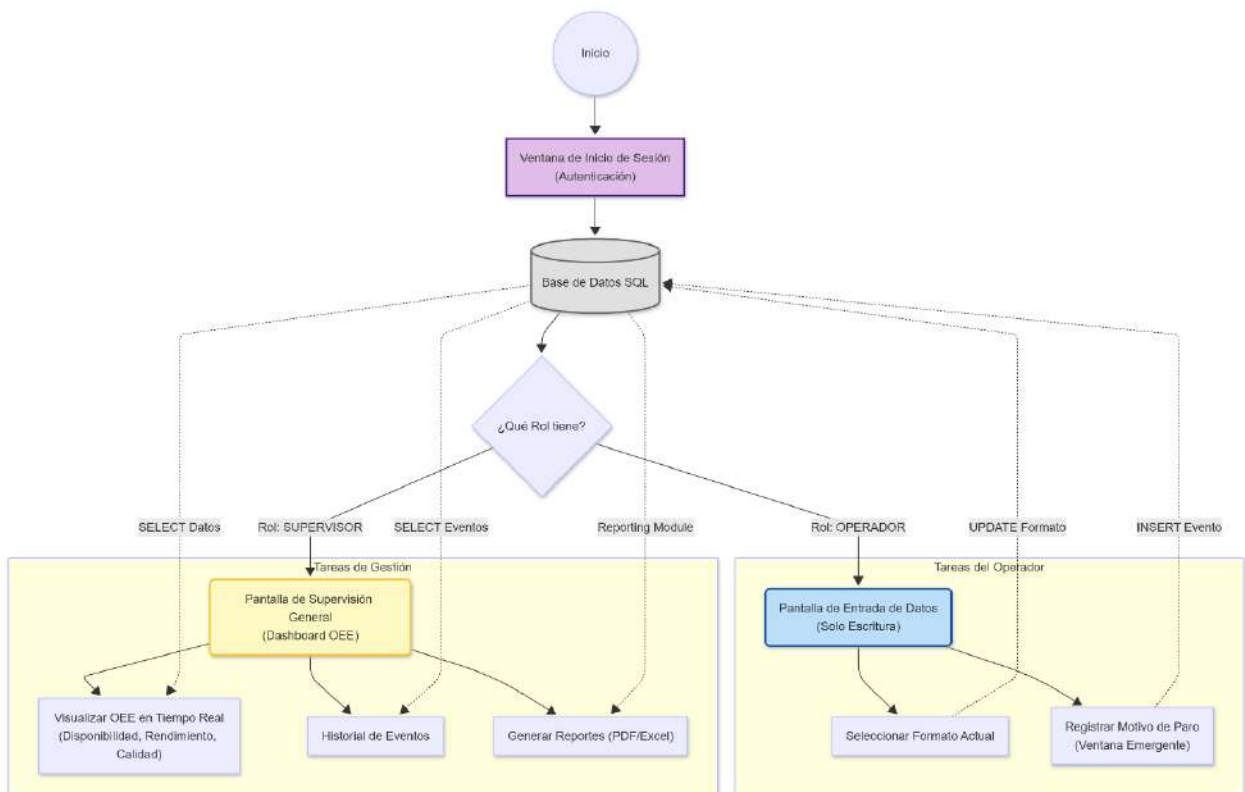


Figura 5.2.. Esquema del funcionamiento de las ventanas del sistema.

Anexo C. Código fuente del simulador de proceso SimRomaco

```
def handleTimerEvent():
    import random
    import system

    # Logger para verificar que el timer corre
    logger = system.util.getLogger("SimRomaco")
    logger.info("SimRomaco ejecutado (PFG.V3)")

    # -----
    # 1) Configuración básica
    # -----
    basePath = "[default]NBP950/"
    maquina_id = 1 # ID de la Romaco en tu BD

    # -----
    # 2) Lectura de tags necesarios
    # -----
    tags_lectura = [
        basePath + "EstadoMaquina",
        basePath + "SimVelocidadBPM",
        basePath + "TotalCount",
        basePath + "GoodCount",
        basePath + "BadCount"
    ]

    resultados = system.tag.readBlocking(tags_lectura)
    estado = resultados[0].value
    velocidad = resultados[1].value
    total = resultados[2].value or 0
    bueno = resultados[3].value or 0
    malo = resultados[4].value or 0

    # Si no está en RUN o no hay velocidad, no producimos nada
    if estado != 1 or velocidad is None or velocidad <= 0:
        return

    # -----
    # 3) Cálculo de producción simulada (enteros)
    # -----
    # Producción "ideal" en este ciclo (en piezas por segundo)
    incremento_base = float(velocidad) / 60.0
    factor_var = 1.0 + random.uniform(-0.1, 0.1)
    produccion_float = max(0, incremento_base * factor_var)

    # Número de piezas que realmente se producen en este ciclo (entero)
    piezas_producidas = int(round(produccion_float))
```

```
# Simular 2% de rechazo pieza por pieza
rechazadas = 0
for i in range(piezas_producidas):
    if random.random() < 0.02: # 2% de prob. de ser mala
        rechazadas += 1

buenas = piezas_producidas - rechazadas

# Actualizar contadores
total += piezas_producidas
bueno += buenas
malo += rechazadas

# (opcional, por seguridad)
total = max(0, int(total))
bueno = max(0, int(bueno))
malo = max(0, int(malo))

logger.info("SimRomaco contadores -> total=%s, bueno=%s, malo=%s"
           % (total, bueno, malo))

system.tag.writeBlocking(
    [
        basePath + "TotalCount",
        basePath + "GoodCount",
        basePath + "BadCount"
    ],
    [total, bueno, malo]
)

# -----
# 4) Simulación de mensajes / errores (EVENTOS SCADA)
# - Usa todos los códigos de tu catálogo
# - Solo los "críticos" para la máquina virtual
# -----

# Grupos de códigos
info_warn_codes = [1, 6, 11] # no críticos
process_error_codes = [27, 28] # error de proceso,
no paran
critical_error_codes = [21, 22, 23, 24, 25, 26, 29] # estos SÍ paran la
máquina

prob_error = random.randint(1, 100)

# Por ejemplo: 7% de probabilidad por ciclo de generar algún mensaje
if prob_error <= 7:

    # Primero elegimos si será mensaje "suave" o error
```

```
if random.random() < 0.5:
    # 50%: aviso / advertencia / estado
    codigo_error = random.choice(info_warn_codes + process_error_codes)
else:
    # 50%: errores (algunos críticos)
    codigo_error = random.choice(critical_error_codes)

# Guardar código en el tag
system.tag.writeBlocking(
    [basePath + "MensajeCodigo"],
    [codigo_error]
)

# --- CONSULTA A LA BD (USANDO runScalarPrepQuery) ---
descripcion = None
try:
    descripcion = system.db.runScalarPrepQuery(
        "SELECT descripcion FROM tbl_catalogo_mensajes WHERE codigo = ?",
        [codigo_error],
        "SCADA_Blistera_tesis"
    )
except Exception, e:
    logger.warn("Error al consultar descripcion de mensaje: %s" % e)

if descripcion is None:
    descripcion = "Mensaje simulado código %d" % codigo_error

# Construimos lista de tags/valores a escribir
tags_escritura = [
    basePath + "MensajeTexto"
]
valores_escritura = [
    descripcion
]

# Si es ERROR CRÍTICO -> parar la máquina virtual
if codigo_error in critical_error_codes:
    logger.warn("ERROR CRÍTICO simulado (%s): se detiene la máquina." %
codigo_error)

    # Paramos la máquina y marcamos paro activo
    tags_escritura.extend([
        basePath + "EstadoMaquina",
        basePath + "ParoActivo",
        basePath + "ParoCodigo"      # tipo de paro que usaremos (por
ahora NP)
    ])
    valores_escritura.extend([
        0,      # EstadoMaquina STOP
```

```
        True,          # ParoActivo
        "NP"          # Paro no planificado (puedes cambiar el mapping
luego)
    ])

# Escribimos todos los tags en un solo writeBlocking
system.tag.writeBlocking(tags_escritura, valores_escritura)

# Registrar evento en la tabla de mensajes
try:
    system.db.runPrepUpdate(
        """
        INSERT INTO tbl_eventos_mensaje (
            maquina_id,
            codigo,
            fecha_hora_inicio,
            texto_mensaje_hmi,
            acknowledged
        )
        VALUES (?, ?, NOW(), ?, FALSE)
        """,
        [maquina_id, codigo_error, descripcion],
        "SCADA_Blistera_tesis"
    )
except Exception, e:
    logger.warn("Error al insertar en tbl_eventos_mensaje: %s" % e)
```

Anexo D. Lógica de simulación de la blistera

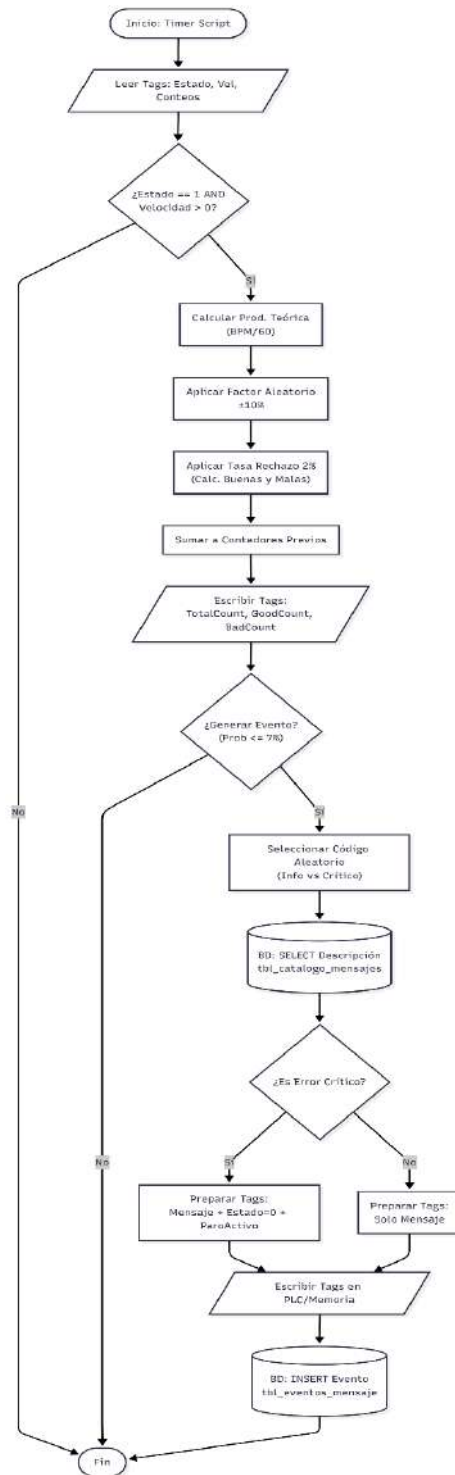


Figura 5.3.. Esquema logico de la simulacion de la Romaco

Anexo E. Código fuente de cálculo de OEE

```
def handleTimerEvent():
```

```
import system

logger = system.util.getLogger("CalcOEE")

basePath = "[default]NBP950/"

# 1) Leer tags necesarios
tags = [
    basePath + "EstadoMaquina",      # 0 = parada, 1 = RUN
    basePath + "SimVelocidadBPM",    # velocidad real
    basePath + "TotalCount",
    basePath + "GoodCount",
    basePath + "BadCount",
    basePath + "FormatoID",

    basePath + "OEE_TiempoPlanificado_s",
    basePath + "OEE_TiempoOperando_s",
    basePath + "OEE_TiempoParoNP_s",
    basePath + "OEE_TiempoParoPlanif_s",
    basePath + "OEE_LastTS"
]

vals = system.tag.readBlocking(tags)

estado      = vals[0].value or 0
vel_real_bpm = vals[1].value or 0
total       = vals[2].value or 0
bueno       = vals[3].value or 0
malo        = vals[4].value or 0
formato_id  = vals[5].value

t_planif    = float(vals[6].value or 0.0)
t_run       = float(vals[7].value or 0.0)
t_paro_np   = float(vals[8].value or 0.0)
t_paro_planif = float(vals[9].value or 0.0)
last_ts     = vals[10].value

now = system.date.now()

# Leer flags de paro
extra = system.tag.readBlocking([
    basePath + "ParoActivo",
    basePath + "ParoCodigo"
])
paro_activo = bool(extra[0].value or False)
paro_codigo = (extra[1].value or "").strip()

# 2) Calcular delta de tiempo desde la última ejecución
if last_ts is None:
```

```
# Primera vez: solo inicializamos timestamp
system.tag.writeBlocking([basePath + "OEE_LastTS"], [now])
return

dt_ms = system.date.millisBetween(last_ts, now)
dt_s = dt_ms / 1000.0
if dt_s <= 0:
    return

# 3) Acumular tiempos
if estado == 1:
    # Máquina RUN
    t_run += dt_s
else:
    # Máquina parada: ver si es planificado o no
    if paro_activo and paro_codigo in ("CF", "M"):
        # Cambio de formato o mantenimiento -> planificado
        t_paro_planif += dt_s
    else:
        # Todo lo demás lo consideramos NO planificado
        t_paro_np += dt_s

# El tiempo planificado es la suma de los tres
t_planif = t_run + t_paro_np + t_paro_planif

# 4) Obtener velocidad nominal del formato desde la BD
vel_nom_bpm = None
if formato_id not in (None, 0):
    try:
        vel_nom_bpm = system.db.runScalarPrepQuery(
            "SELECT velocidad_nominal_bpm "
            "FROM tbl_formatos WHERE formato_id = ?",
            [formato_id],
            "SCADA_Blistera_tesis"
        )
    except Exception, e:
        logger.warn("Error al leer velocidad_nominal_bpm: %s" % e)
        vel_nom_bpm = None

# 5) Calcular factores de OEE
disponibilidad = 0.0
rendimiento = 0.0
calidad = 0.0

if t_planif > 0:
    disponibilidad = t_run / t_planif # 0-1

if t_run > 0 and vel_nom_bpm is not None and vel_nom_bpm > 0:
    # Producción teórica a velocidad nominal
```

```
produccion_teorica = vel_nom_bpm * (t_run / 60.0) # blisters
if produccion_teorica > 0:
    rendimiento = float(total) / float(produccion_teorica)
    # Limitar a 120 %
    if rendimiento > 1.2:
        rendimiento = 1.2

if total > 0:
    calidad = float(bueno) / float(total)

oee = disponibilidad * rendimiento * calidad

# 6) Convertir a porcentajes con 2 decimales
disp_pct = round(disponibilidad * 100.0, 2)
rend_pct = round(rendimiento * 100.0, 2)
cal_pct = round(calidad * 100.0, 2)
oee_pct = round(oee * 100.0, 2)

# 7) Escribir resultados y actualizar timestamp
system.tag.writeBlocking(
    [
        basePath + "OEE_TiempoPlanificado_s",
        basePath + "OEE_TiempoOperando_s",
        basePath + "OEE_TiempoParoNP_s",
        basePath + "OEE_TiempoParoPlanif_s",
        basePath + "OEE_Disponibilidad",
        basePath + "OEE_Rendimiento",
        basePath + "OEE_Calidad",
        basePath + "OEE_Actual",
        basePath + "OEE_LastTS"
    ],
    [
        t_planif,
        t_run,
        t_paro_np,
        t_paro_planif,
        disp_pct,
        rend_pct,
        cal_pct,
        oee_pct,
        now
    ]
)

logger.info("OEE -> D=%.2f%, R=%.2f%, C=%.2f%%, OEE=%.2f%%"
            % (disp_pct, rend_pct, cal_pct, oee_pct))
```

Anexo F. Lógica del cálculo de OEE

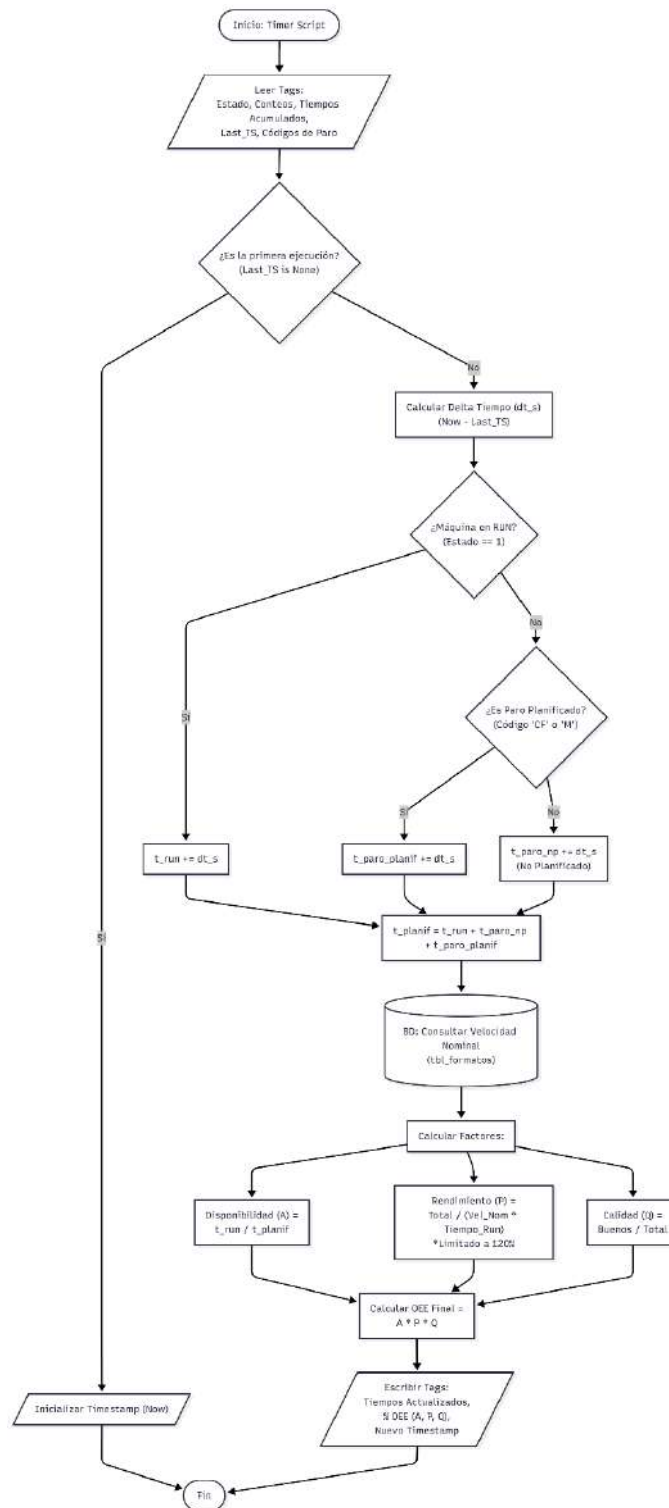


Figura 5.4. Esquema logico del calculo de OEE

Anexo G. script de creación de base de datos SQL

-- 1. Tabla de Usuarios

```
CREATE TABLE IF NOT EXISTS tbl_usuarios (  
    usuario_id SERIAL PRIMARY KEY,  
    username VARCHAR(50) UNIQUE NOT NULL,  
    password VARCHAR(200) NOT NULL, -- En producción: usar encriptación  
    rol VARCHAR(20) NOT NULL, -- 'Administrador', 'Operador'  
    activo BOOLEAN NOT NULL DEFAULT TRUE,  
    fecha_creacion TIMESTAMP NOT NULL DEFAULT NOW()  
);
```

-- 2. Tabla de Áreas

```
CREATE TABLE IF NOT EXISTS tbl_areas (  
    area_id SERIAL PRIMARY KEY,  
    nombre_area VARCHAR(50) UNIQUE NOT NULL,  
    descripcion VARCHAR(100)  
);
```

-- 3. Tabla de Máquinas

```
CREATE TABLE IF NOT EXISTS tbl_maquinas (  
    maquina_id SERIAL PRIMARY KEY,  
    area_id INTEGER REFERENCES tbl_areas(area_id),  
    nombre_maquina VARCHAR(50) UNIQUE NOT NULL,  
    modelo VARCHAR(50),  
    numero_serie VARCHAR(50),  
    activo BOOLEAN NOT NULL DEFAULT TRUE  
);
```

-- 4. Tabla de Formatos

```
CREATE TABLE IF NOT EXISTS tbl_formatos (  
    formato_id SERIAL PRIMARY KEY,  
    maquina_id INTEGER NOT NULL  
        REFERENCES tbl_maquinas(maquina_id) ON DELETE CASCADE,  
    codigo_hmi VARCHAR(30) UNIQUE NOT NULL, -- Ej: '950-43384'  
    version_hmi INTEGER NOT NULL DEFAULT 1, -- Version del formato  
    velocidad_nominal_bpm INTEGER, -- blisters por minuto  
    productos_por_blister INTEGER, -- cavidades / unidades por blister  
    numero_pistas SMALLINT, -- Número de pistas  
    avance_mm NUMERIC(10,2), -- Longitud de avance  
    ancho_blister_mm NUMERIC(10,2), -- Ancho del blister  
    largo_blister_mm NUMERIC(10,2), -- Largo del blister  
    fecha_ultima_modificacion TIMESTAMP,  
    usuario_ultima_modificacion VARCHAR(50) REFERENCES tbl_usuarios(username)  
);
```

-- 5. Tabla de Parámetros de Formato

```
CREATE TABLE IF NOT EXISTS tbl_formatos_parametros (  
    parametro_id SERIAL PRIMARY KEY,
```

```
formato_id INTEGER NOT NULL
    REFERENCES tbl_formatos(formato_id) ON DELETE CASCADE,
seccion VARCHAR(100) NOT NULL,      -- Ej: 'Basic parameters', 'Heating
station'
nombre_parametro VARCHAR(100) NOT NULL,
valor_texto VARCHAR(100),          -- Para valores tipo ON/OFF, modo,
etc.
valor_numerico NUMERIC(18,6),      -- Para valores numéricos
unidad VARCHAR(20),                -- Ej: mm, %, °C, etc.
orden_seccion SMALLINT,            -- Para mantener el orden
orden_parametro SMALLINT,
    UNIQUE (formato_id, seccion, nombre_parametro)
);
```

-- 6. Tabla de Turnos

```
CREATE TABLE IF NOT EXISTS tbl_turnos_planificados (
    turno_id SERIAL PRIMARY KEY,
    maquina_id INTEGER NOT NULL
        REFERENCES tbl_maquinas(maquina_id) ON DELETE CASCADE,
    nombre_turno VARCHAR(20),
    fecha_turno DATE NOT NULL,
    hora_inicio TIMESTAMP NOT NULL,
    hora_fin TIMESTAMP NOT NULL,
    usuario_responsable VARCHAR(50) REFERENCES tbl_usuarios(username)
);
```

-- 7. Tabla de Tipos de Paro

```
CREATE TABLE IF NOT EXISTS tbl_catalogo_paros (
    tipo_paro_id SERIAL PRIMARY KEY,
    codigo_paro VARCHAR(10) UNIQUE NOT NULL, -- 'NP', 'AJ', 'CF', 'M', 'MAT'
    descripcion_paro VARCHAR(100) NOT NULL,
    es_paro_planificado BOOLEAN NOT NULL DEFAULT FALSE
);
```

-- 8. Tabla de Eventos de Paro

```
CREATE TABLE IF NOT EXISTS tbl_eventos_paro (
    evento_id SERIAL PRIMARY KEY,
    maquina_id INTEGER NOT NULL
        REFERENCES tbl_maquinas(maquina_id) ON DELETE CASCADE,
    tipo_paro_id INTEGER NOT NULL
        REFERENCES tbl_catalogo_paros(tipo_paro_id),
    formato_id INTEGER REFERENCES tbl_formatos(formato_id),
    turno_id INTEGER REFERENCES tbl_turnos_planificados(turno_id),
    fecha_hora_inicio TIMESTAMP NOT NULL,
    fecha_hora_fin TIMESTAMP,
    duracion_segundos INTEGER,
    comentario_operador TEXT,
    usuario_registro VARCHAR(50) REFERENCES tbl_usuarios(username)
);
```

-- 9. Tabla de Producción OEE

```
CREATE TABLE IF NOT EXISTS tbl_oe_produccion (  
    registro_id SERIAL PRIMARY KEY,  
    maquina_id INTEGER NOT NULL  
        REFERENCES tbl_maquinas(maquina_id) ON DELETE CASCADE,  
    formato_id INTEGER NOT NULL  
        REFERENCES tbl_formatos(formato_id),  
    fecha DATE NOT NULL,  
    turno_id INTEGER REFERENCES tbl_turnos_planificados(turno_id),  
    total_blisters_producidos INTEGER DEFAULT 0,  
    total_blisters_buenos INTEGER DEFAULT 0,  
    total_blisters_malos INTEGER DEFAULT 0,  
    tiempo_total_turno_s INTEGER DEFAULT 0, -- fin - inicio del turno  
    tiempo_paros_no_planif_s INTEGER DEFAULT 0,  
    tiempo_paros_planif_s INTEGER DEFAULT 0,  
    disponibilidad_percent NUMERIC(5,2),  
    rendimiento_percent NUMERIC(5,2),  
    calidad_percent NUMERIC(5,2),  
    oee_percent NUMERIC(5,2)  
);
```

-- 10. Tabla de Semáforo OEE

```
CREATE TABLE IF NOT EXISTS tbl_oe_semaforo_config (  
    config_id SERIAL PRIMARY KEY,  
    maquina_id INTEGER NOT NULL  
        REFERENCES tbl_maquinas(maquina_id) ON DELETE CASCADE,  
    formato_id INTEGER NOT NULL  
        REFERENCES tbl_formatos(formato_id) ON DELETE CASCADE,  
    oee_verde_min NUMERIC(5,2) NOT NULL, -- Límite inferior del verde  
    oee_amarillo_min NUMERIC(5,2) NOT NULL, -- Límite inferior del amarillo  
    oee_rojo_max NUMERIC(5,2) NOT NULL, -- Valor por debajo del cual es rojo  
    UNIQUE (maquina_id, formato_id)  
);
```

```
CREATE TABLE IF NOT EXISTS tbl_oe_semaforo_eventos (  
    evento_id SERIAL PRIMARY KEY,  
    maquina_id INTEGER NOT NULL  
        REFERENCES tbl_maquinas(maquina_id) ON DELETE CASCADE,  
    formato_id INTEGER REFERENCES tbl_formatos(formato_id),  
    turno_id INTEGER REFERENCES tbl_turnos_planificados(turno_id),  
    estado VARCHAR(10) NOT NULL CHECK (estado IN ('VERDE', 'AMARILLO', 'ROJO')),  
    fecha_hora_inicio TIMESTAMP NOT NULL,  
    fecha_hora_fin TIMESTAMP,  
    duracion_segundos INTEGER  
);
```

-- 11. Tabla de Mensajes SCADA

```
CREATE TABLE IF NOT EXISTS tbl_clases_mensaje (  

```

```
clase_id INTEGER PRIMARY KEY,          -- 1, 6, 11, 21-29
nombre_clase VARCHAR(50) NOT NULL,
color_mensaje VARCHAR(10) NOT NULL,   -- 'Rojo','Naranja','Azul','Verde'
tipo_scada VARCHAR(20) NOT NULL      --
'Error','Advertencia','Aviso','Estado'
);
```

```
CREATE TABLE IF NOT EXISTS tbl_catalogo_mensajes (
  mensaje_id SERIAL PRIMARY KEY,
  codigo INTEGER NOT NULL,             -- 1,6,11,21-29,100-104...
  descripcion VARCHAR(200) NOT NULL,
  clase_id INTEGER REFERENCES tbl_clases_mensaje(clase_id),
  origen VARCHAR(20) NOT NULL,        -- 'Oficial' / 'SCADA'
  severidad VARCHAR(10) NOT NULL,     -- 'Alta','Media','Baja','Info'
  significado_funcional TEXT,
  accion_recomendada TEXT,
  UNIQUE (codigo)
);
```

```
CREATE TABLE IF NOT EXISTS tbl_eventos_mensaje (
  evento_mensaje_id SERIAL PRIMARY KEY,
  maquina_id INTEGER NOT NULL
    REFERENCES tbl_maquinas(maquina_id) ON DELETE CASCADE,
  codigo INTEGER NOT NULL
    REFERENCES tbl_catalogo_mensajes(codigo),
  fecha_hora_inicio TIMESTAMP NOT NULL,
  fecha_hora_fin TIMESTAMP,
  duracion_segundos INTEGER,
  texto_mensaje_hmi TEXT,
  acknowledged BOOLEAN NOT NULL DEFAULT FALSE,
  usuario_ack VARCHAR(50) REFERENCES tbl_usuarios(username)
);
```

-- 12. Insertar Datos Básicos

```
INSERT INTO tbl_usuarios (username, password, rol)
VALUES
  ('admin', '1234', 'Administrador'),
  ('operador1', '0000', 'Operador'),
ON CONFLICT (username) DO NOTHING;
```

```
INSERT INTO tbl_areas (nombre_area, descripcion)
VALUES ('Sólidos', 'Tabletas y Blisters')
ON CONFLICT (nombre_area) DO NOTHING;
```

```
INSERT INTO tbl_maquinas (area_id, nombre_maquina, modelo)
VALUES (1, 'Blíster Romaco', 'NBP 950')
ON CONFLICT (nombre_maquina) DO NOTHING;
```

```
INSERT INTO tbl_formatos (
```

```
maquina_id,
codigo_hmi,
version_hmi,
velocidad_nominal_bpm,
productos_por_blister,
numero_pistas,
avance_mm,
ancho_blister_mm,
largo_blister_mm,
fecha_ultima_modificacion
)
VALUES
-- Formatos ALU (datos tomados del resumen maestro)
(1, '950-43364AL', 7, 120, NULL, NULL, 86.00, NULL, NULL, NULL),
(1, '950-43365AL', 11, 120, NULL, NULL, 90.00, NULL, NULL, NULL),
(1, '950-43366AL', 6, 50, NULL, NULL, 149.00, NULL, NULL, NULL),
(1, '950-43367AL', 6, 60, NULL, NULL, 71.00, NULL, NULL, NULL),

-- Formatos estándar (tabla grande de formatos)
(1, '950-43368', 9, 120, 10, 2, 82.00, 41.00, 100.00, '2020-12-07 14:08:43'),
(1, '950-43369', 2, 120, 8, 2, 82.00, 41.00, 100.00, '2020-12-07 14:11:45'),
(1, '950-43372', 7, 100, 10, NULL, NULL, NULL, 98.00, '2020-12-07
14:18:15'),
(1, '950-43373', 2, 120, 10, 2, 78.00, 39.00, 98.00, '2020-12-07 14:26:40'),
(1, '950-43374', 2, 120, 1, 2, 78.00, 39.00, 98.00, '2020-12-07 14:29:22'),
(1, '950-43375', 5, 120, 4, 2, 82.00, 41.00, 100.00, '2020-12-07 14:32:06'),
(1, '950-43376', 3, 120, 4, 2, 82.00, 41.00, 100.00, '2020-12-07 14:35:00'),
(1, '950-43377', 3, 100, NULL, NULL, NULL, NULL, NULL, '2020-12-07
14:37:32'),
(1, '950-43378', 2, 120, 8, 2, 78.00, 41.00, 100.00, '2020-12-07 14:40:42'),
(1, '950-43379', 2, 100, 4, NULL, NULL, NULL, 98.00, '2020-12-07
14:45:53'),
(1, '950-43380', 2, 80, NULL, NULL, NULL, NULL, NULL, '2020-12-07
13:28:16'),
(1, '950-43381', 4, 100, 10, NULL, NULL, NULL, 100.00, '2020-12-07
14:49:37'),
(1, '950-43382', 3, 120, 10, 2, 78.00, 39.00, 98.00, '2020-12-07 14:52:42'),
(1, '950-43383', 2, 120, 4, 2, 82.00, 41.00, 100.00, '2020-12-07 14:55:17'),
(1, '950-43384', 3, 120, 8, 2, 82.00, 41.00, 100.00, '2020-12-07 15:01:41')
ON CONFLICT (codigo_hmi) DO UPDATE
SET
version_hmi = EXCLUDED.version_hmi,
velocidad_nominal_bpm = EXCLUDED.velocidad_nominal_bpm,
productos_por_blister = EXCLUDED.productos_por_blister,
numero_pistas = EXCLUDED.numero_pistas,
avance_mm = EXCLUDED.avance_mm,
ancho_blister_mm = EXCLUDED.ancho_blister_mm,
largo_blister_mm = EXCLUDED.largo_blister_mm,
fecha_ultima_modificacion = EXCLUDED.fecha_ultima_modificacion;
```

```
INSERT INTO tbl_catalogo_paros (codigo_paro, descripcion_paro,  
es_paro_planificado)
```

```
VALUES
```

```
    ('NP', 'Paro No Planificado (Falla)', FALSE),  
    ('AJ', 'Ajustes de Máquina', FALSE),  
    ('CF', 'Cambio de Formato', TRUE),  
    ('M', 'Mantenimiento', TRUE),  
    ('MAT', 'Falta de Material', FALSE),  
    ('E', 'Espera Anterior/Posterior', FALSE)
```

```
ON CONFLICT (codigo_paro) DO NOTHING;
```

```
INSERT INTO tbl_clases_mensaje (clase_id, nombre_clase, color_mensaje,  
tipo_scada)
```

```
VALUES
```

```
    (1, 'Aviso general', 'Azul', 'Aviso'),  
    (6, 'Estado de máquina', 'Verde', 'Estado'),  
    (11, 'Advertencia general', 'Naranja', 'Advertencia'),  
    (21, 'Parada de emergencia máquina', 'Rojo', 'Error'),  
    (22, 'Parada de emergencia máquina ext.', 'Rojo', 'Error'),  
    (23, 'Puertas de seguridad abiertas maq.', 'Rojo', 'Error'),  
    (24, 'Puertas seguridad abiertas ext.', 'Rojo', 'Error'),  
    (25, 'Error de sistema', 'Rojo', 'Error'),  
    (26, 'Error de sistema externo', 'Rojo', 'Error'),  
    (27, 'Error mínimo/máximo máquina', 'Rojo', 'Error'),  
    (28, 'Error mínimo/máximo máquina ext.', 'Rojo', 'Error'),  
    (29, 'Error general de máquina', 'Rojo', 'Error')
```

```
ON CONFLICT (clase_id) DO NOTHING;
```

```
INSERT INTO tbl_catalogo_mensajes (codigo, descripcion, clase_id, origen,  
severidad, significado_funcional, accion_recomendada)
```

```
VALUES
```

```
    (1, 'Aviso general', 1, 'Oficial', 'Baja', 'Mensaje informativo sin impacto  
directo sobre la seguridad ni la producción.', 'Revisar la condición informada si  
es relevante para la operación.'),
```

```
    (6, 'Mensaje de estado de máquina', 6, 'Oficial', 'Info', 'Indica un estado  
operativo de la máquina (p. ej. PREPARADO, AUTOMÁTICO).', 'Monitorear únicamente;  
no requiere acción correctiva inmediata.'),
```

```
    (11, 'Advertencia general', 11, 'Oficial', 'Media', 'Condición anómala que  
puede evolucionar a fallo si no se atiende.', 'Verificar el motivo de la  
advertencia y corregir la causa antes de que detenga la producción.'),
```

```
    (21, 'PARADA DE EMERGENCIA de máquina', 21, 'Oficial', 'Alta', 'La máquina ha  
sido detenida mediante un pulsador de parada de emergencia integrado.', 'Verificar  
la causa, restablecer dispositivos y ejecutar reinicio seguro.'),
```

```
    (22, 'PARADA DE EMERGENCIA de máquina externa', 22, 'Oficial', 'Alta',  
    'Una instalación externa asociada ha activado una parada de emergencia.',  
    'Revisar la instalación externa, eliminar el peligro y coordinar el  
reinicio.'),
```

(23, 'Puertas de seguridad abiertas - Máquina', 23, 'Oficial', 'Alta',
'Alguna puerta de seguridad o cubierta de protección de la máquina está abierta durante el funcionamiento.',
'Cerrar correctamente todas las puertas y confirmar enclavamientos.'),

(24, 'Puertas de seguridad abiertas - Máquina externa', 24, 'Oficial', 'Alta',
'Una puerta de seguridad perteneciente a un equipo externo está abierta.',
'Cerrar las protecciones del equipo externo y verificar enclavamientos.'),

(25, 'Error de sistema', 25, 'Oficial', 'Alta',
'Fallo interno del sistema de control de la máquina.',
'Revisar diagnóstico del PLC/HMI y contactar mantenimiento.'),

(26, 'Error de sistema externo', 26, 'Oficial', 'Alta',
'Fallo en un sistema o módulo externo conectado a la máquina.',
'Verificar el equipo externo, cableado y comunicaciones.'),

(27, 'Error de mínimo/máximo - Máquina', 27, 'Oficial', 'Media',
'Algún valor de proceso de la máquina está fuera de los límites mínimo/máximo permitidos.',
'Ajustar parámetros de proceso y revisar sensores asociados.'),

(28, 'Error de mínimo/máximo - Máquina externa', 28, 'Oficial', 'Media',
'Un valor de proceso de un sistema externo está fuera de los límites permitidos.',
'Verificar parámetros y sensores del equipo externo.'),

(29, 'Error general de máquina', 29, 'Oficial', 'Alta',
'Fallo general no categorizado de la máquina.',
'Revisar mensajes detallados en el HMI y aplicar el procedimiento de fallo.'),

-- Códigos SCADA propios (errores en serie)

(100, 'Error en serie: empalme detectado', 29, 'SCADA', 'Alta',
'El control en serie ha detectado repetidamente la presencia de un empalme en la lámina.',
'Inspeccionar la unión de la lámina, verificar el sensor y descartar el material afectado.'),

(101, 'Error en serie: código erróneo', 29, 'SCADA', 'Alta',
'Se han registrado varios códigos de lectura incorrectos de forma consecutiva.',
'Verificar impresión, limpieza del lector y alineación del producto.'),

(102, 'Error en serie: código ausente', 29, 'SCADA', 'Alta',
'El sistema ha detectado varios productos consecutivos sin código legible.',
'Comprobar sistema de impresión y presencia del código antes de continuar la producción.'),

(103, 'Error en serie: marca de impresión no detectada', 29, 'SCADA', 'Alta', 'El sensor de marca de impresión no detecta la marca durante varios ciclos.',

'Revisar contraste y posición de la marca, ajustar el sensor y sincronismo del avance de lámina.'),

(104, 'Error en serie: fisura ALU detectada', 29, 'SCADA', 'Alta',

'El control de fisuras en la lámina de aluminio ha detectado repetidamente fisuras.',

'Inspeccionar la bobina de ALU, ajustar parámetros y sustituir el tramo de lámina dañado.')

ON CONFLICT (codigo) DO NOTHING;

Anexo H. 1.Arquitectura general del sistema

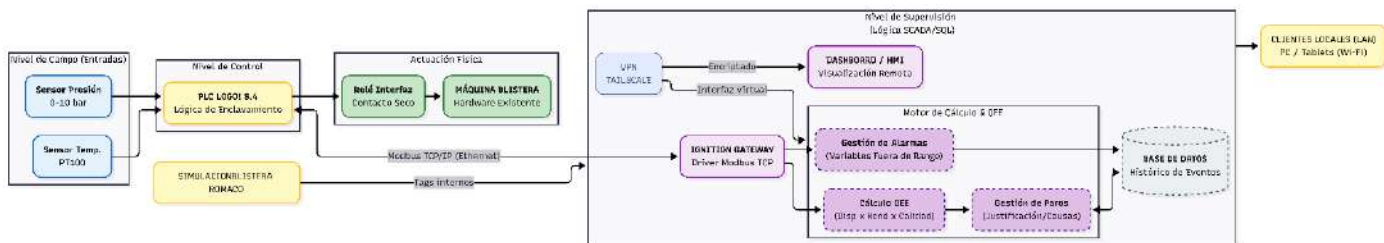


Figura 5.5.. Representación esquemática de los componentes del sistema

Anexo I. HMI del operador



Figura 5.6. Ventana de justificación de Paro

Anexo J. Vista de supervisión SCADA de la blistera

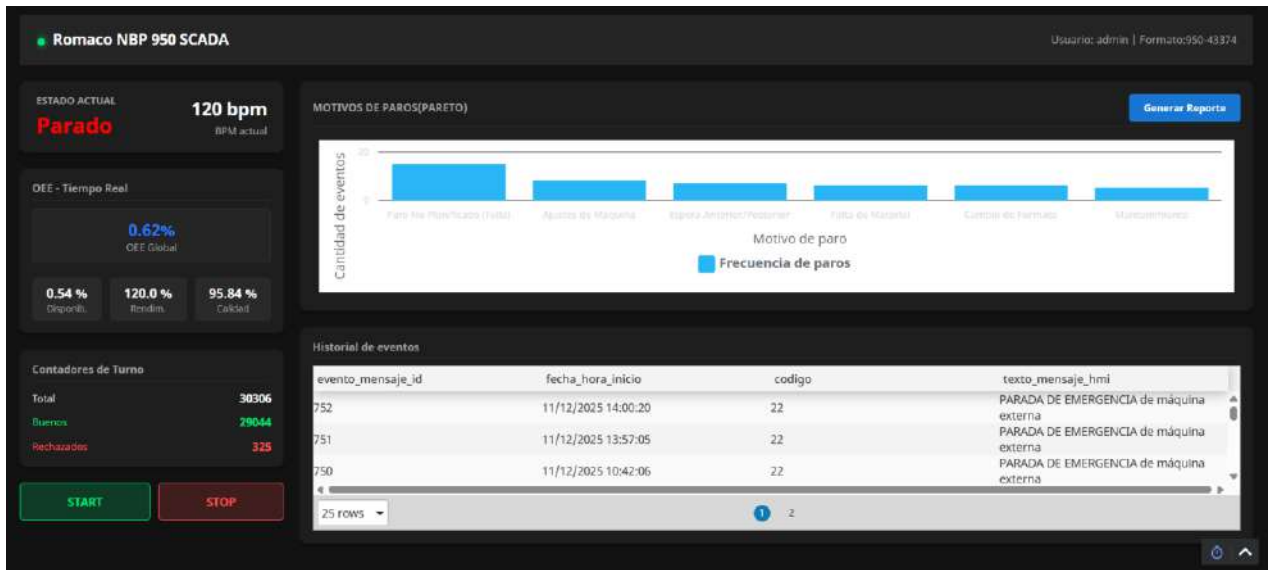


Figura 5.7.. Interfaz principal del administrador del sistema SCADA

Anexo K. Vista general esquemática de la blistera Romaco y ubicación de componentes principales

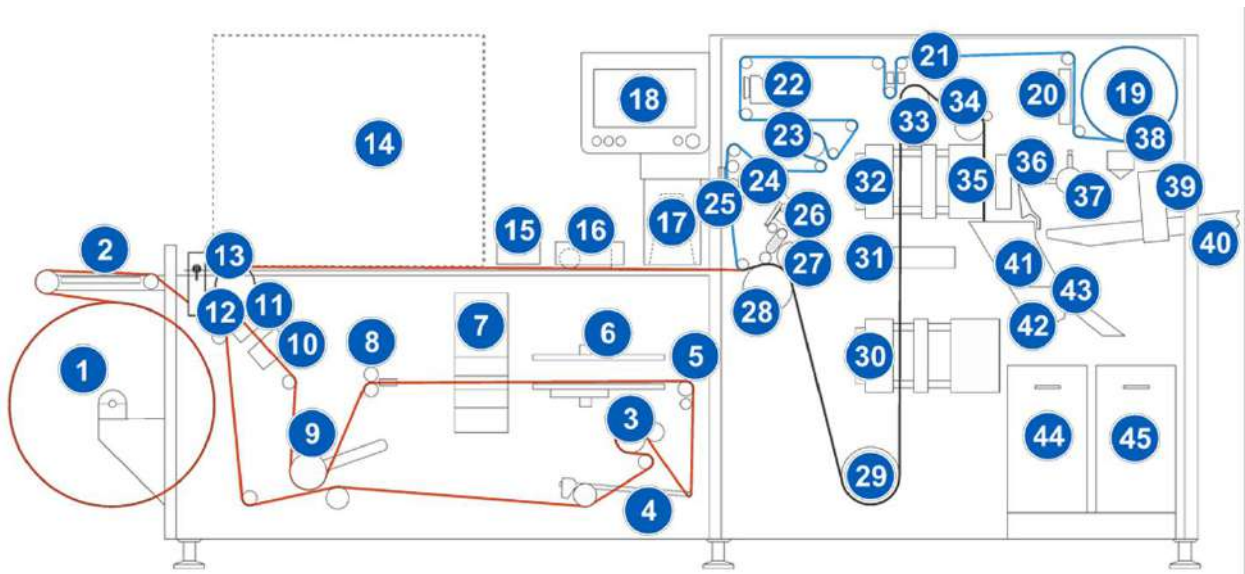


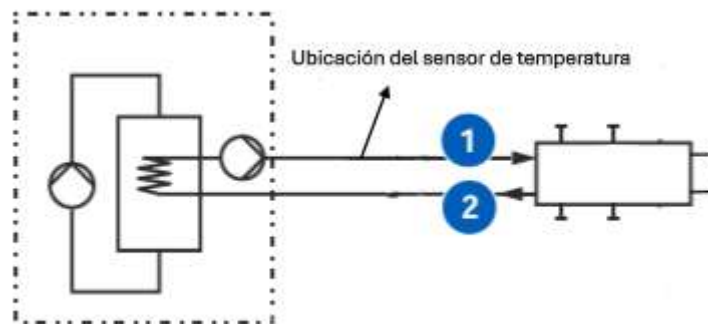
Figura 5.8. Vista general esquemática de la máquina

Anexo L. Desglose visual del OEE (Disponibilidad, Rendimiento y Calidad) y tiempos asociados de operación y pérdida



Figura 5.9.. Desglose visual de los tres factores determinantes de OEE.

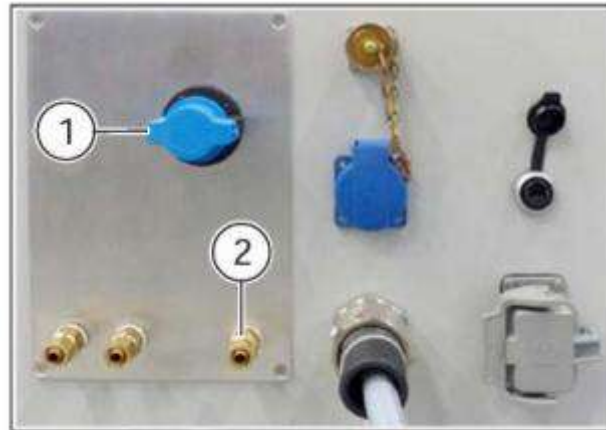
Anexo M. Conexiones de agua de refrigeración



- 1) Entrada de agua de refrigeración
- 2) Salida de agua de refrigeración

Figura 5.10.. Puntos de conexión del agua de refrigeración para instrumentación del servicio.

Anexo N. Conexiones de aire comprimido



- 1) Válvula de arranque
- 2) Conexión de aire comprimido

Figura 5.11. Puntos de conexión de aire comprimido

Anexo O. Diagrama de flujo del enclavamiento de arranque (start permissive) por presión de aire y temperatura de agua

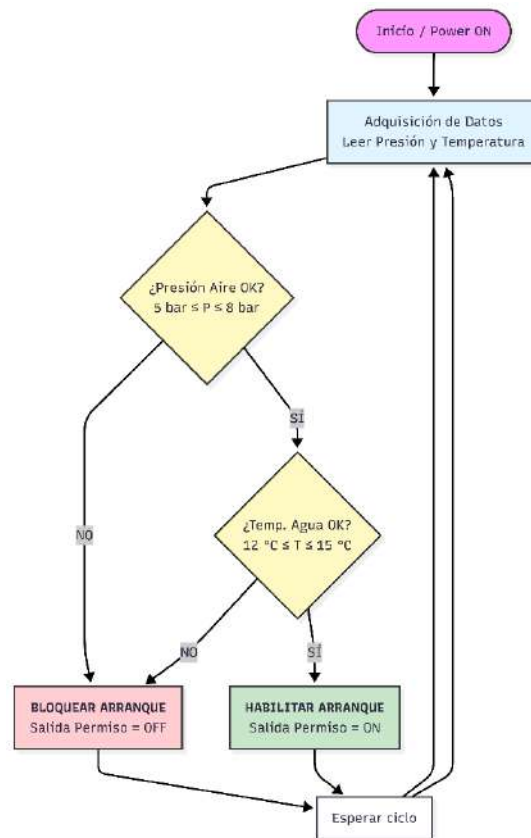


Figura 5.12. Diagrama de flujo del enclavamiento de arranque

Anexo P. Arquitectura de hardware propuesta del enclavamiento de arranque e integración con SCADA/SQL

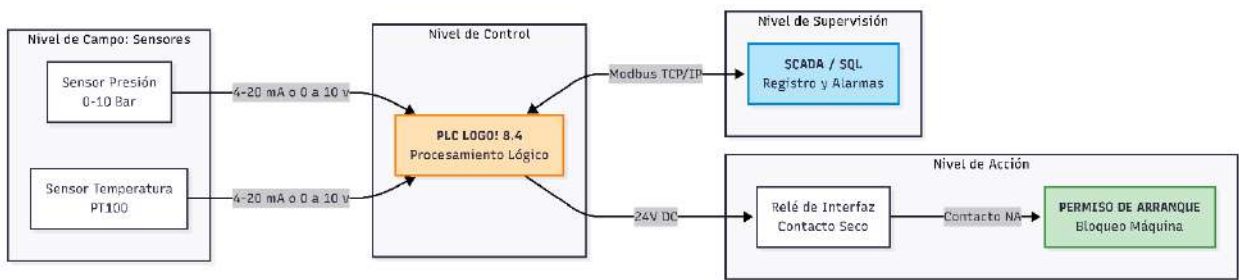


Figura 5.13. Arquitectura del enclavamiento de arranque

Anexo Q. Esquema de conexionado del Módulo de instrumentación electrónica

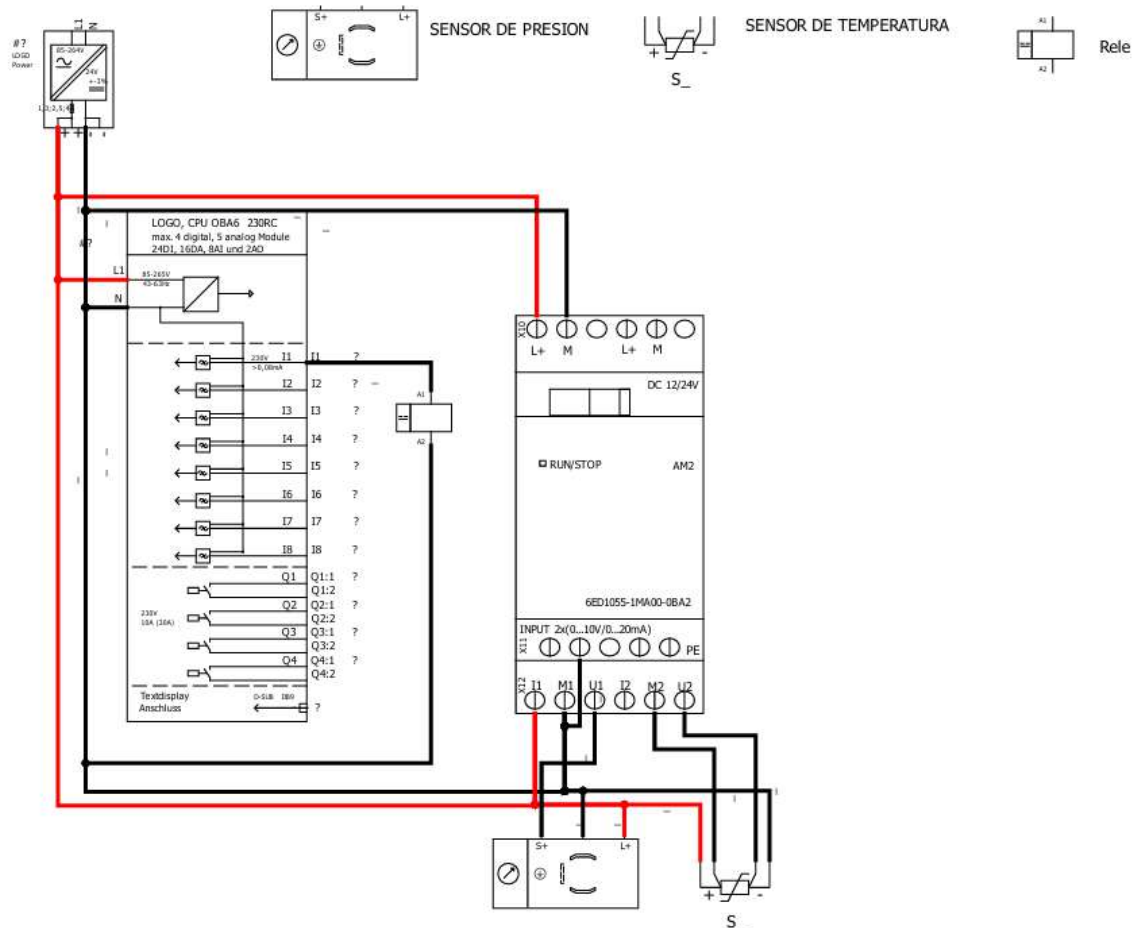


Figura 5.14. Esquema de conexionado: sensores 4-20 mA hacia módulo LOGO! AM2 y salida por relé

Anexo R. Lista de componentes y características

Anexo R.1. CPU LOGO! 8.4 24RCE

Referencia típica: **6ED1052-1MD08-0BA1** característica: displ. FA/E/S: 12/DC 24 V/relé, 8 DI (4 AI)/4 DO, mem. 400 bloques, modularmente ampliable, Ethernet, servidor web integrado, Datalog, variantes personalizadas de páginas web, variante estándar de tarjeta microSD para LOGO! Soft Comfort a partir de V8.4, versiones anteriores de proyectos ejecutables, conexión a la nube, MQTT en todas las unidades base LOGO! 8.4

Precio aprox.: \$150 – \$220



Anexo R.2. Módulo de expansión LOGO! AM2

Referencia típica: **6ED1055-1MA00-0BA2**

Alimentación	12/24VDC
Entradas	2 Análogas
Salidas	Ninguna
Rango de entrada	0 - 10 VDC 4 - 20 mA

Precio aprox.: \$80 – \$120



Anexo R.3. Sensor de presión wika A-10

Rango	0–10 bar
Señal de salida	4–20 mA (preferido)
Alimentación	24 VDC
Conexión	3 hilos (L+, S+, GND)
Rosca	G 1/4" o 1/4" NPT
Precisión	±0,5 % FS o mejor
Protección	IP65 o superior
Uso	Aire / agua / fluidos industrial



Precio aprox.: 80 -120 \$

Anexo R.4. Sensor RTD PT100 RS PRO de 4 hilos

elemento de detección de alta estabilidad con una salida de precisión

- Precisión/tolerancia de clase B conforme a IEC 60751
- Rango de medición de temperatura de -50 a +200 °C.
- 4 cables para lecturas precisas
- Los Cables se conectan a cualquier instrumento PT100 de 2 3 o 4 cables
- Funda rígida de acero inoxidable resistente a la corrosión 316 para proteger el elemento de detección
- El aislamiento Flexible de Teflón proporciona aislamiento térmico y eléctrico, resistencia a impactos y protege contra la humedad, la corrosión y sustancias químicas abrasivas incluso en temperaturas extremas



Precio aprox.: \$30 – \$70

Anexo R.5. Fuente de alimentación 24VDC

Descripción: Tensión de entrada: 100 VAC - 240VAC

Tensión de Salida: 24 VDC

Corriente de Salida (máx.) (A): 5



Precio aprox.: \$25 – \$45

Anexo R.6. RELE TRS 24VDC 1CO

Módulo de relé

- Número de contactos: 1
- Contacto conmutado AgNi
- Tensión de mando nominal: 24 V DC $\pm 20\%$
- Intensidad permanente: 6 A
- Conexión brida-tornillo



Precio aprox.: 15 -35\$